ISSN: 2277-9655
Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Analysis and Implementation of Lifting Scheme for Image Compression

Mrs.A.F. Mulla[*1], Dr.Smt.R.S.Patil [2]
[*1]Assistant Professor, ETC, BVC, Kolhapur, India
[2]Professor, ETX, DYPCT, Kolhapur, India
aieshamulla@yahoo.com

Abstract

This paper proposes an improved version of lifting based 2D Discrete Wavelet Transform (DWT) VLSI architecture. In this paper, high-efficient lifting-based architecture for the 5/3 discrete wavelet transform (DWT) is presented. In this architecture, all multiplications are performed using less shifts and additions. The Discrete Wavelet Transform (DWT) is based on time-scale representation. It provides efficient multi-resolution. DWT is implemented by convolution method. For such an implementation it requires a large number of computations and a large storage features that are not suitable for either high-speed or low-power applications. Hence for a high speed lifting based 2D (DWT) VLSI architecture is available. The lifting based DWT architecture has the advantage of lower computational complexities and also requires less memory. This lifting scheme has several advantages, including in-place computation of the DWT, integer-to-integer wavelet transforms (IWT), symmetric forward and inverse transform. The whole architecture is arranged in efficient way to speed up and achieve higher hardware utilization. It is desirable for high-speed VLSI applications. The most important property of this concept is simple and fast applications into FPGA chip. It requires fewer operations and provides in-place computation of the wavelet coefficients. This paper presents a method which implements 2-D lifting wavelet by FPGA. This architecture has an efficient pipeline structure to implement high-throughput processing without any on-chip memory/first in first out access. The proposed VLSI architecture is more efficient than the previous proposed architectures in terms of memory access, hardware regularity and simplicity and throughput.

Keywords: Discrete Wavelet Transform, VLSI architecture, Very-Large-Scale Integration (VLSI), High-Speed, Lifting Scheme, lifting, image compression.

Introduction

The basic goal of image compression is to represent an image with minimum number of bits of an acceptable image quality. Image compression is the application of Data compression on digital images. Data compression is the technique to reduce the redundancies in data representation in order to decrease data storage requirements and hence communication costs. Image compression schemes can be broadly classified into two types: Lossless compression and Lossy compression. In lossless compression, the image after compression and decompression is identical to the original image and every bit of information is preserved during the decompression process. The reconstructed image after compression is an exact replica of the original one. In lossy compression, the reconstructed image contains degradations with respect to the original image. Here prefect reconstruction of the image is sacrified by the elimination of some amount of redundancies in the image to achieve a higher compression ratio. In lossy compression, a higher compression ratio can be achieved when compared to lossless compression.DWT can decompose the signals into different sub bands with both time and frequency information and facilitate to arrive at high compression ratio. DWT architecture reduces the memory requirements and increases the speed of communication by breaking up the image into the blocks. A methodology for implementing lifting based DWT has been proposed because of lifting based DWT has many advantages over convolution based one. The lifting structure largely reduces the number of multiplication. FPGA is used in this system due to low cost and high computing speed with reprogrammable property.

A. LIFTING SCHEME

The lifting scheme can reduce the computational complexity and it requires fewer multipliers and adders than the convolution scheme. Andra proposed a 2-D DWT architecture which composes of simple processing units and computes one stage of DWT at a time [8].Lifting scheme is a relatively new method to construct wavelet bases. This scheme is called the second-generation wavelet which leads to a fast in-place implementation of the DWT.

B. IMPROVED WT BY LIFTING SCHEME

The main feature of the lifting based DWT scheme is to break up the high pass and low pass filters into a sequence of upper and lower triangular matrices and convert the filter implementation into banded matrix multiplications. The lifting scheme is a new algorithm proposed for the implementation of the wavelet transforms [2]. It can reduce the computational complexity of DWT involved with the convolution implementation. Furthermore, the extra memory required to store the results of the convolution can also be reduced by in place computation of the wavelet coefficient with the lifting scheme. Lifting wavelet transform has its advantages over the ordinary wavelet transform by way of reduction in memory required for its implementation. The basic principle of the lifting scheme is to factorize the poly phase matrix (using Euclidean algorithm , Laurent polynomial) of a wavelet filter into a sequence of alternating upper and lower triangular matrices and a diagonal matrix [1], [2]. This leads to the wavelet implementation by means of banded-matrix multiplications.

Euclidean algorithm –It is an efficient method for computing the greatest common divisor (GCD) of two integers, also known as the greatest common factor (GCF) or highest common factor (HCF).In its simplest form, Euclid's algorithm starts with a pair of positive integers and forms a new pair that consists of the smaller number and the difference between the larger and smaller numbers. The process repeats until the numbers are equal. That number then is the greatest common divisor of the original pair.

According to the Euclidean algorithm, the polyphase matrices can be factored into lifting steps. The polyphase matrices are produced by complementary filter pairs((h,g),(h1,g1)) .The forward wavelet transform can be factored through lifting the polyphase matrix and its dual matrix, and there always exist Laurent polynomials $s_i(z)$, $t_i(z)$ for $1<i<M$ and a nonzero constant k .

Laurent polynomial –It is one variable over a field F. It is a linear combination of positive and negative powers of the variable with coefficients in F. Laurent polynomials in X form a ring denoted $F[X, X^{-1}]$. They differ from ordinary polynomials in that they may have terms of negative degree. A Laurent polynomial with coefficients in a field F is an expression of the form

$$P = \sum_k p_k X^k, \qquad p_k \in F \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

where X is a formal variable, the summation index k is an integer (not necessarily positive) and only finitely many coefficients p_k are non-zero. Two Laurent polynomials are equal if their coefficients are equal. Such expressions can be added, multiplied, and brought back to the same form by reducing similar terms. Formulas for addition and multiplication are exactly the same as for the ordinary polynomials, with the only difference that both positive and negative powers of X can be present.

Poly phase Representation-

Euclidean algorithm is used to factorize P(z) dual poly phase factorization using Laurent polynomial.

$$P(z) = \begin{pmatrix} he(z) & ge(z) \\ ho(z) & go(z) \end{pmatrix}$$

he(z), ge(z) = the even coefficients
ho(z),go(z) = the odd coefficients

The ***poly phase*** representation of a signal is an important concept for lifting scheme. Each signal is form of ***phases***, which consist of every *N*-th sample beginning with some index. For example, if it index a time series from *n*=0 and take every other sample starting at *n*=0, it extracts the even samples. If it take every other sample starting from *n*=1, it extracts the odd samples. These are the even and odd polyphase components of the data. For lifting, it is sufficient to concentrate on the even and odd polyphase components. The Figure 1 diagram explains this operation for an input signal.

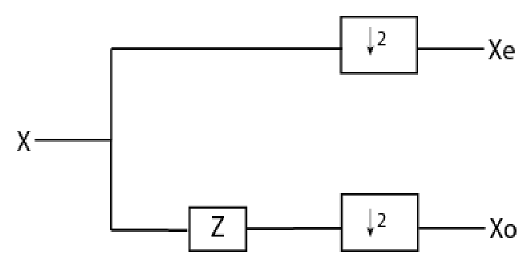


Figure 1 Poly phase input signal

where *Z* denotes the unit advance operator and the downward arrow with the number 2 represents down sampling by two. In the language of lifting, the operation of separating an input signal into even and odd components is known as the ***split*** operation, or the ***lazy wavelet.***

A lifting stage is comprised of the three steps namely Split, Predict and Update. The lifting scheme is a technique for both designing wavelets and performing the discrete wavelet transform. First, merge these steps and then design the wavelet filters while performing the

ISSN: 2277-9655
Impact Factor: 1.852

wavelet transform. This is then called the second generation wavelet transform. Lifting scheme [8] is a new approach in the construction of second generation wavelets. Second generation wavelets are those, which need not be necessarily translations and dilations of one function like first generation wavelets (classical wavelets).

Each such transform of the filter bank that is the respective operation in a wavelet transform is called a lifting step. A sequence of lifting steps consists of alternating lifts, that is, once the low pass is fixed and the high pass is changed and in the next step the high pass is fixed and the low pass is changed. Successive steps of the same direction can be merged. So, the lifting scheme is an efficient tool for constructing second generation wavelets and has advantages such as faster implementation, fully in place calculation, perfect reconstruction with low memory and low computational complexity. This is mainly because it achieves higher Compression ratios, due to the sub band decomposition it involves, while it eliminates the `blocking' artifacts that deprive the reconstructed image of the desired smoothness and continuity. The lifting scheme captures the information contained in the image. The process starts by splitting the input data into odd and even samples (split phase). Assuming that the data is smooth, the even samples used to predict the value of the odd samples by means of interpolation (predict phase). The odd samples replaced by the difference between the prediction and the original value of the odd samples. With a good prediction, the difference considered small and represented with less number of bits. The even samples are only a sub-sample of the original data. In the case of images it is important to keep the mean of the samples constant. Therefore the (update phase) updated the even samples using the newly calculated odd samples such that the mean of the data is preserved. These three steps are repeated on the even samples which resulted in splitting the even samples into half in each level, until all samples were transformed.

The main feature of the lifting based DWT scheme is to break up the high pass and low pass filters into a sequence of upper and lower triangular matrices and convert the filter implementation into banded matrix multiplications.

Triangular matrix- In the linear algebra, a triangular matrix is a special kind of square matrix. A square matrix is called lower triangular matrix if all the entries above the main diagonal are zero and a square matrix is called upper triangular matrix if all the entries below the main diagonal are zero. A triangular matrix is one that is either lower triangular or upper triangular. Due to triangular matrix, matrix equations can solve very easily. In the LU (Lower, Upper) decomposition algorithm, lower triangular matrix L is multiplied with upper triangular matrix U. This product is also called as invertible matrix.

A matrix of the form

$$L = \begin{bmatrix} l_{1,1} & & & & 0 \\ l_{2,1} & l_{2,2} & & & \\ l_{3,1} & l_{3,2} & \ddots & & \\ \vdots & \vdots & \ddots & \ddots & \\ l_{n,1} & l_{n,2} & \ldots & l_{n,n-1} & l_{n,n} \end{bmatrix} \ldots\ldots(2)$$

This is called a lower triangular matrix or left triangular matrix.

$$U = \begin{bmatrix} u_{1,1} & u_{1,2} & u_{1,3} & \ldots & u_{1,n} \\ & u_{2,2} & u_{2,3} & \ldots & u_{2,n} \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & u_{n-1,n} \\ 0 & & & & u_{n,n} \end{bmatrix} \ldots(3)$$

Thus is called an upper triangular matrix or right triangular matrix. The variable *L* (standing for lower or left) is commonly used to represent a lower triangular matrix, while the variable *U* (standing for upper) or *R* (standing for right) is commonly used for upper triangular matrix. A matrix that is both upper and lower triangular is diagonal.

Lifting Architecture

The lifting scheme is a new algorithm proposed for the implementation of the wavelet transforms [2].It can reduce the computational complexity of DWT involved with the convolution implementation. Furthermore, the extra memory required to store the results of the convolution can also be reduced by in place computation of the wavelet coefficient with the lifting scheme. Lifting wavelet transform has its advantages over the ordinary wavelet transform.

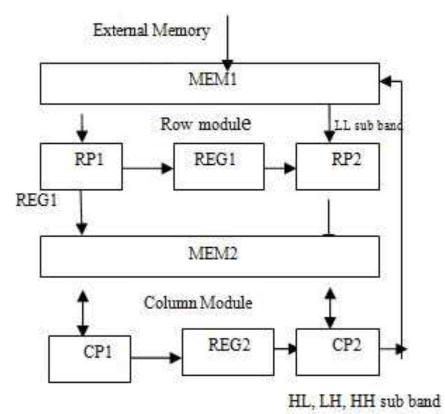


Figure 2 One Step Lifting Architecture

The architecture can compute a large set of filters for both the 2D forward and inverse transforms. It supports two classes of architectures based on whether lifting is implemented by one or two lifting steps. The one step architecture corresponds to implementation using one lifting step or two factorization matrices [3].The block diagram shows one step lifting architecture is shown in Figure 2 above. It consists of the row and column computation modules and two memory units, MEM1 and MEM2. The row module consists of two processors RP1 and RP2, along with a resister file REG1, and column module consists of two processors CP1 and CP2 along with register file REG2. All the four processor RP1, RP2, CP1, CP2 contain 2 adders, 1 multiplier and 1shifter. RP1and CP1 is predicting filters and RP2 and CP2 are update filters. The architecture can compute a large set of filters for both the 2D forward and inverse transforms. It supports two classes of architectures based on whether lifting is implemented by one or two lifting steps. The one step architecture corresponds to implementation using one lifting step or two factorization matrices [3].

Analysis and Imlplementation of Lifting Scheme

To understand lifting mathematically, it is necessary to understand the z-domain representation of the even and odd poly phase components. The z-transform of the even poly phase component is

$$X_0(z) = \sum_n x(2n)\, z^{-n} \quad \ldots\ldots (4)$$

The z-transform of the odd poly phase component is

$$X_1(z) = \sum_n x(2n+1)\, z^{-n} \quad \ldots\ldots (5)$$

The z-transform of the input signal is the sum of dilated versions of the z-transforms of the poly phase components.

$$X(z) = \sum_n x(2n)\, z^{-2n} + \sum_n x(2n+1)\, z^{-2n+1}$$

$$= X_0(z^2) + z^{-1} X_1(z^2) \quad \ldots\ldots (6)$$

Split —Split the signal into disjoint components. Extract the even and odd poly phase components as explained in Poly Phase Representation.

Predict - The even samples are multiplied by the time domain equivalent of t (z) and are added to the odd samples. Predict the odd poly phase component based on a linear combination of samples of the even polyphase component. The samples of the odd polyphase component are replaced by the difference between the odd polyphase component and the predicted value. The predict operation is also referred to as the ***dual lifting step***.

$$odd_{j+1,i} = odd_{j,i} - P(even_{j,i}) \quad \ldots\ldots (7)$$

Replace x (2n+1) with d (n) =x (2n+1)–x (2n). The predict operator is simply x (2n).The predict step replaces the odd elements with the difference between the odd elements and the predict function. The differences that replace the odd elements is high frequency components of the signal. This can be viewed as a high pass filter. The linear function "predicts" that an odd element is located at the mid-point of a line between its two even neighbors. The difference between the predicted value and the actual value of the odd element replaces the odd element.

$$Odd_{j+1,i} = odd_{j,i} - (even_{j,i} + even_{j,i+1})/2 \quad \ldots\ldots (8)$$

If here are 2^n elements, there will be 2^{n-1} even elements, or averages. An example is shown below, where $2^n = 8$.

$S_{j,0}, S_{j,1}, S_{j,2}, S_{j,3}, S_{j,4}, S_{j,5}, S_{j,6}, S_{j7}$

$$S_{j+1,0} = (S_{j,0} + S_{j,1}) / 2$$
$$S_{j+1,1} = (S_{j,2} + S_{j,3}) / 2$$
$$S_{j+1,2} = (S_{j,4} + S_{j,5}) / 2$$
$$S_{j+1,3} = (S_{j,6} + S_{j,7}) / 2$$

$$= \sum_{i=0}^{2^{n-1}-1} S_{j+1}[i] \quad \ldots\ldots (9)$$

$$= 1/2 \sum_{k=0}^{2^n-1} S_j[k] \quad \ldots\ldots (10)$$

• *Update* — Update step, where updated odd samples are multiplied, by the time domain equivalent of s (z) and are added to the even samples. Update the even poly phase component based on a linear combination of difference samples obtained from the predict step. The update step is also referred to as the ***primal lifting step***. Replace x (2n) with x (2n) +d (n)/2. This is equal to (x (2n) +x (2n+1))/2.

The update step replaces the even elements with a local average. The result is an approximation of the signal that is smoother than the signal at the previous level. This can be viewed as a low pass filter, since the smoother signals contains fewer high frequency components.

U =Update= ¼ ($odd_{j+1,i-1}$ +$odd_{j+1,i}$) ……………(11)

And forward transform is

$even_{j+1,i}$ = $even_{j,i}$ +1/4 ($odd_{j+1,i-1}$ +$odd_{j+1,i}$)……..(12)

Calculation of the update step –

$even_{j+1,i} = even_{j,i} + A\,(odd_{j+1,i-1} + odd_{j+1,i})$

Where A is constant.
The result of the update step preserves the average.

$even_{j+1,i} = even_{j,i} + A\,odd_{j+1,i-1} + A\,odd_{j+1,i}$

$Odd_{j+1,i-1} = odd_{j,i-1} - (even_{j,i-1} + even_{j,i})/2$

$Odd_{j+1,I} = odd_{j,i} - (even_{j,i} + even_{j,i+1})/2$

$even_{j+1,I} = even_{j,i} + A\,(odd_{j,i} - (even_{j,i} + even_{j,i+1})/2) + A\,(odd_{j,i-1} - (even_{j,i-1} + even_{j,i})/2)$

$even_{j+1,I} = even_{j,i} + A\,odd_{j,i} - A(even_{j,i} + even_{j,i+1})/2 + A\,odd_{j,i-1} - A\,(even_{j,i-1} + even_{j,i})/2$

$even_{j+1,i} = even_{j,i} + A\,(odd_{j,i} + A\,odd_{j,i-1}) - A/2\,(2even_{j,i} + even_{j,i+1} + even_{j,i-1})$

$even_{j+1,I} = even_{j,i} + A\,(odd_{j,i} + A\,odd_{j,i-1}) - A/2\,(2even_{j,i}) - A/2(even_{j,i+1} + even_{j,i-1})$

$even_{j+1,i} = even_{j,i} + A\,(odd_{j,i} + A\,odd_{j,i-1}) - A\,(even_{j,i}) - A/2(even_{j,i+1} + even_{j,i-1})$

$$\sum_n S_{j+1}[n] = 1/2 \sum_n even_{j,i} - A\,(even_{j,i}) - A/2\,(even_{j,i+1} + even_{j,i-1}) + A\,(odd_{j,i} + A\,odd_{j,i-1})$$

$$\sum_n S_{j+1}[n] = (1-2A) \sum_n even_{j,i} + 2A \sum_n odd_{j,i}$$

So, A = 1/4
So, in summary, the forward lifting step equation is

U = ¼ ($odd_{j+1,i-1}$ + $odd_{j+1,i}$)
And the forward transform is

$even_{j+1,i}$ = $even_{j,i}$ +1/4 ($odd_{j+1,i-1}$ +$odd_{j+1,i}$)

Lifting

Let H (Z) and G (Z) be the low pass and high pass analysis filters, and let h (z) and g (z) is the low pass and high pass synthesis filters. The corresponding poly phase matrices are defined as,

$$P_1(Z) = \begin{pmatrix} K & 0 \\ 0 & 1/k \end{pmatrix} \prod_{i=1}^{M} \begin{pmatrix} 1 & s_i(z) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ t_i(z) & 1 \end{pmatrix} \quad \text{.............(13)}$$

$$P_2(Z) = \begin{pmatrix} K & 0 \\ 0 & 1/k \end{pmatrix} \prod_{i=1}^{M} \begin{pmatrix} 1 & 0 \\ t_i(z) & 1 \end{pmatrix} \begin{pmatrix} 1 & s_i(z) \\ 0 & 1 \end{pmatrix} \quad \text{............(14)}$$

Where k is a constant, ti (z) and si (z) are denoted as primary lifting and dual lifting polynomial respectively and m represents the total lifting steps required. The dual lifting in the z domain can be written in the following matrix form

$$\begin{pmatrix} 1 & 0 \\ -t(z) & 1 \end{pmatrix} \begin{pmatrix} X_0(z) \\ X_1(z) \end{pmatrix}$$

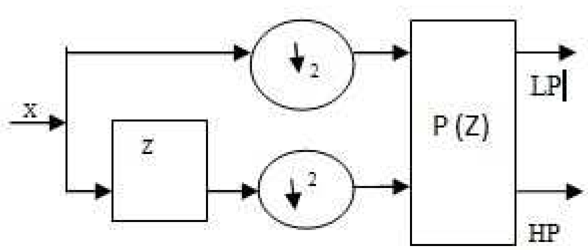
With t (z) =1
The primal lifting can be written in the z domain in the following matrix form

$$\begin{pmatrix} 1 & s(z) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ t(z) & 1 \end{pmatrix} \begin{pmatrix} X_0(z) \\ X_1(z) \end{pmatrix}$$

With s (z) =1
Finally, the primal and dual normalization can be incorporated as follows.
That is banded matrix multiplication s (z) and t (z) is Laurent polynomials.

$$\begin{pmatrix} 2^{1/2} & 0 \\ 0 & 1/2^{1/2} \end{pmatrix} \begin{pmatrix} 1 & s(z) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -t(z) & 1 \end{pmatrix} \begin{pmatrix} X_0(z) \\ X_1(z) \end{pmatrix}$$

Write the transform in the poly phase form. Lifting can be made using matrices with Laurent polynomial entries s(z) and t(z).A lifting matrix step then becomes with all diagonal entries equal to one.



ISSN: 2277-9655
Impact Factor: 1.852

Figure 3 Poly phase Matrix representation with delay

The poly phase representation of a signal is an important concept.

System Basic Block Diagram

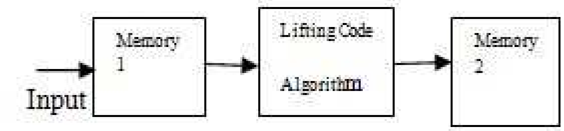


Figure 4 System Basic block diagram

To implement above scheme, first take uncompressed input image (BMP), and then use compression algorithm using Discrete Wavelet Transform (DWT). Resolution of this image is 8 bit. This is done by Memory 1.Then in lifting code algorithm, translation of different mathematical modules into functional blocks of area & timing efficient hardware (examples like multipliers, dividers etc.) is done. In this hardware, area means precise number of gate controls & thereby transistor control. Timing efficient hardware means propagation delay is reduced. Then there is partition of hardware architecture modules and micro modules (multipliers, half adder and full adder).For this each module & component, create VHDL code. After VHDL coding, develop test benches for these modules. Then, simulation & verification is done. In which, function debugging, synthesis, actual implementation, hardware testing will be done on FPGA board. The image compression of selected image is done according to JPEG 2000 with DWT. In memory 2, output is compressed image which is in RAM. Processor CP1 reads the data from MEM2, performs the column wise DWT along alternate rows, and writes the HH and LH sub bands into MEM2 and Ext.MEM. Processor CP2 reads the data from MEM2, performs the column-wise DWT along the rows on which the CP1 did not work, and writes LL sub-band to MEM1 and HL sub-band to Ext.MEM.

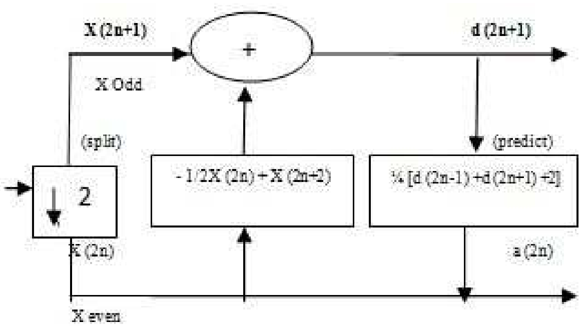


Figure 5 Lifting scheme decomposition of 5/3 filter

CDF (Cohen Daubechies Feauveau) Wavelet 5/3, 9/7 filters [5]

CDF Wavelet 5/3 Filter is also called LeGall 5/3 wavelet. It is used in JPEG as lossless Compression whereas CDF Wavelet 9/7 filter is lossy Compression.Cohen-Daubechies-Feauveau wavelets are the historically first family of wavelets. These are different than orthogonal Daubechies wavelets, in shape and properties. However their construction idea is the same.For every positive integer A, there exists a unique polynomial $Q_A(X)$ of degree A-1 satisfying the identity

$$(1-X/2)^A Q_A(X) + (X/2)^A Q_A(2-X) = 1 \quad \text{.................(15)}$$

This is the same polynomial as used in the construction of the Daubechies wavelets.

For A=1 one obtains the orthogonal Haar wavelet.
For A=2 one obtains in this way the LeGall 5/3-wavelet:
For A=4 one obtains the 9/7-CDF-wavelet.
For A = 2, $Q_A(X) = 1+X$,
For A = 4,

$$Q_A(X) = 1+2X+5/2\,X^2+5/2X^3 \quad \text{............... (16)}$$

For A=4 It is 9/7-CDF-wavelet.

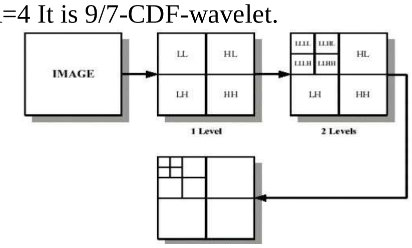


Figure 6 Three level decomposition of an image

A two dimensional digital image is represented by a 2-D array X (m, n) with m rows and n columns, where m, n are positive integers. First, a one dimensional DWT is performed on rows to get low frequency L and high frequency H components of the image. Then, once again a one dimensional DWT is performed column wise on this intermediate result to form the final DWT coefficients LL, HL, LH, HH. These are called sub-bands. The LL sub-band can be further decomposed into four sub-bands by following the above procedure. This process can continue to the required number of levels. This process is called multi level decomposition**.** High pass and low pass filters are used to decompose the image first row-wise and then column wise. Each decomposition level shown in Figure 6 comprises two stages: stage 1 performs horizontal filtering and stage 2 performs vertical filtering. In the first-level decomposition, the size of the input image is N* N, and the outputs are the three sub bands LH, HL, and HH, of size N/2*N/2.In the second-level decomposition, the

ISSN: 2277-9655
Impact Factor: 1.852

input is the LL band and the outputs are the three sub bands LLLH, LLHL, and LLHH, of size N/4*N/4.The multi-level 2-D DWT can be extended in an analogous manner. The arithmetic computation of DWT can be expressed as basic filter convolution and down sampling.

The input pixels arrive serially row-wise at one pixel per clock cycle and it will get split into even and odd. So after the manipulation with the lifting coefficients 'a' and 'b' is done, the low pass and high pass coefficients will be given out. Hence for every pair of pixel values, one high pass and one low pass coefficients will be given as output respectively. Here Modelsim tool is used in order to simulate the design and checks the functionality of the design. Once the functional verification is done, the design is taken to the Xilinx tool for Synthesis process and the net list generation.

Results

Figure 7 a) Original Lena

Figure 7 b) 1/4th compressed image of Lena

Image Quality Measures (IQM)

1] MSE –MEAN SQURE ERROR

MSE gives the difference between the original image and the reconstructed image.

Let X (m, n) denotes the samples of original image, and X^T (m n) denotes the samples of compressed image. M and N are number of pixels in row and column directions respectively .Mean Square Error is given by:

$$MSE = 1/MN \sum_{m=1}^{M} \sum_{n=1}^{N} (X(m, n) - X^T(m, n)) \ldots (17)$$

MSE for Lena is 86.02 at .08bpp bit rate.
MSE for Lena is 67.84 at .1bpp bit rate.

MSE for Cameraman is 73.31 at .08bpp bit rate.
MSE for Cameraman is 57.23 at .1bpp bit rate.

2] PSNR – PEAK SIGNAL TO NOISE RATIO

The Peak Signal to Noise Ratio (PSNR) is the quality of the reconstructed image and is the inverse of M. It is in the range between 25 to 30db within normal limit. The smaller MSE and larger PSNR values are correspond to lower levels of distortion.

$$PSNR = 10 \log_{10}((255)^2 / MSE)) \ldots\ldots\ldots\ldots (18)$$

The PSNR defined as:

$$PSNR = 10 * \log_{10} (MAX^2_I / MSE)$$
$$= 20 * \log_{10} (MAX_I / MSE^{½})$$
$$= 20 * \log_{10} (MAX_I) – 10*\log_{10} (MSE)$$

Where MAX_I = maximum possible pixel value of the image

When the pixels are 8 bits per sample, MAX_I is 255.

For Lena,

$$PSNR = 20 * \log_{10} (MAX_I) – 10*\log_{10} (MSE)$$
$$= 20* \log_{10} (255) -10 * \log_{10} (86.02)$$
$$= 28.785db$$

Similarly for MSE of Lena = 67.84, PSNR is 29.82.
For Cameraman MSE = 73.31, PSNR =23.54db

3] MD-MAXIMUM DIFFERENCE

This parameter gives maximum difference in pixel values of two images.

$$MD = Max (|X (m, n)-X^T (m, n)|) \ldots\ldots\ldots (19)$$
$$= 81 \text{ at .08bpp for Lena}$$
$$= 75 \text{ at .1bpp for Lena}$$

Conclusion

The architecture is tested for lossy compression, which is based on the lifting algorithm. The advantages of this architecture are saving embedded memories, fast computing time, low power consumption, and low control complexity. This hardware was designed to be used as part of a complete high performance and low power JPEG2000 encoder system for digital cinema applications. This Architecture could also extend to multi level DWT.

References

[1] Alice Blessie, J. Nalini and S. C. Ramesh “Image Compression Using Wavelet Transform Based on the Lifting Scheme and its Implementation”, IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 1, May 2011.

[2] Sugreev Kaur and Rajesh Mehra “High Speed and Area Efficient 2D DWT Processor Based Image Compression”, Signal and Image Processing: An International Journal (SIPIJ) Vol.1, No.2, December 2010 .DOI: 10.5121/sipij.2010.1203.

[3] A. Mansuri, A. Ahaitouf, and F. Abdi, “An Efficient VLSI Architecture and FPGA Implementation of High –Speed and Low Power 2-D DWT for (9,7) Wavelet Filter”, IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.3, March 2009 BP : 2202 FES MOROCCO.

[4] M. Jeyaprakash, “FPGA Implementation of Discrete Wavelet Transform (DWT) for JPEG 2000”, International Journal of Recent Trends in Engineering, Vol 2, No. 6, November 2009.

[5] Mountassar Maamoun, Mehdi Neggazi , Abdelhamid Meraghni,and Daoud Berkani, “VLSI Design of 2-D Discrete Wavelet Transform for Area-Efficient and High speed Image Computing”.

[6] Ali M. - Haj Department of Computer Engineering, “An FPGA-Based Parallel Distributed Arithmetic Implementation of the 1-D Discrete Wavelet Transform”, Informatics 29 (2005) 241-247, February 2004.

[7] Naseer M. Basheer, Mustafa Mushtak Mohammed, “Design and FPGA Implementation of a Lifting Scheme 2D DWT Architecture”, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume -2, Issue-1, March 2013.

[8] Kishore Andra, Chaitali Chakrabarti, Member, IEEE, and Tinku Acharya, Senior Member, IEEE, “A VLSI Architecture for Lifting-Based Forward and Inverse Wavelet Transform”, IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 50, NO. 4, APRIL 2002.

[9] Mehboob Alam, Wael Badawy, Vassil Dimitrov and Graham Jullient, “An Efficient Architecture for a Lifted 2D Biorthogonal DWT”, Journal of VLSI Signal Processing 40, 335–342, 2005 _c 2005 Springer Science + Business Media, Inc.

[10] Maria E. Angelopoulos and Peter Y. K. Cheung Konstantinos, Masselos, “Implementation and Comparison of the 5/3 Lifting 2D Discrete Wavelet Transform Computation Schedules on FPGAs”, Journal of VLSI Signal Processing 2007 Springer Science + Business Media, LLC.

[11] Hao Xu, Jizheng Xu, Member, IEEE, and Feng Wu, Senior Member, IEEE, “Lifting-Based Directional DCT-Like Transform for Image Coding”, IEEE Transactions on Circuits and Systems for Video Technology, VOL. 17, NO. 10, October 2007.

[12] G.Bheemeswara Rao, T. Surya Kavitha, U. Yedukondalu, “FPGA Implementation of JPEG2000 Image Compression using Modified DA based DWT and Lifting DWT-IDWT Technique”, International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 6, November- December 2012.