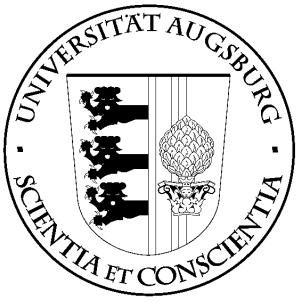
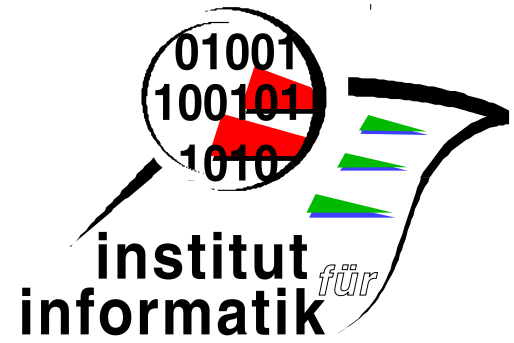
UNIVERSITÄT AUGSBURG



Lossless Audio Compression using Reversible Transforms

Jonghwa Kim

Report 2007-9 October 2007



INSTITUT FÜR INFORMATIK

D-86135 AUGSBURG

Copyright © Jonghwa Kim
Institut für Informatik
Universität Augsburg
D–86135 Augsburg, Germany
http://www.Informatik.Uni-Augsburg.DE
— all rights reserved —

Lossless Audio Compression using Reversible Transforms

October 10. 2007

Abstract

In this paper we developed lossless audio compression by using reversible transforms that map their coefficients as integers to integers. Three reversible transforms, S-transform, TS-transform, and SP-transform were implemented to decorrelate redundant signals in music samples. Golomb-Rice code combined with the Laplacian probability density function (PDF) was used to encode the decorrelated signal. Various audio samples excerpted from the SQAM-CD were tested by prototypical lossless compression system we developed and compression results are presented with comparing with the result obtained by using linear prediction method in our previous work. ...

1 Introduction

Despite tremendous growth of storage capacity and transmission bandwidth of the digital world today, the demand for higher-quality multimedia (more word size, more sample rate, and more channel) associated with audio, image, and video continues to outpace it. Hence the importance of data compression, as a key technology to allow efficient storage and transmission, is not likely to diminish. The general idea behind data compression is to remove the redundancy present in the data to find more compact representation. Two families of algorithms exist in the data compression. When the information can be exactly recovered from the bits, the source coding or compression is called *lossless*; otherwise, it is *lossy*. To achieve higher compression ratio, lossy algorithms remove the information that is not perceptible or transform in a way that the compressed signal comes close to the original. In this case we allow approximate representation of the original, instead of trying to represent the original exactly, and have only a modified version of the original after transmission. Lossless algorithms, however, respect the integrity of the original signal. After transmission and reconstruction an exact copy of the original signal is available.

Two main drawbacks are related with lossless audio compression. First, time varying compression ratio of lossless compression makes difficult to allocate a fixed bandwidth to transmitting data and complicates editing of compressed material. The second is its poor compression rate compared to the lossy case. The lossless audio compression usually achieves a compression ratio of 2 to 3 without any loss of quality, while lossy compression achieves a compression ratio of 8 to 40 or higher. With the higher compression ratio, however, the lossy audio compression is very objectionable in high fidelity audio compression applications because of unexpected artifacts introduced even by the most heavily engineered schemes using perceptual auditory models. It would be more critical if the audio signal undergoes multiple encoding/decoding operations.

Most transforms result in real-valued coefficients with an infinite precision and main compression effect arises during quantization and encoding of these coefficients. Consequently the traditional transform coding is a lossy compression method. In order to realize a lossless compression based on the transformation, we therefore need to develop a transform method that maps the coefficients as integers to integers. The construction of a reversible transform is relatively simple. By using appropriate quantization or rounding, any linear transform can be modified so that it can be computed using finite-precision arithmetic with preserving lossless invertibility. However, an efficient construction of reversible transform is by no means a straightforward task, although the idea behind generating the transform is easily stated. For example, due to the quantization error the resulting reversible transform is generally nonlinear and only serves to approximate the linear transform from which it was derived. If the reversible transform fails to mimic the

behavior of its parent transform, desirable property of the parent transform will likely be lost and poor results will be obtained. So the key consideration in the design process is to construct efficient reversible transforms that successfully approximate their parent transform.

Theoretically, linear transformations and perfect reconstruction (PR) filter banks are losslessly invertible but this invertibility can be guaranteed only if the transform and its inverse transform use an exact arithmetic. It is because of the fact that output of most non-singular linear transforms consists of rational or real numbers due to their floating point coefficients. Even when the input data consist of sequences of integer samples (this is the case for audio, image, and video signal in digital world), the outputs no longer consist of integers. Furthermore, finite-precision arithmetic is employed to perform the transforms, and such arithmetic is inherently inexact due to errors introduced by system rounding. So it means that the transforms, which are losslessly invertible in exact arithmetic, are lossy in a strict sense because of finite-precision arithmetic of computer, for example. Furthermore, the quantization of the transform coefficients in such compression system would likely increase the loss of information.

In this paper, we discuss the lossless audio signal compression using reversible transforms. We first review the various reversible transforms that can be used for lossless audio compression, such as S-transform, RTS transform, and S+P transform. We test the transforms in decorrelation stage of a prototypical lossless audio compression scheme and compare the performance of the system with the linear prediction method reported in our previous work [1].

2 Reversible subband transforms

2.1 S-Transform

A classic example of a reversible transform is the sequential transform (S-transform) proposed in [2] [3] which has become quite popular for lossless signal compression (especially for image compression). A sequence of random integers $x[n]$ with length N, can be perfectly represented by the two sequences with length $N/2$ defined by

$$\begin{aligned} lp[n] &= \lfloor (x[2n] + x[2n+1])/2 \rfloor \,, \\ hp[n] &= x[2n] - x[2n+1] \,, \end{aligned} \tag{2.1}$$

where the floor $\lfloor \cdot \rfloor$ represents a maximum integer not exceeding a real number x. This is so called (2×2)[1] S-transform. Several slightly different definitions of this transform exist in the literature. In fact, the S-transform is a nonlinear approximation to a scaled version of the Haar transform. The Haar transform itself is one of the simplest two-channel subband transforms. Transfer functions of Haar transform are defined as following,

$$\begin{aligned} H(z) &= (1+z)/2 \,, & G(z) &= 1 - z \,, \\ \widetilde{H}(z) &= 1 + z \,, & \widetilde{G}(z) &= (1-z)/2 \,. \end{aligned}$$

This transform is a scaled version of an orthogonal transform. Therefore, the coefficients of the S-transform must be weighted on a per subband basis in order to approximate an orthogonal transform. The inverse transformation of (2.1) is

$$x[2n] = lp[n] + \lfloor (hp[n] + 1)/2 \rfloor \,, \tag{2.2}$$

$$x[2n+1] = x[2n] - hp[n] \,. \tag{2.3}$$

The idea behind the reversibility of the S-transform is the observation of the facts; the sum and the difference of two integers are sufficient knowledge to recover the numbers and have the same parity, i.e., they share the same least significant bit. Therefore the division by 2 (or a shift right by 1) in Eq. (2.1) eliminates a redundant least significant bit. Figure 2.1 shows a block diagram of the transform.

In fact, this transformation is equal to subband decomposition, except for the truncation procedure. Therefore $lp[n]$ and $hp[n]$ are the lowpass and highpass components, respectively. The main idea behind

[1] Let the numbers of taps of the lowpass filter and that of the highpass filter be $(n \times m)$.

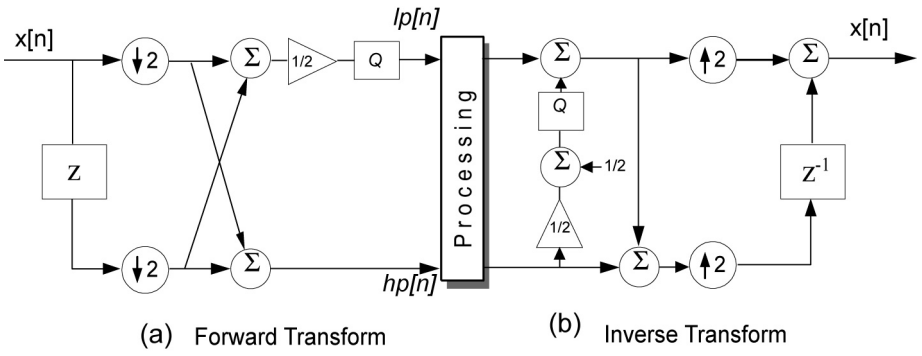


Figure 2.1: Filtering structure of S-transform ($Q(x) = \lfloor x \rfloor$).

this representation is that, if the correlation coefficient of $x[2n]$ and $x[2n+1]$ is larger than $1/3$, then the average variance of $lp[n]$ and $hp[n]$ is smaller than the variance of $x[n]$. In this case, $hp[n]$ normally has small variance, while the variance of $lp[n]$ is approximately equal to the variance of $x[n]$. The main advantage of S-transform is also that it is easy to find the truncation allowing the PR system. Moreover, there is no data expansion, i.e., it uses use the same number of samples of the original signal. Unfortunately, the ideas on which the S-transform is based do not generalize to transforms using more complicated relationships that simple pairwise sums and differences. Consequently the S-transform does not provide any further insight into how other classes of reversible transforms might be constructed.

2.2 Reversible TS-Transform

To design a symmetric short kernel filter (SSKF), Gall and Tabatabei in [4] used a factorization of a product filter into two linear phase low-pass components. These correspond to the lowpass analysis and synthesis filters. By using the quadrature mirror filter (QMF) properties the highpass filters are derived. In their most important example, the following product filter is factorized

$$P(z) = \tfrac{1}{16}(1 + z^{-1})^3(-1 + 3z^{-1} + 3z^{-2} - z^{-3})\,. \tag{2.4}$$

Its two factorized versions are given in [4]

$$\begin{aligned} P_1(z) &= [\tfrac{1}{4}(1 + z^{-1})^3] \times [\tfrac{1}{4}(-1 + 3z^{-1} + 3z^{-2} - z^{-3})] \\ P_2(z) &= [\tfrac{1}{2}(1 + z^{-1})^3] \times [\tfrac{1}{8}(1 + z^{-1})(-1 + 3z^{-1} + 3z^{-2} - z^{-3})]\,. \end{aligned} \tag{2.5}$$

Using this factorization method, another version is considered in [5],

$$P_3(z) = [\tfrac{1}{2}(1 + z^{-1})] \times [\tfrac{1}{8}(1 + z^{-1})^2(-1 + 3z^{-1} + 3z^{-2} - z^{-3})]\,. \tag{2.6}$$

From the third version, a (2×6) PR subband filter can be defined with following filter coefficients

$$\begin{aligned} h &= \frac{1}{\sqrt{2}}(1, 1) \\ g &= \frac{1}{8\sqrt{2}}(-1, -1, 8, -8, 1, 1)\,. \end{aligned} \tag{2.7}$$

This transform is called *TS(two-six)-transform* in the literature [6]. A reversible version (RTS) of Eq. (2.7) is proposed in [7],

$$lp[n] = \left\lfloor \frac{x[2n] + x[2n+1]}{2} \right\rfloor \tag{2.8}$$

$$\begin{aligned} hp[n] = \Bigg\lfloor \frac{1}{4} \Bigg(&- \left\lfloor \frac{(x[2n] + x[2n+1])}{2} \right\rfloor + 4(x[2n+2] - x[2n+3]) \\ &+ \left\lfloor \frac{(x[2n+4] + x[2n+5])}{2} \right\rfloor \Bigg) \Bigg\rfloor \end{aligned} \tag{2.9}$$

The expression for hp can be simplified and written with the use of lp, and the integer division by 4 can be rounded by adding a 2 to the numerator. These result in,

$$lp[n] = \left\lfloor \frac{x[2n] + x[2n+1]}{2} \right\rfloor \tag{2.10}$$

$$hp[n] = x[2n+2] - x[2n+3] + \left\lfloor \frac{-lp[n] + lp[n+2] + 2}{4} \right\rfloor \tag{2.11}$$

The inverse transform of the RTS-transform is quite simple,

$$x[2n] = lp[n] + \left\lfloor \frac{s[n] + 1}{2} \right\rfloor \tag{2.12}$$

$$x[2n+1] = lp[n] - \left\lfloor \frac{s[n]}{2} \right\rfloor \tag{2.13}$$

where

$$s[n] = hp[n-1] - \left\lfloor \frac{-lp[n-1] + lp[n+1] + 2}{4} \right\rfloor \tag{2.14}$$

As the case of S-transform, the lowpass signal of RTS-transform has the same range of values as the input signal. Particularly, this property is important in a pyramid system where the lowpass signal is successively decomposed. Note that there is no systemic error due to rounding in the integer implementation of the transform, so all error in a lossy system can be controlled by quantization.

2.3 S+P Transform

S+P transform (S-transform + prediction) is a reversible transform that maps integers to integers and is parameterized by two sets of filter coefficients, initially proposed by Said and Perlman [8]. This transform is a further refinement of the S-transform where the S-transformed highpass output hp_o is replaced by the difference between the $hp_o[n]$ and the estimate $\hat{hp}[n]$ obtained using the prediction;

$$lp[n] = \lfloor \tfrac{1}{2}(x[2n] + x[2n+1]) \rfloor \tag{2.15}$$

$$hp[n] = hp_o[n] - \lfloor \hat{hp}[n] + 1/2 \rfloor \tag{2.16}$$

$$\tag{2.17}$$

where

$$hp_o[n] = x[2n] - x[2n-1] \tag{2.18}$$

$$\hat{hp}[n] = \sum_{i=L_0}^{L_1} \alpha_i \Delta lp[n+i] - \sum_{j=1}^{H} \beta_j \, hp_o[n+j] \,, \tag{2.19}$$

where $\Delta lp[n] = lp[n-1] - lp[n]$. The use of $\Delta lp[n]$ instead of $lp[n]$ allows to have zero-mean estimation terms, and thus there is no need to subtract the mean from $x[n]$. Note that the index i can be negative because $lp[n]$ is not replaced by a prediction error. The optimal predictor coefficients α and β can be found by solving the Yule-Walker equations. The inverse transform uses $lp[n]$ and $hp[n]$ to reconstruct the original input signal $x[n]$ as given by

$$x[2n] = lp[n] + \lfloor \tfrac{1}{2}(hp_o[n] + 1) \rfloor \tag{2.20}$$

$$x[2n+1] = x[2n] - hp_o[n] \,, \tag{2.21}$$

where

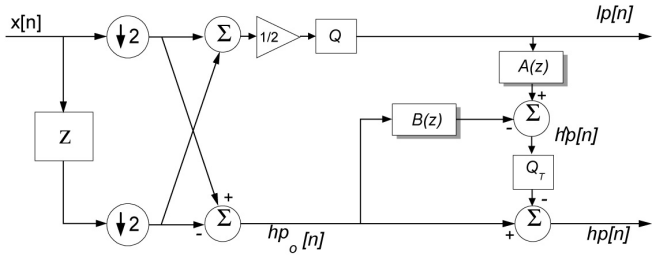
$$hp_o[n] = hp[n] + \lfloor \hat{hp}[n] + \tfrac{1}{2} \rfloor \,, \tag{2.22}$$

and $\hat{hp}[n]$ is as given above.

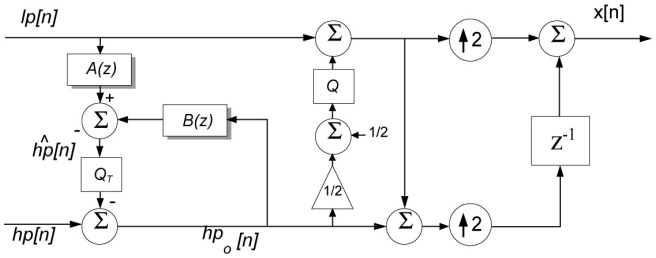
Type	α_{-1}	α_0	α_1	β_1
A	0	1/4	1/4	0
B	0	2/8	3/8	2/8
C	-1/16	4/16	8/16	6/16

Table 1: S+P transform predictor coefficients

In Table 1[2], three sets of predictor coefficients are listed that have been suggested in [9]. The predictor A in the table has the smallest computational complexity and yields a reversible version of the TS-transform. In the degenerate case where all of the predictor coefficients are zero ($\alpha_i = \beta_i = 0$), the S-transform is obtained. The filtering structure for the S+P transform is shown in Figure 2.2, where $A(z)$, $B(z)$, $Q_T(x)$,



(a) Forward Transform



(b) Inverse Transform

Figure 2.2: Structure of S+P transform. (a) Forward transform, (b) Inverse transform

and $Q(x)$ are defined as

$$A(z) = (1 + z^{-1}) \sum_{i=L_0}^{L_1} \alpha_i z^{-1}, \quad B(z) = \sum_{j=1}^{H} \beta_j z^{-j}, \tag{2.23}$$

$$Q(x) = \lfloor x \rfloor, \quad Q_T(x) = \lfloor x + \tfrac{1}{2} \rfloor . \tag{2.24}$$

Note that the first part of the forward transform structure and last part of the inverse transform structure are nothing more than the S-transform. If we disregard the effects of the truncation in the S+P transform, we have a linear subband transform that corresponds to a QMF bank having analysis filters with transfer functions,

$$\begin{aligned} H(z) &= \tfrac{1}{2}(1 + z), \\ G(z) &= -\tfrac{1}{2}(1 + z)A(z^2) + (1 - z)[1 + B(z^2)] \end{aligned} \tag{2.25}$$

So the S+P transform does not directly approximate an orthogonal or near-orthogonal transform. By weighting the transform coefficients associated with each subband by an appropriate constant, a near-orthogonal transform can be obtained. To handle finite-length signal, it should be assumed that the signal

[2]In image compression applications, the predictor B is the best suited for natural images and the predictor C is for very smooth medical images.

is defined for $n = 0, 1, 2, \ldots, N-1$ where N is even. This assumption is required so that the input signal is split into two polyphase components. If the signal is not of even length, the simple solution is to pad the signal by one sample. The S+P transform is typically applied in a pyramid fashion such that the lowpass signal is successively decomposed. In this case, we have a reversible wavelet transform.

3 Experiment

3.1 Test audio material

Nine audio materials are chosen for our experiment; six from SQAM-CD [10][3] and two from published music CDs (see Table 2). All materials are sampled at 44.1kHz, 16 bits step size, and stereo channel (except for speech).

Nr.	Length	Description
1	1:11	SQAM Track 8, Violin, Arpeggio and Melodious Phrase
2	0:46	SQAM Track 13, Flute, Arpeggio and Melodious Phrase
3	0:21	SQAM Track 53, Female Speech, German (Mono)
4	1:32	SQAM Track 60, Piano, Schubert
5	1:22	SQAM Track 67, Wind Ensemble, Mozart
6	0:33	SQAM Track 69, ABBA, *Pop*
7	0:21	SQAM Track 70, Eddie Rabbitt, *Country*
8	0:29	Def Leppard "Adrenalize", Track 1 "Let's get rocked", Bludgeon Riffola Ltd, *Metal Rock*
9	0:29	Stan Getz "The Artistry of Stan Getz", Track 10 "Litha", Polygram Records, *Soft Jazz*

Table 2: Description of test audio materials

3.2 Design of lossless compression system

We tested the three reversible transforms, S-transform, TS-transform, and S+P transform, by integrating them into decorrelation stage of a prototype audio compression system developed in our previous work [1] (Fig. 3.1).

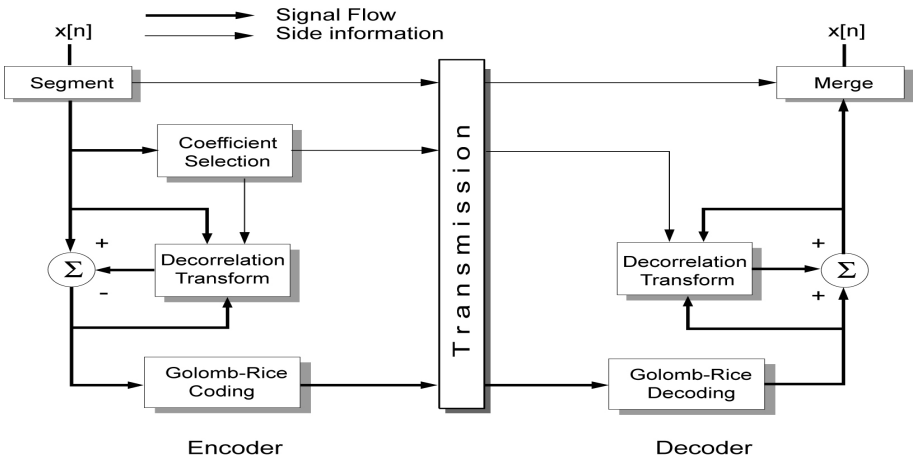


Figure 3.1: Block diagram of lossless audio compression system.

[3] SQAM(Sound Quality Assessment Material), European Broadcasting Union

Materials	S	SPB	TS	LP-IIR[1]	Gzip
Nr. 1 violin, solo	9.12[2] (1.74[3])	8.92 (1.79)	8.91 (1.78)	**8.69** (**1.84**)	12.05
Nr. 2 flute, solo	7.54 (2.12)	**7.49** (**2.14**)	7.67 (2.09)	7.52 (2.13)	11.04
Nr. 3 speech, fem.	7.32 (2.19)	**7.30** (**2.19**)	7.35 (2.18)	7.42 (2.16)	8.62
Nr. 4 piano, solo	5.95 (2.69)	5.82 (2.75)	5.96 (2.68)	**5.81** (**2.75**)	10.43
Nr. 5 classic, orch.	7.50 (2.13)	**7.36** (**2.17**)	7.46 (2.14)	7.43 (2.15)	12.79
Nr. 6 pop, abba	8.35 (1.92)	**8.29** (**1.93**)	8.97 (1.78)	8.32 (1.92)	11.76
Nr. 7 country	**7.42** (**2.16**)	7.53 (2.12)	7.61 (2.10)	7.55 (2.12)	9.65
Nr. 8 rock, metal	**12.44** (**1.29**)	12.49 (1.28)	12.59 (1.27)	12.47 (1.28)	14.86
Nr. 9 jazz, soft	9.54 (1.68)	**9.39** (**1.70**)	9.47 (1.69)	9.44 (1.69)	13.93
Average	8.35 (1.92)	**8.29** (**1.93**)	8.44 (1.90)	**8.29** (**1.93**)	11.68

[1] Results by using IIR linear prediction in the previous work [1]

Table 3: Test compression results with compressed bit rate (bits/sample)[2] and compression ratio (original/compressed)[3]

For the entropy coding following the decorrelation stage in the compression system, we used Golomb-Rice code with fixed parameter k. The estimation of an optimal parameter k is linearly related to the variance of signal. The Golomb-Rice code provides a good approximation to the distribution of the prediction residual samples, because it is optimized for a block of signals having a Laplacian-like probability density (double-sided exponential distribution). For S+P transform we used single level decomposition with the coefficients of the type B in the Table 1, i.e.

$$\alpha_0 = \frac{2}{8},\ \alpha_1 = \frac{3}{8},\ \beta_1 = \frac{2}{8}$$
$$\hat{hp}[n] = \frac{1}{8}\{2(\Delta lp[n] + \Delta lp[n+1] - hp_o[n+1]) + \Delta lp[n+1]\} \tag{3.1}$$

3.3 Results

Table 3 shows the compression results in bit rates. It turned out that there is no single reversible transform that provides superior compression performance for all types of audio materials. Furthermore the effectiveness of the transform depends strongly on the content of audio signals to be compressed. For the smooth audio signals such as speech, classic, and soft pop music, the SPB was most efficient decorrelation method, while S-transform, which requires lowest computational complexity, performs best for audio signals with large dynamic range and relatively stronger treble energy such as rock and metal music. The compression result from the IIR linear prediction developed in the previous work [1] is also showed in the table. Overall it turned out that SPB transform performs as good as the IIR linear prediction for all audio signals. Particularly, for single instrument music samples, the IIR prediction seems to be the best choice and the SPB-transform for the others.

4 Conclusion and future work

For lossless audio compression we tested three reversible transforms, S-transform, SPB-transform, and TS-transform to decorrelate redundant signals in music samples. Regardless of which decorrelation method used, simple linear prediction or complex reversible transform, the compression result presented in this work is not sufficient to object to the generally accepted opinion that compression ratio of 3 to 4 would be the limit of lossless audio compression. However, there are still some issues that could help overcome the limitation in lossless transform coding. For example, parametric signal segmentation could improve the performance of decorrelation. Since signal content of a music sample is quasi-periodic with repeating rhythm and harmonic progress, one can consider an adaptive block-length estimation according to the rhythmic/beat period, in order to exploit cross-correlation between these blocks. The performance of the SPB-transform, which showed best compression performance (especially for dynamic music samples) in our experiment, can be further improved. In the experiment we implemented just single level decomposition using the transform with a common set of coefficients. This can be extended by employing a multilevel decomposition scheme based on low-passed signal parts. In this case, the order of coefficients should be optimized depending on the number of decomposition level. Since such scheme approaches to a wavelet-like decomposition, it might easily be adapted to the type of music samples.

References

[1] J. Kim, "Predictive modeling for lossless audio compression," Tech. Rep. TR-Nr. 2004-05, Institut für Informatik, Universität Augsburg, Germany, 2004.

[2] K. Irie and R. Kishimoto, "A study on the perfect reconstructive subband coding," *IEEE Trans. Circuits and Syst. Video Technol.*, vol. 1, pp. 42–48, Mar. 1991.

[3] I. Shah, O. Akiwumi-Assani, and B. Johnson, "A chip set for lossless image compression," *IEEE Journal Solid-State Circuits*, vol. 26, no. 3, pp. 237–244, 1991.

[4] D. L. Gall and A. Tabatabai, "Subband coding of digital images using symmetric short kernel filters and arithmetic coding techniques," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc.*, (New York), pp. 761–765, 1988.

[5] D. Speck, "Low-complexity subband coding for image compression," tech. rep., Computer Research Laboratory, University of California at Santa Cruz, May 1993.

[6] J. Villasenor, B. Belzer, and J. Liao, "Filter evaluation and selection in wavelet image compression," in *IEEE Data Compression Conference*, (Snowbird, Utah), 1994.

[7] A. Zandi, J. Allen, E. Schwartz, and M. Boliek, "CREW: Compression with reversible embedded wavelets," in *IEEE Data Compression Conference*, (Snowbird, Utah), Mar. 1995.

[8] A. Said and W. Pearlman, "Reversible image compression via multiresolution representation and predictive coding," in *Proc. SPIE in Visual Commun. and Image Proc.*, vol. 2094, pp. 664–674, Nov. 1993.

[9] A. Said and W. A. Pearlman, "An image multiresolution representation for lossless and lossy compression," *IEEE Trans. on Image Processing*, vol. 5, pp. 1303–1310, Sept. 1996.

[10] "Sound Quality Assessment Material CD." Technical Centre of the European Broadcasting Union (EBU), 1988. 4222042.