

---

DE-NOISING OF ECG SIGNAL AND MRI IMAGE  
USING  
WAVELET TRANSFORM

SUBMITTED IN THE PARTIAL FULFILLMENT FOR THE DEGREE OF

**Master of Engineering**

(CONTROL AND INSTRUMENTATION)  
(2004-2006)

Submitted By

**Devanjali Agnihotri**  
University Roll No.: - 8672

Guided By

**Dr. PARMOD KUMAR**  
PROF. & HEAD ELECTRICAL ENGG. DEPTT



DEPARTMENT OF ELECTRICAL ENGINEERING  
DELHI COLLEGE OF ENGINEERING  
UNIVERSITY OF DELHI

---

---

## **CERTIFICATE**

It is certified that **Ms. DEVANJALI AGNIHOTRI**, Roll No. 8672, student of ME, Control and Instrumentation, Department of Electrical Engineering, Delhi College of Engineering, has submitted the dissertation entitled “**DE-NOISING OF ECG SIGNAL AND MRI IMAGE USING WAVELET TRANSFORM**”, under my guidance towards partial fulfilment of the requirements for the award of the degree of Master of Engineering (Control & Instrumentation Engineering).

This dissertation is a bonafide record of project work carried out by her under my guidance and supervision. Her work is found to be excellent and her discipline impeccable during the course of the project.

I wish her success in all her endeavors.

**(Dr. Parmod Kumar)**  
Professor & Head  
Deptt. of Electrical Engineering  
Delhi College of Engineering

---

---

## **ACKNOWLEDGEMENT**

I would like to extend my gratitude and thanks to my guide **Dr. PARMOD KUMAR, Prof. & H.O.D Electrical Engineering Department, Delhi College of Engineering, Delhi** for his valuable guidance, constant encouragement and helpful discussions throughout the course of this work. Obviously the progress I had now will be uncertain without his assistance.

I am very thankful to all my friends for listening to my problems and extending their helping hand during the hour of need. I would like to pay my vote of thank to my friend **Ms. Beena Antony** for providing immense help, advice and cooperation to complete the work.

At last but not the least I am grateful to my parents, brother and sister for their encouragement, continuous support and dedication which have helped me in a great way to complete the work.

**(Devanjali Agnihotri)**

---

---

## TABLE OF CONTENTS

<b>Abstract</b>	VI
<b>List of figures</b>	VII
<b>List of tables</b>	XII
<b>List of the used abbreviations</b>	XIII
	<b>Page No.</b>
<b>Chapter I Introduction</b>	
1.1 Introduction	1
1.2 Objective	2
1.3 Problem Identification	3
1.4 Features of Fourier Analysis	3
1.5 Features of Wavelet Analysis	6
1.6 Dissection of thesis	9
<b>Chapter II Literature Review</b>	
2.1 Introduction	10
2.2 History Review	10
2.3 Wavelets in the area of medical diagnosis	13
2.4 Conclusion	14
<b>Chapter III Mathematical Model of Wavelet Transform</b>	
3.1 Introduction	15
3.2 The Continuous wavelet transform (CWT)	16
3.3 The Discrete wavelet transform (DWT)	24
3.3.1 One-Stage Filtering: Approximations and Details	27
3.3.2 High & Low Pass Wavelet Filters	29
3.3.3 Multiresolution	30
3.3.4 Multiple-Level Decomposition	31
3.3.5 Wavelet Reconstruction	34
3.4 Wavelet packet decomposition	41

---

---

3.5	Discrete Stationary Wavelet Transform (SWT)	44
	3.5.1 e-Decimated DWT	46
3.6	Conclusion	48

## **Chapter IV Wavelet in Bio-Signal Processing**

4.1	Introduction	49
4.2	Biopotentials in the Heart	49
4.3	Brain Structures and their Functions	52
4.4	Peak Signal-to-Noise ratio and Signal-to-Noise ratio	55
4.5	Noise artifacts in ECGs and MRI Scan/Images	57
4.6	Wavelet Shrinkage and Thresholding operators for de-noising	58
4.7	Selection of Thresholding Method	60
	4.7.1 Global Threshold	61
	4.7.2 Level-Dependent Threshold	61
	4.7.3 Spatial Adaptive Threshold	61
4.8	Conclusion	62

## **Chapter V Algorithm to work with Signal and Images in MATLAB 7.0**

5.1	Working with Signals	
	5.1.1 Working in MATLAB Command line	63
	5.1.2 Working in GUI toolbox	64
5.2	Working with Images	
	5.2.1 Working in MATLAB Command line	70
	5.2.2 Working in GUI toolbox	70
5.3	Conclusion	75

## **Chapter VI Results and Discussion**

## **Chapter VII Conclusion and Future scope**

95

---

---

**References** 97

**Appendix-Source Code** 103

---

---

## ABSTRACT

Most of the signals in practice are time-domain signals in their raw format. Time-amplitude representation is not always the best representation of the signal for most signal processing related applications. In many cases, the most distinguished information is hidden in the frequency content of the signal. To measure frequency, or to find the frequency content of a signal Fourier Transform (FT) is used. Fourier analysis has a serious drawback that in transforming to the frequency domain, time information is lost. This drawback isn't very important if signal is a *stationary* signal. However, most interesting signals are non-stationary (ECG, EMG, EEG etc) and thus Fourier analysis is not suited to detecting transitory characteristics. The traditional Fourier transform only provides the spectral information of a signal and thus it is not suitable for the analysis of non-stationary signals and hence a more suitable technique i.e. the Wavelet Technique has been applied for the study.

The main goal of the thesis is to develop the de-noising algorithms based upon the discrete wavelet transform (DWT) and Stationary wavelet transform (SWT) that can be applied successfully to enhance noisy MRI and ECG signal.

The thesis present 2-D image decomposition, thresholding and reconstruction using the DWT and SWT as a new approach which can be used in the processing of biomedical images. The thesis also presents the theory of the fundamental mathematical tools (Fourier transform and Wavelet Transform) that are used for the analysis of a signal. Wavelet Transform plays an increasingly important role in the de-noising of MR images. The performance of the de-noising algorithms are quantitatively assessed using different criteria viz. peak signal to-noise ratio (PSNR) (for MR Image), signal to-noise ratio (SNR)(for ECG signal) and the visual appearance. All the work has been done in MATLAB and the results are discussed in accordance to the noise variance and wavelets implemented. The properties of wavelets make them special in that they have a good time and frequency localization which make them ideal for the processing of non-stationary signals like the biomedical signals ( ECG,..) and images (MR).

---

---

## LIST OF FIGURES

Figure No.	Title	Page No.
1.1	Fourier Transform	4
1.2	Short-Time Fourier Transform	5
1.3	Wavelet Transform	7
1.4	Example of spatial-frequency tiling of various transformations. x-axis: spatial resolution. Y-axis: frequency resolution. <b>(a)</b> discrete sampling (no frequency localization), <b>(b)</b> Fourier transform (no temporal localization), <b>(c)</b> windowed Fourier transform (constant Heisenberg boxes), <b>(d)</b> wavelet transform (variable Heisenberg boxes).	7
1.5	Time-Frequency plane of Wavelet transform	8
2.1	History Tree of Wavelet theory	11
3.1	Fourier Transform of a signal	16
3.2	Wavelet Transform of a signal	17
3.3	Scaling of a signal	17
3.4	Correspondence between scale and frequency	18
3.5	Shifting of a signal	18
3.6	<b>(a)</b> Calculating wavelet coefficients using CWT	19
	<b>(b)</b> Calculating wavelet coefficients with shifted version of wavelet using CWT	19
	<b>(c)</b> Calculating wavelet coefficients with higher scale of wavelet using CWT	20
3.7	Coefficient plot generated using CWT	20
3.8	Front view of Coefficient plot generated using CWT	21
3.9	Single level filter analysis	25
3.10	3 level filter bank	26
3.11	4 output scales produced from signal of 16 samples, in frequency range of 0 to $f_n$	27

---



---

3.12	Filtering process	28
3.13	Notion of <i>Downsampling</i> <b>(a)</b> without downsampling, <b>(b)</b> with downsampling	28
	<b>(c)</b> Notion of Downsampling of sinusoid with high- frequency noise	29
3.14	Multiresolution of a Signal. An arbitrary signal can be decomposed into approximation and detail components.	31
3.15	Wavelet Decomposition Tree	32
3.16	Wavelet Decomposition Tree of a signal showing valuable information	32
3.17	Hierarchical model of wavelet Decomposition	33
3.18	Wavelet Decomposition Example. <i>HP</i> is the high pass wavelet filter, and <i>LP</i> is the low pass scaling filter. <i>B</i> , <i>2B</i> , and <i>4B</i> are the coefficients at each level. At each level the low pass approximation information is divided into another level of approximation and detail information by applying the low pass scaling filter and the high pass wavelet filter .	34
3.19	Reconstruction Process	34
3.20	Upsampling	35
3.21	Notion of Decomposition and Reconstruction	35
3.22	<b>(a)</b> Constructing approximation and detail coefficients	35
	<b>(b)</b> Reconstruction of approximation $A_1$	36
	<b>(c)</b> Reconstruction of Detail $D_1$	36
3.23	Reconstructing signal using Multilevel analysis	37
3.24	Multistep Decomposition and Reconstruction	37
3.25	Wavelet Packet decomposition over 3 levels	42
3.26	Wavelet Packet decomposition Tree	42
3.27	<b>(a)</b> Dyadic wavelet decomposition tree. <b>(b)</b> wavelet packets decomposition tree. <b>(c)</b> An example of an orthogonal basis tree with wavelet packets decomposition.	43
3.28	3 level SWT filter bank	44
3.29	SWT filters	44

---

---

3.30	Convolving the signal with the appropriate filters without downsampling (SWT).	47
3.31	1D SWT	47
4.1	Heart	50
4.2	ECG Leads	51
4.3	ECG Signal	51
4.4	Lobes of the Cerebral Cortex	53
4.5	<a href="#">Limbic System</a>	54
4.6	Representation to explain Signal to Noise Ratio	56
4.7	Example of thresholding functions, assuming that the input data was normalized to the range of [-1,1]. <b>(a)</b> Hard thresholding. <b>(b)</b> Soft thresholding. <b>(c)</b> Affine thresholding. The threshold level was set to $T=0.5$	60
5.1	Level 3 Decomposition of signal	64
5.2	Wavelet Toolbox Main Menu	65
5.3	1D Wavelet Analysis Toolbox	65
5.4	Loading Signal in 1D Wavelet Analysis Toolbox	66
5.5	Selecting wavelet type and level of decomposition in 1D wavelet Analysis Toolbox	66
5.6	Window showing original signal, approximation (a1) and detail (d1) of level 1 Decomposition	67
5.7	Zooming toolbar	67
5.8	Zoomed version of original signal, approximation (a1) and detail (d1) of level 1	68
5.9	Toolbar to select appropriate action	68
5.10	Window showing Original signal and details of level 3 decomposition	69
5.11	Window showing Original signal, De-noise signal and details of level 3	69
5.12	2D wavelet analysis tool	71
5.13	Loading Image in 2D wavelet analysis tool	71
5.14	Selecting wavelet type and level of Decomposition in 2D wavelet Analysis Toolbox	72

---

---

5.15	Window showing Original Image, Synthesise Image, and Decomposition (Approximation and Detail) at level 2	72
5.16	Zooming the region of Interest in Original Image	73
5.17	Zoom Toolbar	73
5.18	Zoomed version of original and synthesized image	74
5.19	Window showing Original and De-noised Image, with level 1 and 2 Horizontal, vertical and diagonal detail	74
6.1	Original ECG Signal	76
6.2	Approximation and Detail after single-level decomposition using DWT	77
6.3	Approximation A1 after level 1 decomposition using DWT	78
6.4	Details (D1, D2 & D3) after level 3 decomposition using DWT	79
6.5	Original and level 3 approximation using DWT	79
6.6	Original and De-noised ECG Signal using DWT	80
6.7	Overlap Original and De-noised signal (using DWT)	81
6.8	Approximation and Detail after single-level decomposition using SWT	81
6.9	Approximations (A1, A2 & A3) and Details (D1, D2 & D3) after level 3 decomposition using SWT	82
6.10	Original and De-noised ECG Signal using SWT	82
6.11	Overlap Original and De-noised signal (using SWT)	83
6.12	Original and De-noised Signal using sqtwolog threshold	84
6.13	Original and De-noised Signal using minimaxi threshold	84
6.14	Original and De-noised Signal using heursure threshold	85
6.15	Original RGB MR image	86
6.16	Gray scale MR image	86
6.17	Approximation and details after single-level decomposition using DWT	87
6.18	Approximation and details after level 2 decomposition using DWT.	87
6.19	Noisy and De-noised MR image using DWT	88
6.20	Original and Compressed image	88

---

---

6.21	Approximation and details after single-level decomposition using SWT	89
6.22	Approximation (A1 & A2) and details (D1 & D2) after level 2 decomposition using SWT	89
6.23	Noisy and De-noised MR image using SWT	90
6.24	Selected region of interest using mouse	91
6.25	Noisy and De-Noise MR image of selected region shown in figure 6.24.	91
6.26	Selected region of interest by specifying an array	92
6.27	Noisy and De-Noise MR image of selected region shown in figure 6.26.	92
6.28	A sequence of MR image slices	93
6.29	Slice No 6 from sequence of MR image slices	93
6.30	A sequence of Tumour cycle	94

---

---

## LIST OF TABLES

<b>Table No.</b>		<b>PAGE No.</b>
6.1	SNR for ECG signal with DWT and SWT and with different thresholding method: Sqtwolog, minimaxi, heursure using DWT	85
6.2	PSNR for the MR Image with different noise variance $\sigma$ using db1 and db2 wavelet	90

---

## LIST OF THE USED ABBREVIATIONS

1-D	One-dimensional
2-D	Two-dimensional
CWT	Continuous Wavelet Transform
FT	Fourier Transform
FFT	Fast Fourier Transform
DSP	Digital Signal Processing
DWT	Discrete Wavelet Transform
IDWT	Inverse Discrete Wavelet Transform
SWT	Stationary Wavelet Transform
WT	Wavelet Transform
STFT	Short Time Fourier Transform
GUI	Graphical User Interface
PSNR	Peak Signal to Noise Ratio
SNR	Signal to Noise Ratio
MRI	Magnetic Resonance Imaging
MR.	Magnetic Resonance

---

---

# ***Chapter 1***

---

---

# CHAPTER I

## INTRODUCTION

### 1.1 Introduction

Digital signal processing (DSP) describes the science that tries to analyze, generate and manipulate measured real world signals with the help of a digital computer. These signals can be a collection of numbers, or measurements and the most commonly used signals include images, audio (such as digitally recorded speech and music) and medical and seismic data. In most digital signal processing applications, the frequency content of the signal is very important. The Fourier transform (FT) is probably the most popular transform used to obtain the frequency spectrum of a signal. Noise removal or de-noising is an important task in image processing. In general, the results of the noise removal have a strong influence on the quality of the image processing techniques.

Magnetic resonance imaging (MRI) shows the abnormalities of the brain such as: stroke, hemorrhage, tumor, multiple sclerosis or lesions. Noise in MR images consists of random signals that do not come from the tissues but from other sources in the machine and environment that do not contribute to the tissue differentiation. The noise of an image gives it a grainy appearance and mainly the noise is evenly spread and more uniform. Image processing can be defined as the manipulation of an image for the purpose of either extracting information from the image or producing an alternative representation of the image.

Electrocardiograms is a biomedical signals that originate from the action of the human heart. The ECG is the graphical representation of the potential difference between two points on body surface, versus time. ECG signals are largely employed as a diagnostic tool in clinical practice in order to assess the cardiac status of the object. They are used to examine ambulatory patients who are at rest during a recording or performing an exercise program and also patients in intensive care. ECG recording are examined by a physician who visually checks features of the signal and estimates the most

---



---

important parameters of the signal to judge the status of a patient. The recognition and analyzing of ECG signal is difficult, since their size and form may change eventually and there can be a considerably amount of noise in the signal. Since the processing of ECG signal is very important step in the process of ECG examination by physicians, many tools, method and algorithm proceeding from signal processing theory have been proposed, described and implemented. Wavelet Transform (WT) is a new and promising set of tools and techniques for doing this.

Presently ECG signal is processed using a Fast Fourier Transform (FFT) and the most of the commercial requirements are met by this procedure during normal heart health monitoring. However the information provided by the FFT, corresponds to all time instances, since the integration is from minus infinity to plus infinity over time. This is why Fourier transform is not suitable if signal has time varying frequency, i.e., the signal is non-stationary. This means FFT tells whether a certain frequency components exists or not. This information is independent of where in time this components appears.

Thus in this project work selection of Wavelet Transform method for filtering the noise from ECG and MR image is adopted since this method takes into account different resolution at different frequencies and space-time solution simultaneously.

## 1.2 Objective

The goal of the present work is to de-noise the bio-signal using wavelet transform, i.e. ECG signal and MRI, that improve the quality of the bio-signal so that the details are available clearly for diagnosis purpose. Noise removal or de-noising is an important task in image processing used to recover a signal that has been corrupted by noise. Wavelet transform basically process the signal by carrying out four operations **viz:** **(i) Decomposition**, **(ii) Threshold**, **(iii) Reconstruction**, and **(iv) De-noising**. Noise reduction using the wavelet transform has the potential to outperform existing methods for medical images, as it exploits the wavelet's natural ability to separate signal from noise at multiple image scales.

---

---

### **1.3 Problem Identification**

A major class of problems in medical science involves the diagnosis of disease, based upon various tests performed upon the patient. When several tests are conducted, the ultimate diagnosis may be difficult to conclude, even for a medical expert. The primary diagnosis start with merely seeing and analysing the electrical activity of heart or analysing MR images.

Noise present in the ECG signal might effect in the detection of various characteristics of the signal such as the R-peak, QRS complexes, T-waves etc. The various noise that are considered are electromyography interference, 50Hz power line interference, base line drift due to respiration, abrupt base line shifts, and a composite noise constructed from all of the other noise types.

Image de-noising is removing noise without sacrificing important structures. Since this is not possible with linear techniques many nonlinear strategies have been proposed in the last two decades. One of these classes is wavelet methods [1,2,3]. The wavelet transform itself offers great design flexibility. Instead of trying to replace standard image processing techniques, wavelet transforms offer an efficient representation of the signal, finely tuned to its intrinsic properties.

Wavelet transforms are presented to eliminate the noise from MR image and ECG signal their performance is discussed with reference to PSNR and SNR respectively.

### **1.4 Features of Fourier Analysis**

#### **Fourier analysis**

Fourier analysis is a mathematical technique for transforming the signal from time based to frequency-based or in other words which breaks down a signal into constituent sinusoids of different frequencies. For many signals, Fourier analysis is extremely useful because the signal's frequency content is of great importance. But Fourier analysis has a serious drawback that on transforming the signal into frequency domain, time information is lost.

---

---

Historically, the Fourier transform has dominated linear time-invariant signal processing. The associated basis functions are complex sinusoidal waves  $\exp(i\omega t)$  that correspond to the eigenvectors of a linear time-invariant operator. A signal  $f(t)$  defined in the temporal domain and its Fourier transform  $\hat{f}(\omega)$ , defined in the frequency domain, have the following relationships.

$$\hat{f}(\omega) = \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt \quad (1.1)$$

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{f}(\omega) e^{i\omega t} d\omega \quad (1.2)$$

Fourier transform characterizes a signal  $f(t)$  via its frequency components. Since the support of the bases function  $e^{i\omega t}$  covers the whole temporal domain (i.e infinite support),  $\hat{f}(\omega)$  depends on the values of  $f(t)$  for all times. This makes the Fourier transform a global transform that cannot analyze local or transient properties of the original signal  $f(t)$ .



**Figure 1.1      Fourier Transform**

FT (as well as WT) is a reversible transform, that is, it allows to go back and forward between the raw and processed (transformed) signals. However, only either of them is available at any given time i.e. no frequency information is available in the time-domain signal, and no time information is available in the Fourier transformed signal.

---

---

## Short-Time Fourier Analysis

Fourier transform of a signal, does not tell when a particular event took place. If the signal is *stationary* signal i.e. signal properties do not change much over time this drawback isn't very important. However, most interesting signals contain numerous non-stationary or transitory characteristics like drift, trends, abrupt changes, and beginnings and ends of events. These characteristics are often the most important part of the signal, and Fourier analysis is not suited to detecting them.

In an effort to correct this deficiency, Dennis Gabor (1946) adapted the Fourier transform to analyze only a small section of the signal at a time — a technique called *windowing* the signal called the *Short-Time Fourier Transform* (STFT), maps a signal into a two-dimensional function of time and frequency.



**Figure 1.2 Short-Time Fourier Transform**

The STFT represents a sort of compromise between the time- and frequency-based views of a signal. It provides some information about both when and at what frequencies a signal event occurs. However, we can only obtain this information with limited precision, and that precision is determined by the size of the window. Another drawback of STFT is that once we choose a particular size for the time window, that window is the same for all frequencies.

There is only a minor difference between STFT and FT. In STFT, the signal is divided into small enough segments, where these segments (portions) of the signal can be assumed to be stationary. For this purpose, a window function "**w**" is chosen. The width of this window must be equal to the segment of the signal where its stationarity is valid. This window function is first located to the very beginning of the signal. The window function and the signal are then multiplied. Then this product is assumed to be just another signal, whose FT is to be taken. The next step, would be

---

---

shifting this window (for some  $t_1$  seconds) to a new location, multiplying with the signal, and taking the FT of the product. This procedure is followed, until the end of the signal is reached by shifting the window with " $t_1$ " seconds intervals.

If signal obtained under particular size of window is stationary then only the result obtained will be true frequency representation. The following definition of the STFT summarizes all the above explanations in one line:

$$STFT_X^{(\omega)}(t, f) = \int_t [x(t) \bullet w^*(t - t')] \bullet e^{-j2\pi ft} dt \quad (1.3)$$

where  $\mathbf{x}(t)$  is the signal itself,  $\mathbf{w}(t)$  is the window function, and  $*$  is the complex conjugate. We can see from the equation, the STFT of the signal is nothing but the FT of the signal multiplied by a window function.

## 1.5 Features of Wavelet Analysis

The WT was developed as an alternative to the STFT to overcome some resolution related problems of the STFT. We basically need Wavelet Transform (WT) to analyze non-stationary signals, i.e., whose frequency response varies in time. Wavelet transforms produce a time–frequency decomposition of the signal which separates individual signal components more effectively than the traditional short time Fourier transform (STFT). This flexible temporal–spectral aspect of the transform allows a local scale-dependent spectral analysis of individual signal features. In this way both short duration, high frequency and longer duration, lower frequency information can be captured simultaneously. Hence the method is particularly useful for the analysis of transients, aperiodicity and other non-stationary signal. Another key advantage of wavelet techniques is the variety of wavelet functions available, thus allowing the most appropriate to be chosen for the signal under investigation. This is in contrast to Fourier analysis which is restricted to one feature morphology: the sinusoid. In its discrete form using orthogonal wavelet bases, the wavelet transform is particularly useful in signal coding, allowing information within the signal to be localized within a number of pertinent coefficients for compression purposes.

---

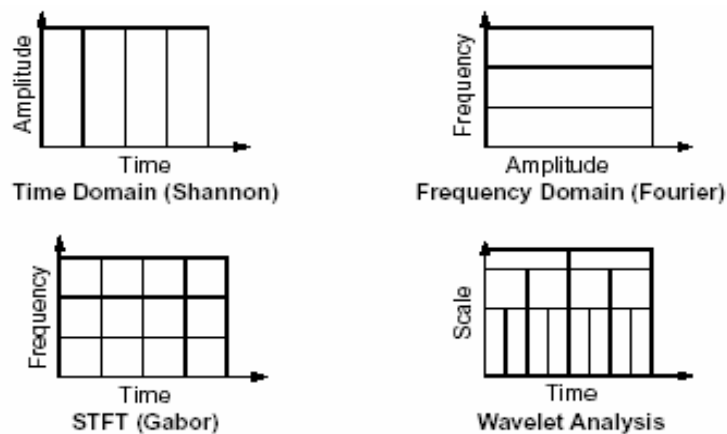
Wavelet analysis represents the next logical step: a windowing technique with variable-sized regions. Wavelet analysis allows the use of long time intervals where we want more precise low-frequency information, and shorter regions where we want high-frequency information.



**Figure 1.3** Wavelet Transform

**Fig: 1.4**

a	b
c	d



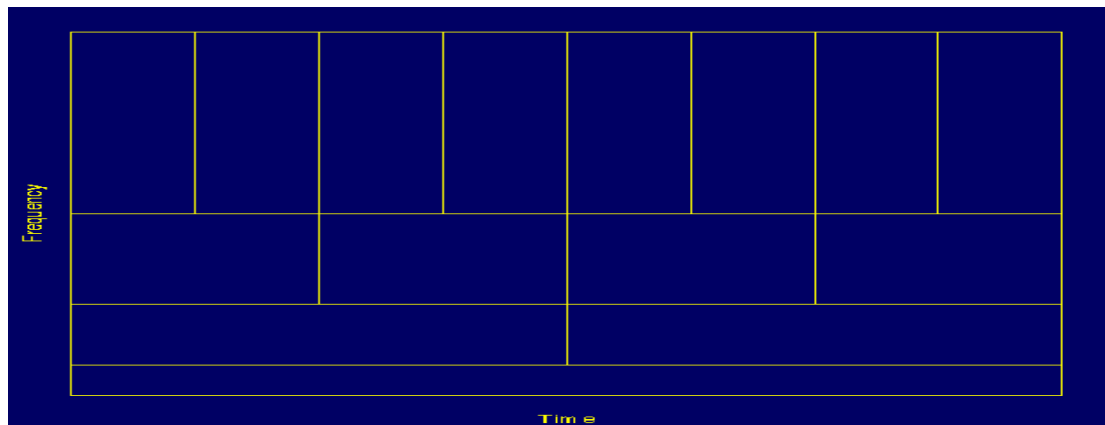
**Figure 1.4: Example of spatial-frequency tiling of various transformations. x-axis: spatial resolution. Y-axis: frequency resolution. (a) discrete sampling (no frequency localization) (b) Fourier transform (no temporal localization). (c) windowed Fourier transform (constant Heisenberg boxes). (d) wavelet transform (variable Heisenberg boxes).**

To analyze transient signal structures of various supports and amplitudes in time, it is necessary to use time-frequency atoms with different support sizes for different temporal locations ( figure 1.4(d) ). For example, in the case of high frequency structures, which vary rapidly in time, we need higher temporal resolution to accurately trace the trajectory of the changes; on the other hand, for lower frequency,

---

we will need a relatively higher absolute frequency resolution to give a better measurement on the value of frequency.

The illustration in Figure 1.5 is commonly used to explain how time and frequency resolutions should be interpreted. Every box in Figure 1.5 corresponds to a value of the wavelet transform in the time-frequency plane. All the points in the time-frequency plane that falls into a box is represented by one value of the WT.



**Figure 1.5 Time-Frequency plane of Wavelet transform**

In Figure 1.5 although the widths and heights of the boxes change, the area is constant. That is each box represents an equal portion of the time-frequency plane, but giving different proportions to time and frequency. At low frequencies, the height of the boxes are shorter (which corresponds to better frequency resolutions, since there is less ambiguity regarding the value of the exact frequency), but their widths are longer (which correspond to poor time resolution, since there is more ambiguity regarding the value of the exact time) and at higher frequencies the width of the boxes decreases, i.e., the time resolution gets better, and the heights of the boxes increase, i.e., the frequency resolution gets poorer.

Thus it is worthwhile to mention that in STFT the time and frequency resolutions are determined by the width of the analysis window, which is selected once for the entire analysis, i.e., both time and frequency resolutions are constant. Therefore the time-frequency plane consists of squares in the STFT case.

---

---

## 1.6 Dissection of Thesis

The material of this dissertation has been arranged in six chapters, references. The contents of the chapter are briefly outlined as indicated below:

Chapter 1 discusses objectives, identify the problem and introduces the features of fourier transform and wavelet transform and their advantages and disadvantages.

Chapter 2 presents the history review from fourier transform to wavelet theory and brief survey of the wavelet applications in biomedical field.

Chapter 3 describes the mathematical model developed for different types of Wavelet transform techniques.

Chapter 4 presents the brief anatomy of brain and electrical activity associated with the heart (ECG) and the cause of noise in the MRI image and ECG signal. It also covers the need of thresholding operator for noise reduction.

Chapter 5 explains the algorithm so as to de-noise the signal and image with MATLAB 7.0 and also explain the ease with which we can use the GUI of Wavelet transform for de-nosing.

Chapter 6 discusses the results obtained on running the MATLAB code.

Chapter 7 concludes the thesis and provides suggestions for future work.

---



---

# ***Chapter II***

---

---

## CHAPTER II

### LITERATURE REVIEW

#### 2.1 Introduction

The concept of wavelet has been discussed in the literature for a very long time however, it is only recently that significant progress has been made in the application of wavelets to practical problems in signal processing. The wavelet transform has been proposed as a flexible tool for the multiresolution decomposition of continuous time signals. The pioneering work of Daubechies in the early 1980s has shown the linkage between the wavelet and subbands transform theories. Since then, there has been an explosion of interest and a flurry of interdisciplinary research and development activities on wavelet and subbands transforms, and their applications [4][Dau90].

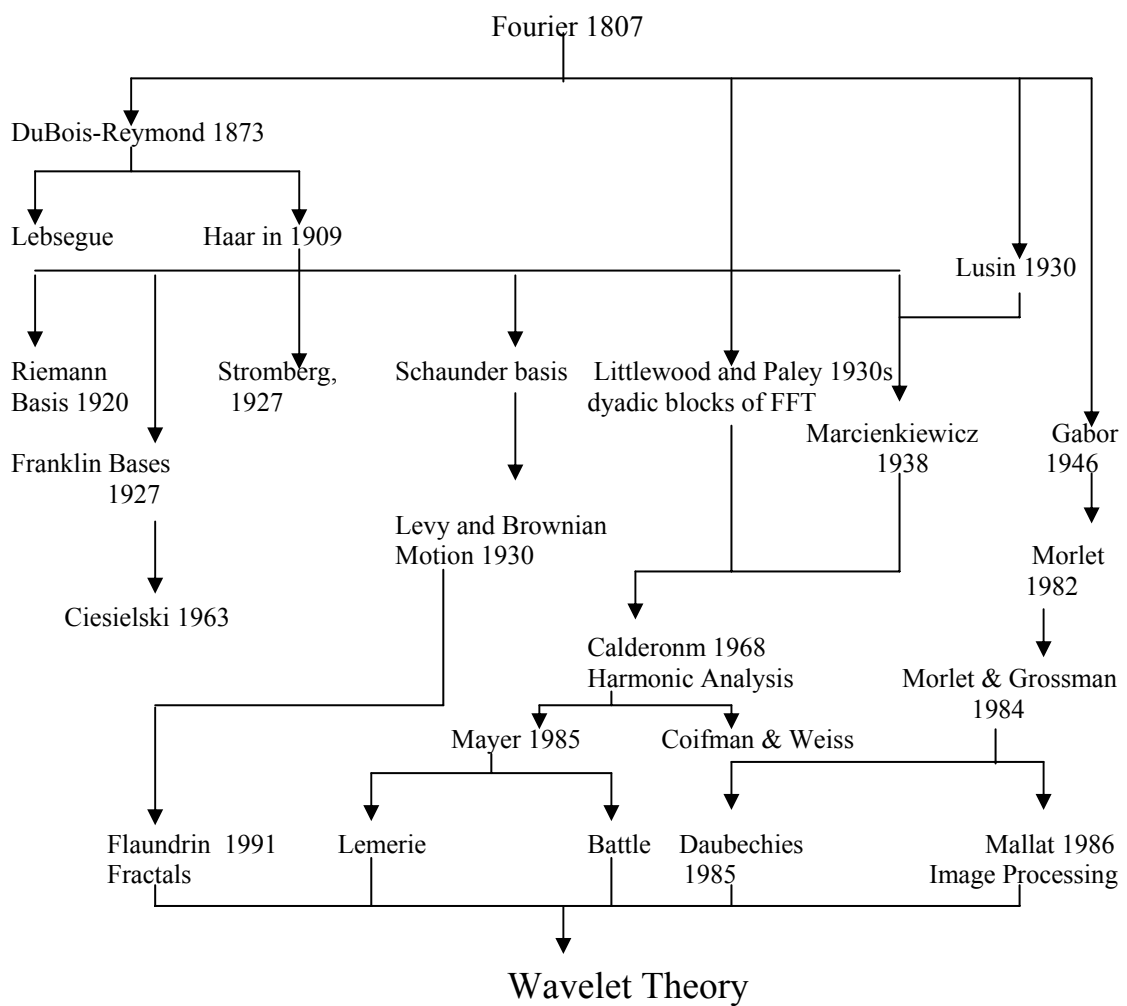
Significant practical application of wavelet have been found in signal and image processing, speed-spectrum and wireless communication, and control analysis, which are in the electrical engineering domain [5][6][Wor96], [Sch96a].

#### 2.2 History Review

The history of wavelet begins with the development of the traditional Fourier series in 1807, as shown in Figure 2.1. The techniques of the Fourier series and Fourier transform were pioneered by the French Physicist Jean Baptiste Joseph, Baron de Fourier (1768-1830). The Fourier transform is very useful in many application , but it had to be modified to deal with the case of singularities or sharp transient signals. This shortcoming of the Fourier Transform was originally identified by Paul DuBois-Reymond in 1873. He also proposed a solution to the singularity problem that eventually was fully developed by Henri Lebesgue. A different solution was designed by Haar in 1909; he replaced the sine and cosine function of the Fourier Transform with another orthonormal basis, now commonly known as the Haar basis. The original idea of the Haar has opened the door to the construction of an infinite number of other bases. Using a very simple function, Haar created a basis using dyadic scaling. The

---

function of the Haar basis suffers from a major disadvantage for many application; as they are discontinuous, they are not optimal for approximation a continuous function  $f(t)$ , especially if the function has a continuous derivative. Nevertheless, the importance of the Haar decomposition lays in the development of an orthonormal basis based on dyadic sampling which opened one of the routes leading to the concept of wavelet and, in particular to what it is now called Multiresolution analysis. The Haar basis is the simplest example to date of a wavelet basis which satisfies the multiresolution properties [7][Mey93].



**Figure 2.1 History Tree of Wavelet theory**

The short comings of the Haar decomposition led to the development of other bases such as Schauder basis and the Riemann basis in the 1910-1920 decades. In 1930s, a lot of interest was given to the study of Brownian motion. The Fourier analysis was adequate for representing the spectral properties of the Brownian motion, especially to

---

extract any particular resonance frequency. Unfortunately, the randomness of the motion itself was not well defined by the Fourier decomposition. It was then Paul Levy decided to apply the Schunder basis to describe the local regularity properties of the Brownian motion. This work has been recently (1991) extended by Patrick Flandrin for the analysis of fractal Brownian motion, following the Noise models of Mandelbrot and Van Ness.

Also in the 1930s, Littlewood and Paley established a new approach to group the Fourier Transform coefficient of a signal in order to extract rapidly and more efficiently any information about singularities in the signal itself. The regrouping proposed by Littlewood and Paley is based on the concept dyadic blocks, which can be used to create an orthogonal basis similar to what was later defined as a wavelet basis. The major difference between the Haar basis and the Littlewood – Paley basis is that whereas in former, the scaling is performed in the time basis, in the later the scaling is set on the Fourier transform coefficients. For this reason, Littlewood – Paley basis is considered the dual basis of the Haar function

While all this development were taking place in the field of Mathematics, Lusin, a physicist was involved in the analysis and synthesis of the function using “atom” or “basis element.” Any signal in a given space can be constructed by the combination of these “atom.” Lusin effectively initiated the field of “ Harmonic Analysis,” even though this term was coined much later by G. Weiss and R. R. Coifmann in the 1980s. In 1938, Marcinkiewicz showed that the Haar function are undoubtedly the simplest atomic decomposition. The theory of the harmonic analysis was extended by Calderon in 1964, in what is known as the Calderon’s Identity. Grossman and Morlet (a quantum physicist and an engineer) rediscovered this identity in 1980, 20 years after the original work of Calderon. However, they gave it a different interpretation, by relating it to the “coherent states” of the quantum mechanics.

Morlet, a geophysical engineer come up with the idea of wavelet as an alternative to Short-Time Fourier Transform. Morlet is the geophysicist who coined the French word “ ondelettes,” later translated into English word “wavelet”. Morlet’s work was put on a through mathematical foundation by Grossman and Mayer who also recognized the connection between wavelet and approximation theory [6][Gro84]. At

---

---

this point, many people contributed significantly. Lemerie, Stromberg, Battle and many other created new basis function for their application. But the major impetus to the development and popularity of the wavelet theory to the scientific community, in general, came from the works of Daubechies [4][Dau90] and Mallat [8][Mal89].

Daubechies introduced the concept of compactly supported wavelet and theory of frames. She also saw the connection between the wavelet theory and the theory of subband decomposition which are independently being pursued by the digital signal processing community of Electrical engineers. Mallat introduced the concept of multiresolution, which is intimately related to multirate digital filters used for subband decomposition.

### **2.3 Wavelets in the area of medical diagnosis**

A number of alternative time–frequency methods are now available for signal analysis. Of these, the wavelet transform has emerged over recent years as the most favored tool by researchers for analyzing problematic signals across a wide variety of areas in science, engineering and medicine [9](Addison 2002). It is especially valuable because of its ability to elucidate simultaneously local spectral and temporal information from a signal in a more flexible way than the STFT by employing a window of variable width.

Wavelet transform analysis has now been applied to a wide variety of biomedical signals including: the EMG, EEG, clinical sounds, respiratory patterns, blood pressure trends and DNA sequences (e.g. [10] Dupuis and Eugene (2000), [11] Hadjileontiadis and Panas (1997), [12] Marrone *et al* (1999), [13] Khalil and Duchene (2000), [14] Petrosian *et al* (2000), [15] Arneodo *et al* (1998)) and the subject of this review, the ECG.

Many of the ideas behind wavelet transforms have been in existence for a long time. However, wavelet transform analysis as we now know it really began in the mid 1980s where it was developed to interrogate seismic signals [16] (Goupillaud *et al* 1984). The application of wavelet transform analysis in science and engineering really

---

---

began to take off at the beginning of the 1990s, with a rapid growth in the numbers of researchers turning their attention to wavelet analysis during that decade.

## **2.4 Conclusion**

A brief history and development stages of wavelet theory is covered in this chapter. Also a brief survey has been studied regarding the application of wavelets in the area of medical diagnosis.

---

---

# ***Chapter III***

---

---

## CHAPTER III

### MATHEMATICAL MODEL OF WAVELET TRANSFORM

#### 3.1 Introduction

Wavelets are functions that satisfy certain mathematical requirements and are used in representing data or other functions. This idea is not new. Approximation using superposition of functions has existed since the early 1800s, when Joseph Fourier discovered that he could superpose sines and cosines to represent other functions. However, in wavelet analysis, the scale that we use to look at data plays a special role. Wavelet algorithms process data at different scales or resolutions. If we look at a signal (or a function) through a large “window” we would notice gross features. Similarly, if we look at a signal through a small “window” we would notice small features.

For many decades scientists have wanted more appropriate functions than the sines and cosines, which are the basis of Fourier analysis, to approximate choppy signals. By their definition, these functions are non-local (and stretch out to infinity). They therefore do a very poor job in approximating sharp spikes. Wavelets are well-suited for approximating data with sharp discontinuities. The wavelet analysis procedure is to adopt a wavelet prototype function, called an *analyzing wavelet* or *mother wavelet*. Temporal analysis is performed with a contracted, high-frequency version of the prototype wavelet, while frequency analysis is performed with a dilated, low-frequency version of the same wavelet. Because the original signal or function can be represented in terms of a wavelet expansion (using coefficients in a linear combination of the wavelet functions), data operations can be performed using just the corresponding wavelet coefficients. And if we further choose the wavelets best adapted to our data, or truncate the coefficients below a threshold, our data are sparsely represented. This sparse coding makes wavelets an excellent tool in the field of data compression.

---



---

### 3.2 The Continuous Wavelet Transform (CWT)

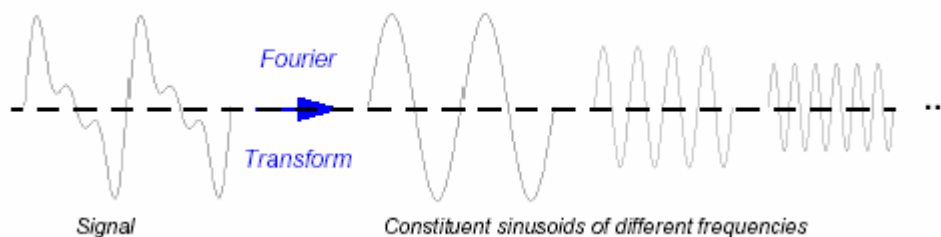
The continuous wavelet transform (CWT) is a time–frequency analysis method which differs from the more traditional short time Fourier transform (STFT) by allowing arbitrarily high localization in time of high frequency signal features.

Mathematically, the process of Fourier analysis is represented by the *Fourier transform*:

$$F(w) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt \quad (3.1)$$

which is the sum over all time of the signal  $f(t)$  multiplied by a complex exponential. (where complex exponential can be broken down into real and imaginary sinusoidal components.)

The results of the transform are the *Fourier coefficients*  $F(w)$ , which when multiplied by a sinusoid of frequency  $w$  yield the constituent sinusoidal components of the original signal. Graphically, the process looks like:



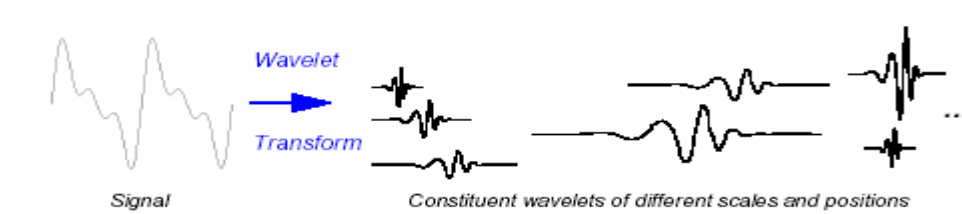
**Figure 3.1** Fourier Transform of a signal

Similarly, the *continuous wavelet transform* (CWT) is defined as the sum over all time of the signal multiplied by scaled, shifted versions of the wavelet function  $\Psi$ :

$$C(scale, position) = \int_{-\infty}^{\infty} f(t)\psi(scale, position, t) dt \quad (3.2)$$

---

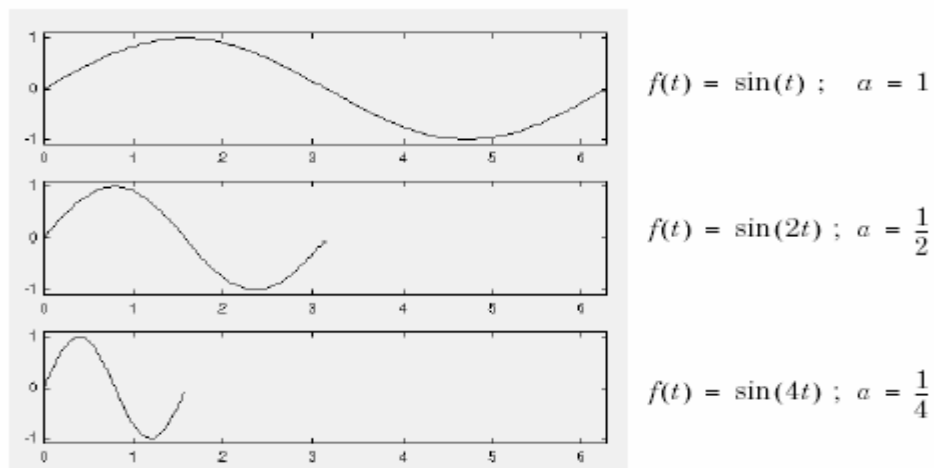
The results of the CWT are many *wavelet coefficients*  $C$ , which are a function of scale and position. Multiplying each coefficient by the appropriately scaled and shifted wavelet yields the constituent wavelets of the original signal:



**Figure 3.2** Wavelet Transform of a signal

### Scaling

Wavelet analysis produces a time-scale view of a signal. Scaling, as a mathematical operation, either dilates or compresses a signal. Scaling a wavelet simply means stretching (or compressing) it. Let us denote the *scale factor*, by the letter  $a$ . Let us see the effect of the scale factor on sinusoids:



**Figure 3.3** Scaling of a signal

Figure 3.3 shows that the scale factor=1/4 gives the compressed wave as compared to the scale factor=1. It is clear that, for a sinusoid, the scale factor  $a$  is related (inversely) to the radian frequency  $\omega$ . Similarly, with wavelet analysis, the scale is related to the frequency of the signal.

---

The scale factor works exactly the same with wavelets. The smaller the scale factor, the more “compressed” the wavelet.

### Scale and Frequency

As already discussed that the higher scales correspond to the most “stretched” wavelets. The more stretched the wavelet, the longer the portion of the signal with which it is being compared, and thus the coarser the signal features being measured by the wavelet coefficients.



**Figure 3.4** Correspondence between scale and frequency

Thus, there is a correspondence between wavelet scales and frequency as revealed by wavelet analysis:

- ❖ Low scale  $a \Rightarrow$  Compressed wavelet  $\Rightarrow$  Rapidly changing details  $\Rightarrow$  High frequency  $\omega$
- ❖ High scale  $a \Rightarrow$  Stretched wavelet  $\Rightarrow$  Slowly changing, coarse features  $\Rightarrow$  Low frequency  $\omega$

### Shifting

Shifting a wavelet simply means delaying (or hastening) its onset. Mathematically, delaying a function  $f(t)$  by  $k$  is represented by  $f(t-k)$ :



**Figure 3.5** Shifting of a signal

---

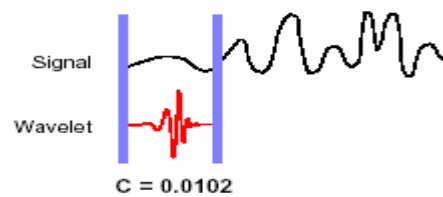
---

## Steps to create a Continuous Wavelet Transform

The continuous wavelet transform is the sum over all time of the signal multiplied by scaled, shifted versions of the wavelet. This process produces wavelet coefficients that are a function of scale and position.

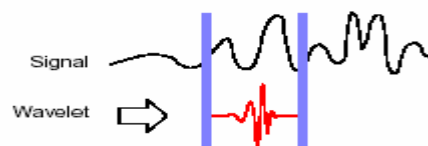
The five steps for creating a CWT:

- 1 Take a wavelet and compare it to a section at the start of the original signal.
- 2 Calculate a number,  $C$ , that represents how closely correlated the wavelet is with this section of the signal. The higher  $C$  is, the more the similarity. More precisely, if the signal energy and the wavelet energy are equal to one,  $C$  may be interpreted as a correlation coefficient. Note that the results will depend on the shape of the wavelet we choose.



**Figure 3.6 (a) Calculating wavelet coefficients using CWT**

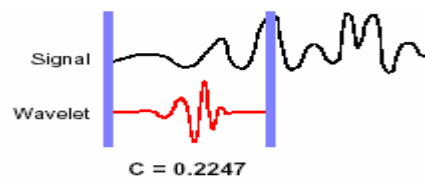
- 3 Shift the wavelet to the right and repeat steps 1 and 2 until we've covered the whole signal.



**Figure 3.6 (b) Calculating wavelet coefficients with shifted version of wavelet using CWT**

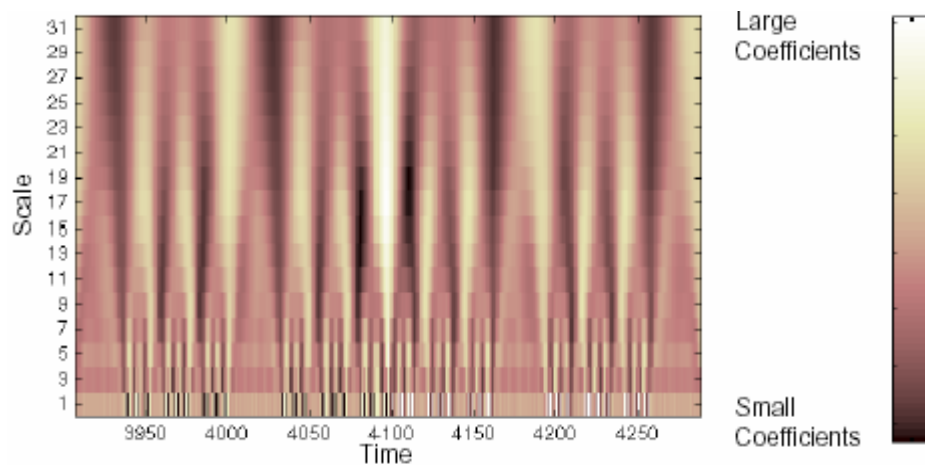
---

- 
- 4 Scale (stretch) the wavelet and repeat steps 1 through 3.



**Figure 3.6 (c) Calculating wavelet coefficients with higher scale of wavelet using CWT**

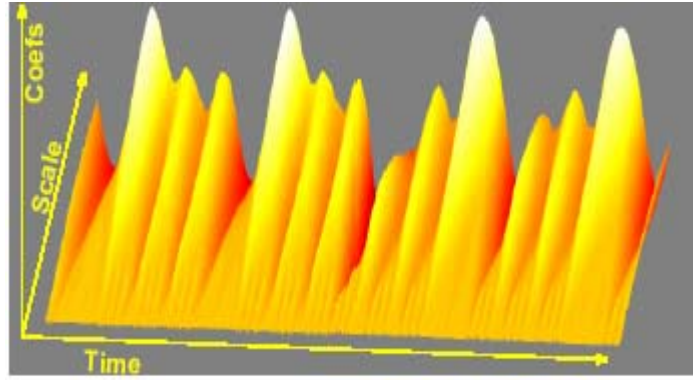
- 5 Repeat steps 1 through 4 for all scales. When we're done, we'll have the coefficients produced at different scales by different sections of the signal. The coefficients constitute the results of a regression of the original signal performed on the wavelets. Now we could make a plot on which the  $x$ -axis represents position along the signal (time), the  $y$ -axis represents scale, and the color at each  $x$ - $y$  point represents the magnitude of the wavelet coefficient  $C$ . These are the coefficient plots generated by the graphical tools.



**Figure 3.7 Coefficient plot generated using CWT**

These coefficient plots resemble a bumpy surface viewed from above. If we could look at the same surface from the side, we might see something like figure 3.8:

---



**Figure 3.8 Front view of Coefficient plot generated using CWT**

The continuous wavelet transform coefficient plots are precisely the time-scale view of the signal.

The CWT uses a variable window width, which is related to the scale of observation—a flexibility that allows for the isolation of the high frequency features. Rather, a large selection of localized waveforms can be employed as long as they satisfy predefined mathematical criteria (described below). The wavelet transform of a continuous time signal,  $x(t)$ , is defined as:

$$T(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t) \psi^*\left(\frac{t-b}{a}\right) dt \quad (3.3)$$

where  $\Psi^*(t)$  is the complex conjugate of the analyzing wavelet function  $\Psi(t)$ ,  $a$  is the dilation parameter of the wavelet and  $b$  is the location parameter of the wavelet. In order to be classified as a wavelet, a function must satisfy certain mathematical criteria. These are:

- (1) It must have finite energy:

$$E = \int_{-\infty}^{\infty} |\psi(t)|^2 dt < \infty \quad (3.4)$$

- (2) If  $\hat{\psi}(f)$  is the Fourier transform of  $\Psi(t)$ , i.e.

$$\hat{\psi}(f) = \int_{-\infty}^{\infty} \psi(t) e^{-i(2\pi f)t} dt < \infty \quad (3.5)$$

---

then the following condition must hold:

$$C_g = \int_0^{\infty} \frac{|\hat{\psi}(f)|^2}{f} df < \infty \quad (3.6)$$

This implies that the wavelet has no zero-frequency component, i.e.  $\hat{\psi}(f) \neq 0$ , or to put it another way, it must have a zero mean. Equation (3.6) is known as the *admissibility condition* and  $C_g$  is called the *admissibility constant*. The value of  $C_g$  depends on the chosen wavelet.

(3) For complex (or analytic) wavelets, the Fourier transform must both be real and vanish for negative frequencies. The contribution to the signal energy at the specific  $a$  scale and  $b$  location is given by the two-dimensional wavelet energy density function known as the scalogram (analogous to the *spectrogram*—the energy density surface of the STFT):

$$E(a,b) = |T(a,b)|^2 \quad (3.7)$$

In practice, all functions which differ from  $|T(a,b)|^2$  by only a constant multiplicative factor are also called scalograms, e.g.  $\frac{|T(a,b)|^2}{C_g}$ ,  $\frac{|T(a,b)|^2}{C_g f_c}$ , etc (where  $f_c$  is a characteristic frequency of the wavelet function). The scalogram can be integrated across  $a$  and  $b$  to recover the total energy in the signal using the admissibility constant,  $C_g$ , as follows:

$$E = \frac{1}{C_g} \int_{-\infty}^{\infty} \int_0^{\infty} |T(a,b)|^2 \frac{da}{a^2} db \quad \left[ = \int_{-\infty}^{\infty} x(t)^2 dt \right] \quad (3.8)$$

The relative contribution to the total energy contained within the signal at a specific  $a$  scale is given by the scale-dependent energy distribution:

$$E(a) = \frac{1}{C_g} \int_{-\infty}^{\infty} |T(a,b)|^2 db \quad (3.9)$$


---

---

Peaks in  $E(a)$  highlight the dominant energetic scales within the signal. We may convert the scale-dependent wavelet energy spectrum of the signal,  $E(a)$ , to a frequency dependent wavelet energy spectrum  $E_W(f)$  in order to compare directly with the Fourier energy spectrum of the signal  $E_F(f)$ . To do this, we must convert from the wavelet  $a$  scale (which can be interpreted as a representative temporal, or spatial, period for physical data) to a characteristic frequency of the wavelet such as the spectral peak frequency, passband centre, central frequency. The spectral components are inversely proportional to the dilation, i.e.  $f \propto 1/a$ . The frequency associated with a wavelet of arbitrary  $a$  scale is given by

$$f = \frac{f_c}{a} \quad (3.10)$$

where the characteristic frequency of the mother wavelet (the archetypal wavelet at scale  $a = 1$  and location  $b = 0$ ),  $f_c$ , becomes a scaling constant and  $f$  is the representative or frequency for the wavelet at arbitrary scale  $a$ . Finally, as with the Fourier transform, the original signal may be reconstructed using an inverse transform:

$$x(t) = \frac{1}{Cg} \int_{-\infty}^{\infty} \int_0^{\infty} T(a,b) \psi_{a,b}(t) \frac{dadb}{a^2} \quad (3.11)$$

In practice a fine discretization of the CWT is computed where usually the  $b$  location is discretized at the sampling interval and the scale is discretized logarithmically. As the wavelet transform given by equation (3.3) is a convolution of the signal with a wavelet function we can use the convolution theorem to express the integral as a product in Fourier space, i.e.,

$$T(a,b) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{x}(\omega) \hat{\psi}_{a,b}^*(\omega) d\omega \quad (3.12a)$$

where

---



---

$$\hat{\psi}_{a,b}^*(\omega) = \sqrt{a}\hat{\psi}^*(a\omega)e^{i\omega b} \quad (3.12b)$$

is the Fourier spectrum of the analysing wavelet at scale  $a$  and location  $b$ . In this way, a fast Fourier transform (FFT) algorithm can be employed in practice to speed up the computation of the wavelet transform.

For its practical implementation the continuous wavelet transform is computed over a finely discretized time–frequency grid. This discretization involves an approximation of the transform integral (i.e. a summation) computed on a discrete grid of  $a$  scales and  $b$  locations. In general, the wavelet transform is approximated in this way over each time step for a range of wavelet scales; there is therefore a heavy computational burden involved in the generation of the CWT and in general an order two in magnitude more wavelet values generated than original signal components.

### 3.3 The Discrete Wavelet Transform (DWT)

Calculating wavelet coefficients at every possible scale is a fair amount of work, and it generates an awful lot of data. If we choose only a subset of scales and positions at which we make our calculations it turns out, rather remarkably, if we choose scales and positions based on powers of two — so-called *dyadic* scales and positions — then our analysis will be much more efficient and just as accurate. We obtain such an analysis from the *discrete wavelet transform* (DWT).

The DWT of a signal  $x$  is calculated by passing it through a series of filters. First the samples are passed through a low pass filter with impulse response  $g$  resulting in a convolution of the two:

$$y[n] = (x * g)[n] = \sum_{k=-\infty}^{\infty} x[k] \cdot g[n-k] \quad (3.13)$$

---

---

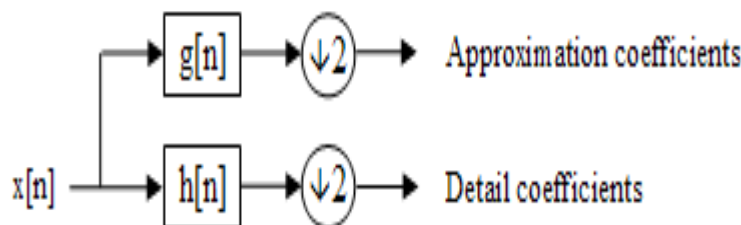
The signal is also decomposed simultaneously using a high-pass filter  $h$ . The outputs giving the detail coefficients (from the high-pass filter) and approximation coefficients (from the low-pass). It is important that the two filters are related to each other and they are known as a quadrature mirror filter.

However, since half the frequencies of the signal have now been removed, half the samples can be discarded according to Nyquist's rule. The filter outputs are then downsampled by 2:

$$y_{low}[n] = \sum_{k=-\infty}^{\infty} x[k] \cdot g[2 \cdot n - k] \quad (3.14a)$$

$$y_{high}[n] = \sum_{k=-\infty}^{\infty} x[k] \cdot h[2 \cdot n - k] \quad (3.14b)$$

This decomposition has halved the time resolution since only half of each filter output characterises the signal. However, each output has half the frequency band of the input so the frequency resolution has been doubled. This is in keeping with the Heisenberg Uncertainty Principle.



**Figure 3.9** Single level filter analysis

With the downsampling operator  $\downarrow$

$$(y \downarrow k)[n] = y[k \cdot n] \quad (3.15)$$

the summation expressed in equation 3.15 can be written more concisely as:-

---

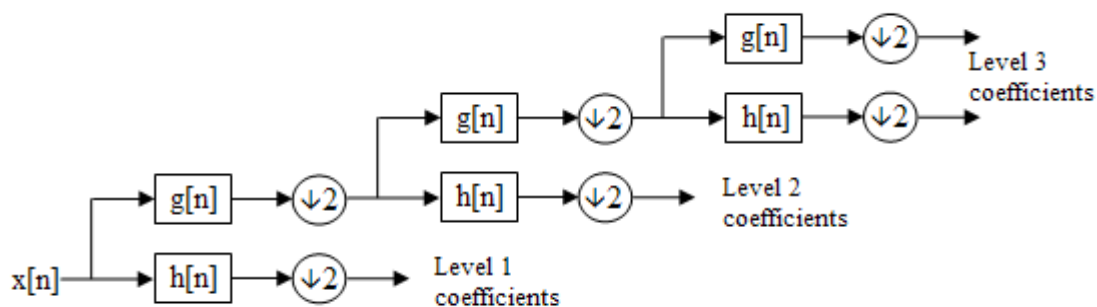
$$y_{low} = (x * g) \downarrow 2 \quad (3.16a)$$

$$y_{high} = (x * h) \downarrow 2 \quad (3.16b)$$

### Cascading and Filter Banks

The decomposition is repeated to further increase the frequency resolution and the approximation coefficients decomposed with high and low pass filters and then down-sampled. This is represented as a binary tree with nodes representing a sub-space with a different time-frequency localisation. The tree is known as a filter bank.

A **filter bank** is an array of band-pass filters that separates the input signal into several components, each one carrying a single frequency subband of the original signal. It also is desirable to design the filter bank in such a way that subbands can be recombined to recover original signal. The first process is called analysis, while the second is called synthesis. The output of analysis is referred as subband signal with as many subbands as there are filters in filter bank.

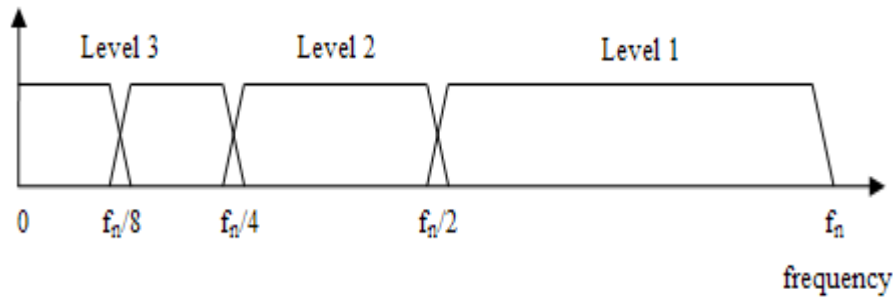


**Figure 3.10** 3 level filter bank

At each level figure 3.10 the signal is decomposed into low and high frequencies. Due to the decomposition process the input signal must be a multiple of  $2^n$  where  $n$  is the number of levels. For example a signal with 16 samples, frequency range 0 to  $f_n$  and 3 levels of decomposition, 4 output scales are produced:

---

Level	Frequencies	Samples
3	0 to $f_n / 8$	4
	$f_n / 8$ to $f_n / 4$	4
2	$f_n / 4$ to $f_n / 2$	8
1	$f_n / 2$ to $f_n$	16



**Figure 3.11** 4 output scales produced from signal of 16 samples, in frequency range of 0 to  $f_n$

### 3.3.1 One-Stage Filtering: Approximations and Details

For many signals, the low-frequency content is the most important part. It is what gives the signal its identity. The high-frequency content, on the other hand, imparts flavor or nuance. Consider the human voice, if we remove the high-frequency components, the voice sounds different, but we can still tell what's being said. However, if you remove enough of the low-frequency components, we hear gibberish.

In wavelet analysis, we often speak of *approximations* and *details*. The approximations are the high-scale, low-frequency components of the signal. The details are the low-scale, high-frequency components.

The filtering process, at its most basic level, looks like figure 3.12:

---

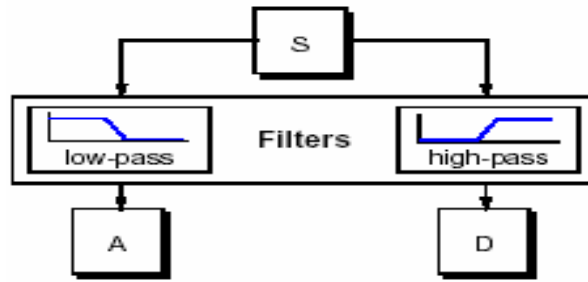


Figure 3.12 Filtering process

The original signal,  $S$ , passes through two complementary filters and emerges as two signals. Unfortunately, if we actually perform this operation on a real digital signal, we wind up with twice as much data as we started with. Suppose, for instance, that the original signal  $S$  consists of 1000 samples of data. Then the resulting signals will each have 1000 samples, for a total of 2000. These signals  $A$  and  $D$  are interesting, but we get 2000 values instead of the 1000 we had. There exists a more subtle way to perform the decomposition using wavelets. By looking carefully at the computation, we may keep only one point out of two in each of the two 2000-length samples to get the complete information. This is the notion of *downsampling* (Figure 3.13). We produce two sequences called  $cA$  and  $cD$ .

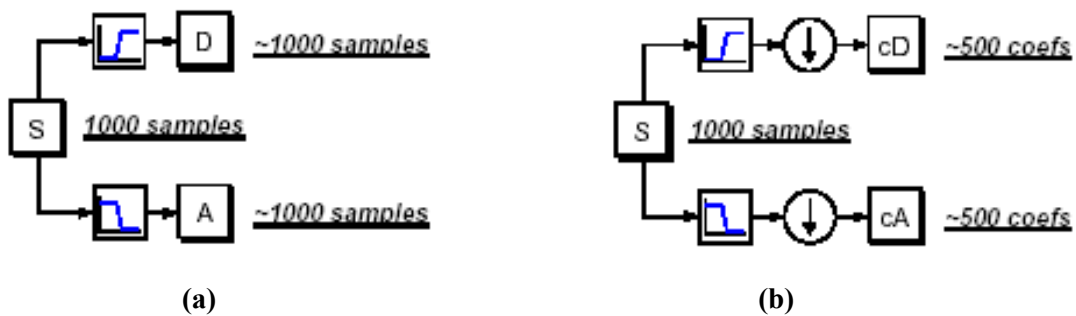


Figure 3.13 Notion of Downsampling (a) without downsampling (b) with downsampling

Figure 3.13(b), which includes downsampling, produces DWT coefficients. To gain a better appreciation of this process, let's perform a one-stage discrete wavelet transform of a signal. Our signal will be a pure sinusoid with high-frequency noise added to it. Here is our schematic diagram with real signals inserted into it:

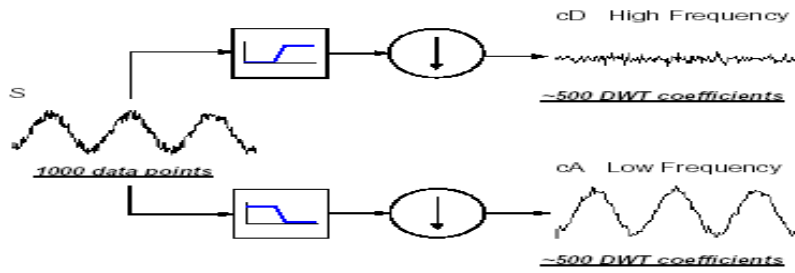


Figure 3.13(c) Notion of Downsampling of sinusoid with high-frequency noise

We can see that high frequency component is filtered out from signal  $s$  and approximation coefficient is free from high frequency (noise).

### 3.3.2 High & Low Pass Wavelet Filters

The first step of de-noising is the application of filters.  $H$  denotes the low pass filter, while  $G$  denotes the high pass filter. For a signal  $s_i$  where  $i = 1, \dots, 2^n$ ,  $H$  and  $G$  are defined by:

$$(Hs)_k = \sum_{j \in \mathbb{Z}} h_{j-2k} s_j \quad (3.17a)$$

and

$$(Gs)_k = \sum_{j \in \mathbb{Z}} g_{j-2k} s_j \quad (3.17b)$$

The  $h_j$  and  $g_j$  are called filter coefficients. For the Haar wavelets, the above equations simplify to the following:

$$(Hs)_k = h_0 s_{2k} + h_1 s_{2k+1} \quad (3.18a)$$

and

$$(Gs)_k = g_0 s_{2k} + g_1 s_{2k+1} \quad (3.18b)$$

---

since  $h_0$ ,  $h_1$ ,  $g_0$ , and  $g_1$  are the only non-zero filter coefficients. Note that for the Haar wavelets

$$h_0 = h_1 = \frac{1}{\sqrt{2}}, \quad (3.19a)$$

$$g_0 = \frac{1}{\sqrt{2}}, \quad (3.19b)$$

and

$$g_1 = -\frac{1}{\sqrt{2}}. \quad (3.19c)$$

The radicals come from normalizing the father and mother wavelets. In the case of the Haar wavelets, filtering simply averages and differences. The low pass filter does the following to a signal  $s = [a, b, c, d, e, f]$

$$s_R = \frac{a+b}{\sqrt{2}} + \frac{c+d}{\sqrt{2}} + \frac{e+f}{\sqrt{2}}. \quad (3.20a)$$

The high pass filter applied to  $s$  yields

$$s_G = \frac{a-b}{\sqrt{2}} + \frac{c-d}{\sqrt{2}} + \frac{e-f}{\sqrt{2}}. \quad (3.20b)$$

Thus, the low pass filter computes averages while the high pass filter accomplishes differencing. The process of differencing detects the noise in the signal. If some detail coefficients are small compared to the others, making them zero will not alter the signal too much. If the noise is located in these areas of the signal, de-noising will have a positive effect on the signal. The art is to choose a threshold level that will eliminate most of the noise, but preserve the other qualities of the signal.

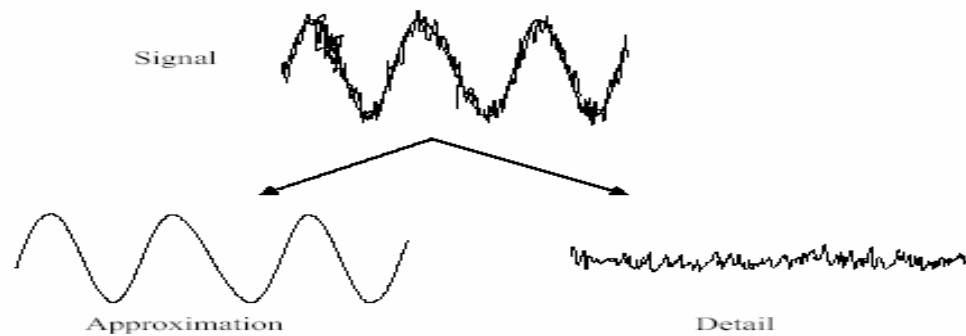
### 3.3.3 Multiresolution

Multiresolution refers to the simultaneous presence of different resolutions in a signal. For example, a signal can be broken down into a smooth background with fluctuations

---

---

on top of it.



**Figure 3.14** Multiresolution of a Signal. An arbitrary signal can be decomposed into approximation and detail components.

Figure 3.14 shows an arbitrary signal. The smooth background (low frequency) is known as the approximation, and the fluctuations (high frequency) are known as the details. Resolution increases as finer and finer details are added to a signal. At a lower resolution, a signal is approximated by the smooth signal, ignoring the detail fluctuations. The smooth, low frequency signal in Figure 3.14 is at a lower resolution than the original signal in Figure 3.14, because the original signal includes the high frequency detail fluctuations.

In a multiresolution analysis, a dataset is broken down into a hierarchy of several levels of approximation and detail maps. The approximation maps contain the image's low frequency information and the detail maps contain the high frequency information. At each hierarchical level, the approximation map is decomposed into descendent detail and approximation maps. In the hierarchical model, the resolution is the highest at the lowest level. As the levels increase, details are removed, and the approximation maps are at increasingly lower resolutions.

### 3.3.4 Multiple-Level Decomposition

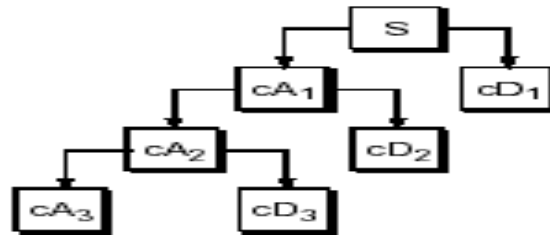
The decomposition process can be iterated, with successive approximations being decomposed in turn, so that one signal is broken down into many lower resolution

---



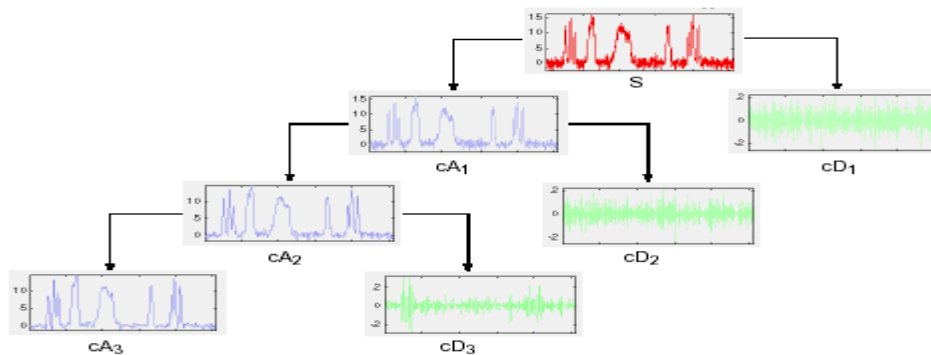
---

components. This is called the *wavelet decomposition tree* (figure 3.15).



**Figure 3.15** Wavelet Decomposition Tree

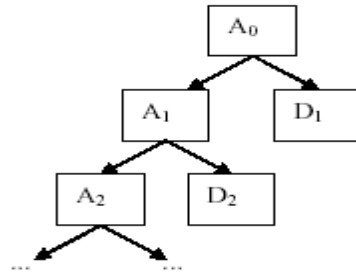
Looking at a signal's wavelet decomposition tree can yield valuable information.



**Figure 3.16** Wavelet Decomposition Tree of a signal showing valuable information

From figure 3.16 we can see that more and more detail component is filtered out at each stage. Consider a multiresolution hierarchy with  $j$  levels. Each level contains an approximation,  $A_j$ , and details  $D_j$ . The original data can be thought of as  $A_0$ . Approximation  $A_1$  is the low frequency components of  $A_0$ , and  $D_1$  is the high frequency components of  $A_0$ . The detail can be thought of as the difference between  $A_1$  and  $A_0$  so that  $A_0 = A_1 + D_1 = A_2 + D_2 + D_1$  etc... Figure 3.17 illustrates the hierarchical model.

---



**Figure 3.17 Hierarchical model of wavelet Decomposition**

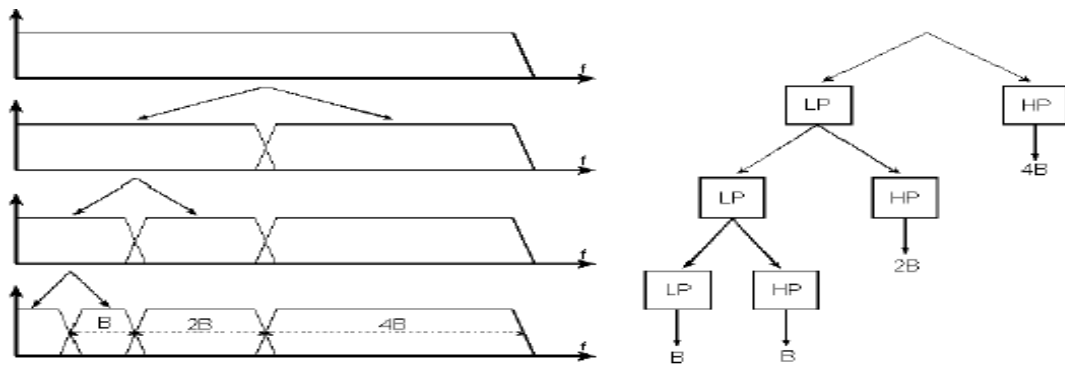
Mathematically: -

$$a_0(t) = a_j(t) + \sum_{k=1}^j d_k(t) \quad (3.21)$$

$$a_j(t) = a_{j+1}(t) + d_{j+1}(t) \quad (3.22)$$

where  $a_0(t)$  is the original signal,  $a(t)$  is the approximation,  $d(t)$  is the detail, and  $j$  is the level. Equation 3.21 says that the original signal can be retrieved from the approximation map at a given level,  $j$ , by adding the detail maps from all previous levels to the approximation map  $a_j(t)$ . Equation 3.22 says that the difference between the approximation map at a given level,  $j$ , and the approximation map at the next level,  $j+1$ , is the detail map at the next level,  $j+1$ . An original dataset is decomposed into a multiresolution hierarchy by exposing the dataset to a filter bank made up of a high pass and low pass filter. The original dataset is at the top of the hierarchy. To construct the next level of the hierarchy, the original dataset is exposed to a high pass filter, which removes the high frequency or detail information, and a low pass filter, which separates the low frequency or approximation information. Iteratively exposing the approximation map at each level to the high pass and low pass filter bank creates the next level of the hierarchy. Wavelet decomposition is a type of multiresolution decomposition. The wavelet decomposition can be thought of as a filter bank is made up of a high pass component called the wavelet function, and a low pass component called the scaling function. Figure 3.18 shows the frequency space during the wavelet decomposition. The scaling function at the first level is the original signal. At each subsequent level, the scaling function is split into approximation and detail information with wavelet and scaling filters.

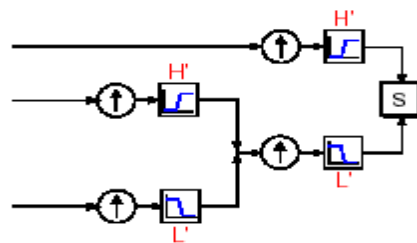
---



**Figure 3.18** Wavelet Decomposition Example. HP is the high pass wavelet filter, and LP is the low pass scaling filter. B, 2B, and 4B are the coefficients at each level. At each level the low pass approximation information is divided into another level of approximation and detail information by applying the low pass scaling filter and the high pass wavelet filter.

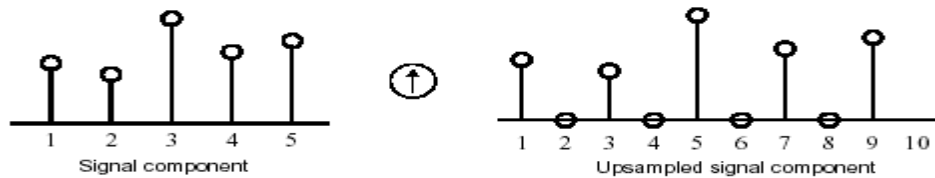
### 3.3.5 Wavelet Reconstruction

We've learned how the discrete wavelet transform can be used to analyze, or decompose, signals and images. This process is called *decomposition* or *analysis*. Now we can see how those components can be assembled back into the original signal without loss of information. This process is called *reconstruction*, or *synthesis*. The mathematical manipulation that effects synthesis is called the *inverse discrete wavelet transform* (IDWT). To synthesize a signal, we reconstruct it from the wavelet coefficients:



**Figure 3.19** Reconstruction Process

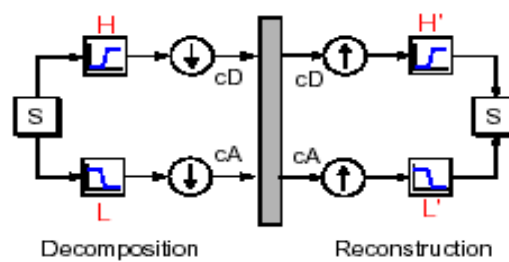
Where wavelet analysis involves filtering and downsampling, the wavelet reconstruction process consists of upsampling and filtering. Upsampling is the process of lengthening a signal component by inserting zeros between samples (figure 3.20):



**Figure 3.20** Upsampling

### Reconstruction Filters

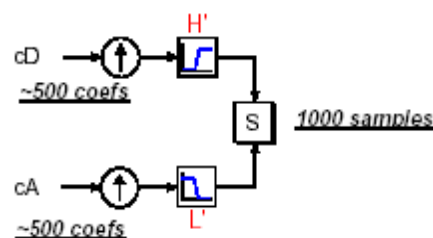
In the filtering part of the reconstruction process, it is the choice of filters that is crucial in achieving perfect reconstruction of the original signal. The downsampling of the signal components performed during the decomposition phase introduces a distortion called aliasing. It turns out that by carefully choosing filters for the decomposition and reconstruction phases that are closely related (but not identical), we can “cancel out” the effects of aliasing.



**Figure 3.21** Notion of Decomposition and Reconstruction.

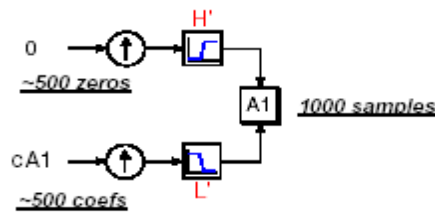
### Reconstructing Approximations and Details

We have seen that it is possible to reconstruct our original signal from the coefficients of the approximations and details.



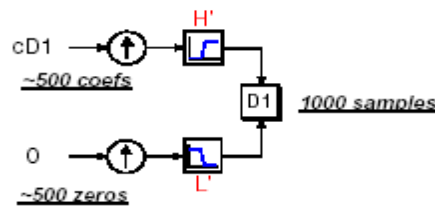
**Figure 3.22 (a)** Constructing approximation and detail coefficients

It is also possible to reconstruct the approximations and details themselves from their coefficient vectors. For example, let's reconstruct the first-level approximation  $A1$  from the coefficient vector  $cA1$ . We pass the coefficient vector  $cA1$  through the same process we used to reconstruct the original signal. However, instead of combining it with the level-one detail  $cD1$ , we feed in a vector of zeros in place of the detail coefficients



**Figure 3.22 (b) Reconstruction of approximation  $A1$**

The process yields a reconstructed *approximation*  $A1$ , which has the same length as the original signal  $S$  and which is a real approximation of it. Similarly, we can reconstruct the first-level detail  $D1$ , using the analogous process:



**Figure 3.22 (c) Reconstruction of Detail  $D1$**

The reconstructed details and approximations are true constituents of the original signal. In fact, we find when we combine them that  $A1 + D1 = S$ . Note that the coefficient vectors  $cA1$  and  $cD1$ , because they were produced by downsampling and are only half the length of the original signal, cannot directly be combined to reproduce the signal. It is necessary to reconstruct the approximations and details before combining them.

Extending this technique to the components of a multilevel analysis, we find that similar relationships hold for all the reconstructed signal constituents.

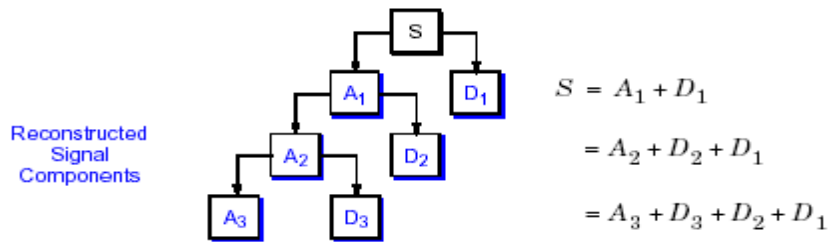


Figure 3.23 Reconstructing signal using Multilevel analysis

### Multistep Decomposition and Reconstruction

A multistep analysis-synthesis process can be represented as:

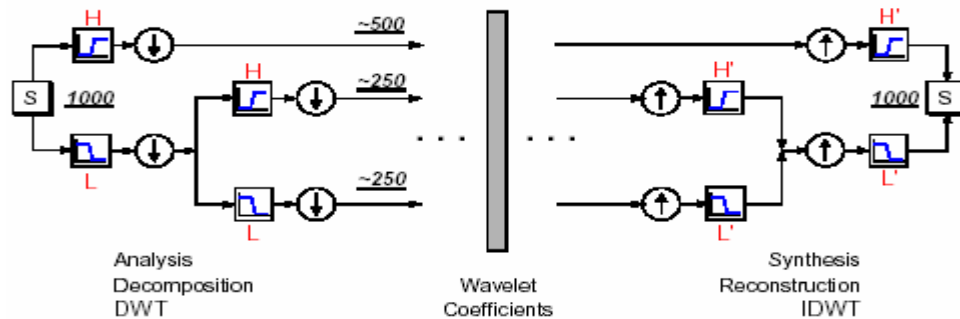


Figure 3.24 Multistep Decomposition and Reconstruction.

This process involves two aspects: breaking up a signal to obtain the wavelet coefficients, and reassembling the signal from the coefficients. Of course, there is no point breaking up a signal merely to have the satisfaction of immediately reconstructing it. We may modify the wavelet coefficients before performing the reconstruction step. We perform wavelet analysis because the coefficients thus obtained have many known uses, de-noising and compression being foremost among them.

In its most common form, the DWT employs a dyadic grid (integer power of two scaling in  $a$  and  $b$ ) and orthonormal wavelet basis functions and exhibits zero redundancy. (Actually, the transform integral remains continuous for the DWT but is determined only on a discretized grid of  $a$  scales and  $b$  locations. In practice, the input

---

signal is treated as an initial wavelet approximation to the underlying continuous signal from which, using a multiresolution algorithm, the wavelet transform and inverse transform can be computed discretely, quickly and without loss of signal information). A natural way to sample the parameters  $a$  and  $b$  is to use a logarithmic discretization of the  $a$  scale and link this, in turn, to the size of steps taken between  $b$  locations. To link  $b$  to  $a$  we move in discrete steps to each location  $b$ , which are proportional to the  $a$  scale. This kind of discretization of the wavelet has the form

$$\psi_{m,n}(t) = \frac{1}{\sqrt{a_0^m}} \psi\left(\frac{t - nb_0 a_0^m}{a_0^m}\right) \quad (3.23)$$

where the integers  $m$  and  $n$  control the wavelet dilation and translation respectively;  $a_0$  is a specified fixed dilation step parameter set at a value greater than 1, and  $b_0$  is the location parameter which must be greater than zero. A common choice for discrete wavelet parameters  $a_0$  and  $b_0$  are 2 and 1 respectively. This power-of-two logarithmic scaling of both the dilation and translation steps is known as the dyadic grid arrangement. The *dyadic grid* is perhaps the simplest and most efficient discretization for practical purposes and lends itself to the construction of an orthonormal wavelet basis. Substituting  $a_0 = 2$  and  $b_0 = 1$  into equation (3.23) we see that the dyadic grid wavelet can be written compactly, as:

$$\psi_{m,n}(t) = 2^{-m/2} \psi(2^{-m} t - n) \quad (3.24)$$

Note that this has the same notation as the general discrete wavelet given by equation (3.23). From here on,  $\Psi_{m,n}(t)$  will be used only to denote dyadic grid scaling with  $a_0 = 2$  and  $b_0 = 1$ . Discrete dyadic grid wavelets are usually chosen to be orthonormal, i.e. they are both orthogonal to each other and are normalized to have unit energy.

This is expressed as:

$$\int_{-\infty}^{\infty} \psi_{m,n}(t) \psi_{m',n'}(t) dt = \begin{cases} 1 & \text{if } m=m' \text{ and } n=n' \\ 0 & \text{otherwise} \end{cases} \quad (3.25)$$


---

---

This means that the information stored in a wavelet coefficient  $T_{m,n}$  obtained from the wavelet transform is not repeated elsewhere and allows for the complete regeneration of the original signal without redundancy. The corresponding family of orthonormal wavelets is an orthonormal basis. (A basis is a set of vectors, a combination of which can completely define the signal,  $x(t)$ . An orthonormal basis has component vectors which, in addition to being able to completely define the signal, are perpendicular to each other).

Using the dyadic grid wavelet of equation (3.24), the *discrete wavelet transform* (DWT) can be written as:

$$T_{m,n} = \int_{-\infty}^{\infty} x(t)\psi_{m,n}(t)dt \quad (3.26)$$

where  $T_{m,n}$  is known as the wavelet (or detail) coefficient at scale and location indices  $(m, n)$ . Before continuing it is important to make clear the distinct difference between the DWT and the discretized approximations of the CWT used in practice. The discretizations of the continuous wavelet transform, required for its practical implementation, involve a discrete approximation of the transform integral (i.e. a summation) computed on a discrete grid of  $a$  scales and  $b$  locations. The inverse continuous wavelet transform is also computed as a discrete approximation. How close an approximation to the original signal is recovered depends mainly on the resolution of the discretization used and, with care, usually a very good approximation can be recovered. On the other hand, for the DWT, as defined in equation (3.26), the transform integral remains continuous but is determined only on a discretized grid of  $a$  scales and  $b$  locations. We can then sum the DWT coefficients to infinity over  $m$  and  $n$  to get the original signal back exactly.

Orthonormal dyadic discrete wavelets are associated with *scaling functions* and their dilation equations. The scaling function is associated with the smoothing of the signal and has the same form as the wavelet, given by:

---



---


$$\phi_{m,n}(t) = 2^{-m/2} \phi(2^{-m}t - n) \quad (3.27)$$

They have the property

$$\int_{-\infty}^{\infty} \phi_{0,0}(t) dt = 1 \quad (3.28)$$

where  $\phi_{0,0}(t) = \phi(t)$  is sometimes referred to as the father scaling function or father wavelet (cf mother wavelet). (The integral of a wavelet function is zero.) The scaling function is orthogonal to translations of itself, *but not to dilations of itself*. The scaling function can be convolved with the signal to produce *approximation coefficients* as follows:

$$S_{m,n} = \int_{-\infty}^{\infty} x(t) \phi_{m,n}(t) dt \quad (3.29)$$

From the above, we can see that the approximation coefficients are simply weighted averages of the continuous signal factored by  $2^{m/2}$ . The approximation coefficients at a specific scale  $m$  are collectively known as the *discrete approximation* of the signal at that scale. A *continuous approximation* of the signal at scale  $m$  can be generated by summing a sequence of scaling functions at this scale factored by the approximation coefficients as follows:

$$x_m(t) = \sum_{n=-\infty}^{\infty} S_{m,n} \phi_{m,n}(t) \quad (3.30)$$

where  $x_m(t)$  is a smooth, scaling-function-dependent version of the signal  $x(t)$  at scale index  $m$ . This continuous approximation approaches  $x(t)$  at small scales, i.e. as  $m \rightarrow -\infty$ . A signal  $x(t)$  can then be represented using a combined series expansion using both the approximation coefficients and the wavelet (detail) coefficients as follows:

$$x(t) = \sum_{n=-\infty}^{\infty} S_{m_0,n} \phi_{m_0,n}(t) + \sum_{m=-\infty}^{m_0} \sum_{n=-\infty}^{\infty} T_{m,n} \psi_{m,n}(t) \quad (3.31)$$


---

---

We can see from this equation that the original continuous signal is expressed as a combination of an approximation of itself, at arbitrary scale index  $m_0$ , added to a succession of signal details from scales  $m_0$  down to negative infinity. The *signal detail* at scale  $m$  is defined as:

$$d_m(t) = \sum_{n=-\infty}^{\infty} T_{m,n} \psi_{m,n}(t) \quad (3.32)$$

hence we can write equation (3.31) as

$$x(t) = x_{m_0}(t) + \sum_{m=-\infty}^{m_0} d_m(t) \quad (3.33)$$

From this equation it is easy to show that

$$x_{m-1}(t) = x_m(t) + d_m(t) \quad (3.34)$$

which tells us that if we add the signal detail at an arbitrary scale (index  $m$ ) to the approximation at that scale we get the signal approximation at an increased resolution (i.e. at a smaller scale, index  $m - 1$ ). This is called a *multiresolution representation* [20][Mal89] (Mallat 1989).

### 3.4 Wavelet packet decomposition

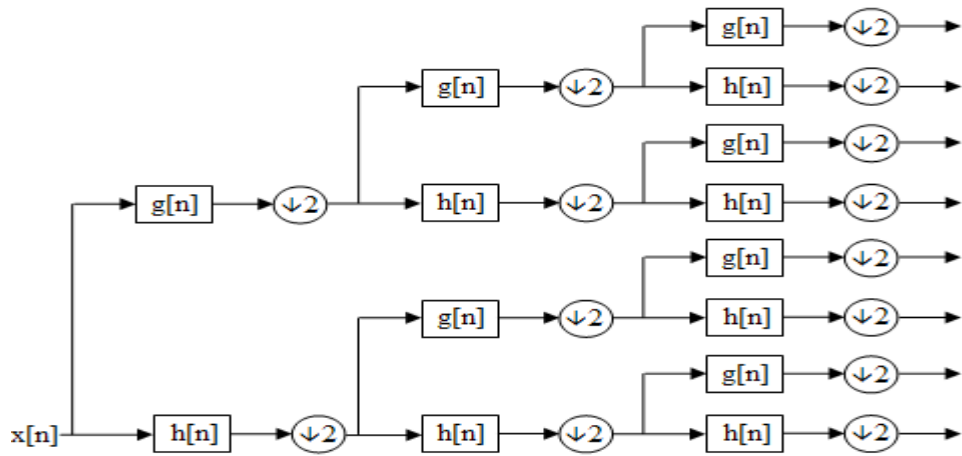
Wavelet packet decomposition (WPD) (sometimes known as just wavelet packets) is a wavelet transform where the signal is passed through more filters than the DWT.

In the DWT, each level is calculated by passing the previous approximation coefficients through a high and low pass filters. However in the WPD, both the detail and approximation coefficients are decomposed.

For  $n$  levels of decomposition the WPD produces  $2^n$  different sets of coefficients (or nodes) as opposed to  $(n + 1)$  sets for the DWT. However, due to the downsampling process the overall number of coefficients is still the same and there is no redundancy.

---

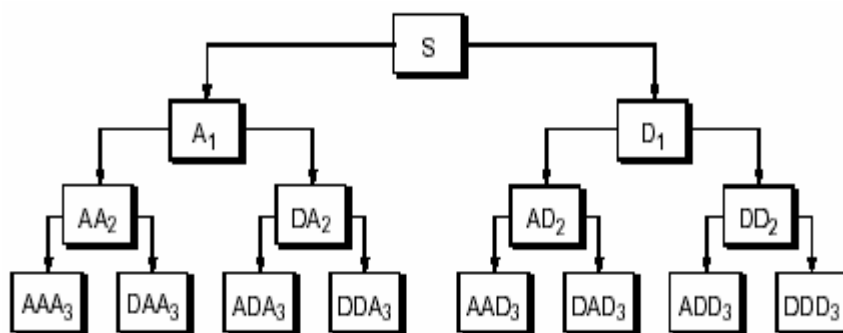
The two-dimensional DWT leads to a decomposition of approximation coefficients at level  $j$  in four components: the approximation at level  $j+1$ , and the details in three orientations (horizontal, vertical, and diagonal).



**Figure 3.25** Wavelet Packet decomposition over 3 levels

The *wavelet packet* method is a generalization of wavelet decomposition that offers a richer range of possibilities for signal analysis. In wavelet analysis, a signal is split into an approximation and a detail. The approximation is then itself split into a second-level approximation and detail, and the process is repeated.

In wavelet packet analysis, the details as well as the approximations can be split. This yields more than  $2^{2^n - 1}$  different ways to encode the signal. This is the *wavelet packet decomposition tree*.

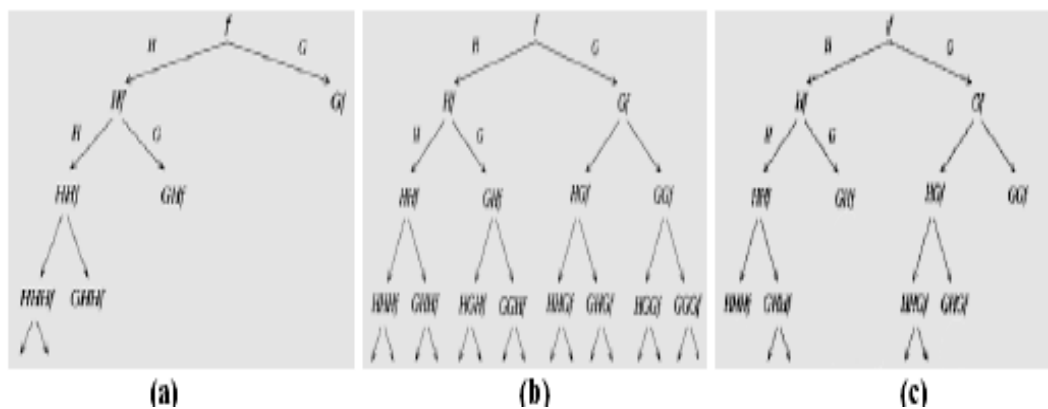


**Figure 3.26** Wavelet Packet decomposition Tree

The wavelet decomposition tree is a part of this complete binary tree. For instance,

wavelet packet analysis allows the signal  $S$  to be represented as  $A1 + AAD3 + DAD3 + DD2$ . This is an example of a representation that is not possible with ordinary wavelet analysis. Choosing one out of all these possible encodings presents an interesting problem. We use an *entropy-based criterion* to select the most suitable decomposition of a given signal.

Unlike dyadic wavelet transform, wavelet packets decompose the low frequency component as well as the high frequency component in every sub-bands [17] {jin\_Coifman\_1992}. Such adaptive expansion can be represented with binary trees where each sub-band high or low frequency component is a node with two children corresponding to the pair of high and low frequency expansion at the next scale. An admissible tree for an adaptive expansion is therefore defined as a binary tree where each node has either 0 or 2 children, as illustrated in Figure 3.27 (c). The number of all different wavelet packet orthogonal basis (also called a wavelet packets dictionary) equal to the number of different admissible binary trees, which is of the order of  $2^{2^J}$ , where  $J$  is the depth of decomposition [18] {jin\_Mallat\_1998}.



**Figure 3.27** (a) Dyadic wavelet decomposition tree. (b) Wavelet packets decomposition tree. (c) An example of an orthogonal basis tree with wavelet packets decomposition.

Obviously, wavelet packets provide more flexibility on partitioning the spatial-frequency domain, and therefore improve the separation of noise and signal into different sub-bands in an approximated sense (this is referred to the near-diagonalization of signal and noise). This property can greatly facilitate the

enhancement and de-noising task of a noisy signal if the wavelet packets basis are selected properly [19] {jin\_Coifman\_1995b}. In practical applications for various medical imaging modalities and applications, features of interest and noise properties have significantly different characteristics that can be efficiently characterized separately with this framework. A fast algorithm for wavelet-packets best basis selection was introduced by Coifman and Wickerhauser in [19] {jin\_Coifman\_1995b}. This algorithm identifies the “best” basis for a specific problem inside the wavelet packets dictionary according to a criterion (referred to as a cost function) that is minimized. This cost function typically reflects the entropy of the coefficients or the energy of the coefficients inside each sub-band and the optimal choice minimizes the cost function comparing values at a node and its children.

### 3.5 Discrete Stationary Wavelet Transform (SWT)

The Stationary Wavelet Transform (SWT) is similar to the DWT except the signal is never subsampled and the filters are different for each level of decomposition.

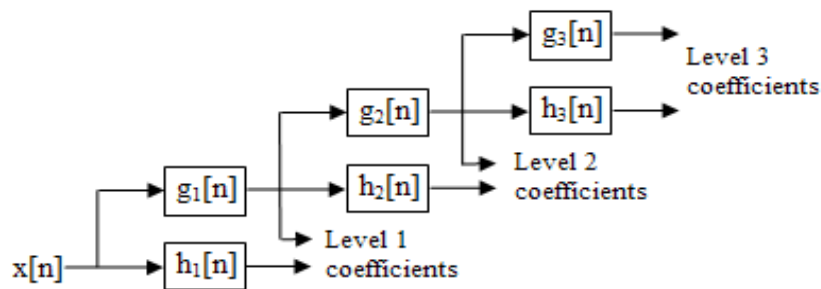


Figure 3.28 3 level SWT filter bank

Each level's filters are up-sampled versions of the previous.

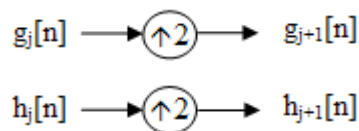


Figure 3.29 SWT filters

---

The SWT is an inherently redundant scheme as each set of coefficients contains the same number of samples as the input – so for a decomposition of  $N$  levels there is a redundancy of  $2N$ .

### **Synonyms**

The idea of omitting the downsampling in the discrete wavelet transform is so obvious, that this variant was invented several times with different names.

- Stationary Wavelet Transform
- Redundant Wavelet Transform
- Algorithmé à trous
- Quasi-continuous wavelet transform
- Translation invariant wavelet transform
- Shift invariant wavelet transform
- Cycle spinning
- Maximal overlap wavelet transform
- Redundant wavelet transform

### **Undecimated wavelet transform**

DWT suffers a drawback that the DWT is not a time-invariant transform. This means that, even with periodic signal extension, the DWT of a translated version of a signal  $X$  is not, in general, the translated version of the DWT of  $X$ .

The idea is to average some slightly different DWT, called e-decimated DWT, to define the stationary wavelet transform (SWT). This property is useful for several applications such as breakdown points detection. The main application of the SWT is de-noising.

The principle is to average several de-noised signals. Each of them is obtained using the usual de-noising scheme, but applied to the coefficients of an e-decimated DWT.

---

---

There is a restriction: we define the SWT only for signals of length divisible by  $2^J$ , where  $J$  is the maximum decomposition level, and we use the DWT with periodic extension.

### 3.5.1 e-Decimated DWT

There exist a lot of slightly different ways to handle the discrete wavelet transform. As we know that the DWT basic computational step is a convolution followed by a decimation. The decimation retains even indexed elements. But the decimation could be carried out by choosing odd indexed elements instead of even indexed elements. This choice concerns every step of the decomposition process, so at every level we chose odd or even. If we perform all the different possible decompositions of the original signal, we have  $2^J$  different decompositions, for a given maximum level  $J$ .

Let us denote by  $e_j = 1$  or  $0$  the choice of odd or even indexed elements at step  $j$ .

Every decomposition is labeled by a sequence of 0's and 1's:  $e = e_1, \dots, e_J$ . This transform is called the e-decimated DWT.

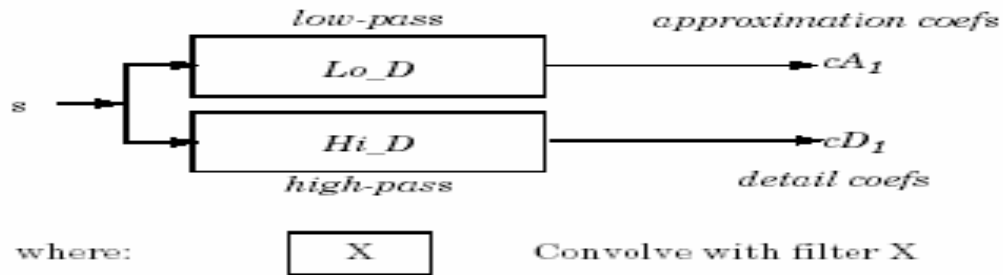
We can obtain the basis vectors of the e-decimated DWT from those of the standard DWT by applying a shift and corresponds to a special choice of the origin of the basis functions.

It is possible to calculate all the e-decimated DWT for a given signal of length  $N$ , by computing the approximation and detail coefficients for every possible sequence  $e$ . Of course to calculate all the e-decimated DWT many computations are performed many times. We shall now describe another way, which is the stationary wavelet transform (SWT).

The SWT algorithm is very simple and is close to the DWT one. More precisely, for level 1, all the e-decimated DWT (only two at this level) for a given signal can be

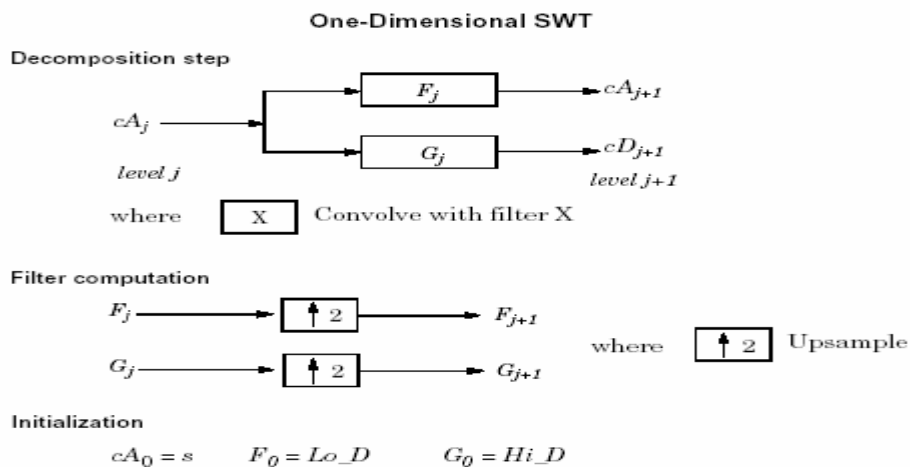
---

obtained by convolving the signal with the appropriate filters as in the DWT case but without downsampling. Then the approximation and detail coefficients at level 1 are both of size  $N$ , which is the signal length. This can be visualized in the following figure.



**Figure 3.30** Convolving the signal with the appropriate filters without downsampling (SWT)

The general step  $j$  convolves the approximation coefficients at level  $j-1$ , with upsampled versions of the appropriate original filters, to produce the approximation and detail coefficients at level  $j$ . This can be visualized in the following figure.



**Figure 3.31** 1D SWT



---

The standard Discrete wavelet transform is based on filter H and G and on a binary decimation operator  $D_o$ . The binary decimation operator simply chooses every even member of a sequence, so that :

$$(D_o x)_j = x_{2j} \quad \text{for all interger } j. \quad (3.35)$$

If x is a finite sequence of length  $2^m$  with periodic boundary conditions applied, then each of  $D_o Gx$  and  $D_o Hx$  will be sequences of periodic length  $2^{m-1}$ .

Basic DWT algorithm can be modified to give stationary wavelet transform. We simply apply appropriate high and low pass filters to the data at each level to produce two sequences at the next level. We do not decimate, and the two new sequences each have same length as the original sequence. Instead we modify the filters at each level , by padding them out with zeros.

Let  $Z$  be the operator that alternates the given sequence with zeroes, so that, for all integers j,  $(Zx)_{2j} = x_j$  and  $(Zx)_{2j-1} = 0$ . Define filters  $H^{[r]}$  and  $G^{[r]}$  to have weights  $Z^r h$  and  $Z^r g$  respectively. Thus the filter  $H^{[r]}$  has weights  $h_{2^r j}^{[r]} = h_j$  and  $h_k^{[r]} = 0$  if k is not a multiple of  $2^r$ , the filter  $H^{[r]}$  is obtained by inserting a zeros between every adjacent pair of elements of the filter  $H^{[r-1]}$ , and similarly for  $G^{[r]}$ .

### 3.6 Conclusion

A brief introduction has been made regarding the mathematical background of different types of wavelet transform and the various aspects which differentiate each type from one another.

---

---

# ***Chapter IV***

---

---

## CHAPTER IV

### WAVELET IN BIOSIGNAL PROCESSING

#### 4.1 Introduction

ECG signal (Electro Cardio Graphy) is the graphical recording of heart's electrical activity. The typical shape of a healthy ECG signal is well known to cardiologists. Any significant deviation from that shape is usually considered to be a symptom of a pathological condition. Recently, the new computerized ECG recorders/analyzers also utilize the frequency information to decide whether a pathological condition exists. A pathological condition can sometimes be diagnosed more easily when the frequency content of the signal is analyzed.

Magnetic resonance imaging (MRI) scan is an imaging technique used primarily in medical field to produce high quality images of the soft tissues of the human body. Using brain images acquired by MRI often allows physicians and engineers to analyze the brain without the need for invasive surgery.

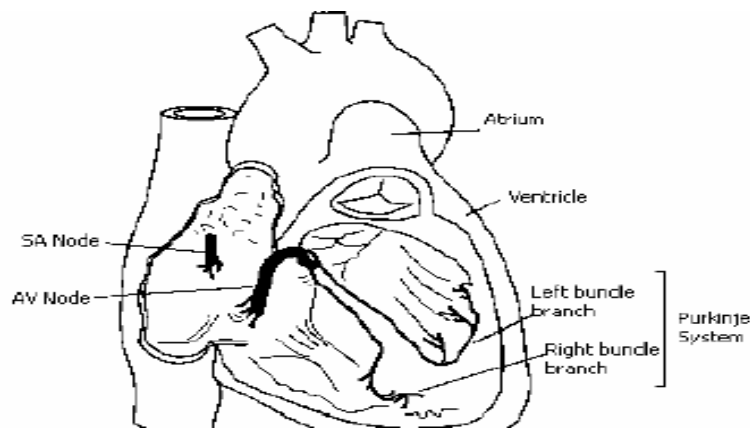
#### 4.2 Biopotentials in the Heart

Muscular contraction is associated with electrical changes known as depolarization. The electrocardiogram (ECG) is a measure of this electrical activity associated with the heart. The ECG is measured at the body surface and results from electrical changes associated with activation first of the two small heart chambers, the atria, and then of the two larger heart chambers, the ventricles. The contraction of the atria manifests itself as the 'P' wave in the ECG and contraction of the ventricles produces the feature known as the 'QRS' complex. The subsequent return of the ventricular mass to a rest state—repolarization— produces the 'T' wave. Repolarization of the atria is, however, hidden within the dominant QRS complex. Analysis of the local morphology of the ECG signal and its time varying properties has produced a variety

---

---

of clinical diagnostic tools. In this section we review the application of the wavelet transform to the analysis of the ECG signal.



**Figure 4.1 Heart**

During diastole, while the heart is at rest, all of the cells are polarized so that the potential inside each cell is negative with respect to the outside. Normally, depolarization occurs first at the SA node, making the outside of the tissue negative with respect to the inside of the cells and making it negative with respect to the tissues not yet depolarized. This imbalance results in an ionic current,  $I$ , Figure 4.2, causing the left arm to measure positive with respect to the right arm. The resulting voltage is called the P wave. After about 90 ms, the atrium is completely depolarized, and the ionic current measured by lead I reduces to zero. The depolarization then passes through the AV node causing a delay of 110ms. The depolarization then passes to the right ventricular muscle depolarizing it and making it negative relative to the still polarized left ventricular muscle. Again the direction of  $I$  causes a + to - voltage from LA to RA called the R-wave. The complete wave form is called an electrocardiogram with labels P, Q, R, S, and T indicating its distinctive features. The P wave arises from the depolarization of the atrium. The QRS complex arises from depolarization of the ventricles. The magnitude of the R-wave within this complex is approximately 1mV. The T- wave arises from re-polarization of the ventricle muscle. The U wave that some times follow the T-wave is second order effect of uncertain origin and is of little diagnostic significance.

---

---

**PR** interval lies between 0.12 to 0.2s

**QRS** interval normally lies between 0.05 to 0.10s

The intervals, segments, and complexes are shown in the Figure 4.3.

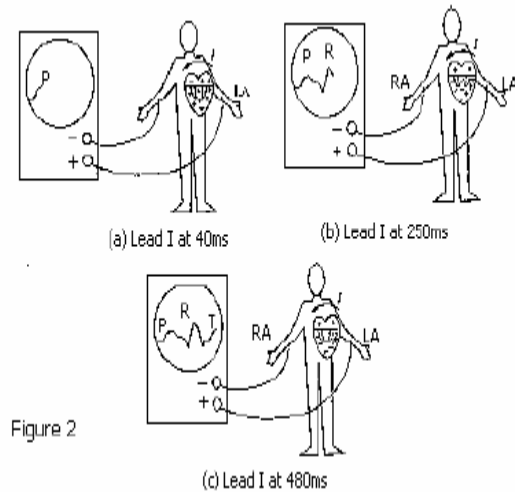
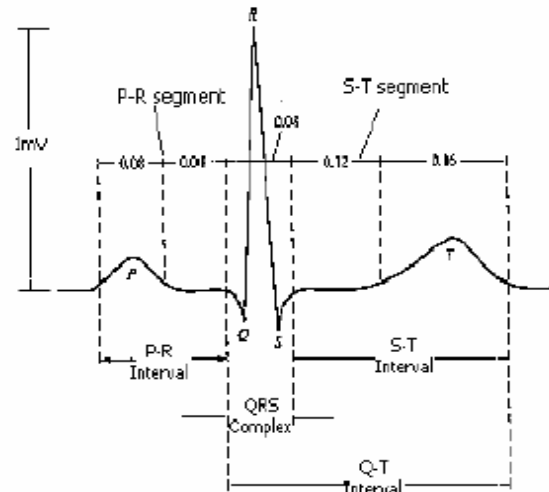


Figure 2

**Figure 4.2 ECG Leads**



**Figure 4.3 ECG Signal**

### Parts of an ECG

The hills and valleys on an ECG recording are called waves. Each wave tells a doctor a story about how the heart is working.

- The P-wave shows our heart's upper chambers (atria) contracting
- The QRS complex shows our heart's lower chamber (ventricles) contracting
- The T-wave shows the heart's lower chamber (ventricles) relaxing.

By noting the shape, consistency, and the time between these waveforms we can learn more about the conduction system, damaged area, and the areas that are not receiving enough oxygen to meet their needs. We can measure the distance of various intervals in the complex of waveforms to learn about the conduction system.

- The first is the P-R interval. This is measured from the beginning of the P-wave to the beginning of the QRS waves. Normal measurement for this
-

---

interval are 0.12seconds to 0.20 seconds.

- Next is the QRS interval. This is measured from the beginning of the first wave in the QRS to where the last wave in the QRS return to the baseline. Normal measurement for this interval are 0.04 sec to 0.12 sec.
- Another interval to look at is the Q-T interval. This is measured from the beginning of the QRS waves to end of the T-wave. Normal measurements for this interval are based on the heart rate. Generally it should be half of the distance between two QRS complexes, 0.4 sec or less.
- The ST interval occur when the heart contracts and when the cell membranes are depolarised but relatively stable, the measurement is 0.05-0.15 sec.

### 4.3 Brain Structures and their Functions

The nervous system is your body's decision and communication center. The central nervous system (CNS) is made of the brain and the spinal cord and the peripheral nervous system (PNS) is made of nerves. Together they control every part of our daily life, from breathing and blinking to helping us memorize facts for a test. Nerves reach from your brain to your face, ears, eyes, nose, and spinal cord and from the spinal cord to the rest of your body. Sensory nerves gather information from the environment, send that info to the spinal cord, which then speed the message to the brain. The brain then makes sense of that message and fires off a response. Motor neurons deliver the instructions from the brain to the rest of your body. The spinal cord, made of a bundle of nerves running up and down the spine, is similar to a superhighway, speeding messages to and from the brain at every second.

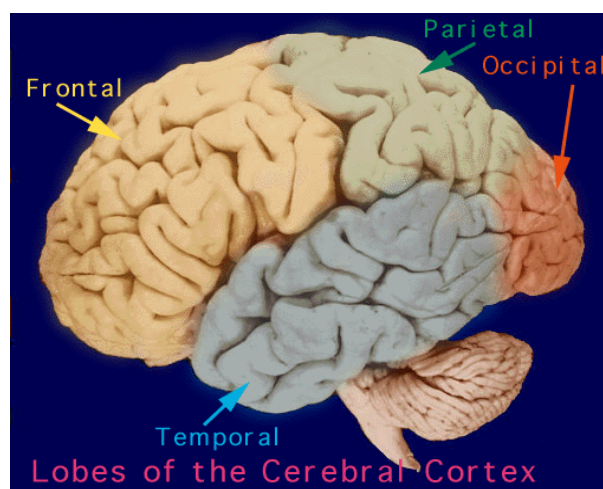
The brain is made of three main parts: the forebrain, midbrain, and hindbrain. **The forebrain** consists of the cerebrum, thalamus, and hypothalamus (part of the limbic system). **The midbrain** consists of the tectum and tegmentum. **The hindbrain** is made of the cerebellum, pons and medulla. Often the midbrain, pons, and medulla are referred to together as the brainstem.

---

---

**The Cerebrum:** The cerebrum or cortex is the largest part of the human brain, associated with higher brain function such as thought and action. The cerebral cortex is divided into four sections, called "lobes": the frontal lobe, parietal lobe, occipital lobe, and temporal lobe. Here is a visual representation of the cortex:

- Frontal Lobe- associated with reasoning, planning, parts of speech, movement, emotions, and problem solving
- Parietal Lobe- associated with movement, orientation, recognition, perception of stimuli



**Figure 4.4 Lobes of the Cerebral Cortex**

- Occipital Lobe- associated with visual processing
- Temporal Lobe- associated with perception and recognition of auditory stimuli, memory, and speech

The cerebral cortex is highly wrinkled. Essentially this makes the brain more efficient, because it can increase the surface area of the brain and the amount of neurons within it. A deep furrow divides the cerebrum into two halves, known as the left and right hemispheres. The two hemispheres look mostly symmetrical yet it has been shown that each side functions slightly different than the other. Sometimes the right hemisphere is associated with creativity and the left hemisphere is associated with logic abilities. The corpus callosum is a bundle of axons which connects these two hemispheres.

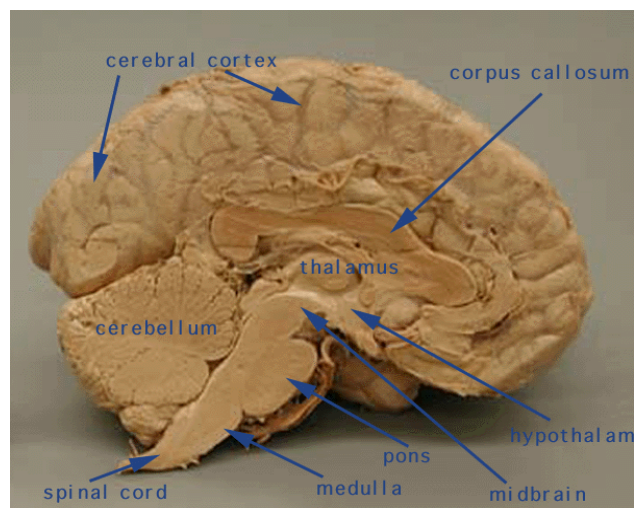
---

Nerve cells make up the gray surface of the cerebrum which is a little thicker than our thumb. White nerve fibers underneath carry signals between the nerve cells and other parts of the brain and body.

The neocortex occupies the bulk of the cerebrum. This is a six-layered structure of the cerebral cortex which is only found in mammals. It is thought that the neocortex is a recently evolved structure, and is associated with "higher" information processing by more fully evolved animals (such as humans, primates, dolphins, etc

**The Cerebellum:** The cerebellum, or "little brain", is similar to the cerebrum in that it has two hemispheres and has a highly folded surface or cortex. This structure is associated with regulation and coordination of movement, posture, and balance.

**Limbic System:** The limbic system, often referred to as the "emotional brain", is found buried within the cerebrum. Like the cerebellum, evolutionarily the structure is rather old. This system contains the thalamus, hypothalamus, amygdala, and hippocampus. Here is a visual representation of this system, from a midsagittal view of the human brain:



**Figure 4.5 Limbic System**

---



---

**Brain Stem:** Underneath the limbic system is the brain stem. This structure is responsible for basic vital life functions such as breathing, heartbeat, and blood pressure. The brain stem is made of the midbrain, pons, and medulla.

#### 4.4 Peak Signal-to-Noise ratio and Signal-to-Noise ratio

##### Peak Signal-to-Noise ratio

The phrase **peak signal-to-noise ratio**, often abbreviated **PSNR**, is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed in terms of the logarithmic decibel scale.

The PSNR is most commonly used as a measure of quality of reconstruction in image compression etc. It is most easily defined via the mean squared error (**MSE**) which for two  $m \times n$  monochrome images  $I$  and  $K$  where one of the images is considered a noisy approximation of the other is defined as:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \|I(i, j) - K(i, j)\|^2 \quad (4.1)$$

The PSNR is defined as:

$$PSNR = 10 \cdot \log_{10} \left( \frac{MAX_I^2}{MSE} \right) = 20 \cdot \log_{10} \left( \frac{MAX_I}{\sqrt{MSE}} \right) \quad (4.2)$$

Here,  $MAX_I$  is the maximum pixel value of the image. When the pixels are represented using 8 bits per sample, this is 255.

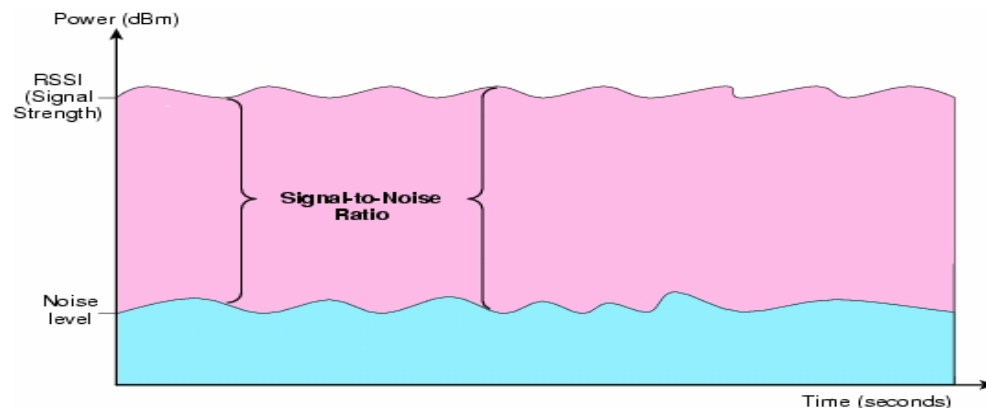
For color images with three RGB values per pixel, the definition of PSNR is the same except the MSE is the sum over all squared value differences divided by image size and by three. Typical values for the PSNR in image compression are between 20 and 40 dB.

---

---

## Signal-to-Noise ratio

**Basic Definition:** Signal to Noise Ratio is the range between the Noise Floor level and the Nominal Level.



**Figure 4.6 Representation to explain Signal to Noise Ratio**

Signal-to-noise ratio is an engineering term for the power ratio between a signal (meaningful information) and the background noise:

$$SNR = \frac{P_{signal}}{P_{noise}} = \left( \frac{A_{signal}}{A_{noise}} \right)^2 \quad (4.3)$$

Because many signals have a very wide dynamic range, SNRs are usually expressed in terms of the logarithmic decibel scale. In decibels, the SNR is 20 times the base-10 logarithm of the amplitude ratio, or 10 times the logarithm of the power ratio:

$$SNR(dB) = 10 \log_{10} \left( \frac{P_{signal}}{P_{noise}} \right) = 20 \log_{10} \left( \frac{A_{signal}}{A_{noise}} \right) \quad (4.4)$$

where P is average power and A is RMS amplitude. Both signal and noise power are measured within the system bandwidth.

---

---

In image processing, the SNR of an image is usually defined as the ratio of the mean pixel value to the standard deviation of the pixel values. Related measures are the "contrast ratio" and the "contrast-to-noise ratio".

#### **4.5 NOISE ARTIFACTS in ECGs and MRI Scan/Image**

- **Noise artifacts in ECGs**

In recent years the trend towards automated analysis of electrocardiograms has gained momentum. Many systems have been implemented in order to perform such tasks as 12-lead offline electrocardiogram analysis, Holter tape analysis in real-time patient monitoring. This requires accurate detection of various parameters of interest even in the presence of noise. For accurate detection however steps have to be taken to filter out or discard the noise. Filtering can alter the signal and may require substantial computational overhead.

Electrocardiographic signals (ECG) may be corrupted by various kinds of noise. Typical examples are:

1. Power line interference
2. Electrode contact noise.
3. Motion artifacts.
4. Muscle contraction.
5. Base line drift.
6. Instrumentation noise generated by electronic devices.
7. Electrosurgical noise.

- **Noise artifacts in MRI Image**

Grey values in MRI reflect two components: (a) signal intensity and (b) unavoidable noise. Noise is, in principle, unavoidable. It is caused by: (a) electromagnetic noise in

---

---

the body due to movement of charged particles and (b) small anomalies in the measurement electronics, which depends on (i) the size of the RF coil and (ii) the bandwidth of the pulse sequence. (Large coils have a large measurement field, but low SNR and vice versa. The closer the coil to the object, the stronger the signal – the smaller the volume, the higher the SNR. Wider bandwidths decrease SNR.)

Artifacts are signal intensities that have no relation to the spatial distribution of the tissues being imaged. There are four types of artifacts (based on appearance): (a) edge artifacts (ghosting, chemical shifts, and ringing), (b) distortions, (c) aliasing (wraparound) artifacts, and (d) flow artifacts.

Motion Artifacts (ghosting and smearing): Artifacts often result from involuntary movements (eg. respiration, cardiac motion and blood flow, eye movements and swallowing) and minor subject movements.

#### **4.6 Wavelet Shrinkage and Thresholding operators for de-noising**

Consider the problem of denoising an unknown time signal  $s$ , from a set of samples

$$x_i = s_i + n_i, \quad (4.5)$$

corrupted by a zero mean white Gaussian noise  $n_i$  ( $i = 1, \dots, N$ ). Let  $W$  denote a  $N$  by  $N$  orthonormal wavelet transformation matrix. In the wavelet domain, the above equation can be expressed as :

$$X_i = S_i + N_i, \quad \text{or,} \quad X = S + N \quad (4.6)$$

with  $X = Wx$ ,  $S = Ws$ , and  $N = Wn$  [23], [21].

Wavelet-domain filtering produces *wavelet shrinkage* estimates. That is, certain wavelet coefficients are reduced to zero. For a smooth function with additive Gaussian white noise of a specific energy level in a particular space of functions,

---

---

there exists a theoretical threshold that completely removed the noise and successfully reproduced the original smooth function [24].

Wavelet shrinkage is usually performed using one of two predominant thresholding schemes. The hard threshold filter  $Hh$  removes coefficients below a threshold value  $t_0$ , determined by the noise variance. This is sometimes referred to as the “keep or kill” method [25]. The soft threshold filter  $Hs$  shrinks the wavelet coefficients above and below the threshold. Soft thresholding reduces coefficients toward zero [21]. The process of denoising is necessarily lossy in that the denoised signal is irreversibly different than the noisy signal. Thresholding is the cause of this loss of information. It has been shown that if we desire the resulting signal to be smooth, the soft threshold filter should be used. However, the hard threshold filter performs better. Both methods result in error within a logarithmic factor of the ideal risk, a performance measure of the ideal shrinkage scheme [23]. Choosing a threshold value can also be difficult. In practical situations, where the noise-free signal is unknown, we seek an approximation of the signal that is smooth and fits the input well. A small threshold value creates a noisy result near the input, while a large threshold value introduces bias. The optimal threshold is somewhere in-between [22].

Experimental studies have shown that for certain applications, the optimal threshold is simply computed as a constant  $c$  times the noise variance [23]. The Universal method assigns a threshold level equal to the variance times  $(\sqrt{2\log(n)})$ , where  $n$  is the sample size [26]. Another approach utilizes Generalized Cross Validation (GCV), a function of the threshold value which is minimized to minimize the mean square error [22].

As a general rule, wavelet coefficients with larger magnitude are correlated with salient features in the image data. In that context, de-noising can be achieved by applying a thresholding operator to the wavelet coefficients (in the transform domain) followed by reconstruction of the signal to the original image (spatial) domain. Typical threshold operators for de-noising include :

---

*hard thresholding*:

$$\rho_T(x) = \begin{cases} x, & \text{if } |x| > T; \\ 0, & \text{if } |x| \leq T; \end{cases} \quad (4.7)$$

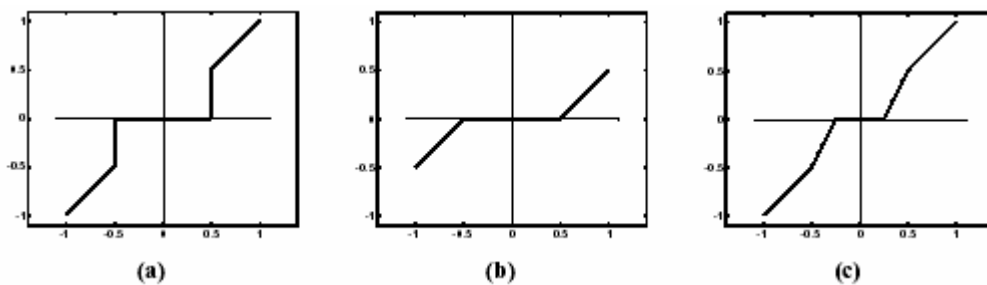
*soft thresholding* (wavelet shrinkage):

$$\rho_T(x) = \begin{cases} x - T, & \text{if } x \geq T \\ x + T, & \text{if } x \leq -T; \\ 0, & \text{if } |x| < T \end{cases} \quad (4.8)$$

and *affine(firm) thresholding*:

$$\rho_T(x) = \begin{cases} x, & \text{if } |x| \geq T \\ 2x + T, & \text{if } -T \leq x \leq -T/2 \\ 2x - T, & \text{if } T/2 \leq x \leq T \\ 0, & \text{if } |x| < T \end{cases} \quad (4.9)$$

The shapes of these thresholding operators are illustrated in Figure 4.8.



**Figure 4.7** Example of thresholding functions, assuming that the input data was normalized to the range of  $[-1,1]$ . (a) Hard thresholding. (b) Soft thresholding. (c) Affine thresholding. The threshold level was set to  $T=0.5$

## 4.7 Selection of Thresholding Method

In the application of De-noising, the threshold level parameter  $T$  plays an essential role. Values too small cannot effectively get rid of noise component, while values too large will eliminate useful signal components. There are a variety of ways to determine the threshold value  $T$  as we will discuss in this section.

---

Depending on whether or not the threshold value  $T$  changes across wavelet scales and spatial locations, the thresholding can be:

**4.7.1 Global Threshold:** a single value  $T$  is to be applied globally to all empirical wavelet coefficients at different scales.  $T = \text{const}$

**4.7.2 Level-Dependent Threshold:** a different threshold value  $T$  is selected for each wavelet analysis level (scale).  $T = T(j)$ ,  $j=1,2, \dots,J$ ,  $J$  is the coarsest level for wavelet expansion to be processed.

**4.7.3 Spatial Adaptive Threshold:** the threshold value  $T$  varies spatially depending on local properties of individual wavelet coefficients. Usually,  $T$  is also level-dependent.  $T = T_j(x, y, z)$

While a simple way of determining  $T$  is a percentage of coefficients maxima, there are different adaptive ways of assigning the  $T$  value according to the noise level (estimated via its variance  $\sigma$ ):

1. **Universal Threshold:**  $T = \sigma\sqrt{2\log n}$  {jin\_Coifman\_1995a}[27], with  $n$  equal to the sample size. This threshold was determined in an optimal context for soft thresholding with random Gaussian noise.

2. **Minimax Threshold:**  $T = \sigma T_n$  {jin\_Donoho\_1994a}[28], where  $T_n$  is determined by a minimax rule such that the maximum risk of estimation error across all locations of the data is minimized. This threshold level depends on the noise and signal relationships in the input data.

3. **Stein Unbiased Estimated of Risk (SURE):** Similar as minimax threshold but  $T_n$  is determined by a different risk rule {jin\_Donoho\_1995a; jin\_Stein\_1981}[29][30].

---

---

**4. Spatial Adaptive Threshold:**  $T = \sigma^2 / \sigma_x$  {jin\_Chang\_2000}[31], where  $\sigma_x$  is the local variance of the observation signal, which can be estimated using a local window moving across the image data or, more accurately, by a context-based clustering algorithm.

In many automatic de-noising methods to determine the threshold value  $T$ , an estimation of the noise variance  $\sigma$  is needed. Donoho and Johnstone {jin\_Donoho\_1995c}[32] proposed a robust estimation of noise level  $\sigma$  based on the median absolute value of the wavelet coefficients as:

$$\sigma = \frac{\text{median}(|W_1(x, y, z)|)}{0.6745}$$

where  $W_1$  is the most detailed level of wavelet coefficients. Such estimator has become very popular in practice and is widely used.

#### **4.8 Conclusion**

This chapter outlines the brief anatomy of brain and production of electrical signal associated with the heart. It also presents various causes of noise in ECG and MR images and need of thresholding operator for noise reduction.

---



---

# ***Chapter V***

---

---

## CHAPTER V

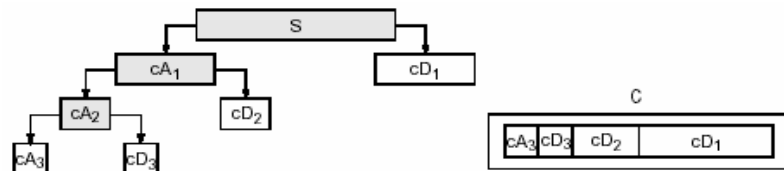
### ALGORITHM TO WORK WITH SIGNALS AND IMAGES IN MATLAB 7.0

Steps taken to de-noise the signal and image with MATLAB 7.0 is now explained in this chapter. We will also see the ease with which we can use the GUI of Wavelet transform for de-nosing, compression etc.

#### 5.1 Working with Signals

##### 5.1.1 Working in MATLAB Command line

1. Load a ECG signal
  2. Perform a single-level wavelet decomposition of a signal using the (Daubechies 1) db1 wavelet This generates the coefficients of the level 1 approximation (cA1) and detail (cD1).
  3. Construct approximations and details (A1 and D1) from the coefficients cA1 and cD1
  4. Display the approximation and detail
  5. Regenerate a signal by inverse wavelet transform  
If the single decomposition does not gives satisfactory results then proceed to next step otherwise jump to step 11
  6. Perform a multilevel wavelet decomposition of a signal  
Perform a level 3 decomposition of the signal (again using the db1 wavelet).  
The coefficients of all the components of a third-level decomposition (that is, the third-level approximation and the first three levels of detail) are returned concatenated into one vector, C. Vector L gives the lengths of each component.
-



**Figure 5.1 Level 3 Decomposition of signal**

7. Extract approximation and detail coefficients.
8. Reconstruct the level 3 approximation and the level 1, 2, and 3 details
9. Display the results of a multilevel decomposition
10. Reconstruct the original signal from the level 3 decomposition
11. Successive approximations become less and less noisy as more and more high-frequency information is filtered out of the signal. The level 3 approximation, A3, is quite clean as compare to original signal.

If result is satisfactory then stop otherwise perform next step

12. Remove noise by different thresholding methods and Compare the results.

Discarding all the high-frequency information, we've also lost many of the original signal's sharpest features. Optimal de-noising requires a more subtle approach called *thresholding*.

### 5.1.2 Working in GUI toolbox

- 1 Start the 1-D Wavelet Analysis Tool. From the MATLAB prompt, type

>> wavemenu

The **Wavelet Toolbox Main Menu** appears.

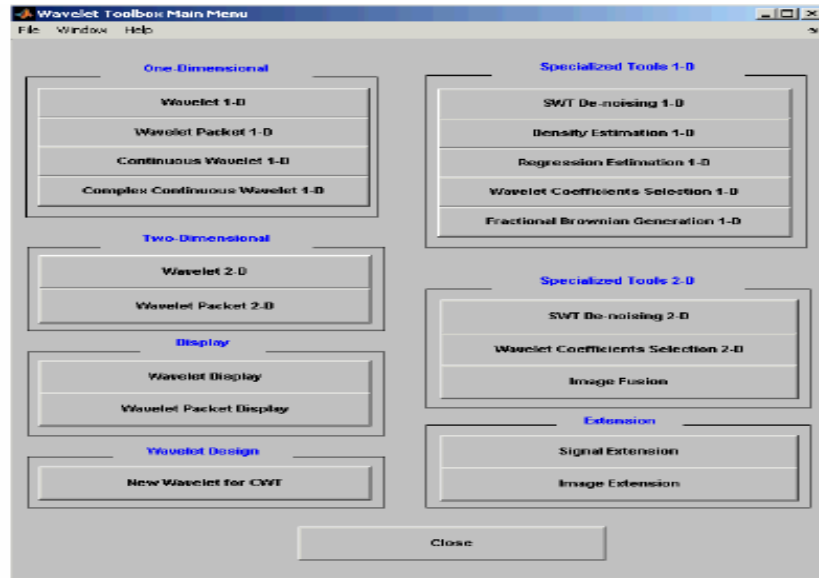


Figure 5.2 Wavelet Toolbox Main Menu

- 2 Click the **Wavelet 1-D** menu item.  
The discrete wavelet analysis tool for one-dimensional signal data appears (figure 5.3).

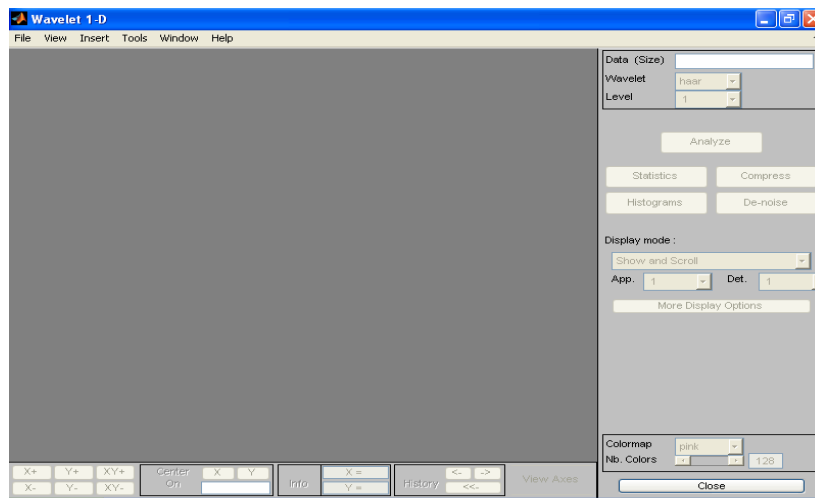
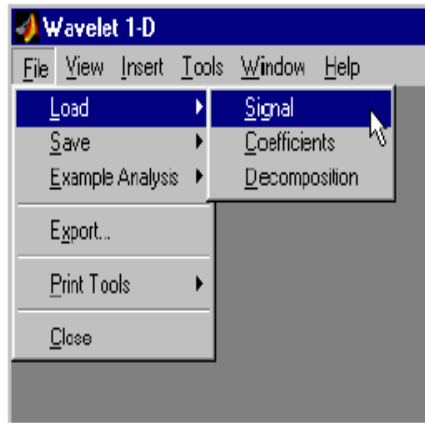


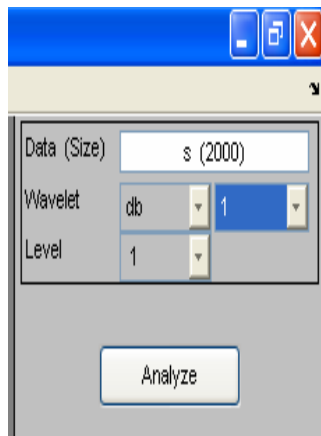
Figure 5.3 1D Wavelet Analysis Toolbox

- 3 Load a signal.  
From the **File** menu, choose the **Load>>Signal** option to load the ECG signal and click OK.



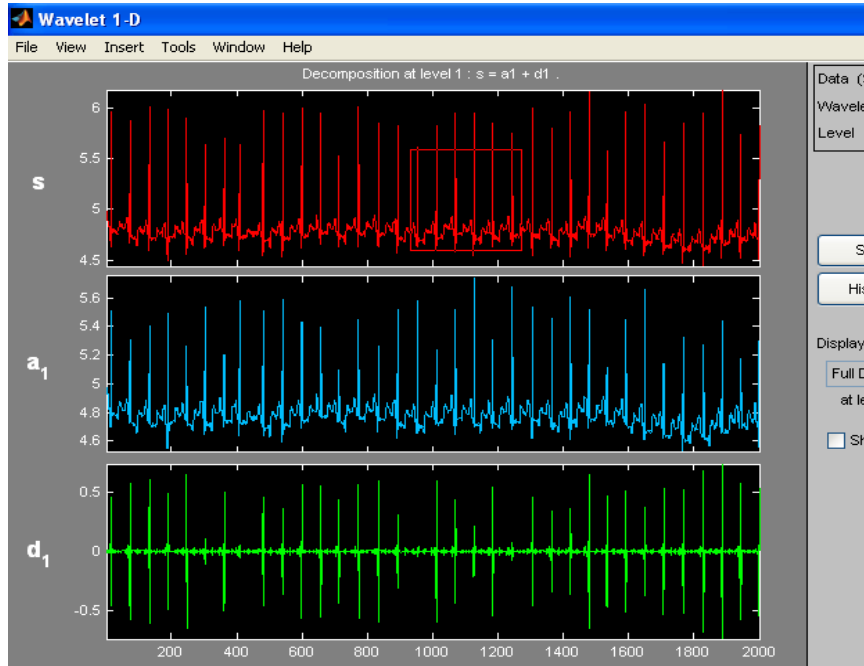
**Figure 5.4 Loading Signal in 1D Wavelet Analysis Toolbox**

- 4 To start the analysis, perform a single-level wavelet decomposition, using the db1 wavelet. In the upper right portion of the **Wavelet 1-D** tool, select the db1 wavelet and single-level decomposition as shown in figure 5.5.



**Figure 5.5 Selecting wavelet type and level of Decomposition in 1D wavelet Analysis Toolbox**

Click the **Analyze** button and the following window, shown in figure 5.6, appears.



**Figure 5.6** Window showing original signal, approximation ( $a_1$ ) and detail ( $d_1$ ) of level 1 Decomposition

Drag a rubber band box (by holding down the left mouse button) over the portion of the ECG signal we want to magnify and then from the zooming toolbar (Figure 5.7), located at the bottom of the screen, click the X+ button to zoom horizontally.



**Figure 5.7** Zooming toolbar

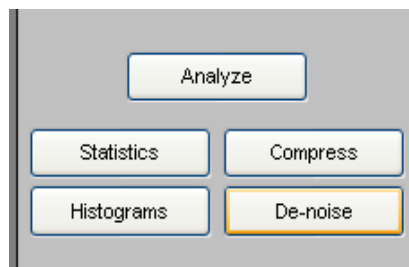
The **Wavelet 1-D** tool zooms all the displayed signals as shown in figure 5.8



**Figure 5.8** Zoomed version of original signal, approximation (a1) and detail (d1) of level 1

To decompose the signal with higher level of decomposition and with other wavelets e.g. db3, db4, bior etc repeat the step 4 by changing the level and wavelet type.

Click the **De-noise** button, located in the middle right of the window, underneath the **Analyze** button.

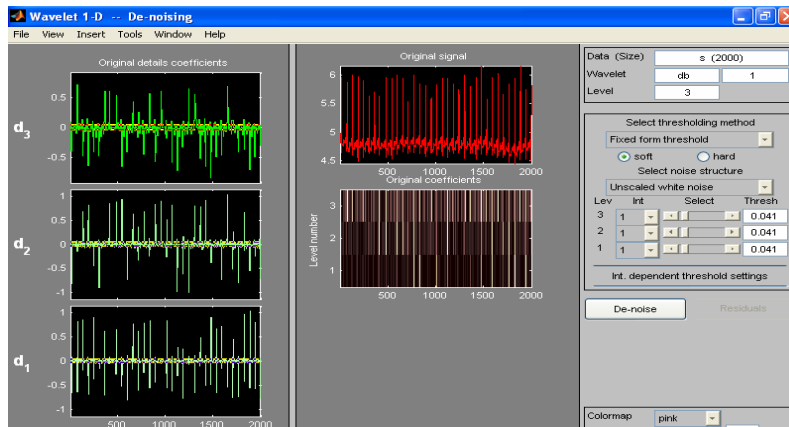


**Figure 5.9** Toolbar to select appropriate action

**Note :-** For Compression click the ‘Compress button’, located in the middle right of the window, underneath the Analyze button.

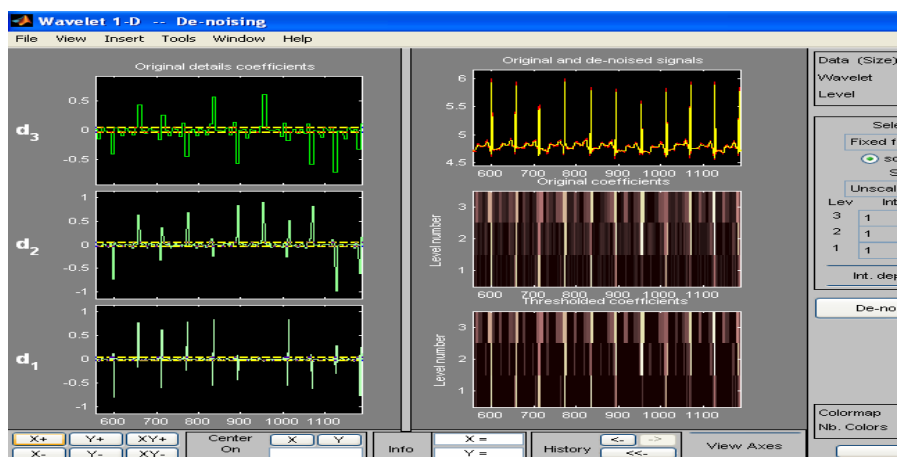
On clicking the de-noise button, window, shown in figure 5.10,

appears, showing the original signal and the detail component of the signal:



**Figure 5.10** Window showing Original signal and details of level 3 decomposition

Again click the **De-noise** button on the window, shown in figure 5.10, to de-noise the signal. **Zoom in** on the plot of the original and de-noised signals for a closer look. For this drag a rubber band box around the pertinent area, and then click the **XY+** button on zoom toolbar (figure 5.7). The **De-noise** window get magnified. By default, the original signal is shown in red, and the de-noised signal in yellow. The window shown in figure 5.11 appears after magnification:



**Figure 5.11** Window showing Original signal, De-noise signal and details of level 3



---

Note :- Click SWT De-noising 1D menu item from wavelet toolbox main menu to de-noise an signal using discrete SWT and the procedure is the same as adopted for de-noising using DWT.

## 5.2 Working with Images

### 5.2.1 Working in MATLAB Command line

1. Load an image.
2. Display the image.
3. Convert an indexed image to a grayscale image.  
If the colormap is smooth, the wavelet transform can be directly applied to the indexed image; otherwise the indexed image should be converted to grayscale format.
4. Perform a multilevel wavelet decomposition. Chose db1 wavelet and Level 2 decomposition.  
This generates the coefficient matrices of the level-two approximation and horizontal, vertical and diagonal details.
5. Reconstruct Level 2 approximations and level 1 and level 2 details from the coefficients.
6. Display the results of a multilevel decomposition.
7. Regenerate or reconstruct an image by Inverse Wavelet Transform.
8. Remove noise by thresholding

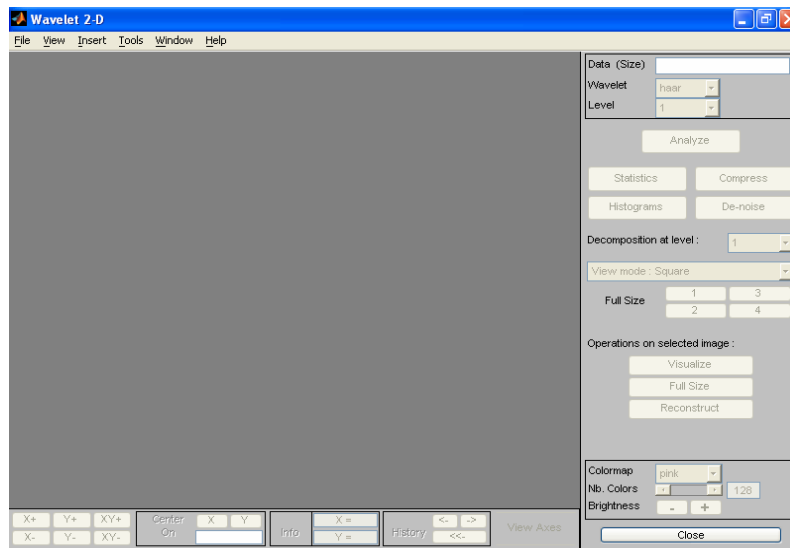
### 5.2.2 Working in GUI toolbox

1. Start the 2-D Wavelet Analysis Tool.  
From the MATLAB prompt, type  
  
`>> wavemenu`

---

The same **Wavelet Tool Main Menu** appears as discussed in section 5.1.2 (figure 5.2)

Click the **Wavelet 2-D** menu item. The discrete wavelet analysis tool for two-dimensional image data appears (figure 5.12).



**Figure 5.12 2D Wavelet analysis toolbox**

2. Load an image.

From the **File** menu, choose the **Load>>Image** option to load the MRI image of the Brain. Click the **OK** button.

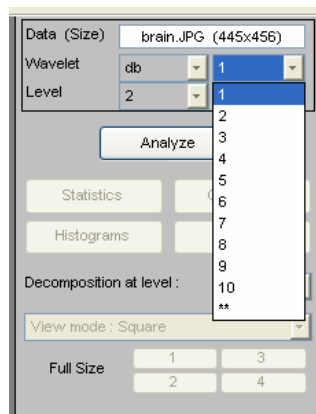


**Figure 5.13 Loading Image in 2D Wavelet analysis toolbox**

---

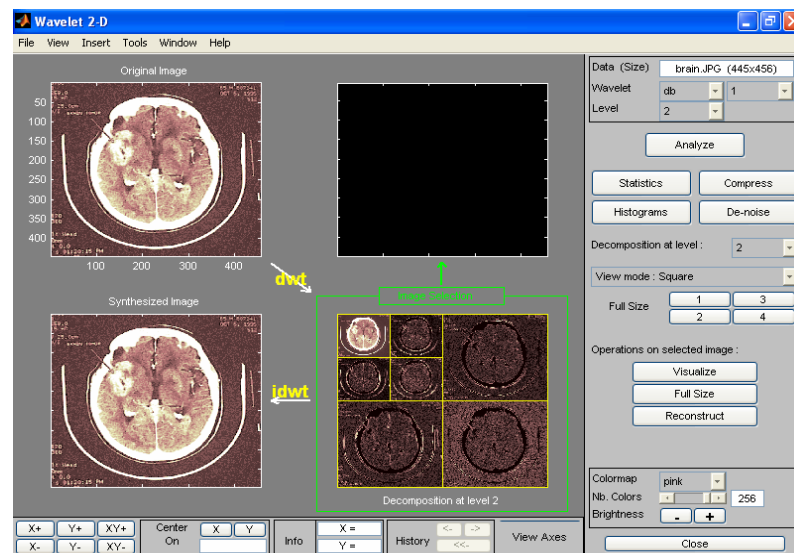
3. Analyze the image.

Using the Wavelet and Level menus located to the upper right, select the wavelet type, and the number of levels to be used for the analysis. For this analysis, select the db1 wavelet at level 2 .



**Figure 5.14** Selecting wavelet type and level of Decomposition in 2D Wavelet Analysis Toolbox.

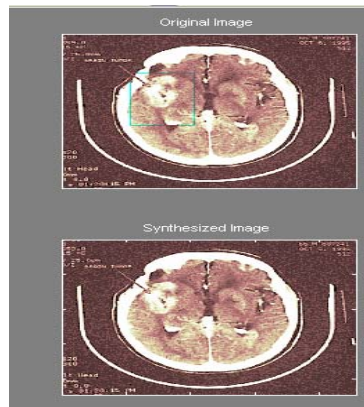
Click the **Analyze** button. The window shown in figure 5.15 appears:



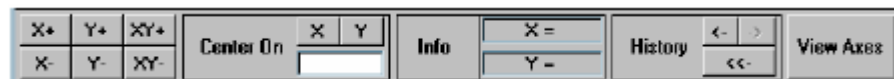
**Figure 5.15** Window showing Original Image, Synthesise Image, and Decomposition (Approximation and Detail) at level 2

The display in figure 5.15 is default **Square Mode Features** which includes four different displays. In the upper left is the original image. Below that is the image reconstructed from the various approximations and details. To the lower right is a decomposition showing the coarsest approximation coefficients and all the horizontal, diagonal, and vertical detail coefficients. Finally, the visualization space at the top right displays any component of the analysis that we want to look at more closely on clicking on any decomposition component in the lower right window. A green border highlights the selected component.

**Zooming in on Detail.** Drag a rubber band box (by holding down the left mouse button) over the portion of the image we want to magnify (as shown in figure 5.16). Then from zooming toolbar, shown in figure 5.17, Click the **XY+** button (located at the bottom of the screen) to zoom horizontally and vertically.

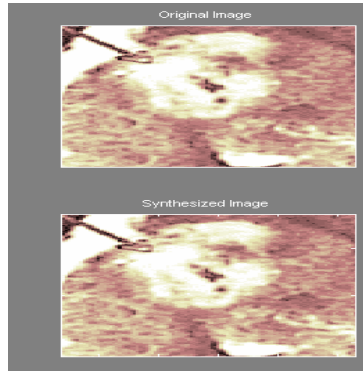


**Figure 5.16** Zooming the region of Interest in Original Image



**Figure 5.17** Zoom Toolbar

The **Wavelet 2-D** tool enlarges the displayed images (shown in figure 5.18).

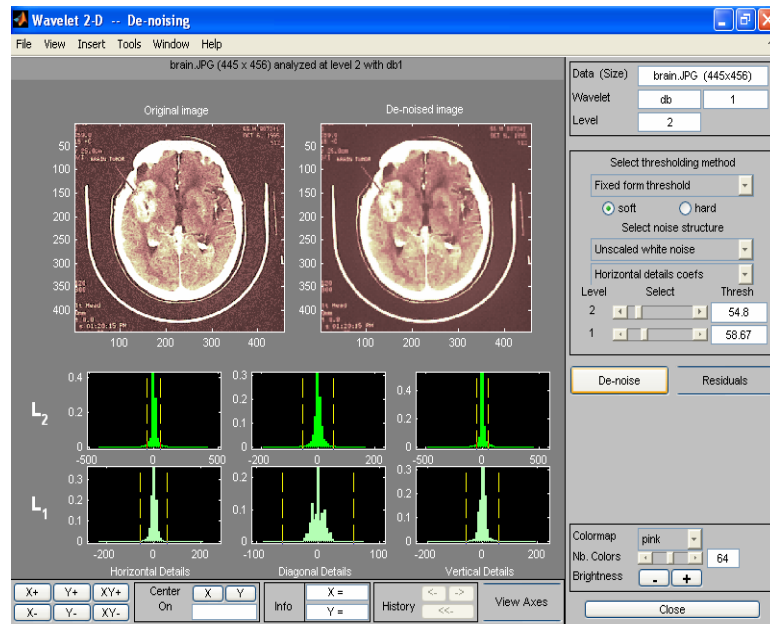


**Figure 5.18** Zoomed version of original and synthesized image

To zoom back to original magnification, click the **History** <<- button.

**4.** De-Noise the image

Click the **De-noise** button located to the upper right of the **Wavelet 2-D** window. The **Wavelet 2-D De-Noising** window appears and again click the De-Noising button located on **Wavelet 2-D De-Noising** window. The following window appears.



**Figure 5.19** Window showing Original and De-noised Image, with level 1 and 2 Horizontal, vertical and diagonal detail

---

Select different thresholding method from 2D-Denoising window (figure 5.19) for better results.

Note:- Click SWT De-noising 2D menu item from wavelet toolbox main menu to de-noise an image using discrete SWT and the procedure is the same as adopted for de-noising using DWT.

### **5.3 Conclusion**

The algorithm to de-noise the signal and image with MATLAB 7.0 and the ease with which we can use the GUI of wavelet transform is explained in this chapter.

---

# ***Chapter VI***

---

---

## CHAPTER VI

### RESULTS AND DISCUSSION

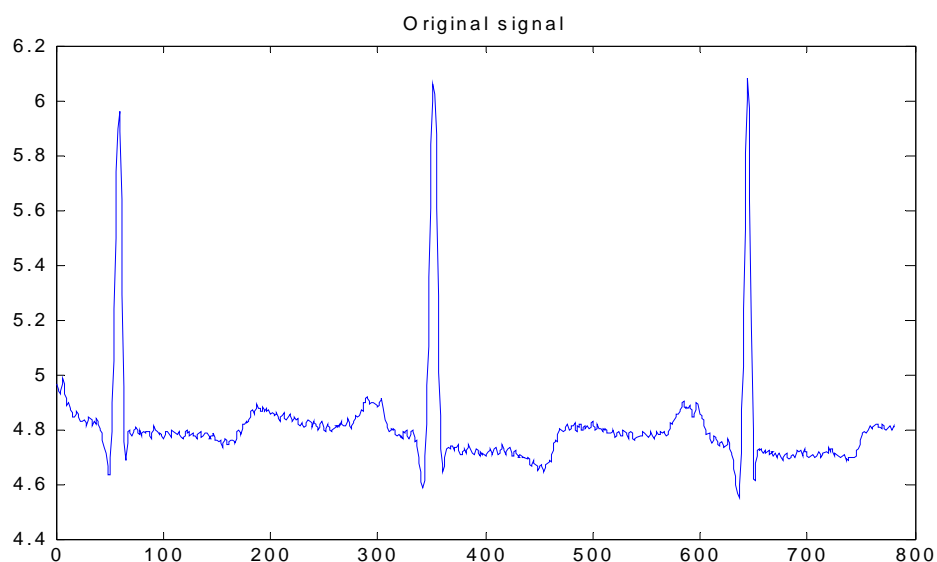
The experiments are conducted on MRI image of size 512 X 512 at different noise levels  $\sigma = 25, 30$  and  $35$  and on ECG signal using different threshold method. To evaluate the quality of the results, SNR (Signal to noise Ratio) is considered for ECG signal and PSNR (Peak Signal to Noise Ratio) for MRI image.

#### I ECG Signal De-noising

The DWT and SWT transform employs db1 wavelet with level 3 decomposition. For ECG signal de-noising three kinds of threshold method, viz. Sqtwolog, minimaxi, heursure, are used with DWT and the results are then compared

##### (A) DWT - Discrete wavelet transform method

The original ECG signal on which wavelet transform is required to be performed is shown below in figure 6.1.



**Figure 6.1 Original ECG Signal**

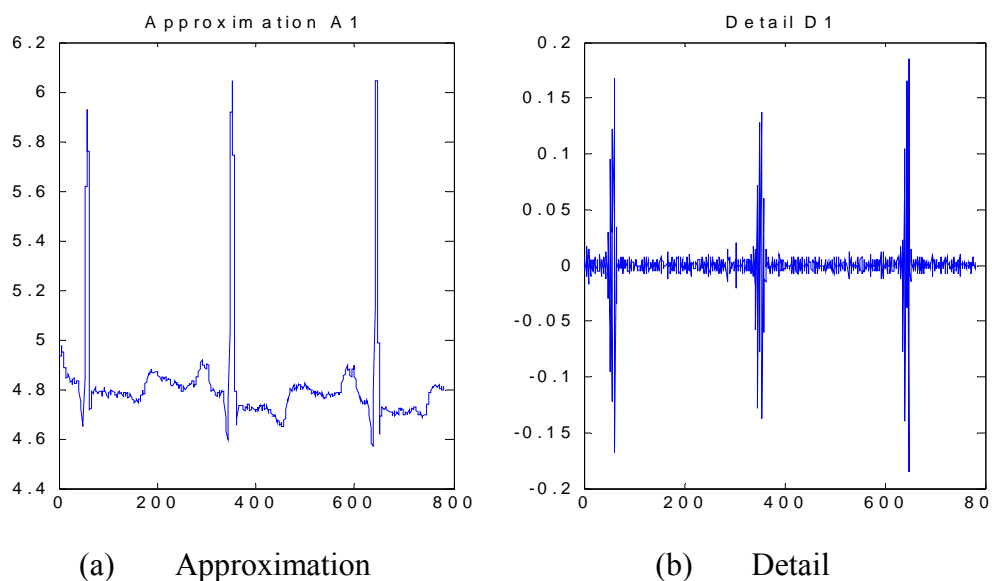
---



---

On loading an ECG signal and performing a single-level wavelet decomposition on a signal using the db1 wavelet it generates the coefficients of the level 1 approximation (cA1) and detail (cD1).

On constructing approximations and details (A1 and D1) from the coefficients cA1 and cD1 we get approximation and detail as shown in figure 6.2(a) and figure 6.2(b) respectively.



**Figure 6.2** Approximation and Detail after single-level decomposition using DWT

The signal can be reconstructed by either

$$P1=A1+D1;$$

or taking idwt i.e.

$$A0 = \text{idwt}(cA1,cD1,db1,1_s);$$

Where P1 and A0 is reconstructed signal

To check the perfect reconstruction by A0 we get error:

$$\text{err} = 2.6645e-015$$

---

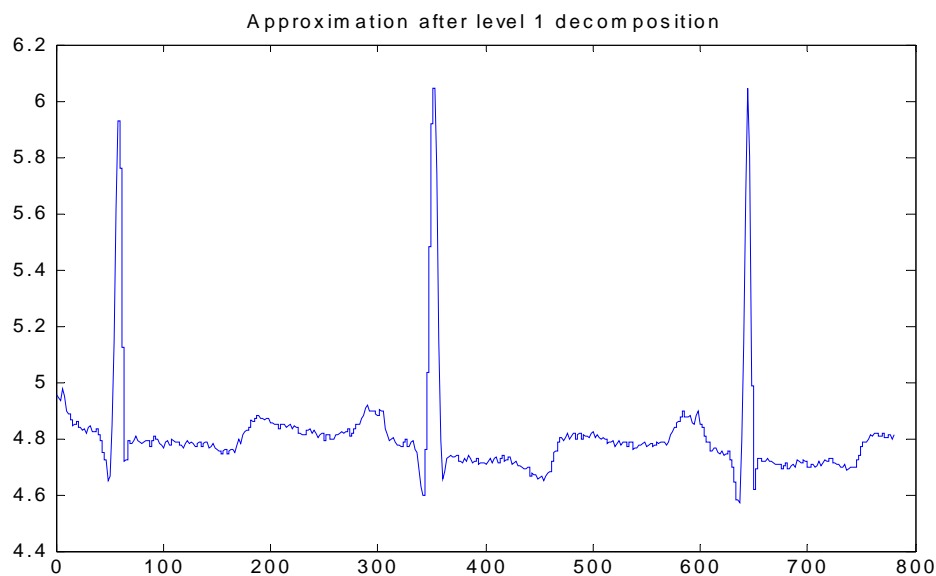
---

and

To check the perfect reconstruction by P1 we get error:

err =2.6645e-015

Where the error (err) is the difference between the original and reconstructed signal which comes out to be very less and thus reconstruction is perfect. As both P1 and A1 give the same result therefore any method can be opted for reconstruction. Approximation A1 after Level 1 is shown in figure 6.3 which is same as figure 6.2(a).

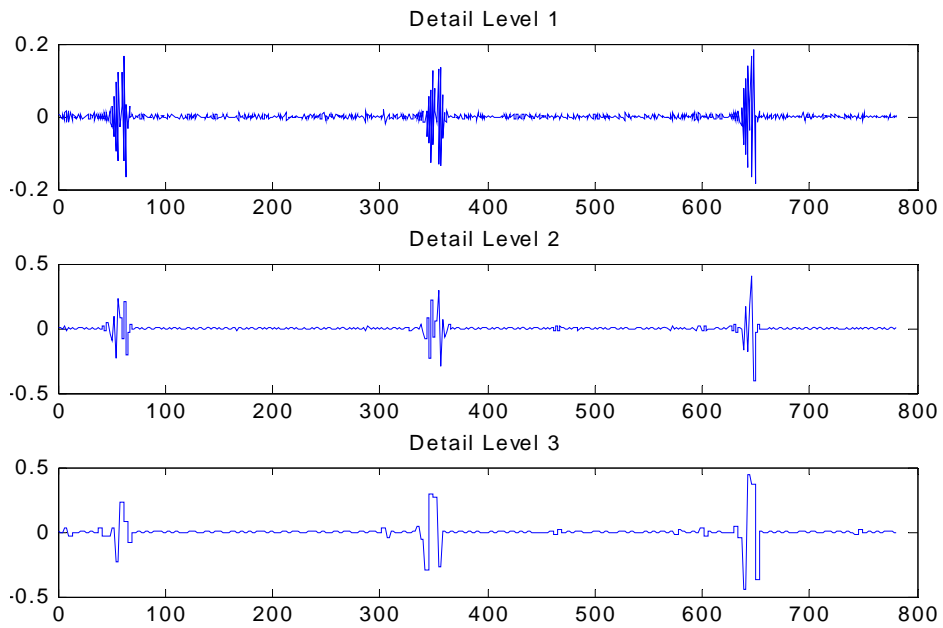


**Figure 6.3 Approximation A1 after level 1 decomposition using DWT**

Figure 6.3 shows that from approximation A1 high frequency signal is somewhat removed. As we have discussed in previous chapters that the details are the high frequency component of the signal therefore these are of less interest. We now note that successive approximations become less and less noisy as more and more high-frequency information is filtered out of the signal.

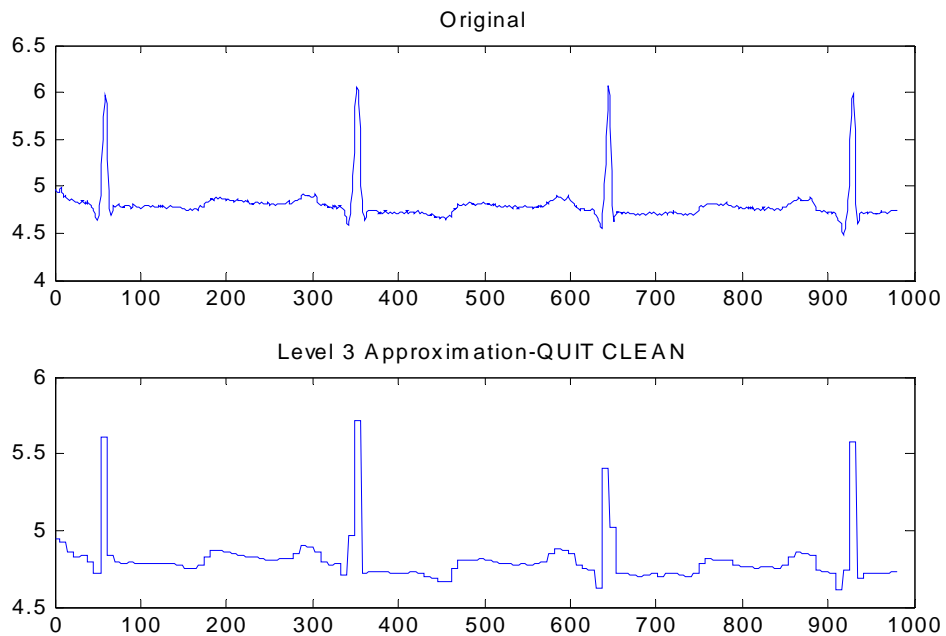
On performing a multilevel wavelet decomposition (level 3 decomposition in this case) of the signal using the db1 wavelet, figure 6.4 shows the details D1, D2, and D3, which is nothing but the high frequency component after each iteration or decomposition level.

---



**Figure 6.4** Details (D1, D2 & D3) after level 3 decomposition using DWT

The level 3 approximation, A3, is quite clean, as shown in figure 6.5, as compared to the original signal but changes the fine details of the signal



**Figure 6.5** Original and level 3 approximation using DWT

---

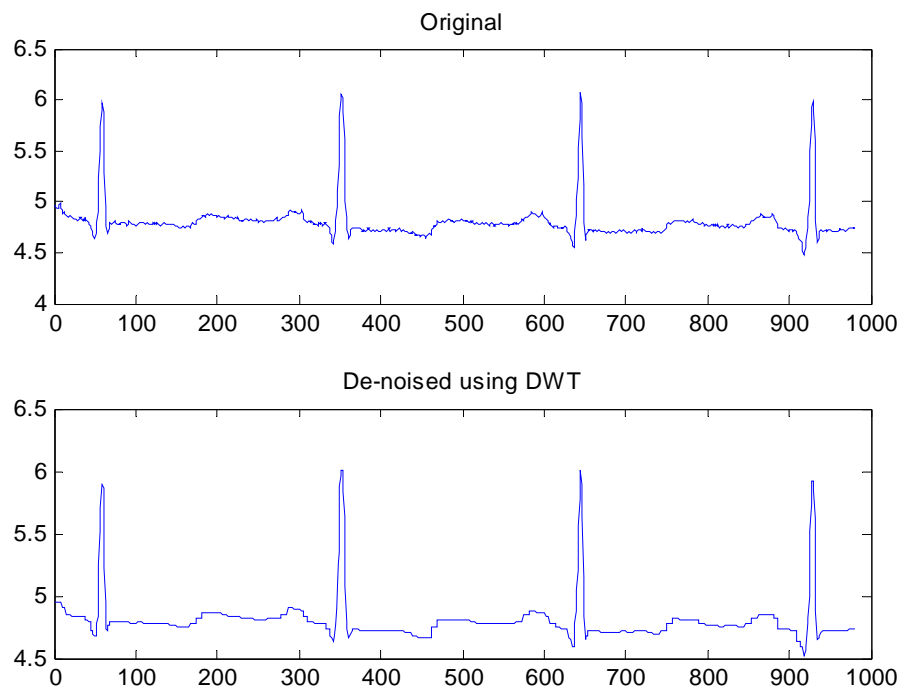
Of course, in discarding all the high-frequency information, we've also lost many of the signal's sharpest features. Optimal de-noising requires a more subtle approach called *thresholding*. This involves discarding only the portion of the details that exceeds a certain limit.

Removing noise by thresholding using DWT, the default values of threshold, generated by Matlab command, is:

```
thr =0.0450
```

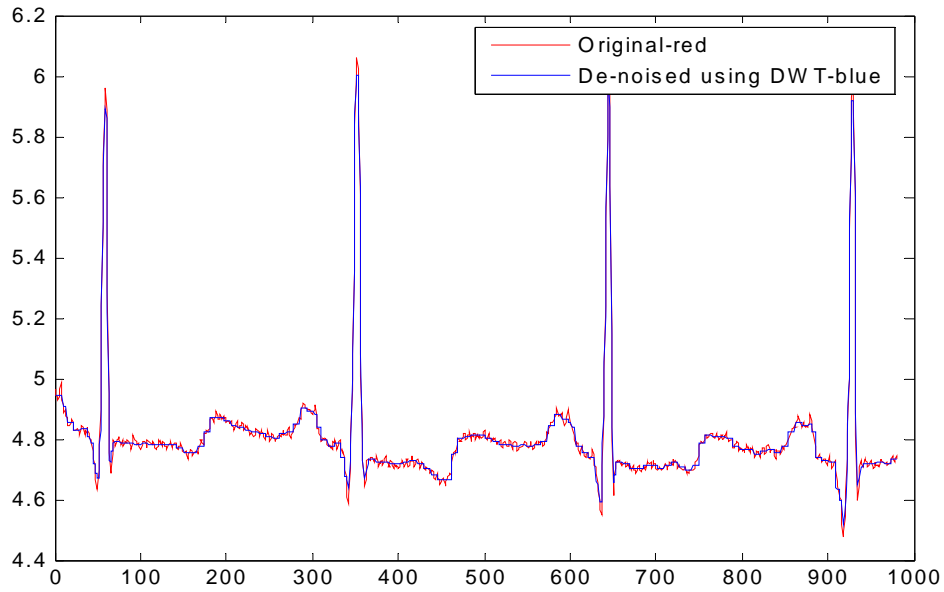
The original and de-noised signals using DWT with thresholding method are shown in figure 6.6. The figure 6.7 shows overlap original signal ( in red ) and De-noised signal (in blue ) for comparison purpose. It shows that de-noised signal preserves the sharp features.

The result after applying thresholding (figure 6.6) are better as compared to decomposition method (approximation of level 3 (figure 6.5))



**Figure 6.6 Original and De-noised ECG Signal using DWT**

---

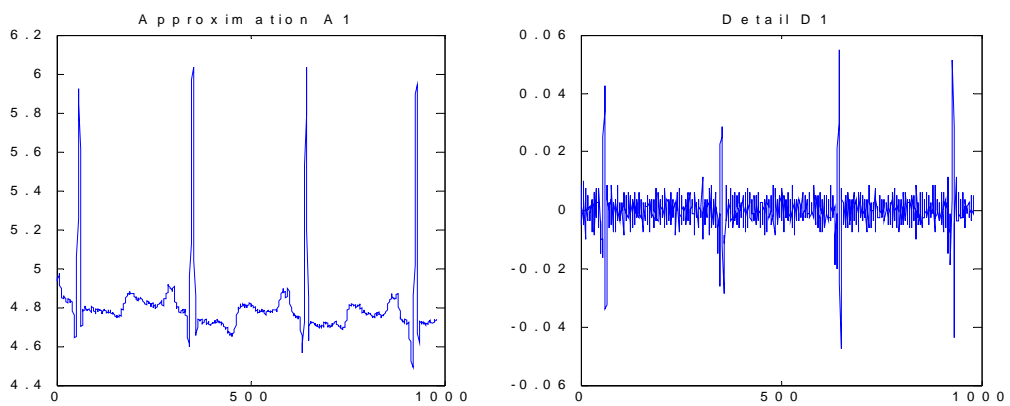


**Figure 6.7** Overlap Original and De-noised signal (using DWT)

Signal to noise ratio with DWT is 24.2127

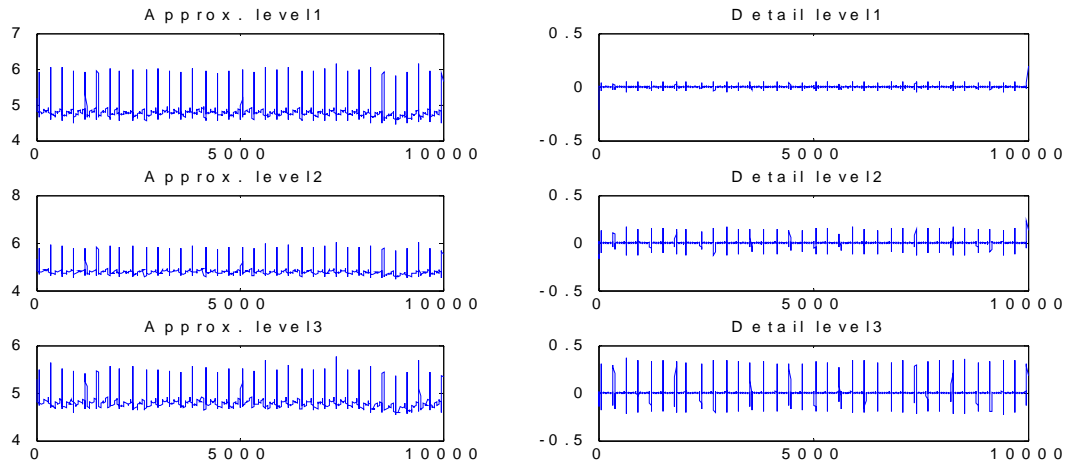
**(B) SWT - Stationary Wavelet Transform**

Performing single-level decomposition on a signal using the db1 wavelet the same way as we did in DWT, SWT also generates the coefficients of the level 1 approximation (swa) and detail (swd) and on constructing approximations and details (A1 and D1) from the coefficients swa and swd we get the following approximation and detail as shown in figure 6.8



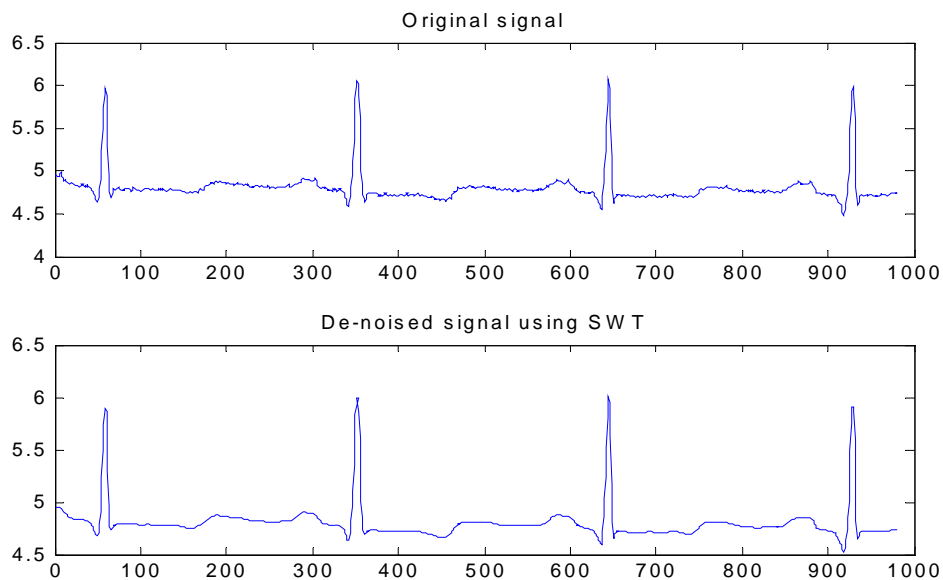
**Figure 6.8** Approximation and Detail after single-level decomposition using SWT

In this case also level 1 decomposition is not sufficient for removing the noise, thus on performing a multilevel wavelet decomposition (level 3 decomposition in this case also) of the signal using the db1 wavelet, approximations, A1, A2 and A3, and details D1, D2 and D3 are shown below in figure 6.9



**Figure 6.9 Approximations (A1, A2& A3) and Details (D1,D2& D3) after level 3 decomposition using SWT**

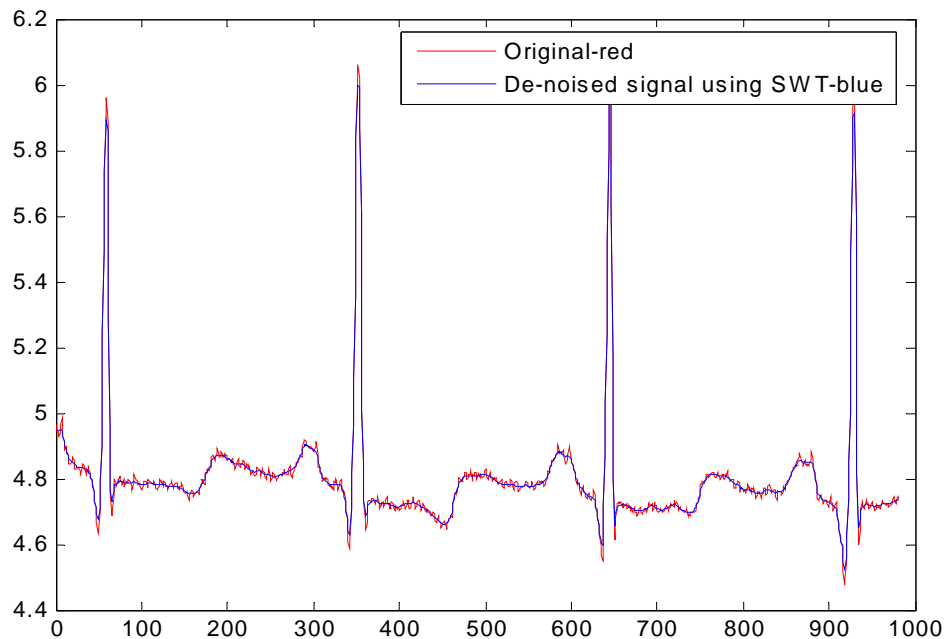
The original signal is reconstructed from approximation and detail and then de-noising the signal using the same threshold value as we used with DWT, figure 6.10 is generated and shows that SWT not only preserves the fine details but is also gives smooth signal as compared to DWT thus SWT is better then DWT .



**Figure 6.10 Original and De-noised ECG Signal using SWT**

---

Figure 6.11 shows the original signal (in red ) and De-noised signal using SWT (in blue ). It shows that the de-noise signal is smooth and SWT preserves the sharp details.



**Figure 6.11 Overlap Original and De-noised signal (using SWT)**

Signal to noise ratio with SWT =27.5799

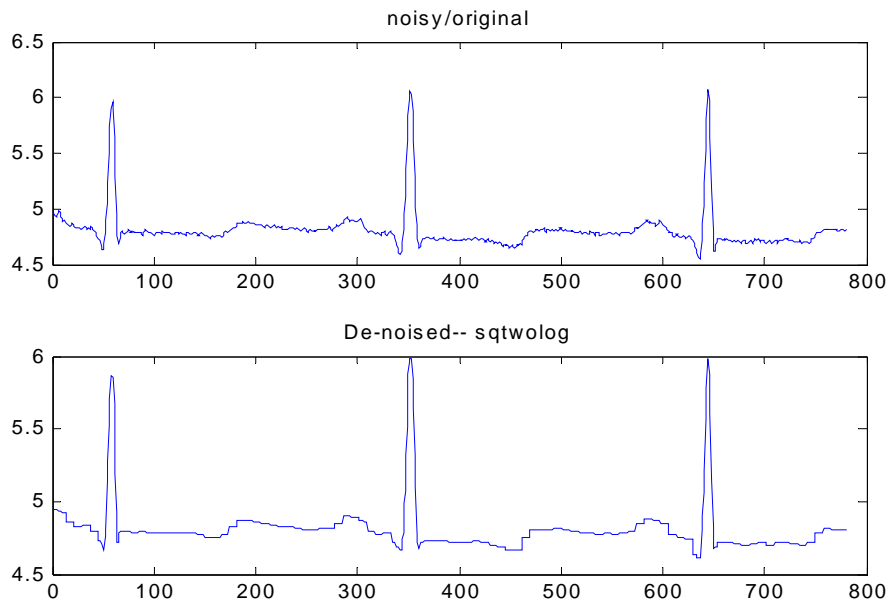
Comparing the results of DWT and SWT, SNR of SWT is better than DWT and thus the SWT remove the noise better than DWT without compromising the details.

De-noising of ECG signal with DWT and SWT with db3 wavelet is also performed and resulting SNR with db1 and db3 is summarized in table 6.1. From table 6.1 it is concluded that the SWT is better option to de-Noise the signal as compare to DWT.

The DWT is now employed with different thresholding method viz. heursure, sqtwolog and minimaxi using db1 wavelet and the results are shown in Figure 6.12, 6.13 and 6.14

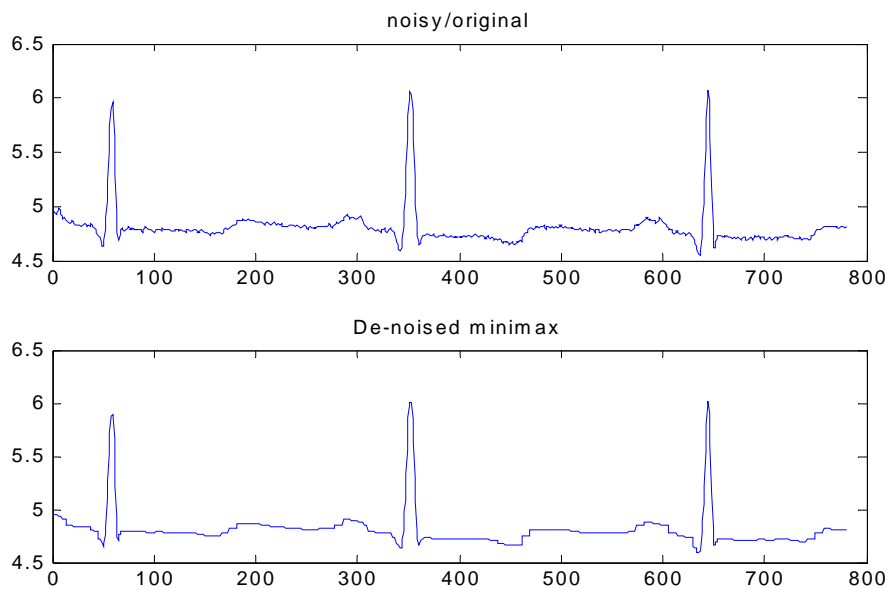
(1) sqtwolog: It gives S/N ratio = 22.6100

---



**Figure 6.12 Original and De-noised Signal using sqtwolog threshold**

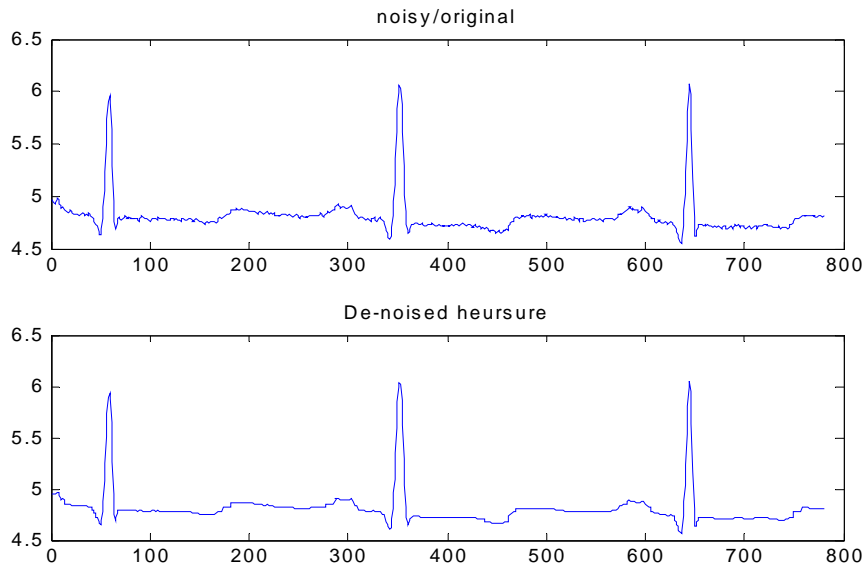
(2) minimaxi : Gives S/N ratio = 24.4473



**Figure 6.13 Original and De-noised Signal using minimaxi threshold**

(3) heursure: It gives S/N ratio = 28.5703





**Figure 6.14** Original and De-noised Signal using heursure threshold

From above figures and SNRs, it is concluded that heursure thresholding method gives better results. Therefore de-noising the signal using DWT with heursure thresholding method is better as it not only produces the smooth de-noised signal but also it does not changes the details of the signal

Selecting a particular wavelet for de-noising has also a key role. The above methods is performed with db1 wavelet. The same procedure is opted for db3 wavelet for performing the de-noising and the result is summarized in Table 6.1

**Table 6.1.** SNR for the ECG signal with DWT and SWT and with different thresholding method: Sqtwolog, minimaxi, heursure using DWT

Wavelet	DWT Thr.=0.0450	DWT			SWT Thr.=0.0450
		Thresholding			
		Sqtwolog	minimaxi	heursure	
db1	24.2127	22.6100	24.4473	28.5703	27.5799
db3	25.6854	24.2531	26.0383	28.6488	27.8702

In ECG signal denoising, in present noise level among all threshold method heursure is best and the worst is Sqtwolog, (Table 6.1). The results of experiment shows that

---

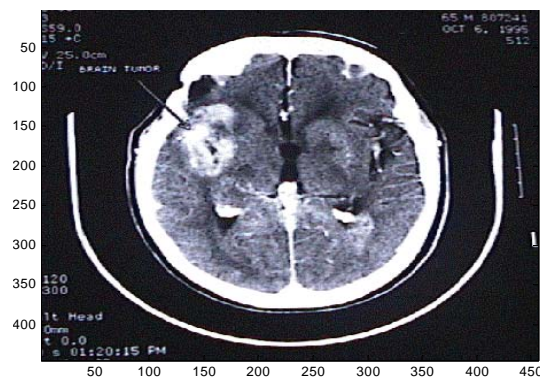
db3 gives better result than db1. Table 6.1 also shows that SWT gives better result as compared to DWT for same value of threshold.

## II Image Denoising

In order to compare the performance of DWT with SWT, the experiments are conducted on MRI image of size 512 X 512 at different noise levels  $\sigma = 25, 30$  and  $35$

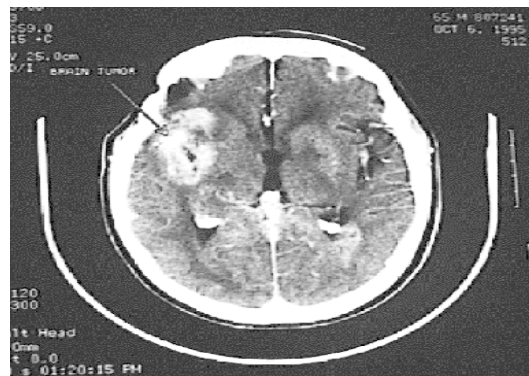
### (A) DWT

The Original RGB MRI image is shown in figure 6.15 :



**Figure 6.15 Original RGB MR image**

On loading an image, we first convert it into an grayscale image before we perform the decomposition. If the colormap is smooth then we don't need to convert it into gray scale. The gray scale MR image is shown in figure 6.16:

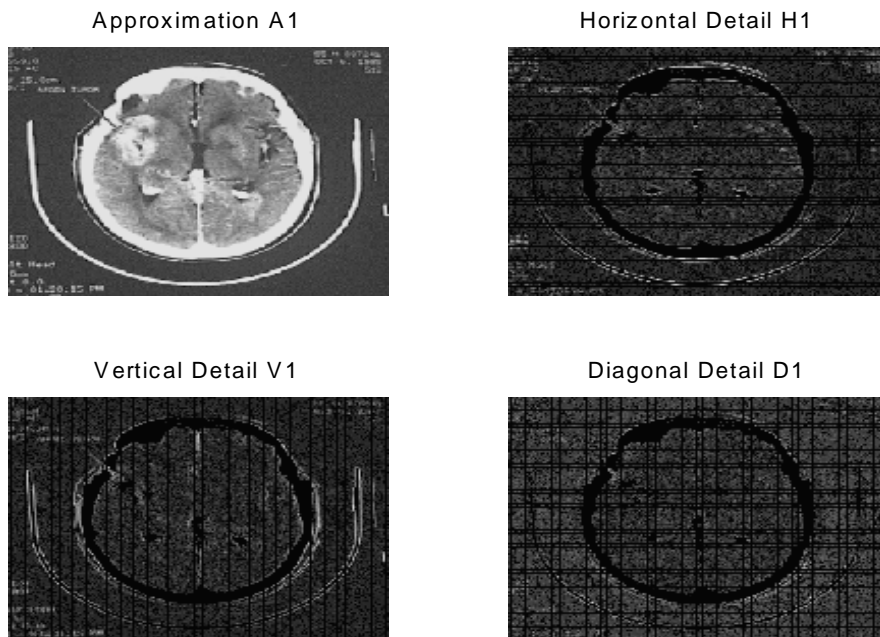


**Figure 6.16 Gray scale MR image**

---

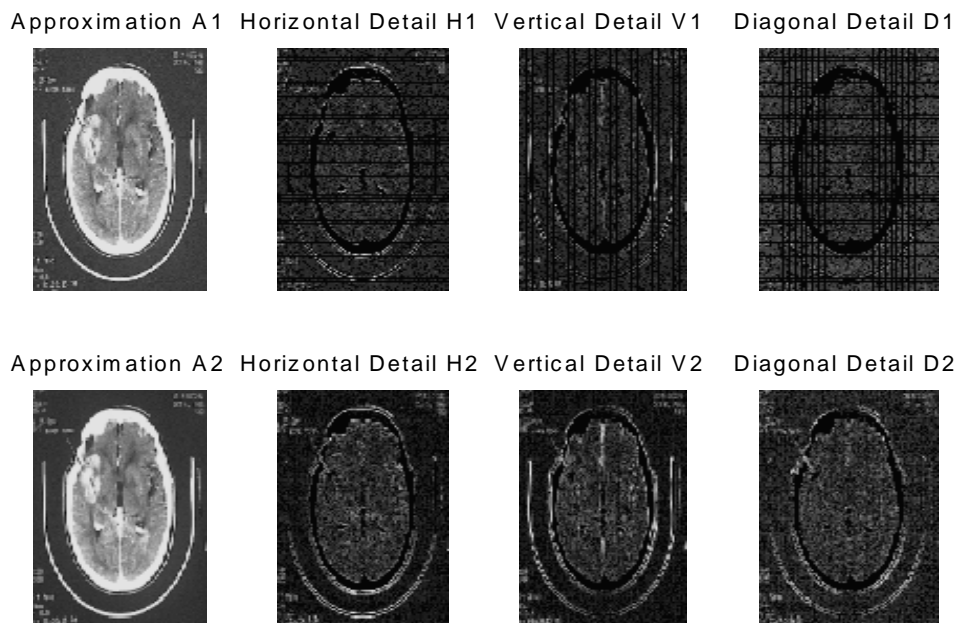
---

On performing single-level decomposition using db1 the approximation and details are shown in figure 6.17:



**Figure 6.17** Approximation and details after single-level decomposition using DWT

Approximation and details after level 2 decomposition is shown in figure 6.18:



**Figure 6.18** Approximation and details after level 2 decomposition using DWT

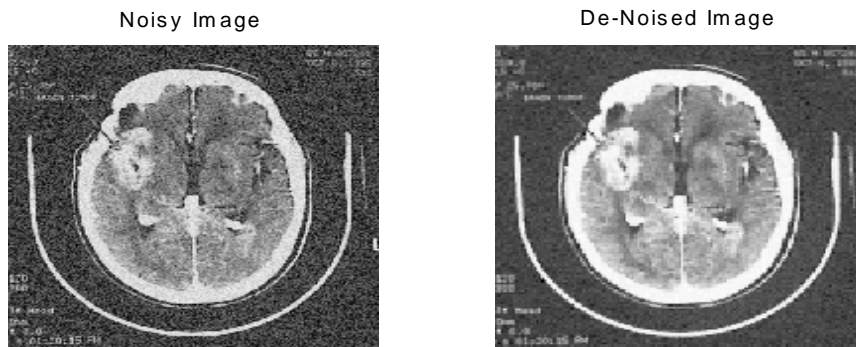
---

---

Original Image is presently is not corrupted with noise, thus we first introduce the noise with noise variance 25 before performing de-noising using DWT and SWT with decomposition level 2.

De-noising the image with DWT with db1 wavelet give PSNR =21.2317

The resultant De-noised image is shown in figure 6.19

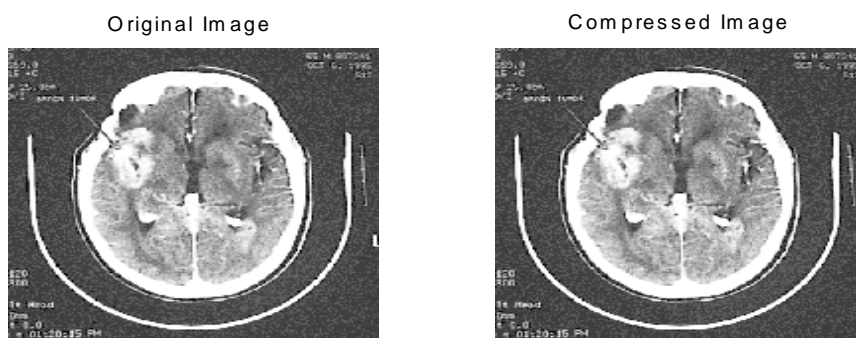


**Figure 6.19** Noisy and De-noised MR image using DWT

The same experiment is also performed with db3 wavelet and also with different noise variance and the result is summerised in Table 6.2

### **Image Compression**

Image data compression technique is concerned with the reduction of the number of bits required to store or transmit images without any appreciable loss in information. Thus we can send image in short time and store the compressed image with ease. The image shown in figure 6.20 is 99.9982 % compressed without loss in information.



**Figure 6.20** Original and Compressed MR image

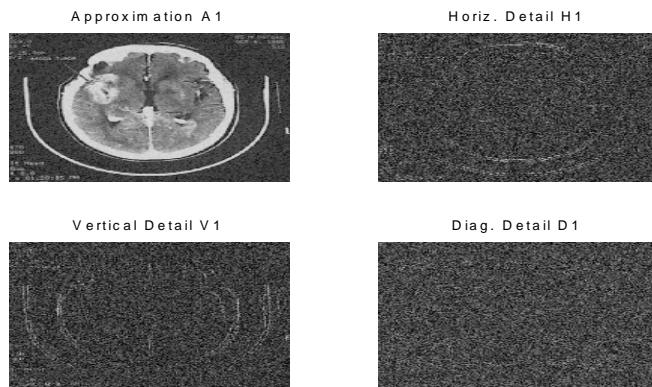
---

---

As explained before **Data compression** is the process of reducing the amount of data by reducing the number of bits required to represent a given quantity of information so in this process those bits or pixels which may be corrupted with a noise is also been eliminated. When compressed image is used for de-noising the PSNR = 21.2524 with db1 and PSNR = 22.4220 with db3 at noise variance =25. Thus when compressed image is used directly for de-Noiseing then the PSNR ratio become somewhat better.

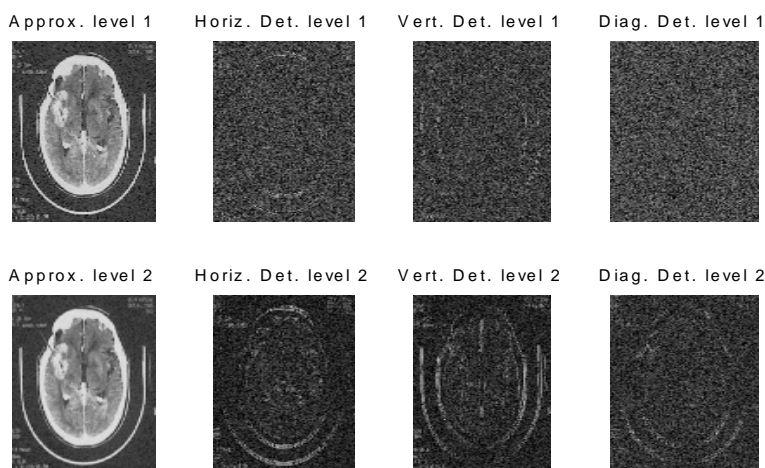
**(B) SWT**

On performing single-level decomposition using db1 we get the Approximation and details shown in Figure 6.21



**Figure 6.21** Approximation and details after single-level decomposition using SWT

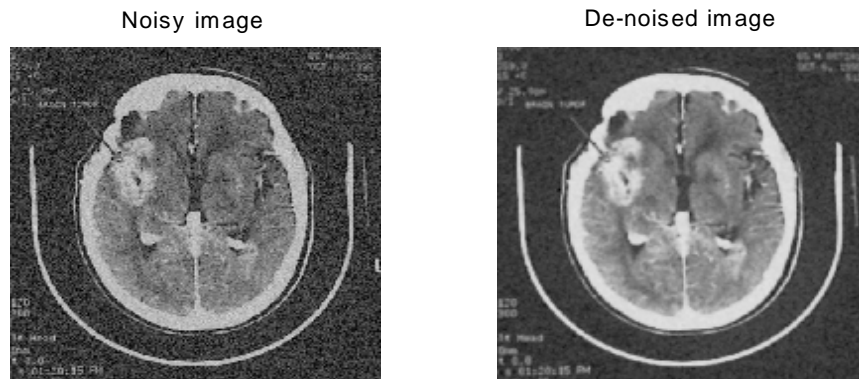
Approximation and details after level 2 decomposition is shown below:



**Figure 6.22** Approximation(A1 & A2) and details (D1 & D2) after level 2 decomposition using SWT

---

On performing de-noising using SWT figure 6.23 is generated.



**Figure 6.23** Noisy and De-noised MR image using SWT

and it gives PSNR =23.4897 at noise variance = 25

The above experiment is also performed with db3 wavelet and also with different noise variances and the result (PSNR) is summarized and compared in table 6.2. It appears from the table 6.2 that the SWT eliminate the noise better as compared to DWT at all noise level. The results of experiment show that db3 gives better results than db1 at particular noise variance.

**Table 6.2** PSNR for the MR Image with different noise variance  $\sigma$  using db1 and db2 wavelet

		<b>DWT</b>	<b>SWT</b>
<b>db1</b>	$\sigma =25$	21.2317	23.4897
	$\sigma =30$	20.7702	23.0581
	$\sigma =35$	20.3759	22.6910
<b>db3</b>	$\sigma =25$	22.3713	23.5017
	$\sigma =30$	21.9347	23.1067
	$\sigma =35$	21.5487	22.7569

---

### III Region of Interest

In order to select the region of interest in the Original image CROPPING is performed. Region of interest in MRI of Brain may be a tumorous growth.

Two Methods are adopted for selecting Region of Interest: -

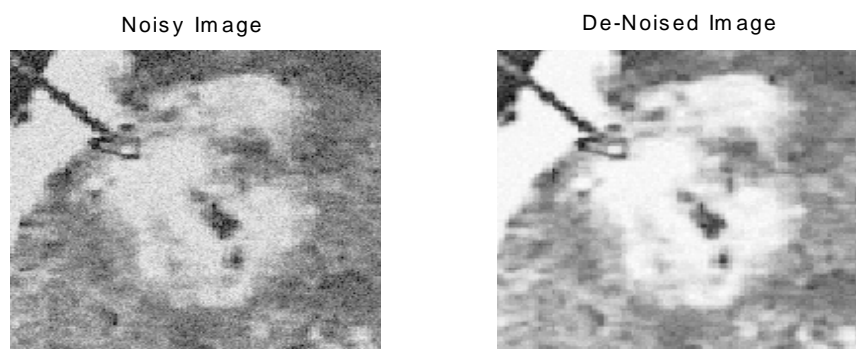
- (1) Selecting the region of interest through a Mouse
- (2) By specifying the region in the form of array

- (1) **Selecting the region of interest through a Mouse:** The region of interest in the image is selected through mouse and on converting the selected image into grey scale the following figure appears:



**Figure 6.24 Selected region of interest using mouse**

As image is not noisy so introducing the noise (with noise variance=25) in figure 6.24 and then de-noising the image using DWT with db1 wavelet we get figure 6.25



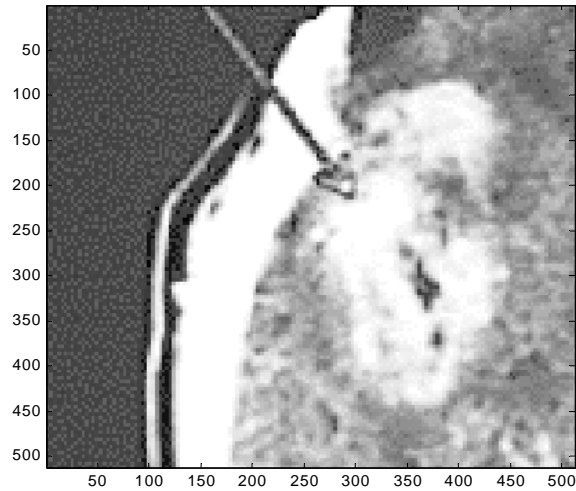
**Figure 6.25 Noisy and De-Noise MR image of selected region shown in figure 6.24**

---

---

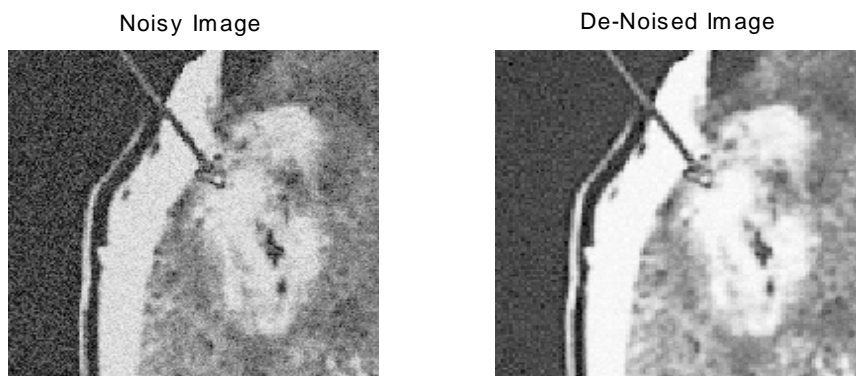
(2) **By specifying the region in the form of array**

By specifying the region of interest in the form of array in the MATLAB code and converting the image in the grey form we get figure 6.26:



**Figure 6.26** Selected region of interest by specifying an array

Introducing the noise (with noise variance=25) in figure 6.26 and de-noising the image using DWT with db1 wavelet figure 6.27 is generated.



**Figure 6.27** Noisy and De-Noise MR image of selected region shown in figure 6.26

MRI acquisition of MRI images helps the biomedical engineers to fully analyze different aspects of the brain thereby reducing the need for surgery. With appropriate image analysis techniques, a biomedical engineer can use one small set of MR images and manipulate them to analyze some interesting facets of the brain.

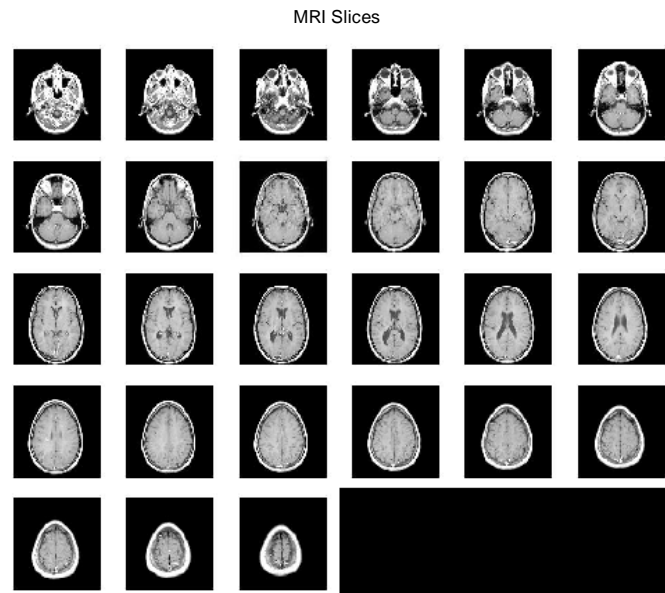
---



---

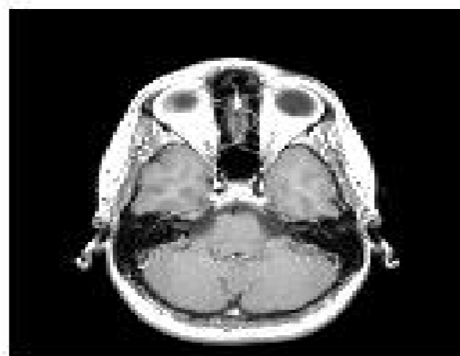
To load MR images of the brain into MATLAB and perform the necessary image analysis specifically the task will require us to carry out the following steps:

- Loading the MRI data set file.
- Displaying a cross sectional view of all MR image frames in one figure



**Figure 6.28** A sequence of MR image slices

- Isolating a frame of interest (e.g. slice no. 6) and display it as an individual figure. Figure 6.29 shows the isolated slice No. 06:



**Figure 6.29** Slice No 6 from sequence of MR image slices

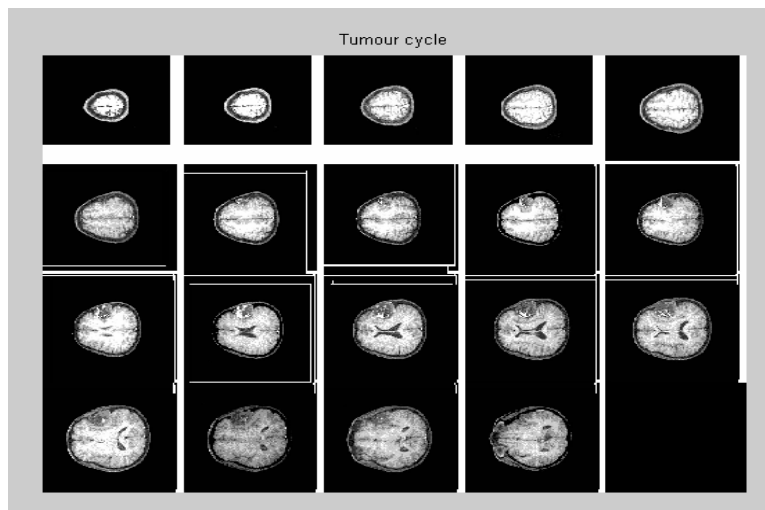
Particular frame of interest can be de-noised in the similar way as we discussed in this chapter with different wavelets and at any noise variance.

---

---

The images are called slices because they look like we have cut the brain in many slices to see what's inside. Using MATLAB movie function it's possible to show all frames of MRI data as a movie.

There are different development stages in which the tumour grows and to study the each stage is essential for diagnosis purpose. The 19 different slices of Tumour cycle is shown below in figure 6.30:



**Figure 6.30** A sequence of Tumour cycle

Using MATLAB movie function it's possible to show all frames of tumour cycle as a movie. Any Slice no. of interest can be isolated and de-noised using same procedure as we have discussed in this chapter.

---

---

# ***Chapter VII***

---

---

## CHAPTER VII

### CONCLUSION AND FUTURE SCOPE

#### Conclusion

The work has been devoted to the de-noising algorithms based upon the discrete wavelet transform that can be applied to enhance noisy MR data sets and ECG signal. The trade-off between noise elimination and detail preservation was analysed using the PSNR (for MR Image), SNR (for ECG signal) and visual criteria. Thus a comparison between the qualities and performance of various wavelet functions were deduced using these criteria. Effectiveness of transform is dependent on the type of image, the error criterion used, the nature and amount of contaminating noise. It was seen that the third-order Daubechies (*db3*) wavelet function performed well for the de-noising of the random noise both in the cases of ECG signal and MR Image as compared to *db1* wavelet, this can be clearly seen with its considerable improvement in PSNR and producing visually more pleasing images thus we can conclude that the higher a vanishing moments is, the better the results are.

The wavelet transform is used to decompose an image and signal into a low-frequency component and a set of higher-frequency details. By analyzing the wavelet transform coefficients, high frequency details which correspond to signal noise, can be eliminated by using SWT and DWT. Noise is reduced from an image without losing the anatomical information which is of interest to medical doctors. Wavelets have demonstrated to be a very powerful tool for analysis, processing and synthesis of relevant image features. Different threshold methods is applied with DWT on ECG signal and the noise is reduced to a great extent while preserving the sharp features of signal which is used by the doctors for diagnosis purposes. SWT shows better result as compared to DWT in both ECG signal and MR Image. Among three thresholding method viz. Sqtwolog, heursure and minimaxi, heursure gives better SNR on de-noising ECG signal using DWT.

---

---

## **Future Scope**

The work presented in this thesis can be extended in several directions. In the analysis of the objects in images it is essential that we can distinguish between the objects of interest and the rest e.g. the background. The techniques that are used to find the objects of interest are usually referred to as segmentation techniques – segmenting the foreground from background. Thus the further work can be devoted in the segmentation of image which subdivides an image into its constituent regions or objects and it is an important step toward the analysis phase.

Over the years computer aided ECG signal analysis is gaining momentum with tremendous amount of work being carried out all over the world. The work presented in this thesis is a small step in that direction. The ECG signal used in this study was downloaded from the internet and all the work is done in MATLAB®. However a separate data acquisition module can also be designed which will capable of picking the analog ECG signals from the existing machines, convert them into digital signals and then transmit them to the computer via the serial port. This data is then stored as a binary file and can be employed for the processing purposes.

Image De-noising plays a vital role in diagnosis purposes. There is a lot of scope for improvements in the proposed image de-noising algorithm. The present work left us with some areas of the image de-noising that are yet unexplored. There are other threshold techniques that can also be used.

De-noising algorithm used in this work uses single wavelet and can also be performed with Multiple Wavelets Transform. Also Wavelet Packet is yet another challenging technique to de-noise the signals and images with greater efficiency and with better promising results which can be used in future for better results.

---

---

# ***REFERENCES***

---

---

## REFERENCES

- [1] A. Chambolle, R. A. DeVore, N. Lee, and B. L. Lucier. Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage. *IEEE Transactions on Image Processing*, 7(3): 319–335, Mar. 1998.
- [2] D. Donoho. De-noising by soft thresholding. *IEEE Transactions on Information Theory*, 41:613–627, 1995.
- [3] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, San Diego, 1998.
- [4][Dau90] I. Daubechies, The wavelet transform, time-frequency localization and signal analysis, *IEEE Trans. Inform. Theory*, vol. 36, pp.961-1005, 1990
- [5][Wor96] G.W.Wornell, “Emerging application of multirate signal processing and wavelets in digital communications,” *Proc. IEEE*, vol. 84, pp.586-603, 1996.
- [6][Sch96a] P. Schroder, Wavelet in Computer graphics, *Proc. of IEEE*, vol. 84, n. 4, pp. 615-625, 1996
- [7][Mey93] Y Meyer, *Wavelets Algorithms and Application*, translated by R. D. ryan, SIAM, Philadelphia, 1993.
- [8][Mal89] S. G. Mallat, A Theory of multiresolution signal decomposition: the wavelet representation, *IEEE Trans. on Pattern Analys. and Machine Intell.* vol. 11m n. 7, pp. 674-693, 1989.
-

- 
- [9] Addison P S 2002 *The Illustrated Wavelet Transform Handbook: Introductory Theory and Applications in Science Engineering, Medicine and Finance* (Bristol: Institute of Physics Publishing)
- [10] Dupuis P and Eugene C 2000 Combined detection of respiratory and cardiac rhythm disorders by high-resolution differential cuff pressure measurement *IEEE Trans. Instrum. Meas.* **49** 498–502
- [11] Hadjileontiadis L J and Panas SM1997 Separation of discontinuous adventitious sounds from vesicular sounds using a wavelet-based filter *IEEE Trans. Biomed. Eng.* **44** 1269–81
- [12] Marrone A, Polosa A D, Scioscia G, Stramaglia S and Zenzola A 1999 Wavelet analysis of blood pressure waves in vasovagal syncope *Physica A* **271** 458–69
- [13] Khalil M and Duchene J 2000 Uterine EMG analysis: a dynamic approach for change detection and classification *IEEE Trans. Biomed. Eng.* **47** 748–55
- [14] Petrosian A, Prokhorov D, Homan R, Dasheiff R and Wunsch D 2000 Recurrent neural network based prediction of epileptic seizures in intra- and extracranial EEG *Neurocomputing* **30** 201–18
- [15] Arneodo A, d'Auberton-Carafa Y, Audit B, Bacry E, Muzy J F and Thermes C 1998 Nucleotide composition effects on the long-range correlations in human genes *Eur. Phys. J. B* **1** 259–63
- [16] Goupillaud P, Grossmann A and Morlet J 1984 Cycle-octave and related transforms in seismic signal analysis *Geoexploration* **23** 85–102
-



- 
- [17] {jin\_Coifman\_1992} Coifman, R. R., Meyer, Y., and Wickerhauser, M. V., Wavelet Analysis and signal processing in Wavelets and their applications, Ruskai, B., Ed. Boston: Jones and Barlett, pp. 153-178, 1992
- [18] {jin\_Mallat\_1998} Mallat, S., A Wavelet Tour of Signal Processing. San Diego, CA: Academic Press, 1998.
- [19] {jin\_Coifman\_1995b} Coifman, R. R. and Woog, L. J., Adapted waveform analysis, wavelet packets, and local cosine libraries as a tool for image processing, Investigative and trial image processing, San Diego, California, Vol. 2567, 1995b.
- [20][Mal89] S. G. Mallat, A Theory of multiresolution signal decomposition: the wavelet representation., *IEEE Trans. on Pattern Analys. and Machine Intell.*, vol. 11 n. 7, pp. 674-693, 1989.
- [21] Ghael, Sandeep P., Sayeed, Akbar M., and Baraniuk, Richard G., "Improved Wavelet Denoising via Empirical Wiener Filtering," Proceedings for SPIE, Mathematical Imaging, San Diego, July 1997.
- [22] Jansen, M. and Bultheel, A., "Smoothing Non-Equidistantly Sampled Data using Wavelets and Cross Validation."
- [23] Lang, M., Guo, H., Odegard, E., Burrus, C. S., and Wells, R. O., "Nonlinear processing of a shift variant DWT for noise reduction," paper accepted for the Proceedings of SPIE, Mathematical Imaging: Wavelet Applications for Dual Use, in SPIE Symposium on OE/Aerospace Sensing and Dual Use Photonics, 17-21, April 1995 Orlando, FL.
-

- 
- [24] Resnikoff, H. and Wells, R., Jr., "Wavelet Analysis: The Scalable Structure of Information," Springer-Verlag, New York, NY, pp. 365.
- [25] Aboufadel, Edward and Schlicker, Steven, "Discovering Wavelets," John Wiley and Sons, New York, NY, pp.14, 1999.
- [26] Donoho, D. L. and Johnstone, I. M., "Ideal spatial adaption via wavelet shrinkage," *Biometrika*, vol. 81, pp. 425- 455, 1994.
- [27] {jin\_Coifman\_1995a}, Coifman, R. and Donoho, D., Translation-invariant De-noising. in *Wavelets and Statistics*, Antoniadis, A. and Oppenheim, G., Eds. New York NY: Springer- Verlag, 1995a.
- [28] {jin\_Donoho\_1994a}, Donoho, D. and Johnstone, I., Ideal Spatial Adaptation via Wavelet Shrinkage., *Biometrika*, Vol. 81, pp. 425-455, 1994a.
- [29] {jin\_Donoho\_1995a; Donoho, D., Nonlinear solution of linear inverse problems by wavelet-vaguelette decompositions, *Journal of Applied and Computational Harmonic Analysis*, Vol. 2, No. 2, pp. 101-126, 1995a.
- [30] {jin\_Stein\_1981}. Stein, C., Estimation of the mean of a multivariate normal distribution., *Annals of Statistics*, Vol. 9, pp. 1135-1151, 1981.
- [31] {jin\_Chang\_2000}, Chang, S., Yu, B., and Vetterli, M., Spatially adaptive wavelet thresholding with context modeling for image denoising, *IEEE Transactions on Image Processing*, Vol. 9, No. 9, pp. 1522-1531, 2000.
- [32] {jin\_Donoho\_1995c} Donoho, D. and Johnstone, I., Adapting to Unknown Smoothness via Wavelet Shrinkage., *Journal of American Statistics Association*, Vol. 90, No. 432, pp. 1200-1224, 1995c.
-

---

**Websites :**

❖ **For Tutorials and Practical guide:**

<http://www.eeglossary.com/wavelets.htm>

❖ **A Really Friendly guide to Wavelet:**

<http://perso.wanadoo.fr/polyvalens/clemens/wavelets/wavelets.html>

❖ **A Wavelet Tour Of Signal Processing By Stéphane Mallat**

[http://cas.ensmp.fr/~chaplais/Wavetour\\_presentation/Wavetour\\_presentation\\_US.htm](http://cas.ensmp.fr/~chaplais/Wavetour_presentation/Wavetour_presentation_US.htm)

❖ **Wavelets and other transforms:**

<http://telin.rug.ac.be/~frooms/links/wavelets.shtml>

❖ **Wavelet Software: -**

<http://atoc.colorado.edu/research/wavelets/software.html>

❖ **Discovering Wavelets: Student Work:**

<http://faculty.gvsu.edu/aboufade/web/students.htm>

❖ **MATLAB Educational Sites: -**

<http://www.eece.maine.edu/mm/matweb.html#as-h3-1932230>

❖ **Medical Images and Illustrations:**

<http://www.mic.ki.se/MEDIMAGES.html#A07>

❖ <http://users.pandora.be/educyclopedia/electronics/control-systems.htm>

❖ <http://ocw.mit.edu/OcwWeb/Electrical-Engineering-and-Computer-Science/index.htm#UndergraduateGraduate>

---

- 
- ❖ <http://www.control.lth.se/~kursolin/>
  - ❖ <http://www.web-ee.com/Downloads/Simulator/simulation.htm>
  - ❖ [www.dcresource.com/reviews/canon/powershot\\_s2-review/index.shtml](http://www.dcresource.com/reviews/canon/powershot_s2-review/index.shtml) -  
101k - 22 Aug 2005
  - ❖ [http://en.wikipedia.org/wiki/Discrete\\_wavelet\\_transform](http://en.wikipedia.org/wiki/Discrete_wavelet_transform)

**Books:**

- ❖ A.K. Jain, (2004) “ **Fundamentals of Digital image processing**” PHI
  - ❖ [Agostino Abbate, Casimer M. DeCusatis, Pankaj K. Das, \(2002\) “ \*\*Wavelets and Subbands- Fundamentals and Applications\*\*” Birkhäuser](#)
  - ❖ [Rafael C. Gonzalez, Richard E. Woods, \(2004\) “\*\*Digital Image Processing\*\*” PEARSON EDUCATION](#)
-

---

# ***APPENDIX***

---

---

## APPENDIX

### (A) MATLAB CODE FOR IMAGE AND SIGNAL DE-NOISING

```
clc;
clear all;
n=menu('IMAGE','ECG signal','MRI IMAGE','Exit');

if n==1
%%%%%%%%%%%%%% Signal De-noising%%%%%%%%%%%%%%
load('C:\Documents and Settings\Administrator\My Documents\DSP\HRTSIG.MAT')
END=length(sig);
x=sig(1:end,1); %Time
y=sig(1:end,2); %ECG data #1
z=sig(1:end,3); %ECG data #2
count=4000;
y2=y;%temp variable
temp=y;
s=y;%temp variable:-s contains a signal
l_s = length(s);
LEVEL=3;
%%%%%%%%%%%%%% Wavelet Selection%%%%%%%%%%%%%%
i=menu('WAVELET for Signal','db1(Haar)','db3','Exit');

if i==1
wavelet='db1';
elseif i==2
wavelet='db3';
else
break
end
[C,L] = wavedec(s,3,wavelet);
%Extract approximation and detail coefficients.
%To extract the level 3 approximation coefficients from wavelet decomposition
%structure [C,L]
cA3 = appcoef(C,L,wavelet,3);
%To extract the levels 3, 2, and 1 detail coefficients from wavelet decomposition
%structure [C,L]
[cD1,cD2,cD3] = detcoef(C,L,[1,2,3]);
A3 = wrcoef('a',C,L,wavelet,3);
disp(' To display the details D1, D2, and D3 ');
D1 = wrcoef('d',C,L,wavelet,1);
D2 = wrcoef('d',C,L,wavelet,2);
D3 = wrcoef('d',C,L,wavelet,3);
pause;
subplot(3,1,1); plot(D1(20:800)); title('Detail Level 1');
subplot(3,1,2); plot(D2(20:800)); title('Detail Level 2');
subplot(3,1,3); plot(D3(20:800)); title('Detail Level 3');
```

---

---

```

pause;
close;
A0 = waverec(C,L,wavelet);
err = max(abs(s-A0))
%The level 3 approximation, A3, is quite clean as a compare to original signal.
pause;
disp('To compare the approximation to the original signal');
pause;
subplot(2,1,1);plot(s(20:1000));title('Original');
subplot(2,1,2);plot(A3(20:1000));title('Level 3 Approximation-QUIT CLEAN');
pause;
close;
pause;

j=menu('DENOISING','DWT','SWT');
if j==1
%%%%%%%%%%%% D W T %%%%%%%%%%%%%
t=menu('DWT thresholding method ','fixed thresholding','heursure','sqrtwolog','minimaxi');
if t==1
disp(' Remove noise by thresholding using DWT');
%Remove noise by thresholding.
[thr,sorh,keepapp] = ddencmp('den','wv',s)
clean = wdencomp('gbl',C,L,wavelet,3,thr,sorh,keepapp);
pause;
disp('To display both the original and de-noised signals using l thresholding ');
pause;
subplot(2,1,1); plot(s(20:1000)); title('Original')
subplot(2,1,2); plot(clean(20:1000)); title('De-noised using DWT(Thresholding)')
pause;
close;
pause;
plot(s(20:1000),'red'); hold on
plot(clean(20:1000),'blue');
legend('Original','De-noised using DWT(Thresholding)')
pause;
close all;
disp('SNR with thresholding(DWT)');
correlation_matrix = corrcoef(s, clean);
C1 = correlation_matrix(1,2);
signal_to_noise_ratio = 10 * log10( C1 / (1 - C1))
pause;break

elseif t==2
TH_M='heursure';
elseif t==3
TH_M='sqrtwolog';
elseif t==4
TH_M='minimaxi';
else
break
end
xd2 = wden(y,TH_M,'s','mln',3,wavelet);

```

---

---

```

correlation_matrix = corrcoef(y, xd2);
C1 = correlation_matrix(1,2);
signal_to_noise_ratio = 10 * log10( C1 / (1 - C1))
pause;
subplot(2,1,1); plot(y(20:800)); title('noisy/original')
subplot(2,1,2); plot(xd2(20:800)); title('De-noised')
pause;
close;
pause;
plot(y(20:1000),'red'); hold on
plot(xd2(20:1000),'blue');
legend('noisy/Original','De-noised -- sqtwolog')
pause;
close all;

elseif j==2
%%%%%%%%%% S W T %%%%%%%%%%
%LEVEL=3;
[swa,swd] = swt(s,1,wavelet);
%To find the inverse transform
A0 = iswt(swa,swd,wavelet);
%To construct the level 1 approximation and detail (A1 and D1) from the
%coefficients swa and swd
nulcfs = zeros(size(swa));
A1 = iswt(swa,nulcfs,wavelet);
D1 = iswt(nulcfs,swd,wavelet);
[swa,swd] = swt(s,LEVEL,wavelet);
%To reconstruct the approximation at level 3, type
mzero = zeros(size(swd));
A = mzero;
A(3,:) = iswt(swa,mzero,wavelet);
%To reconstruct the details at levels 1, 2 and 3
D = mzero;
for i = 1:3
swcfs = mzero;
swcfs(i,:) = swd(i,:);
D(i,:) = iswt(mzero,swcfs,wavelet);
end
pause;
close all;
%To reconstruct the approximations at levels 2 and 3
A(2,:) = A(3,:) + D(3,:);
A(1,:) = A(2,:) + D(2,:);
%To display the approximations and details at levels 1, 2 and 3
kp = 0;
for i = 1:3
disp('Display the approximations and details at levels 1, 2 and 3');
pause;
subplot(3,2,kp+1), plot(A(i,:));title(['Approx. level',num2str(i)])
subplot(3,2,kp+2), plot(D(i,:));title(['Detail level',num2str(i)])
kp = kp + 2;
end

```

---



---

```

pause;
close all;
%Remove noise by thresholding.
[thr,sorh] = ddencomp('den','wv',s);
dswd = wthresh(swd,sorh,thr);
xd = iswt(swa,dswd,wavelet);
subplot(2,1,1), plot(s(20:1000));title('Original signal')
subplot(2,1,2), plot(xd(20:1000));title('De-noised signal using SWT(Thresholding)')
pause;
close all;
pause;
plot(s(20:1000),'red'); hold on
plot(xd(20:1000),'blue');
legend('Original','De-noised signal using SWT( Thresholding)')
pause;
close all;

disp('SNR with thresholding using SWT');
correlation_matrix = corrcoef(s, xd);
C1 = correlation_matrix(1,2);
signal_to_noise_ratio = 10 * log10( C1 / (1 - C1))
pause;
close all;
else
break
end

elseif n==2
%%%%%%%%%%%%%% IMAGE De-noising %%%%%%%%%%%%%%%
rgb = imread('C:\Documents and Settings\Administrator\Desktop\brain.jpg');
NbColors = 255;
I = rgb2gray(rgb);
Y = imresize(I,[512 512]);
Y = double(Y);
map = pink(NbColors);
newmap = rgb2gray(map);
imshow(Y, newmap);
pause;
[m n]=size(Y);
%%%%%%%%%%%%%% wavelets selection%%%%%%%%%%%%%%
s=menu('WAVELET for Image','db1(Haar)','db3','Exit');
% Daubechies
if s==1
wavelet='db1';
elseif s==2
wavelet='db3';
else
break
end
% no. of decomposition levels
level=2;

```

---

---

```

% 2D DWT DECOMPOSITION
Y0 = DECOMP(Y,level,wavelet);
%DECOMP is a function created for 2D DWT Decomposition
r=menu('SELECT','DENOISING','COMPRESSION');
if r==1
    %%%%%%%%%% DENOISING %%%%%%%%%%
    init=2055615866; randn('seed',init);
    l=menu('Noise level','25','30','35');
    if l==1
        NV=25;
    elseif l==2
        NV=30;
    elseif l==3
        NV=35;
    else
        break
    end

    x = Y + NV*randn(size(Y));
    % Array dimensions of the original image
    [m n]=size(Y);
    %%%%%%%%%% TYPE OF DENOISING %%%%%%%%%%
    k=menu('TYPE','DWT','SWT')
    if k==1
        %%%%%%%%%% D W T %%%%%%%%%%
        [thr,sorh,keepapp] = ddencmp('den','wv',x);
        xd = wdencomp('gbl',x,wavelet,2,thr,sorh,keepapp);
        x = imresize(x,[512 512]);xd = imresize(xd,[512 512]);
        colormap(newmap), sm = size(map,1);
        subplot(121), image(wcodemat(x,sm)), title('Noisy Image');axis square;axis off;
        subplot(122), image(wcodemat(xd,sm)), title('De-Noised Image');axis square;axis off;
        Y = imresize(Y,[512 512]);
        Y=double(Y);
        xd = imresize(xd,[512 512]);
        xd=double(xd);
        err = Y - xd;
        PSNR = 20*log10(255/std(err(:)))
        pause;

    elseif k==2
        %%%%%%%%%% S W T %%%%%%%%%%
        [swa,swh,swv,swd] = swt2(x,2,wavelet);
        mzero = zeros(size(swd));
        A = mzero;
        A(:,2) = iswt2(swa,mzero,mzero,mzero,wavelet);
        H = mzero; V = mzero;
        D = mzero;
        for i = 1:2
            swcfs = mzero; swcfs(:,i) = swh(:,i);
            H(:,i) = iswt2(mzero,swcfs,mzero,mzero,wavelet);
            swcfs = mzero; swcfs(:,i) = swv(:,i);
            V(:,i) = iswt2(mzero,mzero,swcfs,mzero,wavelet);

```

---

---

```

    swcfs = mzero; swcfs(:,:,i) = swd(:,:,i);
    D(:,:,i) = iswt2(mzero,mzero,mzero,swcfs,wavelet);
end
%To reconstruct the approximations at levels 1 and 2, type
A(:,:,1) = A(:,:,2) + H(:,:,2) + V(:,:,2) + D(:,:,2);
%To display the approximations and details at levels 1, 2, and 3, type
[thr,sorh,keepapp] = ddencmp('den','wv',x);
dswh = wthresh(swh,sorh,thr);
dswv = wthresh(swv,sorh,thr);
dswd = wthresh(swd,sorh,thr);
xds = iswt2(swa,dswh,dswv,dswd,wavelet);
%To display both the original and de-noised images, type
%colormap(map)
colormap(newmap)
subplot(1,2,1), image(wcodemat(x,192));title('Noisy image');axis square;axis off;
subplot(1,2,2), image(wcodemat(xds,192));title('De-noised image');axis square;axis off;
pause;
close all;
pause;
Y = imresize(Y,[512 512]);
Y=double(Y);
xds = imresize(xds,[512 512]);
xds=double(xds);
err = Y - xds;
PSNR = 20*log10(255/std(err(:)))
pause;
end

elseif r==2
    %%%%%%%%%%Compression %%%%%%%%%%
    [C,S] = wavedec2(Y,2,wavelet);
    [thr,sorh,keepapp] = ddencmp('cmp','wv',Y);
    [Ycomp,CYC,LYC,PERF0,PERFL2] = wdencmp('gbl',C,S,wavelet,2,thr,sorh,keepapp);
    figure(3);
    colormap(newmap);
    subplot(121); image(Y); title('Original Image');axis square;axis off;
    subplot(122); image(Ycomp); title('Compressed Image');axis square;axis off;
    pause;
    ci = norm(Ycomp); %taking vector-norm of the compressed image
    oi = norm(Y); %taking vector-norm of the original image
    CMR=(100*(ci)^2)/((oi)^2)
end
end

```

---

---

**(B) MATLAB CODE TO SHOW ALL FRAMES OF TUMOUR CYCLE AS A  
MOVIE AND DE-NOISING THE ISOLATED FRAME OF INTEREST**

```
clc ;  
clear all;  
pause;  
%There are different development stages in which the tumor grows and to study the each  
%stage is essential for diagnosis purpose
```

```
X1 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC1.jpg');  
X2 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC2.jpg');  
X3 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC3.jpg');  
X4 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC4.jpg');  
X5 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC5.jpg');  
X6 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC6.jpg');  
X7= imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC7.jpg');  
X8 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC8.jpg');  
X9 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC9.jpg');  
X10 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC10.jpg');  
X11 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC11.jpg');  
X12 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC12.jpg');  
X13 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC13.jpg');  
X14 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC14.jpg');  
X15 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC15.jpg');  
X16 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC16.jpg');  
X17 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC17.jpg');  
X18 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC18.jpg');  
X19 = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC19.jpg');
```

```
X1 = imresize(X1,[256 256]);  
X2 = imresize(X2,[256 256]);  
X3 = imresize(X3,[256 256]);  
X4 = imresize(X4,[256 256]);  
X5 = imresize(X5,[256 256]);  
X6 = imresize(X6,[256 256]);  
X7 = imresize(X7,[256 256]);  
X8 = imresize(X8,[256 256]);  
X9 = imresize(X9,[256 256]);  
X10 = imresize(X10,[256 256]);  
X11 = imresize(X11,[256 256]);  
X12 = imresize(X12,[256 256]);  
X13 = imresize(X13,[256 256]);  
X14 = imresize(X14,[256 256]);  
X15 = imresize(X15,[256 256]);  
X16 = imresize(X16,[256 256]);  
X17 = imresize(X17,[256 256]);  
X18 = imresize(X18,[256 256]);  
X19 = imresize(X19,[256 256]);  
NbColors = 155;
```

---

---

```
map = pink(NbColors);
map = double(map);

M(1) = im2frame(X1,map);
M(2) = im2frame(X2,map);
M(3) = im2frame(X3,map);
M(4) = im2frame(X4,map);
M(5) = im2frame(X5,map);
M(6) = im2frame(X6,map);
M(7) = im2frame(X7,map);
M(8) = im2frame(X8,map);
M(9) = im2frame(X9,map);
M(10) = im2frame(X10,map);
M(11) = im2frame(X11,map);
M(12) = im2frame(X12,map);
M(13) = im2frame(X13,map);
M(14) = im2frame(X14,map);
M(15) = im2frame(X15,map);
M(16) = im2frame(X16,map);
M(17) = im2frame(X17,map);
M(18) = im2frame(X18,map);
M(19) = im2frame(X19,map);
axis off;
movie(M)
pause;
close all;

A =cat(19,X1,X2,X3,X4,X5,X6,X7,X8,X9,X10,X11,X12,X13,X14,X15,X16,X17,X18,X19);
figure
montage(A, map)
title('Tumour cycle')
pause;
close all

p= input('Input the slice(from 1 to 19) No to be denoised: ')

switch p;

    case 1

        X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC1.jpg');

    case 2

        X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC2.jpg');

    case 3

        X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC3.jpg');
```

---

---

case 4

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC4.jpg');

case 5

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC5.jpg');

case 6

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC6.jpg');

case 7

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC7.jpg');

case 8

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC8.jpg');

case 9

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC9.jpg');

case 10

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC10.jpg');

case 11

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC11.jpg');

case 12

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC12.jpg');

case 13

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC13.jpg');

case 14

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC14.jpg');

case 15

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC15.jpg');

case 16

X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC16.jpg');

---

---

```
case 17
    X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC17.jpg');
case 18
    X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC18.jpg');
case 19
    X = imread('C:\Documents and Settings\Administrator\Desktop\TumorCycle\TC19.jpg');
otherwise
    disp('MRI Slice Does not exit')
    break;
end
pause;
X = X(:,:,3);
X = imresize(X,[512 512]);
X = double(X);
map = pink(NbColors);
newmap = rgb2gray(map)
init=2055615866; randn('seed',init);
x1 = X + 15*randn(size(X));
[thr,sorh,keepapp] = ddencmp('den','wv',x1);
xd1 = wdencomp('gbl',x1,'db1',2,thr,sorh,keepapp);
colormap(newmap), sm = size(newmap,1);
subplot(221), image(wcodemat(X,sm)), title('Original Image');axis off;
subplot(222), image(wcodemat(x1,sm)), title('Noisy Image');axis off;
subplot(223), image(wcodemat(xd1,sm)), title('De-Noised Image');axis off;
err = X - xd1;
PSNR = 20*log10(255/std(err(:)))
pause;
close all;
pause;
```

---