We are IntechOpen, the world's leading publisher of Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Low Complexity Implementation of Daubechies Wavelets for Medical Imaging Applications

Khan Wahid
University of Saskatchewan,
Canada

1. Introduction

The Discrete Wavelet Transform (DWT) has extensively been used in a wide range of applications, including numerical analysis, image and video coding, pattern recognition, medical and telemetric imaging, etc. The invention of DWT decomposition by Mallat (Mallat, 1998) shows that the DWT can be viewed as a multiresolution decomposition of signal. This means it decomposes the signal into its components in different frequency bands. The Inverse DWT does the opposite, i.e. it reconstructs the signal from its octave band components. After its inclusion in JPEG2000 compression standard (Seo & Kim, 2007), significant research has been done to optimize the DWT implementation to reduce the computational complexity. Among a wide range of wavelets, the Daubechies wavelets include members ranging from highly localized to highly smooth and can provide excellent performance in image compression (Daubechies, 1992). Among the family members, the first two – Daubechies 4-tap (DAUB4) and Daubechies 6-tap (DAUB6) – are popular choices in medical imaging applications.

While compressing medical images, the key here is to preserve as much critical information as possible in the reconstructed image so that accurate diagnosis is possible. There have been several efficient implementations of wavelet filters proposed for applications in image processing (Lee & Lim, 2006; Martina & Masera, 2007; Acharyya et al., 2009; Shi et al., 2009; Lai et al., 2009). But, the use of conventional fixed-point (FP) binary (or any other weighted) representation for implementing discrete wavelet coefficients (that are irrational in nature) introduces round-off or approximation errors at the very beginning of the process. The error is due to the lack of exact representation of the irrational numbers that form the coefficient basis. These errors tend to expand as the calculations progress through the architecture, degrading the quality of image reconstruction (Wahid et al., 2003). A lossless mapping technique, known as Algebraic Integer Quantization (AIQ), can be used to minimize the approximation error and efficiently compute the DAUB4 and DAUB6 coefficients (Wahid et al., 2004). The AIQ scheme is divided into two parts: the first stage is based on factorization and decomposition of transform matrices exploiting the symmetric structure. After the decomposition, we map the irrational transform basis coefficients using multidimensional algebraic integers that results in exact representation and simpler implementation. As a result, less error is introduced in the computation process that yields significantly better

reconstruction of images while keeping critical information, making the scheme suitable for medical and telemetric imaging applications.

As a case study, we apply the scheme to several medical images, such as endoscopic, ultrasound, x-ray, CT-scan images and evaluate the performance. The chapter is organized as follows: Previous related works are presented next. Section 3 presents a brief introduction to Daubechies wavelets. In Section 4, we explain the AIQ scheme applied to Daubechies wavelets. Then the simulation and synthesized results of the case study are summarized in Section 5. Finally, we conclude the work in Section 6.

2. Past work

Lewis and Knowles proposed an architecture for Daubechies wavelets without multipliers (Lewis & Knowles, 1991). A major drawback was that it was heavily dependent on the properties of only one specific wavelet, DAUB4 tap coefficients. At the same time, Aware Inc. came out with a chip called Wavelet Transform Processor (WTP) (Aware, 1991). It essentially consists of a 4-tap filter (4 Multiply-Accumulate cells) and some external memory with control but no specific features that can take advantage of the DWT structure rather it relies heavily on the software to compute the DWT. It is also a complex design requiring extensive user control. Parhi and Nishitani proposed two architectures, folded and digit serial, for 1D DWT (Parhi & Nishitani, 1993). These architectures do not easily scale with the filter size and the number of octaves computed. The number of multipliers is higher, and hence the silicon area is large. In (Vishwanath et al., 1995), the authors proposed linear systolic array architecture. Paek and Kim in proposed recursive and semi-recursive architectures for DWT which has several drawbacks like large area (hardware cost), scheduling control overhead and incomplete data-bus utilization (Paek & Kim, 1998).

Most of the research work to reduce the hardware complexity is inclined towards multiplierless implementations by maneuvering the filter banks (Lee & Lim, 2006; Martina & Masera, 2007; Acharyya et al., 2009) or using lifting schemes (Shi et al., 2009; Lai et al., 2009; Huang et al., 2004). However, in these designs, the use of conventional FP binary representation results in erroneous computation process and degrades image reconstruction. In this chapter, we present an efficient low-cost implementation of the DAUB filters with a demonstration of performance advantages on medical images and noisy environment.

3. Daubechies wavelets

This section provides a brief introduction to Daubechies wavelets. This class of wavelets includes members ranging from highly localized to highly smooth – Daubechies-2 (DAUB2 with two coefficients) to Daubechies-20 (DAUB20 with 20 coefficients) and also provides excellent performance in image compression (Daubechies, 1992). The Daubechies wavelet coefficients are based on computing wavelet coefficients, C_n (where, n = 0, 1, 2,..., N-1 and N is the number of coefficients) to satisfy the following conditions (Mallat, 1998):

1. The conservation of area under a finite length signal $x(t)$: $\sum C_n = 2$
2. The accuracy conditions: $\sum (-1)^n n^m C_n = 0$ (where m = 0, 1, 2,..., p-1 and $p = N/2$)
3. The perfect reconstruction conditions: $\sum_n C_n^2 = 2$ and $\sum_n C_n C_{n+2m} = 0$

Then the low-pass filter is $h(n) = \dfrac{C_n}{2}$ and the high-pass filter is $g(n) = (-1)^{n+1} h(n - N - 1)$.

One of the simplest and most localized members is the DAUB6 which has six coefficients:

$$C_0 = \frac{(1 + z_1 + z_2)}{16\sqrt{2}} \qquad C_1 = \frac{(5 + z_1 + 3z_2)}{16\sqrt{2}} \qquad C_2 = \frac{(10 - 2z_1 + 2z_2)}{16\sqrt{2}}$$

$$\tag{1}$$

$$C_3 = \frac{(10 - 2z_1 - 2z_2)}{16\sqrt{2}} \quad C_4 = \frac{(5 + z_1 - 3z_2)}{16\sqrt{2}} \qquad C_5 = \frac{(1 + z_1 - z_2)}{16\sqrt{2}}$$

Where, $z_1 = \sqrt{10}$ and $z_2 = \sqrt{5 + 2\sqrt{10}}$. For an 8x8 input data, the DAUB6 forward transform matrix (using an assumption of periodicity) is shown in Eq. (2):

$$\psi_6(C) = \begin{bmatrix} C_0 & C_1 & C_2 & C_3 & C_4 & C_5 & 0 & 0 \\ C_5 & -C_4 & C_3 & -C_2 & C_1 & -C_0 & 0 & 0 \\ 0 & 0 & C_0 & C_1 & C_2 & C_3 & C_4 & C_5 \\ 0 & 0 & C_5 & -C_4 & C_3 & -C_2 & C_1 & -C_0 \\ C_4 & C_5 & 0 & 0 & C_0 & C_1 & C_2 & C_3 \\ C_1 & -C_0 & 0 & 0 & C_5 & -C_4 & C_3 & -C_2 \\ C_2 & C_3 & C_4 & C_5 & 0 & 0 & C_0 & C_1 \\ C_3 & -C_2 & C_1 & -C_0 & 0 & 0 & C_5 & -C_4 \end{bmatrix} \tag{2}$$
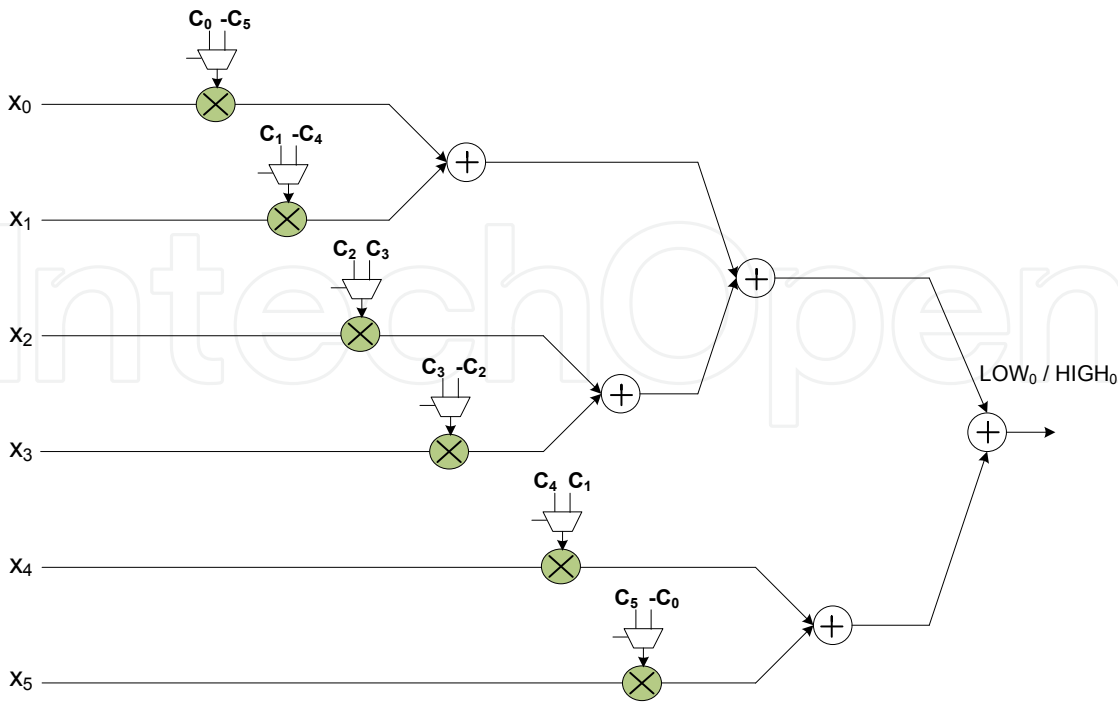

Fig. 1. FP-based DAUB6 filter architecture.

The structure of the matrix uses the set of coefficients, $\{C_0, C_1, ..., C_5\}$ as a smoothing filter (low-pass) and the set, $\{C_5, -C_4, ..., -C_0\}$ as a non-smoothing filter (high-pass). The DWT is invertible and orthogonal - the inverse transform, when viewed as a matrix, is simply the transpose of the forward transform matrix. So, basically we need only 2 sets of multiply-accumulate (MAC) cells each containing 6 multipliers and 5 adders where partial products are computed separately and subsequently added. However, it can be seen that, by introducing additional control circuitry, the same multipliers can be used for both low-pass and high-pass filtering. As a result, the number of multipliers can be reduced to 6 instead of 12. Fig. 1 shows the signal flow graph of the conventional finite-precision (FP) implementation of Eq. (2), where x_i is the input data vector. Since all the coefficients are fixed, for a fixed precision, we can in fact replace all the multipliers by adders and shifters. As a result, the total equivalent additions required to compute the 1-D DAUB6 filter is 44.

4. AIQ-based algorithm

Algebraic integer (AI) is defined by real numbers that are roots of monic polynomials with integer coefficients (Wahid et al., 2004). As an example, let $\omega = e^{\frac{2\pi j}{16}}$ denote a primitive 16th root of unity over the ring of complex numbers. Then ω satisfies the equation: $x^8 + 1 = 0$. The ring $Z(\omega)$ can be regarded as consisting of polynomials in ω of degree 7 with integer coefficients. The elements of $Z(\omega)$ are added and multiplied as polynomials, except that the rule $\omega^8 = -1$ is used in the product to reduce the degree of powers to below 8.

In summary, algebraic integers of an extension of degree n can be assumed to be of the form:

$$a_0\omega_0 + a_1\omega_1 + ... + a_{n-1}\omega_{n-1} \tag{3}$$

Where, $\{\omega_0, \omega_1, ..., \omega_{n-1}\}$ is called the AI basis and the coefficients a_i are integers. The process of mapping with AI is known as Algebraic Integer Quantization (AIQ).

The AIQ technique is useful in computing discrete transforms as first explored by Cozzens and Finkelstein (Cozzens & Finkelstein, 1985). In their work, the algebraic integer number representation, in which the signal sample is represented by a set of (typically four to eight) small integers, combines with the Residue Number System (RNS) to produce processors composed of simple parallel channels. The analog samples must first be quantized into the algebraic integer representation and the final algebraic integer result converted back to an analog or digital form. In between these two conversions, the algebraic integer representation must be converted into and out of two levels of RNS parallelism.

4.1 AIQ-based Daubechies wavelets

Now we present the concept of AIQ to encode the DAUB6 coefficients. First of all, consider the polynomial of two variables:

$$f(z_1, z_2) = \sum_{i=0}^{1} \sum_{j=0}^{1} a_{ij} z_1^i z_2^j = \begin{bmatrix} a_{00} & a_{10} \\ a_{01} & a_{11} \end{bmatrix} \begin{bmatrix} 1 & z_2 \\ z_1 & z_1 z_2 \end{bmatrix} \tag{4}$$

So the corresponding coefficients, a_{ij}, are encoded in the form $\begin{bmatrix} a_{00} & a_{10} \\ a_{01} & a_{11} \end{bmatrix}$. Then all the DAUB6 coefficients are exactly encoded (scaled by $16\sqrt{2}$) and shown below in Eq. (5):

$$C_0 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \qquad C_1 = \begin{bmatrix} 5 & 1 \\ 3 & 0 \end{bmatrix} \qquad C_2 = \begin{bmatrix} 10 & -2 \\ 2 & 0 \end{bmatrix}$$

$$C_3 = \begin{bmatrix} 10 & -2 \\ -2 & 0 \end{bmatrix} \qquad C_4 = \begin{bmatrix} 5 & 1 \\ -3 & 0 \end{bmatrix} \qquad C_5 = \begin{bmatrix} 1 & 1 \\ -1 & 0 \end{bmatrix} \tag{5}$$

The obvious advantages of this approach are: a) Very small dynamic range (numbers ranging from 0 to 10); b) Multiplication by a constant is very easy and efficient (only 1 addition is required in most cases; so, multiplication can be eliminated by add/shift algorithm); and c) We have 3 parallel channels through which data flows independently (since a_{11} is zero for all) and also a very simple scheduling is needed. No quantization errors would be incurred. Fig. 2 shows the signal flow graph of the AIQ-based scheme that requires 42 adders.
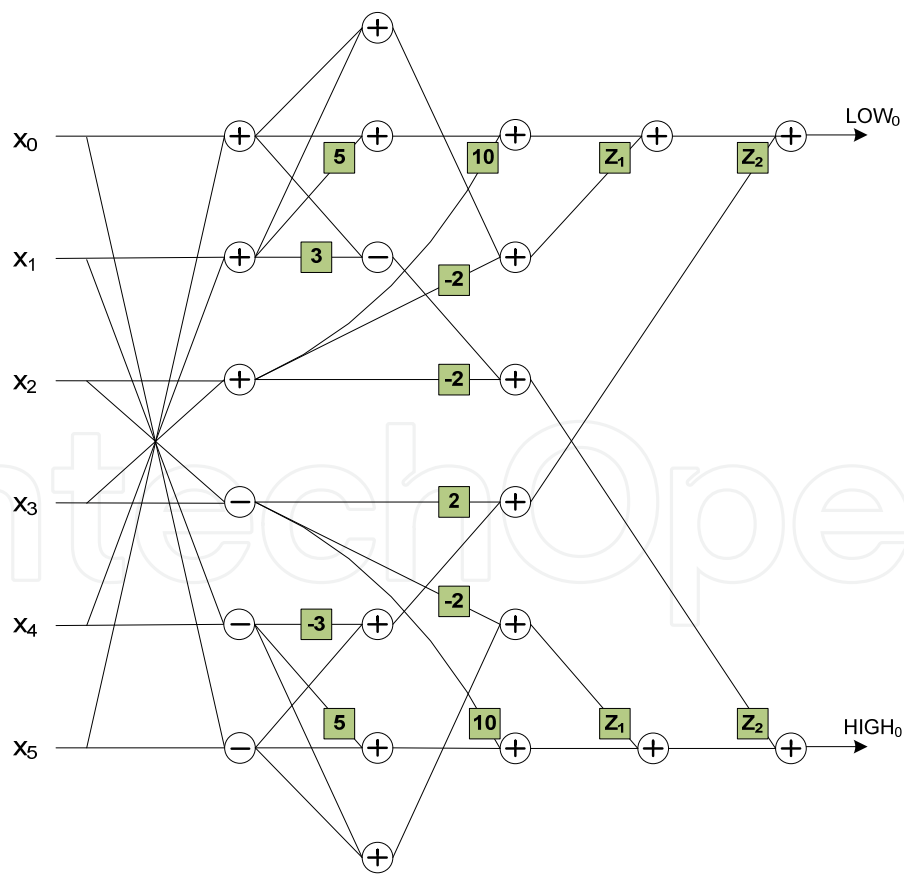

Fig. 2. AIQ-based DAUB6 filter architecture.

The encoding scheme can be easily applied to DAUB4 coefficients. In that case, we need a 2nd degree polynomial of one variable:

$$f(z_3) = a_0 + a_1 z_3 + a_2 z^2_3 \tag{6}$$

Where, $z_3 = \sqrt{3}$. As a result, the error-free mapping of DAUB4 coefficients can computed as given below (scaled by $4\sqrt{2}$):

$$C_0 = 1 + z_3 \, ; \, C_1 = 3 + z_3 \, ; \, C_2 = 3 - z_3 \, ; \, C_3 = 1 - z_3 \tag{7}$$

4.2 Hardware implementation

The AIQ-based architecture is coded in Verilog and prototyped onto Xilinx VirtexE FPGA to assess the performance. A precision of 8-bit is used in the multipliers to minimize the hardware and optimize the operation, which is completed in one clock cycle (CC). Table 1 presents the comparison of the synthesized results along with the fixed-point (FP) designs. It can be seen from the table that the AIQ scheme for both DAUB4 (D4) and DAUB6 (D6) costs lesser hardware resources and has lower critical path delay; in case of DAUB4, the savings is even higher.

Scheme	Fixed-point		AIQ		Overall savings (%)	
	D4	D6	D4	D6	D4	D6
Datapath	4	6	3	6	25	0
Adders	32	44	16	42	50	5
LUTs	268	364	124	340	54	7
Registers	422	520	200	494	53	5
Critical path	Tm+2Ta	Tm+3Ta	5Ta	6Ta	--	--
Frequency (MHz)	98	112	148	120	--	--

[1]Xilinx VirtexE (xcv300epq240-8); D4 = DAUB4; D6 = DAUB6; Tm = latency for multiply operation; Ta = latency for addition operation

Table 1. FPGA[1] implementation and hardware comparison.

A key advantage of the AIQ approach is error reduction. In order to better understand the sources of error induction, we present in Fig. 3 the entire process of digital computation process. Here each arrow represents one key stage in the computation. In case of FP-based approach, there are three stages of error accumulation (or induction). In stage 1, the quantization (or approximation) error is introduced due to the lack of finite representation of transform basis coefficients that are irrational numbers. The error gets larger and larger in later stages as the process continues.

However, in AIQ-based approach, there is only one stage of error induction at the end: in stage 1, the basis coefficients are mapped with algebraic integers (that is error-free); then all the required computations take place (that is also error-free) using the AI representation. Finally, the data is converted back to binary (in stage 3) where some errors may be introduced; but these errors are much less compared to FP-approach, and are introduced very late in the computation process, and hence less affect the quality of image reconstruction.
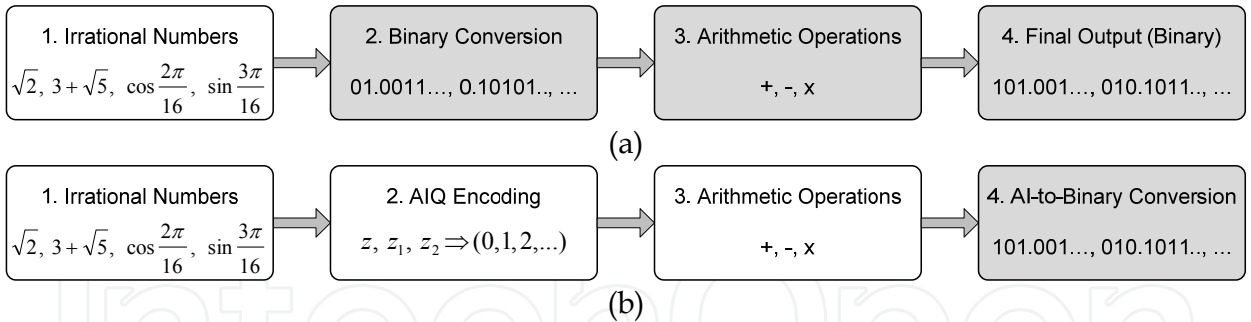
Fig. 3. Stages in computation process (error induced in shaded blocks): (a) FP approach; (b) AIQ approach.

Moreover, the error introduced at the final stage in AIQ approach can be further minimized using higher precision AIQ multipliers. We have performed an error analysis that shows the error incurred for different bit-length of the AIQ multipliers (in Fig. 4). The error is computed taking a multiplier of 16-bit width as reference. The signed digit representation (for 8-bits) is shown below in Eq. (8):

$$z_1 = \sqrt{10} \approx 11.001010 = 3 + 2^{-3} + 2^{-5}$$
$$z_2 = \sqrt{5 + 2\sqrt{10}} \approx 100.\bar{1}0\bar{1}100 = 4 - 2^{-1} - 2^{-3} \quad (8)$$
$$z_3 = \sqrt{3} \approx 10.0\bar{1}00\bar{1}0 = 2 - 2^{-2} - 2^{-5}$$
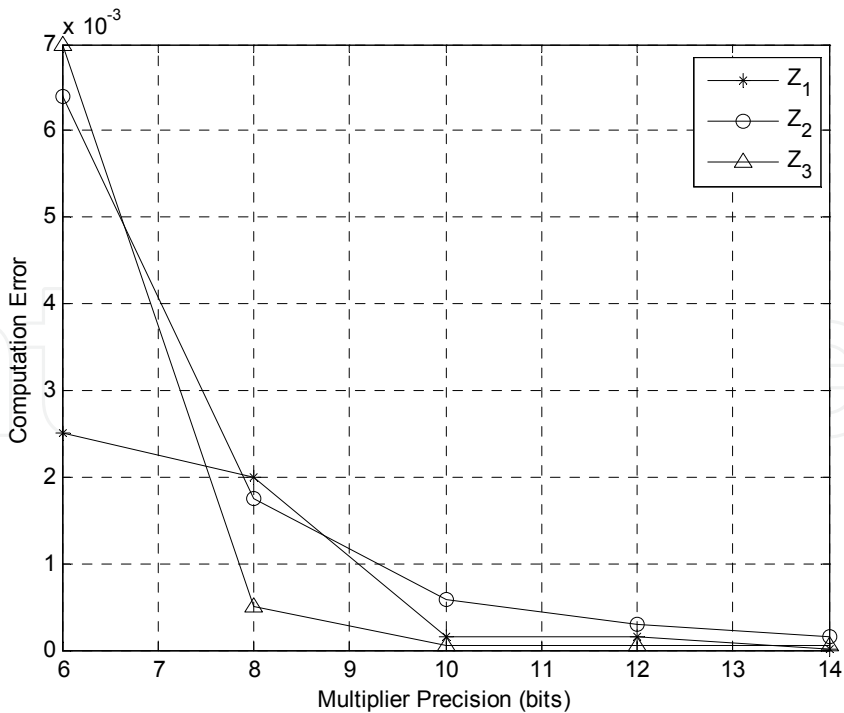

Fig. 4. Computation error in AIQ multipliers.

5. Performance evaluation

The AIQ-based algorithm to compute the Daubechies Wavelet Transform is intended to be used in applications where the quality of image reconstruction is critical, such as biomedical imaging, telemedicine, capsule endoscopy (Wahid et al., 2008), etc. Due to the error-free nature of integer mapping, the AIQ approach results in a much better reconstruction compared to conventional binary approach. Here, we present the results of our study, where we apply the scheme to several standard benchmark and medical images, such as endoscopic, ultrasound, x-ray, CT-scan images and evaluate the performance.

The section is divided into four sub-sections. In the first section, we evaluate the performance of the AIQ scheme for standard images followed by the analysis of medical images. Next, we show the performance of the scheme in a noisy environment. Finally, the results are compared with existing works related to medical image compression. In all these cases, we have used peak-signal-to-noise-ratio (PSNR) as the visual quality assessment index which is given by Eq. (9):

$$PSNR = 20 \times \log_{10} \frac{255}{\sqrt{\frac{1}{M \times N} \sum_{n=1}^{N} \sum_{m=1}^{M} \left(x_{m,n} - x'_{m,n}\right)^2}} \tag{9}$$

Where, M and N are the image width and height namely; x and x' are the original and reconstructed component values namely.

5.1 Performance analysis on standard images

Here, we perform 1-D forward transform on the benchmark "Goldhill" image followed by the inverse transform to get the original image back. No compression was performed, so the image quality degradation is purely due to arithmetic quantization effects. Different hardware precision (number of bit) is used where a full adder is considered as the unit for the hardware cost, and making a simple assumption of a hardware cost of n for an n-bit word (this will be a best case comparison for the fixed-point binary implementation). The results are shown in Fig. 5 and 6.

As shown in Fig. 5(a), the PSNR of the reconstructed image gets higher with the increased bit precision which is expected. In all cases, the AIQ performs much better that FP, i.e., the data recovery rate is higher, especially in lower bit rate region. In other words, for a fixed level of distortion, the number of bits required to transmit the transformed coefficients of AIQ approach would be less than those required for FP technique.

In Fig. 5(b), we present an analysis of reconstruction quality (in PSNR) with the cost of implementation. An interesting comparison is to select similar hardware cost and then compare the reconstruction performance. As an example, for same hardware cost of 150, compare the PSNR of both FP (44dB) and AIQ implementation (82dB). In this case the difference in the PSNR is 38dB in favour of AI – that is for similar cost of implementation, the AIQ scheme produces much better image reconstruction. On the other hand, for same PSNR, say 60 dB, AIQ scheme (95) requires around three times less hardware than FP implementation (270).
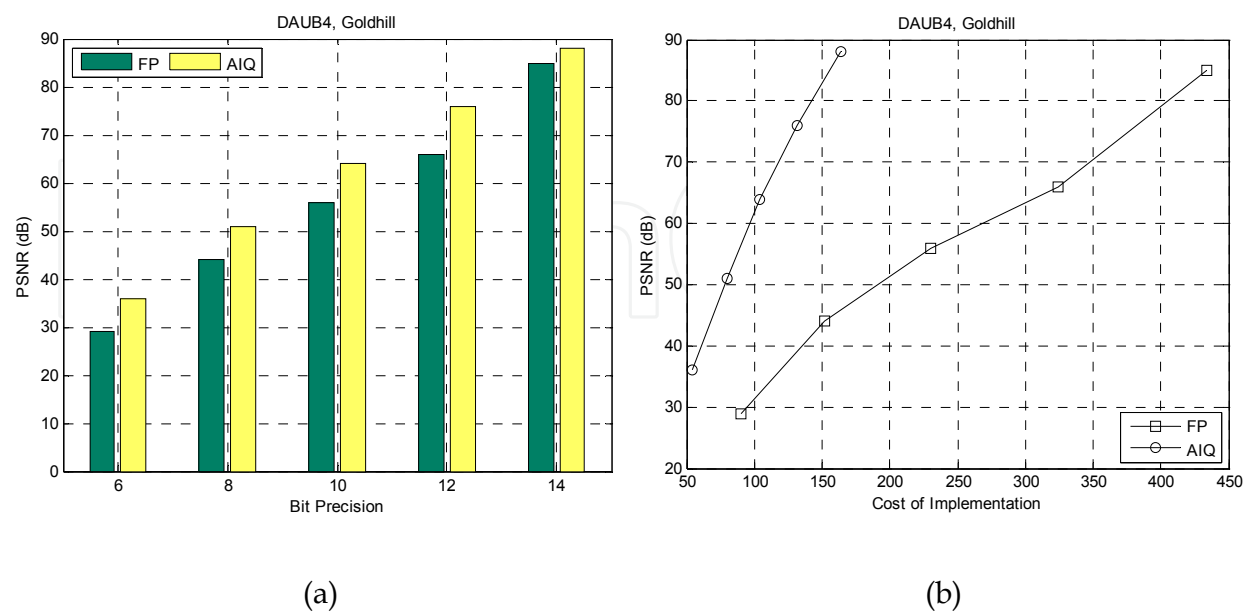
(a)   (b)

Fig. 5. Performance analysis of DAUB4 – FP vs. AIQ: (a) PSNR vs. Precision; (b) PSNR vs. Hardware cost.

(a)   (b)
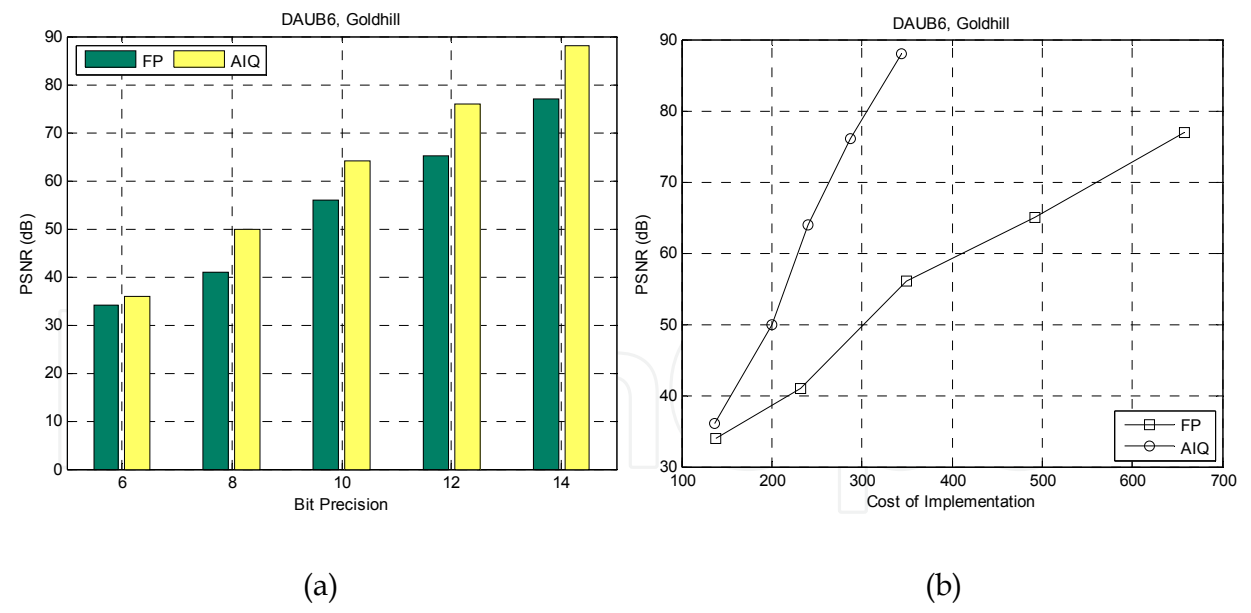
Fig. 6. Performance analysis of DAUB6 – FP vs. AIQ: (a) PSNR vs. Precision; (b) PSNR vs. Hardware cost.

Same kind of superiority is seen for DAUB6 (Figure 6) too. So, not only an improvement in image reconstruction quality is obtained but also hardware cost is reduced. Fig. 7 shows the image reconstruction for a 8-bit FP vs. a 14-bit AIQ for DAUB4. The difference in PSNR is about 45dB and the level of improvement is quite noticeable.

(a) Original   (b) PSNR = 43dB   (c) PSNR= 88dB

Fig. 7. (a) Original goldhill image; (b) Reconstructed image using FP scheme (8-bits); (c) AIQ scheme (14-bits).

5.2 Performance analysis on medical images

We have performed an exhaustive simulation using several medical images, such as, X-ray, CT-scan, Ultrasound (US) and endoscopic images, and the results are presented in Table 2 (showing for two cases of bit precision: 8-bits and 12-bits). In all cases, the AIQ-based scheme produces a high PSNR and outperforms the conventional approach by a far margin. Some sample original and reconstructed images are shown in Fig. 8 – 11. In most cases, the difference in PSNR is around 8dB with noticeable level of improvement.

		D4-FP	D4-AIQ	D6-FP	D6-AIQ
CT	8 bit	44.4	52.5	41.1	49.3
	12 bit	66.7	76.6	65.5	76.6
Endoscopic	8 bit	44.8	51.8	41.4	50.8
	12 bit	67.1	76.0	66.1	75.9
US	8 bit	48.0	55.9	44.8	53.7
	12 bit	70.4	79.9	69.0	79.9
X-ray	8 bit	43.3	50.1	39.9	49.2
	12 bit	65.5	74.1	64.5	74.1

Table 2. Quality of reconstruction (in terms of PSNR in dB) for medical images.

(a) Original   (b) PSNR = 44dB   (c) PSNR= 52dB
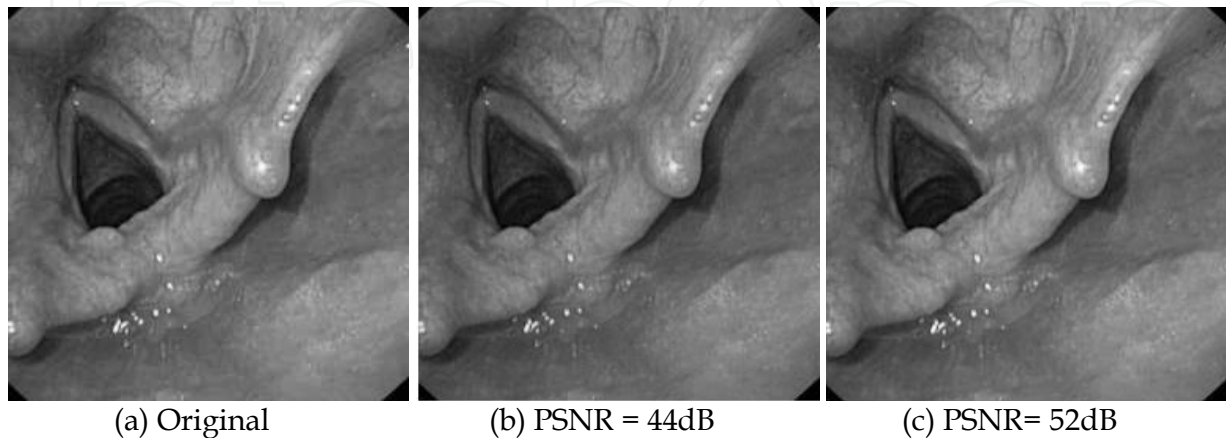
Fig. 8. (a) Original endoscopic image; (b) Reconstructed image using FP scheme; (c) AIQ scheme.

(a) Original   (b) PSNR = 43dB   (c) PSNR= 50dB
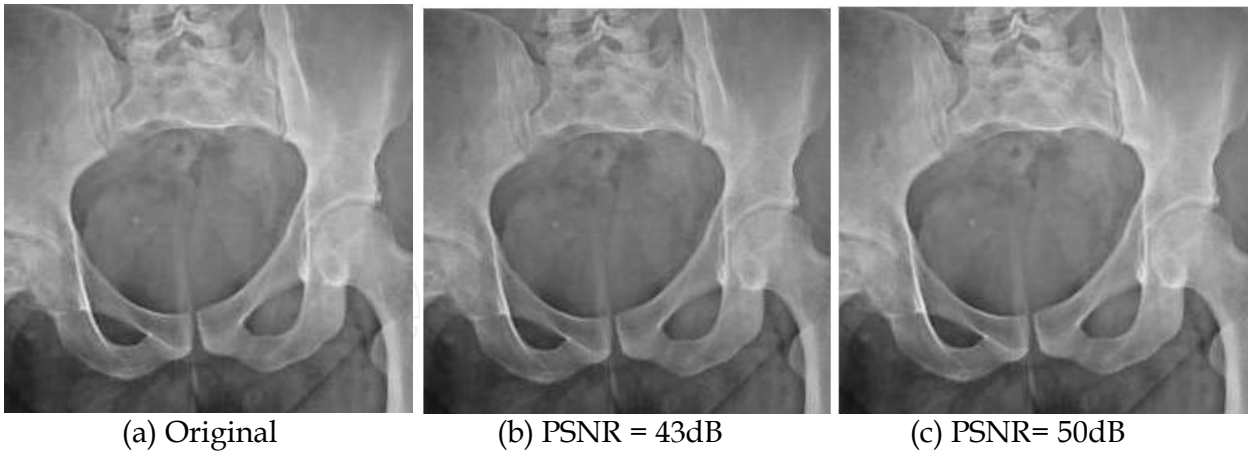
Fig. 9. (a) Original x-ray image; (b) Reconstructed image using FP scheme; (c) AIQ scheme.
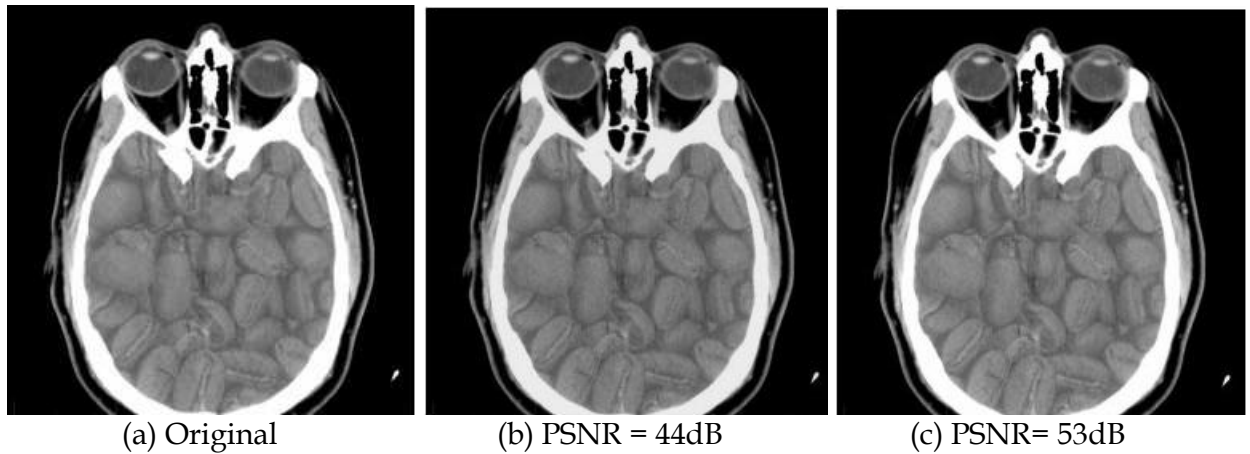

(a) Original   (b) PSNR = 44dB   (c) PSNR= 53dB

Fig. 10. (a) Original CT-scan image; (b) Reconstructed image using FP scheme; (c) AIQ scheme.

(a) Original   (b) PSNR = 48dB   (c) PSNR= 56dB
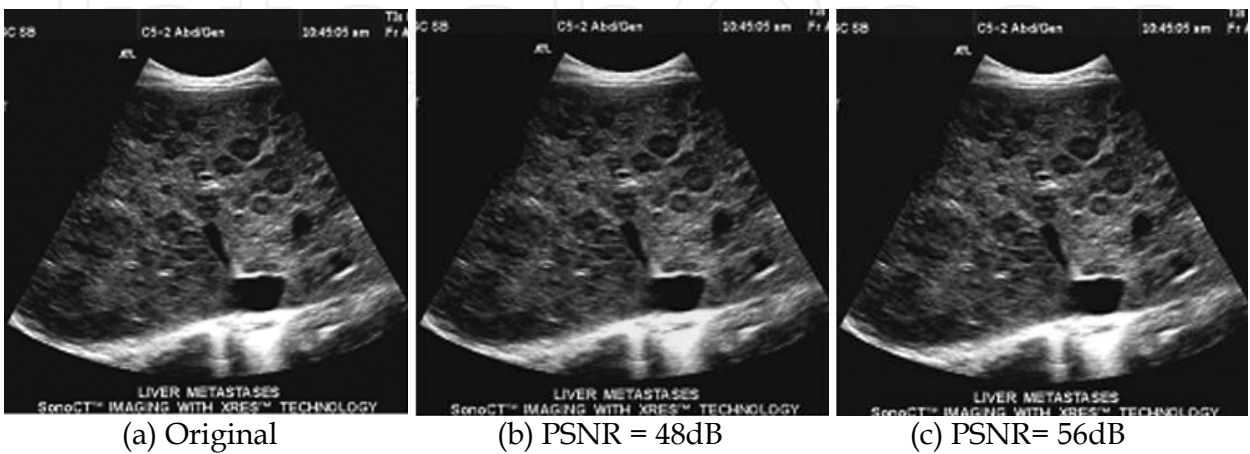
Fig. 11. (a) Original US image; (b) Reconstructed image using FP scheme; (c) AIQ scheme.

5.3 Performance analysis on noisy images

As a final study, the AIQ algorithm is tested under a noisy environment. The Gaussian white noise and Poisson noise are added to the images of all types, and the performance is compared with the FP implementation. The results are tabulated in Table 3 (all using 8-bit precision). Like previous cases, due to less error accumulation in the computation process, the AIQ-based approach is seen to have performed better than FP approaches even in a noisy environment.

Noise	Algorithm	PSNR (dB)				
		Goldhill	US	X-ray	Endoscopic	CT
Gaussian	D4 - FP	43.9	47.5	43.1	44.6	44.4
	D4 - AIQ	51.4	55.5	50.0	51.6	52.4
	D6 - FP	40.6	44.3	39.7	41.3	41.1
	D6 - AIQ	49.9	53.2	49.3	50.6	49.7
Poisson	D4 - FP	44.1	47.6	43.2	44.8	44.6
	D4 - AIQ	51.6	55.6	50.0	51.8	52.5
	D6 - FP	40.7	44.5	39.8	41.5	41.3
	D6 - AIQ	50.1	53.4	49.3	50.9	49.9

Table 3. Quality of reconstruction in noisy environment.

5.4 Comparative analysis

In order to show the effectiveness of the AIQ approach, the algorithm is compared with some compression standards: JPEG, JPEG2000, and existing algorithms targeted to medical imaging. Since there is no benchmark for medical images, we have conducted the experiment with benchmark images like "Lena", "Barbara" and "Goldhill". Table 4 summarizes the comparison results (all using 8-bit precision with 0.25 bits per pixel). From the table, it is clearly observed that the error-free algorithm performs competitively compared to other existing compression schemes.

Algorithm	Lena	Barbara	Goldhill
HS-HIC (Mohammed, 2008)	35.0	26.1	30.5
Hybrid (Yu & Mitra, 1997)	35.0	31.5	32.9
JPEG (Wahid et al., 2008)	32.4	27.7	29.7
JPEG2000 (Seo & Kim, 2007)	34.1	28.8	30.5
OB-HIC (Mohammed & Abd, 2010)	35.9	32.8	33.8
D4-AIQ	44.9	45.0	35.8
D6-AIQ	43.2	43.6	35.3

Table 4. Comparative analysis (in terms of PSNR in dB) of the AIQ scheme.

6. Conclusion

In this chapter, we have presented an efficient approach to compute Daubechies wavelet transforms that is based on encoding the basis set of forward transform coefficients using algebraic integers. The AIQ approach not only reduces the number of arithmetic operations,

but also reduces the dynamic range of the computations. Because of error-free mapping in the earlier stages, less error is introduced in the system, as compared to FP implementation, that results in much better data reconstruction. The performance is validated using standard and medical images in both normal and noisy conditions. In all cases, the AIQ-based approach outperforms the conventional FP scheme by far margin. The rate of data recovery is very high while preserving critical information that makes the scheme suitable for medical and telemetric imaging applications.

7. Acknowledgement

8. References

Acharyya, A., Maharatna, K., Al-Hashimi, B., Gunn, S. (2009), Memory reduction methodology for distributed arithmetic based DWT/IDWT exploiting data symmetry, IEEE Trans. on Circuits and Systems II, vol. 56, no. 4, pp. 285-289.

Aware Inc., (1991) Aware Wavelet Transform Processor WTP) Preliminary, Cambridge, MA.

Cozzens, J. and Finkelstein, L. (1985) Computing the Discrete Fourier Transform using Residue Number Systems in a Ring of Algebraic Integers, IEEE Transactions on Information Theory, vol. 31, pp. 580-588.

Daubechies, I., (1992) Ten lectures on wavelets, SIAM, 1992.

Huang, C., Tseng, P., and Chen, L., (2004) Flipping structure: an efficient VLSI architecture for lifting based discrete wavelet transform, IEEE Trans. Signal Process., vol. 52, no. 4, pp. 1080–1089.

Lai, Y., Chen, L., Shih, Y. (2009) A high-performance and memory-efficient VLSI architecture with parallel scanning method for 2-D lifting-based discrete wavelet transform, IEEE Trans. on Consumer Electronics, vol. 55, no. 2, pp. 400 – 407.

Lee, S. and Lim, S., (2006) VLSI design of a wavelet processing core, IEEE Trans. Cir. Syst. Video Tech., vol. 16, pp. 1350-1360.

Lewis, A. and Knowles, G. (1991) VLSI Architecture for 2D Daubechies Wavelet Transform without Multipliers, IEE Electronics Letters, vol. 27, no. 2, pp. 171-173.

Mallat, S., (1998) A wavelet tour of signal processing, New York: Academic.

Martina, M. and Masera, G., (2007) Multiplierless, folded 9/7-5/3 wavelet VLSI architecture," IEEE Trans. Cir. Syst. II, 54, pp. 770-774, 2007.

Mohammed, U. (2008), Highly scalable hybrid image coding scheme, Digital Signal Processing, Science Direct, vol. 18, pp. 364–374.

Mohammed, U. and Abd-elhafiez, W. (2010), Image coding scheme based on object extraction and hybrid transformation technique, Int. J. of Engineering Science and Technology, vol. 2, no. 5, pp. 1375–1383.

Paek, S. and Kim, L. (1998) 2D DWT VLSI Architecture for Wavelet Image Processing, IEE Electronics Letters, vol. 34, no. 5, pp. 537-538.

Parhi, K. and Nishitani, T. (1993) VLSI Architectures for Discrete Wavelet Transforms, IEEE Transactions on VLSI Systems, vol. 1, no. 2, pp. 191-202.

Seo, Y. and Kim, D. (2007) VLSI architecture of line-based lifting wavelet transform for motion JPEG2000, IEEE J. Solid-State Circuits, vol. 42, no. 2, pp. 431-440.

Shi, G., Liu, W., Zhang, L., Li, F. (2009) An efficient folded architecture for lifting-based discrete wavelet transform, IEEE Trans. on Circuits and Systems II, vol. 56, no. 4, pp. 290-294.

Vishwanath, M., Owens, R. and Irwin, M. (1995) VLSI Architectures for the Discrete Wavelet Transform, IEEE Transactions on Circuits and Systems - II, vol. 42, no. 5, pp. 305-316.

Wahid, K., Dimitrov, V., Jullien, G. (2004) VLSI architectures of daubechies wavelet transforms using algebraic integers, J. of Circuits, Sys., and Comp., vol. 13, no.6, pp. 1251-1270.

Wahid, K., Dimitrov, V., Jullien, G. and Badawy, W. (2003) Error-Free computation of daubechies wavelets for image compression applications, Elect. Lett., vol. 39, no. 5, pp. 428-429.

Wahid, K., Ko, SB. and Teng, D., (2008) Efficient hardware implementation of an image compressor for wireless capsule endoscopy applications, Proc. of the IEEE Int. Joint Conf. on Neural Net., pp. 2762-2766.

Yu, T., and Mitra, S. (1997), Wavelet based hybrid image coding scheme, Proc. IEEE Int Circuits and Systems Symp, vol. 1, pp. 377–380.

Discrete Wavelet Transforms - Algorithms and Applications

Edited by Prof. Hannu Olkkonen

The discrete wavelet transform (DWT) algorithms have a firm position in processing of signals in several areas of research and industry. As DWT provides both octave-scale frequency and spatial timing of the analyzed signal, it is constantly used to solve and treat more and more advanced problems. The present book: Discrete Wavelet Transforms: Algorithms and Applications reviews the recent progress in discrete wavelet transform algorithms and applications. The book covers a wide range of methods (e.g. lifting, shift invariance, multi-scale analysis) for constructing DWTs. The book chapters are organized into four major parts. Part I describes the progress in hardware implementations of the DWT algorithms. Applications include multitone modulation for ADSL and equalization techniques, a scalable architecture for FPGA-implementation, lifting based algorithm for VLSI implementation, comparison between DWT and FFT based OFDM and modified SPIHT codec. Part II addresses image processing algorithms such as multiresolution approach for edge detection, low bit rate image compression, low complexity implementation of CQF wavelets and compression of multi-component images. Part III focuses watermaking DWT algorithms. Finally, Part IV describes shift invariant DWTs, DC lossless property, DWT based analysis and estimation of colored noise and an application of the wavelet Galerkin method. The chapters of the present book consist of both tutorial and highly advanced material. Therefore, the book is intended to be a reference text for graduate students and researchers to obtain state-of-the-art knowledge on specific applications.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Khan Wahid (2011). Low Complexity Implementation of Daubechies Wavelets for Medical Imaging Applications, Discrete Wavelet Transforms - Algorithms and Applications, Prof. Hannu Olkkonen (Ed.), ISBN: 978-953-307-482-5, InTech, Available from: http://www.intechopen.com/books/discrete-wavelet-transforms-algorithms-and-applications/low-complexity-implementation-of-daubechies-wavelets-for-medical-imaging-applications

INTECH
open science | open minds