Indonesian Journal of Electrical Engineering and Computer Science
Vol. 3, No. 2, August 2016, pp. 323 ~ 330
DOI: 10.11591/ijeecs.v3.i2.pp323-330

VLSI Architectures for Lifting Based Discrete Wavelet Transform – A Survey

M. Kalaiarasi[1], T. Vigneswaran[2]
[1]Department of Telecommunication Engineering, Sir. MVIT, Bangalore, India
[2]School of Electronics Engineering, VIT University, Chennai, India
Corresponding author, e-mail: kalaiarasi.m2013@vit.ac.in, vigneswaran.t@vit.ac.in

Abstract

Image compression is a key technology in the development of various multimedia and communication applications. Perfect reconstruction of the image without any loss in picture quality and data is very important. This can be achieved with the Discrete Wavelet Transform (DWT), which is an efficient tool for image compression and video compression. The lifting based DWT architecture has the advantage of lower computational complexities and also requires less memory compared to the conventional convolution method. The existing DWT architectures are represented in terms of folded, flipping and recursive structures. The various architectures are discussed in terms of memory, power consumption and operating frequency involved with the given size of image and required levels of decomposition. This paper presents a survey of these architectures for 2-dimensional and 3-dimensional Discrete Wavelet Transform. This study is useful for deriving an efficient method for improving the speed and hardware complexities of existing architectures.

Keywords*: DWT, DCT, lifting scheme, JPEG 2000, VLSI architecture*

1. Introduction

Discrete Wavelet Transform (DWT) is a multiresolution analysis tool with good resolution in the time and frequency domains. The DWT is best suited for image compression as it offers higher coding efficiency, compression ratio, and quality of image restoration than the traditional Discrete Cosine Transform (DCT). Implementation of Very Large Scale Integration (VLSI) architecture with reduced memory, fast computation speed, low power and low cost is a major area of research. The DWT is used in compression standards such as MPEG-4 and JPEG 2000. Discrete wavelet transform is based on the sub-band coding. This is widely used in image compression due to its good features. DWT supports frequency domain and space domain coefficients analysis, which helps better analysis for image in computer vision. DWT is easy to implement, computation overhead is less and fast computation of wavelets, which makes it useful for image compression. Traditional DWT architectures are based on convolutions, which are not suitable for an efficient hardware implementation due to the complex arithmetic computation. Daubechies and Sweldens derived a lifting-based DWT to reduce complex arithmetic operations [1]. The main feature of the lifting-based DWT scheme is to break up the high-pass and low-pass wavelet filters into a sequence of upper and lower triangular matrices, and convert the filter implementation into banded matrix multiplications [2]. Lifting scheme can be easily implemented by hardware due to many advantages, such as “in-place” computation, reduced computations, and integer-to-integer wavelet transforms.

This paper is organized as follows: Section II briefly describes the operation of DWT, Section III reviews the lifting based DWT, Section IV describes the existing architectures of DWT. Section V illustrates the past work as a Literature survey. Finally, concluding remarks are given in Section VI.

2. Discrete Wavelet Transform

Discrete wavelet transform analyses the data at different frequencies with different time resolutions. DWT is based on the sub-band coding and can decompose the input data into sub groups of even and odd samples. The DWT decomposition involves low-pass “l” and high-pass

Received March 2, 2016; Revised June 25, 2016; Accepted July 10, 2016

“h” filtering of the images in both horizontal and vertical directions. After each filtering, the output is down-sampled by two. Further decomposition is done by applying the above process to LL sub band. The one-Dimensional (1-D), two-Dimensional (2-D) and three-Dimensional (3-D), discrete wavelet transforms are discussed in the following section.

2.1. One-Dimensional Discrete Wavelet Transform

In the one-dimensional discrete wavelet transform, the input discrete signal is filtered at each level of transformation with the help of low pass filter and high-pass filter. It produces two filtered output which are subsampled to achieve the high pass sub-band and low pass sub-bands. These low pass and high pass sub-bands can be written as Equation (1). The signal analysis of 1-D DWT is represented in Figure 1.

$$\begin{aligned} y_L(n) &= \sum_{i=0}^{\frac{N}{2}-1} h(2n-i).x(i) \\ y_H(n) &= \sum_{i=0}^{\frac{N}{2}-1} g(2n-i).x(i) \end{aligned} \qquad (1)$$

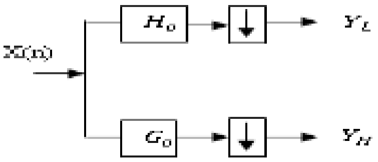

Figure 1. One-Dimensional DWT Signal Analysis

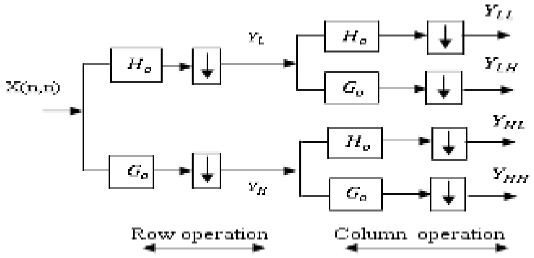

Figure 2. Two-Dimensional DWT Operation

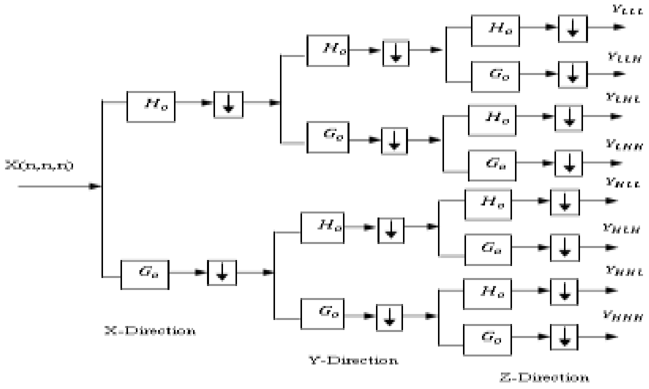

Figure 3. Three-Dimensional DWT Operation

2.2. Two-Dimensional Discrete Wavelet Transform

The basic idea of 2-D architecture is similar to 1-D architecture. The 2-D wavelet filters are separable functions. It can be seen as a 1-D wavelet scheme which transform along the rows and then a 1-D wavelet transform along the column. The 2-D DWT operates in a straightforward manner by inserting array transposition between the two 1-D DWT. First apply the 1-D DWT row-wise to produce Y_L and Y_H subbands and then column-wise to produce four sub bands $Y_{LL,}$ Y_{LH} , Y_{HL} and Y_{HH} as shown in Figure 2 In each level of decomposition.

2.3. Three - Dimensional Discrete Wavelet Transform

The three - dimensional DWT can be considered as a combination of three 1-D DWT in the X, Y and Z directions [3]. First, the process transforms the data in the X-direction. Next, the low and high pass outputs both feed to other filter pairs, which transforms the data in the Y-direction. These four output streams go to the next four filter pairs, performing the final transform in the Z-direction. The process results in eight data streams, seven detailed coefficients and one approximate coefficient as shown in Figure 3. After one-level of 3-D discrete wavelet transform, the image is decomposed into $Y_{HHH,}$, Y_{HHL} , Y_{HLH} , $Y_{HLL,}$ $Y_{LHH,}$ $Y_{LHL,}$ $Y_{LLH,}$ Y_{LLL} signals. The approximate signal, resulting from scaling operations only, goes to the next octave of the 3-D transform.

3. Lifting Based Discrete Wavelet Transform

Discrete Wavelet Transform has been traditionally implemented by convolution or FIR filter bank structures, resulting in both, a large number of arithmetic computations and a large storage. Lifting scheme is a new generation technique to construct biorthogonal wavelets different from convolution, with low complexity and buffers. Filter banks can be factorized into a sequence of lifting steps. Let $\tilde{h}(z)$ and $\tilde{g}(z)$ be low-pass and high-pass filters, the associated polyphase matrix $\tilde{P}(z)$ is described as:

$$\begin{bmatrix} \boldsymbol{LP(z)} \\ \boldsymbol{HP(z)} \end{bmatrix} = \widetilde{\boldsymbol{P}}(\boldsymbol{z}) \begin{bmatrix} \boldsymbol{X_e(z)} \\ \boldsymbol{X_o(z)} \end{bmatrix} \tag{2}$$

The $\tilde{h}_e(z)$ and $\tilde{h}_o(z)$ denote the even and odd polyphase components of the low pass analysis filter and $\tilde{g}_e(z)$ and $\tilde{g}_o(z)$ denote the even and odd polyphase components of the high pass analysis filter. With the polyphase matrix, the wavelet decomposition can be expressed as:

$$\begin{bmatrix} \boldsymbol{LP(z)} \\ \boldsymbol{HP(z)} \end{bmatrix} = \widetilde{\boldsymbol{P}}(\boldsymbol{z}) \begin{bmatrix} \boldsymbol{X_e(z)} \\ \boldsymbol{X_o(z)} \end{bmatrix} \tag{3}$$

$LP(z)$ respresents the low pass coefficients, $HP(z)$ denotes the high pass coefficients, $X_e(z)$ represents the even polyphase components, $X_o(z)$ represents odd polyphase components and $\tilde{P}(z)$ is the polyphase matrix. $\tilde{P}(z)$ can be factorized into a series of matrix multiplications with alternating upper and lower triangulation matrices as:

$$\widetilde{\boldsymbol{P}}(\boldsymbol{z}) = \begin{bmatrix} \boldsymbol{K} & \boldsymbol{0} \\ \boldsymbol{0} & \frac{\boldsymbol{1}}{\boldsymbol{K}} \end{bmatrix} \prod_{i=1}^{m} \left\{ \begin{bmatrix} \boldsymbol{1} & \boldsymbol{0} \\ \tilde{\boldsymbol{t}}_i(\boldsymbol{z}) & \boldsymbol{1} \end{bmatrix} \begin{bmatrix} \boldsymbol{1} & \tilde{\boldsymbol{s}}_i(\boldsymbol{z}) \\ \boldsymbol{0} & \boldsymbol{1} \end{bmatrix} \right\} \tag{4}$$

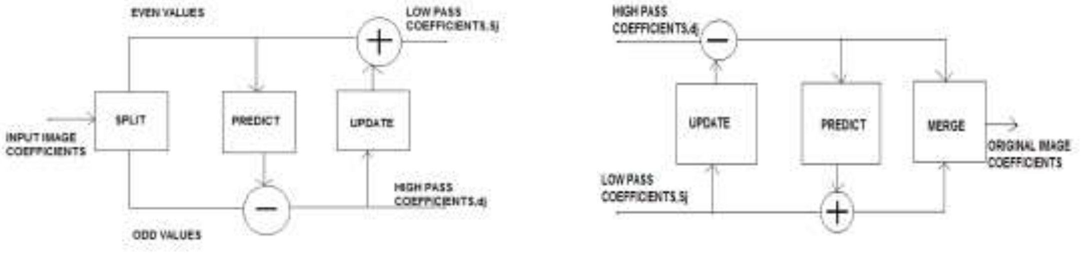

Figure 4. Forward and Inverse Lifting Scheme

Where K is a constant and m is the number of predict and update steps. $\tilde{t}_i(z)$ is predictor and $\tilde{s}_i(z)$ is updater unit. Computation of the upper triangular matrix is known as *primal lifting* and this is referred to in the literature as lifting the low-pass subband with the help of the high-pass subband. Similarly, computation of the lower triangular matrix is called *dual lifting*, which is lifting the high-pass subband with the help of the low-pass subband [1, 2]. These two basic lifting steps are called update and predict. In the lifting scheme the polyphase matrix is transformed into elementary matrix by performing factoring which reduces the complexity. The forward and inverse lifting scheme is shown in Figure 4. This scheme consists of three major stages: (i) Split (ii) Predict (iii) update. Lifting-based DWT architectures can be constructed by combining the above mentioned three basic lifting elements, split, predict and update [3, 4]. The input image coefficients are split into the odd and even coefficients in the split stage. This reduces the number of operations by two. The predict step calculates the wavelet function in the wavelet transform and results in the high pass filter coefficients. The update step calculates the scaling function, which results in low pass filter coefficients.

4. Discrete Wavelet Transform Architectures

Analysis of the various DWT architectures is required for the design of an efficient hardware with reduced memory. In this section the different types of architectures used for the implementation of DWT are discussed below.

4.1. Parallel and Direct Mapped Architectures

Liu, et al., proposed a direct mapping of the data dependency diagram into a pipelined architecture in [5]. The architecture can be sequentially pipelined by combining the previous output of predict stage to current output in this structure. U1 (0) represents the current output of the U1 unit and P1 (-1) represents the previous output of the P1 unit, and so on, shown in Figure 5. The control signal S, which has four states, selects the inputs of the multiplexers sequentially. In the first state, two consecutive input samples arrive and the P1 function with a coefficient is performed on them. In the second state, the U1 function with b coefficient will be imposed on the result of the previous state (first state's output). Similarly, in the third and fourth states, computations for P2 and U2 units will be performed on the results of the previous states. Thus, P2 and U2 produce final outputs for the structure. These steps can be used for parallel processing of inputs**.**

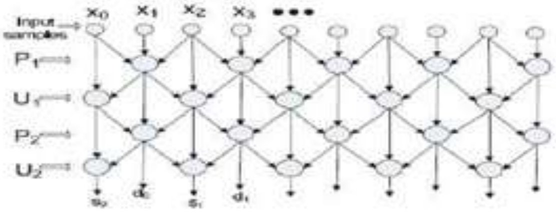

Figure 5. Parallel Architecture [5]

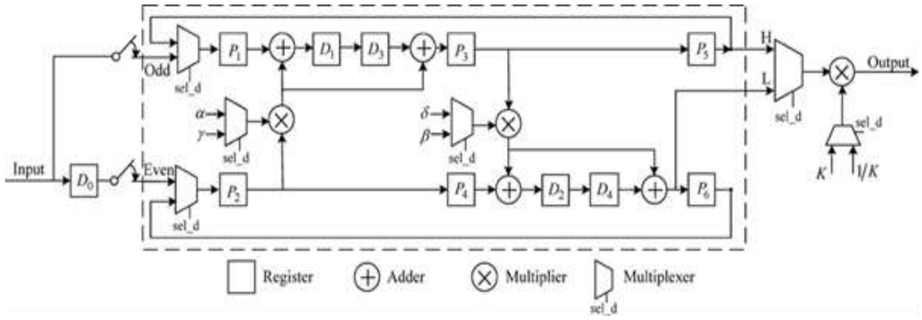

Figure 6. Folded Architecture [6]

4.2. Folded Architectures

Guangming Shi, et al., proposed an efficient folded architecture. This architecture overcomes the limitations on the critical path latency and memory requirement. Here the parallel and pipeline techniques are used to process the intermediate data. The corresponding architecture possesses repeatable property. Intermediate data $d_i^{(1)}$ and $s_i^{(1)}$, which were obtained from the first lifting step, are fed back to pipeline registers P_1 and P_2. They are used for the second lifting step. As a result, the first and second lifting steps are interleaved by selecting their own coefficients. In the proposed architecture, the speed of the internal processing unit is two times that of the even (odd) input/output data. This architecture needs only four adders and two multipliers. For the splitting, one delay register and two switches to split the input into odd/even sequences has been used. With regard to scaling, it consists of one multiplier and one multiplexer. By properly selecting coefficients K and 1/K, the high-pass and low-pass coefficients are normalized. Figure 6 shows the complete structure of the folded architecture, including the splitting, lifting, and scaling steps [6].

4.3. Flipping Architecture

The conventional lifting-based architectures require fewer arithmetic operations, they sometimes have long critical paths. The critical path of the lifting-based architecture for the (9, 7) filter is 4Tm +8Ta, while that of the convolution implementation is Tm + 4Ta. The critical path delay can be improved by pipelining, but this results in a significant increase in the number of registers. Huang et al. proposed a very efficient way of solving the timing accumulation problem. The basic idea is to remove the multiplications along the critical path by scaling the remaining paths by the inverse of the multiplier coefficients. Here two computing units are possibly connected as Figure 7(a), in which s(z) and t(z)are assumed to be two taps.

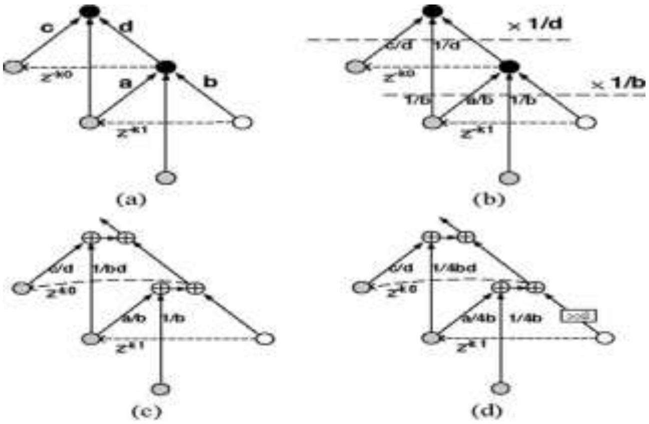

Figure 7. (a) Two connected computing units, (b) Flipping computing units, (c) After splitting computation nodes and merging the multipliers, (d) Flipping with the inverse multiplying by 4 [7]

This architecture can be flipped by the inverse coefficients, as shown in Figure 7(b). Flipping is to multiply the inverse coefficient for every edge on the feed-forward cutset, which is through the selected multiplier. Then, every computation node can be split into two adders, of which one can process in parallel with other computing units, and the other one is on the accumulative path, as illustrated in Figure 7(c). This flipping structure can minimize the size of the internal buffer for line-based DWT architectures under specified timing constraints [7].

4.4. Recursive Architecture

L Hongyu, et al., proposed 1-D and 2-D recursive architectures. The recursive architecture (RA) is a general scheme that can be used to implement any wavelet filter that is decomposable into lifting steps. Two 1-D architectures are proposed, the first architecture calculates three stages of Daub-4 DWT and the second architecture calculates the three stage 9/7 DWT. These architectures exploits the interdependencies among the wavelet coefficients by interleaving, on alternate clock cycles using the same datapath hardware, the calculation of higher order coefficients along with that of the first-stage coefficients. The 2-D recursive architecture is faster than conventional implementations and it requires a buffer that stores only a few rows of the data array instead of a fixed fraction of the entire array [8].

4.5. Memory Scan Techniques

The memory scan techniques can be broadly classified into line-based scan, block-based scan and stripe-based scan. Though most of the existing architectures are based on line scan, all the three techniques are described for the 2-D DWT architectures.

1) Line-based Scan: In line-based scan, the scan order is raster scan. An internal line buffer of size LN is required, where N is the number of pixels in a row and L is the number of rows that are required for that particular filter, shown in Figure 8. A line-based implementation of the traditional convolution based wavelet transform has been discussed in detail in [9].

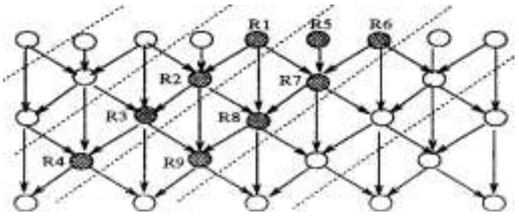

Figure 8. Line Based Scan Method [9]

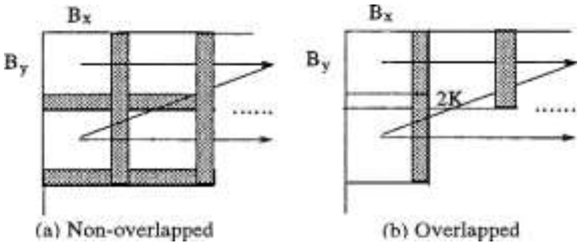

Figure 9. Block Based Scan Method [10]

2) *Block-based Scan:* In block-based scan, the frame memory is scanned block-by-block and the DWT coefficients are also computed block-by-block [10]. Figure 9 shows two configurations of block based methods where the blocks are scanned in the row direction blocks are not overlapped with each other and in the overlapped configuration, the blocks are overlapped by 2K pixels in the column direction. The block-based technique proposed in [11], first performs filtering operation on neighbouring data blocks independently and later combines the partial boundary results together.

3) *Stripe-based Scan*. The stripe-based scan is equivalent to the line-based scan with B_x = N. In other words, the stripe is a very wide block with width N and height S. As in the case of block-based scan, there are two categories, namely, the non-overlapped stripe based scan also referred to as the optimal Z-scan in. The non-overlapped stripe-based scan has an internal buffer of size LN + LS and N^2 frame memory READ accesses [12]. In contrast, the overlapped stripe-based scan has a significantly smaller internal buffer of size LS and N^2 S/(S –2K) frame memory READ accesses [10]. The non-overlapped and the overlapped stripe based scan methods are shown in shown in Figure 10.

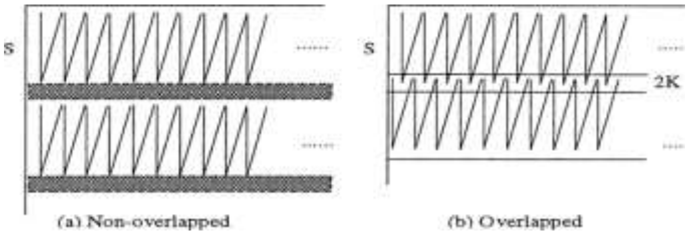

Figure 10. Stripe Based Scan Method

The performance of the available 2-D discrete wavelet transform architectures are studied and tabulated in table 1. The flipping architecture proposed by C T Huang, et al, utilizes 10 multipliers and 16 adders [7]. The same architecture with pipelining requires an increase in the buffer size. The Dual Scan Architecture proposed in [8] requires 12 multipliers and 16 adders with same buffer size as in [7] with an increase in the critical path delay. The RAM based architecture proposed in [13] requires 10 multipliers and 16 adders. The high speed architecture proposed in [14] requires addtional number of multipliers and adders compared to the other architectures, but the critical path delay is improved here. The architecture proposed in [15] is efficient in terms of computational time and hardware cost. C H Hsia, et al, proposed a memory efficient hardware architecture which requires zero multipliers and 16 adder [16].

Table 1. Performance Comparison of 2-D DWT Architectures

Architecture	Multiplier	Adder	Buffer	Critical Path	Throughput	HUE
Flipping [7]	10	16	4N	$T_m + 5T_a$	1	-
Flipping+Pipeline [7]	10	16	11N	T_m	1	-
Dual Scan Architecture[8]	12	16	4N	$4T_m + 8T_a$	1	100%
RAM based [13]	10	16	4N	$4T_m +8T_a$	1	-
Fast Architecture [14]	10	16	5.5N	$T_m + 2T_a$	2	100%
High speed Architecture [14]	18	32	5.5N	$T_m +2T_a$	4	100%
Cheng(L=1) [15]	10	40	14N	$T_m +4T_a$	2	-
C H Hsia [16]	0	16	4N	$2T_m + 4T_a$	2	-

5. Literature Survey

To study and analyze more about the 2-D and 3-D architectures for lifting based Discrete Wavelet Transform, the following literature survey is done and discussed in this section.

A detailed comparative study on the wavelet Image Compression is done by Ansam Ennaciri et.al. Here the image compression is done by keeping the essential quality of the original image with good PSNR and compression ratio.The image compression using different types of wavelets - the Biorthogonale wavelets, Daubechies wavelets, Haar and, Coiflets wavelets is performed and the results are compared [17]. The author Huang et.al, presented a detailed analysis of VLSI architectures for the one-dimensional and two-dimensional discrete wavelet transform. The architectures for convolution based, lifting based and B-spline based method is discussed in terms of hardware complexity, critical path and registers. A B-spline based architecture for Inverse Discrete Wavelet Transform (IDWT) is proposed which require fewer gate counts under the same timing constraints. It can be implemented with smaller area and lower speed. In this paper a overlapped stripe based scan method is proposed which can provide the best tradeoff among external memory access, internal buffer size, and the control complexity [18]. Yusong Hu, et. al., proposed a new parallel lifting based 2-D DWT architecture with high memory efficiency and short critical path. This architecture achieves the smallest memory size of 3n+24S. It has a constant latency and its TP is scalable by scaling the number S of Data Processing Pipe (DPP) in which a block of 2S pixels is processed every clock cycle [19]. B. K. Mohanty, et al., proposed a memory efficient architecture for 3-D DWT using overlapped grouping of frames and has a small cycle period, and offers small output latency. For frame-size 176x144 and frame-rate 60 fps, this structure uses 7.96 times less memory words and 12.3% less average computation time (ACT) than the best of the existing folded designs [20]. The author Shiva K. Madishetty, et al., proposed a multiplier-less architecture based on algebraic integer representation for computing the Daubechies 6-tap wavelet transform for 1-D/2-D signal processing. This architecture has the advantage of minimizing the number of parallel 2-input adder circuits. The algorithm is based on the brute-force numerical optimization of the algebraic integer representation [21].

6. Conclusion

In this paper, a survey of the existing lifting based implementations of 1-dimensional, 2-dimensional and 3-dimensional Discrete Wavelet Transform is presented. The various architectures from parallel to recursive and scan based methods are discussed. The number of multipliers and adders used in the architecture is very important for its performance in terms of hardware utilization and critical path delay. This study aims in developing an architecture with reduced number of computations and high throughput.

References

[1] Daubechies I, Sweldens W. Factoring wavelet transforms into lifting steps. *Journal Fourier Anal. Application*. 1998; 4: 247-269.

[2] W Sweldens. The Lifting Scheme: A Custom-Design Construction of Biorthogonal Wavelets. *Applied and Computational Harmonic Analysis*. 1996; 3(15): 186-200.

[3] Michael Weeks, Magdy A Bayoumi. Three-Dimensional Discrete Wavelet Transform Architectures. *IEEE Transactions on Signal Processing*. 2002; 50(8): 2050-2063.

[4] SA Salehi, S Sadri. Investigation of Lifting-Based Hardware Architectures for Discrete Wavelet Transform. *Journal of Circuits Systems and Signal Processing.* 2009; 28.

[5] CC Liu, H Shiau, JM Jou. *Design and Implementation of a Progressive Image Coding Chip Based on the Lifted Wavelet Transform*. In Proc. of the 11th VLSI Design/CAD Symposium. Taiwan. 2000.

[6] Guangming Shi, Weifeng Liu, Li Zhang, Fu Li. An Efficient Folded Architecture for Lifting-Based Discrete Wavelet Transform. *IEEE Transactions on Circuits and Systems—II: Express Briefs.* 2009; 56(4); 290-294.

[7] Chao-Tsung Huang, Po-Chih Tseng, Liang-Gee Chen. Flipping Structure: An Efficient VLSI Architecture for Lifting-Based Discrete Wavelet Transform. *IEEE Transactions on Signal Processing.* 2004; 52(4): 1080-1089.

[8] L Hongyu, MK Mandal, BF Cockburn. Efficient architectures for 1-D and 2-D lifting- based wavelet transforms. *IEEE Trans. Signal Process*. 2004; 52(5): 1315–1326.

[9] C Chrysafis, A Ortega. Line-Based, Reduced Memory, Wavelet Image Compression. *IEEE Trans. on Image Processing*. 2000; 9(3): 378-389.

[10] CT Huang, PC Tseng, LG Chen. *Memory Analysis and Architecture for Two-Dimensional Discrete Wavelet Transform*. In Proceedings of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing. 2004: 13-16.

[11] W Jiang, A Ortega. Lifting Factorization-Based Discrete Wavelet Transform Architecture Design. *IEEE Transactions on Circuits and Systems for Video Technology*. 2001; 11: 651-657.

[12] MY Chiu, KB Lee, CW Jen. *Optimal Data Transfer and Buffering Schemes for JPEG 20000 Encoder.* In Proceedings of the IEEE Workshop on Design and Implementation of Signal Processing Systems. 2003: 177-182.

[13] PC Tseng, CT Huang, LG Chen. *Generic RAM-based architecture for two-dimensional discrete wavelet transform with line-based method*. In Proc. APCCAS. 2002; 1: 363-366.

[14] C Xiong, J Tian, J Liu. Efficient architectures for two-dimensional discrete wavelet transform using lifting scheme. *IEEE Trans. Image Process*. 2007; 16(3): 607-614.

[15] C Cheng, K Parhi. High-speed VLSI implementation of 2-D discrete wavelet transform. *IEEE Trans. Signal Process.* 2008; 56(1): 393-403.

[16] CH Hsia, JS Chiang, JM Guo. Memory-efficient hardware architecture of 2-D dual-mode lifting-based discrete wavelet transform. *IEEE Trans. Circuits Syst. Video Technol*. 2013; 25(4): 671-683.

[17] Ansam Ennaciri, Mohammed Erritali, Mustapha Mabrouki, Jamaa Bengourram. Comparative Study of Wavelet Image Compression:JPEG2000 Standart*. TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2015; 16(1): 83-90.

[18] Chao-Tsung Huang, Po-Chih Tseng, Liang-Gee Chen. Analysis and VLSI Architecture for 1-D and 2-D Discrete Wavelet Transform. *IEEE Transactions on Signal Processing.* 2005; 53(4): 1575-1585.

[19] Yusong Hu, Ching Chuen Jong. A Memory-Efficient Scalable Architecture for Lifting-Based Discrete Wavelet Transform. *IEEE Transactions on Circuits and Systems—II: Express Briefs.* 2013; 60(8): 502-506.

[20] Basant K Mohanty, Pramod K Meher. Memory-Efficient Architecture for 3-D DWT Using Overlapped Grouping of Frames. *IEEE Transactions on Signal Processing*. 2011; 59(11): 5605-5616.

[21] Shiva K Madishetty, Arjuna Madanayake, Renato J Cintra, Vassil S Dimitrov. Precise VLSI Architecture for AI Based 1-D/ 2-D Daub-6 Wavelet Filter Banks With Low Adder-Count. *IEEE Transactions on Circuits And Systems—I: Regular Papers.* 2014; 61(7): 1984-1993.