

A Reconfigurable Computing Engine for Wavelet Transforms*

Kang Sun, Xuezheng Pan[†] and Lingdi Ping

College of Computer Science and Technology, Zhejiang University, Hangzhou, 310027, China.
ksun@zju.edu.cn

Abstract

In the past a few years, wavelet transforms have become a hot topic of research. Discrete and continuous wavelet transforms have been widely used in signal and multimedia processing. Due to the high performance and flexibility of reconfigurable computing systems, it is very attractive to design a reconfigurable architecture for discrete and continuous wavelet transform of wide range of wavelet filters. In this paper, a unified computation framework for discrete and continuous wavelet transform based on lifting scheme and a reconfigurable architecture that includes reconfigurable lifting step arrays and reconfigurable address generator are proposed. The unified framework is the theory basis of this system. The step array is the computing core of this engine. And the address generator supports several memory scan pattern which is used to generate memory access addresses. In order to validate this architecture, an FPGA prototype is built based on Xilinx VirtexII FPGA to test the reconfiguration of 2-D discrete 5/3 and 9/7 transforms (defined in specification of JPEG2000) and 2-D continuous Haar wavelet transform. Furthermore, a 3-level decomposition for a 512×512 grayscale image is performed and the results show that the decomposition can be finished within 12.16ms when running at 20MHz. It can be concluded that this design is applicable and scalable.

1. Introduction

In the past twenty years, there has been an ever-increasing amount of interest in wavelet transform. There are several types of wavelet transforms depending on the nature of the signal (continuous or discrete) and the time and scale parameters (continuous or discrete). In this paper we focus on the implementation of the discrete wavelet transform (DWT) and the continuous wavelet transform (CWT)

defined in [11]. The DWT and CWT are widely used in signal analysis [3], image compression [8][7], and so on. Generally speaking, certain wavelet is suitable for a class of signals or applications, and different wavelets or adaptive ones are selected according to different signals or applications. So it is significant to design a reconfigurable system by ASIC or reconfigurable hardware like FPGA for discrete and continuous wavelet transform of different wavelet filters.

At algorithmic level, the lifting scheme provides a theoretical basis for the reconfigurable system. In [4], Daubechies and Sweldens proposed the lifting scheme for discrete transform, and Stoffel first extended the lifting scheme to unsubsampling wavelet transform, namely continuous wavelet transform [13]. So, it is possible to build a unified lifting scheme framework for reconfigurable computing of discrete and continuous wavelet transforms.

Lifting scheme can lead to a faster, in place calculation of wavelet transform. So it is widely used to speed up the DWT computation and possibly reduce the memory requirement of 2-D DWT. At implementation level, there have been a lot of efforts on VLSI architecture of discrete wavelet transforms. They did not mainly focus on wide wavelet filters but on one or two specific wavelet filters [9][1][10][5]. As far as the CWT is concerned, little literature is available on mapping the CWT onto VLSI. Literature [2] presents a wide range of algorithms and architectures for computing the 1-D and 2-D DWT and the 1-D and 2-D CWT. However, those architectures are based on convolution computation and the DWT architecture is not compatible with the CWT. In [6], a reconfigurable FPGA prototype of DWT for some wavelet filters is proposed. And literature [12] introduces a systemic and reconfigurable VLSI architecture for DWT. But neither [6] nor [12] discussed the reconfigurable computing of CWT.

The rest of this paper is organized as follows: A unified computation framework for DWT and CWT based on lifting scheme is proposed in section 2. In section 3, the reconfigurable architecture is presented and reconfigurable lifting step kernel and reconfigurable address generator are described in detail. An FPGA prototype for 5/3 and 9/7

*This work is supported by Natural Science Foundation of Zhejiang Province, China (Grant No. Y105355).

[†]contact author: Xuezheng Pan, email: xzpan@zju.edu.cn
1-4244-0910-1/07/\$20.00 ©2007 IEEE.

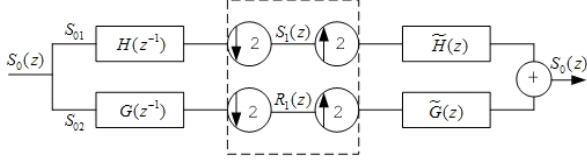


Figure 1. Block diagram of discrete wavelet transform and continuous wavelet transform.

DWT and Haar wavelet CWT is designed in section 4, furthermore, a 3-level decomposition for a 512×512 grayscale image is performed and the performance is tested. Finally, a brief summary in section 5 concludes this paper.

2. Unified Computation Framework of Discrete and Continuous Wavelet Transforms

Lifting scheme [4] is not only a new approach for constructing the second-generation wavelet but also an effective algorithm for accelerating computation of wavelet transforms. Computing wavelet transforms by lifting scheme can provide many benefits, such as allowing faster and fully in-place implementation of the wavelet transforms, immediate computation of the inverse transform, easy to manage the boundary extension, and so on.

The forward DWT consists of two analysis filters - low pass $H(z^{-1})$ and $G(z^{-1})$ high pass followed by subsampling while the inverse DWT first upsamples and then uses two synthesis filters low pass $\tilde{H}(z)$ and high pass $\tilde{G}(z)$. If the subsamplings and upsamples are deleted, this becomes the block diagram of CWT.

From Figure 1, we can see the difference between 2-scale continuous wavelet transform and discrete wavelet transform lies in using of subsampling, which can cause different factorization forms of wavelet filters with lifting scheme.

According to literature [4], forward discrete wavelet transform is computed by the following (for even n):

$$\begin{bmatrix} S_1(z) \\ R_1(z) \end{bmatrix} = \prod_{i=1}^{n/2} \begin{bmatrix} 1 & S_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ t_i(z) & 1 \end{bmatrix} \begin{bmatrix} c_1 & 0 \\ 0 & c_2 \end{bmatrix} \begin{bmatrix} S_{even}(z) \\ S_{odd}(z) \end{bmatrix}. \quad (1)$$

$S_1(z)$ and $R_1(z)$ are low subband and high subband respectively. $S_i(z)$ and $t_i(z)$ are predict lifting step and update lifting step respectively. c_1 and c_2 are normalized factors. $S_{even}(z)$ and $S_{odd}(z)$ are even part and odd part of $S_0(z)$ respectively.

In literature [13], forward continuous wavelet transform

is computed by the following:

$$\begin{bmatrix} S_1(z) \\ R_1(z) \end{bmatrix} = W_e(z) \begin{bmatrix} 1 & Q_{n-1}(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ Q_n(z) - 1 & 1 \end{bmatrix} \begin{bmatrix} cS_0(z) \\ cS_0(z) \end{bmatrix} \quad (2)$$

where

$$W_e(z) = \prod_{i=1}^{\frac{1}{2}n-1} \begin{bmatrix} 1 & Q_{2i-1}(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ Q_{2i}(z) & 1 \end{bmatrix} \quad (3)$$

and lifting steps are $Q_{n-1}(z)$ and $Q_n(z)-1$ and so on. c is a constant factor.

By integrating eq. (1) with eq. (2), we can build a unified computation framework for DWT and CWT.

Computing forward wavelet transform, the wavelet filter is factorized by

$$\begin{bmatrix} S_1(z) \\ R_1(z) \end{bmatrix} = \prod_{i=1}^{n/2} \begin{bmatrix} 1 & Q_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ Q_{i-1}(z) & 1 \end{bmatrix} \begin{bmatrix} c_1 S_{01}(z) \\ c_2 S_{02}(z) \end{bmatrix}. \quad (4)$$

For DWT, $S_{01}(z)$ and $S_{02}(z)$ are even part and odd part of $S_0(z)$ respectively, while for CWT, they are equal to $S_0(z)$.

When performing inverse wavelet transform, the wavelet filter is factorized by

$$\begin{bmatrix} S_{01}(z) \\ S_{02}(z) \end{bmatrix} = \prod_{i=1}^{n/2} \begin{bmatrix} 1 & -Q_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -Q_{i-1}(z) & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{c_1} S_1(z) \\ \frac{1}{c_2} R_1(z) \end{bmatrix}. \quad (5)$$

The specific factorization of wavelet filters is calculated by Euclid's algorithm.

3. Reconfigurable Architecture for DWT and CWT

3.1 Proposed reconfigurable architecture for DWT and CWT

In order to support wide range of wavelet filter kernels and to be compatible with DWT and CWT, a reconfigurable computing architecture is proposed as shown in Fig. 2. The architecture consists of four parts: reconfigurable lifting step array (RLSA), reconfigurable address generator (RAG), two dual port SRAM and main controller (general microprocessor, MCU). Reconfigurable address generator is responsible for calculating the address of access to the two dual-port SRAM memories, according to current work modes (DWT and CWT, 1-D or 2-D, 1 level or multilevel decomposition, etc). Reconfigurable lifting step array is linear array connected by some reconfigurable lifting step kernels. RLSA can be configured for different wavelet transforms by modifying some parameters such as the number of delay registers and pipeline registers. The two dual-port

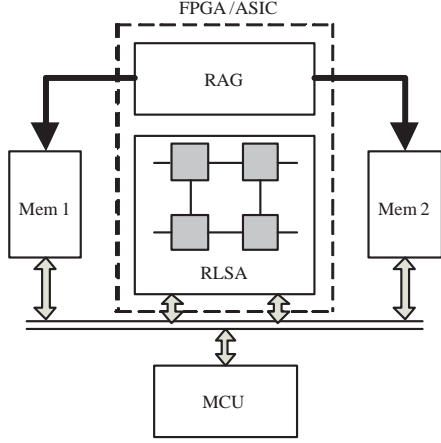


Figure 2. Proposed architecture for reconfigurable computing of DWT and CWT.

SRAM memories are used for storage of image or signal source and computation results and the data width is 16 bits. Main controller is in charge of harmonizing and commanding different parts to work.

The RLSA and RAG are the core of the reconfigurable architecture, which are implemented by FPGA or ASIC. They are satisfied with hardware reconfiguration, while the main controller is able to obtain software reconfiguration by reloading program.

3.2 Reconfigurable Lifting Step kernel

According to the above unified computation framework for DWT and CWT, each lifting step is Z-polynomial. Those polynomials can always be split into such forms as α , $\alpha(1+z^m)$, or αz^m by two matrix transforms:

$$\begin{bmatrix} 1 & a(z) + b(z) \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & a(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & b(z) \\ 0 & 1 \end{bmatrix} \quad (6)$$

and

$$\begin{bmatrix} 1 & 0 \\ a(z) + b(z) & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ a(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ b(z) & 1 \end{bmatrix}. \quad (7)$$

So, we can employ the characteristic of each lifting step final form to design a reconfigurable lifting step kernel as depicted in Fig. 3.

The proposed kernel is a two-input-two output data-path and holds one multiply unit, two adders, and some delay and pipeline registers. The parameter, A_Delay, of delay registers can be selected to support different delay requirement of lifting steps. By modifying the parameter, B_Pipeline and C_Pipeline, of pipeline registers, the multiplier and adder can be pipelined to improve the throughput. Changing of

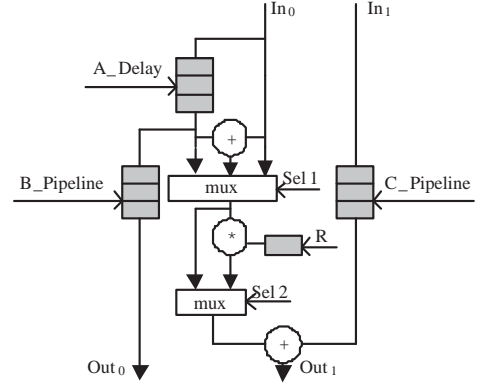


Figure 3. The structure of the reconfigurable lifting step kernel.

Table 1. Definition of reconfigurable parameters.

A_Delay	The number of A delay registers	
B_Pipeline	The number of B pipeline registers	
C_Pipeline	The number of C pipeline registers	
R	Multiplicand value	
Sel_1	00	$\alpha(1+z^m)$
	01	α
	10	αz^m
Sel_2	0	No multiplier
	1	Use multiplier

value of R register and the multiplexers Sel1 and Sel2 can reorganize the computation form of the kernel (according to Table 1.). As far as multiplier is concerned, fixed-point representation seems to be a strongly advisable choice to deliver power-efficient hardware implementation. Moreover, integer arithmetic units can be easily pipelined, in order to achieve very high data throughput.

3.3 Reconfigurable Address Generator

Because CWT and DWT based on lifting scheme is regular in access to memories, the generation of memory address can be reconfigured. The calculation of specific address depends on the size of input signals or images, and decomposition levels, and the modes of wavelet transforms. The modes are as follows: 1) 1-D DWT, 2)1-D CWT, 3) 2-D vertical DWT, 4)2-D horizontal DWT,5) 2-D vertical CWT, 6)2-D horizontal low subband CWT and 7) 2-D horizontal high subband CWT.

According to different modes, the related status of access to the two memories is listed in Table 2. As for the data

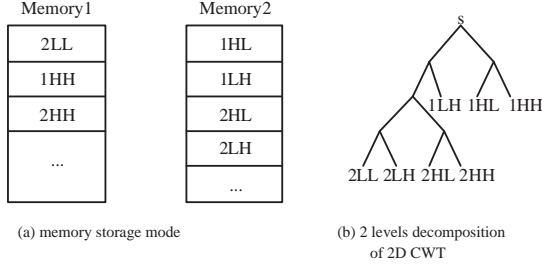


Figure 4. Demonstration of memory storage manner for 2-D CWT.

Table 2. Status of access to the two memories with different modes.

Mode	Memory 1 Status	Memory 2 Status
1	Read, Read	Write, Write
2	Read, Write	Write
3	Read, Read	Write, Write
4	Write, Write	Read, Read
5	Read, Write	Write
6	Read, Write	Write
7	Write	Read, Write

storage ways, DWT coefficients are easily stored in in-place mode, while storing of CWT coefficients is rather complex. Because the size of 1-D CWT coefficients is doubled, the low subband coefficients is stored using in-place mode and high coefficients is stored in the other memory. In Fig. 4, the storage of 2-D CWT coefficients is done in a similar way.

Table 3. Hardware Resource Usage of the Design.

Target Device	VirtexII xc2v500
Number of Slices	1175
Number of Slice Flip Flops	612
Total Number 4 input LUTs	2176
Number of bonded IOBs	123
Number of Block Multipliers	4

4 Implementation Results

In order to validate the performance of the reconfigurable architecture for DWT and CWT, we used Verilog HDL to design the reconfigurable lifting array and the reconfigurable address generator. The RTL code was synthesized

Table 4. Throughput of Three Wavelet Filters with One Level Decomposition.

Applications	Throughput (Pixels Per Clock Cycle)	Runtime (512×512×8 @20MHz)
5/3 Filter	1/(4/N+1)	13.31ms
9/7 Filter	1/(10/N+1)	13.42ms
Haar Filter	1/(4/N+2)	26.35ms

for the Xilinx Virtex II (xc2v500) FPGA. Table 3 presents the hardware resource usage of the design. The reconfigurable lifting array includes four lifting step kernels. The 5/3 and 9/7 wavelet filters that are defined in JPEG2000 standard [8] are chosen for 2-D DWT. The Haar wavelet, which is widely used in image edge detection, is taken as the wavelet filter of 2-D CWT.

According to the unified computation framework introduced in section 2, 5/3, 9/7 and Haar wavelet filter are factorized respectively as follows:

$$\begin{bmatrix} 1 & -\frac{1}{2}(z^{-1} + 1) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \frac{1}{4}(z + 1) & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad (8)$$

$$\begin{bmatrix} 1 & \alpha(1 + z^{-1}) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \beta(z + 1) & 1 \end{bmatrix} \begin{bmatrix} 1 & \gamma(1 + z^{-1}) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \delta(z + 1) & 1 \end{bmatrix} \begin{bmatrix} \rho \\ \frac{1}{\rho} \end{bmatrix} \quad (9)$$

$$(\alpha = -1.5861, \beta = -0.0530, \gamma = 0.8829, \delta = 0.4435, \rho = 1.1496)$$

$$\begin{bmatrix} 1 & \frac{1}{2} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -1 + z^{-1} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad (10)$$

Eq.(8)-(10) show that the number of lifting step kernels for 2-D 5/3 DWT, 2-D 9/7 DWT and 2-D Haar wavelet CWT are 2, 4 and 3. According to eq. (8)-(10) and signal dependence diagram analysis, we obtained configuration parameters of each lifting step kernels. The theoretical throughput of N length input data and practical run time of three wavelet filters are listed in Table IV. We performed a 3-level decomposition for a 512×512 grayscale image by the reconfigurable wavelet engine architecture. The executive clock cycles of our architecture for a N×N tile with 3 levels decomposition is $N^2/4 + (4+N/4)*2N + (4+N/8)*N + (4+N/16)*N/2$. The 3-level decomposition for a 512×512 grayscale image is finished within 12.16ms when running at 20MHz. Simulation results showed that the FPGA prototype system not only realized hardware reconfiguration but also has better performance for 2-D DWT and CWT.

5 Conclusions

In this paper, we proposed a unified computation framework for DWT and CWT based on lifting scheme and design a reconfigurable architecture for DWT and CWT of wide range wavelet filter. Experiment results show the architecture is effective. Because the CWT and DWT are both widely used in signal and multimedia processing, this work is attractive and valuable. However, in the future, some work requires to be done, such as the design of fine granularity lifting step kernels and automatic generation of RTL code and configuration files of FPGA, etc.

References

- [1] H.-H. C. C.-J. Lian, K.-F. Chen and L.-G. Chen. Lifting based discrete wavelet transform architecture for jpeg2000. In *The 2001 IEEE International Symposium on Circuits and Systems*, volume 2, pages 445–448, May 2001.
- [2] C. Chakrabarti and M. Vishwanath. Efficient realizations of the discrete and continuous wavelet transforms: from single chip implementations to mappings on simd array computers. *IEEE Transactions on Signal Processing*, 43(3):759–771, Mar. 1995.
- [3] I. Daubechies. The wavelet transform, time-frequency location and signal analysis. *IEEE Transactions on Information Theory*, 36(5):961–1005, Sept. 1990.
- [4] I. Daubechies and W. Sweldens. Factoring wavelet transforms into lifting steps. Technical report, Bell Laboratories, Lucent Technologies, 1996.
- [5] J. D. L. G. Dillen, B. Georis and O. Cantineau. Combined line-based architecture for the 5-3 and 9-7 wavelet transform of jpeg2000. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(9):944–950, Sept. 2003.
- [6] P. S. G. Kuzmanov, B. Zafarifar and S. Vassiliadis. Reconfigurable dwt unit based on lifting. In *13th Annual Workshop on Circuits, Systems, and Signal Processing (ProRISC2002)*, volume 1, pages 325–333, Nov. 2002.
- [7] ISO/IEC JTC/SC29/WG11, N25021. *Generic Coding of Audio-Visual Objects: Visual 14496-2. FinalDraft*, 1998.
- [8] ISO/IEC15444-1. *Information Technology JPEG2000 Image Coding System*, 2000.
- [9] Y.-H. S. J.-M. Jou and L. C.-C. Efficient vlsi architectures for the biorthogonal wavelet transform by filter bank and lifting scheme. In *The 2001 IEEE International Symposium on Circuits and Systems*, volume 2, pages 529–532, May 2001.
- [10] C. C. K. Andra and T. Acharya. A vlsi architecture for lifting-based forward and inverse wavelet transform. *IEEE Transactions on Signal Processing*, 50(4):966–977, Apr. 2002.
- [11] O. Rioul and P. Duhamel. Fast algorithms for discrete and continuous wavelet transforms. *IEEE Transactions on Information Theory*, 38(2):569–586, Mar. 1992.
- [12] C. T. H. P. C. Tseng and L. G. Chen. Reconfigurable discrete wavelet transform architecture for advanced multimedia systems. In *2003 IEEE Workshop on Signal Processing Systems*, volume 1, pages 137–141, Aug. 2003.
- [13] A. Stoffel. Remarks on the unsubsampled wavelet transform and the lifting scheme. *IEEE Transactions on Signal Processing*, 69(2):177–182, Sept. 1990.