International Journal of Electronics and Communication Engineering (IJECE)
ISSN 2278-9901
Vol. 2, Issue 1, Feb 2013, 87-96

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

HIGH THROUGHPUT AND EFFICIENT MEMORY VLSI STRUCTURE USING 2-D DWT IN MULTILEVEL LIFTING TECHNIQUE

R. P. MEENAAKSHI SUNDARI[1], R. ANITA[2] & S. GANESH BABU[3]

[1]Professor/ECE, Sasurie College of Engineering, Vijayamangalam, Tamil Nadu, India

[2]Professor and Head/EEE, Institute of Road and Transport Technology, Erode, Tamil Nadu, India

[3]II M.E – Applied Electronics, Sasurie College of Engineering, Vijayamangalam, Tamil Nadu, India

ABSTRACT

This paper presents the VLSI structure using 2-D DWT in multilevel lifting technique. The discrete wavelet transform (DWT) has the applications in image and video compression. It has advantage over the discrete cosine transform (DCT) in image compression, so the 2-D DWT is proposed for JPEG applications. The high complexity frame and link buffer are removed in the multilevel lifting. The local registers and RAM can be used instead of link and frame buffer. Pyramid algorithm and recursive pyramid algorithm are involved in the design. The design reduces the overall area-delay product and also has high throughput. The output latency is very low compared to latency of existing structures. The design has less slice delay product (SDP) and it has low value of power per output (PPO) compared to the existing structures.

KEYWORDS: Convolution Method, Discrete Wavelet Transform (DWT), Frame and Link Buffer, Multilevel Lifting Technique

INTRODUCTION

Discrete Wavelet Transform (DWT) has applications in image compression over the discrete cosine transform (DCT), 2-D DWT has been used for fractal analysis, signal analysis, computer vision, object recognition, computer graphics, portable handheld wireless devices and bioinformatics, etc. 2-D DWT is core block for the JPEG-2000 compression standard. The high speed implementation can be easily achieved by 2-D DWT, so it can be mainly used for applications in demand real-time processing.

High-speed computation can be easily achieved in modern VLSI systems. Many of the existing DWT architectures are based on linear convolution property of the wavelet filters. The lifting scheme of computation of DWT is more popular over the convolution-based scheme for its lower computational complexity. Moreover, the lifting-based wavelet decomposition has many useful properties like symmetric forward and inverse transform, in-place computation and integer-to-integer wavelet transform.

Compared to the convolution-based algorithm, lifting technique based algorithm involves much less number of multipliers, adders and storage elements in same architecture. So the lifting easily achieves the low-complexity hardware implementation in 2-D DWT scheme. In 2-D DWT structures have the overall hardware complexity in two components, the first one is arithmetic component and the second one is memory component.

The arithmetic component involves in multipliers, adders and memory elements. It mainly comprises the transposition buffer, temporal and frame-buffer. Transposition and temporal buffer are usually located on-chip memory, while the frame buffer is located off-chip memory because it is very large. The 2-D DWT structure speed and power performance can be highly affected by the size of the on-chip and off-chip memory.

The 2-D DWT has the several lifting scheme based architectures than the fast convolution scheme for efficient implementation. The proposed method has to meet the requirement of high latency, so it requires large internal buffer in a four-processor block-based architecture. In the line-based scanning scheme of 2-D DWT folded architecture can be computed as level-by-level. The column-wise processing is performed for buffering of intermediate output matrix in a few rows by line-based scanning.

The size of the frame-buffer and line-buffer can be reduced by various architectural designs and data access schemes of 2-D DWT for efficient implementation of VLSI system. The transposition buffer size can be used for separable 2-D DWT scheme. Multilevel algorithm has pipeline implementation for eliminate the frame-buffer efficiently. The Hardware Utilization Efficiency (HUE) is poor in the pipeline structure of multilevel DWT in direct mapping. Simple control circuit is given by the pipeline structure.

The line-buffer and frame-buffers are avoided in the proposed method. It provides small on-chip storage for efficient real time application with simple control circuit for achieving the 100% HUE. It is the merits of proposed architecture. In the proposed design overall latency of computation can be reduced and also provides high throughput rate for given VLSI system. The portable and wireless hand-held devices are used for multimedia applications.

The maximum throughput is optimized for scalable design by the usage of on-chip storage. The 2-D DWT is designed by daubenchies (9, 7) filter which can be used in JPEG-2000 compression standard. The proposed structure is modified easily. The fast multiplication algorithm like Modified Booth multiplier can be implemented instead of array multiplier and ripple carry adder can be replaced by fast adder like Carry Save Adder (CSA) in the VLSI design. So the 2-D DWT could be realized through finite lifting steps.

DISCRETE WAVELET TRANSFORM

The Wavelet Series is just a sampled version of CWT and its computation may consume significant amount of time and resources, depending on the resolution required. The Discrete Wavelet Transform (DWT), which is based on sub-band coding is found to yield a fast computation of Wavelet Transform. It is easy to implement and reduces the computation time and resources required.

In CWT, the signals are analyzed using a set of basic functions which relate to each other by simple scaling and translation. In the case of DWT, a time-scale representation of the digital signal is obtained using digital filtering techniques. The signal to be analyzed is passed through filters with different cutoff frequencies at different scales. The one-dimensional, discrete, dyadic, decimating (non-redundant) wavelet transformation (DWT) of a signal is a linear operation that maps the discrete input signal of length k onto the set of k wavelet coefficients.

This decomposition can be repeated for the low pass filtered approximation coefficients until the maximum decomposition level lmax at log2 k (assumed k is a power of two) is reached. For perfect reconstruction of the signal, the dual filters g and h are applied to the coefficients at decomposition level l after up sampling. The two resulting parts are summed up leading to the approximation coefficients at level l. When dealing with images, a two-dimensional wavelet transformation is required. The one-dimensional transformation can be applied to the rows and columns in succession, which is referred to as separable transformation.

2D-DWT

Over the past several years, the wavelet transform has gained widespread acceptance in signal processing in general and in image compression research in particular. In applications such as still image compression, Discrete Wavelet

Transform (DWT) based schemes have outperformed other coding schemes like the ones based on Discrete Cosine Transform (DCT). The DWT has been introduced as a highly efficient and flexible method for sub band decomposition of signals..

The high algorithmic performance of the 2D DWT in image compression justifies its use as the kernel of both the JPEG2000 still image compression standard and the MPEG-4 texture coding standard. It is widely recognized that the LeGall (5, 3) and the Daubechies (9, 7) filters are among the best filters for DWT-based image compression... In fact, the JPEG2000 image coding standard employs the (5, 3) and the (9, 7) filters as the default wavelet filters for respectively loss and lossy compression.

The JPEG2000 can compress images 100 times smaller than the original image. With this compression ratio, the reconstructed image of the JPEG2000 still provides good visual quality. The coding efficiency of the JPEG2000 comes with the cost. The 2D-DWT is one of the main resources intensive components of JPEG2000. It demands massive computations and represents one of the critical parts in the design and implementation of the JPEG2000 standard.

Daubechies (9, 7) Filter

Daubechies wavelets use overlapping windows, so the high frequency coefficient spectrum reflects all high frequency changes. Therefore Daubechies wavelets are useful in compression and noise removal of audio signal processing. Haar wavelet transform is not useful in compression and noise removal of audio signal processing.

IMPLEMENTATION

Usually the 2-D DWT can be implemented by following two techniques:

Convolution Based DWT

Convolution method is a pair of Finite Impulse Response filters (FIR) and it is used to derive the low pass and high pass filter coefficients in parallel structure. Convolution based DWT implementation can be realized by the transform using the FIR filters.

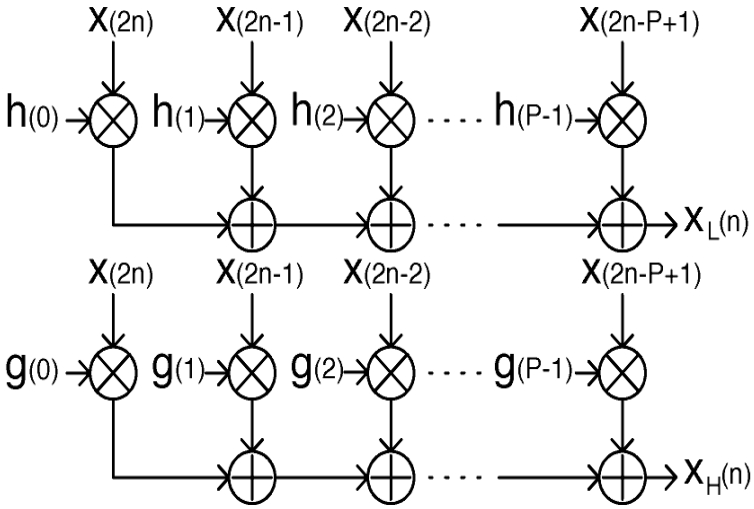

Figure 1: General Convolution Based Architecture

At each transform level the input discrete signal X(n) can be filtered by a low-pass filter (h) and a high-pass filter (g). The two output streams are then sub-sampled and to produce the low frequency and high frequency component of each pixel of an image.

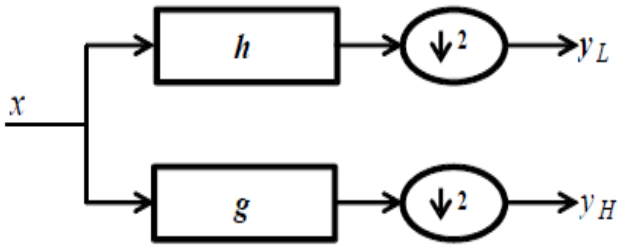

Figure 2: 1-D DWT Decomposition

Lifting Based DWT

The lifting-based technique has the features of providing lower and upper triangular matrices from low pass and high pass wavelet filters breaking operation. The redundancy can be removed by the use of data correlation scheme. The lifting algorithm has three main phases as follows:

Split Phase

The split phase, the input pixel value can be split into even and odd samples.

$$Xe=X(2n),\ Xo=X(2n+1) \tag{1}$$

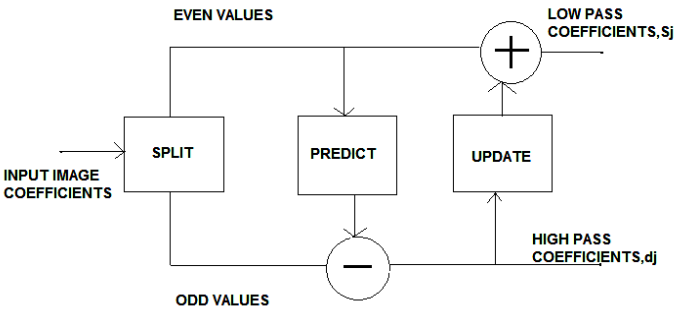

Figure 3: Forward Lifting Scheme

Prediction Phase

The prediction stage is used to eliminate redundancy component in each pixel of an image.

$$Y(2n+1) = Xo(2n+1) - P(Xe) \tag{2}$$

Update Phase

In the update phase neighboring wavelet coefficients are combined to form image.

$$Y(2n) = Y(2n+1) + U(Xe) \tag{3}$$

Where, U is the new update operator.

THE PROPOSED ARCHITECTURE

This proposed DWT structures is designed for image application like compressing and more, here also use MATLAB for frame work of the input. An input image is received from the MATLAB and the pixel values are converted into binary form and that is read from the coding and also that file can be added as package.

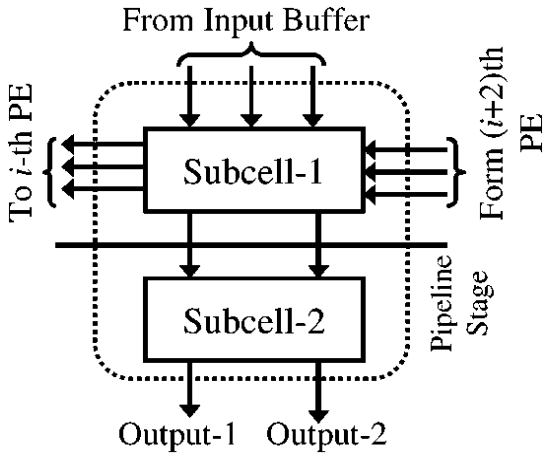

Figure 4: Processing Element

The image pixel value in the binary form will be used as the input for the total system and the output from the total system also will be processed pixel value in binary format. The output pixel value will be stored in a file for image reconstruction in MATLAB.

The DWT is computed using addition and multipliers where the proposed method is designed using the lifting scheme. Lifting method is more advantage when compared with convolution since reduced computation Complexity. In this multilevel lifting for 2D DWT is proposed and that will be computed both row and column of the image will be processed simultaneously.

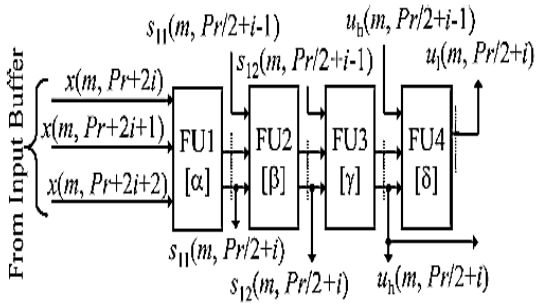

Figure 5: Internal Structure of Subcell-1

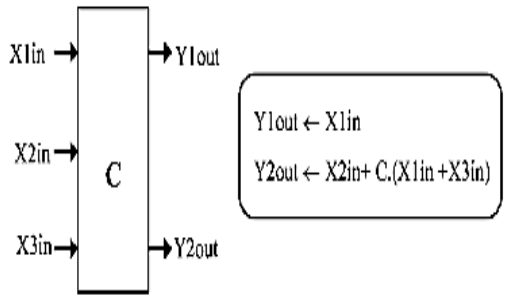

Figure 6: Structure of Functional Unit

The architecture of the 2D DWT is build with subcell-1 and subcell-2. Sub-cell-1 will be constructed using the multiplier and adders, its a functional unit which has input access of three input pixel values. Each sub cell will be built with four functional units which has dedicated lifting coefficient for multiplication. Functional block will have three input and two output.

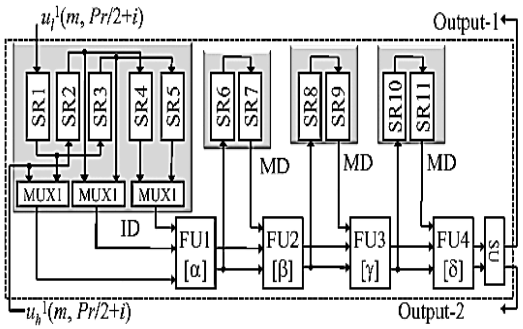

Figure 7: Internal Structure of Subcell-2

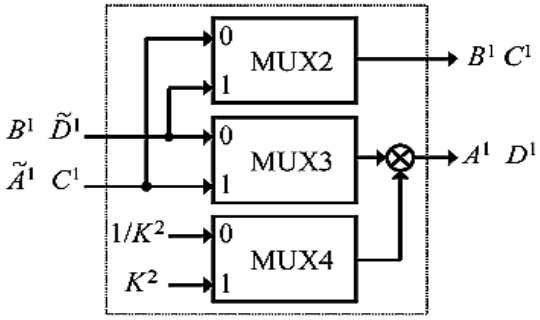

Figure 8: Structure of Scale Normalization Unit (SU)

The first input will be buffered and passed to the next unit and the first and third input will be added and the added result will be multiplied with lifting coefficient using the multipliers, the second input is added with the output of the multiplication to generated the second output of the functional unit.

The output of the subcell-1 is given as input to the subcell-2 which is the high frequency component and the low frequency components. This low frequency component will be processed again for the second dimension wavelet transform.

The sub cell -2 has the delay element and the subcell-1 structure along with scaling unit. The sub cell-1 and the sub cell-2 combine together to produce the processing element and the more number of processing element are connected in serious to build the processing unit which has the internal buffers which computes the multilevel lifting DWT.

MODIFICATION WITH VARIOUS FAST ADDERS

Proposed structure of conventional Array Multiplier with Ripple Carry Adder combination can be modified by replace the Ripple Carry Adder into different fast adders like Carry Save Adder, Square root Carry Select Adder, Binary to Excess-3 Adder.

Carry Save Adder Design

In proposed structure the ripple carry adder can be replaced by fast adder like Carry Save Adder (CSA). So the proposed structure provides the less delay and high throughput compared to existing structures.

Binary to Excess-3 Adder

Binary to Excess-3 adder can be produced the very less delay, area and high throughput compared to existing method of ripple carry adder combination. It is the very efficient structure than the other adder combination with array multiplier.

RESULTS AND COMPARISONS

The simulation results and synthesis of existing and proposed method shows the proposed method has less delay, less area and high throughput compared to existing method.

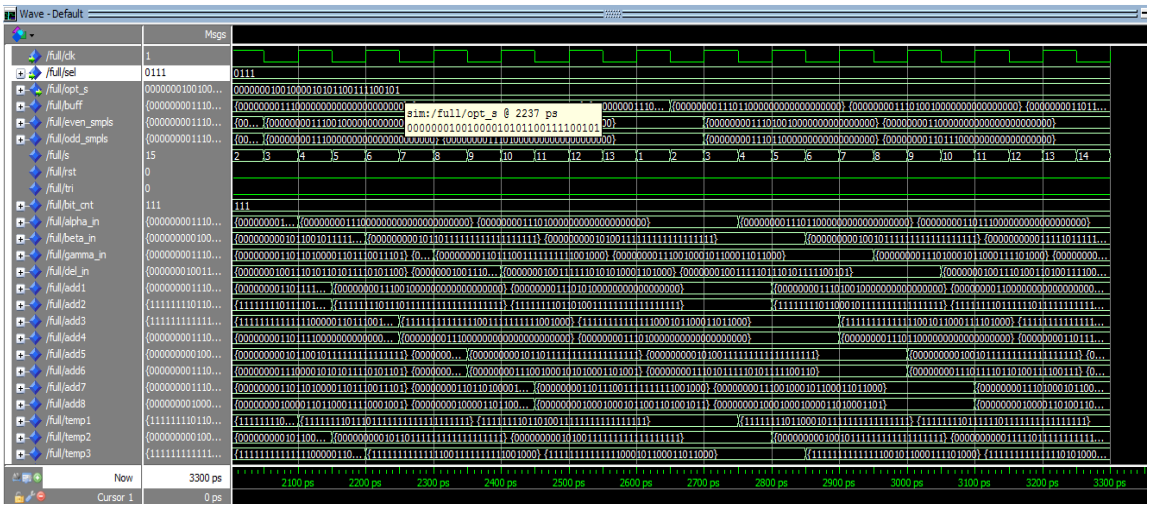

Figure 10: Simulation Results of Proposed Method

Array Multiplier with Carry Save Adder

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Total Number Slice Registers	385	66,560	1%	
Number used as Flip Flops	4			
Number used as Latches	381			
Number of 4 input LUTs	3,072	66,560	4%	
Logic Distribution				
Number of occupied Slices	1,692	33,280	5%	
Number of Slices containing only related logic	1,692	1,692	100%	
Number of Slices containing unrelated logic	0	1,692	0%	
Total Number of 4 input LUTs	3,116	66,560	4%	
Number used as logic	3,072			
Number used as a route-thru	44			
Number of bonded IOBs	33	633	5%	
IOB Latches	32			
Number of GCLKs	4	8	50%	
Total equivalent gate count for design	21,189			
Additional JTAG gate count for IOBs	1,584			

Figure 11: Design Summary of Array Multiplier with Carry Save Adder Combination

```
Timing Summary:
---------------
Speed Grade: -4

   Minimum period: 53.149ns (Maximum Frequency: 18.815MHz)
   Minimum input arrival time before clock: No path found
   Maximum output required time after clock: 7.078ns
   Maximum combinational path delay: No path found
```

Figure 12: Timing Summary of Array Multiplier with Carry Save Adder Combination

Array Multiplier with Binary to Excess-3 Adder Combination

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Total Number Slice Registers	372	66,560	1%	
Number used as Flip Flops	4			
Number used as Latches	368			
Number of 4 input LUTs	2,127	66,560	3%	
Logic Distribution				
Number of occupied Slices	1,191	33,280	3%	
Number of Slices containing only related logic	1,191	1,191	100%	
Number of Slices containing unrelated logic	0	1,191	0%	
Total Number of 4 input LUTs	2,178	66,560	3%	
Number used as logic	2,127			
Number used as a route-thru	51			
Number of bonded IOBs	33	633	5%	
IOB Latches	32			
Number of GCLKs	4	8	50%	
Total equivalent gate count for design	15,340			
Additional JTAG gate count for IOBs	1,584			

Figure 13: Design Summary of Array Multiplier with Binary to Excess-3 Adder Combination

```
Timing Summary:
---------------
Speed Grade: -4

   Minimum period: 39.861ns (Maximum Frequency: 25.087MHz)
   Minimum input arrival time before clock: No path found
   Maximum output required time after clock: 7.078ns
   Maximum combinational path delay: No path found
```

Figure 14: Timing Summary of Array Multiplier with Binary to Excess-3 Adder Combination

Comparison Result of Array Multiplier with Different Fast Adders

Table 1: Comparison of Array Multiplier with Different Fast Adders

Parameters	Array Multiplier with Ripple Carry Adder	Array Multiplier with Carry Save Adder	Array Multiplier with Square Root Carry Select Adder	Array Multiplier with Binary to Excess-3 Adder
Number of 4 input LUT's	3,161	3,116	2,930	2,178
Number of gate count for design	21,740	21,189	20,210	15,340
Estimated delay(ns)	53.935	53.149	48.85	39.861
Maximum Frequency(MHz)	18.451	18.815	20.471	25.087

CONCLUSIONS

The scalable architecture can be easily mapped by appropriate partitioning and appropriate scheduling of given VLSI system decomposition levels in regular pipeline. The proposed structure provides the area-constrained implementations to get high-throughput. All the redundancies can be removed by decimation process for maximize the HUE in the wavelet filtering. The frame-buffer and line-buffer can be saved in the proposed design. Comparing to the existing structures latency, the proposed design has very small latency. The proposed design has less arithmetic resources and higher throughput rate compared to existing recursive structure.

REFERENCES

1. Basant K. Mohanty, and Pramod Kumar Meher, "Memory Efficient Modular VLSI Architecture for Highthroughput and Low-Latency Implementation of Multilevel Lifting 2-D DWT," IEEE Trans. Signal Proces, vol. 59, no. 5, may 2011.
2. C. Cheng and K. K. Parhi, "High-speed VLSI implementation of 2-D discrete wavelet transform," IEEE Trans. Signal Process., vol. 56, no.1, pp. 393–403, Jan. 2008.
3. G. M. Davis, "A wavelet-based analysis of fractal image compression," IEEE Trans. Image Process., vol. 7, no. 2, pp. 141–154, Feb. 1998.
4. C.-T. Huang, P.-C. Tseng, and L.-G. Chen, "Analysis and VLSI architecture for 1-D and 2-D discrete wavelet transform," IEEE Trans. Signal Process., vol. 53, no. 4, pp. 1575–1586, Apr. 2005.
5. C.-T. Huang, P.-C. Tseng, and L.-G. Chen, "Flipping structure: An efficient VLSI architecture for lifting-based discrete wavelet transform," Signal Process., vol. 52, no. 4, pp. 1080–1089, Apr. 2004.
6. C.-T. Huang, P.-C. Tseng, and L.-G. Chen, "Generic RAM-based architectures for two-dimensional discrete wavelet transform with line based method," IEEE Trans. Circuits Syst. Video Technol., vol. 15, no. 7, pp. 910–920, Jul. 2005.
7. S. Lewis and G. Knowles, "Image compression using 2-D wavelet transform," IEEE Trans. Image Process., vol. 1, no. 2, pp. 244–250.
8. S. G. Mallat, "A theory for multi-resolution signal decomposition: The wavelet representation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 11, no. 7, pp. 674–693, Jul. 1989
9. Y. Meyer, Wavelets: Algorithms and Applications. Philadelphia, PA: Soc. Indust. Appl. Math. (SIAM), 1993.
10. L. Shutao, L. Chen, and J. T. Kwok, "Wavelet-based feature extraction for microarray data classification," in Proc. Int. Joint Conf. Neural Netw. (IJCNN '06*)*, Jul. 2006, pp. 5028–5033.
11. A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 still image compression standard," IEEE Signal Process. vol. 18, no. 5, pp. 36–58, Sep. 2001. Apr. 1992.
12. X. H. Wang, R. S. H. Istepanian, and H. S. Yong, "Application of wavelet maxima in microarray spots recognition," IEEE Trans. NanoBiosci., vol. 2, no. 4, pp. 190–192, Dec. 2003.
13. P.-C. Wu and L.-G. Chen, "An efficient architecture for two- dimensional discrete wavelet transform," IEEE Trans. Circuits Syst. Video. Technol., vol. 11, no. 4, pp. 536–545, Apr. 2001.