7

An Improved Low Complexity Algorithm for 2-D Integer Lifting-Based Discrete Wavelet Transform Using Symmetric Mask-Based Scheme

Chih-Hsien Hsia[1], Jing-Ming Guo[1] and Jen-Shiun Chiang[2]
[1]*Department of Electrical Engineering,*
National Taiwan University of Science and Technology
[2]*Department of Electrical Engineering, Tamkang University*
Taipei,
Taiwan

1. Introduction

Communication and multimedia have been developed rapidly in recent years. Digital media and services found in daily life include, such as digital cameras, VCD (Video Compact Disc), DVD (Digital Video Disc), HDTV (High-Definition TeleVision) and video conferences. Several well-known compression schemes, such as Differential Pulse Code Modulation (DPCM)-based method (Habibi & Hershel, 1974), DCT-based methods (Feig et al., 1995)(Kondo & Oishi, 2000), and Wavelet-based methods (Mallat, 1989) have been well-developed in recent years. The lifting-based scheme has recently provided a less-complexity solution for image/video applications, e.g., JPEG2000, Motion-JPEG2000, MPEG-4 still image coding, and MC-EZBC (Motion Compensation- Embedded Zero Block Coding). However, the real-time 2-D DWT (software-based) is still difficult to be achieved. Hence, an efficient transformation scheme for large of multimedia files is highly demanded.

Filter banks for the applications of subband image/video coding were introduced in the 1990s. Wavelet coding has been studied extensively since then. Wavelet coding has been successfully applied to many applications. The most significant applications include subband coding for audio, image, video, signal analysis and representation using wavelets. In the past few years, DWT (Mallat, 1989) has been adopted in a wide range of applications including image coding and video compression, including speech analysis, numerical analysis, signal analysis, image coding, pattern recognition, computer vision and biometrics. The DWT can be viewed as a multi-resolution decomposition of a signal, meaning which decomposes a signal into several components in different wavelet frequency bands. Moreover, 2-D DWT is a modern tool for signal processing applications, such as JPEG2000 still image compression, denoising, region of interest (ROI), and watermarking. By factoring the classical wavelet filter into lifting steps, the computational complexity of the corresponding DWT can be reduced by up to 50% (Daubechies & Sweldens, 1998). The lifting steps can be easily implemented, which is different from the direct finite impulse

response (FIR) implementations of Mallat's algorithm (Daubechies & Sweldens, 1998). Several lifting-based DWT hardware architectures have recently been proposed. The 2-D DWT architecture described by Chiang *et al.* (Chiang et al., 2005) is based on the new interlaced read scan algorithm with pipeline processing to achieve low-transpose memory size and high-speed operation. Chiang *et al.* (Chiang & Hsia, 2005) proposed a 2-D DWT folded architecture to improve the hardware utilization. Andra *et al.* (Andra et al., 2000) and (Andra et al., 2002) proposed simple processing units that compute several stages of the DWT at a time. An architecture performs the lifting-based DWT with the 5/3 filter, which is based on the interleaving technique presented in (Diou et al., 2001). Chen *et al.* (Chen & Wu, 2002) used a 1-D folded architecture to improve the hardware utilization for 2-D 5/3 and 9/7 filters. The recursive architecture is a general scheme to implement any wavelet filter that is decomposable into lifting steps (Lian et al., 2001) in small-size and low-power design. Despite these efficiency improvements of the existing architecture, further improvements in the algorithm and architecture are required. For this, Tan *et al.* (Tan & Arslan, 2003) presented a novel shift-accumulator arithmetic logic units architecture for 2-D lifting-based JPEG2000 5/3 DWT. The architecture has an efficient memory organization, which uses a smaller amount of embedded memory for processing and buffering. Lee *et al.* (Lee et al., 2003) proposed a new signal flow operation approach for the DWT implementation, and adopted only a memory size of N is employed for an $N \times N$ 2-D DWT. Varshney *et al.* (Varshney et al., 2007) presented energy efficient single-processor and fully pipelined architectures for the 2-D 5/3 lifting-based JPEG2000. The single processor performs both the row-wise and column-wise processing simultaneously, thus achieving, full 2-D transform with 100% hardware utilization. In (Chen, 2002) proposed one flexible and folded architecture for 3-level 1-D Lifting-based DWT to increase hardware utilization. Liao *et al*. (Liao et al., 2004) proposed two similar 2-D lifting-based 9/7 DWT generic architectures by employing parallel and pipeline techniques with recursive pyramid algorithms. Those architectures achieve multilevel decomposition using an interleaving scheme that reduces the size of memory and the number of memory accesses, while having a slow throughput rate and inefficient hardware utilization. Some VLSI architectures of 2-D lifting-based DWT reduce the transpose memory requirements and communication between the processors. However, these architectures need large transpose memory and long latency time.

Low-transpose memory requirement and latency reduction are the major concerns in 2-D DWT implementation. This work presents a new approach, namely 2-D Symmetric Mask-based DWT algorithm (SMDWT), to improve the 2-D lifting-based DWT (LDWT), and further applies it 2-D DWT real-time applications.

2. Lifting-based Discrete Wavelet Transform

Filtering and convolution are applied to achieve the signal decomposition in classical DWT. In 1986, Meyer and Mallat found that the orthonormal wavelet decomposition and reconstruction can be implemented in the multi-resolution signal analysis framework (Mallat, 1989). Multi-resolution analysis is now a standard method for constructing the orthonormal wavelet bases. Figure 1 shows the framework of the 2-D DWT. In the decomposition process, the low-pass filter H and high-pass filter G denote the scaling functions and the corresponding wavelets, respectively. Given a filter of length four, the corresponding transfer functions of filters H and G can be represented as,

$$H(z)=h_0+h_1z^{-1}+h_2z^{-2}+h_3z^{-3}, \quad (1)$$

$$G(z)=g_0+g_1z^{-1}+g_2z^{-2}+g_3z^{-3}. \quad (2)$$

The downsampling operation is then applied to the filtered results. A pair of filters are applied to the signal to decompose the image into the low-low (LL), low-high (LH), high-low (HL), and high-high (HH) wavelet frequency bands. Consider an image of size $N\times N$, Each band is subsampled by a factor of two, so that each wavelet frequency band contains $N/2\times N/2$ samples. The four bands can be integrated to generate an output image with the same number of samples as the original.

In most image compression applications, the above 2-D wavelet decomposition can be applied again to the *LL* sub-image, forming four new subband images, and so on to achieve a compact energy in the lower frequency bands.

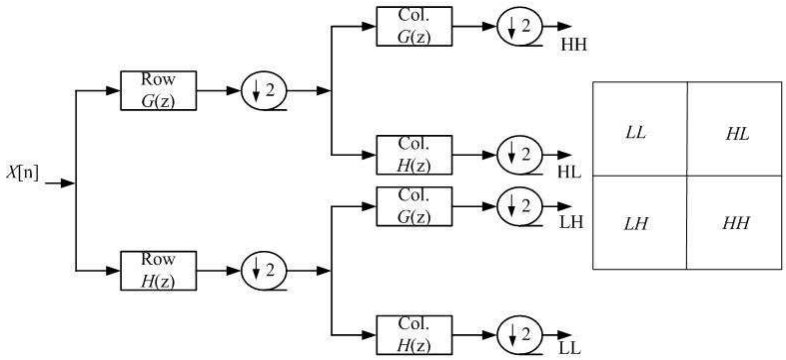


Fig. 1. The 2-D analysis DWT image decomposition process.

2.1 Lifting-based DWT algorithm

The lifting-based scheme proposed by Daubechies and Sweldens requires fewer computations than the traditional convolution-based approach (Sweldens, 1996)(Daubechies & Sweldens, 1998). The lifting-based scheme is an efficient implementation for DWT. It can easily use integer operations, and avoids the problems caused by the finite precision or rounding. The Euclidean algorithm can be used to factorize the poly-phase matrix of a DWT filter into a sequence of alternating upper and lower triangular matrices and a diagonal matrix. The variables $h(z)$ and $g(z)$ in Eq. 3 respectively denote the low-pass and high-pass analysis filters, which can be divided into even and odd parts to generate a poly-phase matrix $P(z)$ as in Eq. 4.

$$\begin{aligned} g(z)&=g_e(z^2)+z^{-1}g_o(z^2),\\ h(z)&=h_e(z^2)+z^{-1}h_o(z^2). \end{aligned} \quad (3)$$

$$P(z)=\begin{bmatrix} h_e(z) & g_e(z) \\ h_o(z) & g_o(z) \end{bmatrix}. \quad (4)$$

The Euclidean algorithm recursively finds the greatest common divisors of the even and odd parts of the original filters. Since $h(z)$ and $g(z)$ form a complementary filter pair, $P(z)$ can be factorized into Eq. 5.

$$P(z)=\prod_{i=1}^{m}\begin{pmatrix}1 & si(z)\\ 0 & 1\end{pmatrix}\begin{pmatrix}1 & 0\\ ti(z) & 1\end{pmatrix}\begin{pmatrix}k & 0\\ 0 & 1/k\end{pmatrix}. \quad (5)$$

where $si(z)$ and $ti(z)$ are Laurent polynomials corresponding to the prediction and update steps, respectively, and k is a nonzero constant. Therefore, the filter bank can be factorized into three lifting steps. As illustrated in Fig. 2, a lifting-based scheme has the following four stages:

1) Split phase: The original signal is divided into two disjoint subsets. Significantly, the variable *Xe* denotes the set of even samples and *Xo* denotes the set of odd samples. This phase is called lazy wavelet transform because it does not decorrelate the data, but only subsamples the signal into even and odd samples.

2) Predict phase: The predicting operator P is applied to the subset *Xo* to obtain the wavelet coefficients d[n] as in Eq. 6.

$$d[n]=Xo[n]+P\times(Xe[n]). \quad (6)$$

3) Update phase: Xe[n] and d[n] are combined to obtain the scaling coefficients s[n] after an update operator U as in Eq. 7.

$$s[n]=Xe[n]+U\times(d[n]). \quad (7)$$

4) Scaling: In the final step, the normalization factor is applied on s[n] and d[n] to obtain the wavelet coefficients. Equations 8 and 9 describe the implementation of the 5/3 integer lifting analysis DWT and are used to calculate the odd coefficients (high-pass coefficients) and even coefficients (low-pass coefficients), respectively.

$$d^*[n]=X(2n+1)-\lfloor X(2n)+X(2n+2)/2\rfloor \quad (8)$$

$$s^*[n]=X(2n)+\lfloor d(2n-1)+d(2n+1)+2/4\rfloor \quad (9)$$

Although the lifting-based scheme has less complexity, its long and irregular data paths constitute a major limitation for efficient hardware implementation. Additionally, the increasing number of pipelined registers increases the transpose memory size of the 2-D DWT architecture.

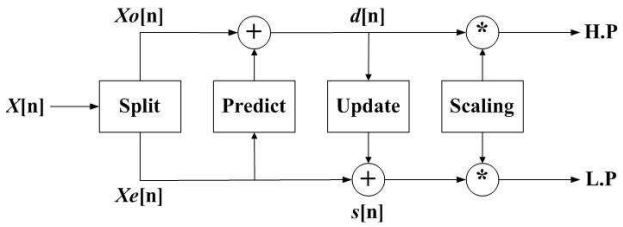
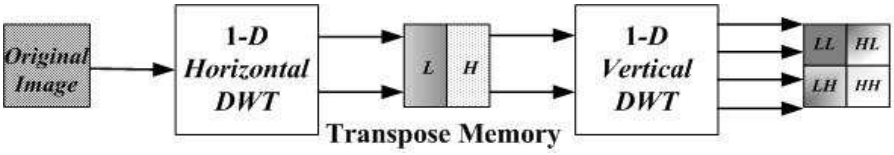


Fig. 2. Block diagram of the lifting-based DWT.

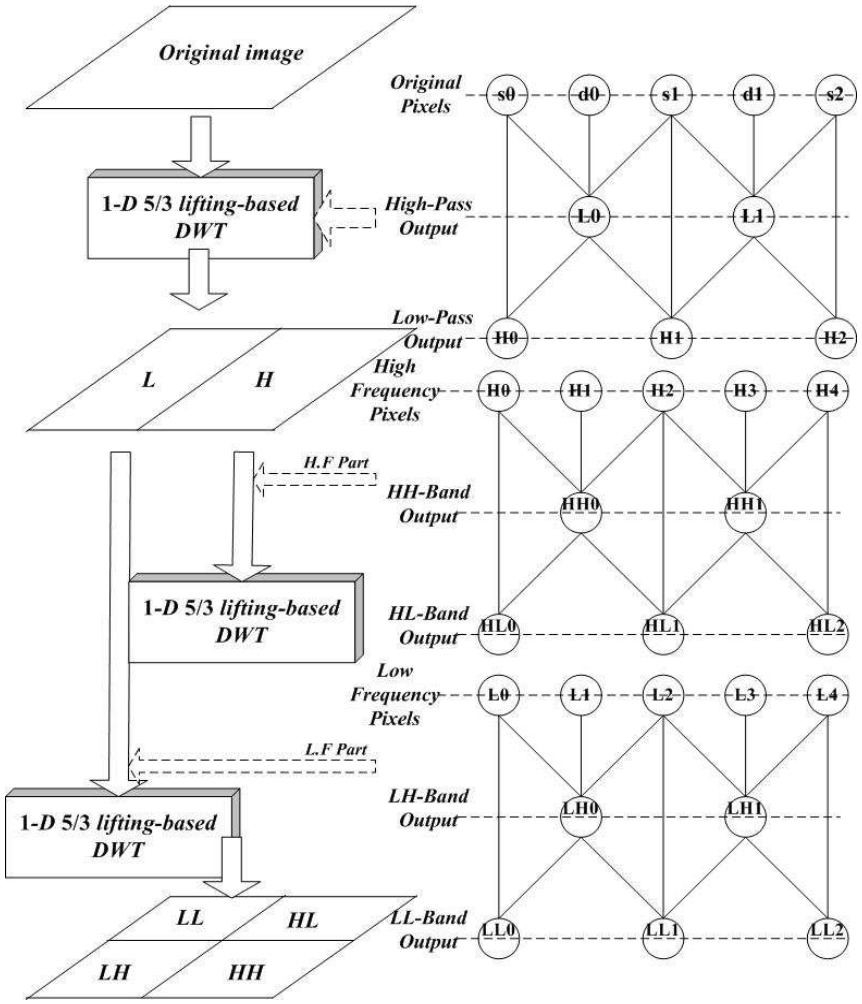
2.2 Lossless 2-D 5/3 lifting-based DWT structure

The 2-D DWT uses a vertical 1-D DWT subband decomposition and a horizontal 1-D DWT subband decomposition to obtain the 2-D DWT coefficients. Therefore, the memory

requirement dominates the hardware cost and complexity of the architectures for 2-D DWT. The 2-D transform operation is shown in Fig. 3.



(a)



(b)

Fig. 3. 2-D LDWT operation. (a) The flow of a traditional 2-D DWT. (b) Detailed processing flow.

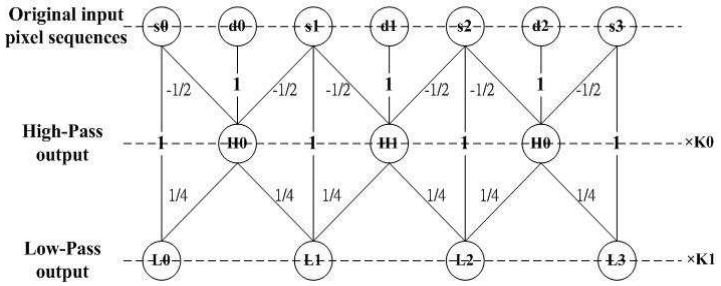


Fig. 4. Lifting-based 5/3 DWT algorithm.

Figure 4 shows the lifting step associated with the wavelet. The original signals including $s0$, $d0$, $s1$, $d1$, $s2$, $d2$, ... are the original input pixel sequences. If the original data are infinite in length, then the first-stage lifting is applied to update the odd index data $s0$, $s1$, …. In Eq. 10, the parameters -1/2 and Hi denote the first stage lifting parameter and outcome, respectively. After all the odd index data points are calculated, the second stage lifting can be performed with Eq. 11, where those parameters denote the second stage lifting parameters and outcomes, respectively. The variables Hn and Ln are the high-pass and low-pass coefficients. The values of the lifting parameters -1/2, 1, and 1/4 as shown in Fig. 4 are used for the prediction module (Hi), the update module (Li) and the Kn module (scaling by Kn=1), respectively.

$$H_i = [(s_i + s_{i+1}) \times -1/2 + d_i] \times K_0, \tag{10}$$

$$L_i = [(H_i + H_{i-1}) \times 1/4 + s_i] \times K_1, \tag{11}$$

$$K_0 = K_1 = 1. \tag{12}$$

3. The proposed 2-D symmetric mask-based Discrete Wavelet Transform

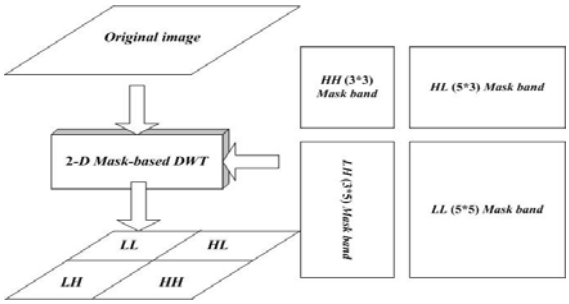
LDWT is widely employed in the visual subband coding, because it inherently has the well-known perfect reconstruction property. However, LDWT has high-transpose memory requirement and operation time in 2-D transform, as shown in Fig. 3. The memory requirement and operation speed are the two major concerns in 2-D DWT implementation. The row and column-wise signal flow operation is generally adopted for an $N \times N$ 2-D DWT. However, the memory requirement of this scheme ranges from $2.5N$ to N^2. To solve the transpose memory access problem, this work proposes a low-latency and low-memory architecture for 1-level 2-D lifting-based DWT. The previous signal flow from row- and column-wise is replaced with mask-based processing, SMDWT, to reduce the transpose memory requirement for the 2-D DWT. The SMDWT has many advanced features, such as short critical path, less latency time, regular signal coding, and independent subband processing. The following subsections introduce the 2-D SMDWT where the coefficients of mask wavelet coefficient derivation are based on the 2-D 5/3 integer lifting-based DWT.

3.1 The 2-D SMDWT structure

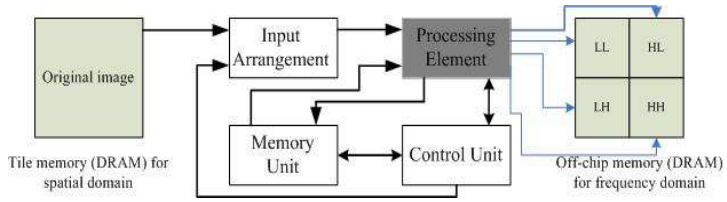
This sub-section, the proposed SMDWT is discussed in three aspects: lifting structure, transpose memory, as well as latency and critical path. The proposed SMDWT algorithm

has the advantages of fast computational speed, less complexity, reduced latency, and regular data flow.

For speed and simplicity, four-masks, 3×3, 5×3, 3×5, and 5×5, are generally used to perform spatial filtering tasks. Moreover, the four-subband processing can be further optimized to speed up and reduce the transpose memory of DWT coefficients. The four-matrix processors consist of four mask filters, and each filter is derived from one 2-D DWT of 5/3 integer lifting-based coefficients. In LDWT implementation, a 1-D DWT needs massive computations, so the computation unit dominants the hardware cost (Chiang & Hsia, 2005)(Andra et al., 2002). A 2-D DWT is compose of two 1-D DWTs and a block of transpose memory, which is of the same size of the processed image. The transpose memory is the main overhead of the computation unit in the 2-D DWT. Figure 3 shows the block diagram of a traditional 2-D DWT. Without loss of generality, the 5/3 lifting-based 2-D DWT is adopted for comparison. Assuming that the image is of size $N \times N$, during the transformation, a large amount of transpose memory (order of N^2) is needed to store the temporary data after the first stage 1-D DWT decomposition. The second stage 1-D DWT is then applied to the stored data to obtain the four-subband (HH, HL, LH, and LL) results of the 2-D DWT. Because the memory requirement of size N^2 is huge and the processing is too long, this work proposes a new approach, called 2-D SMDWT, to reduce the transpose computing latency and critical path. Figure 5(a) shows the concept of the proposed SMDWT architecture, which consists of input arrangement, processing element, memory unit, and control unit, as shown in Fig. 5(b). The outputs are fed to the 2-D DWT four-subband coefficients, HH, HL, LH, and LL. Significant transpose memory can be saved using the proposed approach. This architecture is described in detail in the following subsections, and is illustrated in Figs. 5, 7(c), 8(c), 11(c), and 14(c). This study focuses on the 5/3 lifting-based 2-D DWT complexity reduction.



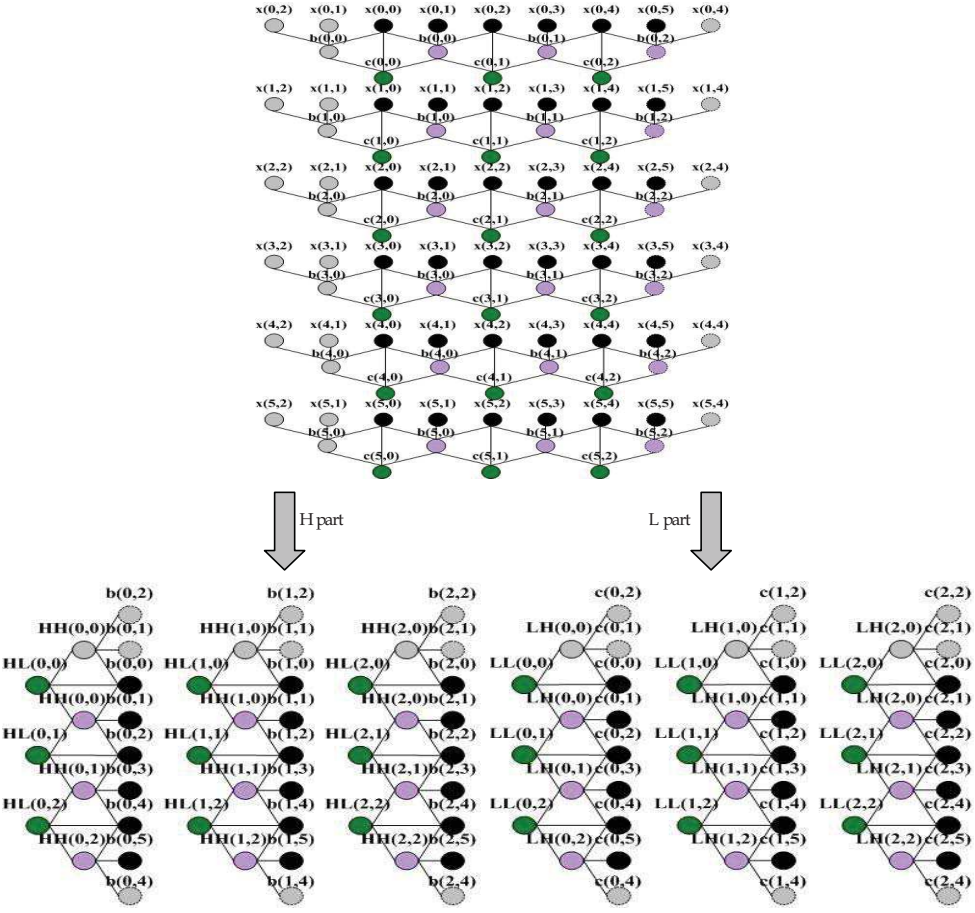
(a)



(b)

Fig. 5. The system block diagram of the proposed 2-D DWT. (a) 2-D SMDWT. (b) Block diagram of the proposed system architecture.

Without loss of generality, let us take a 6×6-pixel image is employed to demonstrate the 5/3 LDWT operations as shown in Fig. 6. In Fig. 6, the variable $x(i,j)$ denotes the original image. The upper part of Fig. 6 shows the first stage 1-D LDWT operations, and the lower part of Fig. 6 shows the second stage 1-D LDWT operations for evaluating the four-subband coefficients, HH, HL, LH, and LL. In the first stage of the 1-D LDWT, three pixels are used to evaluate a 1-D high-frequency coefficient. For example, $x(0,0)$, $x(0,1)$, and $x(0,2)$ are used to calculate the high-frequency wavelet coefficient $b(0,0)$, where



$x(i,j)$: original image, i = 0~5 and j = 0~5
$b(i,j)$: high frequency wavelet coefficient of 1-D LDWT
$c(i,j)$: low frequency wavelet coefficient of 1-D LDWT
HH: high-high frequency wavelet coefficient of 2-D LDWT
HL: high-low frequency wavelet coefficient of 2-D LDWT
LH: low-high frequency wavelet coefficient of 2-D LDWT
LL: low-low frequency wavelet coefficient of 2-D LDWT

Fig. 6. Example of 5/3 LDWT operations.

$b(0,0)=-[x(0,0)+x(0,2)]/2+x(0,1)$. The pixels, $x(0,2)$, $x(0,3)$, and $x(0,4)$ are used to calculate the next high-frequency wavelet coefficient $b(0,1)$. Herein $x(0,2)$ is used to calculate both of $b(0,0)$ and $b(0,1)$, and is called the overlapped pixel. The low-frequency wavelet coefficient is calculated using two consecutive high-frequency wavelet coefficients and the overlapped pixel. For example, $b(0,0)$ and $b(0,1)$ cope with $x(0,2)$ to find the low-frequency wavelet coefficient $c(0,1)$, where $c(0,1)=[b(0,0)+b(0,1)]/4+x(0,2)$. The calculated high-frequency wavelet coefficients, $b(i,j)$, and the low frequency wavelet coefficients, $c(i,j)$, are then used in the second stage 1-D LDWT to calculate the four-subbands coefficients, HH, HL, LH and LL. The general form of the mask coefficients is derived first, and the complexity is further reduced by employing the symmetric feature of the mask.

3.2 Simplified 2-D SMDWT using symmetric features

1. High-High (HH) band mask coefficients reduction for 2-D SMDWT

According to the 2-D 5/3 LDWT, the HH band coefficients of the SMDWT can be derived as follows:

$$HH(i,j)=x(2i+1,2j+1)+(1/4)\sum^{1}_{u=0}\sum^{1}_{v=0}x(2i+2u,2j+2v)+(-1/2)\sum^{2}_{u=-1}x(2i+|u|,2j+|1-u|). \quad (13)$$

The mask as shown in Fig. 7(a) can be obtained by Eq. 13, where the variables $\alpha=-1/2$, $\beta=1/4$, and $\gamma=1$. Figure 7(b) shows the DSP architecture and Fig. 7(c) shows the hardware architecture.

The transpose memory requirement is a very important issue in multimedia IC design. Therefore, to make the SMDWT architecture suitable for VLSI implementation, the design processing element must be as simple and modular as possible. However, the product of cost and computation time is always the most important consideration from a standardization provides economies of scale for VLSI solution point of view. Therefore, speed is sometimes sacrificed to obtain less cost hardware, while still satisfying the performance requirement. In other words, the SMDWT architecture can be decomposed so as to adjust the cost and computation time product. Its hardware cost and computation time tradeoffs must be carefully considered to find the optimal design for VLSI implementation. A simple SMDWT method for cost and computation time savings is introduced below.

Figure 7(c) shows the concept of the proposed HH-band architecture for SMDWT. The proposed HH-band architecture consists of a shifter (α, β, and γ) and one adder tree with propagation registers, as shown in Fig. 7(c). The architecture design can be divided as follows:

- Input arrangement unit: Three pixels in a column are inputted into a processing element for address generator circuits in each cycle. Simultaneously, the input arrangement to assign input original signals used in multiplexer (MUX) fetch 3 pixels in each cycle to switch for group 1, group 2 and group 3 to operations, respectively.
- Coefficient shifter unit: The coefficient shifter values are $\alpha=-1/2$, $\beta=1/4$, and $\gamma=1$. Shifters replace multipliers to achieve a high-efficiency architecture by (reducing computational time, critical path, area cost and power consumption (Tan & Arslan, 2003)).
- Adder tree unit: An adder tree architecture is adopted to avoid the long signal path length, signal skewing, and hazards caused by signal dependency. Each adder tree level can be viewed as a parallel pipeline stage. This architecture is suitable for the realization in hardware design.

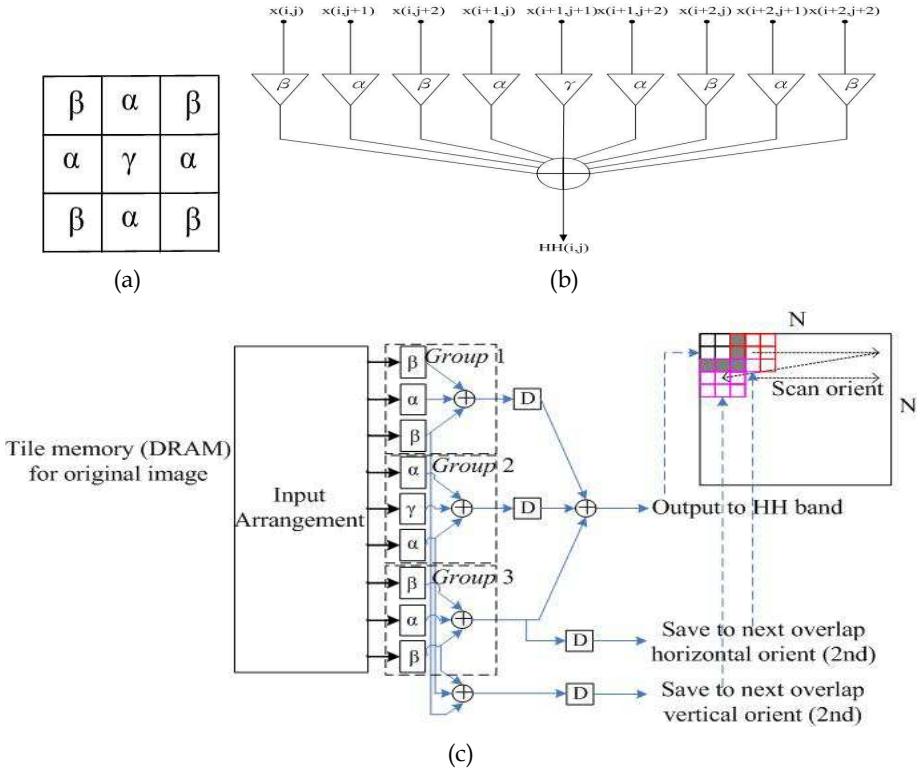


Fig. 7. HH band mask coefficients and the corresponding DSP architecture. (a) Coefficients. (b) DSP architecture. (c) Hardware architecture design.

- Propagation register unit: Current pixels are stored to assign subband coefficients computation needs in each group, and next horizontal or vertical scan oriented computation are stored in propagation registers for data reuse. This approach can reduce the next access time and computations. The pipeline design is the best method to improve the system throughput.

Based on this structure, the coefficient overlap part can be reused as show in Fig. 7(c).
The complexity of the mask-based method is further reduced by employing the symmetric feature of the mask. First, the initial horizontal scan is expressed by:

$$HH(0,0)=\beta\times x(0,0)+\alpha\times x(0,1)+\underline{\beta\times x(0,2)}+\alpha\times x(1,0)+\gamma\times x(1,1)$$
$$+\underline{\alpha\times x(1,2)}+\beta\times x(2,0)+\alpha\times x(2,1)+\underline{\beta\times x(2,2)} \quad (14)$$

The next coefficient can be calculated by:

$$HH(0,1)=\underline{\beta\times x(0,2)}+\alpha\times x(0,3)+\beta\times x(0,4)+\underline{\alpha\times x(1,2)}+\gamma\times x(1,3)+\alpha\times x(1,4)+\underline{\beta\times x(2,2)}+\alpha\times x(2,3)+\beta\times x(2,4)$$
$$=\alpha\times x(0,3)+\beta\times x(0,4)+\gamma\times x(1,3)+\alpha\times x(1,4)+\alpha\times x(2,3)+\beta\times x(2,4)+XM_H$$
$$=\beta\times(x(0,4)+x(2,4))+\alpha\times(x(0,3)+x(1,4)+x(2,3))+\gamma\times x(1,3)+XM_H, \quad (15)$$

where the variable XM_H denotes the repeated part after the horizontal third coefficient, where X denotes group of pixels x, M denotes the mask, and H denotes horizontal orientation. The general form can be derived as:

$$XM_H=\beta\times x(i,2j+2)+\alpha\times x(i+1,2j+2)+\beta\times x(i+2,2j+2). \quad (16)$$

Since γ=1, the general form can be expressed as:

$$HH(i,j+1)=\beta\times(x(i,2j+4)+x(i+2,2j+4))+\alpha\times(x(i,2j+3)+x(i+1,2j+4)+x(i+2,2j+3))+x(i+1,2j+3)+XM_H, \quad (17)$$

where i=0~N-1, j=0~N-2.
The vertical scan can be done in the same way, where HH(0,0) is the same as that in Eq. 14. The next coefficient can be calculated by:

$$HH(1,0)=\underline{\beta\times x(2,0)}+\underline{\alpha\times x(2,1)}+\underline{\beta\times x(2,2)}+\alpha\times x(3,0)+\gamma\times x(3,1)+\alpha\times x(3,2)+\beta\times x(4,0)+\alpha\times x(4,1)+\beta\times x(4,2)$$

$$=\alpha\times x(3,0)+\beta\times x(4,0)+\gamma\times x(3,1)+\alpha\times x(4,1)+\alpha\times x(3,2)+\beta\times x(4,2)+XM_V, \quad (18)$$

where the variable XM_V denotes the repeated part after the vertical third coefficient, where V denotes vertical orientation. The general form can be derived as:

$$XM_V=\beta\times x(2i+2,j)+\alpha\times x(2i+2,j+1)+\beta\times x(2i+2,j+2). \quad (19)$$

Since γ=1, the general form can be expressed as:

$$HH(i+1,j)=\beta\times(x(2i+4,j)+x(2i+4,j+2))+\alpha\times(x(2i+3,j)+x(2i+4,j+1)+x(2i+3,j+2))+x(2i+3,j+1)+XM_V. \quad (20)$$

where i=0~N-1, j=0~N-2.
Finally, the diagonal oriented scan can be derived as:

$$HH(1,1)=\underline{\beta\times x(2,2)}+\underline{\alpha\times x(2,3)}+\underline{\beta\times x(2,4)}+\alpha\times x(3,2)+\gamma\times x(3,3)+\alpha\times x(3,4)+\beta\times x(4,2)+\alpha\times x(4,3)+\beta\times x(4,4)$$

$$=\gamma\times x(3,3)+\alpha\times x(3,4)+\alpha\times x(4,3)+\beta\times x(4,4)+XM_D$$

$$=\beta\times x(4,4)+\alpha\times(x(3,4)+x(4,3))+\gamma\times x(3,3)+XM_D, \quad (21)$$

where the variable XM_D denotes the repeated part after the vertical fifth coefficient, where D denotes diagonal orientation. The general form can be expressed as:

$$XM_D=\beta\times x(2i+2,2j+2)+\alpha\times x(2i+2,2j+3)+\beta\times x(2i+2,2j+4)+\alpha\times x(2i+3,2j+2)+\beta\times x(2i+4,2j+2). \quad (22)$$

Since γ=1, the general form can be expressed as:

$$HH(i+1,j+1)=\beta\times x(2i+4,2j+4)+\alpha\times(x(2i+3,2j+4)+\beta\times x(2i+4,2j+3))+x(2i+3,2j+3)+XM_D. \quad (23)$$

where i=0~N-1, j=0~N-2.
The repeat part is only needed to be calculated once throughout the whole image. Hence it greatly reduces the complexity of the SMDWT.

2. High-Low (HL) and Low-High (LH) band mask coefficients reduction for 2-D SMDWT

According to the 2-D 5/3 lifting-based DWT, the HL-band coefficients of the mask-based DWT can be expressed as follows:

$$HL(i,j)=(3/4)x(2i+1,2j)+(1/16)\sum^{1}_{u=0}\sum^{1}_{v=0}x(2i+4u,2j-2+2v)+(-1/8)\sum^{1}_{u=0}x(2i+4u,2j)$$

$$+(-1/8)\sum^{1}_{u=0}\sum^{1}_{v=0}x(2i+2u,2j-1+2v)+$$

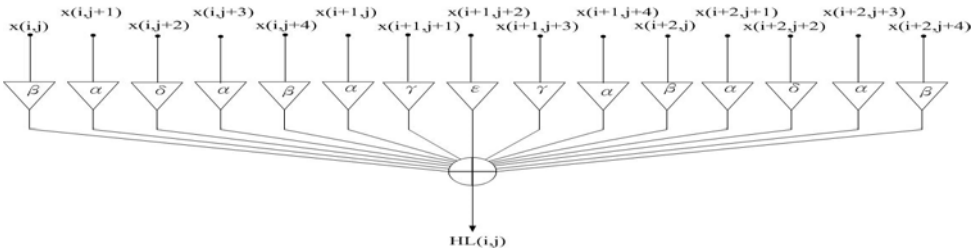
$$(1/4)\sum^{1}_{u=0}x(2i+1,2j-1+2u)+(-3/8)\sum^{1}_{u=0}x(2i+2u,2j). \quad (24)$$

The mask as shown in Fig. 8(a) can be obtained via Eq. 24, where α=-1/8, β=1/16, γ=1/4, δ=-3/8, and ε=3/4. The DSP and hardware architecture are also depicted in Figs. 8(b) and (c). The complexity of the SMDWT is further reduced by employing the symmetric feature of the mask.

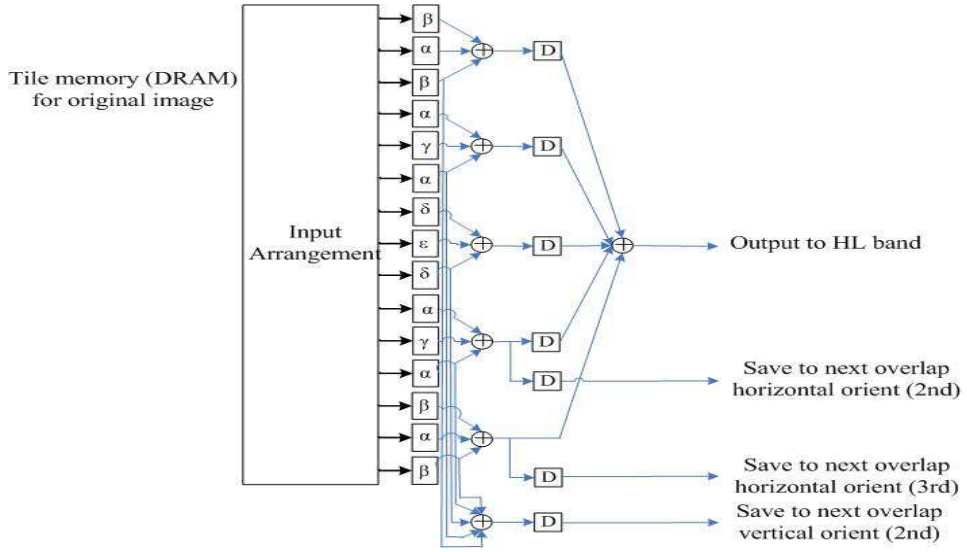
The initial horizontal scan is expressed by:

β	α	δ	α	β
α	γ	ε	γ	α
β	α	δ	α	β

(a)



(b)



(c)

Fig. 8. HL band mask coefficients and the corresponding DSP architecture. (a) Coefficients. (b) DSP architecture. (c) Hardware architecture design.

$$HL(0,0)=\beta\times x(0,0)+\alpha\times x(0,1)+\delta\times x(0,2)+\underline{\alpha\times x(0,3)}+\underline{\beta\times x(0,4)}+\alpha\times x(1,0)+\gamma\times x(1,1)+\varepsilon\times x(1,2)+$$
$$+\gamma\times x(1,3)+\alpha\times x(1,4)+\beta\times x(2,0)+\alpha x(2,1)+\delta\times x(2,2)+\alpha\times x(2,3)+\beta\times x(2,4)$$
$$=\beta\times x(0,0)+\alpha\times x(0,1)+\delta\times x(0,2)+\underline{\beta\times x(0,4)}+\alpha\times x(1,0)+\gamma\times x(1,1)+\varepsilon\times x(1,2)+$$
$$+\underline{\alpha\times x(1,4)}+\beta\times x(2,0)+\alpha\times x(2,1)+\delta\times x(2,2)+\underline{\beta\times (2,4)}+XM_{H+1}, \quad (25)$$

where the variable XM_{H+1} denotes the repeated part after the first horizontal coefficient. The next coefficient can be calculated as:

$$HL(0,1)=\beta\times x(0,2)+\underline{\alpha\times x(0,3)}+\delta\times x(0,4)+\underline{\alpha\times x(0,5)}+\beta\times x(0,6)+\alpha\times x(1,2)+\underline{\gamma\times x(1,3)}+\varepsilon\times x(1,4)+$$
$$+\underline{\gamma\times x(1,5)}+\alpha\times x(1,6)+\beta\times x(2,2)+\underline{\alpha x(2,3)}+\delta\times x(2,4)+\underline{\alpha\times x(2,5)}+\beta\times x(2,6)$$
$$=\beta\times x(0,2)+\delta\times x(0,4)+\underline{\alpha\times x(0,5)}+\beta\times x(0,6)+\alpha\times x(1,2)+\varepsilon\times x(1,4)+\underline{\gamma\times x(1,5)+}$$
$$+\alpha\times x(1,6)+\beta\times x(2,2)+\delta\times x(2,4)+\underline{\alpha\times x(2,5)}+\beta\times (2,6)+XM_{H+1}, \quad (26)$$

The general form of the first horizontal step can be derived as:

$$HL(i,1)=\beta\times x(i,j+2)+\delta\times x(i,j+4)+\alpha\times x(i,j+5)+\beta\times x(i,j+6)+\alpha\times x(i+1,j+2)+\varepsilon\times x(i+1,j+4)+$$
$$+\gamma\times x(i+1,j+5)+\alpha\times x(i+1,j+6)+\beta\times x(i+2,j+2)+\delta\times x(i+2,j+4)+\alpha\times x(i+2,j+5)+\beta\times x(i+2,j+XM_{H+1}, \quad (27)$$

where i=0~N-1, and

$$XM_{H+1}=\alpha\times x(i,3)+\gamma\times x(i+1,3)+\alpha\times x(i+2,3). \quad (28)$$

The next coefficient can be calculated as:

$$HL(0,2)=\underline{\beta\times x(0,4)}+\underline{\alpha\times x(0,5)}+\delta\times x(0,6)+\alpha\times x(0,7)+\beta\times x(0,8)+\underline{\alpha\times x(1,4)}+\underline{\gamma\times x(1,5)+}$$
$$+\varepsilon\times x(1,6)+\gamma\times x(1,7)+\alpha\times x(1,8)+\underline{\beta\times x(2,4)}+\underline{\alpha x(2,5)}+\delta\times x(2,6)+\alpha\times x(2,7)$$
$$+\beta\times x(2,8)=\delta\times x(0,6)+\alpha\times x(0,7)+\beta\times x(0,8)+\varepsilon\times x(1,6)+\gamma\times x(1,7)+\alpha\times x(1,8)+$$
$$+\delta\times x(2,6)+\alpha\times x(2,7)+\beta\times x(2,8)+XM_{H+n}, \quad (29)$$

where the variable XM_{H+n} denotes the repeated part after the second horizontal coefficient. From Eq. 29, the general form can be expressed as:

$$HL(i,j+2)=\delta\times x(i,2j+6)+\alpha\times x(i,2j+7)+\beta\times x(i,2j+8)+\varepsilon\times x(i+1,2j+6)+\gamma\times x(i+1,2j+7)+\alpha\times x(i+1,2j+8)+$$
$$+\delta\times x(i+2,2j+6)+\alpha\times x(i+2,2j+7)+\beta\times x(i+2,2j+8)+XM_{H+n}, \quad (30)$$

where i=0~N-1, j=0~N-2, and

$$XM_{H+n}=\beta\times x(i,2j+4)+\alpha\times x(i,2j+5)+\alpha\times x(i+1,2j+4)+\gamma\times x(i+1,2j+5)+\beta\times x(i+2,2j+4)+\alpha\times x(i+2,2j+5). \quad (31)$$

The vertical scan can be done in the same way, where HL(0,0) is the same as that in Eq. 25. The next coefficient can be calculated as:

$$HL(1,0)=\beta\times x(2,0)+\alpha\times x(2,1)+\delta\times x(2,2)+\alpha\times x(2,3)+\beta\times x(2,4)+\alpha\times x(3,0)+\gamma\times x(3,1)+$$

$$+\varepsilon\times x(3,2)+\gamma\times x(3,3)+\alpha\times x(3,4)+\beta\times x(4,0)+\alpha\times(4,1)+\delta\times x(4,2)+\alpha\times x(4,3)+\beta\times x(4,4)$$

$$=\alpha\times x(3,0)+\gamma\times x(3,1)+\varepsilon\times x(3,2)+\gamma\times x(3,3)+\alpha\times x(3,4)+\beta\times x(4,0)+\alpha\times x(4,1)+$$

$$+\delta\times x(4,2)+\alpha\times x(4,3)+\beta\times x(4,4)+XM_V, \quad (32)$$

where the variable XM_V denotes the repeated part after the vertical fifth coefficient. The general form can be expressed as:

$$HL(i+1,j)=\alpha\times x(2i+3,j)+\gamma\times x(2i+3,j+1)+\varepsilon\times x(2j+3,j+2)+\gamma\times x(2j+3,j+3)+\alpha\times x(2j+3,j+4)+$$

$$+\beta\times x(2j+4,j)+\alpha\times x(2j+4,j+1)+\delta\times x(2j+4,j+2)$$

$$+\alpha\times x(2j+4,j+3)+\beta\times x(2j+4,j+4)+XM_V, \quad (33)$$

where i=0~N-1, j=0~N-1, and

$$XM_V=\beta\times x(2i+2,j)+\alpha\times x(2i+2,j+1)+\delta\times x(2i+2,j+2)+\alpha\times x(2i+2,j+3)+\beta\times x(2i+2,j+4). \quad (34)$$

Finally, the diagonal oriented scan can be expressed as:

$$HL(1,1)=\beta\times x(2,2)+\alpha\times x(2,3)+\delta\times x(2,4)+\alpha\times x(2,5)+\beta\times x(2,6)+\alpha\times x(3,2)+\gamma\times x(3,3)+\varepsilon\times x(3,4)+$$

$$+\gamma\times x(3,5)+\alpha\times x(3,6)+\beta\times x(4,2)+\underline{\alpha x(4,3)}+\delta\times x(4,4)+\alpha\times x(4,5)+\beta\times x(4,6)$$

$$=\alpha\times x(3,2)+\varepsilon\times x(3,4)+\gamma\times x(3,5)+\alpha\times x(3,6)+\beta\times x(4,2)+\delta\times x(4,4)+\alpha\times x(4,5)+\beta\times x(4,6)+XM_{D+1}, \quad (35)$$

where the variable XM_{D+1} denotes the repeated part as shown in the gray part of Fig. 9 after the first diagonal scan. Next, the HL(2,2) is calculated as:

$$HL(2,2)=\beta\times x(4,4)+\alpha\times x(4,5)+\delta\times x(4,6)+\alpha\times x(4,7)+\beta\times x(4,8)+\alpha\times x(5,4)+\gamma\times x(5,5)+$$

$$+\varepsilon\times x(5,6)+\gamma\times x(5,7)+\alpha\times x(5,8)+\underline{\beta\times x(6,4)}+\underline{\alpha\times(6,5)}+\delta\times x(6,6)+\alpha\times x(6,7)+\beta\times x(6,8)$$

$$=\varepsilon\times x(5,6)+\gamma\times x(5,7)+\alpha\times x(5,8)+\delta\times x(6,6)+\alpha\times x(6,7)+\beta\times x(6,8)+XM_{D+n}, \quad (36)$$

x(2,2)	x(2,3)	x(2,4)	x(2,5)	x(2,6)
x(3,2)	x(3,3)	x(3,4)	x(3,5)	x(3,6)
x(4,2)	x(4,3)	x(4,4)	x(4,5)	x(4,6)

Fig. 9. Repeat part (in gray) of the diagonal scanned position HL(1,1).

where the variable XM_{D+n} denotes the repeated part as shown in the gray part of Fig. 10 after the first diagonal scan. The general form of XM_{D+n} can be expressed as:

$$XM_{D+n}=\beta\times x(2i+4,2i+4)+\alpha\times x(2i+4,2i+5)+\delta\times x(2i+4,2i+6)+\alpha\times x(2i+4,2i+7)+\beta\times$$

$$\times x(2i+4,2i+8)+\alpha\times x(2i+5,2i+4)+\gamma\times x(2i+5,2i+5)+\varepsilon\times x(2i+5,2i+6)+$$

$$+\gamma\times x(2i+5,2i+7)+\alpha\times x(2i+5,2i+8)+\beta\times x(2i+6,2i+4)+\alpha\times x(2i+6,2i+5)+$$

$$+\delta\times x(2i+6,2i+6)+\alpha\times x(2i+6,2i+7)+\beta\times x(2i+6,2i+8), \quad (37)$$

x(4,4)	x(4,5)	x(4,6)	x(4,7)	x(4,8)
x(5,4)	x(5,5)	x(5,6)	x(5,7)	x(5,8)
x(6,4)	x(6,5)	x(6,6)	x(6,7)	x(6,8)

Fig. 10. Repeat part (in gray) of the diagonal scanned position HL(2,2).

The general form of the rest part can be expressed as:

$$HL(i+1,j+1)=\beta\times x(2i+6,2j+8)+\alpha\times(x(2i+5,2j+8)+x(2i+6,2j+7))+\gamma\times x(2i+5,2j+7)+$$
$$+\delta\times x(2i+6,2j+6)+\varepsilon\times x(2i+5,2j+6)+XM_{D+n}, \quad (38)$$

where i=1~N-1, j=1~N-1.

The HL-band can be derived in the same way. According to the 2-D 5/3 LDWT, the LH-band coefficients of the SMDWT can be derived as follows:

$$LH(i,j)=(3/4)x(2i,2j+1)+(1/16)\sum^{1}_{u=0}\sum^{1}_{v=0}x(2i-2+2u,2j+4v)+$$
$$+(-1/8)\sum^{1}_{u=0}x(2i,2j+4u)+(-1/8)\sum^{1}_{u=0}\sum^{1}_{v=0}x(2i-1+2u,2j+2v)+$$
$$(1/4)\sum^{1}_{u=0}x(2i-1+2u,2j+1)+(-3/8)\sum^{1}_{u=0}x(2i,2j+2u). \quad (39)$$

The mask as shown in Fig. 11(a) can be obtained via Eq. 39, where α=-1/8, β=1/16, γ=1/4, δ=-3/8, and ε=3/4. The DSP and hardware architecture are depicted in Figs. 11(b) and (c). The complexity of the SMDWT is further reduced by employing the symmetric feature of the mask. First, the initial horizontal scan is calculated by the method that is similar to that of HL SMDWT, where the variable XM_H denotes the repeated part after the horizontal fifth coefficient. The general form can be expressed as:

$$LH(i,j+1)=\alpha\times x(i,2j+3)+\beta\times x(i,2j+4)+\gamma\times x(i+1,2j+3)+\alpha\times x(i+1,2j+4)+\varepsilon\times x(i+2,2j+3)+$$
$$+\delta\times x(i+2,2j+4)+\gamma\times x(i+3,2j+3)+$$
$$\alpha\times x(i+3,2j+4)+\alpha\times x(i+4,2j+3)+\beta\times x(i+4,2j+4)+XM_H, \quad (40)$$

where i=0~N-1, j=0~N-1, and

$$XM_H=\beta\times x(i,2j+2)+\alpha\times x(i+1,2j+2)+\delta\times x(2i+2,j+2)+\alpha\times x(i+3,2j+2)+\beta\times x(i+4,2j+2). \quad (41)$$

Next, the initial vertical scan is calculated by the method similar to that of HL mask-based DWT, where the variable XM_{V+1} denotes the repeated part after the vertical first coefficient. The general form of the first vertical step can be expressed as:

$$LH(1,j)=\beta\times x(i+2,j)+\alpha\times x(i+2,j+1)+\beta\times x(i+2,j+2)+\delta\times x(i+4,j)+\varepsilon\times x(i+4,j+1)+\delta\times x(i+4,j+2)+$$
$$+\alpha\times x(i+5,j)+\gamma\times x(i+5,j+1)+\alpha\times x(i+5,j+2)+$$
$$+\beta\times x(i+6,j)+\alpha\times x(i+6,j+1)+\beta\times x(i+6,j+2)+XM_{V+1}, \quad (42)$$

where i=0, j=0~N-1, and

$$XM_{V+1}=\alpha\times x(2i+3,0)+\gamma\times x(2i+3,1)+\alpha\times x(2i+3,2). \quad (43)$$

Next, the second vertical scan is calculated with the method similar to that of HL SMDWT.

$$LH(i+2,j)=\delta\times x(2i+6,j)+\varepsilon\times x(2i+6,j+1)+\delta\times x(2i+6,j+2)+\alpha\times x(2i+7,j)+\gamma\times x(2i+7,j+1)+$$
$$+\alpha\times x(2i+7,j+2)+\beta\times x(2i+8,j)+\alpha\times x(2i+8,j+1)+\beta\times x(2i+8,j+2)+XM_{V+n}, \quad (44)$$

where i=0~N-1, j=0~N-2, and

$$XM_{V+n}=\beta\times x(2i+4,j)+\alpha\times x(2i+4,j+1)+\beta\times x(2i+4,j+2)+\alpha\times x(2i+5,j)+\gamma\times x(2i+5,j+1)+\alpha\times x(2i+5,j+2). \quad (45)$$

Finally, the diagonal oriented scan can be derived as:

$$LH(1,1)=\alpha\times x(3,4)+\varepsilon\times x(4,3)+\delta\times x(4,4)+\alpha\times x(5,2)+\gamma\times x(5,3)+\beta\times x(6,2)+\alpha\times x(6,3)+\beta\times x(6,4)+XM_{D+1}, \quad (46)$$

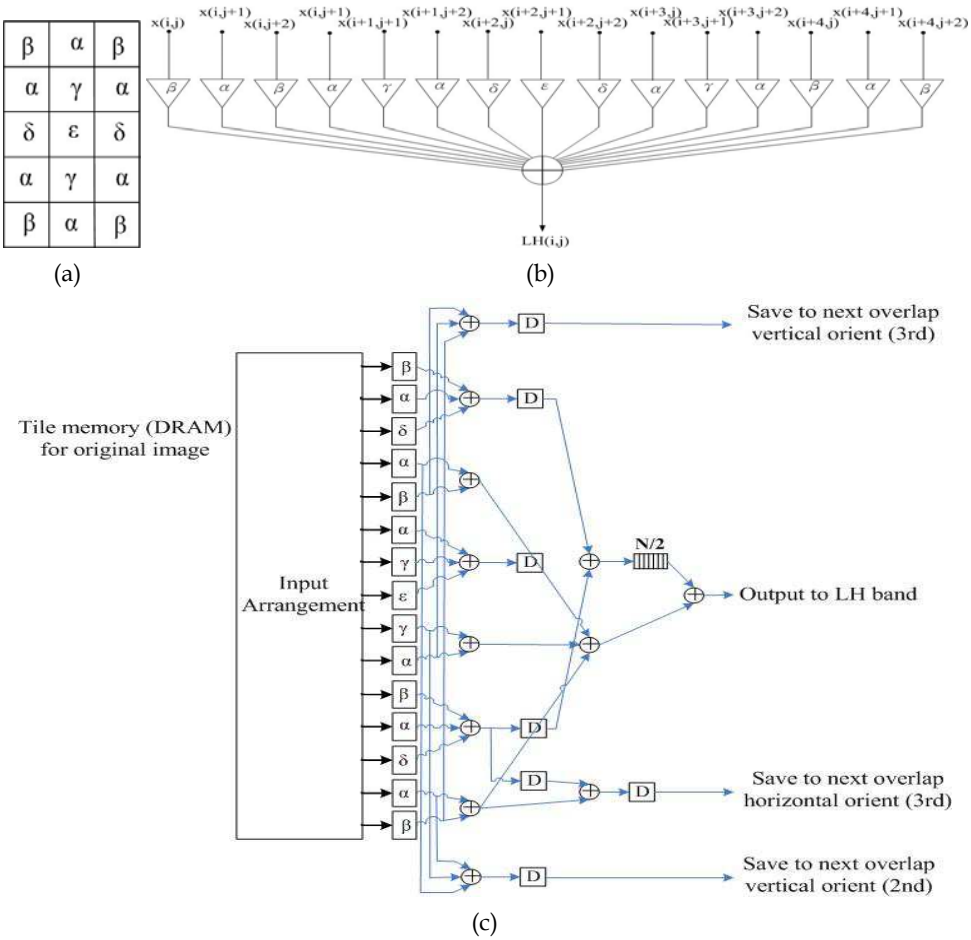


Fig. 11. LH band mask coefficients and the corresponding DSP architecture. (a) Coefficients. (b) DSP architecture. (c) Hardware architecture design.

where the variable XM_{D+1} denotes the repeated part as shown in the gray part of Fig. 12 after the first diagonal scan.
Next the LH(2,2) is calculated as:

$$LH(2,2)=\alpha\times x(5,4)+\varepsilon\times x(6,5)+\delta\times x(6,6)+\gamma\times x(7,5)+\alpha\times x(7,6)+\beta\times x(8,4)+\alpha\times x(8,5)+XM_{D+n}, \quad (47)$$

where the variable XM_{D+n} denotes the repeated part as shown in the gray part of Fig. 13 after the first diagonal scan. The general form of XM_{D+n} can be expressed as:

$$XM_{D+n}=\beta\times x(2i+4,2i+4)+\alpha\times x(2i+4,2i+5)+\beta\times x(2i+4,2i+6)+\alpha\times x(2i+5,2i+4)+\gamma\times x(2i+5,2i+5)+$$
$$+\alpha\times x(2i+5,2i+6)+\delta\times x(2i+6,2i+4)+\alpha\times x(2i+7,2i+4)+\beta\times x(2i+8,2i+4). \quad (48)$$

x(2,2)	x(2,3)	x(2,4)
x(3,2)	x(3,3)	x(3,4)
x(4,2)	x(4,3)	x(4,4)
x(5,2)	x(5,3)	x(5,4)
x(6,2)	x(6,3)	x(6,4)

Fig. 12. Repeat part (in gray) of the diagonal scanned position LH(1,1).

x(4,4)	x(4,5)	x(4,6)
x(5,4)	x(5,5)	x(5,6)
x(6,4)	x(6,5)	x(6,6)
x(7,4)	x(7,5)	x(7,6)
x(8,4)	x(8,5)	x(8,6)

Fig. 13. Repeat part (in gray) of the diagonal scanned position LH(2,2).

The general form of the rest part can be expressed as:

$$LH(i+1,j+1)=\beta\times x(2i+8,2j+6)+\alpha\times(x(2i+7,2j+6)+x(2i+8,2j+5))+\gamma\times x(2i+7,2j+5)+$$
$$+\delta\times x(2i+6,2j+6)+\varepsilon\times x(2i+5,2j+6)+XM_{D+n}. \quad (49)$$

where i=1~N-1, j=1~N-1.

3. Low-Low (LL) band mask coefficients reduction for 2-D SMDWT

According to the 2-D 5/3 LDWT, the LL-band coefficients of the SMDWT can be expressed as follows:

$$LL(i,j)=(9/16)x(2i,2j)+(1/64)\sum^{1}_{u=0}\sum^{1}_{v=0}x(2i\text{-}2+4u,2j\text{-}2+4v)+$$
$$+(1/16)\sum^{1}_{u=0}\sum^{1}_{v=0}x(2i\text{-}1+2u,2j\text{-}1+2v)+(-1/32)\sum^{1}_{u=0}\sum^{1}_{v=0}x(2i\text{-}1+2u,2j\text{-}2+4v)+$$

$$+(-1/32)\sum^{1}_{u=0}\sum^{1}_{v=0}x(2i\text{-}2+4u,2j\text{-}1+2v)+(3/16)\sum^{1}_{u=0}[x(2i\text{-}1+2u,2j)+P(2i,2j\text{-}1+2u)]+$$

$$+(-3/32)\sum^{1}_{u=0}[x(2i\text{-}2+4u,2j)+x(2i,2j\text{-}2+4u)]. \quad (50)$$

The mask as shown in Fig. 14(a) can be obtained via Eq. 50, where α=-1/32, β=1/64, γ=1/16, δ=-3/32, ε=3/16 and ζ=9/16. The DSP and hardware architecture are depicted in Figs. 14(b) and (c). The complexity of the SMDWT is further reduced by employing the symmetric feature of the mask. First, the initial horizontal scan LL(0,0). The next coefficient can be calculated as LL(0,1). where the variable XM_{H+1} denotes the repeated part after the first horizontal coefficient. The general form of the first horizontal step can be expressed as:

$$LL(i,1)=\beta\times x(i,j+2)+\delta\times x(i,j+4)+\alpha\times x(i,j+5)+\beta\times x(i,j+6)+\alpha\times x(i+1,j+2)+\varepsilon\times x(i+1,j+4)+$$

$$+\gamma\times x(i+1,j+5)+\alpha\times x(i+1,j+6)+\delta\times x(i+2,j+2)+\zeta\times x(i+2,j+4)+\varepsilon\times x(i+2,j+5)+\delta\times x(i+2,j+6)+$$

$$+\alpha\times x(i+3,j+2)+\varepsilon\times x(i+3,j+4)+\gamma\times x(i+3,j+5)+\alpha\times x(i+3,j+6)+\beta\times x(i+4,j+2)+$$

$$+\delta\times x(i+4,j+4)+\alpha\times x(i+4,j+5)+\beta\times x(i+4,j+6)+XM_{H+1}, \quad (51)$$

where i=0~N-1, and

$$XM_{H+1}=\alpha\times x(i,3)+\gamma\times x(i+1,3)+\varepsilon\times x(i+2,3)+\gamma\times x(i+3,3)+\alpha\times x(i+4,3). \quad (52)$$

The next coefficient can be calculated as LL(0,2). where the variable XM_{H+n} denotes the repeated part after the second horizontal coefficient. From LL(0,2), the general form can be expressed as:

$$LL(i,j+2)=\delta\times x(i,2j+6)+\alpha\times x(i,2j+7)+\beta\times x(i,2j+8)+\varepsilon\times x(i+1,2j+6)+\gamma\times x(i+1,2j+7)+$$

$$+\alpha\times x(i+1,2j+8)+\zeta\times x(i+2,2j+6)+\varepsilon\times x(i+2,2j+7)+\delta\times x(i+2,2j+8)+$$

$$+\varepsilon\times x(i+3,2j+6)+\gamma\times x(i+3,2j+7)+\alpha\times x(i+3,2j+8)+\delta\times x(i+4,2j+6)+$$

$$+\alpha\times x(i+4,2j+7)+\beta\times x(i+4,2j+8)+XM_{H+n}, \quad (53)$$

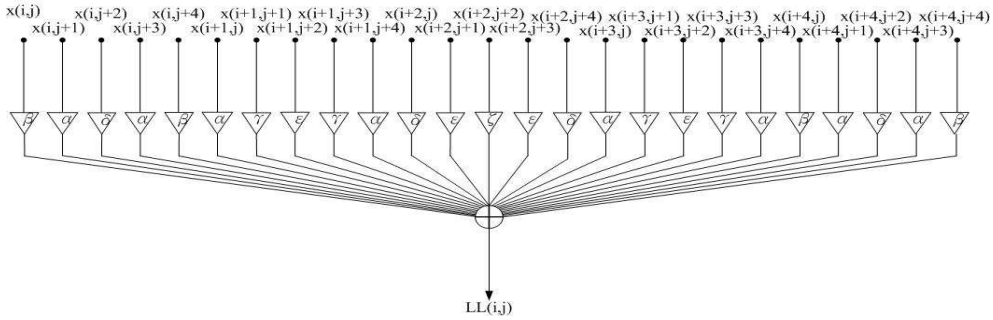
where i=0~N-1, j=0~N-2, and

$$XM_{H+n}=\beta\times x(i,2j+4)+\alpha\times x(i,2j+5)+\alpha\times x(i+1,2j+4)+\gamma\times x(i+1,2j+5)+\delta\times x(i+2,2j+4)+$$

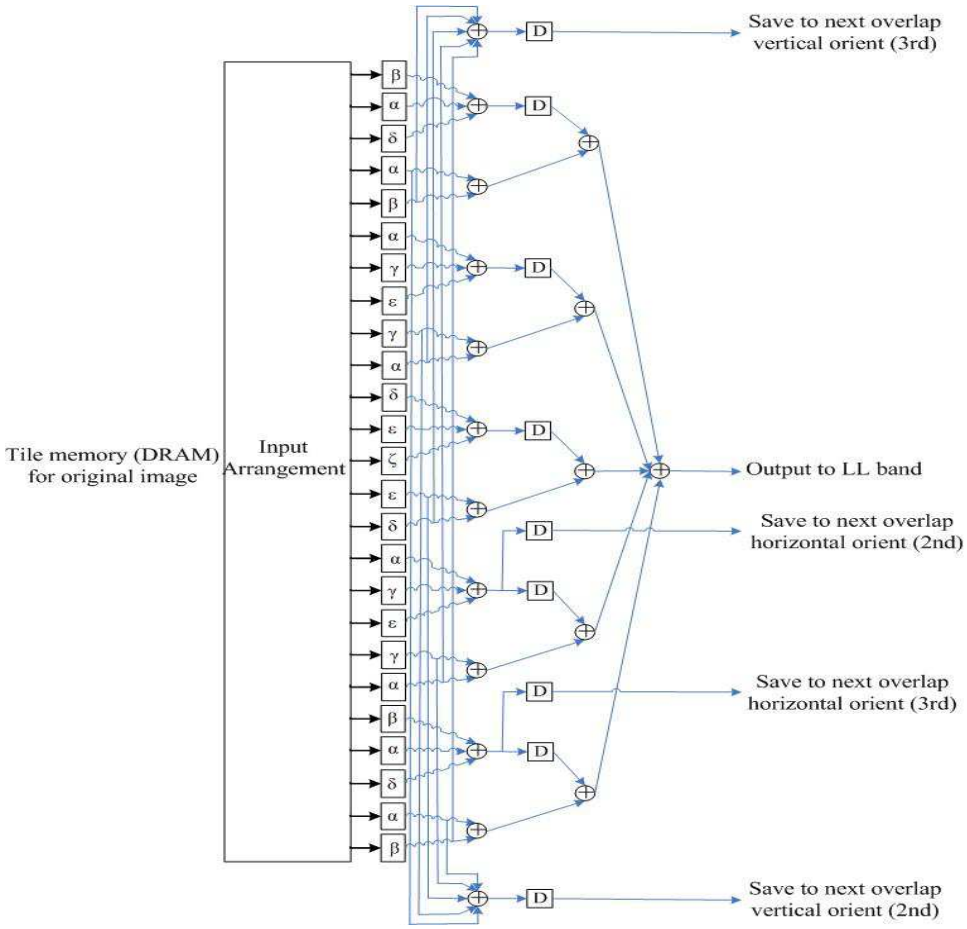
$$+\varepsilon\times x(i+2,2j+5)+\alpha\times x(i+3,2j+4)+\gamma\times x(i+3,2j+5)+\beta\times x(i+4,2j+4)+\alpha\times x(i+4,2j+5). \quad (54)$$

β	α	δ	α	β
α	γ	ε	γ	α
δ	α	ζ	α	δ
α	γ	ε	γ	α
β	α	δ	α	β

(a)



(b)



(c)

Fig. 14. LL band mask coefficients and the corresponding DSP architecture. (a) Coefficients. (b) DSP architecture. (c) Hardware architecture design.

The vertical scan can be done in the same way, where LL(0,0) is the same as that horizontal in LL(0,0). The next coefficient can be calculated as LL(1,0). Next, the initial vertical scan is calculated by the method similar to that of LH SMDWT, where the variable XM_{V+1} denotes the repeated part after the vertical first coefficient. The general form of the first vertical step can be expressed as:

$$\begin{aligned} LL(1,j)=&\beta\times x(2i,j)+\alpha\times x(2i,j+1)+\delta\times x(2i,j+2)+\alpha\times x(2i,j+3)+\beta\times x(2i,j+4)+\delta\times x(2i+4,j)+\\ &+\varepsilon\times x(2i+4,j+1)+\zeta\times x(2i+4,j+2)+\varepsilon\times x(2i+4,j+3)+\delta\times x(2i+4,j+4)+\alpha\times x(2i+5,j)+\\ &+\gamma\times x(2i+5,j+1)+\varepsilon\times x(2i+5,j+2)+\gamma\times x(2i+5,j+3)+\alpha\times x(2i+5,j+4)+\\ &+\beta\times x(2i+6,j)+\alpha\times(2i+6,j+1)+\delta\times x(2i+6,j+2)+\alpha\times x(2i+6,j+3)+\beta\times x(2i+6,j+4)+XM_{V+1}, \end{aligned} \quad (55)$$

where i=0, j=0~N-1, and

$$XM_{V+1}=\alpha\times x(3,j)+\gamma\times x(3,j+1)+\varepsilon\times x(3,j+2)+\gamma\times x(3,j+3)+\alpha\times x(3,j+4). \quad (56)$$

Next, the second vertical scan is calculated by the method similar to that of LH SMDWT.

$$\begin{aligned} LL(i+2,j)=&\delta\times x(2i+6,j)+\varepsilon\times x(2i+6,j+1)+\zeta\times x(2i+6,j+2)+\varepsilon\times x(2i+6,j+3)+\delta\times x(2i+6,j+4)+\\ &+\varepsilon\times x(2i+7,j+2)+\gamma\times x(2i+7,j+1)+\varepsilon\times x(i,2j+7)+\gamma\times x(2i+7,j+3)+\alpha\times x(2i+7,j+4)+\beta\times x(i,2j+8)+\\ &+\alpha\times x(2i+8,j+1)+\delta\times x(2i+8,j+2)+\alpha\times x(2i+8,j+3)+\beta\times x(2i+8,j+4)+XM_{V+n}, \end{aligned} \quad (57)$$

where i=0~N-1, j=0~N-2, and

$$\begin{aligned} XM_{V+n}=&\beta\times x(2i+4,j)+\alpha\times x(2i+4,j+1)+\delta\times x(2i+4,j+2)+\alpha\times x(2i+4,j+3)+\beta\times x(2i+4,j+4)+\\ &+\beta\times x(2i+5,j)+\gamma\times x(2i+5,j+1)+\varepsilon\times x(2i+5,j+2)+\gamma\times x(2i+5,j+3)+\alpha\times x(2i+5,j+4). \end{aligned} \quad (58)$$

Finally, the diagonal oriented scan can be derived as:

$$\begin{aligned} LL(1,1)=&\beta\times x(2,2)+\alpha\times x(2,5)+\beta\times x(2,6)+\zeta\times x(4,4)+\varepsilon\times x(4,5)+\alpha\times x(5,2)+\varepsilon\times x(5,4)+\\ &+\gamma\times x(5,5)+\alpha\times x(5,6)+\beta\times x(6,2)+\delta\times x(6,4)+\alpha\times x(6,5)+\beta\times x(6,6)+XM_{D+1}, \end{aligned} \quad (59)$$

where the variable XM_{D+1} denotes the repeated part as shown in the gray part of Fig. 15 after the first diagonal scan.
Next the HL(2,2) is calculated as:

$$\begin{aligned} LL(2,2)=&\varepsilon\times x(6,5)+\zeta\times x(6,6)+\varepsilon\times x(6,7)+\gamma\times x(7,5)+\varepsilon\times x(7,6)+\gamma\times x(7,7)+\alpha\times x(7,8)+\\ &+\alpha\times x(8,5)+\delta\times x(8,6)+\alpha\times x(8,7)+\beta\times x(8,8)+XM_{D+n}, \end{aligned} \quad (60)$$

where the variable XM_{D+2} denotes the repeated part as shown in the gray part of Fig. 16 after the first diagonal scan. The variable XM_{D+1} denotes the repeated part as shown in the gray part of Fig. 17 after the first diagonal scan. The general form of XM_{D+n} can be expressed as:

x(2,2)	x(2,3)	x(2,4)	x(2,5)	x(2,6)
x(3,2)	x(3,3)	x(3,4)	x(3,5)	x(3,6)
x(4,2)	x(4,3)	x(4,4)	x(4,5)	x(4,6)
x(5,2)	x(5,3)	x(5,4)	x(5,5)	x(5,6)
x(6,2)	x(6,3)	x(6,4)	x(6,5)	x(6,6)

Fig. 15. Repeat part (in gray) of the diagonal scanned position LL(1,1).

x(4,4)	x(4,5)	x(4,6)	x(4,7)	x(4,8)
x(5,4)	x(5,5)	x(5,6)	x(5,7)	x(5,8)
x(6,4)	x(6,5)	x(6,6)	x(6,7)	x(6,8)
x(7,4)	x(7,5)	x(7,6)	x(7,7)	x(7,8)
x(8,4)	x(8,5)	x(8,6)	x(8,7)	x(8,8)

Fig. 16. Repeat part (in gray) of the diagonal scanned position LL(2,2).

x(6,6)	x(6,7)	x(6,8)	x(6,9)	x(6,10)
x(7,6)	x(7,7)	x(7,8)	x(7,9)	x(7,10)
x(8,6)	x(8,7)	x(8,8)	x(8,9)	x(8,10)
x(9,6)	x(9,7)	x(9,8)	x(9,9)	x(9,10)
x(10,6)	x(10,7)	x(10,8)	x(10,9)	x(10,10)

Fig. 17. Repeat part (in gray) of the diagonal scanned position LL(3,3).

$$XM_{D+n}=\beta\times x(2i+6,2i+6)+\alpha\times x(2i+6,2i+7)+\delta\times x(2i+6,2i+8)+\alpha\times x(2i+6,2i+9)+$$
$$+\beta\times x(2i+6,2i+10)+\alpha\times x(2i+7,2i+6)+\gamma\times x(2i+7,2i+7)+\varepsilon\times x(2i+7,2i+8)+$$
$$+\gamma\times x(2i+7,2i+9)+\alpha\times x(2i+7,2i+10)+\delta\times x(2i+8,2i+6)+\varepsilon\times x(2i+8,2i+7)+$$
$$+\delta\times x(2i+8,2i+10)+\alpha\times x(2i+9,2i+6)+\gamma\times x(2i+9,2i+7)+\beta\times x(2i+10,2i+6)+\alpha\times x(2i+10,2i+7). \quad (66)$$

The general form of the rest part can be expressed as:

$$LL(i+1,j+1)=\zeta\times x(2i+8,2i+8)+\varepsilon\times x(2i+8,2i+9)+\varepsilon\times x(2i+9,2i+8)+\gamma\times x(2i+9,2i+9)+$$
$$+\alpha\times x(2i+9,2i+10)+\delta\times x(2i+10,2i+8)+\alpha\times x(2i+10,2i+9)+\beta\times x(2i+10,2i+10)+XM_{D+n}, \quad (67)$$

where i=1~N-1, j=1~N-1.

3.3 Summary of the complexity reduction

The four-matrix frameworks, HH, HL, LH, and LL lead to four different architectures. Each of these is described by the structural behavior of different components that makes up the digital signal processing (DSP) as shown in Table 1. The discussion above shows that the

complexity of the proposed SMDWT can be significantly reduced by exploiting the symmetric feature of the masks. Tables 2-5 show the overall complexity reductions from the original SMDWT to the simplified SMDWT.

HH	9	2
HL	15	2
LH	15	2
LL	25	2

Table 1. The Subband Mask for DSP.

XM_H of HH(i,j+1)	β×x(i,2j+2)+α×x(i+1,2j+2)+β×x(i+2,2j+2).
Complexity reduction	Original SMDWT: adder is 8, and multiplier is 9. (number of operations) Simplified SMDWT: adder is 5, and multiplier is 0. (The shifter is used to replace multiplier)
XM_V of HH(i+1,j)	β×x(2i+2,j)+α×x(2i+2,j+1)+β×x(2i+2,j+2).
Complexity reduction	Original SMDWT: adder is 8, and multiplier is 9. Simplified SMDWT: adder is 6, and multiplier is 0.

Table 2. HH-Band Wavelet Coefficient (Mask of Size 3×3).

XM_{H+1} of HL(i,1)	α×x(i,3)+γ×x(i+1,3)+α×x(i+2,3).
Complexity reduction	Original SMDWT: adder is 14, and multiplier is 15. Simplified SMDWT: adder is 12, and multiplier is 0.
XM_{H+n} of HL(i,j+2)	β×x(i,2j+4)+α×x(i,2j+5)+α×x(i+1,2j+4)+γ×x(i+1,2j+5)+β×x(i+2,2j+4)+α×x(i+2,2j+5).
Complexity reduction	Original SMDWT: adder is 14, and multiplier is 15. Simplified SMDWT: adder is 9, and multiplier is 0.
XM_V of HL(i+1,j)	β×x(2i+2,j)+α×x(2i+2,j+1)+δ×x(2i+2,j+2)+α×x(2i+2,j+3)+β×x(2i+2,j+4).
Complexity reduction	Original SMDWT: adder is 14, and multiplier is 15. Simplified SMDWT: adder is 10, and multiplier is 0.

Table 3. HL-Band Wavelet Coefficient (Mask of Size 5×3).

XM_H of LH(i,j+1)	β×x(i,2j+2)+α×x(i+1,2j+2)+δ×x(2i+2,j+2)+α×x(i+3,2j+2)+β×x(i+4,2j+2).
Complexity reduction	Original SMDWT: adder is 14, and multiplier is 15. Simplified SMDWT: adder is 10, and multiplier is 0.
XM_{V+1} of LH(1,j)	α×x(2i+3,0)+γ×x(2i+3,1)+α×x(2i+3,2).
Complexity reduction	Original SMDWT: adder is 14, and multiplier is 15. Simplified SMDWT: adder is 12, and multiplier is 0.
XM_{V+n} of LH(i+2,j)	β×x(2i+4,j)+α×x(2i+4,j+1)+β×x(2i+4,j+2)+α×x(2i+5,j)+γ×x(2i+5,j+1)+α×x(2i+5,j+2).
Complexity reduction	Original SMDWT: adder is 14, and multiplier is 15. Simplified SMDWT: adder is 9, and multiplier is 0.

Table 4. LH-Band Wavelet Coefficient (Mask of Size 3×5).

XMH+1 of LL(i,1)	α×x(i,3)+γ×x(i+1,3)+ε×x(i+2,3)+γ×x(i+3,3)+α×x(i+4,3).
Complexity reduction	Original SMDWT: adder is 24, and multiplier is 25. Simplified SMDWT: adder is 20, and multiplier is 0.
XMH+n of LL(i,j+2)	β×x(i,2j+4)+α×x(i,2j+5)+α×x(i+1,2j+4)+γ×x(i+1,2j+5)+δ×x(i+2,2j+4)+ε×x(i+2,2j+5)+α×x(i+3,2j+4)+γ×x(i+3,2j+5)+β×x(i+4,2j+4)+α×x(i+4,2j+5).
Complexity reduction	Original SMDWT: adder is 24, and multiplier is 25. Simplified SMDWT: adder is 15, and multiplier is 0.
XMV+1 of LL(1,j)	α×x(3,j)+γ×x(3,j+1)+ε×x(3,j+2)+γ×x(3,j+3)+α×x(3,j+4).
Complexity reduction	Original SMDWT: adder is 24, and multiplier is 25. Simplified SMDWT: adder is 20, and multiplier is 0.
XMV+n of LL(i+2,j)	β×x(2i+4,j)+α×x(2i+4,j+1)+δ×x(2i+4,j+2)+α×x(2i+4,j+3)+β×x(2i+4,j+4)+β×x(2i+5,j)+γ×x(2i+5,j+1)+ε×x(2i+5,j+2)+γ×x(2i+5,j+3)+α×x(2i+5,j+4).
Complexity reduction	Original SMDWT: adder is 24, and multiplier is 25. Simplified SMDWT: adder is 15, and multiplier is 0.

Table 5. LL-Band Wavelet Coefficient (Mask of Size 5×5).

4. Experimental results and performance comparisons

The proposed 2-D SMDWT algorithm is generally used to performing the 2-D DWT for still images. Figure 18 shows the schematic diagram of the 2-D SMDWT. The wavelet transform provides a multi-scale representation of image/video in the spatial-frequency domain. Besides the energy compaction and decorrelation properties that facilitate compression, a major advantage of the DWT is its scalability. The proposed algorithm is based on the four-subband matrices (HH, HL, LH, and LL) which are processed to achieve the same performance as the 5/3 LDWT algorithm. The SMDWT is implemented in the JPEG2000 reference software VM 9.0 and is compared with the original JPEG2000. The test image the used in this experiment was Lena of size 512×512. Experimental results show that the proposed algorithm not only significantly improves lifting-based latency, but also has the same visual quality as the normal 2-D 5/3 LDWT as shown in Fig. 19.

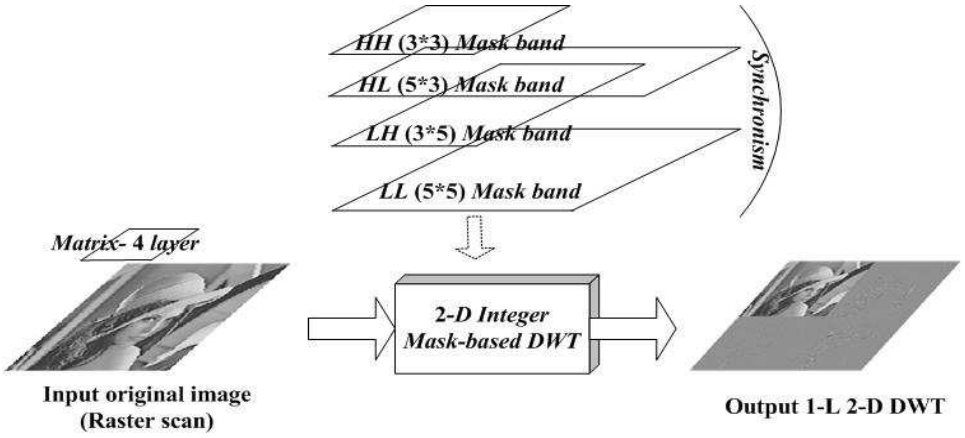


Fig. 18. Schematic diagram of the 2-D SMDWT.

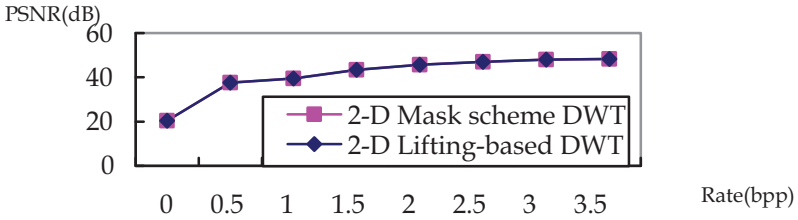
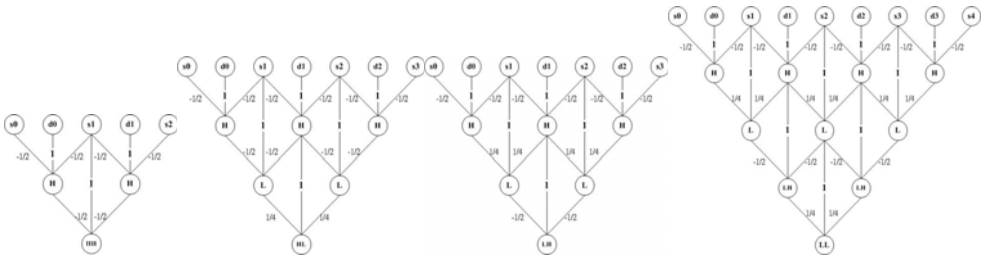


Fig. 19. PSNR (dB) versus Rate (bpp) comparison between 2-D LDWT and the proposed 2-D SMDWT.

The architecture of the 2-D SMDWT has many advantages compared to the 2-D LDWT. For example, the critical path of the 2-D LDWT is potentially longer than that of SMDWT. Moreover, the 2-D LDWT is frame-based with the implementation bottleneck being the huge amount of the transpose memory size. This work uses the symmetric feature of the masks in SMDWT to improve the design. Experimental results, as shown in Table 7 show that the proposed algorithm is superior to most of the previous works. The proposed algorithm has efficient solutions for reducing the critical path (which is defined as the longest, time-weighted sequence of events from the start of the program to its termination with examples shown in Figs. 7(c), 8(c), 11(c), 14(c)), latency (the time between the arrival of a new signal and its first signal output becoming available in the system), and hardware cost, as shown in Figs. 7, 8, 11, 14, and 20, and Table 6. The SMDWT approach requires a transpose memory of size (N/2)+26 ((N/2) is on-chip memory of size and 26 is number of register). The proposed 2-D DWT adopts parallel and pipeline schemes are employed to reduce the transpose memory and increase the operating speed. The shifters and adders replace multipliers in the computation to increase the hardware utilization and reduce the hardware cost. A $N \times N$ 2-D lifting-based DWT is RTL (Register Transistor Level) designed and simulated with VerilogHDL in this paper.



(a) (b) (c) (d)

Fig. 20. 2-D LDWT critical path. (a) HH band. (b) HL band. (c) LH band. (d) LL band.

Subbands	LDWT critical path	SMDWT critical path
HH	$2T_M+2T_A$	$1T_M+2T_A$ Fig.7(c)
HL	$3T_M+3T_A$	$1T_M+2T_A$ Fig.8(c)
LH	$3T_M+3T_A$	$1T_M+3T_A$ Fig.11(c)
LL	$4T_M+4T_A$	$1T_M+3T_A$ Fig.14(c)

*T_M: Multiplier operation time; T_A: Adder operation time

Table 6. Subband Lifting-Based V.S. Mask-Based for Integer 2-D DWT.

Methods	2-D DWT	Wave stage	[1]Transpose memory	[2]Latency	[3]Computing time	Complexity
Chiang et al., 2005	LDWT	Integer	[4]N	7	$(3/4)N^2+7$	Simple
Chiang & Hsia, 2005	LDWT	Integer	$N^2/4+5N$	3	N^2	Medium
Diou et al., 2001	LDWT	Integer	3.5N	N/A	N/A	Simple
Chen & Wu, 2002	LDWT	Integer	2.5N	N/A	N^2	Complexity
Andra et al., 2002	LDWT	Integer	3.5N	2N+5	$(N^2/2)+N+5$	Simple
Tan & Arslan, 2003	LDWT	Integer	3N	N/A	$(N^2/2)+N+5$	Medium
Lee et al., 2003	LDWT	Integer	N	5	$(N^2/2)+5$	Medium
ISO/IEC, 2000	LDWT	Integer	N^2	N/A	N/A	Simple
Varshney et al., 2007	LDWT	Integer	3N	13	N/A	Medium
Chen, 2002	LDWT	Integer	3N	N/A	$(N^2/2)+N+5$	Medium
Proposed	*SMDWT*	Integer	*(N/2)+26*	2	*$N^2/4+3$*	*Simple*

[1] Transpose memory is used to store frequency coefficients in the 1-L 2-D DWT.
[2] In a system, latency is often used to mean any delay or waiting time that increases real or perceived response time beyond the response time desired. For example, specific contributors to 2-D DWT latency include from original image input to first subband output in signal.
[3] In a system, computing time represents the time used to compute an image of size $N \times N$.
[4] Suppose image is of size $N \times N$.

Table 7. Performance Comparisons.

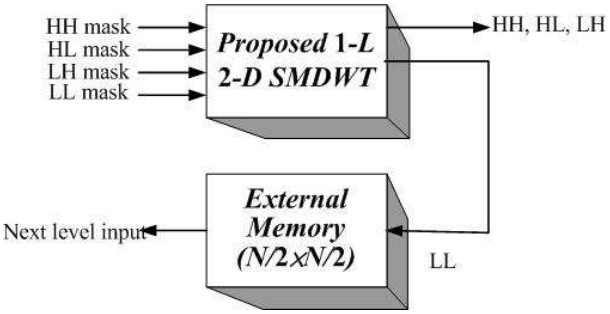


Fig. 21. The multilevel 2-D DWT architecture.

The multi-level DWT computation can be implemented similarly by the proposed 2-D SMDWT. For the multi-level computation, this architecture needs $(N^2/4)$ off-chip memory. As illustrated in Fig. 21, the off-chip memory temporarily stores the LL subband coefficients for the next iteration computations. The second level computation requires $N/2$ counters and $N/2$ FIFOs for the control unit. The third level computation requires $N/4$ counters and $N/4$ FIFOs for the control unit. Generally, the jth level computation needs $N/2^{j-1}$ counters and $N/2^{j-1}$ FIFOs. Therefore, the proposed architecture is suitable for multilevel DWT

computations. The SMDWT also has the advantages of regular signal coding, short critical path, reduced latency time, and independent subband coding processing. Moreover, SMDWT can easily reduce the transpose memory access time and overlap original signal access so that power consumption of 2-D LDWT can also be easily improved by SMDWT.

5. Conclusions

This work proposes a novel 2-D SMDWT fast algorithm, which is superior to the 5/3 LDWT. The algorithm solves the latency problem in the previous schemes caused by multiple-layer transpose decomposition operation. Moreover, it provides real-time requirement and can be further applied to the 3-D wavelet video coding [30].

The proposed 2-D SMDWT algorithm has the advantages of a fast computational speed, less complexity, reduced latency. Low-transpose memory and regular data flow, and is suitable for VLSI implementation. Possible future works are described below:

1. The Dual-Mode 2-D SMDWT on JPEG2000: The dual-mode 2-D SMDWT can be developed to support 5/3 (lossless) lifting or 9/7 (lossy) lifting using similar hardware architecture, since the 5/3 and 9/7 are very similar and both have less complexity.
2. High Performance JPEG2000 Codec: Since part of the JPEG2000 encoder is symmetric to the decoder the complexity of both the encoder and the decoder can be reduced.
3. An independent four-subband mask can be used in other visual coding fields (eg. visual processing, visual compression and visual recognition).

6. References

Andra, K.; Chakrabarti, C. & Acharya, T. (2000), A VLSI architecture for lifting-based wavelet transform, *IEEE Workshop on Signal Processing Systems*, (October 2000), pp. 70-79.

Andra, K.; Chakrabarti, C. & Acharya, T. (2002), A VLSI architecture for lifting-based forward and inverse wavelet transform, *IEEE Transactions on Signal Processing*, vol. 50, no.4, (April 2002), pp. 966-977.

Chen, S.-C. & Wu, C.-C. (2002). An architecture of 2-D 3-level lifting-based discrete wavelet transform, *Proceeding of the VLSI Design/ CAD Symposium*, (August 2002), pp. 351-354.

Chen, P.-Y. (2002). VLSI implementation of discrete wavelet transform using the 5/3 filter, *IEICE Transactions on Information and Systems*, vol. E85-D, no.12, (December 2002), pp. 1893-1897.

Chen, P. & J. W. Woods. (2004). Bidirectional MC-EZBC with lifting implementation, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 10, (October 2004), pp. 1183-1194.

Chiang, J.-S.; Hsia, C.-H. & Chen, H.-J. (2005). 2-D discrete wavelet transform with efficient parallel scheme, *International Conference on Imaging Science, Systems, and Technology: Computer Graphics*, (June 2005), pp. 193-197.

Chiang, J.-S.; Hsia, C.-H.; Chen, H.-J. & Lo, T.-J. (2005). VLSI architecture of low memory and high speed 2-D lifting-based discrete wavelet transform for JPEG2000 applications, *IEEE International Symposium on Circuits and Systems*, (May 2005), pp. 4554-4557.

Chiang, J.-S. & Hsia, C.-H. (2005). An efficient VLSI architecture for 2-D DWT using lifting scheme, *IEEE International Conference on Systems and Signals*, (April 2005), pp. 528-531.

Daubechies, I. & Sweldens, W. (1998). Factoring wavelet transforms into lifting steps, *The Journal of Fourier Analysis and Applications*, vol. 4, no.3, (1998), pp. 247-269.

Diou, C.; Torres, L. & Robert, M. (2001). An embedded core for the 2-D wavelet transform, *IEEE on Emerging Technologies and Factory Automation Proceedings*, vol. 2, (2001), pp. 179-186.

Feig, E.; Peterson, H. & Ratnakar, V. (1995). Image compression using spatial prediction, *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, (May 1995), pp. 2339-2342.

Habibi, A. & Hershel, R. S. (1974). A unified representation of differential pulse code modulation (DPCM) and transform coding systems, *IEEE Transaction on Communications*, vol. 22, no. 5, (May 1974), pp. 692-696.

Huang, C.-T.; Tseng, P.-C. & Chen, L.-G. (2005). Analysis and VLSI architecture for 1-D and 2-D discrete wavelet transform, *IEEE Transactions on Signal Processing,* vol. 53, no. 4., (April 2005), pp. 1575-1586.

ISO/IEC JTC1/SC29 WG1. (2000). *JPEG 2000 Part 1 Final Committee Draft Version 1.0*, Information Technology.

ISO/IEC JTC1/SC29/WG1 Wgln 1684. (2000). *JPEG 2000 Verification Model 9.0*.

ISO/IEC JTC1/SC29 WG11. (2001). *Coding of Moving Pictures and Audio,* Information Technology.

ISO/IEC ISO/IEC 15444-3. (2002). *Motion JPEG2000,* Information Technology.

Kondo, H. & Oishi, Y. (2000). Digital image compression using directional sub-block DCT, *International Conference on Communications Technology*, vol. 1, (August 2000), pp.21-25.

Lan, X.; Zheng, N. & Liu, Y. (2005). Low-power and high-speed VLSI architecture for lifting-based forward and inverse wavelet transform, *IEEE Transactions on Consumer Electronics*, vol. 51, no.2, (May 2005), pp. 379-385.

Lee, W.-T.; Chen, H.-Y.; Hsiao, P.-Y. & Tsai, C.-C. (2003). An efficient lifting based architecture for 2-D DWT used in JPEG2000, *Proceeding of the VLSI Design/ CAD Symposium*, (August 2003), pp. 577-580.

Lian, C.-J.; Chen, K.-F.; Chen, H.-H. & Chen, L.-G. (2001). Lifting based discrete wavelet transform architecture for JPEG2000, *IEEE International Symposium on Circuits and Systems*, vol. 2, (May 2001), pp. 445-448.

Liao, H.; Mandal, M. Kr. & Cockburn, B. F. (2004). Efficient architecture for 1-D and 2-D lifting-based wavelet transforms, *IEEE Transactions on Signal Processing*, vol. 52, no. 5, (May 2004), pp. 1315-1326.

Mallat, S. G. (1989). A theory for multi-resolution signal decomposition: The wavelet representation, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, (July 1989), pp. 674-693.

Ohm, J.-R. (2005). Advances in scalable video coding, *Proceedings of the IEEE,* Invited Paper, vol. 93, no.1, (January 2005), pp. 42-56.

Srinivasan, K. S. S. (2002). VLSI implementation of 2-D DWT/IDWT cores using 9/7-tap filter banks based on the non-expansive symmetric extension scheme, *IEEE International Conference on VLSI Design*, (January 2002), pp. 435-440.

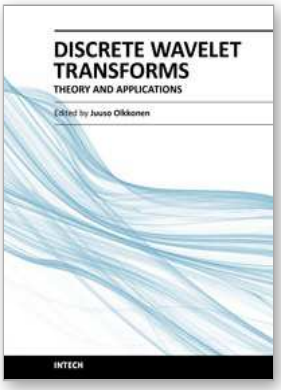
Sweldens, W. (1996). The lifting scheme: A custom-design construction of biorthogonal wavelets, *Applied and Computation Harmonic Analysis*, vol. 3, no. 0015, (1996), pp.186-200.

Tan, K.C.B. & Arslan, T. (2003). Shift-accumulator ALU centric JPEG 2000 5/3 lifting based discrete wavelet transform architecture, *IEEE International Symposium on Circuits and Systems,* vol. 5, (May 2003), pp. V161-V164.

Varshney, H.; Hasan, M. & Jain, S. (2007). Energy efficient novel architecture for the lifting-based discrete wavelet transform, *IET Image Process*, vol. 1, no. 3, (September 2007), pp.305-310.

Weeks, M. & Bayoumi, M. A. (2002). Three-dimensional discrete wavelet transform architectures, *IEEE Transactions on Signal Processing*, vol. 50, no.8, (August 2002), pp. 2050-2063.

Wu, B.-F. & Lin, C.-F. (2005). A high-performance and memory-efficient pipeline architecture for the 5/3 and 9/7 discrete wavelet transform of JPEG2000 codec, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no.12, (December 2005), pp. 1615-1628.



Discrete Wavelet Transforms - Theory and Applications

Edited by Dr. Juuso T. Olkkonen

ISBN 978-953-307-185-5
Hard cover, 256 pages
Publisher InTech
Published online 04, April, 2011
Published in print edition April, 2011

Discrete wavelet transform (DWT) algorithms have become standard tools for discrete-time signal and image processing in several areas in research and industry. As DWT provides both frequency and location information of the analyzed signal, it is constantly used to solve and treat more and more advanced problems. The present book: Discrete Wavelet Transforms: Theory and Applications describes the latest progress in DWT analysis in non-stationary signal processing, multi-scale image enhancement as well as in biomedical and industrial applications. Each book chapter is a separate entity providing examples both the theory and applications. The book comprises of tutorial and advanced material. It is intended to be a reference text for graduate students and researchers to obtain in-depth knowledge in specific applications.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821