Advances in Computer Science and Engineering
Volume 9, Number 2, 2012, Pages 151-159
Published Online: December 2012
Available online at http://pphmj.com/journals/acse.htm
Published by Pushpa Publishing House, Allahabad, INDIA

DWT BASED COMPRESSION ENCODER FOR VLSI APPLICATIONS

Juuso T. Olkkonen*, Kari Jarvinen and Hannu Olkkonen

*VTT Technical Research Centre of Finland
P. O. Box 1000
02044 Espoo, Finland
e-mail: juuso.olkkonen@vtt.fi

Department of Applied Physics
University of Eastern Finland
P. O. Box 1627
70211 Kuopio, Finland
e-mail: hannu.olkkonen@uef.fi

Abstract

Multi-channel sensor networks and wireless data transmission methods are becoming widely used. This requires effective data compression encoding and the construction of the high-fidelity VLSI transceivers. In this paper, we introduce a tree structured discrete wavelet transform (DWT) for design of the real-time signal compression encoder. The impulse response of the symmetric half-band wavelet filter approaches the sinc-interpolation kernel, which has optimal compression efficiency. The compression encoder is implemented using the lifting wavelet network, which can be efficiently carried out in VLSI circuits.

Keywords and phrases: wavelet transform, compression encoder, pyramid algorithms, VLSI.
This work was supported by the National Technology Agency of Finland (TEKES).
Received October 31, 2012

1. Introduction

There is a current need for wireless transceivers capable in transmission of the high fidelity multi-channel signals. This requires effective real-time compression methods, which can be implemented in VLSI environment. In conventional pyramidal compression [1, 2] the data is low-pass filtered and the difference between the original and approximated signal has been coded. An advantage of the data pyramids is that the low-pass filters can be designed so that they maximize the compression efficiency. The disadvantage of the pyramidal algorithms is data redundancy, which increases by a factor of two [2].

Recently the discrete wavelet transform (DWT) algorithms have gained a main role in signal and image compression. The original DWT structures were based on the compactly supported conjugate quadrature filters (CQFs) [3, 4]. However, an obstacle in CQFs is related to the nonlinear phase effects such as image blurring and spatial dislocations in multi-scale analyses. On the contrary, in biorthogonal discrete wavelet transform (BDWT) the scaling and wavelet filters are symmetric and linear phase. Usually the BDWTs are constructed by a ladder-type network called *lifting scheme* [5]. Efficient lifting BDWT structures have been developed for VLSI environment [6]. Many BDWT based data and image processing tools has outperformed the previous compression techniques, including the discrete cosine transform (DCT), which has been the basic compression tool in several areas of signal and image analysis. For example, in JPEG2000 Standard [7], the DCT has been replaced by the lifting BDWTs.

In this paper, we describe the signal compression encoder based on the tree structured wavelet transform. The design of the wavelet filter is based on the symmetric half-band filter (HBF), which has interpolating characteristics. The frequency response of HBF approaches asymptotically the sinc-interpolation kernel, which has optimal compression efficiency. In the following representation we first review the tree structured DWT and the structure of the symmetric HBF. Then we describe the design of the symmetric half-band wavelet filter and the implementation of the compression encoder using the lifting network.

2. DWT Based Compression Method

2.1. Tree structured discrete wavelet transform

The tree structured DWT (Figure 1) consists of the low-pass scaling $H_0(z)$ and high-pass wavelet $H_1(z)$ filters and decimators. The low-pass filtered scaling coefficients are further fed to the next scale. The transform sequence consists of the wavelet coefficient sequences $w_1[n]$, $w_2[n]$, $w_3[n]$, ..., $w_n[n]$ and one scaling coefficient sequence $s_n[n]$. If the input data comprises of N data points, then the length of the transformed sequence is $N/2 + N/4 + \cdots + 2N/2^n = N$. This means that the redundancy of the tree structured DWT algorithm is one.

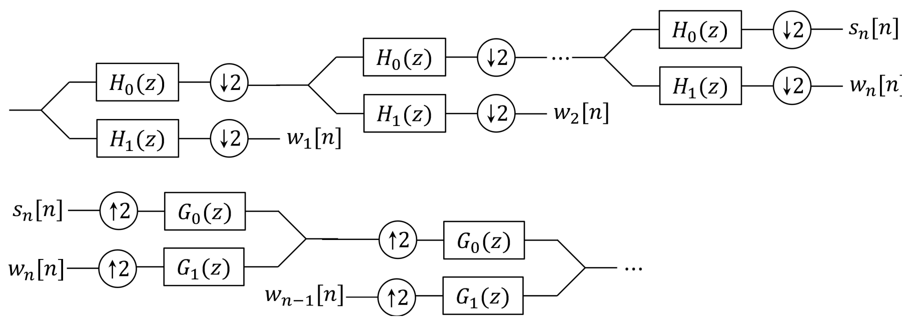

Figure 1. The analysis and synthesis parts of the tree structured discrete wavelet transform.

The reconstruction of the signal is carried out by the low-pass $G_0(z)$ and high-pass $G_1(z)$ synthesis filters (Figure 1). The DWT filter bank is related to the analysis/synthesis equations [8]

$$H_0(z)G_0(z) + H_1(z)G_1(z) = 2z^{-k},$$
$$H_0(-z)G_0(z) + H_1(-z)G_1(z) = 0. \qquad (1)$$

Equations (1) define the perfect reconstruction (PR) condition of the DWT filter bank. From the second equation we obtain the relations for the reconstruction filters $G_0(z) = H_1(-z)$ and $G_1(z) = -H_0(-z)$. The impulse response of the scaling filter $H_0(z)$ should obey Daubechies regulatory condition [4]

$$\sum_{n=0}^{N} (-1)^n n^m h_0[n] = 0; \quad m = 0, 1, ..., M - 1; \quad N \text{ odd.} \tag{2}$$

This moment relation implies that $H_0(z)$ contains Mth-order zero at $z = -1$ $(\omega = \pi)$, where M denotes the number of vanishing moments.

2.2. Symmetric half-band filter

The general structure for the symmetric half-band filter (HBF) is for k odd

$$H(z) = z^{-k} + B(z^2), \tag{3}$$

where $B(z^2)$ is a symmetric polynomial in z^{-2}. The impulse response of the HBF contains only one odd point. For example, we may parameterize the eleven point HBF impulse response as $h[n] = [c\,0\,b\,0\,a\,1\,a\,0\,b\,0\,c]$, which has three adjustable parameters.

2.3. Design of the symmetric half-band wavelet filter

The compression efficiency of the tree structured DWT improves when the frequency response of the high-pass wavelet filter approaches the frequency response of the sinc-function, which has HBF structure. However, the impulse response of the sinc-interpolator is infinite, which prolongs the computation time. In this work we propose the compactly supported HBF prototype (3) as a wavelet filter in the compression encoder. The length of the wavelet filter should be minimized to follow the rapid changes in the signal. Hence, we select seven point HBF prototype as a wavelet filter, which has the impulse response

$$h_1[n] = [b\,0\,a\,1\,a\,0\,b] \tag{4}$$

containing two adjustable parameters. In our pervious work we have introduced the modified regulatory condition for computation of the parameters of the wavelet filter [6]

$$\sum_{n=0}^{N} n^m h_1[n] = 0; \; m = 0, 1, ..., M - 1; \; N \text{ odd.} \tag{5}$$

This relation implies that $H_1(z)$ contains *M*th-order zero at $z = 1$ ($\omega = 0$), where *M* is the number of vanishing moments. Writing (5) for the prototype filter (4) we obtain two equations $2a + 2b + 1 = 0$ and $20a + 36b + 9 = 0$, which gives the solution $a = -9/16$ and $b = 1/16$. The wavelet filter (4) has the *z*-transform

$$H_1(z) = (1 - z^{-1})^4(1 + 4z^{-1} + z^{-2})/16 \tag{6}$$

having fourth order root at $z = 1$. The wavelet filter (6) can be effectively realized in the HBF form

$$H_1(z) = z^{-3} - A(z^2);\ A(z^2) = (-1 + 9z^{-2} + 9z^{-4} - z^{-6})/16. \tag{7}$$

2.4. Lifting wavelet transform architecture

The simplest filter bank obeying the PR condition (1) is (for *k* odd) $H_0(z) = 1$, $H_1(z) = z^{-k}$, $G_0(z) = z^{-k}$ and $G_1(z) = 1$. Sweldens [6] has observed that if the scaling $H_0(z)$ and wavelet $H_1(z)$ filters obey the PR condition, the filter

$$H_{1L}(z) = H_1(z) + H_0(z)D(z^2) \tag{8}$$

obeys also the PR condition. $H_{1L}(z)$ is called as the *lifted wavelet filter* and the $D(z^2)$ as the down lifting filter (Figure 2). Correspondingly, via the uplifting filter $U(z^2)$ we obtain the lifted scaling filter

$$H_{0L}(z) = H_0(z) + [H_1(z) + H_0(z)D(z^2)]U(z^2). \tag{9}$$

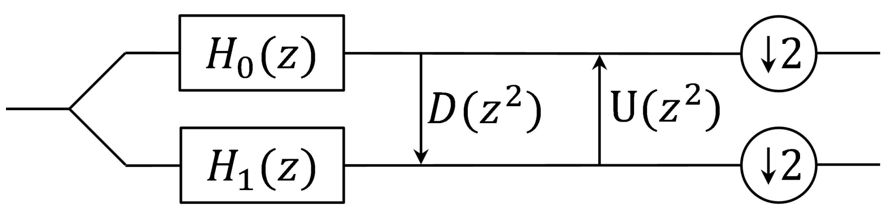

Figure 2. The lifting wavelet transform.

By inserting $H_0(z) = 1$, $H_1(z) = z^{-3}$ and $D(z^2) = -A(z^2)$ in (9) and using the Daubechies regulatory condition (2), we obtain a solution for the lifted scaling filter

$$H_{0L}(z) = (1+z^{-1})^2(1-2z^{-1}-5z^{-2}+28z^{-3}-5z^{-4}-2z^{-5}+z^{-6})/64 \quad (10)$$

which has the second order zero at $z = -1$. The impulse response of the lifted scaling filter is

$$h_{0L}[n] = [1\ \ 0\ \ -8\ \ 16\ \ 46\ \ 16\ \ -8\ \ 0\ \ 1]/64; \quad (11)$$

which is not, however, HBF structured. The compression encoder can be effectively realized in VLSI environment using the equivalence

$$[H(z^2)]_{\downarrow 2} \equiv (\downarrow 2)H(z) \quad (12)$$

which enables the removal of the lifting filters after the decimation operators (Figure 3). The structure of the compression encoder using the simplified lifting scheme is described in Figure 4.

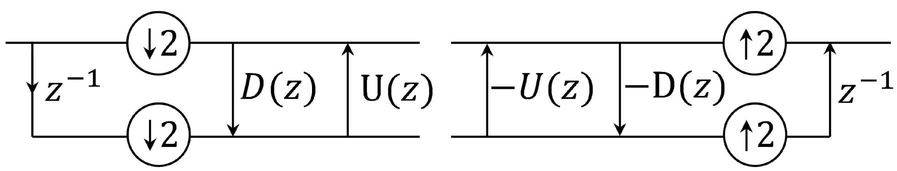

Figure 3. The analysis and synthesis parts of the simplified lifting wavelet transform.

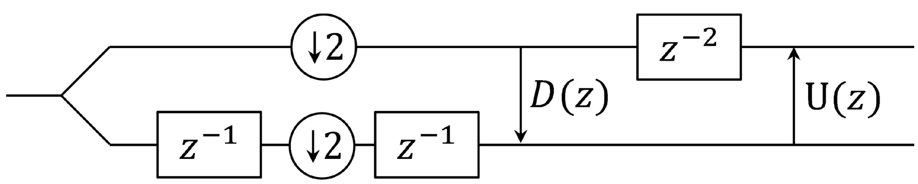

Figure 4. The structure of the compression encoder. The downlifting filter $D(z) = (1-9z^{-1}-9z^{-2}+z^{-3})/16$ and the uplifting filter $U(z) = (1+z^{-1})/4$.

2.5. Analysis of the data compression algorithm

The tree structured wavelet transform produces the wavelet coefficient sequences $w_i[n]$ (Figure 1). By writing the input signal using polyphase components

$$X(z) = X_e(z^2) + z^{-1}X_0(z^2), \tag{13}$$

where $X_e(z)$ and $X_0(z)$ denote the even and sequences, we may represent the wavelet coefficients as

$$W_i(z) = z^{-2}X_0(z) - A(z)X_e(z). \tag{14}$$

$A(z)$ works as an approximating filter yielding an estimate of the odd data points based on the even sequence. The wavelet sequence can be interpreted as the difference between the odd points and their estimate.

3. Discussion

Compared to the conventional pyramidal data compression methods, which have the redundancy of two, the present tree structured DWT has redundancy of one. The amplitude response of the HBF (14) diminishes towards zero at the Nyquist frequency $\omega = \pi$ $(z = -1)$. The phase response of the HBF is linear at the whole frequency range $-\pi \leq \omega \leq \pi$. The improvement in amplitude response can be obtained by increasing the length of the HBF. The next possible choice would be 15 point wavelet filter, which has sixth order zero at $z = 1$. However, the length of the filter would increase over twice and prolong the computation time. The short filter follows rapid changes in the signal and the compression efficiency is better.

Pyramid data compression methods perform optimally when the impulse response of the compression filter approaches sinc-function. A compactly supported HBF serves as a good candidate for the interpolation kernel. Its frequency response is close to the ideal interpolation, i.e., constant in the frequency range $-\pi \leq \omega \leq \pi$. The wavelet sequence can be interpreted as the difference between the odd points and their estimate. The performance of the present method depends on how well the $A(z)$ filter (14) can produce an estimate of the odd points based on the even points. In this context the wavelet coefficients differ significantly from the pyramidal compression, where all the adjacent (even and odd) data values are used for the computation of the estimate.

In this work it appeared that the low-pass scaling filter cannot be HBF structured, if the DWT is based on the FIR filters. To obtain the HBF scaling and wavelet filters, more complex FFT based computation [9] is needed. In VLSI applications the requirement for the FFT chip is usually unrealistic and too expensive. In previous works *B*-spine filters have applied to construct tree structured DWTs [10, 11]. At the moment the present approach based on the lifting DWT is probably the simplest architecture for compression encoder. The lifting steps (Figure 4) can be carried out with register shifts and summations in hardware implementations. The reconstruction of the compressed signal or image is lossless compared with lossy compression methods, such as jpeg standard.

References

[1] P. Burt and E. Adelson, The Laplacian pyramid as compact image code, IEEE Trans. Comm. 3 (1983), 532-540.

[2] H. Olkkonen and P. Pesola, Gaussian pyramid wavelet transform for multiresolution analysis of images, Graphical Models and Image Processing 4 (1996), 394-398.

[3] M. J. T. Smith and T. P. Barnwell, Exact reconstruction for tree-structured subband coders, IEEE Trans. Acoust. Speech Signal Process. 34 (1986), 434-441.

[4] I. Daubechies, Orthonormal bases of compactly supported wavelets, Commun. Pure Appl. Math. 41 (1988), 909-996.

[5] W. Sweldens, The lifting scheme: a construction of second generation wavelets, SIAM J. Math. Anal. 29 (1988), 511-546.

[6] H. Olkkonen, J. T. Olkkonen and P. Pesola, Efficient lifting wavelet transform for microprocessor and VLSI applications, IEEE Signal Processing Letters 12 (2005), 120-122.

[7] ITU-T Recommend, T.800-ISO FCD15444-1: JPEG2000 Image Coding System, International Organization for Standardization, ISO/IEC JTC1 SC29/WG1, 2000.

[8] H. Olkkonen and J. T. Olkkonen, Simplified biorthogonal discrete wavelet transform for VLSI architecture design, Signal, Image Video Process. 2 (2008), 101-105.

[9] H. Olkkonen, J. T. Olkkonen and P. Pesola, FFT-based computation of shift invariant analytic wavelet transform, IEEE Signal Process. Lett. 14(3) (2007), 177-180.

[10] A. Aldroubi, M. Eden and M. Unser, Discrete *B*-spline filters for multiresolutions and wavelets of ℓ_2, SIAM J. Math. Anal. 25 (1994), 1412-1432.

[11] H. Olkkonen and J. T. Olkkonen, Half-delay *B*-spline filter for construction of shift invariant wavelet transform, IEEE Trans. Circuits and Systems II 54 (2007), 611-615.