Filtering (Denoising) in the Wavelet Transform Domain

*Yousef M. Hawwar**, *Ali M. Reza**, *Robert D. Turney*+

*DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
UNIVERSITY OF WISCONSIN-MILWAUKEE
P. O. BOX 784 MILWAUKEE, WI 53201-0784
+CORE SOLUTIONS GROUP, XILINX, INC.
2100 LOGIC DRIVE SAN JOSE, CA 95124-3450

ABSTRACT

Filtering noise, in real time, has applications in speech and image processing. Considerable interest has arisen in recent years regarding filtering in the wavelet transform domain. This technique has been effective in noise removal with minimum side effects on important features such as image details and edges. In this paper, the effectiveness of both soft and hard thresholding for desired detail levels has been demonstrated. Efficient hardware implementation based on FPGA technology is proposed.

1. INTRODUCTION

Considerable interest has arisen in recent years regarding wavelet as a new transform technique for both speech and image processing applications. This technique has shown effective results in several applications such as image compression, edge detection, feature extraction, and nonlinear noise filtering [1][2]. In this work, several nonlinear filtering techniques, generally referred to as denoising in the wavelet literature, are studied. Our main emphasis in this work is in the hardware implementation of denoising algorithms for real-time image processing.

The method of wavelet denoising has been researched extensively. This approach is generally simple and effective. The processing in this class of algorithms is carried on in the transform domain. In this approach, the discrete wavelet transform (DWT) of a signal is calculated and the resultant wavelet coefficients are passed through a threshold testing. In this case, the coefficients that are smaller than a certain value are removed. Then the resultant coefficients are used to reconstruct the signal. With this method, it is possible to remove noise with little loss of details. If a signal has its energy concentrated in a small number of wavelet coefficients, its coefficient values will be relatively large compared to the noise that has its energy spread over a large number of coefficients.

In traditional Fourier based signal processing; the spectrum of the signal is assumed to have little overlap with spectrum of the noise and therefore a linear time-invariant filtering will be employed. This linear filtering approach cannot separate noise from signal where their Fourier spectra overlap. In DWT analysis, the method is entirely different. The idea in this case is based on the assumption that the amplitude, rather than the location, of the spectra of the signal to be as different as possible for that of the noise. This allows clipping, thresholding, and shrinking of the amplitude of the coefficients to separate signals or remove noise. It is the localizing or concentrating properties of the wavelet transform that makes it particularly effective when used with this nonlinear filtering method [2].

Thresholding generally gives a lowpass and "smoother" version of the original noisy signal.

The objective is to suppress the additive noise $w(k)$ from the signal $x(k)$, where $x(k) = u(k) + w(k)$. The signal $x(k)$ is first decomposed into L-level of wavelet transform. Then, the thresholding of the resultant wavelet coefficients, for noise suppression, is carried out. The thresholding is based on a value δ that is used to compare with all the detailed coefficients. Two types of thresholding are more popular. One is based on hard thresholding and the other one is based on soft thresholding. Both of these, methods are discussed in this work.

This paper is organized as follow: In the next section, the 1-D DWT, used for transforming the signal, is presented. The denoising process in the transform domain is discussed in Section 3. The proposed algorithm for FPGA hardware implementation of the denoising technique is presented in Section 4. In Section 5, preliminary results of the denoising algorithm when applied to 2-D images are reported.

2. 1-D Discrete Wavelet Transform

The general form of an L-level DWT is written in terms of L detail sequences, $d_j(k)$ for $j = 1,2,\cdots,L$, and the L–th level approximation sequence, $c_L(k)$ as follows:

$$x(t) = \sum_{j=1}^{L} \sum_k d_j(k)\psi_j(t) + \sum_k c_L(k)\varphi_L(t) \tag{1}$$

In (1), $\varphi_L(t)$ is the L–th level scaling function and $\psi_j(t)$ for $j = 1,2,\cdots,L$ are wavelet function sequences for L different levels.

In order to work directly with the wavelet transform coefficients, the relationship between the detailed coefficients at a given level in terms of those at previous level is used. In general, the discrete signal is assumed the highest achievable approximation sequence, referred to as 0-th level scaling coefficients. It is shown [3] that the approximation and detail sequences at level $j+1$ are related to the approximation sequence at level j by

$$c_{j+1}(k) = \sum_m h_o(m-2k)c_j(m) \tag{2}$$

and

$$d_{j+1}(k) = \sum_m h_1(m-2k)c_j(m) \tag{3}$$

Equations (2) and (3) state that approximation sequence at higher scale (lower level index), along with the wavelet and scaling filters, $h_o(k)$ and $h_1(k)$ respectively, can be used to calculate the detail and approximation sequences (or *discrete wavelet transform* coefficients) at lower scales.

In practice, a discrete signal, at its original resolution is assumed the 0-th level approximation sequence; i.e., $c_0(k) = x(k)$. For a given wavelet system, with known wavelet filters $h_o(k)$ and $h_1(k)$, it is possible to use (2) and (3), in a recursive fashion, to calculate the discrete wavelet transform coefficients at all desired lower scales (higher lever). In most engineering applications, the wavelet systems are chosen such that the two wavelet filters have finite number of non-zero coefficients. In signal processing terminology, these filters are referred to as *finite impulse response* (FIR) filters. Under this assumption, and by using ideas from multirate signal processing literature [1], it is possible to calculate the two summations in (2) and (3) by using two FIR filters.

In this paper, the number of decomposition level used is assumed to be five, namely $L = 5$. Data rate of the incoming signal is assumed to be 80 MHz. After each level of decomposition, the input data is branched into two outputs, one associated to the upper half-band of the input signal and the other one to the lower half-band. In each branch, the rate of data is half of that of the input data to that branch. Therefore, the overall rate of data remains constant. Since decomposition of the DWT proceeds only on one branch (the lower half-band branch), there will be different data

Levels	Level 0 $x(k)$	Level 1 $d_1(k)$	Level 2 $d_2(k)$	Level 3 $d_3(k)$	Level 4 $d_4(k)$	Level 5 $d_5(k)$	Level 5 $c_5(k)$
Rates	80 MHz	40 MHz	20 MHz	10 MHz	5 MHz	2.5 MHz	2.5 MHz

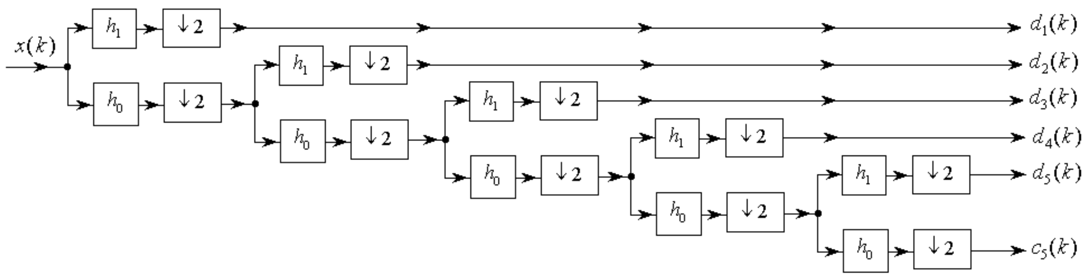

Figure 1 – Five level decomposition tree of the discrete wavelet transform (DWT).

D 1	D 2	D 1	D 3	D 1	D 2	D 1	D 4	D 1	D 2	D 1	D 3	D 1	D 2	D 1	D 5	D 1	D 2	D 1	D 3	D 1	D 2	D 1	D 4	D 1	D 2	D 1	D 3	D 1	D 2	D 1	C 5

Figure 2 - The structure of the output coefficients ordering for having a fixed 80 MHz data rate at the output of the five-level DWT decomposition.

rates for different branches that provide output data. For examples, data rate at level 1 output is 40 MHz, at level 2 output is 20 MHz and so on. For the five-level decomposition used in this work, the decomposition tree and rates are shown in Figure 1. This multirate output data require special handling for proper output transmission in the hardware implementation. With a particular arrangement, It is possible to output the overall data, at the fixed rate of the input data. This arrangement requires proper ordering of the output of different decomposition level. The overall rate of the output coefficients, which is obtained by adding the rates of the L detail levels and the fifth coarse level, is the same as the input rate 80MHz. It is therefore possible to transmit the output with a proper ordering such that the overall rate remains constant. The general structure of the ordering of the output stream is as shown in Figure 2.

3. Denoising 1-D DWT coefficients

In this section, the procedure for denoising, applied to detail sequences is discussed. The approximate sequence at the fifth level is obtained by processing the original signal through five stages of lowpass filtering. It is therefore unnecessary to use the approximate sequence, $c_L(k)$, in the denoising process. One assumption that we use in denoising is that the statistics of additive Gaussian noise is known. It is known that soft thresholding provides smoother results in comparison with the hard thresholding technique. Hard thresholding technique, however, provides better edge preservation in comparison with the soft thresholding. Based on these properties, we decided to apply the soft thresholding technique to the first two detail sequences and hard thresholding technique to the remaining detail sequences. By this method, we are able to remove the noise from the signal without disturbing important signal features [2][4].

Hard thresholding is the usual process of setting to zero the coefficients whose absolute values are lower than the threshold. Soft thresholding is an extension of hard thresholding by first setting to zero coefficients whose absolute values are lower than the threshold and then shrinking the nonzero coefficients toward zero. For threshold δ, the hard thresholding technique is obtained by

$$\tilde{d}_j^H(k) = \begin{cases} d_j(k) & \text{if } |d_j(k)| > \delta \\ 0 & \text{if } |d_j(k)| \le \delta \end{cases} \tag{4}$$

In the soft thresholding case, for the same threshold, the calculation is given by

$$\tilde{d}_j^S(k) = \begin{cases} \operatorname{sign}\left(d_j(k)\right) \cdot \left(|d_j(k)| - \delta\right) & \text{if } |d_j(k)| > \delta \\ 0 & \text{if } |d_j(k)| \le \delta \end{cases} \tag{5}$$

Where sign(x) is +1 if x is positive and –1 if x is negative. Implementation of this denoising approach is discussed in the next section.

4. FPGA Implementation

Implementation of the proposed denoising algorithm is performed by using Xilinx Virtex FPGA. In this discussion, implementation of the DWT (wavelet decomposition) and IDWT (signal reconstruction) are presented in block diagrams but details of soft and hard thresholding are presented for FPGA implementation. The calculation of the 1D DWT or IDWT requires 2500 Logic Cells to support a 80 MSa/s bandwidth. In the Virtex Family this would require a XCV100 device to perform DWT or IDWT.

Architecture of the denoising system is shown in Figure 3, which includes both soft and hard thresholding . The coefficient and the threshold value δ are input on a sample basis. Using available Virtex technology, operation at 80 MHz is easily achievable. System clock rates of 100 MHz and 120 MHz are also available for more aggressive system bandwidth requirements. Calculation of the data path for this algorithm is straightforward. The architecture requires 144 Logic Cells. Each coefficient will be compared to a unique threshold as shown in Figure 3. When the input coefficient is greater than the threshold, the output is logical value of `1'. If, however, the coefficient is less than or equal to the threshold, the output is logical value of '0'.

The decision of which stage should use hard or soft thresholding is flexible and adaptable. Figure 4 shows an example for the denoising of a 1-D signal.

5. Denoising for 2-D image

Application of the proposed denoising algorithm is extended to the 2-D images by using similar strategy for thresholding the detailed coefficients. Implementation of the denoising algorithm is similar to that of the 1-D case. Figure 5 shows the output of the implementation of the denoising algorithm on the Barbara image.

6. Conclusion

General problem of signal and image denoising has been discussed and some simulation results have been presented. VLSI implementation for the developed algorithm, using Xilinx FPGA has been presented. Pipelining of the algorithm allows it to be used for real- time speech and video processing.

References

[1] Strang, G. and Nguyen, T., *Wavelets and Filter Banks*. Wellesley-Cambridge Press, Wellesley, Massachusetts, 1996.

[2] Dohono, D. L. and Johnston, I. M., “Ideal spatial adaptation via wavelet shrinkage,” *Biometrika*. Vol. 81, pp. 425-455, 1994.

[3] Burrus, C. S.; Gopinath, R. A.; and Guo, H., *Introduction to Wavelets and Wavelet Transforms: A Primer*. Prentice Hall Inc., Upper Saddle River, New Jersey, 1998.

[4] Zong, X.; Laine, A.; and Geiser, E., “Speckle reduction and contrast enhancement of echocardiograms via multiscale nonlinear processing,” *IEEE Transactions on Medical Imaging*. Vol. 17, No. 4, August 1998.

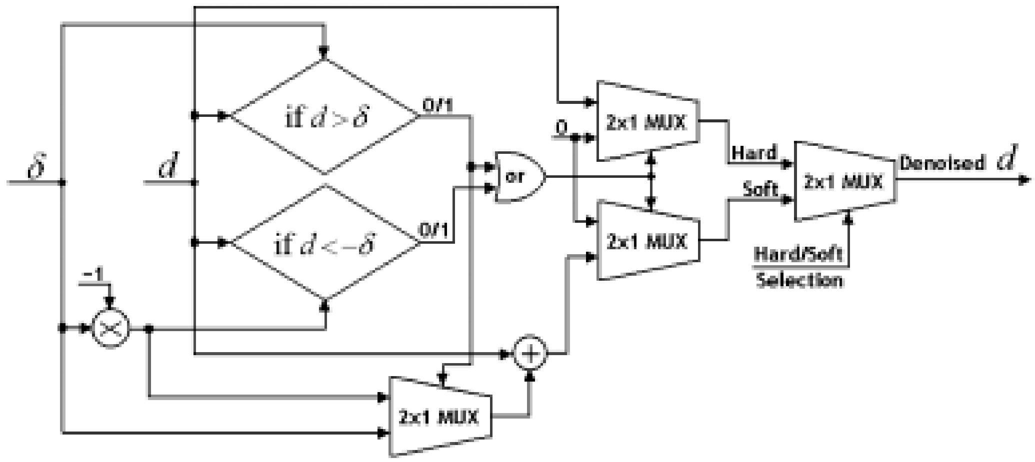

Figure 3 - Block diagram of the thresholding procedure, including both hard and soft thresholding.

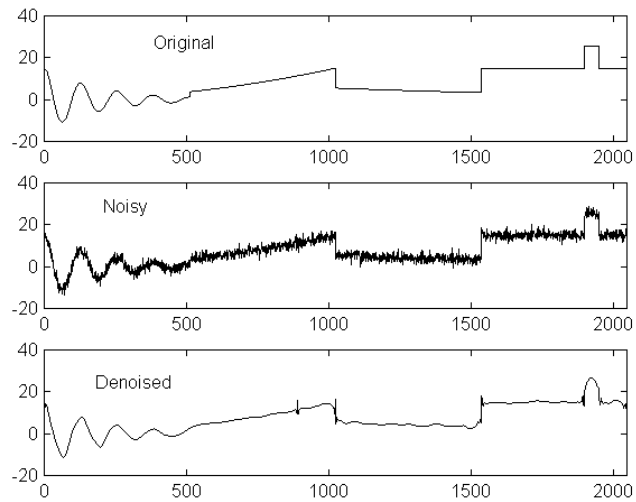

Figure 4 - Result of the denoising algorithm for a 1-D signal.

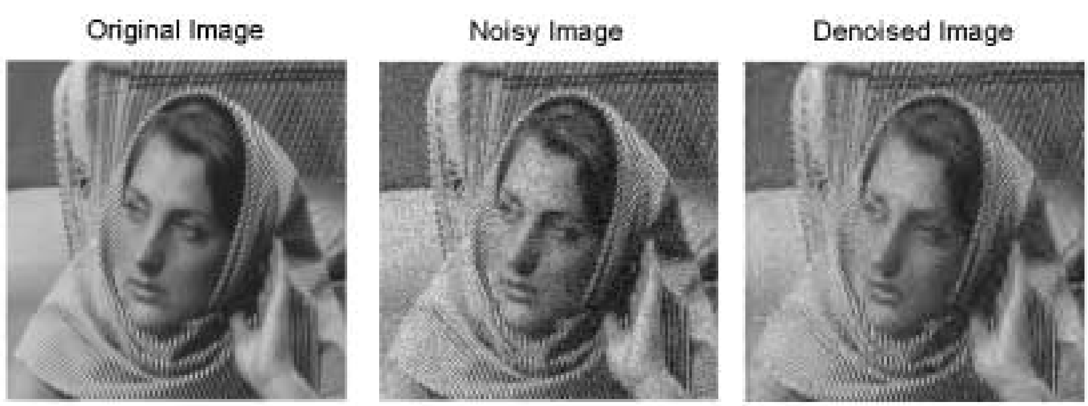

Figure 5– Original, noisy and the filtered image using the proposed algorithm.