

**NEURO-FUZZY FILTERING AND  
MULTIWAVELET TRANSFORMATION FOR  
EEG SIGNAL FOR EFFICIENT REMOVAL OF  
ARTIFACTS**

**A THESIS**

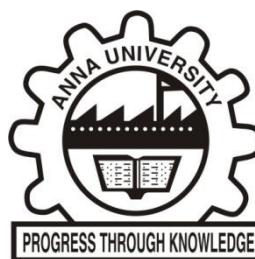
*Submitted by*

**PAULCHAMY B**

*in partial fulfilment for the requirement of award of the degree*

*of*

**DOCTOR OF PHILOSOPHY**



**FACULTY OF INFORMATION AND  
COMMUNICATION ENGINEERING**

**ANNA UNIVERSITY  
CHENNAI 600 025**

**MARCH 2013**

**ANNA UNIVERSITY  
CHENNAI 600 025**

**BONA FIDE CERTIFICATE**

Certified that this thesis titled "**NEURO-FUZZY FILTERING AND  
MULTIWAVELET TRANSFORMATION FOR EEG SIGNAL FOR  
EFFICIENT REMOVAL OF ARTIFACTS**" is the bonafide work of  
**Mr. PAULCHAMY B** who carried out the research under my supervision.

Certified further, that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion to this or any other scholar.

Place:

**Dr. ILA VENNILA**

Date:

SUPERVISOR

Associate Professor

Department of Electrical and  
Electronics Engineering  
PSG College of Technology,  
Coimbatore – 641 004

## ABSTRACT

Brain is the most complex organ in the human body. The brain creates a range of electric potential for every action done by the human. For brain diagnosis the Electroencephalogram (EEG) is the signal of interest. But EEG which should read the scalp electrical activity of the human body also reads its physiological and extra physiological activities which are collectively called as ‘artifacts’. These artifacts which are the interference to EEG should be eliminated for proper diagnosis. In this thesis, four methods are developed for the efficient removal of artifacts.

The first method describes the basic principle behind the independent component analysis technique. The contrast functions for different routes to independence are clearly depicted. Independent Component Analysis is a technique to separate signals from a mixture. ICA method discusses the functions that measure the non-Gaussianity of any dataset which leads to the gradient algorithm to maximize the non-gaussianity of datasets which in turn is the basis of Fast-ICA algorithm. It explains the higher order cumulants followed by a discussion of cumulant tensors. The role of cumulant tensors in the ICA algorithm Joint approximate Diagnolization of Eigen matrices (JADE) is discussed. The proposed improved JADE algorithm uses

significant eigen pairs of the cumulant tensor  $F$  ( $M$ ) to find out the estimated values of independent components. The tensor eigen value decomposition is considered as more of a preprocessing step. The improved JADE gives better results among the four ICA algorithms in terms of their convergence speed, entropy and signal to noise ratio.

The second method proposes principle behind the neuro fuzzy system clearly. In this work the neuro-fuzzy in fuzzy modeling research field is divided into two areas: linguistic fuzzy modeling that is focused on interpretability, mainly the Mamdanimodel; and precise fuzzy modeling that is focused on accuracy, mainly the Takagi-Sugeno-Kang (TSK) model. It starts with the development of a “fuzzy neuron” based on the understanding of biological neuronal morphologies, followed by learning mechanisms .Using neuro fuzzy filter the artifacts are extracted from EEG signal. Compare to ICA Algorithms it provides better results in terms of Signal to Noise Ratio (SNR).

The third method presents the principle of Haar transform. This transform cross-multiplies a function against the Haar wavelet with various shifts and stretches, like the Fourier transform cross-multiplies a function against a sine wave with two phases and many stretches. The Haar transform can be thought of as a sampling process in which rows of the transform for noise removal. It provides shortest path and time consumption. In this, work

the Haar wavelet of higher orders is used to decompose the recorded EEG signal to detect the exact moment when the state of the eyes changes and on subsequent section to eye-blinks and movements of the eyeballs as well. Compare to Neuro fuzzy filter it provides better results in terms of SNR and Power Spectral Density (PSD).

The fourth method presents the details behind the multiwavelet transform. They are defined using several wavelets with several scaling functions. Multiwavelet has several advantages incomparison with scalar wavelet. The features such as compact support, orthogonally, symmetry, and higher order approximation are known to be important in signal processing. In this method thresholding technique is used for signal de-noising. Decomposing a signal using the wavelet transform, a set of wavelet coefficients that correlates to the high frequency sub bands. These high frequency sub bands consist of the details in the data set. If these details are small enough, they might be omitted without substantially affecting the main features of the data set. The de-noising of EEG signal is carried out by using different combinations of threshold limit, thresholding function and window sizes. Choice of threshold limit and thresholding function is a crucial step in the denoising procedure, as it should not remove the original signal coefficients leading to loss of critical information in the analyzed data.

It can be seen that the Multiwavelet transform is more efficient in removal of artifacts than the other methods namely Improved JADE, Neuro fuzzy filter and Haar wavelet transform. The efficiency is measured in terms of SNR and Correlation factor. Hence Multiwavelet is considered as the most consistent and robust method for the removal of artifacts in EEG.

## ACKNOWLEDGEMENT

First of all I thank Almighty for showering his blessings on me with supernatural grace and mercies in abundance, without which this thesis would not have been completed successfully.

I would like to express my deepest gratitude to my supervisor **Dr. Ila Vennila**, Associate Professor, Department of Electrical and Electronics Engineering, PSG College of Technology, Coimbatore for her constant encouragement, unwavering support and valuable advice. Her inspirational support has made this work possible.

I thank my Doctoral committee members, **Dr.K.Thanushkodi**, Director, Akshaya College of Engineering and Technology, Coimbatore and **Dr. M.L. Valarmathi**, Associate professor, Government college of Technology, Coimbatore for providing valuable comments during this research work. I thank **Dr.R.Rudramoorthy**, principal and the management PSG College of Technology, for the infrastructure provided to pursue this research work and also I thank **Dr.P.Navaneethan**, Professor and Head and all the faculty members of EEE department, PSG college of Technology for the encouragement given towards this research work. I am highly grateful to **Dr. T.Ravichandran**, Principal, of Hindusthan Institute of Technology, Coimbatore for the help rendered to carry out this research work.

I want to express my full hearted gratefulness and whole hearted appreciation to my wife **Dr.K.Maheshwari** for her interest, patience, love, affection and consistent encouragement during the whole period of my research work. I wish to express my sincere thanks to **my parents** and all who have helped me in many ways directly or indirectly to complete my research work.

**PAULCHAMY B**

## TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	<b>ABSTRACT</b>	<b>iii</b>
	<b>LIST OF TABLES</b>	<b>xvi</b>
	<b>LIST OF FIGURES</b>	<b>xvii</b>
	<b>LIST OF ABBREVIATION</b>	<b>xxi</b>
<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
	1.1 ELECTROECNCHAPAHALOGRAM	4
	1.1.1 The Evolution Of EEG	4
	1.1.2 Electrode Placement	6
	1.1.3 Brain Wave Classification	7
	1.1.3.1 Delta	8
	1.1.3.2 Theta	8
	1.1.3.3 Alpha	8
	1.1.3.4 Beta waves	9
	1.2 EEG ARTIFACTS	9
	1.2.1 Physiological Artifacts	9
	1.2.1.1 Muscle artifacts	9
	1.2.1.2 Eye movement artifacts	10
	1.2.1.3 ECG artifacts	10
	1.2.1.4 Pulse artifacts	10
	1.2.1.5 Respiration artifacts	10
	1.2.2 Extraphysiological Artifact	11
	1.2.2.1 AC artifacts	11
	1.2.2.2 Others	11

<b>CHAPTER NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
1.3	MOTIVATION FOR THE THESIS	12
1.4	LITERATURE REVIEW	12
1.5	NEED FOR THE THESIS	16
1.6	SCOPE OF THE THESIS	20
1.7	STRUCTURE OF THE THESIS	21
<b>2</b>	<b>INDEPENDENT COMPONENT ANALYSIS</b>	<b>24</b>
2.1	INTRODUCTION	24
2.2	ICA MODEL	26
	2.2.1 Assumptions for ICA Model	29
	2.2.2 Preprocessing of data for ICA	30
2.3	ALGEBRAIC ICA ALGORITHM	30
2.4	FAST ICA ALGORITHM	32
	2.4.1 Measures of Non-Gaussianity	32
	2.4.1.1 Kurtosis	32
	2.4.1.2 Negentropy	34
	2.4.2 Approximations to Negentropy	35
	2.4.3 Algorithms based on Negentropy	38
	2.4.4 Gradient Algorithm	39
2.5	IMPLEMENTATION OF FAST ICA ALGORITHM	41
	2.5.1 Estimating Several Components at the Same Time	47
2.6	JADE ALGORITHM	49
	2.6.1 Cumulants	49
	2.6.2 Cumulant Tensors	51
	2.6.3 Eigenmatrices of Cumulant Tensors	52
2.7	IMPROVED JADE ALGORITHM	54
	2.7.1 Significant Eigen Pairs	55

<b>CHAPTER NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
2.7.2	Extended Jacobi Method for Joint Diagonalization	56
2.8	INFOMAX AND EXTENDED INFOMAX ALGORITHM	60
2.8.1	Mutual Information	61
2.8.2	Mutual Information – ICA Criterion	61
2.9	INFOMAX METHOD	63
2.9.1	Infomax Algorithm	65
2.10	EXTENDED INFOMAX METHOD	68
2.10.1	Deriving a Learning Rule to Separate Sub- and Supergaussian Sources	69
2.11	EYEBLINK COMPONENT IDENTIFICATION	71
2.11.1	EEG Reconstruction	72
2.12	RESULTS	72
2.12.1	Results of Fast ICA Algorithm	74
2.12.2	Results of IJADE Algorithm	75
2.12.3	Results of Infomax:	76
2.12.4	Results of Extended Infomax	77
2.12.5	Performance Comparisons of Algorithms	78
2.13	SUMMARY	82
<b>3</b>	<b>ARTIFACTS REMOVAL FROM EEG USING NEURO FUZZY FILTER</b>	<b>83</b>
3.1	INTRODUCTION	83
3.2	EXTRACTING KNOWLEDGE FROM EXISTING METHODS	84
3.2.1	Armax Modeling Method	84
3.2.2	Disadvantages	84

<b>CHAPTER NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
3.3	ADAPTIVE FILTERING METHOD	85
3.3.1	Adaptive Filtering	87
3.3.2	Cancellation of Artifacts	88
3.3.3	Methodology of Adaptive Filter	91
3.3.4	Advantages	94
3.3.5	Limitation of Adaptive Filtering Method	95
3.4	ARTIFICIAL NEURAL NETWORKS	95
3.4.1	Neural Networks	95
3.4.2	Features Of Neural Networks	97
3.4.3	Architecture Of Artificial Neural Network	98
3.4.3.1	Single Layer Feed Forward Network	98
3.4.3.2	Multi Layer Feed Forward Network	99
3.5	LEARNING METHODS	99
3.5.1	Supervised Learning	101
3.5.2	Unsupervised Learning	102
3.6	BACK PROPAGATION NETWORK	102
3.6.1	Architecture Of Feed Forward Back Propagation Network	103
3.7	FUZZY SYSTEM	104
3.7.1	Fuzzy Sets	105
3.7.2	Membership Function	105
3.7.3	Advantages of Fuzzy Logic	106
3.7.4	Fuzzification	106
3.7.4.1	Knowledge Base	107
3.7.4.2	Fuzzy Inference Engine	107

<b>CHAPTER NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
3.8	DE-FUZZIFICATION	109
3.9	NEURO-FUZZY SYSTEM	110
3.10	PROPOSED METHODOLOGY OF NEURO-FUZZY FILTER	115
	3.10.1 Preprocessing of EEG	115
3.11	FUZZY NEURONS	119
3.12	RESULTS AND DISCUSSION	126
	3.12.1 Artifact Removal Using Adaptive Filter	128
	3.12.2 SNR Value Resulted In Adaptive Filtering	129
	3.12.3 Artifacts Removal Using Neuro-Fuzzy Filter	129
	3.12.4 ECG Artifacts Removal Using Neuro-Fuzzy Filter	130
	3.12.5 EMG Artifacts Removal Using Neuro-Fuzzy Filter	131
	3.12.6 EOG Artifacts Removal Using Neuro-Fuzzy Filter	132
	3.12.7 SNR Value Resulted In Neuro- Fuzzy Filtering	133
	3.12.8 Power Spectral Density of Denoised EEG Signal	134
3.13	CORRELATION BETWEEN ORIGINAL AND ARTIFACTS REMOVED EEG SINGAL	135
	3.13.1 Comparative Study of PSD	135
	3.13.2 Correlative SNR Value	136
3.14	SUMMARY	137

<b>CHAPTER NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
<b>4</b>	<b>ARTIFACTS REMOVAL USING WAVELET TRANSFORM</b>	<b>139</b>
4.1	INTRODUCTION	139
4.2	EXTRACTING KNOWLEDGE FROM EXISTING METHODS	140
4.2.1	Artifact Removal	140
4.2.2	Subtraction Method	140
4.2.3	linear Combination and Regression Method	141
4.2.4	Principal Component Analysis	142
4.2.5	Canonical Correlation Analysis	142
4.2.6	Blind Source Separation(BSS)	143
4.2.7	Linear Flitering	143
4.3	ARTIFACTS REMOVAL FROM EEG USING HAAR WAVELET TRANSFORM	144
4.3.1	Wavelet Transform	145
4.3.2	Wavelet Families	147
4.3.3	Haar Transform	148
4.3.4	Properties	148
4.3.5	Advantages	149
4.3.6	Limitations	149
4.4	HAAR WAVELET BASED DETECTION OF CHANGE IN THE STATE OF THE EYES	150
4.4.1	Amplitude Dependence on the State of the Eye	150
4.4.2	EEG Recorded During Change in State of the Eye	151

<b>CHAPTER NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
4.5	DETECTION OF CHANGE IN THE STATE OF EYES: NEED FOR A HAAR WAVELET BASED APPROACH	151
4.6	PROPOSED METHODOLOGY	152
	4.6.1 Adaptive Noise Cancellation Method	153
	4.6.2 Flow Chart	154
4.7	DE-NOISING PROCEDURE PRINCIPLES	157
4.8	PREPROCESSING OF BRAIN SIGNALS	157
	4.8.1 Need for Preprocessing	158
	4.8.2 Preprocessing-Discussion	158
4.9	OCULAR ZONE IDENTIFICATION	158
	4.9.1 Haar Transform for Analyzing EEG Signal	158
	4.9.2 Automatic Identification of Ocular Artifacts Zones Using Haar Wavelet	160
4.10	DECOMPOSITION USING DISCRETE WAVELET TRANSFORM	161
4.11	SOFT THRESHOLDING	161
4.12	POWER SPECTRAL DENSITY	164
	4.12.1 Properties of Power Spectral Density	164
4.13	SIGNAL-TO-NOISE RATIO	165
4.14	RESULT & DISCUSSION	166
	4.14.1 EEG Signal	166
	4.14.2 Contaminated Signal	166
	4.14.3 Corrected EEG	167

<b>CHAPTER NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
	4.14.4 Power Spectral Density	168
	4.14.5 Correlation Plot	168
4.15	SNR COMPARISON FOR NOISY AND DENOISY SIGNAL	169
4.16	SUMMARY	170
<b>5</b>	<b>MULIWAVELET TRANSFORM</b>	<b>172</b>
5.1	INTRODUCTION	172
5.2	MULTIWAVELET	175
5.3	PRE PROCESSING OF MULTIWAVELETS	178
5.4	MOTIVATION OF MULTIWAVELETS	180
5.5	MULTIWAVELET PACKETS	182
5.6	MULTIWAVELET DENOISING TECHNIQUE	184
5.7	ASSESMENT CRITERIA	186
5.8	RESULTS AND DISCUSSION	187
5.9	SUMMARY	196
<b>6</b>	<b>CONCLUSION AND FUTURE WORK</b>	<b>198</b>
6.1	CONCLUSION	198
6.2	FUTURE ENHANCEMENTS	202
	<b>REFERENCES</b>	<b>203</b>
	<b>LIST OF PUBLICATIONS</b>	<b>210</b>
	<b>CURRICULUM VITAE</b>	<b>213</b>

## LIST OF TABLES

<b>TABLE NO</b>	<b>TITLE</b>	<b>PAGE NO</b>
2.1	Computation of time for ICA algorithm in seconds	78
2.2	Entropy of original EEG	79
2.3	Entropy of EEG after removal of EOG	79
2.4	Signal to Noise Ratio in dB	81
3.1	SNR value of adaptive filter	129
3.2	SNR value of Neuro-Fuzzy filters	133
3.3	SNR value comparision between adaptive and Neuro-Fuzzy filter	136
4.1	SNR comparision between corrupted signal and denoised signal	169
5.1	Comparision of computational complexities of wavelets and multiwavelets	183
5.2	Summary of five trials artifacts removal results (CC in Percent)	193
5.3	SNR values for proposed methods	194

## LIST OF FIGURES

<b>FIGURE NO</b>	<b>TITLE</b>	<b>PAGE NO</b>
1.1	Interconnected Neurons	2
1.2	Structure of Human Brain	3
1.3	A Four Second Sample of EEG Data	5
1.4	First EEG Recording by Hans Berger	6
1.5	The 10-20 System of Electrode Placement	7
1.6	Brain Wave Samples with Dominant Frequencies	9
1.7	Artifacts Waveforms	11
2.1	A Framework for using ICA	26
2.2	Illustration of Mixing and Separation System for ICA	27
2.3	Diagram to Illustrate the Operation of JADE	59
2.4	Structure of the Infomax ICA System	64
2.5	EEG Recording of a Subject During Mental Multiplication	74
2.6	Independent Components Obtained using Fast ICA.	75
2.7	Independent Components Obtained using JADE.	75
2.8	Independent Components Obtained using Infomax	76
2.9	Independent Components Obtained using Extended Infomax	77

<b>FIGURE NO</b>	<b>TITLE</b>	<b>PAGE NO</b>
2.10	Computation time curve for ICA algorithm	79
2.11	Entropy curve for EEG after removal of EOG	80
2.12	Signal to noise ratio curve for ICA methods	81
3.1	Block Diagram of Adaptive Filter	87
3.2	Structure of an Adaptive Filter	88
3.3	Adaptive Filters Cascade	90
3.4	Adaptive noise canceller scheme	91
3.5	Single Layer Feed Forward Network	98
3.6	Multi-Layer Feed Forward Network	99
3.7	Supervised Learning	101
3.8	Unsupervised Learning	102
3.9	Architecture of Feed Forward Back Propagation Network	103
3.10	The First Model of Fuzzy Neural System	113
3.11	The Second Model of Fuzzy Neural System	113
3.12	Structure f Neuro-Fuzzy Filter	116
3.13	Simple Neuron Set	119
3.14	AND Fuzzy Neuron	121
3.15	OR Fuzzy Neuron	122
3.16	Implication-OR Fuzzy Neuron	123
3.17	Kwan and Cai's Fuzzy Neuron	124
3.18	Kwan and Cai's Max Fuzzy Neuron	125
3.19	Kwan and Cai's Min Fuzzy Neuron	126
3.20	Result of Adaptive Filter	128

<b>FIGURE NO</b>	<b>TITLE</b>	<b>PAGE NO</b>
3.21	Result of Neuro-Fuzzy Filters	130
3.22	Result of Neuro-Fuzzy Filters (EEG+ECG)	131
3.23	Result of Neuro-Fuzzy Filters (EEG+EMG)	132
3.24	Result of Neuro-Fuzzy Filters (EEG+EOG)	133
3.25	SNR curve for Neuro-Fuzzy Filtering	134
3.26	PSD Of Neuro-Fuzzy Filters	134
3.27	Correlation Plot using Neuro-fuzzy Filters	135
3.28	Power Spectral Density Plot	136
3.29	SNR curve between adaptive and Neuro-fuzzy Filter	137
4.1	Wavelet Families	149
4.2	Adaptive Noise Cancellation	153
4.3	Flow Chart	154
4.4	Threshold Families	163
4.5	EEG Signal	166
4.6	Contaminated Signal	167
4.7	Corrected EEG Signal	167
4.8	Power Spectral Density	168
4.9	Correlation Plot	168
4.10	SNR curve for corrupted signal and denoised signal	170
5.1	Decomposition of the Signal	178
5.2	Original EEG	187
5.3	Trial 1 EEG With EOG Signal	188

<b>FIGURE NO</b>	<b>TITLE</b>	<b>PAGE NO</b>
5.4	Trial 1 Power Spectra Plot	188
5.5	Trial 2 EEG With EOG Signal	189
5.6	Trial 2 Power Spectra Plot	189
5.7	Trial 3 EEG With EOG Signal	190
5.8	Trial 3 Power Spectra Plot	190
5.9	Trial 4 EEG With EOG Signal	191
5.10	Trial 4 Power Spectra Plot	191
5.11	Trial 5 EEG signal With EOG Signal	192
5.12	Trial 5 Power spectra plot	192
5.13	SNR and CC curve for five trial using multiwavelet	194
5.14	SNR curve for corrupted signal and proposed artifacts removal methods	195
5.15	Correlation plot for noisy signal and proposed artifacts removal methods	195
5.16	Power spectral density plot for proposed methods	196

## LIST OF SYMBOLS AND ABBREVIATIONS

### **SYMBOLS**

f	-	Activation Function
N	-	Additive Noise
w	-	Arbitrary Vector
$\gamma_{xx(n)}$	-	Auto correlation function
$w_0$	-	Bias Weight
*	-	Complex Conjugate
$R^x$	-	Covariance Matrix
$r_{xx}(n)$	-	Cross-correlation Function
F(M)	-	Cumulant Matrix
Cum(x <sub>i</sub> ,y <sub>i</sub> )	-	Cumulants
D	-	Diagonal matrix
D	-	Distortion
n	-	Eighen values
H(y)	-	Entropy
e(n)	-	Error Signal
y	-	Estimated Random Variable
S	-	Estimation of the Matrix
E[.]	-	Expectation Operation
k	-	Filter Coefficient Vector
$H_k, G_k$	-	Filter Coefficients
L	-	Filter Order
K <sub>4</sub>	-	Fourth order cumulant
$\theta, \delta, \alpha, \beta$	-	Frequency ranges
$y_{gauss}$	-	Gaussian Random Variable
v	-	Gaussian Variable

$Y_{\text{gauss}}$	-	Gaussian random variable vector
$p(u)$	-	Guassian Model
$H$	-	Hessian Matrix
$H$	-	Hessian Matrix
$I$	-	Identity Matrix
$Z(t)$	-	Input
$\pi$	-	Input Information
$x(n)$	-	Input Signal
$A^{-1}$	-	Inverse Mixing Matrix
$p(y)$	-	Joint Density
$J_{\text{JADE}}$	-	Jointed diagnalisation
$\lambda_1, \lambda_2$	-	Laggrangian Multipliers
$\lambda$	-	Laplacian density
$n, m$	-	Latent Matrix
$n$	-	Latent Variables
$m \times n$	-	Matrix Size
$\mu$	-	Mean
$\delta(\text{mad})$	-	Median absolute deviation
$\mu(x)$	-	Membership function
$\mu v$	-	Microvolt
$A$	-	Mixing Matrix
$P_{\text{noise}}$	-	Noise power signal
$G$	-	Non-quadratic Function
$K$	-	Normal Commuting Matrices
$m$	-	Observed Signals
1-D	-	One dimensional signal
$X_{\text{or}}$	-	Original Signal
$P_{\text{signal}}$	-	Original Signal Power
$g_k$	-	Orthogonal Projection Coefficient

$y(n)$	-	Output Signal
$p_x$	-	Power of the Input Signal
$G(f)$	-	Power Spectrum
$P_x(X)$	-	Probability Density Function
$p_{y(n)}$	-	Probability density function
$P(x)$	-	Probability density random variable
$Z_c$	-	Randam Variable
$X_{re}$	-	Reconstructed Signal
$EEG_{vcc}(t)$	-	Recorded EEG
$G$	-	Rotation Angle
$R_q$	-	Rotation Matrix
$\Phi(t)$	-	Scaling function
$x_i$	-	Signal
$S$	-	Source
$V_0$	-	Subspace
$h(c)$	-	Symmetric Matrix
$T_k$	-	Threshold value
$T$	-	Time
$T$	-	Triangular Norm
$S(t)$	-	True Signal
$B$	-	Un mixing Matrix
$U$	-	Unitary Matrix
$W^n$	-	Unmixing Matrix
$\gamma$	-	Updating Parameter
$S_1, S_2, S_3$	-	Variables
$\sigma$	-	Variance
$W(n)$	-	Vector
$U_n(t)$	-	Vector -Velued Function
$\Psi(t)$	-	Wavelet function

wi	-	Weight matrix
m	-	Weighted matrix
z	-	Whitened Dataset
$w^T$	-	Whitened mixing matrix
a	-	Width and Height of the Density Function

## ABBREVIATIONS

AR	-	Auto Regressive
ARIC	-	Approximate Reasoning Based Intelligent Control
AWGN	-	Additive White Gaussian Noise
BCI	-	Brain computer Interface
BSS	-	Blind Source Separation
CCA	-	Canonical Correlation Analysis
CPF	-	Cumulative Probability Function
DSP	-	Digital Signal Processing
DWT	-	Discrete Wavelet Transform
ECG	-	Electrocardiogram
EEG	-	Electroencephalogram
EMG	-	Electromyogram
EOG	-	Electro-occulargram
ERP	-	Event Related Potential
EVD	-	Eigen Value
FFT	-	Fast Fourier Transform
FNN	-	Fuzzy Neural Network
FOBI	-	First Order Blind Identification
FPGA	-	Field Programmable Gate Array
ICA	-	Independent Component Analysis
JADE	-	Joint Approximate Diagnalization of Eigen matrices
LMS	-	Least Mean Square
MI	-	Mutual Information
MRA	-	Multi Resolution Analysis
MSF	-	Maximum Signal Fraction
NFS	-	Neuro Fuzzy System
NN	-	Neural Network

OA	-	Ocular Artifacts
PCA	-	Principal Component Analysis
PDF	-	Probability Density Function
PSD	-	Power Spectral Density
REB	-	Rapid Eye Blink
RMSE	-	Root Mean Square Error
SNR	-	Signal to Noise Ratio
SOBI	-	Second Order Blind Identification
STFT	-	Short Time Fourier Transform
TSK	-	Takagi –Sugeno-Kang

## CHAPTER 1

### INTRODUCTION

The human brain, apart from being the Centre of human nervous system, plays an incredibly a remarkable role in controlling all the physical and mental activities and considered as the most complex organ in human beings. The human brain has amazed and baffled many doctors and scientists that they devoted their entire lives in learning how the brain works. Though the recent advancements in science and medicine have provided a better understanding regarding the inner-working of the brain, mysteries regarding many of its simplest achievements are not yet unfolded. Indeed, the most powerful supercomputers are unable to compete with the computational power of the human brain due to its incalculable complexity. The powerful chess computer which owned the pride of defeating the world chess champion Garry Kasparov in 1997 was only capable of a mere 1/30th of the estimated power of the human brain.

The human brain is composed of nearly 100 billion neurons, thus forming an enormous network as shown in Figure 1.1. They are not only interconnected with each other but also capable of communicating with each other through their axons using small electric impulses in the order of  $\mu\text{V}$ .



**Figure 1.1 Interconnected neurons**

During each and every mental activity, variations in the electric potential of the active regions of the brain take place. These small electric variations, when summed over a region, gives a potential variation in the space, these variations are then decomposed in a series of electric maps. Thus, any mental activity in the human brain can be studied as a sequential continuation of brain electrical states.

The recordings of the electrical activity of the brain provide knowledge regarding those dynamic functional states of the brain. They help to identify the mental tasks that may occur at the moment and help even to recognize the mental activity that may occur later (Fakhreddine Karray et al 2008).

Though human beings and all other mammals have a similarity in the general structure of the brain, the human brain is over three times larger than the brain of a typical mammal with an equivalent body size. The cerebral cortex in the human brain which is a convoluted layer of the neural tissue covering the surface of the forebrain, occupies nearly 85% of the human brain, thus providing a major contribution for this expansion in size. Apart

from this, the expanded frontal lobes associated with executive functions such as self-control, planning, reasoning and abstract thought and the portion of the brain devoted to vision and the occipital lobe is also greatly enlarged in human beings.

The human brain continuously receives sensory information via the neurons, analyzes the data and then responds instantly. Thus, the human brain plays a significant role in regulating and monitoring the body's actions and reactions. Some of the major functions like breathing, heart rate and other autonomic processes independent of conscious brain functions are controlled by the brainstem. The neocortex serves as the center of higher-order thinking, learning, and memory. The body's balance, posture, and the coordination of movement are controlled by the cerebellum. The figure 1.2 shows the structure of human brain.



**Figure 1.2 Structure of human brain**

The human brain is placed in a well-protected structure formed by the skull and suspended in the cerebrospinal fluid. Though it is isolated from the bloodstream by the blood-brain barrier, the human brain is easily affected by many diseases and damages. Closed head injuries such as a blow to the

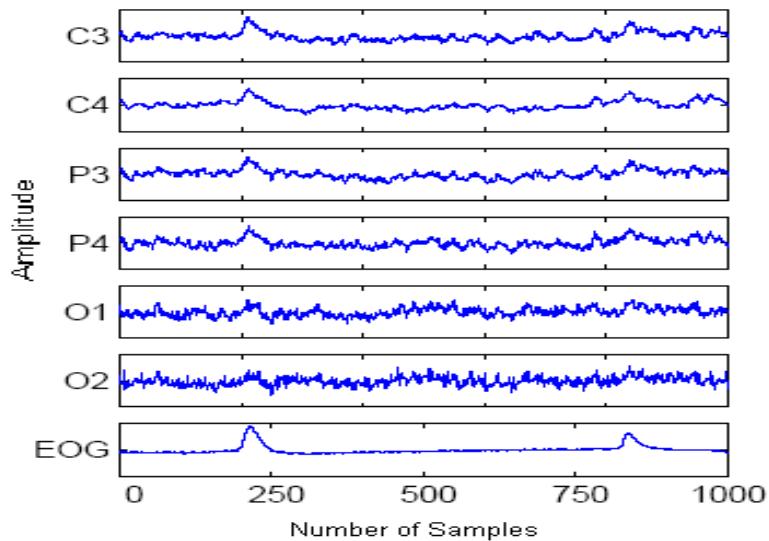
head or a stroke may prove fatal to the human brain. The human brain is immune sensitive to poisoning by a wide variety of chemicals that can act as neurotoxins (Jonathan et al 2008).The biological barriers protect the human brain from serious infections. Parkinson's disease, multiple sclerosis, and Alzheimer's disease are some of the degenerative disorders which are likely to affect the human brain. Though a number of psychiatric disorders such as schizophrenia and depression are thought to be associated with brain dysfunctions, the nature of such brain anomalies is not yet well understood.

## **1.1 ELECTROENCEPHALOGRAPHY**

A measurement of the continuous brain-wave patterns or electrical activity of the brain, as recorded with the placement of small metal discs called electrodes positioned in a standardized pattern on the scalp. The resulting tracing reflects the summation of the activity of millions of individual neurons. The voltage and frequency is interpreted and it is useful for assessing brain death, seizure activity, and for determining stages of sleep.

### **1.1.1 The Evolution of EEG**

EEG is a brain wave imaging technique used to measure the spontaneous electrical activity of the brain over a short period of time via the metal electrodes placed on the scalp and the conductive media (Teplan 2002). EEG plays a vital role in detecting the abnormalities related to the electrical activities of the brain such as epilepsy, sleep disorders and brain tumours.



**Figure 1.3 A Four Second Sample of EEG Data.**

Figure 1.3 shows the four second sample of EEG data. Electrocardiogram is the measurement of EEG by placing electrodes directly on the exposed cerebral cortex. The usage of depth probes in EEG measurement is known as Electrogram. Thus, electroencephalographic reading can be recognized as a complete non-invasive procedure that allows the repeated usage to patients, normal adults, and children with virtually no risk or limitation.

The electrical activity of the brain was first discovered by an English physician named Richard Caton in 1875 (Teplan 2002) from the exposed cerebral hemispheres of rabbits and monkeys. In order to amplify the electrical currents in the brain, he utilized an invention of Lord Kelvin who invented the famous absolute temperature theory. Caton observed that when light is introduced in an animal's eye, variations in the electrical activity of the brain occurred (Bickford 1987). He further observed that these changes occurred only in the side of the brain which is opposite to the stimulated eye.

The EEG signals were first recorded on paper by Dr. Hans Berger who used his ordinary radio equipment to amplify the electrical activity of the brain so that the electrical signals can be recorded on a paper. Shown in Figure 1.4.

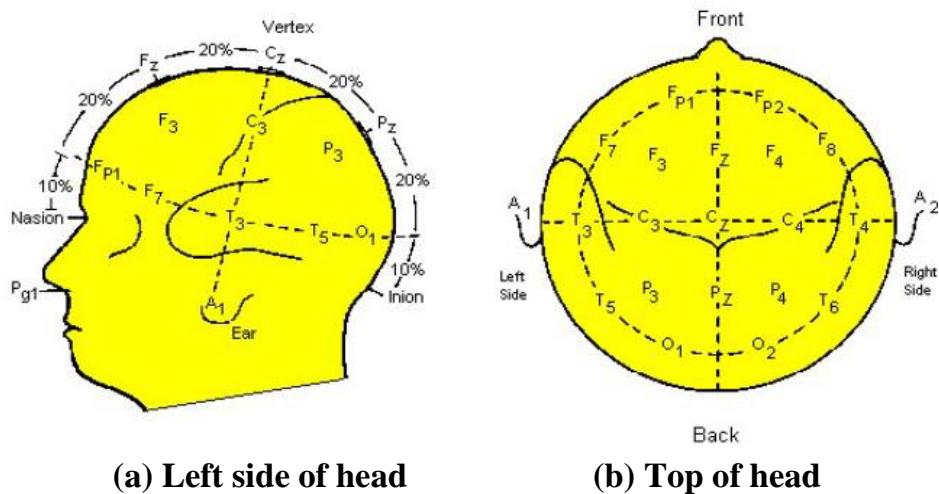


**Figure 1.4 First EEG recording by Hans Berger**

### **1.1.2 Electrode Placement**

The EEG signals are measured along the skull by the use of electrodes. The “10-20 system” is an internationally recognized guide system for the placement of the electrodes on the human scalp (Teplan 2002). The relationship between the location of an electrode and the underlying cerebral cortex area is the basis for the 10-20 system. Figure 1.5(a) shows left side of head 10-20 system of electrode placement. The 10-20 system is derived from the fact that the distances between the adjacent electrodes are either 10% or 20% from one of the four anatomical landmarks on the head. These four landmarks are comprised of two landmarks at the front (the nasion, or bridge of the nose) and back (the inion, or bump at the back of the head), and two landmarks on the right and left sides (the preauricular points, or depressions in front of the ears above the cheekbones). There are odd number of electrodes on the left and even number of electrodes on the right with a letter indicating the anatomical area in this standard 10-20 system. The Frontal, Temporal, Central, Parietal and Occipital lobes of the brain are differentiated and identified by the letter. The electrodes on the midline are represented by the letter z. The hemisphere locations are identified by the numbers 1 to 8.

The electrode positions on the right hemisphere are identified by the even numbers and the electrode positions on the left hemisphere are identified by the odd numbers. Figure 1.5(b) shows the top of head 10-20 system of electrode placement. The electrodes are considered to be closer to the midline between the hemispheres when the number is small.



**Figure 1.5 The 10-20 System of Electrode Placement**

### 1.1.3 Brain wave classification

On the basis of the frequency ranges, the EEG waves are mostly classified into four major types of continuous sinusoidal waves. There is no exact frequency range for each type of EEG wave (Scott Makeig 1997). Figure 1.6 shows the brain wave samples with dominant frequencies.

#### 1.1.3.1 Delta

The Delta waves have frequency within the range 0-4Hz. These are the slowest waves having the highest amplitude. These waves occur in adults during deep sleep and in infants. It may occur focally with subcortical lesions and also in general distribution with diffuse lesions and certain encephalopathy. The occurrence of these waves in awake state indicates serious organic brain disease.

### **1.1.3.2 Theta**

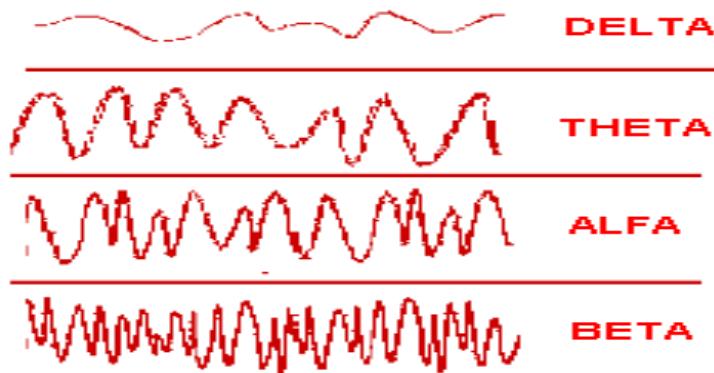
These waves have frequency between 4-8Hz width an amplitude larger than alpha waves seen in parietal and temporal regions in children. These waves are associated with drowsiness, childhood, adolescence and young adulthood. Hypertension can sometimes produce the same frequency of Theta waves. Which can be seen during hypnagogic such as trances, hypnosis, deep day dreams, lucid dreaming and light sleep and preconscious state just upon waking, and just before falling sleep.

### **1.1.3.3 Alpha**

Alpha waves have frequency ranges between 8-12Hz and are the most prominent component of EEG. Alpha waves occur when the eyes are closed while it attenuates with drowsiness and open eyes on both the sides in the posterior side and in the dominant side with higher amplitude. An alpha like normal variant is called as  $\mu$  (mu) rhythm which occurs in the motor cortex (cerebral scalp) and attenuates with the movement of the limbs or mental imaginary movement.

### **1.1.3.4 Beta waves**

Beta waves occur within the frequency range 13-30Hz and are characterized with low amplitude and varying frequencies. They occur symmetrically on both sides in the frontal area during active, busy or anxious thinking and active concentration. The cerebral cortex remains in an excitable condition under the effects of various pathologies or drugs due to which the beta waves get gradually enhanced and the area of beta waves is expanded (Maan Shaker 2005).



**Figure 1.6 Brain Wave Samples with Dominant Frequencies**

## 1.2 EEG ARTIFACTS

The EEG signals are recorded along with the electrical activity generated by the other organs in the human body (physiological) and from external sources (extraphysiological). These electrical activities are nothing but artifacts which create a problem in EEG analysis (Ashish Raj et al 2012). Figure 1.7(a) shows the clean EEG .Some of the commonly identified EEG artifacts are:

### 1.2.1 Physiological artifacts

The Physiological Artifacts are mostly classified into the following major types:

#### 1.2.1.1 Muscle artifacts

When the electrical activity in the muscles are higher than the electrical activity in the brain, the muscle artifacts occur and it is the most common type of EEG artifacts which is shown in Figure 1.7(b). The muscle artifacts are easy to be recognized and are mostly characterized by shorter duration, different shape and higher frequency than the brain electrical activity.

#### **1.2.1.2 Eye movement artifacts**

The human eye also contributes to a major type of artifacts called as Eye Movement Artifact acting as an electric dipole whose positive pole lies at the cornea and the negative pole at the retina. When the patient blinks his eyes, the eyeball rotates backward within the head known as Bell's phenomenon. This causes the poles to move in relation to the electrodes which produce a deflection in the EEG which is shown in Figure 1.7(c), eye blink artifacts is shown in Figure 1.7(d).

#### **1.2.1.3 ECG artifacts**

The muscles in the heart which are used to pump blood inside the human body contribute to a particular type of artifacts called as ECG Artifact more common in people who have short and wide neck. The ECG artifacts can easily be recognized in background EEG but, when the patient's condition is abnormal, both EEG and ECG obscure the underlying cerebral activity.

#### **1.2.1.4 Pulse artifacts**

The human circulatory system produces another type of artifact called Pulse Artifact when the electrodes are placed close to the pulsating blood vessels. Figure 1.7(e) shows pulse artifacts.

#### **1.2.1.5 Respiration artifacts**

The artifacts produced by the human respiratory system are called as Respiratory artifacts and are of two types. Those which occur during inhalation and exhalation are slow or sharp waves but those that occur during the body movement related to respiration are slow rhythmic waves. These rhythmic artifacts mostly intervene during the monitoring of abnormal EEG activity.

### 1.2.2 Extraphysiological Artifact

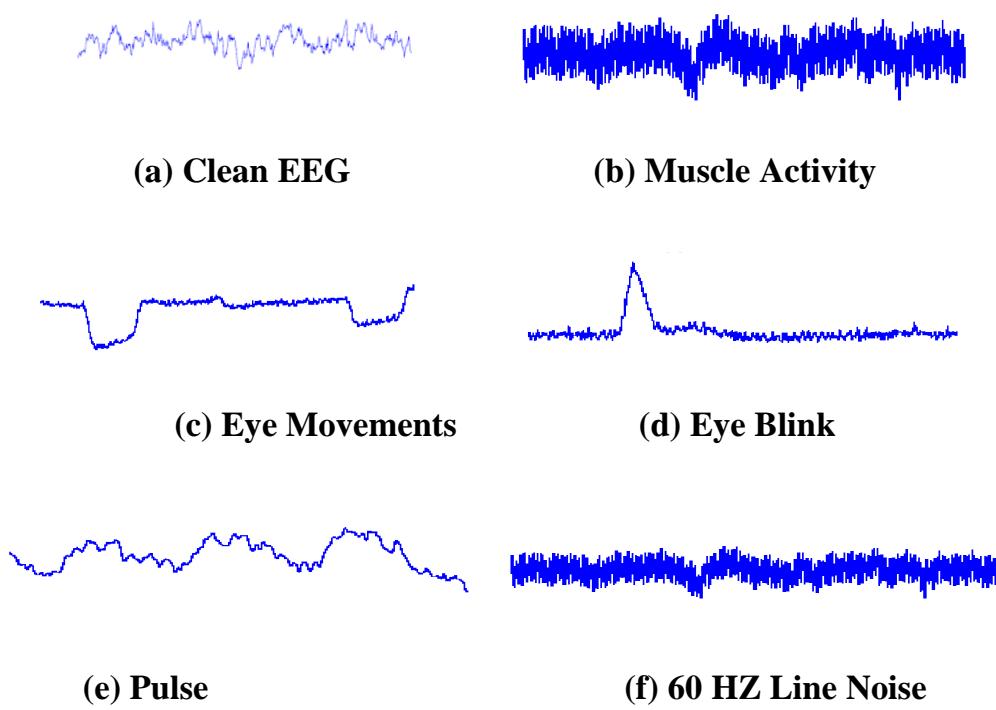
The extraphysiological artifacts are commonly classified into the following types.

#### 1.2.2.1 AC artifacts

The main power in the recording equipment or medical equipment or the surrounding equipment such as lamps contributes to an artifact called AC artifacts which are shown in Figure 1.7(f).

#### 1.2.2.2 Others

The movement of other people around the patient, gravity fed intravenous infusion (drip), infusion motors, ventilators, radio and TV and other electronic devices are other sources of artifact generation.



**Figure 1.7 Artifact Waveforms**

### **1.3 MOTIVATION FOR THE THESIS**

The major drawback in the removal of artifacts is the accuracy in the classification of artifacts. To overcome this problem, the usage of important techniques such as Independent component analysis, Haar wavelet transform, Neuro fuzzy filter and Multiwavelet transform has improved the accuracy in the classification of artifacts.

It is impossible to obtain complete information regarding the causes for the real world problems. This difficulty is mainly due to the system complexities and noninvasive techniques employed by scientists and engineers to analyze the real world problems. Such problems can be analyzed using signal and image processing techniques but they tend to be blind due to the unknown origin of signals. In the earlier days, EEG signals were removed from artifacts using training techniques. In modern real time systems, blind signal processing techniques play a crucial role adopted for research in many areas like biomedical engineering, medical imaging, speech enhancement, remote sensing, communication systems, exploration seismology, geophysics, econometrics, data mining, sensor networks and so on.

The three major areas in blind signal processing are Blind Signal Separation and Extraction, ICA and Multichannel Blind Deconvolution and Equalization. The above techniques can also be applied to the other two blind signal processing areas. These artifacts removal techniques turn up to be much interesting adopted as the central domain of this present work.

### **1.4 LITERATURE REVIEW**

One of the most commonly employed techniques in the detection and removal of artifacts is discarding the affected segments of EEG. Once an artifact is detected, a segment of fixed length, usually one second is discarded

from the time when the artifact was detected. A sudden increase in voltage above a threshold, usually above 100  $\mu$ V, serves as an indication for the occurrence of eye blink in an eye movement artifacts. All the other physiological and extra physiological artifacts are either ignored or manually marked by a practitioner and discarded. However, the amount of data available for analysis is reduced to a great extent due to the process of discarding the segments of EEG.

The artifacts that occur due to eye blinks are the first to be removed from the EEG signals. The most common technique that is employed in artifacts removal is regression that requires the proper measure of the artifact signal so that this unwanted signal can be ruled out or subtracted the desired EEG signal. In the removal of eye blink artifacts, regression using the Electroculogram channel was attempted in both time and frequency domain (Hillyard and Galambos 1970, Verleger and Gasser 1982). But EOG is contaminated with EEG signals. Therefore, regression causes the removal of desirable EEG signals which is the major disadvantage of this technique.

Independent Component Analysis (ICA) can be effectively implemented in the nearly guassian signals. Dijuwari successfully demonstrated the usage of ICA in the separation of EMG from the affected EEG signals. ICA can be employed in the separation of any signals if the signals possess distributions close to guassian. The major working principle of ICA is the estimation of unmixing matrix which allows the estimation of independent components. This shows that the estimated independent components are nothing but the linear combination of recorded data possessing some original sources when the number of sources is higher than the number of recordings. In case the original sources are predominant, the estimated independent components similarity to the original sources. This clearly shows that when the number of recordings is less than the total

number of sources (artifacts sources), ICA is capable of separating the artifact components with higher magnitude.

Due to the recent advancements in the analytical techniques, the multivariate statistical analysis techniques such as Principal Component Analysis (PCA) are implemented in the detection and removal of artifacts from EEG (Berg and Scherg 1991). PCA assumes that the signals are temporally and spatially uncorrelated.

Seungjin Choi et al blind Source Separation techniques (BSS) are indulged in the process of separating the source EEG signals into separate components and later construct the EEG signals. ICA is one of the most commonly employed BSS techniques. The fundamental principle of ICA is that it blindly separates mixtures of independent sources into independent components. ICA is most commonly employed in the removal of ocular artifacts from EEG signals. Preliminary studies indicate that ICA can be utilized to increase the strength of motor-related signal components in the Mu rhythms. ICA plays a major role in the removal of artifacts in BCI systems. Several artifacts such as EOG, EMG and ECG artifacts can be effectively removed from EEG using BSS methods. The BSS method does not necessitate the usage of reference artifacts for the removal of artifacts from the EEG signals. This is the major advantage of using BSS methods. However, the major limitation is that these methods require visual inspection for the detection of artifact signals.

Principal component analysis (PCA) can be developed from many different points of view, but it is most useful in the context of the artifact removal to view PCA as an optimization problem. PCA finds a linear transformation of a data set that maximizes the variance of the transformed variables subject to orthogonality constraints on the transformation and transformed variables (Berg and Scherg 1991).

Once the artifact signals have been identified, the automatic removal of the artifact signals is made possible by implementing online filtering systems. One of the approaches suggests that the artifacts can be recognized by measuring the structure of the signal that can be measured using fractal dimension and a metric based on Auto-Regressive (AR) coefficient (Cichoki and Vorobyov 2000). For example, the eye blinks and heart beats possess consistent fractal dimensions. Jung has described that each of the artifacts is characterized by distinct spectral structure, which allows the automatic detection of the artifact signals (Verobyov and Cichocki 2002). Jung (Jung and Humphries, 2000). Kalman filters and extended Kalman filters have successfully been implemented in the artifacts removal. But the success of this method depends on the type of the artifacts. One second windows containing muscle and movement artifacts can be successfully identified using this method.

The noise in satellite imagery can effectively be reduced using maximum signal fraction (MSF) approach. In order to maximize the signal-to-noise ratio, a linear transformation is obtained by solving an optimization problem. This method is based on a set of source signals  $S$  corrupted by additive noise  $N$  generate the data  $X$  as follows in Equation (1.1).

$$X = S + N \quad (1.1)$$

However, MSF problem requires the estimation of the noise covariance.

The most common method implemented in extracting similarity between two data sets is Canonical correlation analysis (CCA). This method is based on the assumption that a set of EEG observations can be partitioned into two sets namely  $X$  and  $Y$ , usually representing the left hemisphere electrodes and right hemisphere electrodes. Two linear transformations

corresponding to X and Y have been calculated by CCA. These linear transformations maximize the correlation between X and Y in the new coordinates.

The multivariate time series can be decomposed as a set of signals using the method of delays. The high sampling rates allow the detection of delays due to the signal propagation across the scalp.

## **1.5 NEED FOR THE THESIS**

Artifact removal is the process of identifying and removing artifacts from brain signals. Common methods for removing the artifacts in EEG signals are classified into (i) Principal Component Analysis (PCA) (ii) Canonical Correlation Analysis (CCA) (iii) Regression Method and (iv) Blind source separation

- **Principal Component Analysis**

Lagerlund et al (2009) used Principal Component Analysis (PCA) to remove the artifacts from EEG. It outperformed the regression based methods. However, PCA cannot completely separate ocular artifacts from EEG, when both the waveforms have similar voltage magnitudes. PCA decomposes the components into uncorrelated, but not necessarily independent components that are spatially orthogonal and thus it cannot deal with higher-order statistical dependencies.

- **Canonical Correlation Analysis**

CCA is used as a Blind Source Separation technique (BSS) for artifacts removal from EEG signal. It measures the linear relationship between two multi-dimensional variables, by finding two bases and bases are

optimal with respect to correlation. CCA method has considerable amount of spectral error and thus it cannot be implemented in real time.

- **Regression Method**

It is based on complex regression analysis. It is suitable for handling transfer of EOG activity to EEG which can have different frequency and phase characteristics, because the regression formula is used in frequency domain. This technique is demanding because it requires quantitative data relating to several thousand individuals. Implementing the data collection can be time consuming and expensive.

- **Blind Source Separation**

BSS techniques separate the EEG signals into components that “build” the EEG signals. They identify the components that are attributed to artifacts and reconstruct the EEG signal without these components. It has been widely applied to remove ocular artifacts from EEG signals (Amari et al 1996). They usually need prior visual inspection to identify artifact components.

The major limitation in all these methods is that either the extracted EEG signal may contain noise or the discarded noise signal may contain desirable EEG signal. In this thesis, an attempt is made to improve the signal to noise ratio of the extracted EEG signal, and also to ensure that the removed noise signal does not contain the EEG component.

Independent Component Analysis observed from the literature (Acharya 2008) that JADE performs better than the other ICA Algorithms namely Fast ICA, Infomax, extended infomax in terms of execution time only. The JADE Algorithm uses significant Eigen pairs of the cumulant

tensor  $F(M)$  to find out the estimated values of independent components. In this Improved JADE Algorithm, the tensor Eigen value decomposition is considered as more of preprocessing step. The Eigen value decomposition can also be viewed as diagonalization. Improved JADE provides better results among with four ICA algorithms in terms of their convergence speed, entropy and signal to Noise ratio. Even though, the fundamental limitation of ICA is that all but one of the underlying sources must be non-Gaussian to be completely recovered, as otherwise the central limit theory does not apply. Furthermore, the independent components can only be calculated up to their sign and scale.

Neuro-fuzzy systems which are an integration of neural networks and fuzzy logic. The computational process envisioned for neuro-fuzzy systems starts with the development of a “fuzzy neuron” based on the understanding of biological neuronal morphologies, followed by learning mechanisms (Krishnaveni et al 2006). This leads to the functions of a fuzzy neural computational process. Development of fuzzy neural models motivated by biological neurons models synaptic connections which incorporates fuzziness into neural network. Development of learning algorithms is the method of adjusting the synaptic weights .In this report Neuro fuzzy filter gives the better results in terms of Signal to Noise ratio and correlation factor compare with IJADE. Even though it has some limitation, the various artifacts mixed in the EEG signal cannot be filtered directly because they pass through the human body and turn into an interference component. This interference component cannot be estimated directly because the spectrum of the EEG signal and the interference signal overlap each other and also because of the characteristics of noise and the EEG signal which vary with time.

The Haar transform which serves as a prototype for all other wavelet transforms is the simplest of all the transforms. A discrete signal is decomposed into two sub signals of half its length using Haar transform. One of the sub signals denotes the running average of trend while the other denotes the running difference or fluctuation. The selection of threshold and thresholding function plays a vital role in signal denoising .In order to discard the noise coefficients efficiently, a proper thresholding function is essential. The thresholding function can be implemented to retain the wavelet coefficients. The retained the wavelet coefficients represent the de-noised signal while the discarded coefficients represent the noise signals. Inverse wavelet transform is implemented to obtain the denoised EEG signal. In this work, Ocular artifact zones are identified by the use of applied wavelet based adaptive thresholding algorithm, which prevents the removal of background EEG information.Haar wavelet transform gives the better result in terms of signal to Noise ratio compare with Neuro fuzzy filter. The technical disadvantage of the Haar wavelet is that it is not continuous, and therefore not differentiable. Haar wavelet transforms have single scaling function and single wavelet function. It is in the form of square (Bukhari et al 2011). So, all the signal to be analysed in the square form only. It gives better representation of approximation only. Detailed representation is very less. To overcome this disadvantage Multiwavelet Transform is used in this work. Multiwavelet transform represent both approximation and detail better than wavelet transform.

Multiwavelet has several advantages in comparison with scalar wavelet. The features such as compact support, orthogonality, symmetry, and higher order approximation are known to be important in signal processing. In this method thresholding technique is used for signal de-noising. Decomposing a signal using the wavelet transform, a set of wavelet coefficients that correlates to the high frequency sub bands. These high

frequency sub bands consist of the details in the data set. If these details are small enough, they might be omitted without substantially affecting the main features of the data set. The de-noising of EEG signal is carried out by using different combinations of threshold limit, thresholding function and window sizes. Choice of threshold limit and thresholding function is a crucial step in the de-noising procedure, as it should not remove the original signal coefficients leading to loss of critical information in the analyzed data. Because of using this transform the artifacts in the EEG signal could be removed without loss of information. Compare to previous methods, multiwavelet transform has outperformed IJADE, Neuro fuzzy filter and wavelet transform as far as SNR and correlation factor concerned.

## **1.6 SCOPE OF THE THESIS**

The focus of this work is to use the different novel approach methods like Improved JADE algorithm, neuro fuzzy filter, Haar wavelet transform and Multiwavelet transform in artifact removal in order to speed up signal processing and improve the SNR and correlation factor.

Hence, the objectives of this thesis are identified as follows:

- **Independent Component Analysis :** To create a framework to accommodate Improved JADE Algorithm to estimate the convergence speed of the algorithms in removal of artifacts from EEG data.
- **Neuro fuzzy filter:** To apply Neural network and fuzzy logic based filter for the removal of artifacts evaluate the performance of extracted EEG..

- **Haar Wavelet transform:** Create a new method to remove artifacts from EEG signal based on Haar transform and frequency analysis.
- **Multiwavelet transform:** Generate a method thresholding technique is used for signal de-noising

## 1.7 STRUCTURE OF THE THESIS

The thesis is organized as follows:

The Introduction chapter gives the explanation of brain and its structure. It also provides the general introduction EEG and its artifacts removal techniques. In this chapter, the motivation, scope of the research work and review of the related work, this provides the ground work for the research.

**Chapter 2** describes the basic principle behind the independent component analysis technique is described. The contrast functions for different routes to independence are clearly depicted. Different existing algorithms for ICA are briefly illustrated. This chapter discusses Improved JADE Algorithm, the functions that measure the non-Gaussianity of any dataset which leads to the gradient algorithm to maximize the non-gaussianity of datasets which in turn is the basis of Fast-ICA algorithm. It explains the higher order cumulants followed by a discussion of cumulant tensors. The role of cumulant tensors in the ICA algorithm IJADE is discussed. IJADE presents the better results in terms of their convergence speed and entropy compare with other ICA.

In **Chapter 3** illustrates the principle of Haar transform is described clearly. This transform cross-multiplies a function against the Haar wavelet with various shifts and stretches, like the Fourier transform cross-

multiplies a function against a sine wave with two phases and many stretches. The Haar transform can be thought of as a sampling process. It provides shortest path and time consumption is less. In this chapter , the Haar wavelets higher orders are used to decompose the recorded EEG signal and to detect the exact moment when the state of the eyes changes and on subsequent section to eye-blinks and movements of the eyeballs as well.

In **Chapter 4** discusses the principle behind the neuro fuzzy system is depicted clearly. In the field of artificial intelligence, neuro-fuzzy refers to combinations of artificial neural networks and fuzzy logic. The strength of neuro-fuzzy systems involves two contradictory requirements in fuzzy modeling: interpretability versus accuracy. In practice, one of the two properties prevails. The neuro-fuzzy in fuzzy modeling research field is divided into two areas: linguistic fuzzy modeling that is focused on interpretability, mainly the Mamdanimodel; and precise fuzzy modeling that is focused on accuracy, mainly the Takagi-Sugeno-Kang (TSK) model. Thus by using neuro fuzzy filter the artifacts are extracted from EEG signal.

**Chapter 5** explains the details behind the multiwavelet transform. They are defined using several wavelets with several scaling functions. Multiwavelet has several advantages in comparison with scalar wavelet. The features such as compact support, orthogonality, symmetry, and higher order approximation are known to be important in signal processing. Multiwavelets provide one alternative to the wavelet transform. Multiwavelets are very similar to wavelets but have some important differences. In particular, where as wavelets have an associated scaling function and wavelet function, multiwavelets have two or more scaling and wavelet function. Multifilter construction methods are already being developed to exploit the useful properties such as ortogonality, symmetry and high order of approximation.

Because of using this transform the artifacts in the EEG signal could be removed without loss of information.

**Chapter 6** concludes with contributions made in the thesis and suggestions for further research that can be pursued on the reported work.

## CHAPTER 2

### **INDEPENDENT COMPONENT ANALYSIS**

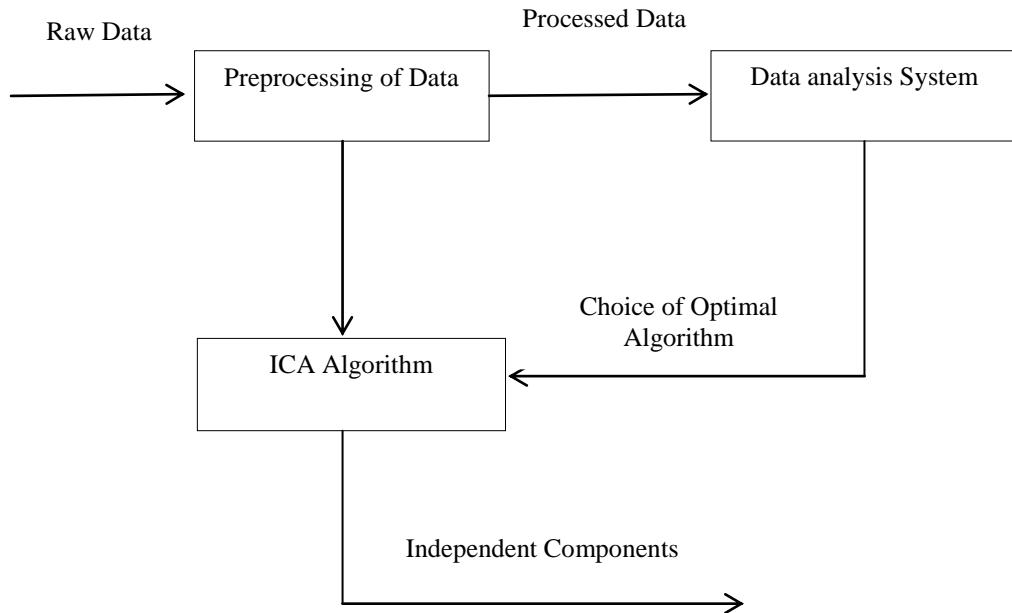
#### **2.1 INTRODUCTION**

Blind source separation (BSS) refers to the problem of recovering signals from several observed linear blind mixtures. The attraction of blind source separation modeling in signal processing is only a mutual statistical independence between source signals that is assumed. The *a priori* information such as characteristics of the source signals, the mixing matrix or the arrangement of sensors is not required. Several active signal sources that occur simultaneously at different spatial locations can then be separated by exploiting the mutual independence of sources. Among various BSS methods the Independent component Analysis (ICA) is extensively used for biomedical signal processing, as it separates the components in terms of their statistical independence (Hyvarinen, 1999).

ICA is an extension of Principal component Analysis(PCA) but it is more powerful in the field of signal analysis than PCA. In mid 90's several new ICA algorithms (Pierre Comon 1992) were introduced with impressive demonstrations focusing on problems like separating different speech signals from a mixed signal. The applications of ICA include not limiting to the fields of biomedical, telecommunications, audio and video signal processing feature

extraction, data mining, and functional time series analysis. In general, ICA technique can be regarded as a technique to separate signals from a mixture. These signals can be mixed either linearly or non-linearly. But mostly it is assumed that signals are mixed linearly and the research is based on this.

There are several ICA algorithms in use. Some of those algorithms are Fast ICA, Joint Approximate Diagonalization of Eigen matrices(JADE), First Order Blind Identification (FOBI), Maximum Likelihood and Infomax, algorithms based on Kernel methods, and algorithms that use time structure like Second Order Blind Identification (SOBI) and so on. If the independent components have non-Gaussian distribution, the above mentioned algorithms and other algorithms cannot be implemented. If data is time dependent and the independent components have Gaussian distribution, time-structured algorithms like SOBI can be used to find independent components. Hence, most of the algorithms do converge as long as independent components have non-Gaussian distribution, as it is concerned theoretically for convergence (Stone 2004). However practical observations are different. It is possible that some algorithms converge for one type of data while others do not. Convergences of different algorithms have already been explored and it has been established that some algorithms converge better and faster than the others. But it is quite possible that faster converging algorithms do not converge for some specific type of data while some other algorithm does or give better estimate of the mixing matrix while others do not. Figure 2.1 shows how to frame work for selecting the ICA algorithm.



**Figure 2.1 A Framework for using ICA**

This chapter initially gives an introduction and background to ICA, ICA theories followed by an explanation of different ICA algorithms to achieve the goal of separation besides introducing the major preprocessing steps to implement ICA algorithms.

## 2.2 ICA MODEL

A simple mathematical representation of ICA model (Acharya 2008) is as follows:

Consider a simple linear model which consists of **N** sources of **T** samples i.e.,  $s_i = [s(1)_i \dots s(t)_i \dots s_i(T)]$ . The symbol **t** here represents time but it may represent some other parameter like space. **M** weighted mixtures of the sources are observed as **X**, where  $x_i = [x_i(1) \dots x_i(t) \dots x_i(T)]$ . This can be represented as in Equation (2.1)

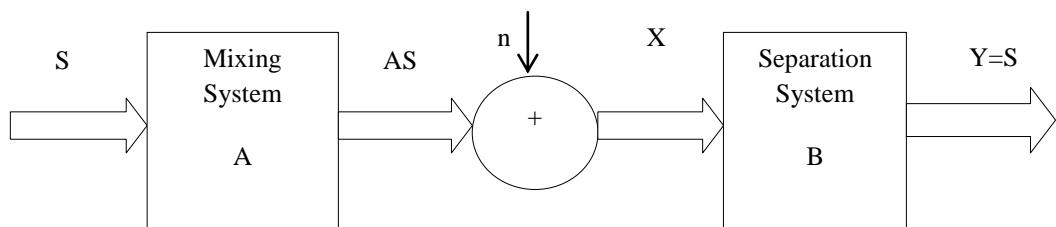
$$\mathbf{x} = \mathbf{As} + \mathbf{n} \quad (2.1)$$

$$\text{where } X = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_M \end{bmatrix}, S = \begin{bmatrix} s_1 \\ s_2 \\ \cdot \\ \cdot \\ s_N \end{bmatrix} \quad \text{and } n = \begin{bmatrix} n_1 \\ n_2 \\ \cdot \\ \cdot \\ n_K \end{bmatrix} \quad (2.2)$$

and  $\mathbf{n}$  represents the Additive White Gaussian Noise (AWGN). It is assumed that there are at least as many observations as sources i.e.,  $M \geq N$ . The  $M \times N$  matrix  $\mathbf{A}$  is represented as in Equation (2.3)

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1N} \\ a_{21} & a_{22} & \dots & a_{2N} \\ \cdot & \cdot & \cdot & \cdot \\ a_{M1} & a_{M2} & \dots & a_{MN} \end{bmatrix} \quad (2.3)$$

This relates  $\mathbf{X}$  and  $\mathbf{S}$ .  $\mathbf{A}$  is called as mixing matrix. The estimation of the matrix  $\mathbf{S}$  with knowledge of  $\mathbf{X}$  is the linear source separation problem. This is schematically shown in Figure 2.2.  $\mathbf{A}$  is the mixing matrix and  $\mathbf{B}$  is the unmixing matrix.



**Figure 2.2 Illustration of Mixing and Separation System for ICA**

The source separation problem cannot be solved if there is no knowledge of either  $\mathbf{A}$  or  $\mathbf{S}$  apart from the observed mixed data  $\mathbf{X}$ . If the mixing matrix  $\mathbf{A}$  is known and the additive noise  $\mathbf{n}$  is negligible, the original

sources can be estimated by evaluating the pseudo inverse of the matrix  $\mathbf{A}$  which is known as the unmixing matrix  $\mathbf{B}$  as shown in Equation (2.4).

$$\mathbf{BX} = \mathbf{BAS} = \mathbf{S} \quad (2.4)$$

For cases where the number of observations  $\mathbf{M}$  equals number of sources  $\mathbf{N}$  (i.e.  $\mathbf{M} = \mathbf{N}$ ), the mixing matrix  $\mathbf{A}$  is a square matrix with full rank and  $\mathbf{B} = \mathbf{A}^{-1}$ .

The necessary and sufficient condition for the pseudo-inverse of  $\mathbf{A}$  to exist is that it should be of full rank. When there are more observations than the sources (i.e.  $\mathbf{M} > \mathbf{N}$ ), there exist many matrices  $\mathbf{B}$  which satisfy the condition  $\mathbf{BA} = \mathbf{I}$ . Here the choice  $\mathbf{B}$  depends on the components of  $\mathbf{S}$  that which is interested in. When the number of observations is less than the number of sources (i.e.  $\mathbf{M} < \mathbf{N}$ ), a solution does not exist unless further assumptions are made.

On the other side of the problem, if there is no prior knowledge of the mixing matrix  $\mathbf{A}$  then the estimation of both  $\mathbf{A}$  and  $\mathbf{S}$  is known as the BSS problem. The problem defined in Equation (2.1) under the assumption of negligible Gaussian noise  $\mathbf{n}$  is solvable with the following restrictions.

- The sources (i.e. the components of  $\mathbf{S}$ ) are statistically independent.
- At most one of the sources is Gaussian distributed
- The mixing matrix is of full rank.

Above discussion leads to following definition of ICA:

**Definition:** ICA is a linear transformation  $S = WX$  of a multivariate signal  $X$ , such that the components  $S$  are as independent as possible in the sense of maximizing some objective function  $f(S_1, S_2, \dots, S_N)$ , which is a measure of statistical independence.

### 2.2.1 Assumptions For ICA Model

The following assumptions (Acharya 2008) made ensure that the ICA model estimates the independent components meaningfully. The first assumption is only the true requirement which ICA demands. The other assumptions ensure that the estimated independent components are unique.

- (1) The latent variables (or independent components) are statistically independent and the mixing is linear.
- (2) There is no more than one Gaussian signal among the latent variables and the latent variables have cumulative density function not much different from a logistic sigmoid
- (3) The number of observed signals,  $m$ , is greater than or equal to the number of latent variables,  $n$  (i.e.  $m \geq n$ ). If  $n > m$ , the special category of Independent Component Analysis called ICA with over-complete bases .In such a case the mixed signals do not have enough information to separate the independent components. There have been attempts to solve this particular problem but no rigorous proofs exist yet. If  $m > n$ , then there is redundancy in the mixed signals. The ICA model works ideally when  $n = m$ .
- (4) The mixing matrix is of full column rank, which means that the rows of the mixing matrix are linearly independent. If the mixing matrix is not of full rank, the mixed signals will be linear multiples of one another.

(5) The propagation delay of the mixing medium is negligible.

### 2.2.2 Preprocessing of Data for ICA

Before applying an ICA algorithm to the data, it is usually very useful to preprocess the data (Acharya 2008). In this section, some preprocessing techniques that make the problem of ICA estimation simpler and better conditioned are given.

- **Centering:** Centering is achieved simply by subtracting the mean of signal from each reading of that signal.
- **Whitening:** A covariance matrix can be formed by taking the covariance between every pair of signals and forming a matrix. The covariance matrix are square and symmetric. Perform Eigen value decomposition on the covariance matrix and then transform the data so that the covariance matrix of the transformed data is equal to the identity. This procedure is also called whitening (sphereing) since it normalizes the eigenvalues of the covariance matrix.

### 2.3 ALGEBRAIC ICA ALGORITHM

An algebraic solution to ICA is proposed by Taro Yamaguchi. This is a non-iterative algorithm but becomes extremely complex to compute when the number of sources goes more than two (Mackay 1996). For two sources separation it works very fast .Two observed signals  $x_1$  and  $x_2$  are given by linear mixture of two independent original signals  $s_1$  and  $s_2$  as in Equation (2.5).

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 & \alpha \\ \beta & 1 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} \quad (2.5)$$

where  $\alpha$  and  $\beta$  are unknown mixing rates. The algebraic solution to  $\alpha$  and  $\beta$  are given by Equation (2.6) and Equation (2.7).

$$\beta = \frac{\alpha C_2 - C_3}{\alpha C_3 - C_1} \quad (2.6)$$

$$(C_2 C_{10} - C_{11} C_3) \alpha^4 + (3C_9 C_3 - 3C_8 C_2 - C_3 C_{10} + C_1 C_{11}) \alpha^3 + \\ (3C_6 C_2 + 3C_8 C_3 - 3C_9 C_1 - 3C_7 C_3) \alpha^2 \\ + (C_5 C_3 + 3C_7 C_1 - 3C_6 C_3 - 3C_2 C_4) \alpha + (C_3 C_4 - C_1 C_5) = 0 \quad (2.7)$$

where  $C_1, C_2, C_3, C_4, C_5, C_7, C_8, C_9, C_{10}$ , and  $C_{11}$  are as shown in Equation (2.8)

$$C_1 = E[X_1^2] - \{E[X_1]\}^2, C_2 = E[X_2^2] - \{E[X_2]\}^2, \\ C_3 = E[X_1 X_2] - E[X_1]E[X_2], C_4 = E[X_1^4] - E[X_1^3]E[X_1] \\ C_5 = E[X_1^3 X_2] - E[X_1^3]E[X_2], C_6 = E[X_1^3 X_2] - E[X_1^2 X_2]E[X_1], \\ C_7 = E[X_1^2 X_2^2] - E[X_1^2 X_2]E[X_2], C_8 = E[X_1^2 X_2^2] - E[X_1 X_2^2]E[X_1], \\ C_9 = E[X_1 X_2^3] - E[X_1 X_2^2]E[X_2], C_{10} = E[X_1 X_2^3] - E[X_1]E[X_2^3], \\ C_{11} = E[X_2^4] - E[X_2^3]E[X_2] \quad (2.8)$$

where  $E[.]$  denotes the expectation operation.

$\alpha$  and  $\beta$  are obtained by solving the Equation (2.6), Equation (2.7) and Equation (2.8) with the Ferrari method. Excluding the solutions having non-zero imaginary parts and negative sizes, the proper solution is selected. Original independent signals are computed from Equation (2.7) by solving value of  $\alpha$  and  $\beta$ .

## 2.4 FAST ICA ALGORITHM

Fast ICA tends to maximize the measure of non-Gaussianity for the input dataset thus yielding independent components. This Algorithm starts with the discussion that the function that measures the non-Gaussianity which leads to the gradient algorithm to maximize the non-Gaussianity, which in turn is the basis of Fast ICA algorithm.

### 2.4.1 Measures of Non-Gaussianity

There are two functions that are used to measure the Gaussianity or non-Gaussianity of a dataset, Kurtosis and Negentropy. These two functions are the basis of algorithms based on the maximization of non-Gaussianity like Fast ICA. Explanation of these two functions is given as follows:

#### 2.4.1.1 Kurtosis

The fourth-order statistics of a random variable is called Kurtosis (Amari and Cichocki 1996). More specifically it is the fourth cumulant of a random variable. For a random variable with zero mean, the kurtosis is given by Equation (2.9).

$$\text{kurt}(x) = E\{x^4\} - 3E[\{x^2\}]^2 \quad (2.9)$$

If  $x$  also has unit variance then Equation (2.9) becomes as Equation (2.10).

$$\text{kurt}(x) = E\{x^4\} - 3 \quad (2.10)$$

For Gaussian variables, kurtosis is equal to zero. Variables having positive value of kurtosis are called super Gaussian and the one having negative kurtosis are called sub Gaussian.

Super Gaussian random variables are the one having distributions that resemble to Gaussian distribution i.e. a high value usually at zero (for zero-mean variable) and smaller values as the variable moves away from zero. Laplacian random variables are super Gaussian with probability density given by Equation (2.11).

$$P_x(x) = \frac{\lambda}{2} \exp(-\lambda|x|) \quad (2.11)$$

where  $\lambda$  determines both the variance and height of the peak of Laplacian density. Sub Gaussian variables have typically a flat probability density function. Common example is uniform random variables. For a zero-mean uniform variable the probability density function is given by Equation (2.12).

$$\begin{aligned} P_x(x) &= \frac{1}{2a}, & x \in [-a, a] \\ &= 0 & \text{elsewhere} \end{aligned} \quad (2.12)$$

where  $a$  determines the width and height of the density function.

To use kurtosis to find independent components, one starts with an arbitrary vector  $\mathbf{w}$  and calculates the kurtosis of  $y = \mathbf{w}^T \mathbf{z}$ , where  $\mathbf{z}$  is the input whitened dataset. Then using one of the gradient methods, a new vector  $\mathbf{w}_{new}$  is being calculated that gives a new value of  $kurt(y)$  moving towards more negative if the old kurtosis value was negative or more positive if it was positive until it reaches to its maxima or minima (extreme point). At that point  $y$  is the estimation of one of the original signal  $s$ . (Hyvarinen et al 2001)

The absolute value of kurtosis is widely used in estimating non-Gaussian variables in ICA and related fields. It is simple to use, both computationally and theoretically. However estimating non-Gaussianity

through maximization (minimization) of kurtosis has its disadvantages. The problem with kurtosis is that it can be very sensitive to outliers (a value far from most others in a set of data). Hence its value may depend on only few data points in the tails of the distribution, which may be erroneous or irrelevant observations. Thus in case of observed signal where there are some outliers due to some error, it is not wise to use kurtosis. Negentropy on the other hand gives a better option to estimate non-Gaussianity of datasets and that is the reason to be used as the basis of Fast ICA algorithm in this research.

#### **2.4.1.2 Negentropy**

Negentropy (Amari and Cichocki 1996) is based on the information theoretic quantity of (differential) entropy. Entropy of a random variable is defined as the degree of information that observations of a variable gives. The more randomness in a variable, the larger its entropy will be.

For a discrete random variable  $Y$ , entropy can be given by Equation (2.13) ,

$$H(Y) = - \sum_i P(Y = a_i) \log P(Y = a_i) \quad (2.13)$$

For continuous random variables, entropy is called Differential Entropy, given by Equation (2.14).

$$H(y) = - \int_{-\infty}^{\infty} P_Y(y) \log P_Y(y) dy \quad (2.14)$$

Gaussian variables have the largest entropy among all the random variables having equal variances. That means Gaussian distribution is the most random or least structured distribution.

Negentropy is a slightly modified version of differential entropy. For a random variable vector  $\mathbf{y}$  it can be given as in Equation (2.15).

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y}) \quad (2.15)$$

where  $\mathbf{y}_{gauss}$  is a Gaussian random variable vector of the same covariance matrix as  $\mathbf{y}$ .

Since Gaussian variables have the highest value of entropy, negentropy cannot be negative. It will be positive if entropy of  $\mathbf{y}$  is smaller than that of  $\mathbf{y}_{gauss}$  and zero if  $\mathbf{y}$  is a Gaussian random vector. Negentropy is the optimal estimator of non-Gaussianity as far as statistical properties are concerned but calculation of Negentropy using the definition given in Equation (2.15) can be very complex for certain data and so the approximations to calculate Negentropy are generally used.

The advantage of using negentropy or equivalently, differential entropy, as a measure of non-Gaussianity is that it is well justified by statistical theory. In fact, Negentropy, in some sense, is the optimal estimator of non-Gaussianity as far as statistical properties are concerned. The problem in using negentropy is, however, computationally very difficult. Estimating Negentropy using the definition would require an estimate (possibly non-parametric) of the pdf. Therefore, simpler approximations of negentropy are very useful, which is given in next section.

#### 2.4.2 Approximations to Negentropy

The estimation of negentropy is difficult, as mentioned above, and therefore this contrast function remains mainly a theoretical one. In practice, some approximations have to be used. Here approximations that have very

promising properties, are introduced and to be used in the following to derive an efficient method for ICA.

Classically negentropy is approximated using higher order moments. One way of the approximation is as in Equation (2.16).

$$J(y) \approx \frac{1}{12} E\{y^3\}^2 + \frac{1}{48} kurt(y)^2 \quad (2.16)$$

The random variable  $y$  is assumed to be of zero mean and unit variance.

However, the validity of such approximations may be rather limited. In particular, these approximations suffer from the non-robustness encountered with kurtosis.

To avoid the problems encountered with the preceding approximations of negentropy, new approximations were developed (Hyvärinen 1999, Amari and Cichocki 1996). These approximations were based on the maximum-entropy principle. In general to obtain the approximation as in Equation (2.16).

$$J(y) \approx \sum_{i=1}^p k_i [E\{G_i(y)\} - E\{G_i(v)\}]^2 \quad (2.17)$$

where  $k_i$  are some positive constants, and  $v$  is a Gaussian variable of zero mean and unit variance (i.e. standardized). The variable  $y$  is the estimated random variable with zero mean and unit variance, and the functions  $G_i$  are some non-quadratic functions (Hyvärinen, 1999 , Amari and Cichocki 1996). Note that even in cases where this approximation is not very accurate, Equation (2.17) can be used to construct a measure of non-Gaussianity that is

consistent in the sense that it is always non-negative, and equal to zero if  $y$  has a Gaussian distribution.

In case of using only one non-quadratic function  $G$ , the approximation in Equation (2.17) becomes as Equation (2.18).

$$J(y) \propto [E\{G(y)\} - E\{G(v)\}]^2 \quad (2.18)$$

for practically any non-quadratic function  $G$ . This is clearly a generalization of the moment-based approximation in Equation (2.16), if  $y$  is symmetric. Indeed, taking  $G(y) = y^4$ , one then obtains exactly Equation (2.16), i.e. a kurtosis-based approximation.

But it is to notify that by choosing  $G$  wisely, one obtains approximations of Negentropy that are much better than the one given by Equation (2.16). In particular, choosing  $G$  that does not grow too fast, one obtains more robust estimators. Some of the choices of  $G$  are in Equation (2.19).

$$\begin{aligned} G_1(u) &= \frac{1}{a_1} \log[\cosh(-a_1 u)] \\ G_2(u) &= -\exp\left(-\frac{u^2}{2}\right) \end{aligned} \quad (2.19)$$

where  $1 \leq a \leq 2$  is some suitable constant.

Thus approximations of Negentropy give a very good compromise between the properties of the two classical non-Gaussianity measures given by kurtosis and Negentropy. They are conceptually simple, fast to compute, yet have appealing statistical properties, especially robustness. Therefore, one can use these contrasting functions in ICA methods. Since kurtosis can be expressed in this same framework, it can still be used by ICA methods. A

practical algorithm based on these contrast functions will be presented in the following sections.

### 2.4.3 Algorithms based on Negentropy

Before discussing gradient algorithm based on negentropy that leads to the Fast ICA algorithm, it is important to discuss the condition imposed on the unmixing vectors  $\mathbf{w}_i$  which constitute the unmixing matrix  $\mathbf{W}$

Assume the general ICA model,  $\mathbf{x} = \mathbf{A} \mathbf{s}$ , with two unknown variables  $s_1$  and  $s_2$ , both having zero mean and unit variances. Assume the data is not whitened and the estimated value of vector  $\mathbf{s}$  can be given by  $\mathbf{y} = \mathbf{W} \mathbf{x}$  where  $\mathbf{W}$  is the estimate of inverse of the mixing matrix  $\mathbf{A}$ . To find out the optimization landscape of gradient algorithm, assume one of the components of  $\mathbf{y}$  say  $y_i$  as given by the Equation (2.20).

$$y_i = \mathbf{W}_i^T \mathbf{x} \quad (2.20)$$

where  $\mathbf{W}_i^T$  is the transpose of one of the columns of matrix  $\mathbf{W}$ . Substituting value of  $\mathbf{x}$  from the general model of ICA into Equation (2.20), yields the Equation (2.21) ,

$$y_i = \mathbf{W}_i^T \mathbf{A} \mathbf{s} = q^T \mathbf{s} = q_1 s_1 + q_2 s_2 \quad (2.21)$$

where  $q^T = \mathbf{W}_i^T \mathbf{A}$  . Since variances of  $s_1$  and  $s_2$  are unity, hence it can be assumed that  $y_i$  also has unit variance as shown in Equation (2.22) i.e.,

$$E\{y_i^2\} = q_1^2 E\{s_1^2\} + q_2^2 E\{s_2^2\} = q_1^2 + q_2^2 = 1 \quad (2.22)$$

Geometrically, vector  $\mathbf{q}$  is constrained to a unit circle in a 2-D plane.

Now consider the whitened input vector  $\mathbf{z} = \mathbf{V} \mathbf{x}$ . Replacing  $\mathbf{x}$  with  $\mathbf{z}$  in Equation (2.20) will result in maximizing  $\mathbf{w}_i^T \mathbf{z}$  for non-Gaussianity. From Equation (2.21), the new expression for  $\mathbf{q}$  can be given by the Equation (2.23).

$$q = (\mathbf{VA})^T W_i \quad (2.23)$$

Therefore the square of the norm of  $\mathbf{q}$  can be given as in Equation (2.24).

$$\|q\|^2 = (W_i^T \mathbf{VA})(A^T V^T W_i) = \|W_i\|^2 \quad (2.24)$$

Hence it turns out to be constraining  $\mathbf{q}$  on a unit circle that is equivalent to constraining  $\mathbf{w}_i$  on a unit circle in case of whitened data. Thus the value of Negentropy corresponding to  $\mathbf{w}_i^T \mathbf{z}$  will be maximized under the constraint that  $\|\mathbf{w}_i\| = 1$ . This constraint requires that after updating  $\mathbf{w}_i$  through algorithm it should be divided by its norm. It should be noted here that the linear combinations  $\mathbf{w}_i^T \mathbf{z}$  can be interpreted as projections of vector  $\mathbf{z}$  on the vector  $\mathbf{w}$ , thus each point on the unit sphere (unit circle in the case of 2-D as considered so far) corresponds to one of the projections of  $\mathbf{z}$ .

#### 2.4.4 Gradient Algorithm

Assume only one non-quadratic function for Equation (2.18), and then the Negentropy can be given as Equation (2.25).

$$J(y) \propto [E\{G(y)\} - E\{G(v)\}]^2 \quad (2.25)$$

Let  $y = \mathbf{w}^T \mathbf{z}$  be one of the estimated input data vector and assume  $\gamma = [E\{G(W^T Z)\} - E\{G(v)\}]$ . The gradient algorithm to maximize negentropy can be evaluated as taking the gradient of Equation (2.25) with

respect to  $\mathbf{w}$  and then normalizing the result on  $\|\mathbf{w}\|^2 = 1$ , as shown in Equation (2.26).

$$\begin{aligned}\Delta \mathbf{W} &\propto \gamma E\{G^{-1}(W^T Z)\} \\ \mathbf{W} &\leftarrow \mathbf{W} + \Delta \mathbf{W} \\ \mathbf{W} &\leftarrow \frac{\mathbf{W}}{\|\mathbf{W}\|}\end{aligned}\tag{2.26}$$

The constant  $\gamma$  gives the algorithm a ‘self-adaptation’ quality.

The algorithm can further be simplified by observing that the constant  $\gamma$  would not change the stationary points of the learning rule, only its sign affects the stability of the learning rule (or convergence of the learning rule). If there is some priori information on the distribution of the independent components available then the sign of  $\gamma$  can be estimated and used in the weight update expression instead of using and updating  $\gamma$ . The following is a brief summary of the gradient algorithm using negentropy:

- Step 1** : Center the data to make its mean zero
- Step 2** : Whiten the data to get  $\mathbf{z}$
- Step 3** : Choose an initial value of  $\mathbf{w}$  randomly that has unit norm and an initial value of  $\gamma$
- Step 4** : Update  $\Delta \mathbf{W} \propto \gamma E\{z G^{-1}(W^T z)\}$
- Step 5** : Normalize  $\mathbf{w}$  on a unit sphere, i.e.,  $\mathbf{w} \leftarrow \frac{\mathbf{w}}{\|\mathbf{w}\|}$
- Step 6** : If the sign of  $\gamma$  is not known prior, update  

$$\Delta \gamma \propto [G(W^T Z) - E\{G(v)\}] - \gamma$$
- Step 7** : If not converged, i.e., a new  $\mathbf{w}$  is not close enough to new  $\mathbf{w}$ , go back to step 4.

## 2.5 IMPLEMENTATION OF FAST ICA ALGORITHM

The main advantage of using the gradient method to maximize negentropy is that the inputs  $\mathbf{z}(t)$  can be used in the algorithm at once, thus enabling fast adaptation in non-stationary environment. However convergence is slow and depends on a good choice of learning rate  $\gamma$ .

To make this method efficient, a fast-fixed point algorithm is devised, which is also called Fast ICA for Negentropy. To understand this algorithm it should be noted that at a stable point of the gradient algorithm, the gradient must be pointing towards  $\mathbf{w}$  or it must be a scalar multiple of  $\mathbf{w}$ . This means that adding the gradient of negentropy in  $\mathbf{w}$  would not change its direction at the stable points and hence convergence can be obtained. This means that Equation (2.26) can be written as Equation (2.27).

$$\begin{aligned} \mathbf{w} &\leftarrow E\{\mathbf{z}\mathbf{G}^{-1}(\mathbf{w}^T \mathbf{z})\} \\ \mathbf{w} &\leftarrow \frac{\mathbf{w}}{\|\mathbf{w}\|} \end{aligned} \quad (2.27)$$

The coefficient  $\gamma$  is omitted because it would be eliminated by the normalization. Iteration in Equation (2.27) does not however have convergence as good as the one using kurtosis. The reason is that the non-polynomial moments ( $G$ 's) do not have same nice algebraic properties as cumulants like kurtosis. Hence to have a better convergent algorithm the iteration in Equation (2.27) has to be modified. This modification can simply be done by adding some multiple of  $\mathbf{w}$  to the both sides of the iterant term in Equation (2.27) and then changing the value of multiple to find a better convergence speed as in Equation (2.28).

$$\mathbf{w} = E\{\mathbf{z}\mathbf{G}^{-1}(\mathbf{w}^T \mathbf{z})\} \Leftrightarrow (1 + \alpha)\mathbf{w} = E\{\mathbf{z}\mathbf{G}^{-1}(\mathbf{w}^T \mathbf{z})\} + \alpha\mathbf{w} \quad (2.28)$$

Adding a multiple of  $\mathbf{w}$  to the both sides of Equation (2.27) would not change the direction of the vector. After normalization in the next step,  $\mathbf{w}$  will be constrained to the unit sphere again. A suitable value of  $\alpha$  and thus the Fast ICA algorithm can be found using Newton's method for solving Equation s. Newton's method can briefly explained as follows:

To find a maxima or minima of any function with respect to some variable, first the function is expanded using Taylor's series and the terms above the quadratic terms are dropped to keep it manageable (since higher order terms don't contribute a lot in the total value of the function). Let  $E$  (not expectation) be a cost or error function which has to be minimized around vector  $\mathbf{w}(n)$  having  $m$  elements and  $n$  being the number of iteration. The change in the cost function can be written as in Equation (2.29).

$$\begin{aligned}\Delta E(W_{n+1}) &= E(W(n+1)) - E(W(n)) \\ &\approx E(W(n)) + g^T(n)\Delta W(n) + \frac{1}{2}\Delta W(n)^T H(n)\Delta W(n) - E(W(n)) \\ &= g^T(n)\Delta W(n) + \frac{1}{2}\Delta W(n)^T H(n)\Delta W(n)\end{aligned}\tag{2.29}$$

where  $\mathbf{g}(n)= mx1$  gradient vector of cost function evaluated at  $\mathbf{w}(n)$ , and,  $\mathbf{H}(n)$  is an  $mxm$  2<sup>nd</sup> order derivative matrix of the cost function  $E(\mathbf{w}(n))$  evaluated at  $\mathbf{w}(n)$ , called Hessian Matrix. Hessian Matrix  $\mathbf{H}$  is given by Equation (2.30).

$$H = \Delta^2 E(W(n))$$

$$\begin{aligned}
& \left[ \frac{\partial^2 E}{\partial w_1^2} \frac{\partial^2 E}{\partial w_1 \partial w_2} \dots \frac{\partial^2 E}{\partial w_1 \partial w_m} \right] \\
& \left[ \frac{\partial^2 E}{\partial w_2^2} \frac{\partial^2 E}{\partial w_1 \partial w_2} \dots \frac{\partial^2 E}{\partial w_2 \partial w_m} \right] \\
= & \left[ \dots \right] \\
& \left[ \dots \right] \\
& \left[ \dots \right] \\
& \left[ \frac{\partial^2 E}{\partial w_m^2} \frac{\partial^2 E}{\partial w_1 \partial w_2} \dots \frac{\partial^2 E}{\partial w_m \partial w_1} \right]
\end{aligned} \tag{2.30}$$

Differentiating  $\Delta E(W(n))$  w.r.t  $\Delta w(n)$  to find out the minimal value of  $\Delta E$  gives the condition as in Equation (2.31).

$$\begin{aligned}
g(n) + H(n)\Delta W(n) &= 0 \\
\Rightarrow \Delta W(n) &= -H^{-1}(n)g(n) \\
\Rightarrow W(n+1) &= W(n) - H^{-1}(n)g(n) \\
\Rightarrow W(n+1) &= W(n) - \left[ \frac{\partial^2 E(W(n))}{\partial W^2} \right]^{-1} \left[ \frac{\partial E(W(n))}{\partial W} \right]
\end{aligned} \tag{2.31}$$

Expression in Equation (2.31) is the Newton's method for updating the vector  $w$  to move towards the minimization of the cost function. The advantage of Newton's method is fast convergence but it is computationally more intensive since one has to calculate inverse of Hessian matrix at each step.

In order to avoid the cost and time consuming calculation of the inverse of Hessian matrix in the Newton's method, an approximation of this method is developed that avoids the use of matrix inversion without sacrificing its essence to employ ICA algorithm (Wu and Yu 2005). The approximation of Newton's method calls for the use of Lagrangian rule for

constrained optimization. Lagrangian rule for constrained optimization can be briefly described as follows.

Assume a cost function  $E(\mathbf{w})$  ( $E$  is not expectation) which is supposed to be minimized or maximized under some constraint  $H_i(\mathbf{w}) = 0$ , where  $i = 1, 2, 3, \dots, k$ . One can write the Lagrangian function based on the given information as in Equation (2.32).

$$L(W, \lambda_1, \lambda_2, \dots, \lambda_k) = E(W) + \sum_{i=1}^k \lambda_i H_i(W) \quad (2.32)$$

where  $\lambda_1, \lambda_2, \dots, \lambda_k$  are called Lagrangian multipliers. The minimum (maximum) point of Equation (2.32) where its gradient is zero with respect to both  $\mathbf{w}$  and all of the  $\lambda_i$  gives the solution to the original constrained problem, i.e., minimization of  $E(\mathbf{w})$  under some constraint  $H_i(\mathbf{w}) = 0$ . The gradient of  $L(W, \lambda_1, \lambda_2, \dots, \lambda_k)$  with respect to  $\lambda_i$  gives the  $i^{\text{th}}$  constraint function  $H_i(\mathbf{w})$ , so putting all these to zero will give the original constraint condition. When gradient of  $L(W, \lambda_1, \lambda_2, \dots, \lambda_k)$  is taken with respect to  $\mathbf{w}$  and equate it to zero, one will get the Equation (2.33).

$$\frac{\partial L(W, \lambda_1, \lambda_2, \dots, \lambda_k)}{\partial W} = \frac{\partial E(W)}{\partial W} + \sum_{i=1}^k \lambda_i \frac{\partial H_i(W)}{\partial W} = 0 \quad (2.33)$$

Hence the minimization problem has been reduced to two sets of Equations that are much easier to solve. A possible way to solve these two sets, one given by the constraints, the other by Equation (2.33), is some appropriate iteration method like Newton iteration.

Moving back to the application of Lagrangian method to approximate Newton's method for a better algorithm for ICA using negentropy gradient, first observe that maxima of approximation of

negentropy as given in Equation (2.25) for  $y = \mathbf{w}^T \mathbf{z}$  are typically obtained at certain optima of  $E\{G(\mathbf{w}^T \mathbf{z})\}$ . Now from Equation (2.33) the optima of  $E\{G(\mathbf{w}^T \mathbf{z})\}$  for constraint  $\|\mathbf{w}\|^2 = 1$  can be evaluated as in Equation (2.34),

$$\begin{aligned} \frac{\partial E\{G(W^T z)\}}{\partial W} + \lambda \frac{\partial (\|W\|^2 - 1)}{\partial W} &= 0 \\ \Rightarrow E\{zG'(W^T z)\} + 2\lambda w &= 0 \\ \Rightarrow E\{zG'(W^T z)\} + \beta w &= 0 \end{aligned} \quad (2.34)$$

where  $H(\mathbf{w}) = \|\mathbf{w}\|^2 - 1 = 0$  is the only constraint to find out the extreme of  $E\{G(\mathbf{w}^T \mathbf{z})\}$ . To solve Equation (2.34) one can use Newton's method to find the optima with respect to  $\mathbf{w}$ . Let  $F = E\{zG'(W^T z)\} + \beta w$ , the derivative of  $F$ , i.e., the second derivative of Lagrangian function can be evaluated as in Equation (2.35),

$$\frac{\partial F}{\partial W} = E\{zz^T G''(W^T z)\} + \beta I \quad (2.35)$$

Thus the Newton iteration from Equation (2.31) can be written as in Equation (2.36).

$$W \leftarrow W - \frac{\begin{bmatrix} \frac{\partial L}{\partial W} \\ \frac{\partial F}{\partial W} \end{bmatrix}}{\begin{bmatrix} \frac{\partial F}{\partial W} \end{bmatrix}} = W - \frac{E\{zG'(W^T z)\} + \beta W}{E\{zz^T G''(W^T z)\} + \beta I} \quad (2.36)$$

To simplify the calculations, since the vector  $\mathbf{z}$  is spherized then

$\frac{\partial F}{\partial W}$  can be approximated as in Equation (2.37).

$$\frac{\partial F}{\partial W} = E\{zz^T G''(W^T z)\} + \beta I$$

$$\begin{aligned}
& \approx E\{zz^T\}E\{G''(W^Tz)\} + \beta I \\
& = E\{G''(W^Tz)\} + \beta I = [E\{G''(W^Tz)\} + \beta]I
\end{aligned} \tag{2.37}$$

Hence the gradient becomes a diagonal matrix and can easily be inverted. Thus the algorithm becomes as in Equation (2.38).

$$W \leftarrow W - \frac{E\{zG'(W^Tz)\} + \beta W}{[E\{G''(W^Tz)\} + \beta]I} \tag{2.38}$$

Multiplying both sides of Equation (2.38) by  $-[E\{G''(W^Tz)\} + \beta]I$  and simplifying the resulting expression can be written as Equation (2.39),

$$-W[E\{G''(W^Tz)\} + \beta]I \leftarrow E\{zG'(W^Tz)\} - EG''(W^Tz)W \tag{2.39}$$

Left hand side of Equation (2.39) is nothing but a new variable to which right hand side value will be assigned. Hence the Fast ICA algorithm based on negentropy will become as in Equation (2.40).

$$W_{new} \leftarrow E\{zG'(W^Tz)\} - E\{G''(W^Tz)\}W_{old} \tag{2.40}$$

A brief summary of Fast ICA algorithm based on negentropy for finding one maximally non-Gaussian direction, i.e., estimating one independent component is as follows: The expectations are estimated in practice as an average over the available data set.

1. Center the data to make its mean zero.
2. Whiten the data to get  $\mathbf{z}$ .
3. Choose an initial weight vector  $\mathbf{w}$  of unit norm.

4. Choose a non-quadratic function  $G$  that does not grow too fast (to obtain a better estimation of negentropy). Some of the example functions are given in Equation (2.20).
5. Calculate the new value of  $\mathbf{w}$  using formula given by Equation (2.40).
6. Divide the new value of weight vector by its norm to constrain it to a unit sphere.
7. If not converged, i.e., if direction of the new  $\mathbf{w}$  is not close enough to the old  $\mathbf{w}$  (their absolute value of dot product is not close enough to unity), go back to step 5.

### 2.5.1 Estimating Several Components at the Same Time

The main task of any ICA algorithm is to yield an estimate of the input independent components, say  $\mathbf{y}$ , or in other words to estimate the inverse mixing matrix  $\tilde{\mathbf{A}}^{-1}$  (or  $\mathbf{A}^{-1}$ ). Let  $\mathbf{W}$  be the estimate of inverse of  $\tilde{\mathbf{A}}$  and since  $\tilde{\mathbf{A}}$  is orthogonal,  $\mathbf{W}$  will also be orthogonal with its column vectors  $\mathbf{w}_i$  that are orthogonal to each other. This can be proved in Equation (2.41) by observing that independent components must be uncorrelated, hence for two independent components  $y_i$  and  $y_j$ :

$$E\{y_i y_j\} = E\{(\mathbf{W}_i^T \mathbf{z})(\mathbf{W}_j^T \mathbf{z})\} = \mathbf{W}_i^T E\{zz^T\} \mathbf{W}_j = \mathbf{W}_i^T \mathbf{W}_j = 0 \quad (2.41)$$

Therefore for whitened data, uncorrelatedness is equivalent to orthogonality.

So far the Fast ICA algorithm is given for estimating only one component at a time (also called one-unit algorithm). One way to estimate  $n$  different independent components is to run the one-unit algorithm at least  $n$  times with each run yielding vectors  $\mathbf{w}_1, \dots, \mathbf{w}_n$  respectively. To prevent

different vectors converging at the same point and thus yielding the same value, one must orthogonalize the vectors  $\mathbf{w}_1, \dots, \mathbf{w}_n$  after every iteration. A more efficient method is to orthogonalize all the columns of  $\mathbf{W}$  simultaneously using symmetric orthogonalization.

To carry out symmetrical orthogonalization, first  $(\mathbf{W}^T \mathbf{W})^{-\frac{1}{2}}$  is calculated using eigenvalues decomposition of the symmetric matrix  $(\mathbf{W}^T \mathbf{W})$  as in Equation (2.42).

$$\mathbf{W}^T \mathbf{W} = \mathbf{E} \mathbf{D} \mathbf{E}^{-1} = \mathbf{E} \mathbf{D} \mathbf{E}^T \quad (2.42)$$

where  $\mathbf{E}$  is the matrix whose columns are eigen vectors of  $\mathbf{W}^T \mathbf{W}$  and  $\mathbf{D}$  is a diagonal matrix with the eigen values corresponding to the column eigen vectors of  $\mathbf{E}$  as its main diagonal. Since  $\mathbf{W}^T \mathbf{W}$  is symmetric, its eigen matrix  $\mathbf{E}$  is orthogonal, i.e.,  $\mathbf{E}^{-1} = \mathbf{E}^T (\mathbf{W}^T \mathbf{W})^{-\frac{1}{2}}$  will then be equal to the Equation (2.43).

$$(\mathbf{W}^T \mathbf{W})^{-\frac{1}{2}} = \mathbf{E} \mathbf{D}^{-\frac{1}{2}} \mathbf{E}^T \quad (2.43)$$

Finally the symmetric orthogonalization of  $\mathbf{W}$  is given the Equation (2.44).

$$\mathbf{W} \leftarrow (\mathbf{W}^T \mathbf{W})^{-\frac{1}{2}} \mathbf{W} = \mathbf{E} \mathbf{D}^{-\frac{1}{2}} \mathbf{E}^T \mathbf{W} \quad (2.44)$$

In this research the Fast ICA algorithm with symmetric orthogonalization of unmixing vectors is used for the experiments can be summarized as follows:

1. Center the data to make its mean zero.
2. Whiten the data to get  $\mathbf{z}$ .

3. Choose  $m$ , the number of independent components to estimate.
4. Choose initial values of vectors  $\mathbf{w}_i$ ,  $i = 1, 2, \dots, m$  randomly, each of unit norm.
5. For every  $\mathbf{w}_i$ ,  $i = 1, 2, \dots, m$ , run one-unit algorithm as given by Equation (2.40) in parallel.
6. Construct a  $\mathbf{W}$  matrix with  $\mathbf{w}_i$  as its columns.
7. Do a symmetric orthogonalization of  $\mathbf{W}$  as given in Equation (2.44)
8. If not converged go back to step 5

## 2.6 JADE ALGORITHM

JADE is developed by Cardoso and Souloumiac (Bell and Sejnowski 1995) and is based on the higher-order cumulant tensors, more specifically the fourth order cumulant tensor of the input data vector. Tensors are considered to be generalization of matrices or linear operators. Cumulants tensors are the generalization of covariance matrix if the data has zero mean. This chapter begins with an explanation of the high-order cumulants followed by a discussion of cumulant tensors. The role of cumulant tensors in ICA is explained later and finally algorithm based on eigen matrices of the fourth-order Cumulants tensors, JADE, is explained.

### 2.6.1 Cumulants

Let  $x$  be a real-valued, zero-mean, continuous scalar random variable with probability density function  $P_x(x)$ . The first characteristic function of  $x$  is defined as the continuous Fourier transform of the pdf  $P_x(x)$  is given in Equation (2.45),

$$\phi(\omega) = E\{e^{j\omega x}\} = \int_{-\infty}^{\infty} e^{j\omega x} P_x(x) dx = \sum_{k=0}^{\infty} E\{x^k\} \frac{(j\omega)^k}{k!} \quad (2.45)$$

where the last term in the above Equation is Taylor series expansion of the characteristic function. It can be seen from Equation (2.45) that the coefficients of the expansion are moments  $E\{x_k\}$  of  $x$ . Hence the first characteristic function is also called moment generating function.

The second characteristic function is the natural logarithm of the first characteristic function as in Equation (2.46).

$$\phi(\omega) = \ln(\varphi(\omega)) = \ln(E\{e^{j\omega}\}) = \sum_{k=0}^{\infty} (-j)^k \frac{d^k \phi(\omega)}{d\omega^k} |_{\omega=0} \frac{j\omega^k}{k!} \quad (2.46)$$

where the last term again is the Taylor series expansion of the second characteristic function of  $x$ . The  $k$ -th Cumulant is now defined as the coefficients of the Taylor series expansion of the second characteristic Equation of  $x$  in Equation (2.47), i.e.

$$K_k = (-j)^k \frac{d^k \phi(\omega)}{d\omega^k} |_{\omega=0} \quad (2.47)$$

The first three cumulants for a zero-mean random variable  $x$  are equal to its respective moments while the fourth cumulant is kurtosis.

Similarly if  $\mathbf{x}$  is a multivariate random vector, the first and second order characteristic functions will still be the similar Equations as given by Equation (2.45) and Equation (2.46) with the exception that now instead of scalar quantities, vector quantities will be used for the transformed variable space  $\omega$  and the probability density function of vector  $\mathbf{x}$  will be given by  $P_x(x)$ . Now the cumulants will be called cross-cumulants in analogy to cross-covariance. It can be shown that the second, third and fourth order

cross-cumulants for a zero-mean random vector  $\mathbf{x}$  are given by the Equation (2.48)

$$\text{Cum} (x_i, x_j) = E\{x_i x_j\}$$

$$\text{Cum} (x_i, x_j, x_k) = E\{x_i x_j x_k\}$$

$$\text{Cum} (x_i, x_j, x_k, x_l) = E\{x_i x_j x_k x_l\} - E\{x_i x_j\} E\{x_k x_l\}$$

$$- E\{x_i x_k\} E\{x_j x_l\} - E\{x_i x_l\} E\{x_j x_k\} \quad (2.48)$$

Hence, covariance matrix is the second-order cross-Cumulant. similarly the third-order cross cumulant is the third moment but the fourth-order cumulant differs from the fourth moment. In general the higher-order moments correspond to correlation used in the second-order statistics and cumulants are the higher-order counterparts of covariances. Cumulants can always be expressed as the sum of products of moments, thus they contain the same statistical information as moments.

Earlier, EVD was used to whiten the data. This means that the original data was multiplied by a matrix  $\mathbf{V}$  so that the resultant data vector's second- order correlation (i.e. cross-correlation or cross-covariance) was zero. This results in an identity covariance matrix. Similarly one can use the fourth-order Cumulant tensor to make the fourth order Cumulant zero or at least as small as possible. This type of (approximate) higher order decorrelation gives the current class of method for ICA estimation.

## 2.6.2 Cumulant Tensors

As shown in Equation (2.48) that fourth-order Cumulant  $\text{Cum}(x_i, x_j, x_k, x_l)$  where each index goes from 1 to  $n$ , is a four dimensional matrix. All the fourth-order cumulants of linear combination of  $x_i$  can be obtained as linear combinations of the cumulants of  $x_i$ . The fourth-order Cumulant, i.e.,

kurtosis of a linear combination (e.g., output of a neuron) is given by the Equation (2.49).

$$\begin{aligned} \text{Kurt } (\sum_i w_i x_i) &= \text{cum } (\sum_i w_i x_i, \sum_j w_j x_j, \sum_k w_k x_k, \sum_l w_l x_l) \\ &= \sum_i \sum_j \sum_k \sum_l w_i w_j w_k w_l \text{cum } (x_i, x_j, x_k, x_l) \end{aligned} \quad (2.49)$$

Hence fourth order Cumulant contains all the fourth-order information of the data while covariance contains all the second –order information of the data.

Cumulant tensor is a linear operator defined by the fourth-order Cumulant  $\text{Cum } (x_i, x_j, x_k, x_l)$ . The transformation using Cumulant tensor is in the space of  $n \times n$  matrices. The space of such matrices is a linear space of dimension  $n \times n$ . The basic property of tensor is that its components transform under a rotation of coordinate axes so as to keep its geometrical or physical meaning invariant. The  $i,j$ -th element of the matrix given by the transformation, say  $t_{ij}$ , is defined by the Equation (2.50).

$$T = F(M) \leftrightarrow t_{ij} = \sum_{kl} m_{kl} \text{cum } (x_i, x_j, x_k, x_l) \quad 1 \leq i, j, k, l \leq n \quad (2.50)$$

where  $m_{kl}$  are the elements of the matrix  $\mathbf{M}$  that has been transformed. The Cumulant matrix  $\mathbf{F}(\mathbf{M})$  may be seen as linear combination of parallel Cumulant slices with the entries of  $\mathbf{M}$  as coefficients.

### 2.6.3 Eigenmatrices of Cumulant Tensors

For a  $d$ -dimensional random vector  $x$  with fourth-order cumulants, there exists  $d^2$  real numbers  $\lambda_1, \lambda_2, \dots, \lambda_{d^2}$  called eigenvalues and  $d^2$  matrices  $M_1, M_2, \dots, M_{d^2}$  called eigen matrices such that  $F(M_r)$  is given by the Equation (2.51).

$$F(M_r) = \lambda_r M_r \quad (2.51)$$

$$r = 1, 2, \dots, d^2.$$

To check the structure of eigenvalues and eigen matrices of a standard ICA model, consider whitened data vector  $z$  given in Equation (2.52)

$$z = Vx = VAs = W^T s \quad (2.52)$$

where  $W^T$  is whitened mixing matrix. The Cumulant tensor of  $z$  has special structure that can be seen in EVD of  $z$ . Every matrix of the form as in Equation (2.53).

$$M = W_m W_m^T \quad (2.53)$$

for  $m = 1, 2, \dots, n$  is an eigen matrix. The vector  $W_m$  is one of the rows of  $W^T$ . To check the validation of (2.53), substitute (2.53) in (2.50) which gives the Equation (2.54).

$$\begin{aligned} F(W_m W_m^T) &= \sum_{kl} W_{mk} W_{ml} \text{cum} (Z_i, Z_j, Z_k, Z_l) \\ &= \sum_{kl} W_{mk} W_{ml} \text{cum} (\sum_a W_{ai} S_a, \sum_b W_{bj} S_b, \sum_c W_{ck} S_c, \sum_d W_{dl} S_d) \\ &= \sum_{klabcd} W_{mk} W_{ml} W_{ai} W_{bj} W_{ck} W_{dl} \text{cum} (S_a, S_b, S_c, S_d) \end{aligned} \quad (2.54)$$

Since  $s_i$ 's are independent, hence only those cumulants where  $a = b = c = d$  are non-zero, then Equation (2.54) becomes the Equation (2.55).

$$\begin{aligned} F(W_m W_m^T) &= \sum_{kla} W_{mk} W_{ml} W_{ai} W_{aj} W_{ak} W_{al} \text{cum} (S_a, S_a, S_a, S_a) \\ &= \sum_{kla} W_{mk} W_{ml} W_{ai} W_{aj} W_{ak} W_{al} \text{Kurt} (S_a) \end{aligned} \quad (2.55)$$

Since rows of  $\mathbf{W}$  are orthogonal and of unit length then,  $\sum_k W_{mk} W_{ak} = \delta_{ma}$  i.e., it will be unity only if  $m = a$  and zero otherwise.

Likewise for the index of  $l$ ,  $\sum_l W_{mk} W_{al} = \delta_{ml}$ . Hence Equation (2.55) can be evaluated first by taking sum over  $k$  and then over  $l$  which gives the Equation (2.56).

$$\begin{aligned} F(W_m W_m^T) &= \sum_a w_{ai} w_{aj} \delta_{ma} \delta_{ma} kurt(s_a) \\ &= w_{mi} w_{mj} kurt(s_m) \end{aligned} \quad (2.56)$$

Hence the matrices of the form of Equation (2.53) give the form of Equation (2.51) and it proves that they are the eigen matrices of the tensor and the eigen matrices of the Cumulant tensor leading to the independent components. The spectrum of  $\mathbf{F}(\mathbf{M})$  is made of  $n(n-1)$  zero eigenvalues and  $n$  eigenvalues equal to the kurtosis of the independent components. These  $n$  eigenvalues and corresponding eigen matrices are called significant eigen pairs of  $\mathbf{F}(\mathbf{M})$ .

## 2.7 IMPROVED JADE ALGORITHM (IJADE)

The JADE algorithm uses significant eigen pairs of the cumulant tensor  $\mathbf{F}(\mathbf{M})$  to find out the estimated values of independent components. In this IJADE algorithm, the tensor eigen value decomposition is considered as more of a preprocessing step. Eigen value decomposition can also be viewed as diagonalization. The idea is to diagonalize  $\mathbf{F}(\mathbf{M})$  for any  $\mathbf{M}$  using the matrix  $\mathbf{W}$ . In other words,  $\mathbf{WF}(\mathbf{M})\mathbf{W}^T$  is diagonal. This is because the matrix  $\mathbf{F}$  consists of a linear combination of terms of the form  $\mathbf{w}_i \mathbf{w}_i^T$  assuming that the ICA model holds. Hence the goal is to take a set of significant eigen matrices,  $\mathbf{M}_i$ , and try to make the matrices  $\mathbf{WF}(\mathbf{M})\mathbf{W}^T$  as diagonal as possible.

They might not be made exactly diagonal since the model doesn't hold exactly because of some sampling errors.

Let  $\mathbf{Q} = \mathbf{WF}(\mathbf{M})\mathbf{W}^T$ , then the diagonality of  $\mathbf{Q}$  can be measured, for example, as the sum of the squares of the off-diagonal elements  $\sum_{k=1}^n q_{kj}^2$ .

Since an orthogonal matrix  $\mathbf{W}$  does not change the total sum of the squares of the matrix, minimization of the sum of the off-diagonal elements is equivalent to the maximization of the sum of the square of diagonal elements of the Equation (2.57).

$$J_{JADE}(W) = \sum_i \left\| \text{diag}(\mathbf{WF}(\mathbf{M}_i)\mathbf{W}^T) \right\|^T \quad (2.57)$$

Maximization of  $J_{JADE}$  is the one method of joint approximate diagonalization of the  $\mathbf{F}(\mathbf{M}_i)$ .

Two of the main steps in the JADE algorithm are used to find the significant eigen pairs of the cumulant tensor  $\{\lambda_r, \mathbf{M}_r | 1 \leq r \leq n\}$ , and to jointly diagonalize the JADE criterion  $J_{JADE}(\mathbf{W})$ . These two steps that lead to the JADE algorithm are discussed next.

### 2.7.1 Significant Eigen Pairs

The significant eigen pairs of Cumulant tensor can be found using a classic stacking-unstacking device: the relation  $\mathbf{T}=\mathbf{F}(\mathbf{M})$  is put in a vector-matrix form  $\tilde{\mathbf{T}} = \tilde{\mathbf{F}}\tilde{\mathbf{M}}$  by mapping the  $n \times n$  matrices  $\mathbf{T}$  and  $\mathbf{M}$  into  $n^2 \times 1$  vectors  $\tilde{\mathbf{T}}$  and  $\tilde{\mathbf{M}}$  respectively and fourth-order Cumulant matrix with dimensions  $n \times n \times n \times n$  into an  $n^2 \times n^2$  matrix  $\tilde{\mathbf{F}}$ . The simplest mapping can be defined as in Equation (2.58).

$$\tilde{T}_a = n_{ij}, \tilde{M}_a = m_{ij}, \tilde{F}_{ab} = \text{cum}(Z_i, Z_j, Z_k, Z_l) \quad (2.58)$$

where  $1 \leq i, j, k, l \leq n$ ,  $a = i + (j-1)n$ , and  $b = k + (l-1)d$ . Matrix  $\tilde{F}$  can easily be checked to be a hermitian matrix with  $n^2$  real eigenvalues  $\lambda_1, \lambda_2, \dots, \lambda_n$  and corresponding  $n^2$  eigenvectors. The unstacking of these eigenvectors yield  $n^2$  eigen matrices  $\mathbf{M}_1, \mathbf{M}_2, \dots, \mathbf{M}_n$ . Out of these  $n^2$  eigenvalues,  $n(n-1)$  will be zero (or very close to zero), rest will be the significant eigenvalues and the corresponding Eigen matrices will be the significant eigen matrices.

### 2.7.2 Extended Jacobi Method for Joint Diagonalization

The famous Jacobi eigenvalues algorithm can be extended to diagonalize any number of commuting matrices simultaneously through unitary (or orthonormal, if real) matrices. It consists of maximizing the diagonalization criterion, say as given by Equation (2.57), by successive Givens rotations to minimize the off-diagonal elements.

Let  $\mathbf{C} = \{\mathbf{C}_k | k=1, K\}$  be the set of  $K$  normal commuting matrices, each of dimension  $n \times n$ . The off-diagonal terms in  $\mathbf{C}$  can be set to zero by minimizing the composite objective function in Equation (2.59),

$$O = \sum_{k=1}^K \text{off}(U \mathbf{C}_k U^T) \quad (2.59)$$

by a unitary matrix  $\mathbf{U}$ . The extended Jacobi technique for simultaneous diagonalization constructs  $\mathbf{U}$  as a product of plane Givens rotations globally applied to all of the matrices in  $\mathbf{C}$ . Thus the final value of  $\mathbf{U}$  can be written as in Equation (2.60).

$$\mathbf{U} = \mathbf{R}_1 \mathbf{R}_2 \dots \mathbf{R}_q \quad (2.60)$$

where  $\mathbf{R}_q$  is the rotation matrix corresponding to the  $q$ -th sweep. A sweep is defined as a one complete iteration consisting of rotation matrices corresponding to each off-diagonal element to be eliminated. Hence each sweep can be written as Equation (2.61).

$$\mathbf{R}_q = \mathbf{R}_{12}^q \mathbf{R}_{13}^q \dots \mathbf{R}_{n(n-1)}^q \quad (2.61)$$

The rotation matrices  $\mathbf{R}_{ij}^q$  can be given by  $n \times n$  identity matrices except for the Equation (2.62) elements,

$$\begin{bmatrix} r_{ii} & r_{ij} \\ r_{ji} & r_{jj} \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \quad (2.62)$$

where  $\theta$  is the angle that the input symmetric matrices are rotated to make the  $(i,j)$ -th element of all the matrices as close to zero as possible ( $i \neq j$ ).

To find the rotation angle  $\theta$ , first a matrix  $\mathbf{h}(\mathbf{C})$  is defined as in the Equation (2.63).

$$h(C) = \begin{bmatrix} C_1(i,i) - C_1(j,j) & \dots & C_1(i,j) + C_1(j,i) \\ C_2(i,i) - C_2(j,j) & \dots & C_2(i,j) + C_2(j,i) \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ C_K(i,i) - C_K(j,j) & \dots & C_K(i,j) + C_K(j,i) \end{bmatrix} \quad (2.63)$$

From  $\mathbf{h}(\mathbf{C})$  a symmetric matrix with dimensions  $2 \times 2$  can be formed as in Equation (2.64),

$$\mathbf{G} = \mathbf{h}(\mathbf{C})^T \mathbf{h}(\mathbf{C}) \quad (2.64)$$

Matrix  $\mathbf{G}$  can be utilized to calculate the Jacobi angles (rotation angles)  $\theta$  in closed form using the following theorem:

Under constraint  $|\cos\theta|^2 + |\sin\theta|^2 = 1$ , the objective function as given in Equation (2.59) is minimized using terms in Equation (2.65).

$$\cos \theta = \sqrt{\frac{x+r}{2r}}, \sin \theta = \frac{y}{\sqrt{2r(x+r)}}, r = \sqrt{x^2 + y^2} \quad (2.65)$$

where  $[x, y]^T$  is any eigenvector associated to the largest eigenvalue of  $\mathbf{G}$ .

Rotation angle  $\theta$  can either calculated from Equation (2.65) or alternatively by using the following relation in Equation (2.66).

$$\theta = \frac{\cos^{-1}(|x|)}{2} \quad (2.66)$$

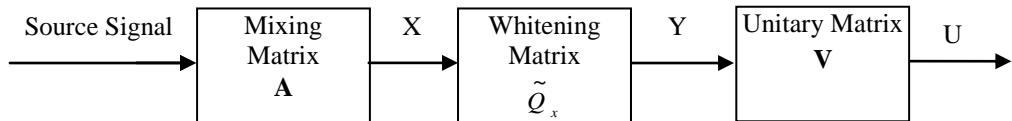
Figure 2.3 gives schematic illustration of JADE process.

A summary of JADE algorithm is given as follows:

1. Preprocess input data matrix  $\mathbf{x}$  to get centered and whitened data matrix  $\mathbf{z}$
2. Calculate fourth-order cross-cumulants of  $\mathbf{z}$
3. Unstack the fourth-order cross-cumulant of  $\mathbf{z}$  into an  $n^2 \times n^2$  matrix  $\tilde{F}_{ab}$  as given in Eqn.(2.58)
4. Find the  $n$  most significant eigen pairs of  $\tilde{F}_{ab}, \{ \lambda_r, \mathbf{M}_r \mid 1 \leq r \leq n \}$
5. Use extended Jacobi method to jointly diagonalize  $\lambda_r, \mathbf{M}_r$ . The orthogonal matrix that approximately jointly diagonalize  $\lambda_r, \mathbf{M}_r$  will be the estimated value of the whitened mixing matrix  $\mathbf{VA}$ , using the Equation (2.67).

$$\text{est}(\mathbf{VA}) = \mathbf{U} = \mathbf{R}_1 \mathbf{R}_2 \dots \mathbf{R}_q \quad (2.67)$$

The criteria of convergence, i.e. the last sweep will be the one with all  $R_{ij}^q$  yielding almost identity matrices. This means that the rotation angle  $\theta = \frac{\cos^{-1}(|x|)}{2}$  is always close to zero for all the rotation matrices in the last sweep thus not changing the unitary matrix in Equation (2.67)



**Figure 2.3 Diagrams to Illustrate the Operation of JADE**

The operation of JADE is outlined in the following:

1. The covariance matrix ( $\hat{R}_x$ ) of the mixtures is obtained. This operation is based on the assumption that the signal source (S) have been normalized to the unity variance so as to ensure their amplitude information is contained in the matrix A. this enables ( $\hat{R}_x$ ) to expressed as  $(\hat{R}_x) = AA^H$ .where  $A^H$  is Hermitian matrix of A. The whitening matrix  $\hat{Q}_x$  is computed by considering the whitening condition  $I = \hat{Q}_x \hat{R}_x \hat{Q}_x^H$ .Replacing ( $\hat{R}_x$ ) gives  $I = \hat{Q}_x AA^H \hat{Q}_x^H$ ,where I is the identity matrix. This implies that  $\hat{Q}_x A$  is a unitary matrix (V) and therefore A can be factorized as  $A = \hat{Q}_x^H V$ .
2. The mixtures are then whitened according to  $\hat{z} = \hat{Q}_x x$ .The whitened matrix ( $\hat{z}$ ) obeys the model  $\hat{z} = \hat{Q}_x AS$ .Substituting for A gives  $\hat{z} = \hat{Q}_x \hat{Q}_x^H VS = VS$ .In order to determine V, the fourth order cumulants of whitened mixtures are computed (Jutten and Karhunen 2004). Their n most significant

eigenvalues ( $\lambda$ ) and their corresponding eigen matrices ( $M_i$ ) are determined. An estimate of the unitary matrix ( $V$ ) is obtained by maximizing the criteria  $N = \lambda_i M_i$  by means of joint diagonalization. If  $N$  cannot be diagonalized, the maximization of the criteria defines a ‘joint approximate diagonalization’.

3. An estimate of unmixing matrix ( $\hat{W}$ ) can be obtained by

$$\hat{W} = \hat{Q}_x V .$$

## **2.8 INFOMAX AND EXTENDED INFOMAX ALGORITHM**

The Infomax algorithm is proposed by Bell and Sejnowski (Lee and Sejnowski 1997). The Infomax method uses a gradient-based algorithm which leads to low complexity in terms of implementation.

The basic idea of the Infomax principle is to match the slope of the nonlinear transfer function of the elementary processing unit (e.g., neuron) in a network with the input Probability Density Function (PDF). The blind source Infomax nonlinear information maximization algorithm performs on-line stochastic gradient ascent in the Mutual Information (MI) between outputs and inputs of a neural-like network. Maximizing the information transfer in a nonlinear Neural Network (NN) minimizes the MI among the outputs when optimization is done over both the synaptic weights and the nonlinear transfer function. By minimizing the MI between its outputs, the network factorizes the input into independent components (James and Hesse 2005)

This chapter begins with an explanation of mutual information estimation and its role in component analysis followed by a comprehensive discussion of Infomax and Extended Infomax methods and finally some of the pros and cons of these methods are discussed.

### 2.8.1 Mutual Information

Mutual information is a natural measure of dependency between random variables i.e. it is a measure of the information that a member of a set of random variables has on the other random variable in the set. If  $\mathbf{y}$  is a  $n$ -dimensional random variable and  $p_y(\eta)$  its probability density function, then vector  $\mathbf{y}$  has mutually independent components, if and only if  $p_y(\eta)$  can be written as in Equation (2.68).

$$p_y(\eta) = p_{y_1}(\eta_1) p_{y_2}(\eta_2) \dots p_{y_n}(\eta_n) \quad (2.68)$$

A natural way of checking whether  $\mathbf{y}$  has ICs is to measure a distance between both sides of the above Equation (2.68) is given as Equation (2.69).

$$I(p_y) = \delta(p_y, \prod p_{y_i}) \quad (2.69)$$

the average mutual information of  $\mathbf{y}$  as given by the Equation (2.70)

$$I(p_y) = \int p_y(\eta) \log\left(\frac{p_y(\eta)}{\prod p_{y_i}(\eta)}\right) d\eta \quad (2.70)$$

The average mutual information vanishes if and only if the variables are mutually independent and are otherwise strictly positive.

### 2.8.2 Mutual Information – ICA Criterion

ICA essentially consists of finding a transformation of the observation vector  $\mathbf{X}$  into a vector  $\mathbf{Y}$  whose components are mutually independent. This can be achieved in several ways, but a natural one is to choose a measure of the mutual dependence of the components  $Y_i$ , and then to optimize the analysis system  $\mathbf{W}$ , so that it minimizes this dependence

measure. There are several sensible measures of mutual dependence, but one that is generally considered as being among the best is Shannon's MI, defined as in Equation (2.71).

$$I(Y) = \sum H(Y_i) - H(Y) \quad (2.71)$$

where  $H$  denotes Shannon's entropy, for discrete variables, or Shannon's differential entropy is given by the Equation (2.72).

$$H(X) = - \int p(x) \log p(x) dx \quad (2.72)$$

for continuous variables,  $p(x)$  being the probability density of the random variable  $X$ . The differential entropy of a multidimensional random variable, such as  $H(Y)$ , is defined in a similar way, with the single integral replaced with a multiple integral extending over the whole domain of  $Y$ .

The mutual information  $I(Y)$  measures the amount of information that is shared by the components of  $Y$ . It is always non-negative, and is zero only if the components of  $Y$  are mutually independent, i.e., if the Equation (2.73) is true.

$$P(Y) = \prod_i P(Y_i) \quad (2.73)$$

$I(Y)$  is Equal to the Kullback-Leibler divergence between  $\prod_i p(Y_i)$  (the joint density that the components  $Y_i$  would have if they were independent but had the same marginal distributions) and the actual joint density  $p(Y)$ . For this reason,  $I(Y)$  is generally considered one of the best measures of dependence of the components of  $Y$ , because it is based on Shannon's concepts of entropy and mutual information, which probably are the best concepts of such quantities in most situations. It has also been used by several

authors as their choice of dependence measure within the nonlinear ICA context

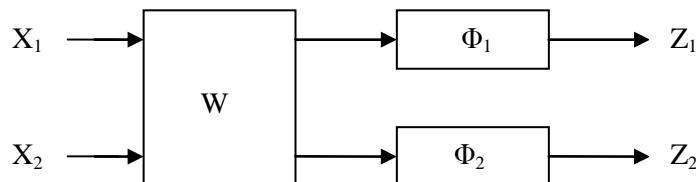
The mutual information has another important property that will be useful in this context. Assume that one applying transformations  $Z_i = \phi_i(Y_i)$ , resulting in new random variables  $Z_i$ , and these transformations are all continuous and monotonic (and thus also invertible). Then, it can be easily shown that  $I(Z) = I(Y)$ . This property has quite a pleasant intuitive meaning: Inspite of not mixing the components  $Y_i$  and have made only invertible transformations on them, the information that they share doesn't change.

## 2.9 INFOMAX METHOD

The Infomax method has been proposed for performing linear ICA based on a principle of maximum information preservation. However, it can also be seen as a maximum likelihood method, or as a method based on the minimization of mutual information. Infomax uses a network whose structure is depicted in Figure 2.4 (the figure shows the case of two components; extension to a larger number of components is straightforward).  $W$  is a linear block, yielding the Equation (2.74).

$$Y = WX \quad (2.74)$$

This block performs a product by a square matrix (Designate both the block and the matrix by the same letter since this will cause no confusion). After optimization, the components of  $Y$  are expected to be as independent from one another as possible. Blocks  $\phi_i$ , which are being used only during the optimization phase, are auxiliary. Each of them implements a nonlinear function (Shall also designate by  $\phi_i$ ). These functions must be increasing, with values in  $[0; 1]$ . The optimization of  $W$  is done by maximizing the output entropy,  $H(Z)$ .



**Figure 2.4 Structure of the Infomax ICA System**

Since each  $\mathbf{z}_i$  is related to the corresponding  $y_i$  by an invertible transformation, that is  $I(Y) = I(Z)$ . Assume that choosing for each nonlinearity  $y_i$ , the Cumulative Probability Function (CPF) of the corresponding component  $y_i$ . Then  $z_i$  will have a uniform distribution in  $[0; 1]$  and  $H(z_i) = 0$ . Consequently, from the relation  $I(Y) = I(Z)$  to get the Equation (2.75)

$$\begin{aligned}
 I(Y) &= I(Z) \\
 &= \sum H(z_i) - H(Z) = -H(Z)
 \end{aligned} \tag{2.75}$$

Maximization of the output entropy  $H(\mathbf{z})$  will therefore be equivalent to the minimization of  $I(y)$ , which is the mutual information of the estimated components. Infomax can therefore be viewed as minimizing this mutual information, with an apriority choice of the estimated distributions of the components that can be performed through the choice of the  $y$  nonlinearities (James and Hesse 2005). These should approximate the CPFs of the actual components as closely as possible. However, as mentioned above, linear ICA is a rather constrained problem, and therefore Infomax usually performs well even if the output nonlinearities are only crude approximations to these cumulative functions. For example, it is known that logistic sigmoid can be used as nonlinearities for most unskewed, supergaussian distributions (Bell and Sejnowski 1995)

### 2.9.1 Infomax Algorithm

A simple case of a one-input one-output system is considered to derive the ICA algorithm. The general multi-input multi-output system is similarly derived with  $n$ -dimensional matrices of vector-valued random variables in place of the scalar valued functions.

Consider a scalar-valued function  $x$  with a Gaussian pdf  $f_x(x)$  that passes through a transformation function  $y = g(x)$  to produce the output with pdf  $f_y(y)$ . This is analogous to matrix operation in Equation (2.76).

$$Y = WX \quad (2.76)$$

The transformation function  $y$  to be the logistic sigmoid function is given in Equation (2.77).

$$y = g(x) = \frac{1}{1 + e^{-u}} , \quad u = W_x + W_0 \quad (2.77)$$

where  $w$  = slope of  $y$  (also called the weight),  $w_0$  = bias weight to align the high density parts of the input with  $y$ .

As discussed in previous sections, an increase in the joint entropy of the output,  $H(y)$ , results in decrease in its mutual information. The entropy of the output is maximized by aligning the high density parts of pdf of  $x$  with the high sloping parts of the function  $g(x)$  (hence the need for the biasing weight  $w_0$ ). The function  $g(x)$  is monotonically increasing (i.e. has a unique inverse) and thus the pdf of the output  $f_y(y)$  can be written as a function of the pdf of the input  $f_x(x)$  as in Equation (2.78).

$$f_y(y) = \frac{f_x(x)}{\left| \frac{\partial y}{\partial x} \right|} \quad (2.78)$$

The entropy of the output is given by the Equation (2.79).

$$H(y) = -E\{\ln f_y(y)\} = - \int_{-\infty}^{\infty} f_y(y) \ln f_y(y) dy \quad (2.79)$$

Substituting Equation (2.78) into Equation (2.79) gives the Equation (2.80).

$$H(y) = E\left(\ln \left| \frac{\partial y}{\partial x} \right|\right) - E\{\ln f_x(x)\} \quad (2.80)$$

$H(y)$  of Equation (2.80) has to be maximized for statistical independence. Looking at the right hand side, can see that the function  $x$  is fixed and the variable  $y$  can be changed. Or more precisely, the slope,  $w$ , of  $y$ . Hence the partial derivative of  $H(y)$  with respect to  $w$  is taken. The second term in Equation (2.80) does not depend on  $w$  and therefore can be ignored. The change in slope,  $\Delta w$ , necessary for maximum change in entropy is given by the Equation (2.81).

$$\Delta w \propto \frac{\partial H(y)}{\partial w} = \frac{\partial}{\partial w} E\left(\ln \left| \frac{\partial y}{\partial x} \right|\right) \quad (2.81)$$

Derivative can be computed from the Equation (2.81), but the expectation of the natural logarithm term cannot be computed. Hence, Bell made the stochastic gradient approximation:

$E\left(\ln \left| \frac{\partial y}{\partial x} \right|\right) \approx \ln \left| \frac{\partial y}{\partial x} \right|$ , to get rid of the expectation. The Equation (2.81) is simplified to the Equation (2.82) as follows:

$$\Delta w \propto \frac{\partial H(y)}{\partial w} = \frac{\partial}{\partial w} \left( \ln \left| \frac{\partial y}{\partial x} \right| \right) = \left( \frac{\partial y}{\partial x} \right)^{-1} \frac{\partial}{\partial w} \left( \frac{\partial y}{\partial x} \right) \quad (2.82)$$

The above Equation (2.82) is the general form of the weight change rule for any transformation function  $y$ . For the logistic sigmoid function Equation (2.77), the terms in Equation (2.82) are evaluated as in Equation (2.83) and Equation (2.84).

$$\frac{\partial y}{\partial x} = wy(1 - y) \quad (2.83)$$

$$\frac{\partial}{\partial w} \left( \frac{\partial y}{\partial x} \right) = y(1 - y)(1 + wx(1 - 2y)) \quad (2.84)$$

Substituting the above Equation (2.83) and Equation (2.84) into Equation (2.82) gives the weight update rule for  $y$  = logistic sigmoid function as in Equation (2.85).

$$\Delta w \propto w^{-1} + (1 - 2y)x \quad (2.85)$$

Similarly, the bias weight update,  $\Delta w_0$ , can be evaluated with the Equation (2.86).

$$\Delta w_0 \propto 1 - 2y \quad (2.86)$$

Bell and Sejnowski's adaptive learning algorithm (Lee and Sejnowski 1997) blindly separates mixtures,  $X$  of independent sources and  $S$  using information maximization (Infomax) is described by the following steps:

- i) The demixing matrix  $W$  is initialized to an identity matrix.
- ii) The signal sources are estimated by  $X = AS \rightarrow S = WX$  and they are transformed by a nonlinear transfer function. For a sigmoidal transfer function, the resulting signals  $Y$  are expressed as in Equation (2.87).

$$Y = g(S) = \frac{1}{1 + e^{-(S + w_0)}} \quad (2.87)$$

where  $w_0$  is a vector of bias weights which is initialized to a zero vector

- iii) The nonlinearly transformed signals  $Y$  are processed by a learning rule which maximizes their joint entropy that can approximately minimize their mutual information. This is achieved by changing the weight matrix by an amount  $\Delta W$ , where  $\Delta W$  is given in Equation (2.88).

$$\Delta W = [W^T]^{-1} + (1 - 2y)x^T \quad (2.88)$$

The change in the bias weight is expressed by  $\Delta w_0 = 1 - 2y$

- iv) The ICA algorithm is trained by repeating the steps (ii) and (iii). After each iteration, the demixing matrix  $W$  is updated by  $\Delta W$  until the convergence is achieved.

The algorithm stops training when the rate of change falls below a predefined small value, e.g.  $1.0 \times 10^{-6}$ . The rate of change is computed by squaring the difference between corresponding elements of the demixing matrix before and after iteration and then summing the values (Lee 1997).

## 2.10 EXTENDED INFOMAX METHOD

The algorithm of Bell and Sejnowski which uses a sigmoidal activation function is specifically suited to separate signals with super-Gaussian distribution (i.e. positive kurtosis). Lee and Sejnowski (Sarah Hosni and Mahmoud Gadallah 2007) proposed an extension of Infomax ICA that is able to separate with sub and as well as super Gaussian distribution. This preserves the ICA architecture of Infomax algorithm. But it uses a learning rule derived by Girolami and Fyfe. It determines the sign changes (positive to negative and vice versa) required by the algorithm to handle both sub and super Gaussian distributions.

### 2.10.1 Deriving a Learning Rule to Separate Sub- and Supergaussian Sources

The objective of the Extended Infomax algorithm is to provide a simple learning rule with a fixed nonlinearity that can separate sources with a variety of distributions. One way of generalizing the learning rule to sources with either sub- or supergaussian distributions is to approximate the estimated p.d.f. with an Edge worth expansion or Gram-Charlier expansion (Girolami and Fyfe 1997). Girolami in 1997 used a parametric density estimate to derive the same learning rule without any approximations, as shown below. A symmetric strictly subgaussian density can be modeled using a symmetrical form of the Pearson mixture model as shown in Equation (2.89).

$$p(u) = \frac{1}{2}(N(\mu, \sigma^2) + N(-\mu, \sigma^2)) \quad (2.89)$$

where  $N(\mu, \sigma^2)$  is the normal density with mean  $\mu$  and variance  $\sigma^2$ . The density  $p(u)$  for  $\sigma^2 = 1$  with varying  $\mu = [0, \dots, 2]$ . For  $\mu = 0$ ,  $p(u)$  is a Gaussian model whereas for  $\mu_i = 1.5$ , for example, the  $p(u)$  is clearly bimodal. The kurtosis  $k_4$  (normalized fourth-order cumulant) of  $p(u)$  is given by Equation (2.90)

$$k = \frac{c_4}{c_2^2} = \frac{-2\mu^4}{(\mu^2 + \sigma^2)^2} \quad (2.90)$$

where  $c_i$  is the  $i$  th-order cumulant. Depending on the values of  $\mu$  and  $\sigma^2$  the kurtosis lies between -2 and 0. So Equation (2.89) defines a strictly subgaussian symmetric density when  $\mu > 0$ . Defining  $a = \frac{\mu}{\sigma^2}$  and applying Equation (2.89),  $\Delta W_0 = \varphi(u)$  can be written as in Equation (2.91).

$$\varphi(u) = -\frac{\partial p(u)}{\partial u} = \left[ -\frac{\frac{\partial p(u_1)}{\partial u_1}}{p(u_1)}, \dots, -\frac{\frac{\partial p(u_N)}{\partial u_N}}{p(u_N)} \right]^T$$

$$\varphi(u) = -\frac{\partial p(u)}{\partial u} = \frac{u}{\sigma^2} - a \left( \frac{\exp(-au) - \exp(-au)}{\exp(-au) + \exp(-au)} \right) \quad (2.91)$$

Using the definition of the hyperbolic tangent, Equation (2.91) can be written as Equation (2.92).

$$\varphi(u) = \frac{u}{\sigma^2} - \frac{u}{\sigma^2} \tanh\left(\frac{\mu}{\sigma^2}u\right). \quad (2.92)$$

Setting  $\mu = 1$  and  $\sigma^2 = 1$ , Eqn.(2.92) reduces to Equation (2.93).

$$\varphi(u) = u - \tanh(u) \quad (2.93)$$

The learning rule for strictly subgaussian sources is given from Equation (2.85) and Equation (2.86) that can be written as Equation (2.94)

$$\Delta W \propto [I + \tanh(u)u^T - uu^T]W \quad (2.94)$$

In the case of unimodal supergaussian sources, the density model is given in Equation (2.95)

$$p(u) \propto p_G(u) \sec h^2(u) \quad (2.95)$$

where  $p_G(u) = N(0,1)$  is a zero-mean Gaussian density with unit variance. Figure 6.4 shows the density model for  $p(u)$ . The nonlinearity  $\varphi(u)$  can be written as Equation (2.96).

$$\varphi(u) = -\frac{\partial u}{p(u)} - u + \tanh(-u) \quad (2.96)$$

The learning rule for supergaussian sources from Equation (2.85) and Equation (2.86) is given in Equation (2.97) as

$$\Delta W \propto [I - \tanh(-u)u^T - uu^T]W \quad (2.97)$$

The difference between the supergaussian learning rule in Equation (2.98) and the subgaussian learning rule in Equation (2.97) is the sign before the tanh function:

$$\begin{aligned} \Delta W &\propto [I - \tanh(-u)u^T - uu^T]W && : \text{Supergaussian} \\ \Delta W &\propto [I + \tanh(-u)u^T - uu^T]W && : \text{Supergaussian} \end{aligned} \quad (2.98)$$

**Switching Between Nonlinearities** The switching between the sub and supergaussian learning rule is

$$\begin{aligned} \Delta W &\propto [I - K_i \tanh(-u)u^T - uu^T]W . & K_i=1 & : \text{Supergaussian} \\ K_i=-1 & & & : \text{Subgaussian.} \end{aligned}$$

## 2.11 EYEBLINK COMPONENT IDENTIFICATION

After ICA, Independent sources related to eye blinks must be identified from the independent components of EEG. The scalp topography of each component provides evidence of its physiological origin. An eye blink component's scalp map has a strong far-frontal projection. A simple rule for eye blink components identification was developed based on this fact and these components must be removed from the components matrix  $s_j(t)$  in order to reconstruct a clean EEG segment.

Denoting the  $j^{\text{th}}$  column of  $W^{-1}$  by  $W^{-1}_j$  it represents the intensity distribution at each electrode (i.e. the scalp map) of the corresponding component  $s_j(t)$ , where  $j = 1,..,7$ .  $w^{-1}_{ij}$  denote the  $i^{\text{th}}$  element in  $W^{-1}_j$  where  $i = 1,..,7$  as the number of sources is equal to the number of channels. Then the eye blink component identification rule is as given in Equation (2.99).

$$\text{If } \max(W^{-1}_j) = w^{-1}_{7j}$$

$$\text{Then } s'_j(t) = 0$$

$$W^{-1}_j = 0$$

$$\text{Else } s'_j(t) = \hat{s}_j(t), \quad (2.99)$$

$\max(W^{-1}_j) = w^{-1}_{7j}$  means that the ocular activity contributes the most to this component activity regardless of the sign of that activity. By applying this artifact identification method for independent components, the EOG is completely removed from EEG.

### 2.11.1 EEG Reconstruction

After identifying and removing artifactual components, EEG data are reconstructed using the new independent components matrix as given in Equation (2.100).

$$x'(t) = W^{-1} s'(t) \quad (2.100)$$

where  $s'(t)$  is the matrix of the recovered sources  $s(t)$  with rows representing artifactual sources set to zero and  $x'(t)$  is the corrected EEG segment .

## 2.12 RESULTS

The data used here were recorded using six recording channels (electrodes) from four subjects performing the five mental tasks. The recordings of mental tasks were conducted for several 10 seconds trials and

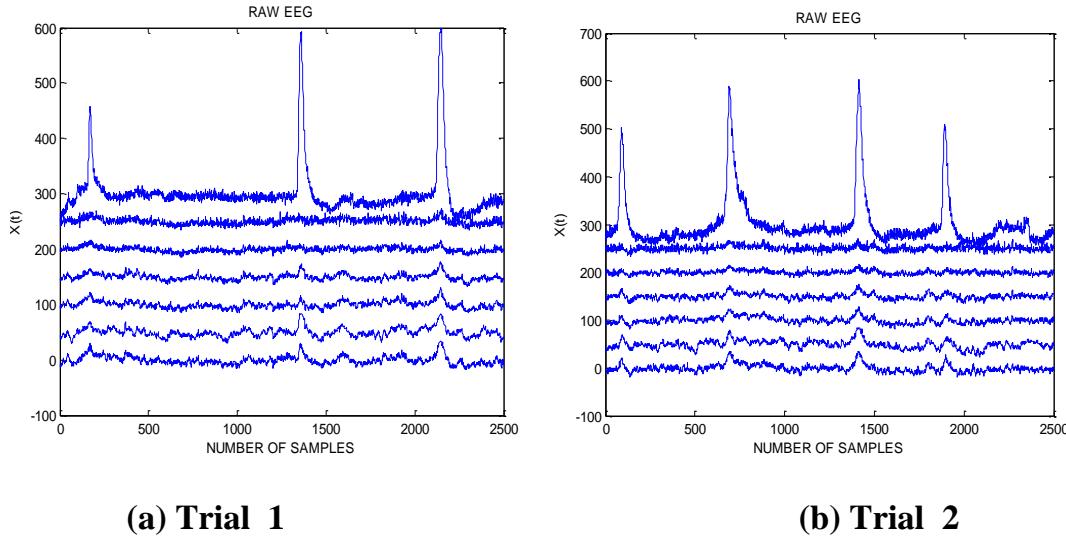
trials were made in two sessions. Each session was conducted on a separate day. The data are available online at ([http://www.cs.colostate.edu/~anderson.](http://www.cs.colostate.edu/~anderson/))

An Electro-Cap named elastic electrode cap was used to record from positions C3, C4, P3, P4, O1, and O2, shown in previous chapter and defined by the 10-20 system of electrode (sensor) placement. Eye blinks were detected by means of a separate channel of data (EOG channel). The mental tasks were:

- (1) ***Baseline task:*** The subjects were asked to relax as much as possible.
- (2) ***Letter task:*** The subjects were instructed to mentally compose a letter to a friend or relative without vocalizing.
- (3) ***Math task:*** The subjects were given nontrivial multiplication problems, such as 49 times 78.
- (4) ***Visualized counting:*** The subjects were asked to imagine a blackboard and to visualize numbers being written on the board sequentially
- (5) ***Geometric object rotation:*** The subjects were asked to visualize a particular three dimensional block figure that is being rotated about an axis. Data was recorded for 10 seconds during each task and each task was repeated five times per session.

Data were recorded for 10 seconds during each task and each task was repeated five times per session. The subjects who attended two sessions were recorded on separate weeks, resulting in a total of ten trials for each task. With a 250 Hz sampling rate, each 10 second trial produces 2,500

samples per channel. Figure 2.5 shows one subject's EEG data obtained from doing math tasks twice.



**Figure 2.5 EEG Recording of a Subject during Mental Multiplication**

### 2.12.1 Results of Fast ICA Algorithm

Parameters:

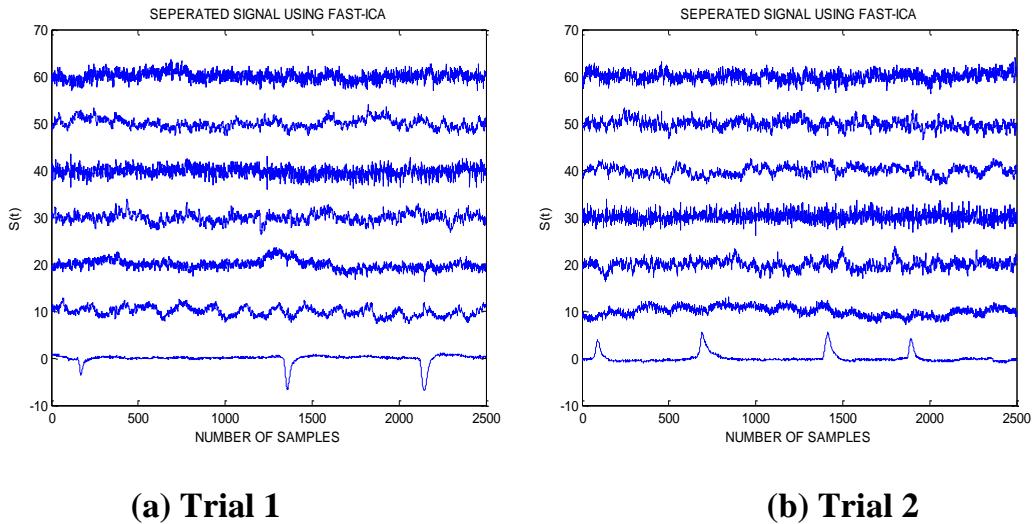
Nonlinearity:  $\log(\cosh(y))$

No. of iterations: 100

Max. weight change: 10e-300

Figure 2.6 shows the independent components obtained using the Fast ICA algorithm from the EEG data mixed with EOG which is shown Figure 2.5. In Figure 2.5, the EOG recording appears in all channels of original EEG data. But from Figure 2.6, it can be found that the EOG artifact is concentrated in component 7(order from top to bottom) and does not appear in other independent components.

Execution time in seconds: Trial 1: 1.59, Trial 2: 1.61

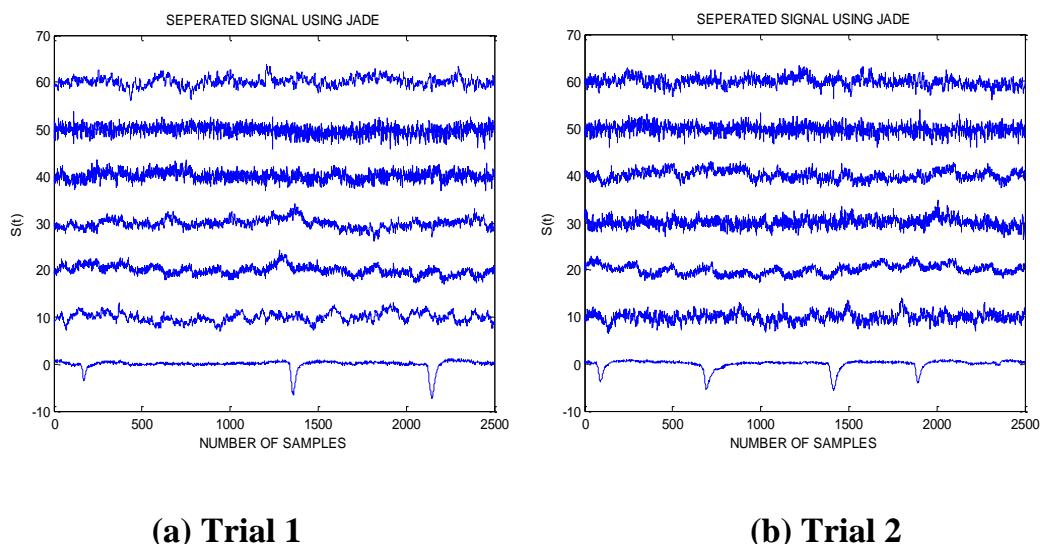


**Figure 2.6 Independent Components Obtained using Fast ICA**

### 2.12.2 Results of IJADE Algorithm

No adjustable parameters.

Figure 2.7 shows the independent components obtained using the IJADE algorithm for the EEG data of Figure 2.5.



**Figure 2.7 Independent Components Obtained using IJADE**

Execution time in seconds: Trial1: 0.99, Trial 2: 1.01

### 2.12.3 Results of Infomax:

Parameters: learning rate= 0.1;

Max. Change in weight = $1e-3$

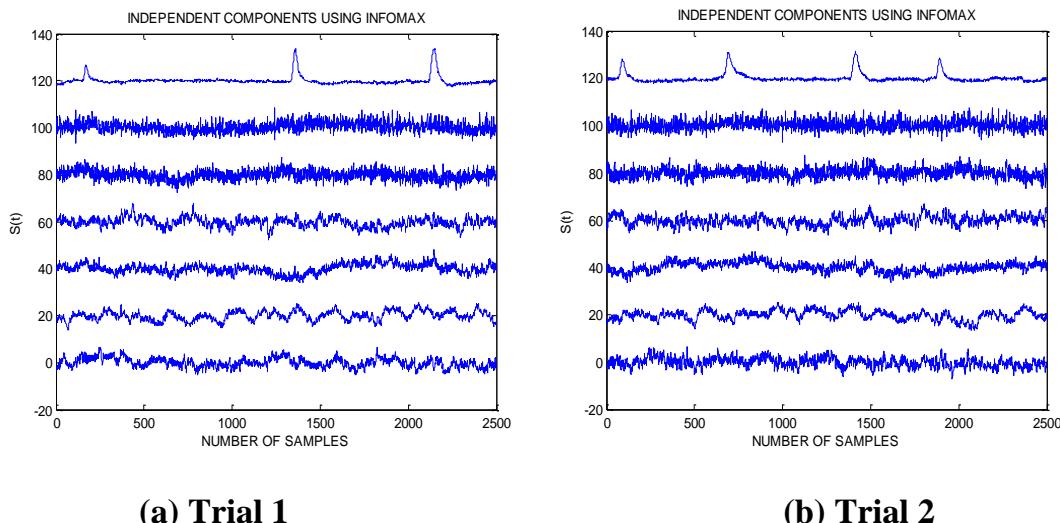
Transformation function =logistic sigmoid = $\frac{1}{1 + e^{-u}}$

Number of iterations: 512

Bias weight =0

Initial weight =Identity Matrix

Figure 2.8 shows the independent components obtained using the Infomax algorithm for the EEG data of Figure 2.5.



**Figure 2.8 Independent Components Obtained using Infomax**

Execution time in seconds: Trial 1: 2.20, Trial 2: 2.52

### 2.12.4 Results of Extended Infomax

Parameters: Min. Weight-change: 1e-3

Number of iterations=512

Signs: -1: subgaussian

1: supergaussian

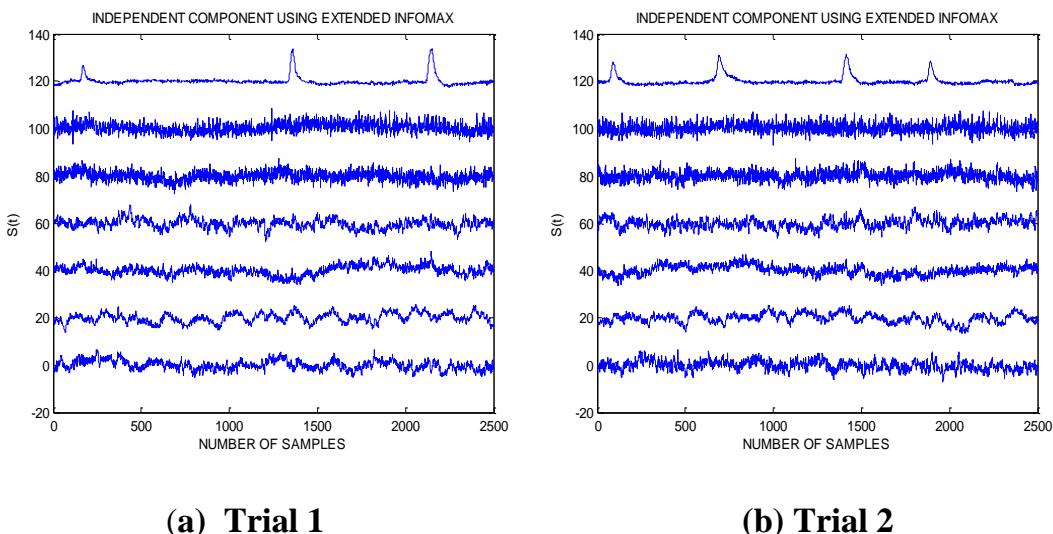
Initial weight= Identity matrix

Nonlinearity =  $\tanh(u)$

Bias =0

Learning rate =0.1

Figure 2.9 shows the independent components obtained using the Extended Infomax algorithm for the EEG data of Figure 2.5.



**Figure 2.9 Independent Components Obtained using Extended Infomax**

Execution time in seconds: Trial 1: 3.07, Trial 2: 3.03.

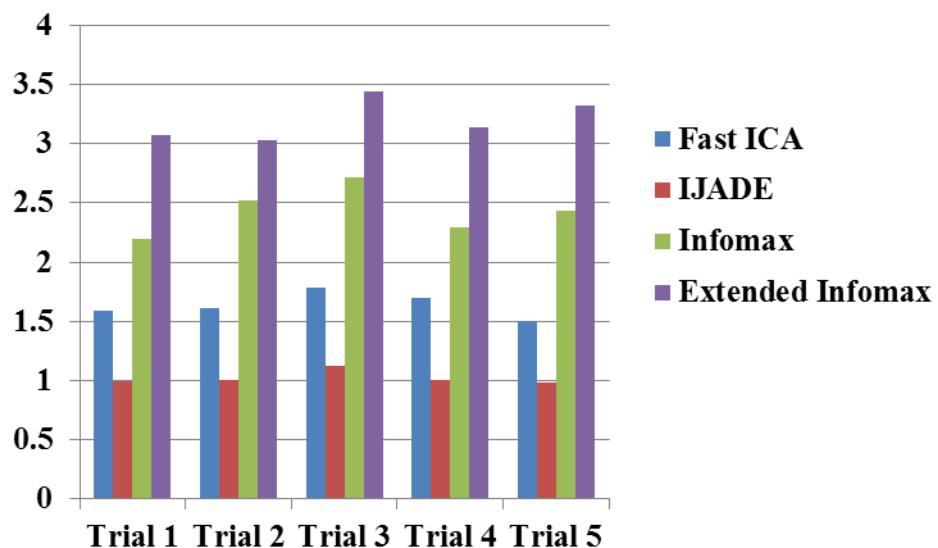
### 2.12.5 Performance Comparisons of Algorithms

- **Computation Time:** The computation time i.e., the time taken by algorithm to separate the EEG signal is measured. For comparison of algorithms mental multiplication task EEG which is measured for five Trial s is used. The computation time in seconds is tabulated in Table 2.1.
- **Entropy:** Entropy of original mental multiplication EEG signals of five Trial s is given in Table 2.2. Entropy of the separated signal by different ICA algorithms is given in Table 2.3.

**Table 2.1 Computation Time for ICA Algorithms in Seconds**

Task (multiplication)	Algorithm	Fast ICA	IJADE	Infomax	Extended Infomax
Trial 1	<b>1.59</b>	<b>0.99</b>	<b>2.20</b>	<b>3.07</b>	
Trial 2	<b>1.61</b>	<b>1.01</b>	<b>2.52</b>	<b>3.03</b>	
Trial 3	<b>1.78</b>	<b>1.12</b>	<b>2.71</b>	<b>3.44</b>	
Trial 4	<b>1.70</b>	<b>1.01</b>	<b>2.29</b>	<b>3.14</b>	
Trial 5	<b>1.50</b>	<b>0.98</b>	<b>2.43</b>	<b>3.32</b>	

Figure 2.10 shows the computation time curve for four ICA Algorithms



**Figure 2.10 Computation Time curve for ICA Algorithms**

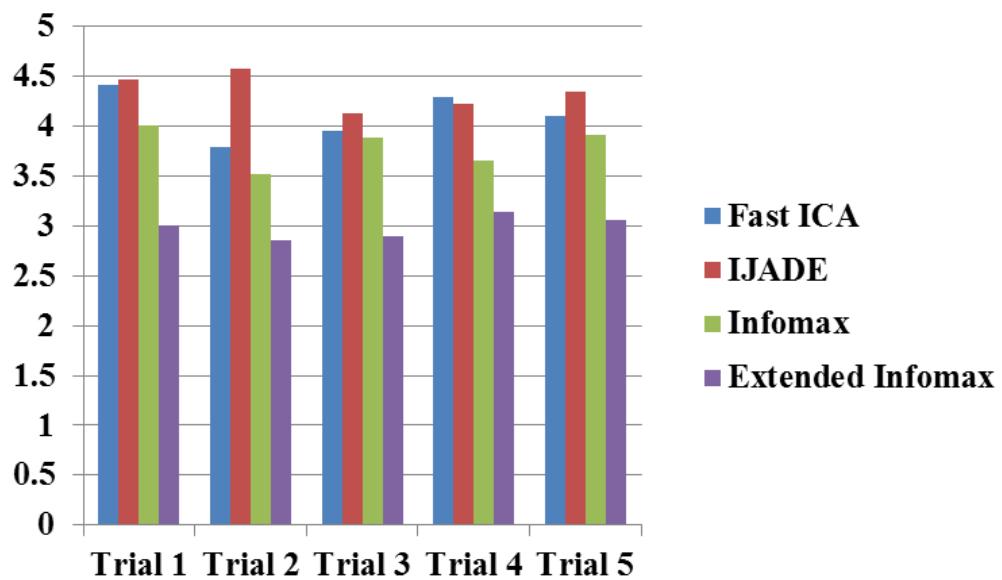
**Table 2.2 Entropy of Original EEG**

Trial 1	Trial 2	Trial 3	Trial 4	Trial 5
<b>1.646</b>	<b>1.664</b>	<b>1.759</b>	<b>1.787</b>	<b>1.705</b>

**Table 2.3 Entropy of EEG after Removal of EOG**

Algorithm Multiplication	Fast ICA	IJADE	Infomax	Extended Infomax
<b>Trial 1</b>	<b>4.409</b>	<b>4.468</b>	<b>4.001</b>	<b>2.996</b>
<b>Trial 2</b>	<b>3.781</b>	<b>4.570</b>	<b>3.518</b>	<b>2.858</b>
<b>Trial 3</b>	<b>3.955</b>	<b>4.124</b>	<b>3.878</b>	<b>2.899</b>
<b>Trial 4</b>	<b>4.293</b>	<b>4.222</b>	<b>3.650</b>	<b>3.139</b>
<b>Trial 5</b>	<b>4.092</b>	<b>4.335</b>	<b>3.91</b>	<b>3.050</b>

Figure 2.11 Shows the Entropy curve for EEG after removal of EOG.



**Figure 2.11 Entropy curve for EEG after removal of EOG**

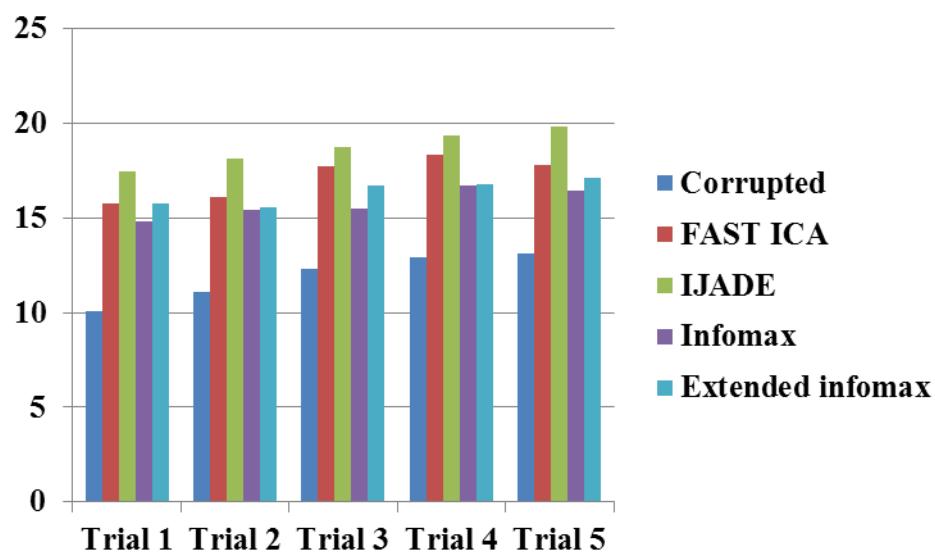
- **SIGNAL-TO-NOISE RATIO**

Signal to Noise ratio is a measure used in science and engineering that compares the level of a desired signal to the level of background noise. It is defined as the ratio of signal power to the noise power.

Signal to Noise ratio of the four ICA methods are tabulated in Table 2.4. Figure 2.12 shows the Signal to Noise ratio curve for four ICA Methods.

**Table 2.4 Signal to Noise Ratio in dB**

Trial s	Corrupted signal SNR in dB	SNR in db Using ICA Methods			
		Fast ICA	IJADE	Infomax	Extended infomax
Trial 1	10.1040	15.7810	17.4210	14.8010	15.7210
Trial 2	11.1020	16.1010	18.1010	15.4210	15.5220
Trial 3	12.3010	17.7020	18.7020	15.4840	16.7220
Trial 4	12.9210	18.3030	19.3030	16.7030	16.7840
Trial 5	13.1020	17.840	19.8040	16.4010	17.1340

**Figure 2.12 Signal to Noise ratio curve for ICA Methods**

The five trials result shows the comparative evaluation of signal to noise ratio. Compare with corrupted signal SNR the independent component t analysis methods SNR value is very high. Among these methods JADE only give the better result in terms of correlation factor and computation time also.

## 2.13 SUMMARY

In this chapter, the basic principle behind the independent component analysis technique is discussed. The contrast functions for different routes to independence are clearly depicted. Different ICA algorithms are briefly illustrated and are critically examined with special reference to their algorithmic properties. The ambiguities present in these algorithms are also presented. EEG signals will maintain the similarity in their patterns when subject is performing the mental task. The traditional methods applied for remove artifacts can only compromise between eliminating artifacts and protecting useful signals so that the result is not very satisfying.

However, ICA method can protect the useful signals as well as obviously weaken even entirely remove the artifacts in multi-channel EEG signals, this characteristic of ICA is the key to get stable EEG patterns which can be used for mental task classification. Two functions which are used for maximization of non-gaussianity by the using Fast ICA have been discussed. Negentropy is proposed as best estimation functions for estimating independent components from mixed data. Detailed description of mathematical derivation of Fast ICA has been examined for estimation of several independent components parallelly. Different non quadratic functions are introduced for finding unmixing matrix by the use of Fast ICA. Improve JADE Algorithm is explained in this chapter. The explanation of high-order cumulants followed by a discussion of cumulant tensors also given in this chapter for IJADE. The role of cumulant tensors in IJADE is explained. The IJADE Algorithm uses significant eigen pairs of the cumulant tensor  $F(M)$  to find out the estimated values of independent components. In this Improved JADE Algorithm, the tensor eigen value decomposition is considered as more of preprocessing step. The eigen value decomposition can also be viewed as diagonalization. IJADE provides better results among with four ICA algorithms in terms of their convergence speed, entropy and signal to Noise ratio.

## CHAPTER 3

### ARTIFACTS REMOVAL FROM EEG USING NEURO FUZZY FILTER

#### 3.1 INTRODUCTION

In recent years, applications in the fields like image processing, pattern recognition, process control, forecasting, credit evaluation, medical diagnosis and so on have used hybrid fuzzy neural filter. Different fuzzy neural network architecture has been proposed for different applications.

Neuro-fuzzy systems are the combination of neural networks and fuzzy logic and so called hybrid. These hybrid systems take the advantages of both the systems and have overcome the disadvantages of individual systems (García et al 2008). Neural networks, used alone have uncertainty but the combination of neuro-fuzzy has good expressiveness and uncertainty is reduced adapting according to the changes in the environments.

- **Human expertise:** Soft computing makes use of human expertise and frame if-then rules in fuzzy logic. The practical problems are solved with the help of the conventional knowledge representation.
- **Model-free learning:** Neural networks and adaptive fuzzy inference system have the ability to construct models using only target system sample data. Deep study about the target

system enables to set up the initial model structure, but it is not compulsory.

- **Intensive computation:** Neuro-fuzzy and soft computing make use of high speed number crunching computation to find rules or regularity in data sets rather than assuming about the background knowledge of the problem.
- **Fault tolerance:** Like any other system, neural networks and fuzzy inference systems also have fault tolerance. If there is any violation in the neural network or if the rules in fuzzy logic are violated it does not cease the system. The system continues to function because of its parallel and redundant architecture, but the performance of the system is reduced, that is the quality deteriorates.
- **Goal driven characteristics:** Neuro-fuzzy mainly concentrates on the goal. The direction taken from the current state to the solution does not matter as long as the function is correctly pointing towards the goal.

## **3.2 EXTRACTING KNOWLEDGE FROM EXISTING METHODS**

### **3.2.1 Armax Modeling Method**

The basic assumption of ARMAX modeling is that EEG signal is a combination of EOG artifacts and the background EEG. ARMAX method provides estimation of EEG and the artifacts that background the EEG. Then the background EEG is estimated via estimation of ARMAX parameters. The background EEG is an uncorrelated white noise with zero mean according to this model. Propagation characteristics is the same for all frequencies of EOG (García et al 2007). These assumptions are not completely true so they must

be relaxed. In order to do so the measured EEG is modeled as an ARMAX process described as in Equation (3.1)

$$y(n) = \sum_{k=1}^p A_k y(n-k) + \sum_{k=0}^q B_k u(n-k) + \sum_{k=1}^r C_k w(n-k) + w(n) \quad (3.1)$$

where  $(p, q, r)$  is the model order. The coefficients of this model are estimated with the help of Recursive Extended Least Squares estimator. Then the background EEG is estimated by using the previous and the present values of  $y$  and  $u$

### 3.2.2 Disadvantages

The main disadvantage of the ARMAX method is that it needs more complicated calculations to eliminate the negative spike that appears at blinking time. This elimination can be handled only in higher orders of ARMAX. Model order estimation is necessary for a good performance of ARMAX but it complicates the procedure. This problem is overcome by adaptive filtering method that does not need any calibration trial or parameter estimation. Also in ARMAX model, it is difficult to interpret the values of the covariate coefficients. The ARMAX model structure includes disturbance dynamics.

## 3.3 ADAPTIVE FILTERING METHOD

An adaptive filter is a very efficient filter in which the transfer function is adjusted according to an optimization algorithm driven by an error signal. It is basically a computational device and its function is to model the relationship between two signals in real time in an iterative manner. They are often realized with the help of a set of program instructions running on an arithmetical processing device such as a microprocessor or DSP chip, or a set

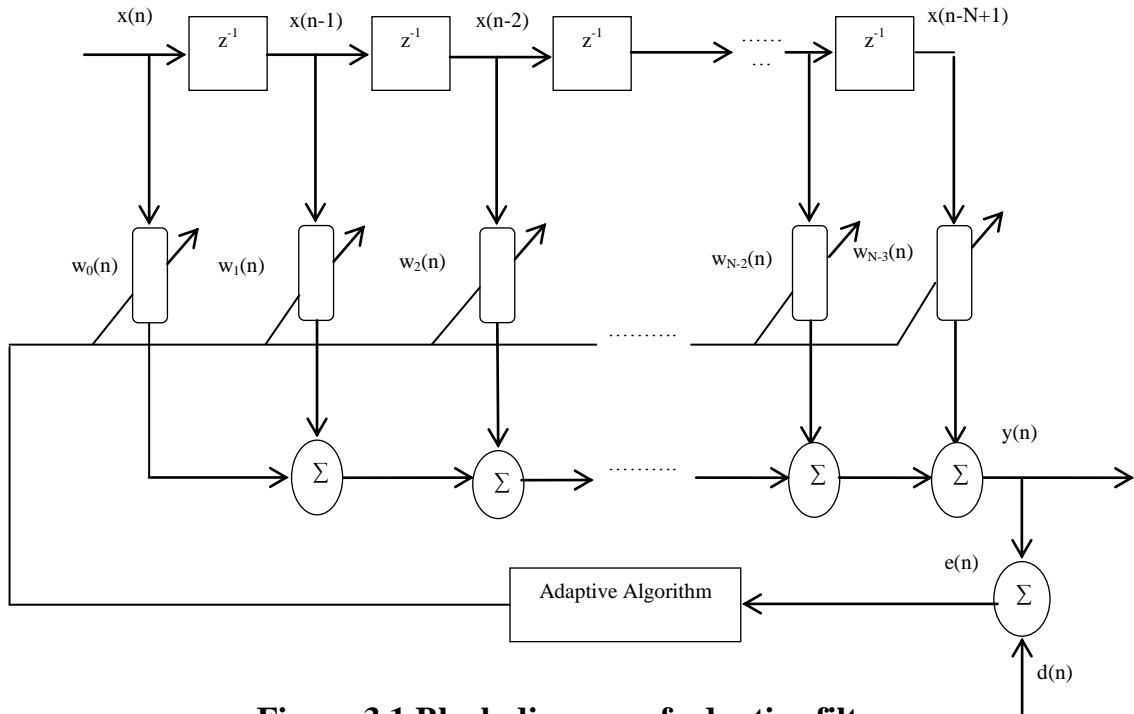
of logic operations implemented in a field-programmable gate array (FPGA) or a semi-custom or custom VLSI integrated circuit.

However, any errors introduced by numerical precision effects in these implementations, the fundamental operation of an adaptive filter, can be characterized independent of the specific physical realization that it takes. Adaptive filters are mostly digital filters due to the complexity of the optimization algorithms. In contrast to this, a non-adaptive filter has a static transfer function. Adaptive filters are used where some parameters of the desired processing operation (for instance, the locations of reflective surfaces in a reverberant space) are not known in advance.

- **An adaptive filter is defined by four aspects:**
  1. The signals that are being processed by the filter
  2. The structure that explains the function how the output signal of the filter is computed from its input signal
  3. The parameters within this structure that can be iteratively changed to alter the filters input-output relationship
  4. The adaptive algorithm that describes how the parameters are adjusted from one time instant to the next.

The adaptive filter uses feedback. The error signal is used as feedback to refine its transfer function to match the changing parameters. The number and type of parameters can be adjusted according to the adaptive filter structure. The adaptive algorithm can take many myriad forms but often used as optimization procedure that minimizes an error criterion that is useful for the task at hand. The general adaptive filtering problem and the mathematical notation for representing the form and operation of the adaptive filter in discuss this section. A simple derivation for the Least Mean Square

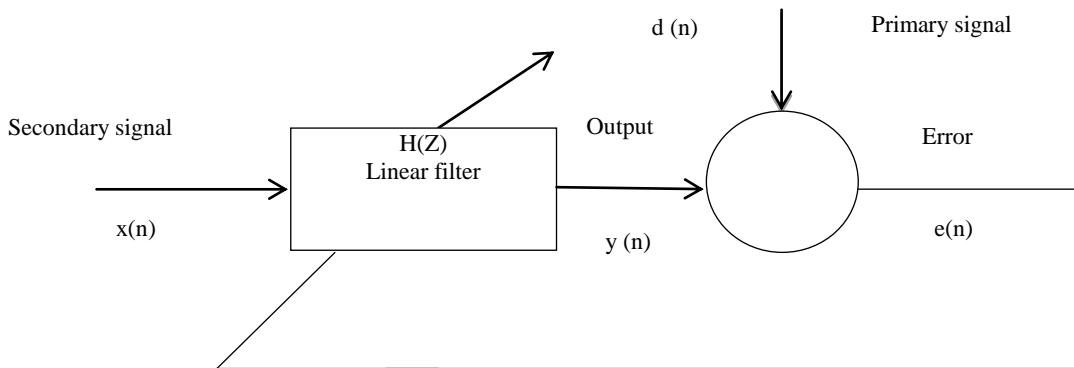
(LMS) algorithm, which is perhaps the most popular method for adjusting the coefficients of an adaptive filter, is given below and also some of its properties are discussed.



**Figure 3.1 Block diagram of adaptive filter**

### 3.3.1 Adaptive Filtering

EEG signal and artifacts have overlapping spectra and the conventional filtering cannot be used to eliminate the artifacts. Because of this problem adaptive filters are opted which are based on the optimization theory. Adaptive filters are very flexible and can modify their properties according to the selected features of the signals that are being analyzed. Figure 3.2 illustrates the structure of basic adaptive filter. There is a primary signal  $d(n)$  and a secondary signal  $x(n)$ . The linear filter  $H(z)$  produces an output  $y(n)$ . The difference between  $y(n)$  and  $d(n)$  is the error  $e(n)$ .



**Figure 3.2 Structure of an adaptive filter**

### 3.3.2 Cancellation of Artifacts

The function of an adaptive filter is to change (adapt) the coefficients of the linear filter which in turn changes its frequency response, to generate a signal similar to the noise present in the signal that has to be filtered. The adaptive process reduces the cost function that is used to determine the filter coefficients. The coefficients are adjusted by adaptive filters to minimize the squared error between its output and a primary signal. In stationary conditions, the filter should converge to the Wiener solution. The converse is also true, that is in non-stationary circumstances, and the coefficients will change with time, according to the signal variation, thus forming an optimum filter.

- **In an adaptive filter, there are basically two processes:**
  - The first process is filtering in which an output signal is the response of a digital filter. Generally FIR filters are used for this purpose because they are simple and stable.
  - The second process is the adaptive process in which the transfer function  $H(z)$  of the filter is adjusted according to an optimization algorithm (Widrow and Stearns 1985). The most commonly used optimization algorithm is Least Mean Square (LMS) algorithm. According to this algorithm, adaptation is

directed by the error signal which is the difference between the primary signal and the filter output.

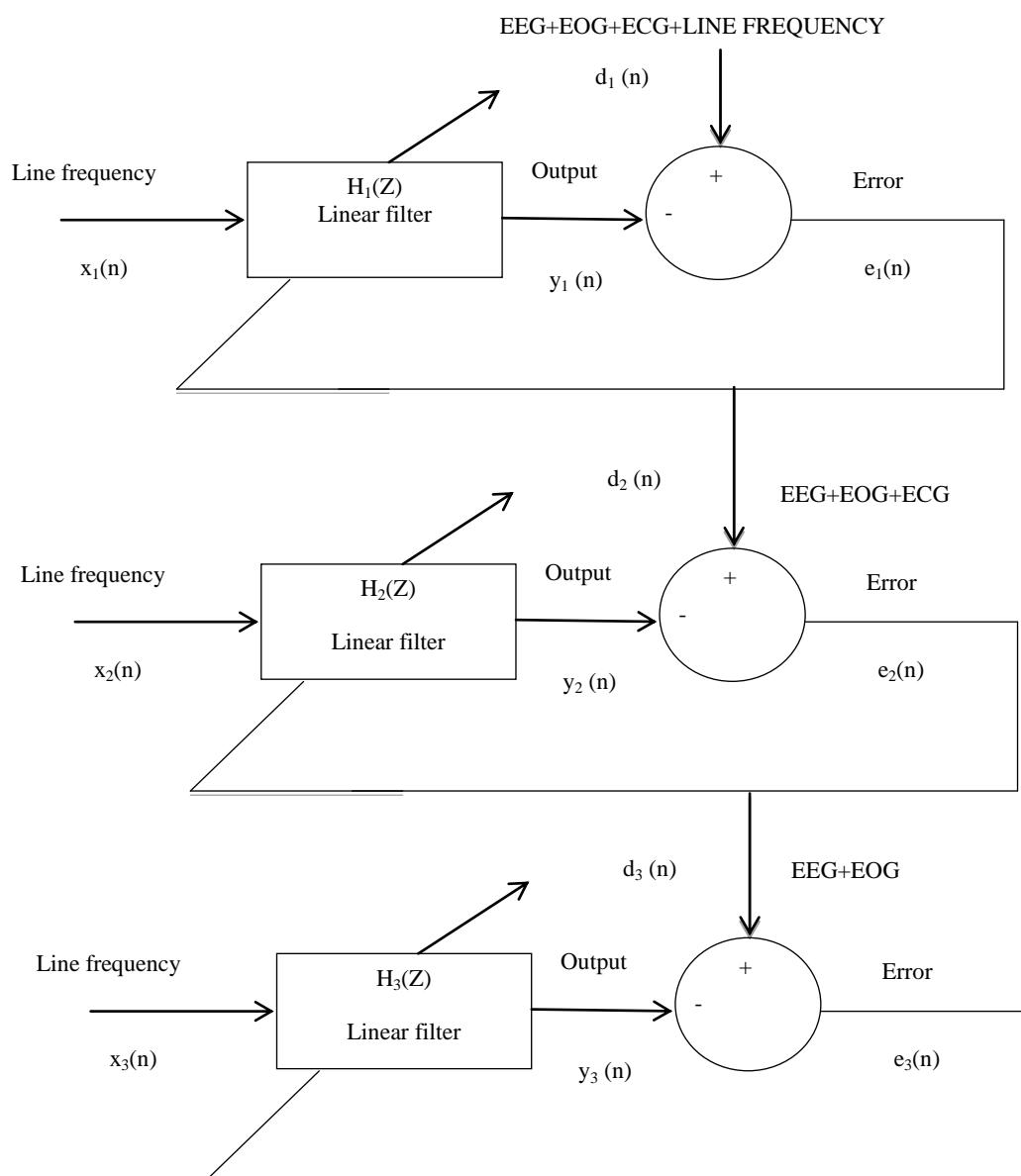
The adaptive interference cancellation is a very efficient method to remove the artifacts from the signals especially when the signals and interferences have overlapping spectra. In external electroencephalogram records and in impedance cardiographs this method is used prominently. Other applications in biomedical signals are removal of maternal ECG in fetal ECG records, detection of ventricular fibrillation and tachycardia, and cancellation of heart sound interference in tracheal sounds.

According to this scheme, the corrupted signal  $d(n)$  is the summation of the desired signal  $s(n)$  and noise  $n_0(n)$ , which is additive and not correlated with  $s(n)$ . Likewise, the reference  $x(n)$  is uncorrelated with  $s(n)$  and correlated with  $n_0(n)$ . The reference  $x(n)$  is fed to the filter to produce an output  $y(n)$  that is a close estimate of the noise  $n_0(n)$ .

To accomplish the objectives of this research, arrangement of a cascade of three adaptive filters was made. The input  $d_1(n)$  to the first stage is the EEG signal with artifacts (EEG + line-frequency + ECG+ EOG). The reference  $x_1(n)$  in the first stage is a sine function generated with 50 or 60 Hz, depending on the type of the line. The optimum values of  $L$  and  $\mu$  are determined by means of various tests. The order  $L$  of  $H_1(z)$ ,  $H_2(z)$  and  $H_3(z)$  were found to be 128 and the coefficient convergence rates  $\mu_1$ ,  $\mu_2$  and  $\mu_3$  were 0.001. The output of  $H_1(z)$  is  $y_1(n)$ , which is an estimation of the line artifact present in the EEG. The error  $e_1(n)$  is the difference between the corrupted signal  $d_1(n)$  and the output of  $H_1(z)$ , which is the EEG without line interference. The adaptive filters cascade form is shown in Figure 3.3.

Thus in the first stage line interference is eliminated. The  $e_1(n)$  error is fed as the corrupted input signal  $d_2(n)$  to the second stage. Real or artificial ECG is given as the reference input  $x_2(n)$  to the second stage. The

output of  $H_2(z)$  is  $y_2(n)$ , which is the close estimate of the ECG artifact present in the EEG record. Signal  $y_2(n)$  is subtracted from  $d_2(n)$  and error  $e_2(n)$  is produced. Thus line and ECG interference is removed from the EEG (Garces Correa et al 2007). Then,  $e_2(n)$  is fed into the third stage as the signal  $d_3(n)$ . The reference input  $x_3(n)$  of filter  $H_3(z)$  is also a real or artificial EOG and its output is  $y_3(n)$ , which is a replica of the EOG artifacts present in the record. The difference between  $y_3(n)$  and  $d_3(n)$  gives error  $e_3(n)$ . It is the final output of the cascade filter, that is, the EEG without artifacts (all the artifacts are removed in the cascaded stages).

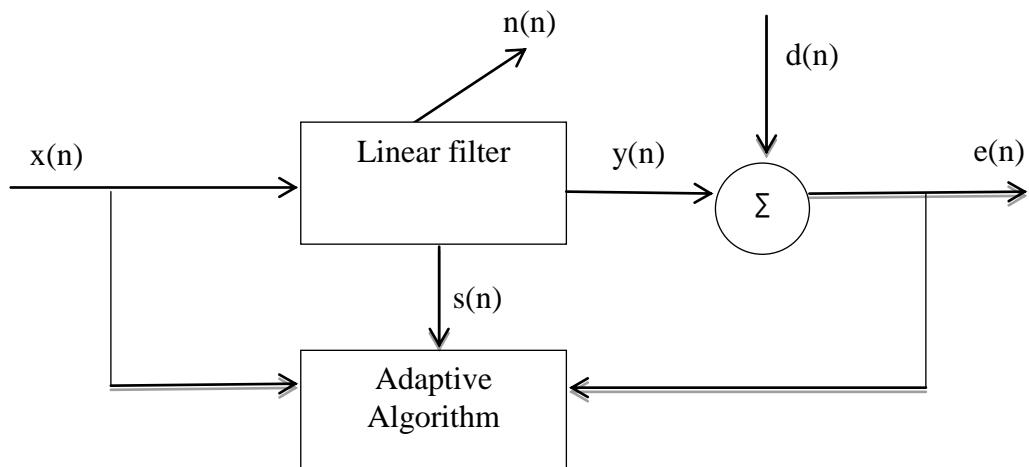


**Figure 3.3 Adaptive filters cascade**

The input signal is the corrupted EEG. The three references are line-frequency, ECG, and EOG which are the artifacts present in EEG. The output  $e_3(n)$  represents the final output which estimates the EEG record without the above mentioned artifacts.

### 3.3.3 Methodology of Adaptive Filter

Figure 3.4 is the basic scheme of adaptive noise canceller where the primary signal is called “corrupted signal” and the secondary is called “reference signal”. It is assumed that the corrupted signal  $d(n)$  is composed of the desired signal  $s(n)$  and noise signal  $n(n)$ , which is additive and not correlated with  $s(n)$ . Likewise, the reference signal  $x(n)$  is uncorrelated with  $s(n)$  and correlated with  $n(n)$ . The reference  $x(n)$  feeds the filter to produce the output  $y(n)$  that is a close estimate of  $n(n)$ .



**Figure 3.4 Adaptive noise canceller scheme**

The structure of the FIR can be represented as in Equation (3.2).

$$y(n) = \sum_{k=0}^L w_k(n-k) \quad (3.2)$$

where  $L$  is the order of the filter,  $x(n)$  is the secondary input signal,  $w_k$  are the filter coefficients and  $y(n)$  is the filter output.

The error signal  $e(n)$  is defined as the difference between the primary signal  $d(n)$  and the filter output  $y(n)$ , that is given in Equation (3.3).

$$e(n) = d(n) - y(n) \quad (3.3)$$

where  $y(n)$  is shown in Equation (3.4)

$$y(n) = d(n) - \sum_{k=0}^L w_k x(n-k) \quad (3.4)$$

The squared error is given in Equation (3.5)

$$e^2(n) = d^2(n) - 2d(n) \sum_{k=0}^L w_k x(n-k) + \left[ \sum_{k=0}^L w_k x(n-k) \right]^2 \quad (3.5)$$

The squared error expectation for  $N$  samples is given by Equation (3.6) and Equation (3.7)

$$\zeta(n) = E[e^2(n)] = \sum_{k=0}^N e^2(n) \quad (3.6)$$

$$\zeta(n) = \sum_{n=1}^N [d^2(n)] - 2 \sum_{K=0}^L w_k r_{dx}(n) + \sum_{K=0}^L \sum_{K=0}^L w_k w_l r_{xx}(k-l) \quad (3.7)$$

where  $r_{dx}(n)$  is the cross-correlation function between primary and secondary input signals and  $r_{xx}(n)$  is the autocorrelation function of the secondary input (Amble 1987), that is given in Equation (3.8) and Equation (3.9).

$$r_{dx}(n) = \sum_{K=0}^N d(n)x(n-k) \quad (3.8)$$

$$r_{xx}(n) = \sum_{n=0}^N x(n)x(n-k) \quad (3.9)$$

Adaptation process is used to minimize the squared error which describes the performance of the filter. To achieve this goal different optimization techniques can be used. Steepest descent method is used in this work (Sornmo et al 2005).

The filter coefficient vector for each iteration  $k$  having information about the previous coefficients and gradient, multiplied by a constant can be calculated with the help of steepest descent method. It is given by in Equation (3.10)

$$w_k(n+1) = w_k(n) + \mu(-\Delta_k) \quad (3.10)$$

where  $\mu$  is a coefficient that the rate of adaptation.

Deriving with respect to  $w_k$  and replacing leads to,  $w_k(n+1)$  in Equation (3.11), Equation (3.12), Equation (3.13) and Equation (4.14).

$$w_k(n+1) = w_k(n) - \mu \frac{\partial \{e^2(n)\}}{\partial w_k(n)} \quad (3.11)$$

$$w_k(n+1) = w_k(n) - 2\mu e(n) \frac{\partial \{e^2(n)\}}{\partial w_k(n)} \quad (3.12)$$

$$w_k(n+1) = w_k(n) - 2\mu e(n) \frac{\partial \{d(n) - \sum_{k=0}^L w_k x(n-k)\}}{\partial w_k(n)} \quad (3.13)$$

Since  $d(n)$  and  $x(n)$  are independent with respect to  $w_k$ , then,

$$w_k(n+1) = w_k(n) - 2\mu e(n) x(n-k) \quad (3.14)$$

Equation (3.14) is the final description of the algorithm that is used to compute the filter coefficients as function of the signal error  $e(n)$  and the reference input signal  $x(n)$ . The coefficient  $\mu$  is a constant that must be

chosen carefully for quick adaptation and must be chosen in such a way to maintain stability.  $\mu$  must satisfy the following condition for the filter to be stable. It can be defined in Equation (3.15)

$$0 < \mu < \frac{1}{(10 \cdot L \cdot P_{xx})} \quad (3.15)$$

where  $L$  is the filter order and  $P_{xx}$  is the power of the input signal computed as, follows in Equation (3.16).

$$P_x \approx \frac{1}{M - 1} \sum_{n=0}^{M-1} x^2(n) \quad (3.16)$$

Advantages of adaptive filters over conventional ones include preservation of components intrinsic to the EEG record.

### 3.3.4 Advantages

The advantages of using a cascade of three filters instead of using a single adaptive filter to filter three signals are,

- a) It is simple to adapt the coefficients individually rather than doing it in a single step in a single filter.
- b) At each stage one artifact is removed from the corrupted signal; such separation (by artifact) may be useful in some applications where the removal of one artifact might be enough. The advantages of adaptive filters over conventional ones include preservation. Conventional filters might destroy some details of sensitive EEG whereas in adaptive filters intrinsic components of EEG record are well preserved. Besides, they can adapt their coefficients to variations in heart frequency, abrupt changes in the line frequency (caused, say,

by ignition of electric devices) or modifications due to eye movements

### **3.3.5 Limitation of Adaptive Filtering Method**

The main problem in this work was the determination of L (filter order) and  $\mu$  (convergence factor). These parameters are very important; L is necessary for the appropriate filtering, and  $\mu$  is used to get an adequate adaptation. The  $\mu$  value has to be selected carefully. The large values of  $\mu$  lead to instability and the small value turns out to be inefficient and thus the adaptation is too slow. Multiple numbers of tests were carried out to determine the optimum value for these parameters.

## **3.4 ARTIFICIAL NEURAL NETWORKS**

An Artificial Neural Network is an information-processing algorithm that works based on the biological nervous systems such as the brain. The key element of this neural network is novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements called neurons. They work together to solve specific problems. Neural networks, like humans, learn by example. An Artificial Neural Network is application oriented, such as pattern recognition or data classification, through a learning process.

### **3.4.1 Neural Networks**

The computation on artificial neural networks has been motivated right from its inception by that of the human brain. The computation of human brain is entirely different from the conventional digital computer. The brain is a highly complex, non-linear and parallel computer (information processing system). The networks has to organize its structural constituents,

known as neurons to perform certain computations (e.g. pattern recognition, perception and motor control) many times faster than the fastest digital computer in existence today.

Artificial neural networks are biologically inspired. The neural networks are processing elements capable of doing processing in parallel and distributed manner as in human brain. Various models are developed based on neuro-processing, each with a variation on parallel and distributed processing ideas. A general framework has to be built in order to categorize the various models that are developed.

Artificial neural networks are a massive interconnection of simple computing cells referred to as “neurons” or “processing units”. A neural network is a large parallel-distributed processor made up of simple processing units which has a new natural propensity for storing experiential knowledge and making it available for use.

➤ **It resembles the brain in two aspects:**

- The network gains knowledge from its environment through a process called learning process.
- Inter connection strengths, known as synaptic weights, are used to store the acquired knowledge.

A set of training data is used to train artificial neural network and they do not operate based upon a set of rules as in Expert Systems or based on fixed algorithms. These networks feature a large parallel architecture that uses a neuron as the basic processing element.

### 3.4.2 Features of Neural Networks

Artificial Neural Networks comprises of elements that perform in a manner that is similar to the most elementary functions of the biological neuron. These networks exhibit a surprising number of brain's characteristics. They learn from experience, generalize from previous examples to new ones, and abstract essential characteristics from input containing irrelevant data.

- **Learning:** Artificial neural network can change their behavior in response to the changes in the environment. Given a set of input, they self-adjust to produce suitable responses. This behavior of self-adjustment is called “Learning”.
- **Generalization:** After training the network response can be insensitive to small variations in its inputs. This ability to see through noise and distortion in the pattern lies within its vital to pattern recognition in the real world environment.
- **Abstraction:** The essence of a set of inputs can be abstracted by some types of neural networks. For example, a network can be trained on a sequence of distorted versions of a character. After sufficient training, application of such a distorted pattern will produce a perfectly formed character. In other words, it has learnt to produce something new and different that it has never seen before.
- **Non-Linearity:** The processing elements or nodes used in networks are non-linear, typically analog. Three types of non-linearity are used
  - Head Limiter
  - Threshold logic elements and

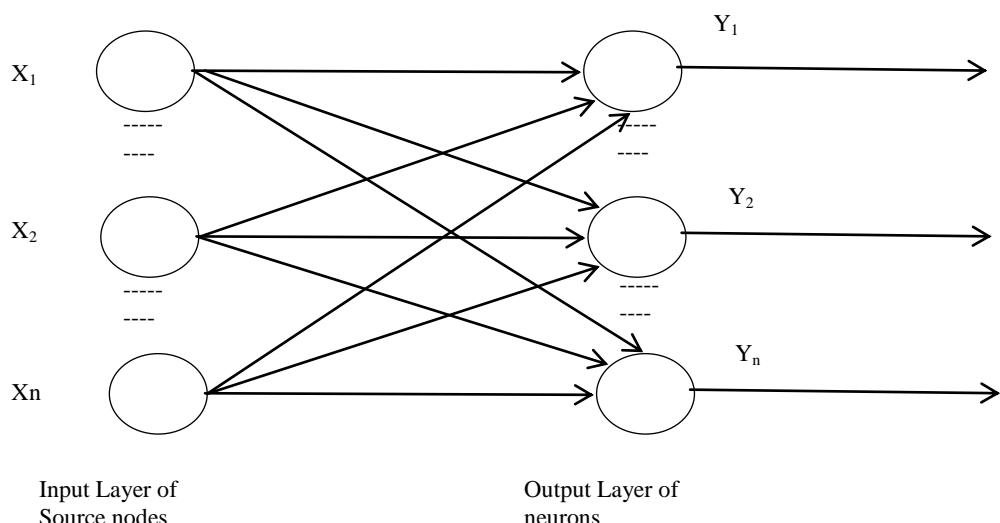
- Sigmoid non-linearity

### 3.4.3 Architecture of Artificial Neural Network

The way in which the neurons of a neural network are structured and how intimately linked with the learning algorithm is used to train the network. General neural networks can be classified into two types based on their architecture.

#### 3.4.3.1 Single-layer feed forward network

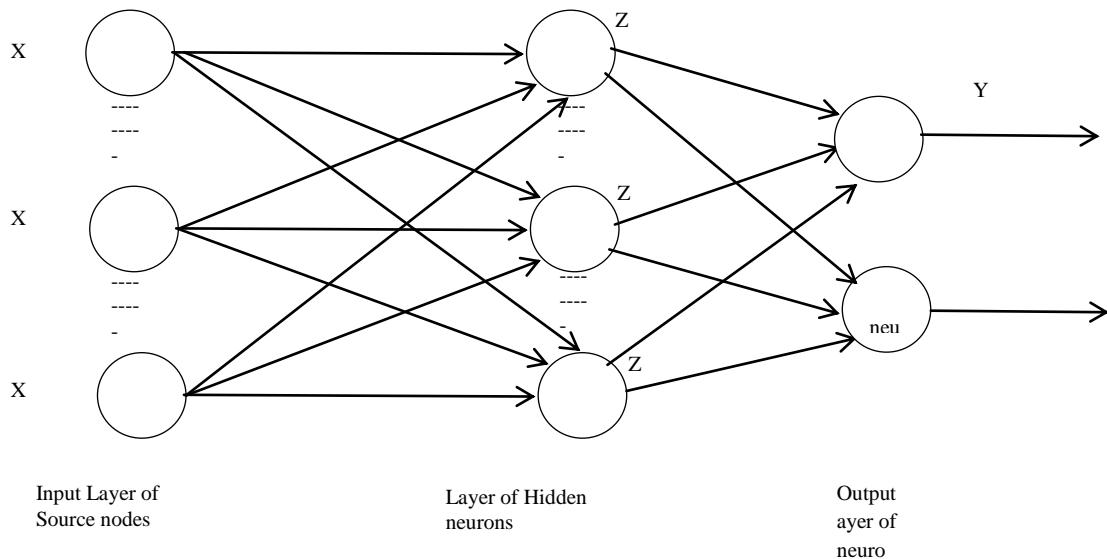
In a layered neural network the neurons are organized in the form of layers. In this simplest form of a layered network, there is an input layer of source nodes that are mapped and projected onto an output layer of neurons (computation nodes), as shown in Figure 3.5. This type of network is strictly a feed forward or a cyclic type. Such a network is called a single-layer network where the term “single-layer” refers to the output layer of computation nodes (neurons).



**Figure 3.5 Single-Layer Feed Forward Network**

### 3.4.3.2 Multi-layer feed forward network

The multi-layer feed forward neural network distinguishes itself by the presence of one or more hidden layers whose computation nodes are correspondingly called hidden neurons or hidden units. Figure 3.6. represents the structure of multi-layer feed forward network. The hidden neuron is used to intervene between the external input and the network output in a some useful manner. The addition of one or more hidden layers enables the network to extract higher-order statistics (Ali 2003). When the size of the input layer is very large the ability of hidden neurons to extract higher-order statistics is particularly valuable.



**Figure 3.6 Multi-Layer Feed Forward Networks**

## 3.5 LEARNING METHODS

The important property of a neural network is its ability to learn from its environment, and to improve its performance through learning. Learning in the context of neural network can be defined as:

Learning is a process by which the free parameters of a neural network are adapted according to the changes in the environment through a process of simulation by the environment in which the network is embedded. The type of learning is determined by the way in which the parameter changes take place in the environment.

This definition of the learning process implies the events that are given in the following sequence. An environment stimulates the neural network.

- As a result of simulation the neural network undergoes changes in its parameters.
- Because of the changes that have occurred in its internal structure, the neural network responds in a new way to the environment.

Learning algorithm is used to prescribe solutions to the problems encountered based on a set of well-defined rules. In general, the learning algorithms differ from each other in the way the adjustment to a synaptic weight of a neuron is formatted. Another factor to be considered is the way in which a neural network (learning machine), made up of a set of interconnected neurons are related to its environment.

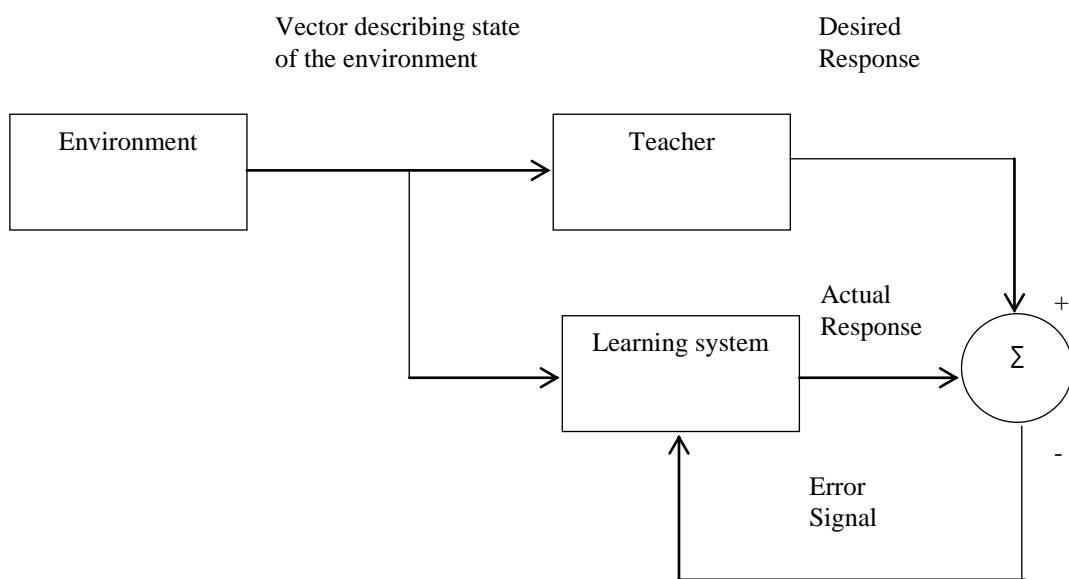
➤ **In neural network, there are five basic learning rules:**

- Error – correction learning which is rooted in optimum filtering.
- Memory – based learning, which operates by memorizing the training data explicitly.
- Hebbian learning, which is inspired by neuro – biological considerations

- Competitive learning, which is also inspired by neuro – biological considerations.
- Boltzmann learning, this is based on the ideas borrowed from the statistical mechanics.

### 3.5.1 Supervised Learning

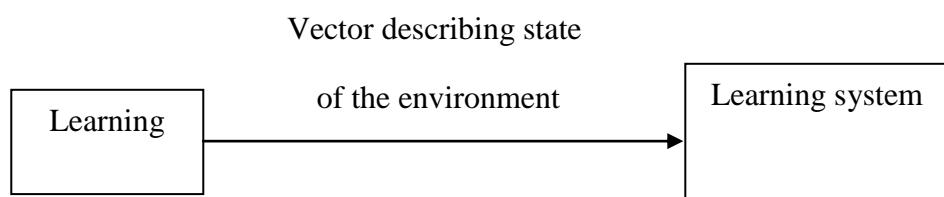
Figure 3.7. shows a block diagram that illustrates the structure of supervised learning. In conceptual terms, a teacher having knowledge of the environment can be represented as a set of input–output examples. The environment that is unknown to the neural network is the region of interest. Consider that the teacher and the neural network are both exposed to a training vector (i.e., example) drawn from the environment. By virtue of built – in prior knowledge, the teacher is able to provide the neural network with a desired response for that training vector. Based on the training vector and the error signal the network parameters are adjusted.



**Figure 3.7 Supervised Learning**

### 3.5.2 Unsupervised Learning

In unsupervised or self – organized learning there is no external teacher or critic to oversee the learning process, as indicated in Figure 3.8. Rather, provision is made for independent learning of the network, and the parameters of the network are optimized with respect to that measure. Once the network has become familiar to the statistical, regularities of the input data, it develops the ability to form internal representations for encoding features of the input and thereby to create new classes automatically.



**Figure 3.8 Unsupervised Learning**

Competitive learning has to be followed to perform unsupervised learning. For example, a neural network, may be considered that consists of two layers, an input layer and a competitive layer. The input layer receives the available data. The competitive layer consists of neurons that compete with each other (in accordance with a learning rule) for the opportunity to respond to the features contained in the input data. The principle behind this is simple, the network operates in accordance with “winner takes all” strategy. The neuron with a greater number of total inputs wins the competition and turns on; all the other neurons are simply switched off.

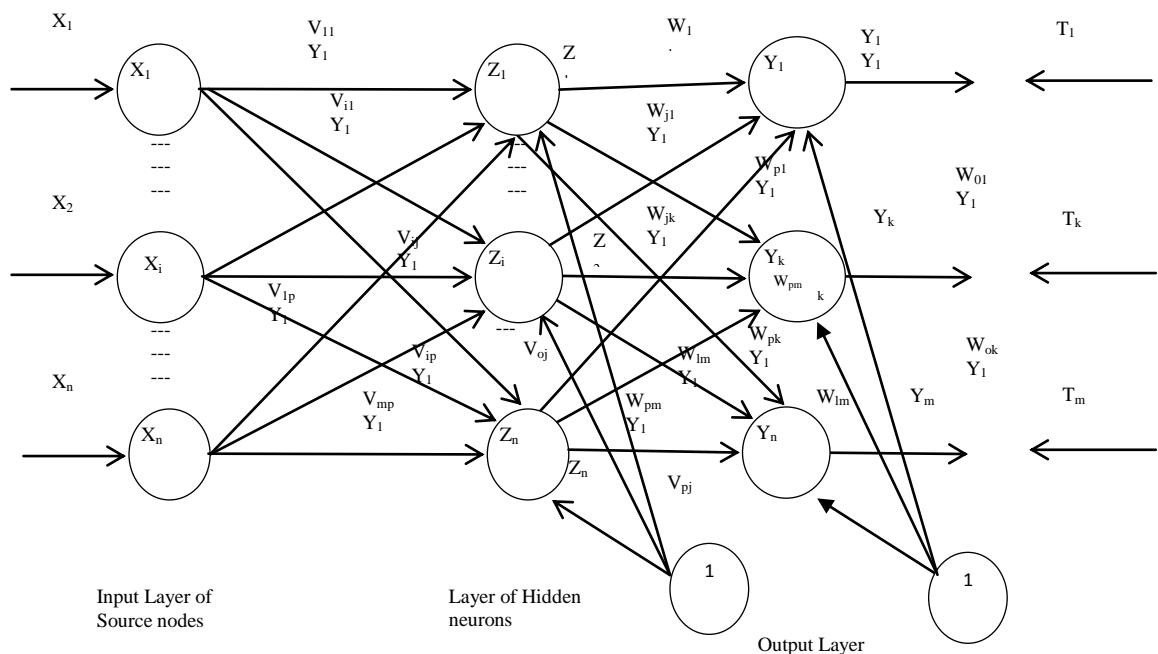
## 3.6 BACK PROPAGATION NETWORK

Back propagation is a systematic method that is used to train multilayer artificial neural networks. It is built on an important mathematical foundation and has very efficient application potential. The problems are solved with the help of back propagation network in many areas. A multi-

layer network can learn only input patterns to an arbitrary accuracy level. Normally one hidden layer is enough however more than one hidden layer is used. The weight in a neural network represents the segment of the information about the input signal that has to be stored.

### 3.6.1 Architecture of Feed Forward Back Propagation Network

Back Propagation neural network is the second type of neural network that is a multi-layer, feed forward neural network with an input layer, an output layer and a hidden layer as shown in Figure 3.9. The neurons in the hidden and output layers have biases that are connections from units whose output must always be equal to 1. The inputs are fed to the back propagation network and the output obtained from it could be in binary (0, 1) or bipolar (-1, +1). The activation function is a function that increases monotonically and is also differentiable (Huang et al 2006). The back propagation network implements the generalized delta rule. It is a gradient descendent method, which minimizes the total squared error of the output of the network



**Figure 3.9 Architecture of Feed Forward Back Propagation Network**

### **3.7 FUZZY SYSTEM**

Among the various changes in science and mathematics in this century, one important change is the concept of uncertainty. In science, this change has been manifested by a gradual transition from the traditional view of avoiding uncertainties by all possible means. According to alternative view, uncertainty is considered as inevitable and essential part of science, it is not only unavoidable plague, but also has a greater utility.

In constructing model, the three key characteristics of a system are model complexity, credibility and uncertainty. Allowing more uncertainty reduces the complexity and increase the credibility of the resulting model (Abdulhamit Subasi and Ergun Ercelebi 2005). The transition from traditional view to the modern view of the uncertainty is based on the emergence of several view theories of uncertainty, distinct from probability theory. They show that probability theory is capable of representing only one of several distinct types of uncertainty.

Lofti Zeadh published the modern concepts of uncertainty. He developed a concept based on the objects called fuzzy sets. The fuzzy sets, are sets with boundaries that are not precise. The membership in a fuzzy set is not a matter of denial, but rather a matter of a degree.

According to Zeadh's uncertainty, consider that A is a fuzzy set and X is a relevant object, then the proposition that "X is a member of A" is not necessarily true or false, as required by two value logic, but it may be true only to some degree, the degree to which X is actually a member of A. The capability of fuzzy to express a gradual transition from membership to non-membership and vice-versa has broad utility. So, it produces a meaningful representation of vague concepts that are expressed in natural language.

A fuzzy set can be defined by mathematical expressions by assigning to each possible individual in the universe of discourse, a value representing its grade in the fuzzy set (Lin and Lee 1995). Fuzzy sets support a flexible sense of membership of elements to a set. According to crisp set theory, an element either belongs to or does not belong to a set, whereas in fuzzy set theory many degrees of membership (between 0 and 1) are allowed.

### **3.7.1 Fuzzy Sets**

A set can be specified by its members, as they characterize a set completely. The list of members  $A = \{0, 1, 2, 3\}$  is a finite set. Nobody can list all elements of an infinite set, and it can state some property which characterizes the elements in the set, for instance the predicate  $x > 10$ . That set is defined by the elements of the universe of discourse which make the predicate true. There are two ways to describe a set: explicitly in a list or implicitly with a predicate.

The crisp set is defined in such a way that the individuals can be divided only into two groups; members and non-members. A fuzzy set can be defined mathematically by assigning to each possible individual in the universe of discourse a value representing its grade in the fuzzy set.

### **3.7.2 Membership Function**

Membership function does not represent the degree to which a fuzzy number belongs to a set. Rather it is the degree of truthfulness. The membership function can take values only between zero and one.

Every element in the universe of discourse is a member of the fuzzy set to some grade that ranges from zero to one. The set of elements that have a non-zero membership is called the support of the fuzzy set (Lin and

Cunningham 1995). The membership function  $\mu(x)$  is the function that ties a number to each element  $x$  of the universe.

Usually the membership function can take many forms in fuzzy set. The most commonly used forms are Triangular, Trapezoidal, Parabolic and Bell shaped. The determination of the shape is application oriented, that is the suitable shape of representation is determined only by the context of a particular application.

### **3.7.3      Advantages of Fuzzy Logic**

The advantages of Fuzzy logic are as follows

- Linguistic, not numerical, variables are used making it similar to the way humans think.
- The solution of previously unsolved problems can be used to model the traditional control systems with complex analytical Equations. This is due to their simplicity.
- Rapid prototyping is possible because a system designer is not required to have a complete prior knowledge about the system before starting work.
- It has high robustness.

### **3.7.4      Fuzzification**

Fuzzification is a process of converting real valued variable into fuzzy variable. Fuzzy variable is system dependent that depends on the nature of the system where it is implemented. Fuzzification involves the following functions

- Measures the values of input variables
- Performs the function of mapping that transforms the range of input variable into the corresponding universe of discourse.
- Performs the function of fuzzification to convert the input data into suitable linguistic values that are viewed as the labels of the fuzzy set.

In general, the linguistic variables are labelled as one of the following SMALL, MEDIUM, LARGE and so on; it differs in different applications and also differs in different sense of humans.

#### **3.7.4.1 Knowledge base**

The knowledge base consisting of fuzzy IF-THEN rules is the heart of fuzzy system. The rule base is a collection of a set of fuzzy rules. The knowledge base contains the membership function of fuzzy subsets. Fuzzy rules may contain fuzzy variables and fuzzy subsets that are characterized by membership function.

#### **3.7.4.2 Fuzzy inference engine**

Inference from a set of fuzzy rules involves fuzzification of the conditions of the rules, then propagating the membership values of the conditions to the outcomes of the rules.

There are two types of fuzzy IF-THEN rules that are most commonly used.

- Mamdani type
- Sugeno type

In mamdani type both the input and output of the fuzzy controller are assigned as fuzzy sets whereas in sugeno type, only the input is a fuzzy set and output maybe a linear or constant element (Lin and Lu 1995).

There are three processes for interpreting if-then rule:

- Resolve all fuzzy statements in the antecedent to a degree of membership between the values 0 and 1.
- If there are multiple parts to the antecedent, apply fuzzy logic operators and resolve the antecedent to a single number between 0 and 1
- Apply the implication method, using the degree of support for the entire rule to shape the output which maybe a fuzzy set or linear element.

Considering the following rule,

➤ **IF AND THEN credit limit is low:**

Inference from this above rule involves (using fuzzification) looking up the membership value (MV) of the condition 'applicant is young' given the applicant's age, and the MV of 'income is low' given the applicant's salary. The method proposed by Lotfi Zade takes the minimum MV of all the conditions available and assigns it to the outcome 'credit limit is low'. This method is enhanced by having a weight for each rule between 0 and 1 which multiplies the MV assigned to the outcome of the rule. This weight can be assigned based on the Pattern rules view, or assigned at run time. By default each rule weight is set to values between 0 and 1.

In a fuzzy rule base a number of rules with the outcome 'credit limit is low' will be fired. The inference engine will assigns the outcome 'credit limit is low', with the maximum MV available from all the fired rules.

➤ **To summarize the fuzzy inference involves the following:**

- Defuzzification is the process where conditions of each rule and assigning the outcome of each rule the minimum MV of its conditions multiplied by the rule weight
- Assigns each outcome with the maximum MV available from its fired rules.
- The outcome of fuzzy inference is the confidence factors (MVs) assigned to each outcome in the rule base.

### **3.8 DE-FUZZIFICATION**

De-fuzzification is done to convert the output fuzzy variable to a crisp value so that it can be used for control purpose. The controller performance is determined by the membership functions, knowledge base and method of de-fuzzification.

The most often used manmade de-fuzzification methods are

- Centre- of –area defuzzification
- Centre –of –sums defuzzification
- First- of – maxima defuzzification
- Middle – of – maxima defuzzification
- Height defuzzification

The most often used sugeno de-fuzzification methods are

- Weighted average defuzzification
- Weighted sum defuzzification

### **3.9 NEURO- FUZZY SYSTEM:**

In the field of artificial intelligence, neuro-fuzzy is the combination of artificial neural networks and fuzzy logic. Neuro-fuzzy was proposed by J. S. R. Jang. Neuro-fuzzy hybridization results in a hybrid intelligent system that combines the advantages of these two techniques by combining the human-like reasoning style of fuzzy systems with the learning and connectionist structure of neural networks. Neuro-fuzzy hybridization is widely termed as Fuzzy Neural Network (FNN) or Neuro-Fuzzy System (NFS) in the literature. Neuro-fuzzy system (the more popular term is used henceforth) incorporates the human-like reasoning style of fuzzy systems by using fuzzy sets and a linguistic model consisting of a set of IF-THEN fuzzy rules. The main strength of neuro-fuzzy systems is that they are universal approximators with the ability to solicit interpretable IF-THEN rules.

The strength of neuro-fuzzy systems is based on two contradictory requirements in fuzzy modeling: interpretability versus accuracy. In practice, one of the two properties prevails (García et al 2008). The neuro-fuzzy in fuzzy modelling research field is divided into two areas: linguistic fuzzy modelling that focuses on interpretability, mainly the Mamdani model; and precise fuzzy modelling that focuses on accuracy, mainly the Takagi-Sugeno-Kang (TSK) model.

Although generally assumed to be the realization of a fuzzy system through connectionist networks, this term is also used to describe some other configurations including the following:

➤ **Deriving fuzzy rules from trained RBF networks.**

- Fuzzy logic based tuning of neural network training.
- Fuzzy logic criteria for increasing a network size.
- Realizing fuzzy membership function through clustering algorithms learning in SOMs and neural networks.
- Representing fuzzification, fuzzy inference and defuzzification through multi-layers feed forward connectionist networks.

It must be pointed out that interpretability of the Mamdani-type neuro-fuzzy systems can be lost during hybridization. Certain measures need to be taken to improve the interpretability of neuro-fuzzy systems, wherein important aspects of interpretability of neuro-fuzzy systems are also discussed.

➤ **Integration of fuzzy logic and neural networks**

Hybrid systems combining fuzzy logic, neural networks, genetic algorithms, and expert systems are providing solutions for a wide variety of real-world problems. Each intelligent technique has its own computational properties (e.g. ability to learn, explanation of decisions) that make them suitable for particular problems and not for others. For example, while neural networks are good at recognizing patterns, they are not good at explaining how they reach their decisions. Fuzzy logic systems, which can reason with imprecise information, are good at explaining their decisions but

they cannot automatically acquire the rules they use to make those decisions. These limitations have been a central driving force behind the creation of intelligent hybrid systems where two or more techniques are combined in a manner that overcomes the limitations of individual techniques. Hybrid systems are also important while considering the varied nature of application domains. Many complex domains have many component problems, each of which requires different types of processing.

If there is a complex application which has two distinct sub-problems, say a signal processing task and a serial reasoning task, they are handled separately by a neural network and an expert system respectively can be used for performing these separate tasks. The intelligent hybrid systems cover a wide range of applications including process control, engineering design, financial trading, credit evaluation, medical diagnosis, and cognitive simulation.

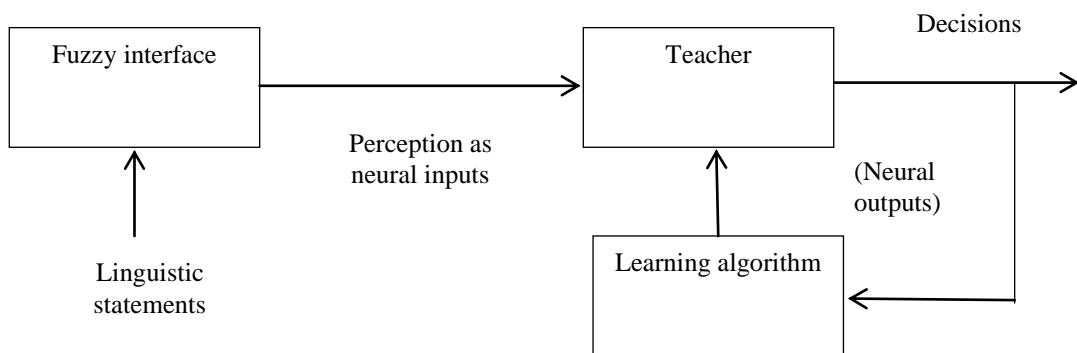
The fuzzy logic provides an inference mechanism under cognitive uncertainty, whereas computational neural networks offer exciting advantages, like learning, adaptation, fault-tolerance, parallelism and generalization. To enable a system to deal with cognitive uncertainties in a manner similar to humans one has to incorporate the concept of fuzzy logic into the neural networks. The computational process of fuzzy neural systems is explained below. The initial stage is the development of a “fuzzy neuron” based on the understanding of biological neuronal morphologies, followed by learning mechanisms. This leads to the following three steps in the process of a fuzzy neural computation

- Fuzzy neural models are developed from the motivation of biological neurons,
- Models of synaptic connections which incorporates fuzziness into neural network,

- Learning algorithms are developed (that is the method of adjusting the synaptic weights)

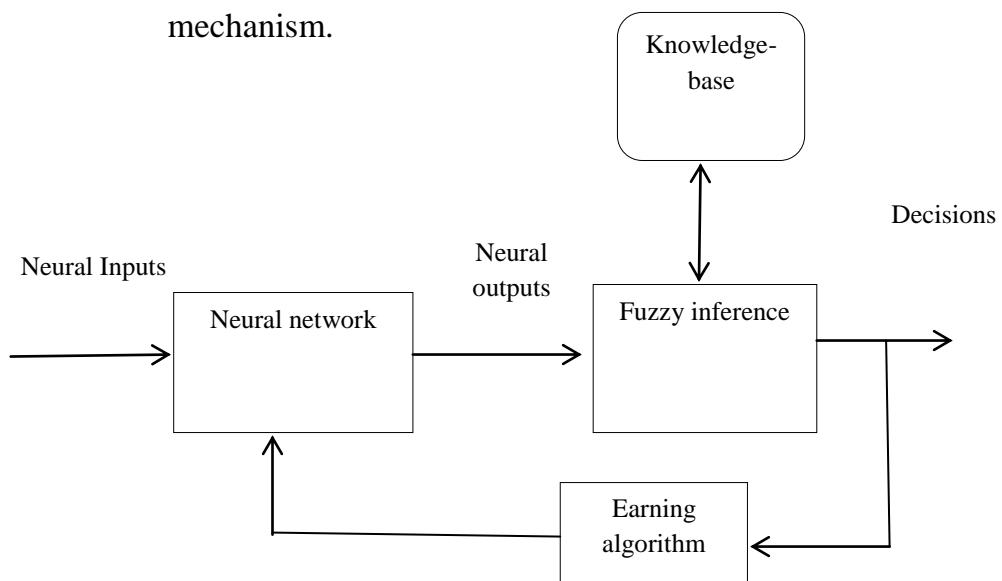
➤ **Two possible models of Neuro Fuzzy Systems are**

- In accordance with linguistic statements, the fuzzy interface block feeds an input vector to a multi-layer neural network. The neural network can adapt to changes (trained) to yield the desired command outputs or decisions. Figure 3.10 shows the first model of fuzzy neural system.



**Figure 3.10.The first model of fuzzy neural system.**

- A multi-layered neural network drives the fuzzy inference mechanism.



**Figure 3.11 The Second model of fuzzy neural system**

The membership functions of fuzzy systems are tuned by the Neural networks that are employed as decision-making systems for controlling equipment. Although fuzzy logic can encode an expert knowledge directly using rules with linguistic labels, it takes more time to design and tune the membership functions which quantitatively define these linguistic labels. The time and cost can be reduced by using Neural network learning techniques which can also automate this process improving performance .Figure 3.11 shows the second model of fuzzy neural system.

In theory, neural networks, and fuzzy systems are convertible that is, they are equivalent, yet in practice, each has its own advantages and disadvantages. The neural networks can acquire knowledge automatically by the back propagation algorithm, but the learning process is relatively slow and the analysis of the trained network is difficult (black box). The structural knowledge(rules) cannot extract from the trained neural network or integrate special information about the problem into the neural network in order to simplify the learning procedure.

Fuzzy systems are much more simple and favorable because their behavior is based on fuzzy rules and thus their performance can be adjusted by making small changes in the rules. But knowledge acquisition is very difficult in fuzzy systems the universe of discourse of each input variable needs to be divided into several intervals.Applications of fuzzy systems are restricted to the fields where expert knowledge is available and the number of input variables is small.

To overcome the problem of knowledge acquisition, neural networks can be extended to automatically extract fuzzy rules from numerical data. Cooperative approaches use neural networks to preprocess data and for optimization of certain parameters of an ordinary fuzzy system and extract fuzzy (control) rules from data. A fuzzy-neuro system is broadly classified

into the fuzzy neural structure as feed forward (static) and feedback (dynamic) based on the computations involved in the process.

A typical fuzzy-neuro system that is more commonly used is Basenji's ARIC (Approximate Reasoning Based Intelligent Control) architecture. It is a neural network model of a fuzzy controller and learns to update its prediction based on the behaviour of the physical system and finetunes a predefined control knowledge base according to the updation.

### **3.10 PROPOSED METHODOLOGY OF NEURO-FUZZY FILTER**

#### **3.10.1 Preprocessing of EEG**

The many dimensions of EEG signal contain only empty noises. Hence the process of reducing the dimensionality of EEG signal is called preprocessing. The preprocessing of the EEG signal is performed to enhance the analysis on these signals. The removal of the artifacts and short-time high-amplitude events enable one to highlight important characteristic features in the EEG signals (Rizon 2010). The main objective of signal pre-processing is to extract necessary information from the sensor responses and prepare the EEG signal for multivariate pattern analysis. There is a number of different methods and tools available to carry out preprocessing. Some of the methods are sampling, feature extraction, transformation and normalization. Using one among these methods one goes for feature extraction. This method pulls out specified data that is significant in some particular context and these pulled out data yield more information about the signal that can be used for further analysis and noise removal.

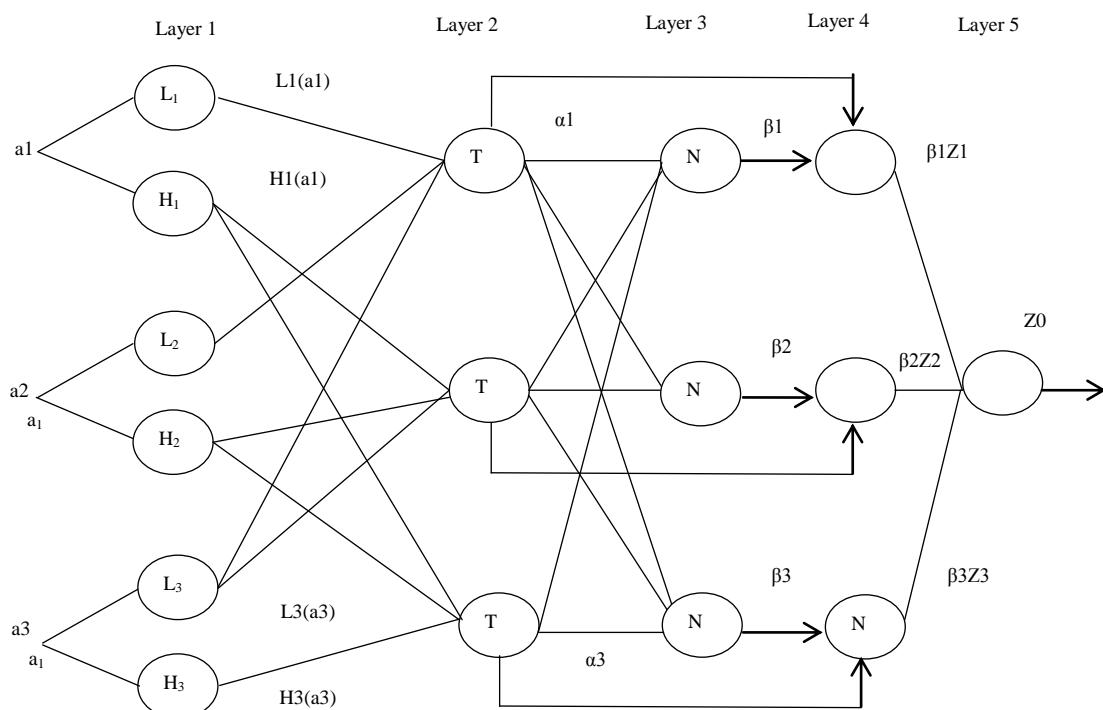
The computational process of fuzzy neural systems is explained below. The initial stage is the development of a “fuzzy neuron” based on the understanding of biological neuronal morphologies followed by learning mechanisms. This leads to the following three steps in the process of a fuzzy

neural computation

**Step I** : Development of fuzzy neural models motivated by biological neurons,

**Step II** : Models of synaptic connections which incorporates fuzziness in to Neural network,

**Step III** : Development of learning algorithms (that is the method of adjusting the synaptic weights).



**Figure 3.12 Structure of Neuro-Fuzzy Filter**

The Figure 3.12 gives the pictorial representation of the neuro-fuzzy filter, that includes subnet works (Bednar et al 1987). The circle denotes aggregation techniques. In this study, signals namely artifacts and delayed artifacts have been taken as inputs and measured EEG signal as target for training the Neuro-Fuzzy filter structure used a generalized bell type as

membership function for tuning the parameters. The filter has the following structure:

**Layer 1 :** The output of the node is the degree to which the given input satisfies the linguistic label associated with this node.

**Layer 2 :** Each node computes the firing strength of the associated rule. The nodes of this layer are called rule nodes.

The output of top neuron is given in Equation (3.17).

$$\alpha_1 = L_1(a_1) \wedge L_2(a_2) \wedge L_3(a_3) \quad (3.17)$$

The output of the middle neuron is given in Equation (3.18).

$$\alpha_2 = H_1(a_1) \wedge H_2(a_2) \wedge L_3(a_3) \quad (3.18)$$

The output of the bottom neuron is given in Equation (3.19).

$$\alpha_3 = H_1(a_1) \wedge H_2(a_2) \wedge H_3(a_3) \quad (3.19)$$

**Layer 3 :** This layer is labelled as N to indicate the normalization of the firing levels. The output of the top, middle and bottom neuron is the normalized firing level of the corresponding rule. It is given in Equation (3.20), Equation (3.21) and Equation (3.22).

$$\beta_1 = \frac{\alpha_1}{\alpha_1 + \alpha_2 + \alpha_3} \quad (3.20)$$

$$\beta_2 = \frac{\alpha_2}{\alpha_1 + \alpha_2 + \alpha_3} \quad (3.21)$$

$$\beta_3 = \frac{\alpha_3}{\alpha_1 + \alpha_2 + \alpha_3} \quad (3.22)$$

**Layer 4 :** The output of the top, middle and bottom neuron is the product of the normalized firing Level. (Medsker et al 1993). It is shown in Equation (3.23), Equation (3.24)and Equation (3.25).

$$\beta_1 = \beta_1 V B^{-1}(\alpha_1) \quad (3.23)$$

$$\beta_2 z_2 = \beta_2 B^{-1}(\alpha_2) \quad (3.24)$$

$$\beta_3 z_3 = \beta_3 S^{-1}(\alpha_3) \quad (3.25)$$

**Layer 5 :** The single node in this layer computes the overall system output as the sum of all incoming signals. The out put is given in Equation (3.26).

$$z_0 = \beta_1 z_1 + \beta_2 z_2 + \beta_3 z_3 \quad (3.26)$$

Assume that the Neuro-fuzzy filter has two inputs  $x, y$  and one output  $z$ .

**Rule 1 :** If  $x$  is  $A_1$  and  $y$  is  $B_1$ , it denoted as in Equation (3.27).

$$f_1 = p_1 x + q_1 y + r_1 \quad (3.27)$$

**Rule 2 :** If  $x$  is  $A_2$  and  $y$  is  $B_2$ , it denoted as in Equation (3.28).

$$f_2 = p_2 x + q_2 y + r_2 \quad (3.28)$$

where  $p, q$  and  $r$  represent consequent parameters. A and B are linguistic labels.

### 3.11 FUZZY NEURONS

Let a simple neural net in Figure 3.13 be considered. All signals and weights are real numbers. The two input neurons do not change the input signals so their output is the same as their input. The signal  $x_i$  interacts with the weight  $w_i$  to produce the product it is given in Equation (3.29).

$$p_i = w_i x_i, i = 1, 2. \quad (3.29)$$

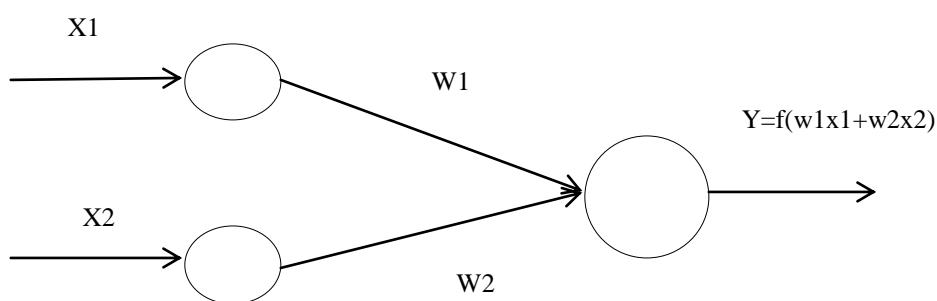
The input information  $p_i$  is aggregated, by addition, to produce the input it is given in Equation (3.30).

$$\text{net} = p_1 + p_2 = w_1 x_1 + w_2 x_2 \quad (3.30)$$

to the neuron. The neuron uses its transfer function  $f$ , which could be a sigmoidal function,  $f(x) = (1 + e^{-x}) - 1$ , to compute the output  $y$  is given by Equation (3.31).

$$y = f(\text{net}) = f(w_1 x_1 + w_2 x_2). \quad (3.31)$$

This simple neural net, which employs multiplication, addition, and sigmoidal  $f$ , will be called as regular (or standard) neural net. Figure 3.13 shows simple neuron set.



**Figure 3.13 Simple neuron set**

If one employs other operations employ like a t-norm, or a t-conorm, to combine the incoming data to a neuron one obtains what is called hybrid neural net.

These modifications lead to a fuzzy neural architecture based on fuzzy arithmetic operations. Let us express the inputs (which are usually membership degrees of a fuzzy concept)  $x_1, x_2$  and the weights  $w_1, w_2$  over the unit interval  $[0, 1]$ .

➤ **Definition-1 :** A neural net with crisp signals and weights and crisp transfer function forms a hybrid neural net. However,

- It can combine  $x_i$  and  $w_i$  using a t-norm, t-conorm, or some other continuous operation,
- It can aggregate  $p_1$  and  $p_2$  with a t-norm, t-conorm, or any other continuous function
- The  $f$  can be any continuous function from input to output

It is emphasized that all inputs, outputs and the weights of a hybrid neural net are real numbers taken from the unit interval  $[0, 1]$ . A processing element of a hybrid neural net is called fuzzy neuron. In this chapter the following some fuzzy neurons are presented.

➤ **Definition-2:** (AND fuzzy neuron)

The signal  $x_i$  and  $w_i$  are combined by a triangular conorm  $S$  to produce  $p_i$ , It is given by Equation (3.32).

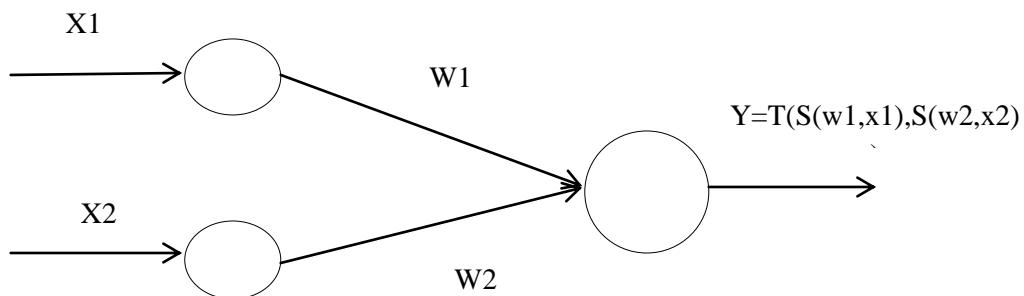
$$p_i = S(w_i, x_i), \quad i = 1, 2 \quad (3.32)$$

Figure 3.14 shows AND fuzzy neuron. The input information  $p_i$  is aggregated by a triangular norm  $T$  to produce the output of the neuron it is given by Equation (3.33).

$$y = \text{AND}(p_1, p_2) = T(p_1, p_2) = T(S(w_1, x_1), S(w_2, x_2)) \quad (3.33)$$

So, if  $T = \min$  and  $S = \max$  then the AND neuron realizes the min-max composition .It is given by Equation (3.34)

$$y = \min \{w_1 \vee x_1, w_2 \vee x_2\}. \quad (3.34)$$



**Figure 3.14 AND fuzzy neuron**

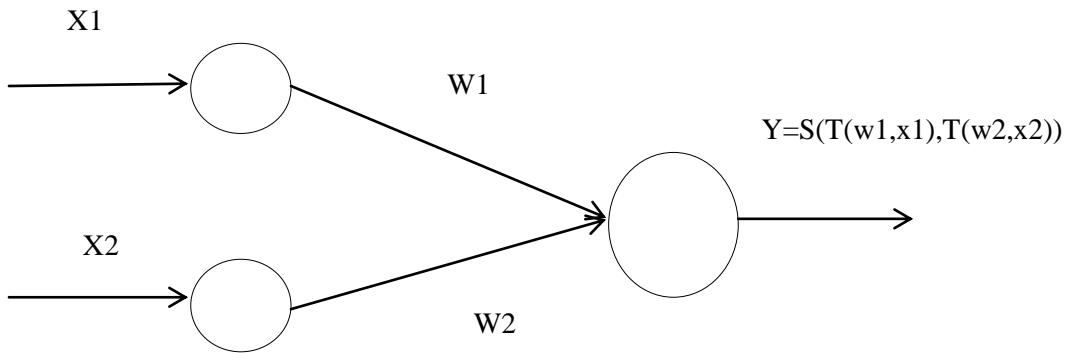
- **Definition-3:** (OR fuzzy neuron)

The signal  $x_i$  and  $w_i$  are combined by a triangular norm  $T$  to produce  $p_i$  is given by Equation (3.35).

$$p_i = T(w_i, x_i), i = 1, 2. \quad (3.35)$$

The input information  $p_i$  is aggregated by a triangular conorm  $S$  to produce the output of the neuron  $y$ .it is given by Equation (3.36).

$$y = \text{OR}(p_1, p_2) = S(p_1, p_2) = S(T(w_1, x_1), T(w_2, x_2)) \quad (3.36)$$



**Figure 3.15 OR fuzzy neuron**

So, if  $T = \min$  and  $S = \max$  then the AND neuron realizes the max-min composition is given by Equation (3.37).

$$y = \max \{w_1 \wedge x_1, w_2 \wedge x_2\} \quad (3.37)$$

The AND and OR fuzzy neurons realize basic logic operations on the membership values. The role of the connections is to differentiate between levels of impact that the inputs individually might have on the result of aggregation. It is noted that (i) the higher the value  $w_i$  the stronger the impact of  $x_i$  on the output  $y$  of an OR neuron, (ii) the lower the value  $w_i$  the stronger the impact of  $x_i$  on the output  $y$  of an AND neuron.

The range of the output value  $y$  for the AND neuron is computed by letting all  $x_i$  equal to zero or one. In virtue of the monotonicity property of triangular norms, to obtain  $y \in [T(w_1, w_2), 1]$  and for the OR neuron one derives boundaries  $y \in [0, S(w_1, w_2)]$ . Figure 3.15 shows OR fuzzy Neuron.

**Definition-4:** (Implication-OR fuzzy neuron)

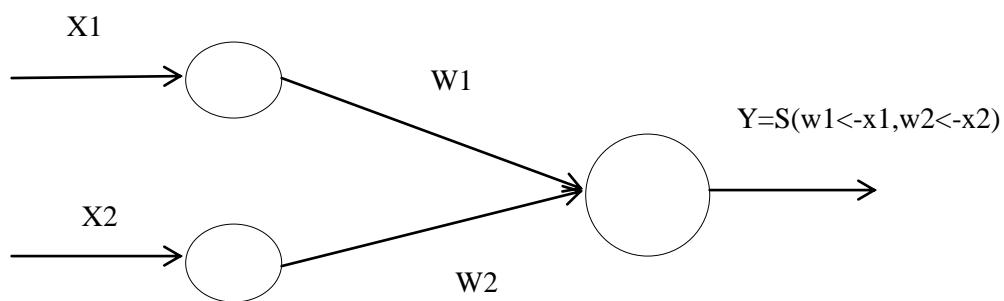
Figure 3.16 shows the implication-OR fuzzy neyron. The signal  $x_i$  and  $w_i$  are combined by a fuzzy implication operator  $I$  to produce  $p_i$  it is

given by Equation (3.38).

$$p_i = I(w_i, x_i) = w_i \leftarrow x_i, i = 1, 2. \quad (3.38)$$

The input information  $p_i$  is aggregated by a triangular conorm  $S$  to produce the output of the neuron the out put  $y$  is given in Equation (3.39).

$$y = I(p_1, p_2) = S(p_1, p_2) = S(w_1 \leftarrow x_1, w_2 \leftarrow x_2) \quad (3.39)$$



**Figure 3.16 Implication-OR fuzzy neuron**

- **Definition-5:** (Kwan and Cai's fuzzy neuron)

The signal  $x_i$  interacts with the weight  $w_i$  to produce the product. It is given in Equation (3.40).

$$p_i = w_i x_i, i = 1, \dots, n \quad (3.40)$$

The input information  $p_i$  is aggregated by an aggregation function  $h$  to produce the input of the neuron is given by Equation (3.41).

$$z = h(w_1 x_1, w_2 x_2, \dots, w_n x_n) \quad (3.41)$$

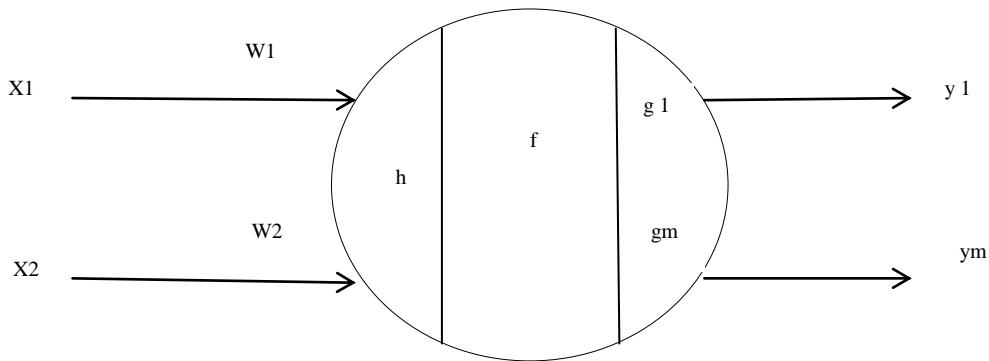
the state of the neuron is computed by  $s$ .it is given in Equation (3.42).

$$s = f(z - \theta) \quad (3.42)$$

where  $f$  is an activation function and  $\theta$  is the activating threshold. And the  $m$  outputs of the neuron are computed by  $y_j$ . It is given in Equation (3.43)

$$y_j = g_j(s), \quad j = 1, \dots, m \quad (3.43)$$

where  $g_j, j = 1, \dots, m$  are the  $m$  output functions of the neuron which represent the membership functions of the input pattern  $x_1, x_2, \dots, x_n$  in all the  $m$  fuzzy sets. Figure 3.17 shows Kwan and Cai's fuzzy neuron



**Figure 3.17 Kwan and Cai's fuzzy neuron**

➤ **Definition-6:** (Kwan and Cai's max fuzzy neuron)

Figure 3.18 shows Kwan and Cai's max fuzzy neurons. The signal  $x_i$  interacts with the weight  $w_i$  to produce the product  $p_i$ . It is given in Equation (3.44).

$$p_i = w_i x_i, \quad i = 1, 2. \quad (3.44)$$

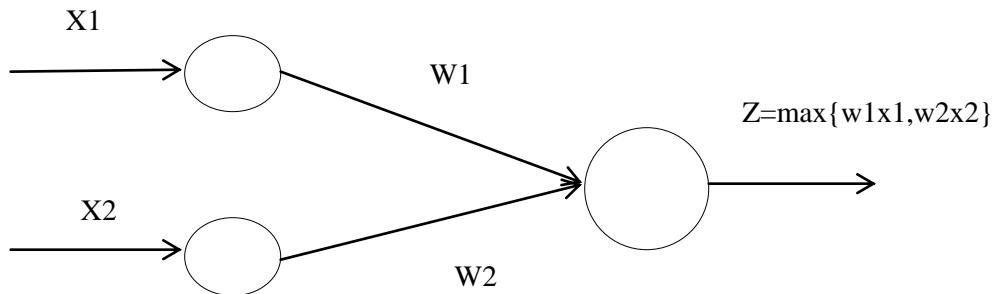
The input information  $p_i$  is aggregated by the maximum conorm is given in Equation (3.45).

$$z = \max\{p_1, p_2\} = \max\{x_1 w_1, x_2 w_2\} \quad (3.45)$$

and the  $j$ -th output of the neuron is computed by Equation (3.46).

$$y_j = g_j(f(z - \theta)) = g_j(f(\max\{w_1 x_1, w_2 x_2\} - \theta)) \quad (3.46)$$

where  $f$  is an activation function.



**Figure 3.18 Kwan and Cai's max fuzzy neurons**

- **Definition-7:** (Kwan and Cai's min fuzzy neurons)

The signal  $x_i$  interacts with the weight  $w_i$  to produce the product is given in Equation (3.47).

$$p_i = w_i x_i, i = 1, 2. \quad (3.47)$$

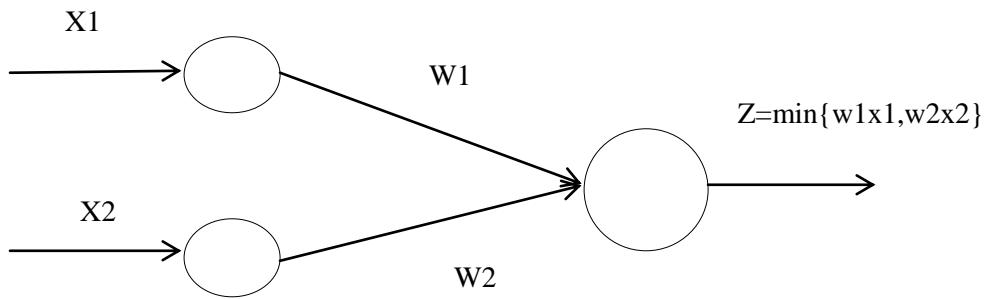
The input information  $p_i$  is aggregated by the minimum norm  $y$  is given in Equation (3.48)

$$y = \min\{p_1, p_2\} = \min\{w_1 x_1, w_2 x_2\} \quad (3.48)$$

and the  $j$ -th output of the neuron is computed by Equation (3.49).

$$y_j = g_j(f(z - \theta)) = g_j(f(\min\{w_1 x_1, w_2 x_2\} - \theta)) \quad (3.49)$$

where  $f$  is an activation function.



**Figure 3.19 Kwan and Cai's min fuzzy neuron**

It is well-known that regular nets are nothing but approximators, i.e. they can approximate any continuous function on a compact set to arbitrary accuracy. In a discrete fuzzy expert system one inputs a discrete approximation to the fuzzy sets and obtains a discrete approximation to the output fuzzy set. Usually discrete fuzzy expert systems and fuzzy controllers are continuous mappings. Conclude that given a continuous fuzzy expert system, or continuous fuzzy controller, there is a regular net that can uniformly approximate it to any degree of accuracy on compact sets. Figure 3.19 shows Kwan and Cai's min fuzzy neuron.

Though hybrid neural nets cannot use directly the standard error back propagation algorithm for learning, they can be trained by steepest descent methods to learn the parameters of the membership functions representing the linguistic terms in the rules.

The direct fuzzification of conventional neural networks is to extend connection weights and/or inputs and/or fuzzy desired outputs (or targets) to fuzzy numbers.

### 3.12 RESULTS AND DISCUSSION

In this Neuro fuzzy filter method three artifacts are (EOG, EMG, and ECG) to the EEG signal. The artifacts are removed using two methods in parallel manner. The two methods are adaptive filtering and neuro-fuzzy

filter. The denoised signals from both the methods are cross correlated using a few parameters such as SNR and PSD. A few suggestions are made for the real-time removal of Artifacts using adaptive filtering. Neuro-fuzzy approaches are very promising for non-linear filtering of noisy images. It has not been proved that they are applicable for signals and so in this method the noise is removed from the EEG signal by both adaptive and neuro-fuzzy filtering method and the performance of them are noted and compared. In this method, the primary input is the measured EEG and the reference input is the artifacts signal.

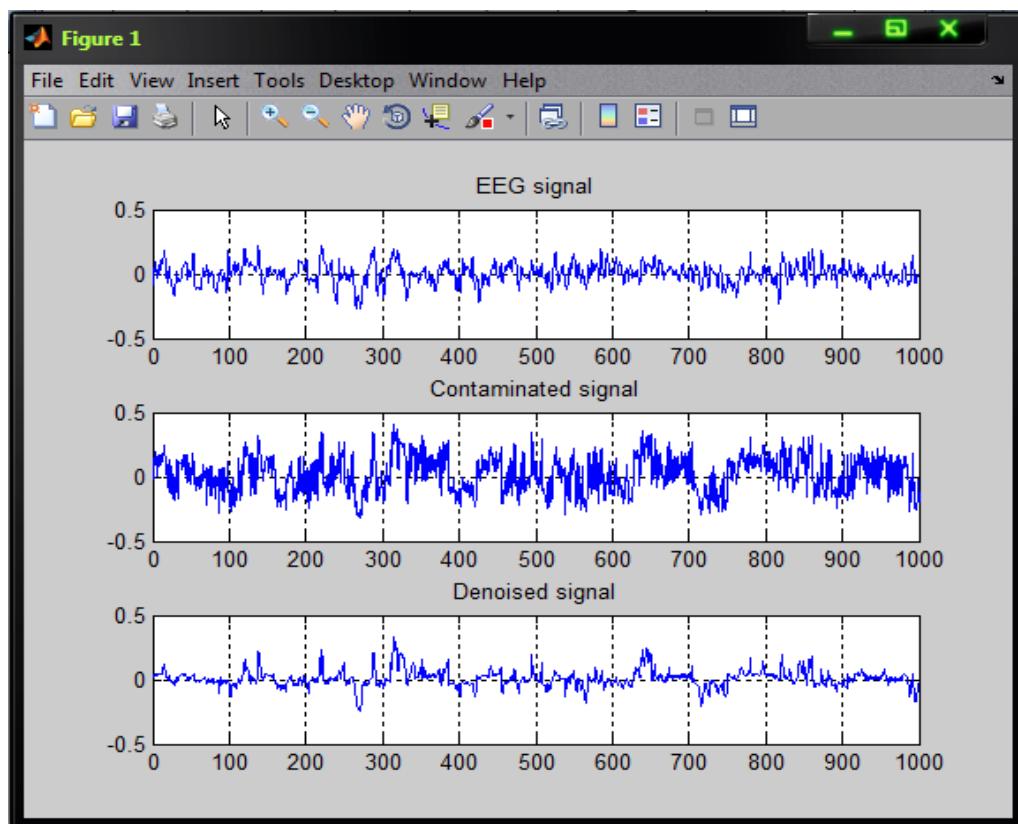
Least Mean Square algorithm is used to implement Adaptive filter. A difficulty found in this work was the determination of  $L$  (filter order) and  $\mu$  (convergence factor). These parameters are very important;  $L$ , because it leads to appropriate filtering, and  $\mu$ , to get adequate adaptation. If  $\mu$  is too big, the filter becomes unstable, and if it is too small, the adaptation may turn out too slow. Several tests were carried out to determine the optimum value for these parameters.

These disadvantages are not available in the proposed Neuro-Fuzzy filter. Neuro-fuzzy structure is specifically designed to combine even different processing strategies for data highly corrupted by one or more noise distributions. The adoption of a sophisticated detail preserving mechanism makes it possible to perform a very effective noise cancellation. Such a mechanism adjusts the output correction depending on the uncertainty occurring in the noise detection process. K means algorithm was found to be more effective than Least Mean Square algorithm.

### 3.12.1 Artifact removal using Adaptive filter

In this section, the results of EEG signal with five Trials of artifact removals using Adaptive filter is discussed. The following results were obtained using MATLAB software.

Let the real EEG signal and the mixed artifacts be considered (EOG+EMG+ECG). Adaptive filtering is performed till the EEG signal is free from the artifacts. Figure 3.20 shows (a) original EEG signal (b) noised signal (c) denoised signal.



**Figure 3.20 Result of Adaptive filter**

### **3.12.2 SNR value resulted in Adaptive filtering**

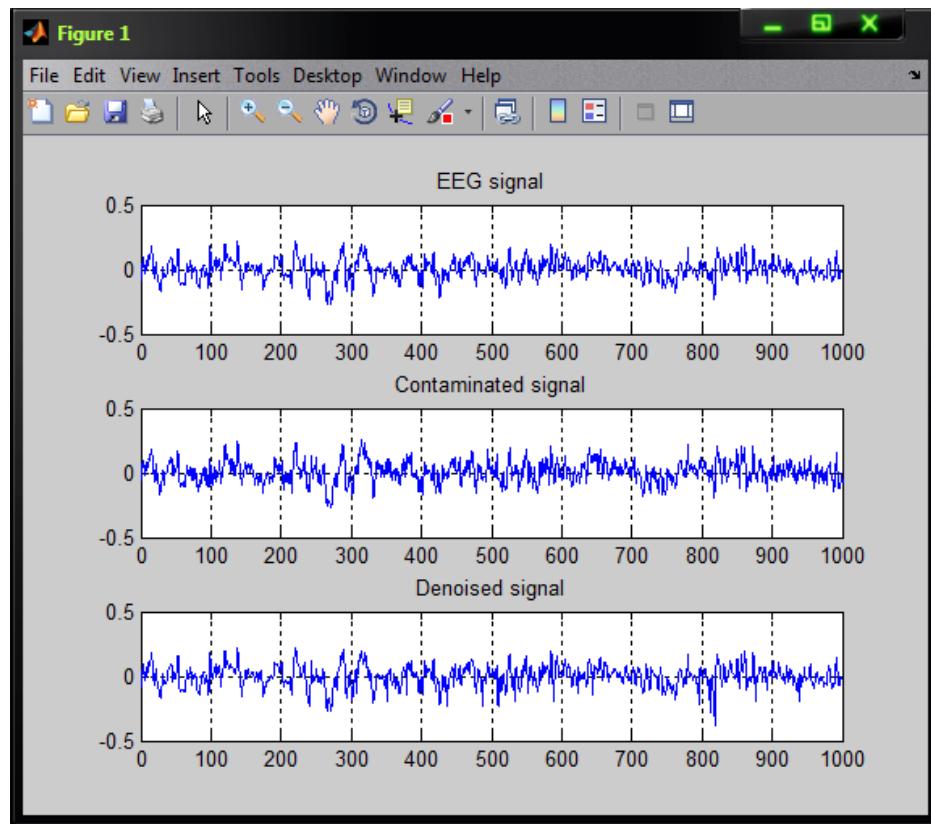
Table 3.1 Summarizes the Signal to Noise ratio of corrupted signal (EEG+ Artifact) and Denoised Signal (Corrected EEG) using Adaptive filter (Paulchamy Balaiah and Ilavennila, 2012). The table clearly shows that the Signal to Noise Ratio of denoised signal is higher than the corrupted Signal

**Table 3.1 SNR value of Adaptive filter**

<b>Trials</b>	<b>SNR for Corrupted Signal</b>	<b>SNR for De-noised Signal</b>
Trial -1	10.1040	12.5020
Trial -2	11.1020	13.1050
Trial -3	12.3010	14.7030
Trial -4	12.9210	14.9010
Trial-5	13.1020	15.1090

### **3.12.3 Artifacts removal using Neuro-Fuzzy filter**

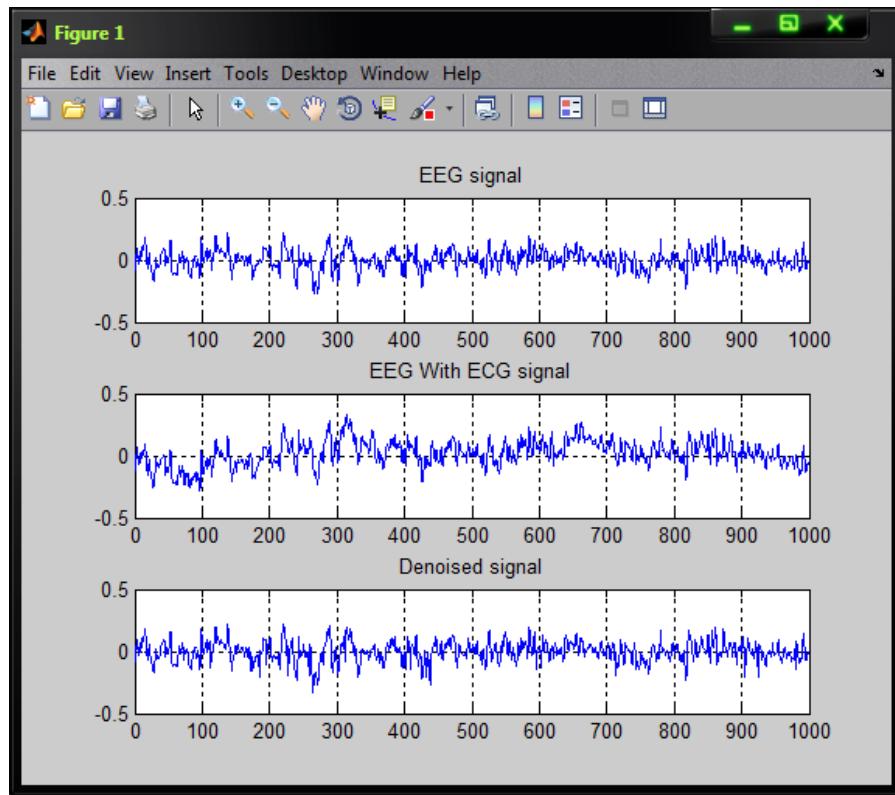
Let the real EEG signal and the mixed artifacts be considered (EOG+EMG+ECG). Neuro fuzzy filtering is performed till the EEG signal is free from the artifacts. The Figure 3.21 shows (a) original EEG signal (b) noised signal (c) denoised signal.



**Figure 3.21 Result of Neuro-Fuzzy filters**

#### **3.12.4 ECG Artifacts removal using Neuro-Fuzzy filter**

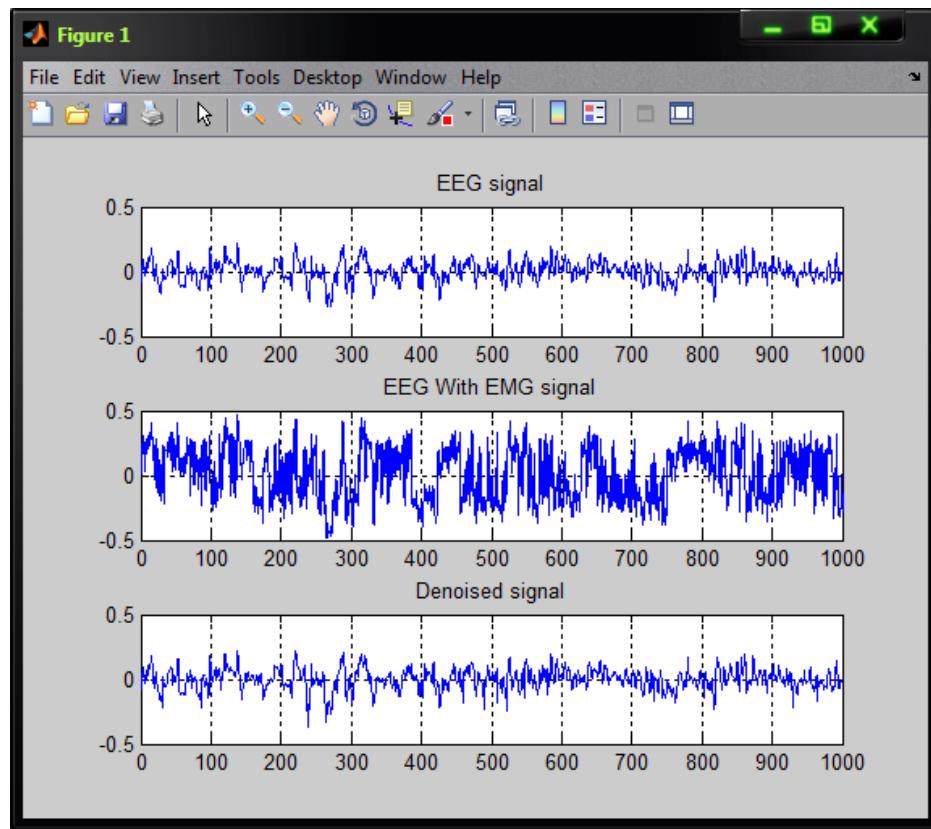
One may consider the real EEG signal and the mixed artifacts (ECG). Neuro fuzzy filtering is performed till the EEG signal is free from the artifacts. The Figure 3.22 shows (a) original EEG signal (b) noised signal (c) denoised signal.



**Figure 3.22 Result of Neuro-Fuzzy filters (EEG+ECG)**

### 3.12.5 EMG Artifacts removal using Neuro-Fuzzy filter

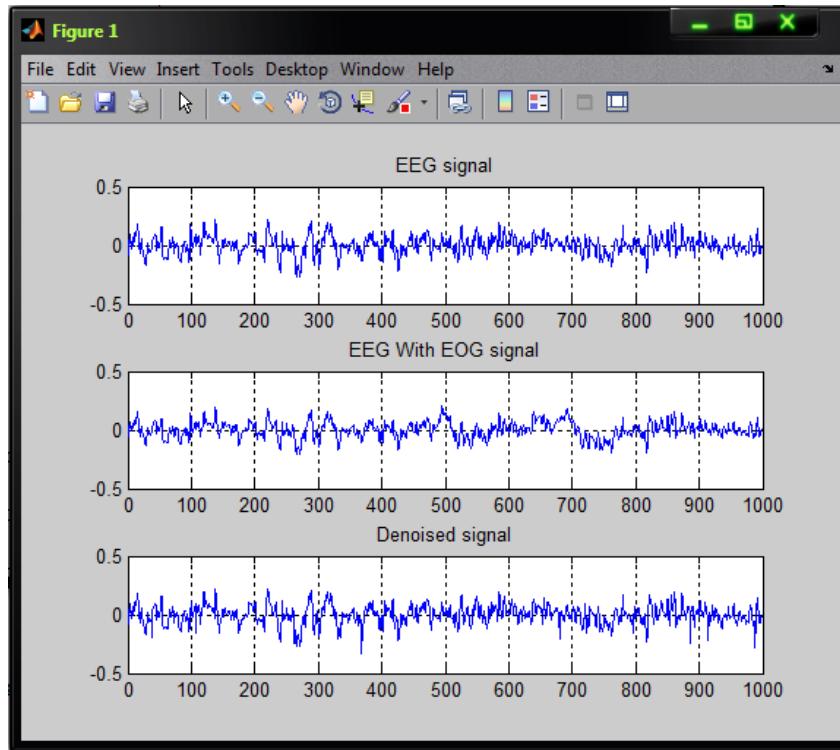
One may consider the real EEG signal and the mixed artifacts (EMG). Neuro fuzzy filtering is performed till the EEG signal is free from the artifacts. The Figure 3.23 shows (a) original EEG signal (b) noised signal (c) denoised signal.



**Figure 3.23 Result of Neuro-Fuzzy filters (EEG+EMG)**

### 3.12.6 EOG Artifacts removal using Neuro-Fuzzy filter

Let the real EEG signal and the mixed artifacts be considered (EOG). Neuro fuzzy filtering is performed till the EEG signal is free from the artifacts. The Figure 3.24 shows (a) original EEG signal (b) noised signal (c) denoised signal.



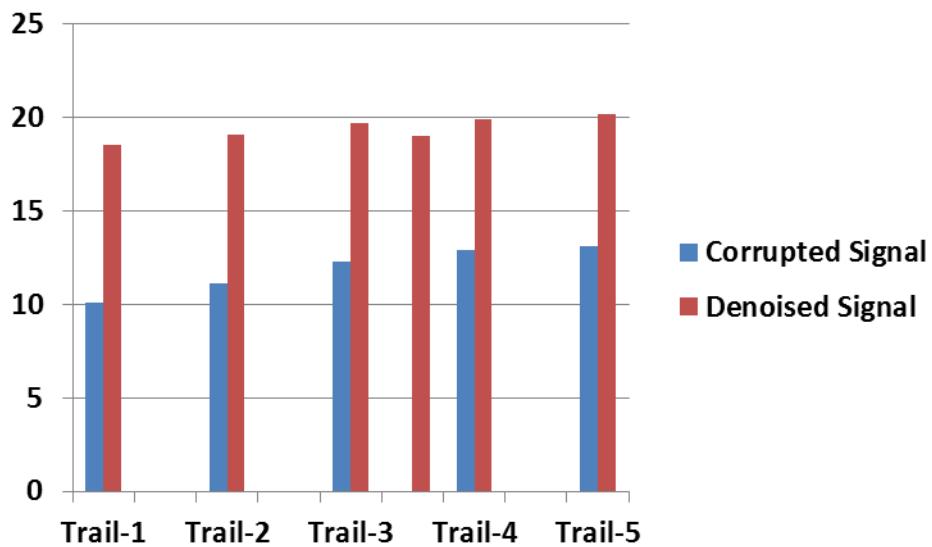
**Figure 3.24 Result of Neuro-Fuzzy filters (EEG+EOG)**

### 3.12.7 SNR value resulted in Neuro-Fuzzy filtering

Table 3.2: Summarizes the Signal to Noise ratio comparison of 10 Trials of Noisy signal (EEG+ECG+EMG+EOG Artifact) and Denoised Signal (Corrected EEG) using Neuro-Fuzzy filter. Figure 3.25 shows the SNR Curve for neuro fuzzy filtering

**Table 3.2 SNR value of Neuro-Fuzzy filters**

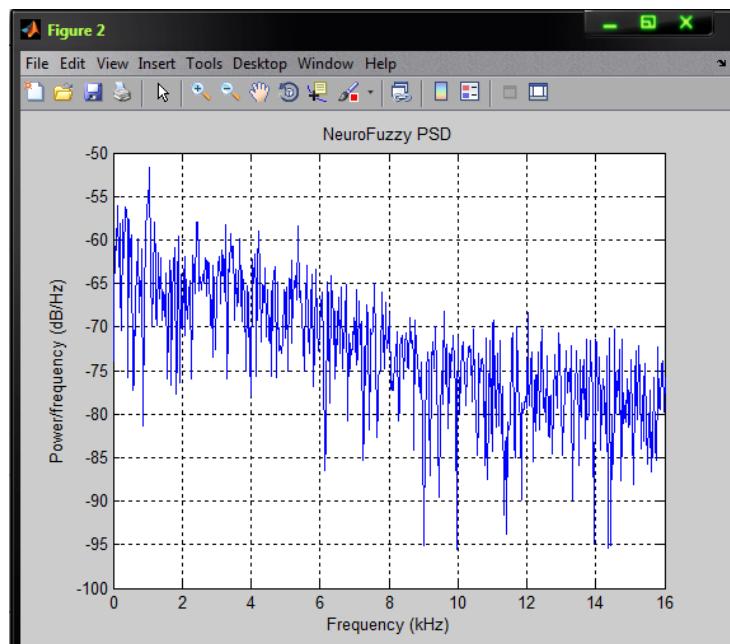
<b>Trials</b>	<b>SNR for Corrupted Signal</b>	<b>SNR for De-noised Signal</b>
Trial -1	10.1040	18.5040
Trial -2	11.1020	19.1020
Trial -3	12.3010	19.7030
Trial -4	12.9210	19.9040
Trial -5	13.1020	20.2060



**Figure 3.25 SNR Curve for neuro fuzzy filtering**

### 3.12.8 Power spectral density of Denoised EEG signal

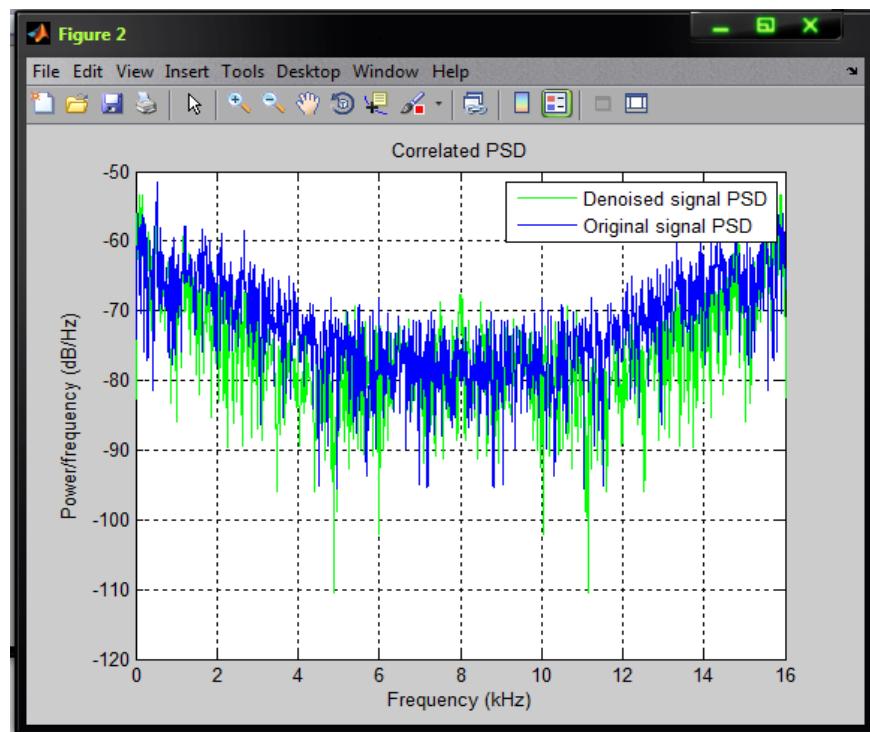
Figure 3.26 shows the Power Spectral Density of the denoised EEG signal in the adaptive filter method (Senthilkumar et al 2008). From this figure it is shown that the powers of the spectral components have been retained.



**Figure 3.26 PSD of Neuro-Fuzzy filters**

### 3.13 CORRELATION BETWEEN ORIGINAL AND ARTIFACTS REMOVED EEG SIGNAL

Figure 3.27. Shows the correlation plot for the each Trial of ORIGINAL EEG and mixed Artifacts Removed EEG in Neuro-Fuzzy filtering. This Shows how close both the signals are in terms of the Shape. (X-Axis Frequency ,Y-Axis Correlation Co-efficient)

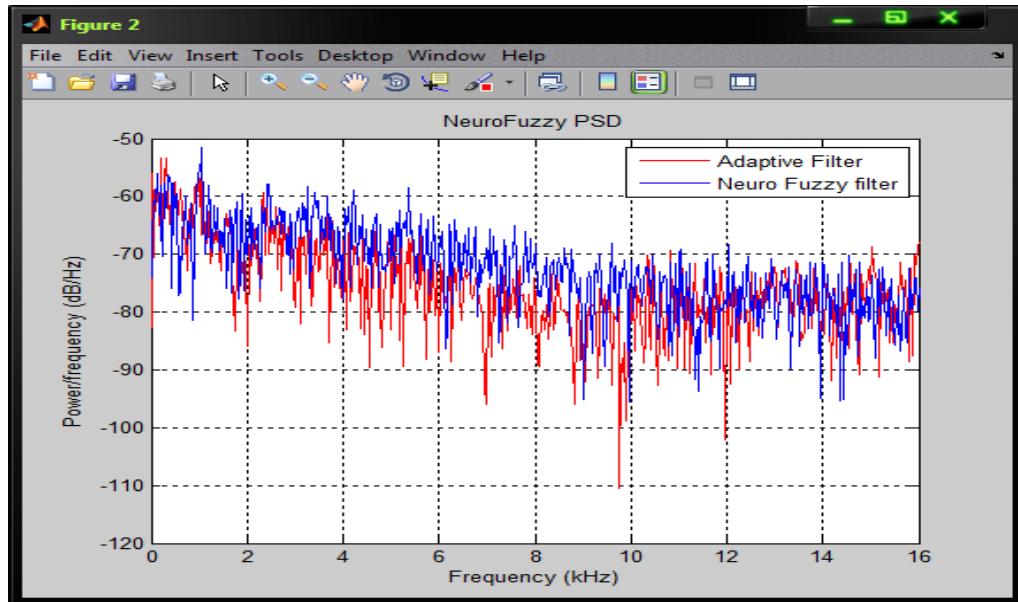


**Figure 3.27 Correlation Plot using Neuro fuzzy filter**

#### 3.13.1 Comparative study of PSD

The following plot shows the Power Spectral Density of the denoised EEG signal in both adaptive and neuro-fuzzy filter methods.

Figure 3.28. Shows the Power Spectra of the denoised EEG in both adaptive and neuro-fuzzy filter methods. From this figure it is shown that the powers of the spectral components have been retained.



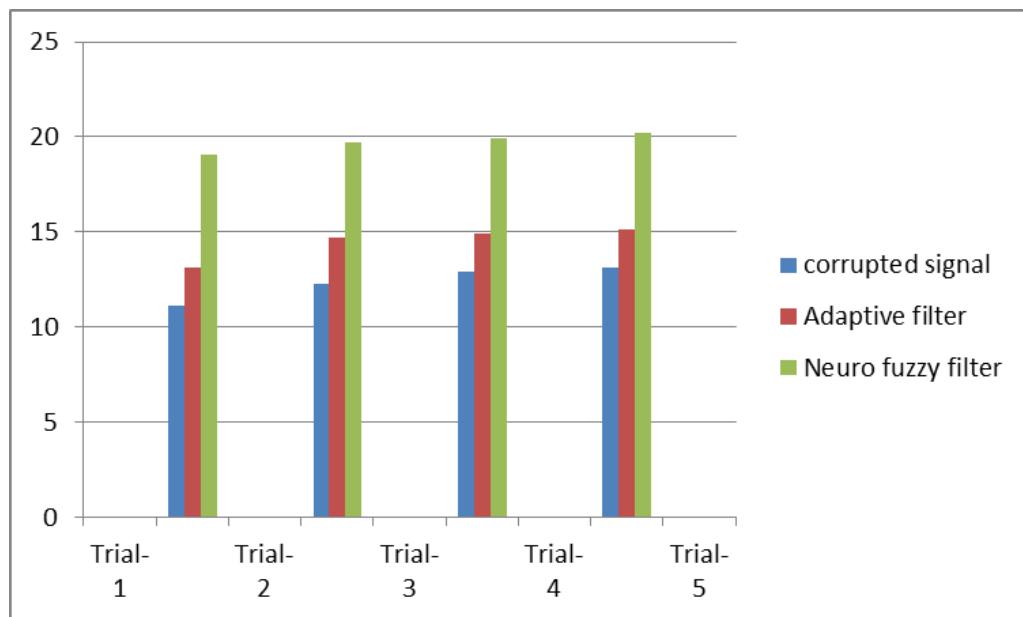
**Figure 3.28 Power Spectral density plot**

### 3.13.2 Correlative SNR value

Table 3.3: Summarizes the Signal to Noise ratio of Denoised Signal (Corrected EEG) using Adaptive filter and Neuro-Fuzzy filter. Concluding, that the Signal to Noise Ratio of denoised signal using Neuro-Fuzzy filter is higher than the denoised signal using Adaptive filter.

**Table 3.3 SNR Value Comparison between Adaptive and Neuro fuzzy filter**

<b>Trial s</b>	<b>SNR for Corrupted Signal</b>	<b>SNR for De-noised Signal</b>	
		<b>Adaptive Filter</b>	<b>Neuro fuzzy filter</b>
Trial-1	10.1040	12.5020	18.5040
Trial-2	11.1020	13.1050	19.1020
Trial-3	12.3010	14.7030	19.7030
Trial-4	12.9210	14.9010	19.9040
Trial-5	13.1020	15.1090	20.2060



**Figure 3.29 SNR curve between adaptive and neuro fuzzy filtering**

Figure 3.29 shows the signal to noise ratio curve for adaptive filter and neuro fuzzy filter along with corrupted signal. This curve clearly shows that the neuro fuzzy filter proves better result than adaptive filter.

### 3.14 SUMMARY

In chapter 3 a detailed explanation of adaptive filtering method is seen that provides details of adaptive filter that have the capability of modifying their properties according to selected features of the signals being analyzed and its structure, the procedure in removing artifacts, its advantages and disadvantages. The statistical result has also been shown that ECG and EOG components were attenuated in smaller proportion. By the proposed technique called Neuro-fuzzy filtering method, it is discussed in detail. Since Neuro-fuzzy refers to the combination of artificial neural network which are composed of artificial neurons or nodes and fuzzy logic that are often regarded as concepts which in their application are neither completely true or completely false, or which are partly true and partly false are discussed in brief. Also, the structure and methodology of Neuro-fuzzy filter in removal of

artifacts are discussed. Ultimately the results of EEG signal with several artifact removal using Adaptive filter and Neuro-Fuzzy filter are discussed. In the proposed method EEG is subjected to noise signal and it is contaminated. Then the noise is removed by means of Adaptive filter and Neuro-fuzzy filter. The SNR ratio for both noised and denoised signal is calculated and it is observed that the SNR of the denoised signal is higher than the noised one. Also the power spectral density of the denoised signal is plotted. From the observations the performance of both adaptive filter and the Neuro-fuzzy filter is noted. By the comparative study conclude that, the performance of Neuro-Fuzzy filter is better than the Adaptive filter. The fidelity of the reconstructed EEG signal is assessed quantitatively using parameters such as SNR (Signal to Noise Ratio) and PSD (Power Spectral Density).

## **CHAPTER 4**

### **ARTIFACTS REMOVAL USING WAVELET TRANSFORM**

#### **4.1 INTRODUCTION**

This chapter shows a new method to remove the eye-movement artifact from the Electro-encephalogram (EEG) which is based on Haar transform and frequency analysis. Transfer of eye-movement activity to EEG can have frequency dependent amplitude and phase characteristics. The proposed method is suitable for handling such transfer because the threshold formula is used in the frequency domain. The method is demonstrated with artificial signal-in-noise EOG (Electro-oculogram) and EEG series. In the EEG noise an event related potential (ERP) is buried as a constant signal and in the EOG noise a changing EOG response (saccadic eye-movement) is simulated before adding the whole series to the EEG series.

The decreasing levels of transfer from EOG on EEG leads are also simulated as the EOG artifact diminishes from the frontal to the occipital area. Owing to this possible frequency dependent phase characteristics a time-shift of the EOG is also simulated. This chapter discusses a method to automatically identify slow varying Ocular Artifact zones applying wavelet based adaptive thresholding algorithm only to the identified Ocular Artifacts zones, which avoids the removal of background EEG information. The adaptive thresholding applied only to the Ocular Artifacts zone does not affect the low frequency components in the non-Ocular Artifacts zones but

preserves the shape (waveform) of the EEG signal in non-artifact zones which is of great importance in clinical diagnosis. The subtraction formula, corrects accurately the influence of the EOG artifact on EEG activity.

## **4.2 EXTRACTING KNOWLEDGE FROM EXISTING METHODS**

### **4.2.1 Artifact Removal**

Artifact removal is the process of identifying and removing artifacts from brain signals. An artifact removal method should be able to remove the artifacts as well as keep the related neurological phenomenon intact.

Common methods for removing the artifacts in EEG signals are shown below:

- Linear filtering
- Linear combination and regression
- Blind source separation, principle component analysis
- Wavelet transform
- Nonlinear adaptive filtering and source dipole analysis (SDA)

### **4.2.2 Subtraction Method**

Subtraction methods are based on the assumption that the measured EEG is a linear combination of an original EEG and a signal caused by any sort of body movement E.g. Eye movement, called EOG (electrooculogram) is a potential produced by movement of the eye or eyelid. The original EEG is hence, recovered by subtracting separately the recorded EOG from the

measured EEG using appropriate weights .This method proves good even for the other types of artifacts such as EMG, ECG and so on.

#### **4.2.3 Linear Combination and Regression Method**

Using a linear combination of the EOG-contaminated EEG signal and the EOG signal is the most common technique for removing ocular artifacts from EEG signals. The linear combination technique is based on the following Equation (4.1).

$$EEG_{rec}(t) = EEG_{true}(t) + s.EOF(K) \quad (4.1)$$

where,  $EEG_{rec}(t)$  - Recorded EEG which is contaminated signal and holds artifacts

$EEG_{true}(t)$  - EEG due to the cortical activity (i.e., Brain activity)

$s.EOF(K)$  - Propagated ocular artifact due to eye blinks and movements having impact over the recording site.s is an unknown constant.

A popular method that aims at minimizing theeffect of noise on the estimates employs linear regression using least square criterion to estimate the value ofK.A question arises as to whether the value of K should be calculated separately for each type of EOGartifact and for the different frequencies of a particular EOG artifact (Carrie AJoyce et al 2009) .One problem with using the above linearcombination and regression approach is that the EOG signal to be subtracted from the EEG signal. However,subtracting the EOG signal may also remove part of the EEG signal. EMG artifacts do not have any referencechannels, and applying regression using signals from multiple muscle groups requires multiple referencechannels. Regression techniques for the removal of head-movement artifacts, and jaw clenching, spit swallowingcan be applied.

#### **4.2.4 Principal Component Analysis**

Multi-channel EEG recordings can be expressed by a P(time points) x N(channels) matrix, E, anddecomposed as a product of three matrixes, $E=USVT$ , where U is an  $P \times N$  matrix such that  $UTU = I$ , S is a  $N \times N$  diagonal matrix, and V is an  $N \times N$  matrix such that  $VTV=VVT=I$ .If E is an EEG epoch of N channels and time points, U contains its N normalized Principal Components that are decor related linearly and can be re-mixed to reconstruct the original EEG. PCA uses the eigenvectors of the covariance matrix of the signal to transform the data into a new coordinate system and to find the projection of the input data with greater variances. The components of the signal are then extracted by projecting the signal onto the eigenvectors. PCA has been shown to be an effective method for removing ocular artifacts from EEG signals. A disadvantage of PCA is that artifacts are uncorrelated with the EEG signal (Lagerlund et al2009). This is a stronger requirement than the independency requirement of ICA. It has been observed that PCA cannot completely separate eye-movement artifacts, EMG and ECG artifacts from the EEG signal, especially when they have comparable amplitudes. Besides ,PCA does not necessarily decompose similar EEG features into the same components applied to different epochs.

#### **4.2.5 Canonical Correlation Analysis**

The Canonical Correlation Analysis (CCA) is developed to overcome the disadvantages of ICA. CCA is used as a Blind Source Separation technique (BSS) for artifacts removal from EEG signal. CCA based BSS method utilizes the temporal auto correlation in the source signal as a contrast function. It measures the linear relationship between two multi-dimensional variables, by finding two bases which are optimal with respect to correlation. CCA method has a considerable amount of spectral error and thus it cannot be implemented in the real time.

#### **4.2.6      Blind Source Separation (BSS)**

BSS techniques separate the EEG signals into components that “build” the EEG signals. They identify the components that are attributed to artifacts and reconstruct the EEG signal without these components. Among the BSS methods, Independent Component Analysis (ICA) is most widely used. ICA is a method that blindly separates mixtures of independent source signals, forcing the components to be independent and widely applied to remove ocular artifacts from EEG signals. Preliminary studies have shown that ICA increases the strength of the motor-related signal components in the Mu rhythms, and is thus useful for removing artifacts in BCI systems. BSS methods have been used to remove EOG, EMG and ECG artifacts in EEG clinical studies; an advantage of using BSS methods such as ICA is that they do not rely on the availability of reference artifacts for separating the artifacts from the EOG signals. A disadvantage of ICA, along with other BSS techniques, is that they usually need prior visual inspection to identify artifact components.

#### **4.2.7      Linear filtering**

Linear filtering is useful for removing artifacts located in certain frequency bands that do not overlap with those of the neurological phenomena of interest. For example, low-pass filtering can be used to remove EMG artifacts and high-pass filtering can be used to remove EOG artifacts. Linear filtering was commonly used in the early clinical studies to remove artifacts in EEG signals.

The advantage of using filtering is its simplicity. Also the information from the EOG signal is not needed to remove the artifacts. This method, however, fails when the neurological phenomenon of interest and the EMG, ECG or EOG artifacts overlap or lie in the same frequency band. As a

result, a simple filtering approach cannot remove EMG or EOG artifacts without removing a portion of the neurological phenomenon. More specifically, since EOG artifacts generally consist of low-frequency components, using a high-pass filter will remove most of the artifacts and for EMG artifacts, using a low pass filter will remove some artifacts.

### **4.3 ARTIFACTS REMOVAL FROM EEG USING HAAR WAVELET TRANSFORM**

Wavelet transforms are signal-processing algorithms similar to Fourier transforms that are used to convert complex signals from time to frequency domains. However, unlike Fourier transforms, wavelets are able to functionally localize a signal in both the time and frequency space, thus allowing the transformed data to be simultaneously analyzed in both the domains (frequency and time). The wavelet transform of the noisy signal generates the wavelet coefficients which denote the correlation coefficients between the noisy EEG and the wavelet function. Depending on the choice of the mother wavelet function (which may resemble the noise component), the larger coefficients will be generated corresponding to the noise affected zones. Ironically smaller coefficients will be generated in the areas corresponding to the actual EEG.

The larger coefficients will now be an estimate of noise. Appropriate threshold limit is to be found which separates the noise coefficients and the signal coefficients. A proper thresholding function is to be chosen to discard the noise coefficients appropriately. The thresholding functions decide which coefficients should be retained and what should be done to them. Hence, discarded coefficients would result in the removal of noise, and the retained coefficients represent the wavelet coefficients of the de-noised EEG signal. On taking the inverse wavelet transform, the de-noised signal is obtained and so the selection of the threshold and thresholding

function plays a crucial role in EEG de-noising(Croft and Barry2009). Applied wavelet based adaptive thresholding algorithm only to identify ocular artifact zones, which avoids the removal of background EEG information.

### **4.3.1 Wavelet Transform**

The Wavelet transform is an emerging signal processing technique that can be used to represent the real-time non stationary signals with high efficiency. Indeed, the wavelet transform is gaining momentum to become an alternative tool to traditional time-frequency representation techniques like the Discrete Fourier Transform and the Discrete Cosine Transform. By virtue of its multi-resolution representation capability, the Wavelet transform has been used effectively in vital applications such as transient signal analysis, numerical analysis, computer vision, image compression, among many other audio visual applications. Wavelet transform needs to be embedded in consumer electronics, and thus a single chip hardware implementation is more desirable than a multi-chip parallel system implementation.

The wavelet transform is a transform similar to the Fourier transform (or much more to the windowed Fourier transform) with a completely different merit function. The main difference is this: Fourier transform decomposes the signal into sines and cosines, i.e. the functions localized in Fourier space; in contrary the wavelet transform uses functions that are localized in both the real and Fourier space. Generally, the wavelet transform can be expressed by the following Equation (4.2).

$$F(a, b) = \int_{-\infty}^{\infty} f(x) \psi^{*(a, b)}(x) dx \quad (4.2)$$

where the  $*$  is the complex conjugate symbol and function  $\psi$  is some function. This function can be chosen arbitrarily provided that it obeys certain rules.

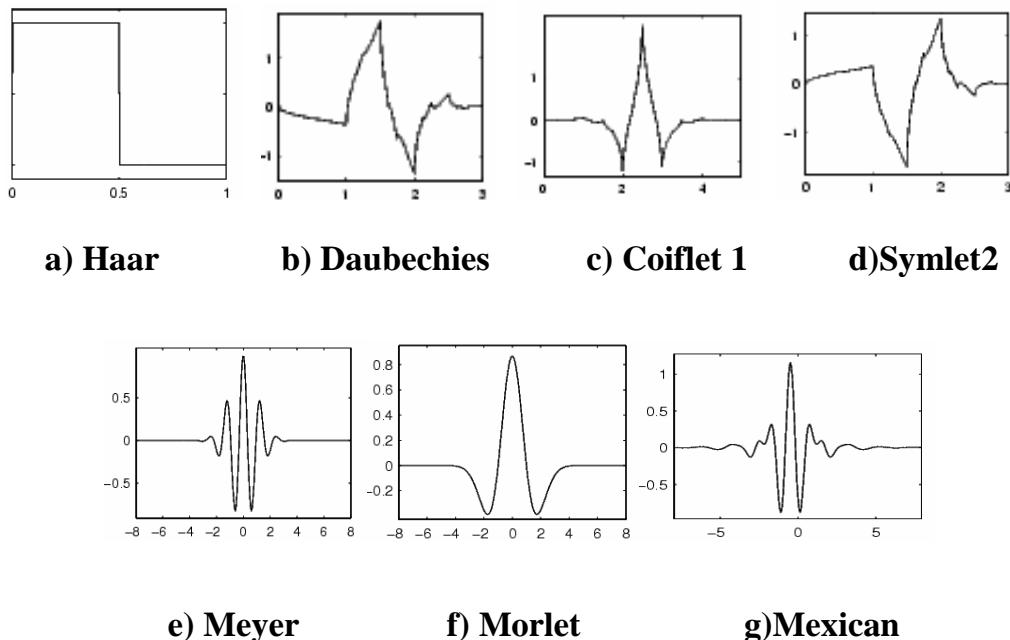
As seen, the wavelet transform is infact an infinite set of various transforms, depending on the merit function used for its computation. This is the main reason, why the term “wavelets transform” is used in different situations and applications. There are also many ways to sort out the types of the wavelet transforms(Krishnaveni et al 2011). Here only the divisions based on wavelet orthogonally is shown. One can use orthogonal wavelets for discrete wavelet transform development and non-orthogonal wavelets for continuous wavelet transform development. These two transforms have the following properties:

1. The discrete wavelet transform returns a data vector of the same length as the input is. Usually, even in this vector many data are almost zero. This corresponds to the fact that it decomposes into a set of wavelets (functions) that are orthogonal to its translations and scaling. Therefore such signals are decomposed to the same or lower number of the wavelet coefficient spectrum as it is the number of signal data points. Such a wavelet spectrum is very good for signal processing and compression, for example, no redundant information can be had here.
2. The continuous wavelet transforms in contrary returns an array one dimension larger than the input data. For an 1D data, one can obtain an image of the time-frequency plane. The signal frequencies evolution can easily be seen during the signal and can compare the spectrum with other signals spectra. As the non-orthogonal set of wavelets, is used here

data are correlated highly, and so a big redundancy is seen here. This helps to see the results in a more humane form.

### 4.3.2 Wavelet Families

There are a number of basic functions that can be used as the mother wavelet for Wavelet Transformation. Since the mother wavelet produces all wavelet functions used in the transformation through translation and scaling, it determines the characteristics of the resulting Wavelet Transform. Therefore, the details of the particular application should be taken into account and the appropriate mother wavelet should be chosen in order to use the Wavelet Transform effectively. The wavelet families are shown in Figure 4.1.



**Figure 4.1 Wavelet families**

Haar the wavelet is one of the oldest and simplest wavelet. Therefore, any discussion of wavelets starts with the Haar wavelet.

- Daubechies wavelets are the most popular wavelets. They represent the foundations of the wavelet signal processing and are used in numerous applications. These are also called Maxflat wavelets as their frequency responses the maximum flatness at frequencies 0 and  $\pi$ . This is a very desirable property in some applications.
- The Haar, Daubechies, Symlets and Coiflets are compactly supported orthogonal wavelets. These wavelets along with Meyer wavelets are capable of a perfect reconstruction.
- The Meyer, Morlet and Mexican Hat wavelets are symmetric in shape. The wavelets are chosen based on their shape and their ability to analyze the signal in a particular application.

#### **4.3.3 Haar Wavelet**

The Haar transform is the simplest of the Wavelet transforms that cross-multiplies a function against the Wavelet with various shifts and stretches, much like the Fourier transform cross-multiplies a function against a sine wave with two phases and much stretches. In discrete form, Haar wavelets are related to a mathematical operation called the Haar transform. The Haar transform serves as a prototype for all other wavelet transforms. Like all wavelet transforms, the Haar transform decomposes a discrete signal into two sub signals of half its length. One sub signal is a running average of trend; the other is a running difference or fluctuation.

#### **4.3.4 Properties**

The Haar transform has several desirable properties. They are given below:

- The Haar transform is a fast transform.
- Haar transform is real and orthogonal.
- The basis of the Haar matrix is ordered in a sequence manner.

#### **4.3.5      Advantages**

The Haar wavelet transform has a number of advantages.

- It is conceptually simple.
- It is fast.
- It is memory efficient, since it can be calculated in place without a temporary array.
- It is exactly reversible without the edge effects that create a problem with other wavelet transforms.

#### **4.3.6      Limitations**

The Haar transform also has limitations which can be a problem for some applications.

- In generating each set of averages for the next level and each set of coefficients, the Haar transform performs an average and difference on a pair of values. Then the algorithm shifts over by two values and calculates another average and difference on the next pair.
- The high frequency coefficient spectrum should reflect all high frequency changes. The Haar window is only two elements wide. If a big change takes place from an even value

to an odd value, the change will not be reflected in the high frequency coefficients.

So, Haar wavelet transform is not useful in compression and noise removal of audio signal processing.

#### **4.4 HAAR WAVELET BASED DETECTION OF CHANGE IN THE STATE OF THE EYES**

The need to continuously monitor the EOG while recording the EEG Signal and its corruption due to concentration on the part of the user so as not to move or blink his eyes forces one to device an alternate method for detecting and removing the ocular artifacts (Krishnaveni et al 2011). The EEG Signal which is picked up by non-invasive methods over the scalp of the subject is corrupted by a multitude of artifacts of which those caused by the EOG cause maximum distortion.

In this section a brief description of the effect that the Ocular Artifacts have in the amplitude and frequency spectrum of the EEG data that is recorded is discussed a novel and elegant technique is described which the sharply varying Haar Wavelets to accurately detect changes in the uses state of the eye and this is to be extended in the subsequent section to detect eye-blanks and eyeball movements.

##### **4.4.1 Amplitude dependence on the state of the eye**

It has been known for quite some time now that the Alpha Rhythm of the EEG,which is the principal resting rhythm of the brain in adults when they are awake, is directly influenced by visual stimuli. Auditory and mental arithmetic tasks with the eyes closed leads to strong alpha waves, which are suppressed when the eyes are open. This property of the EEG has been used,

ineffectively, for a long period of time to detect eye blinks and movements. The slow response of an effective de-noising technique forced researchers to study the frequency characteristics of the EEG as well.

#### **4.4.2 EEG Recorded During Change in State of the Eye**

The successful de-noising of the recorded EEG signal is directly dependent upon the precise detection of change in state of the eye from the open state to the closed state and vice-versa. For this one requires a continuous recording of the EEG signal regardless of change in the state of the eye. The immediate increase (or decrease) in the amplitude of the EEG signal when the eyes are closed (or opened) has been known to medical scientists for quite some time, but using this difference in amplitude levels to control external devices by thresholding had gone unnoticed for over 40 years and it was not until 1998 that a team of scientists at the university of technology, Sydney, Australia noticed this fact and made what is known today, as the ‘Mind switch’ but amplitudethresholding though useful for Bio-Control, cannot be used to effectively detect the eye blinks that occur quite rapidly. So the focus of research on detection and de-noising of these Ocular Artifacts in EEG, shifted from the time domain to frequency domain.

#### **4.5 DETECTION OF CHANGE IN STATE OF THE EYES: NEED FOR A HAAR WAVELET BASED APPROACH**

After the analysis of the frequency spread of the EEG data that contained the Ocular Artifacts, researchers found that the difference in frequency of the Spikes caused due to Rapid Eye Blink (REB) and the EEG signal could be used along with a simultaneous recording of the EOG to detect and remove these artifacts. But correlation of the EEG and EOG is futile, especially because of inherent corruption of the EEG data by the restraint on the user’s eye movements and blinks. The failure of accurate

detection of these artifacts by singular observation of the time or frequency domains forces one to use wavelets to study time-frequency maps.

In this research ,the Haar wavelet of higher orders is used to decompose the recorded EEG signal to detect the exact moment when the state of the eyes changes and on subsequent section to eye-blanks and movements of the eyeballs as well.

#### **4.6 PROPOSED METHODOLOGY**

The flow chart of the proposed method is shown in Figure. EEG signal and EOG signal is taken at 1x4000. It is then sampled at the length of 1x1000. The EEG recordings are contaminated by EOG signal (Woestengurg et al 2007). The EOG signal is a non-cortical activity. The eye and brain activities have physiologically separate sources, so the recorded EEG is a superposition of the true EEG and some portion of the EOG signal. It can be represented as in Equation (4.3)

$$\text{EEG}_{\text{rec}}(t) = \text{EEG}_{\text{true}}(t) + k \cdot \text{EOG}(t) \quad (4.3)$$

$\text{EEG}_{\text{rec}}(t)$  - Recorded contaminated EEG,

$\text{EEG}_{\text{true}}(t)$  - EEG due to the cortical activity (i.e., Brain activity)

$k \cdot \text{EOG}(t)$  - Propagated ocular artifact from eye to the recording site.

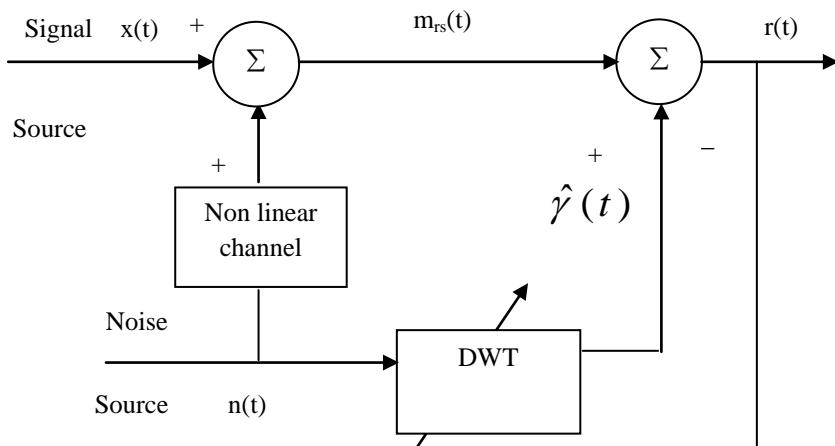
$\text{EEG}_{\text{true}}(t)$  is to be estimated from  $\text{EEG}_{\text{rec}}(t)$  by efficiently removing the  $k \cdot \text{EOG}(t)$  at the same time retaining the EEG activity.

The proposed Algorithm in this paper involves the following steps:

- i) Apply Discrete Wavelet Transform to the contaminated EEG with Haar wavelet as the basis function to detect the Ocular Artifact zone.
- ii) Apply Stationary Wavelet Transform with Coif 4 as the basis function to the contaminated EEG with OA zones identified for removing Ocular Artifacts.
- iii) For each identified OA zone, select optimal threshold limit at each level of decomposition based on minimum Risk value and apply that to the soft-like thresholding function which best removes noise.
- iv) Apply inverse stationary wavelet transform to the threshold wavelet coefficients to obtain the de-noised EEG signal.

#### 4.6.1 Adaptive Noise Cancellation Method

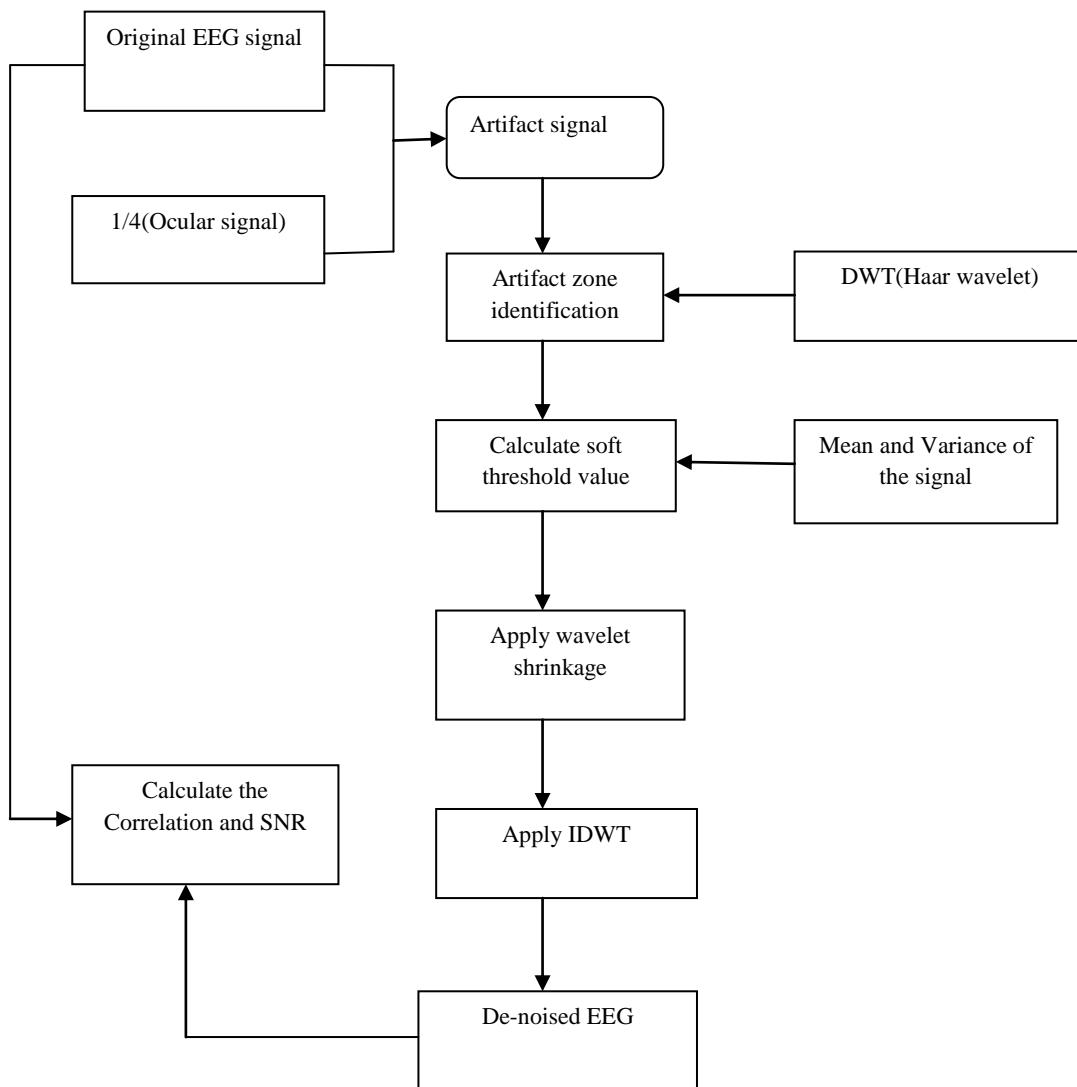
The adaptive interference cancellation is a very efficient method to solve the problem when signals and interference have overlapping spectra. Figure 4.2 shows the Adaptive Noise Cancellation Method.



**Figure 4.2 Adaptive Noise Cancellation**

- To accomplish the objective of this paper a non-linear channel with DWT is used. The input  $n(t)$  is the EEG corrupted with artifacts (EEG+EOG).
- The reference signal  $X(t)$  is on original EOG(without artifacts). The output of non-linear channel is  $r(t)$  which is an estimation of original EEG.
- This signal  $r(t)$  is subtracted from the corrupted  $m_{rs}(t)$  to produce the error  $\gamma(t)$ , which is the EEG without artifacts.

#### 4.6.2 Flowchart



**Figure 4.3 Flowchart**

## **STEP 1**

- In first step, combining original EEG signal recorded from human brain with one-third of ocular artifacts (i.e.) EOG signal taken during blink effect and eye ball movement.
- Artifact signal is obtained by combining both EEG and EOG signal.

## **STEP 2**

### **Artifact zone identification**

- There are several methods to identify the artifact zone. The artifact zone is identified using haar wavelet transform.
- Since it is a type of wavelet transform, it removes the artifact in lesser time. It is applicable to the real time as compared to other existing transforms like PCA, ICA, and Regression.
- The Haar wavelet is also the simplest possible wavelet. The technical disadvantage of the Haar wavelet is that it is not continuous, and therefore not differentiable.
- This property can, however, be an advantage for the analysis of signals with sudden transitions, such as monitoring of tool failure in machines.

## **STEP 3**

- This step involves finding the soft threshold value of noisy signal. This threshold value is calculated using mean and variance of the signal. For noisy signal the threshold value must always greater.

- Adaptive thresholding applied only to the ocular Artifacts zone does not affect the low frequency components in the non-ocular Artifacts zones and also preserves the shape (waveform) of the EEG signal in non-artifact zones which is of very much importance in clinical diagnosis.

#### **STEP 4**

- After finding the threshold value, the noisy signals are eliminated using wavelet shrinkage. This step involves the removal of the noise by wavelet shrinkage. This removes the noise using threshold coefficient value.
- Wavelet coefficients having small absolute value are considered to encode mostly noise and very fine details of the signal. In contrast, the important information is encoded by the coefficients having large absolute value.
- Removing the small absolute value coefficients and then reconstructing the signal should produce signal with lesser amount of noise.

#### **STEP 5**

##### **Reconstruction**

- After applying wavelet shrinkage inverse discrete haar wavelet transform is taken.
- This step rejoins the signal to get a continuous de-noisy signal other than eliminated noisy signal.

## **STEP 6**

- Finally SNR ratio for de-noisy signal was calculated by taking the correlation of the de-noisy signal and compared it with the original EEG signal. Figure 4.3 shows the flow chart of proposed methodology.

## **4.7 DE-NOISING PROCEDURE PRINCIPLES**

The general de-noising procedure involves three steps (Tatjana Zikov et al 2010). The basic version of the procedure follows the steps described below.

- **Decompose**
  - Choose a wavelet; choose a level N.
  - Compute the wavelet decomposition of the signal  $s$  at level N.
- **Threshold Detail Coefficients**
  - For each level from 1 to N, select a threshold and applysoft thresholding to the detail coefficients.
- **Reconstruct**
  - Compute wavelet reconstruction using the original approximation coefficients of level N and the modified detail coefficients of levels from 1 to N.

## **4.8 PREPROCESSING OF BRAIN SIGNALS**

Before brain signals can be analyzed, they need to be appropriately processed, for example, to remove artifacts; this section is devoted to such pre-processing methods, first explaining why preprocessing is necessary, and then outlining the state of the-art in preprocessing of brain signals.

### **4.8.1      Need for preprocessing**

EEG recordings typically contain not only electrical signals from the brain, but also several unwanted signals:

- Interference from electronic equipment, as for example the 50 or 60Hz power supply signals.
- Electromyography (EMG) signals evoked by muscular activity
- Ocular artifacts, due to eye movement or blinking.

Those unwanted components may bias the analysis of the EEG, and may lead to wrong conclusions.

### **4.8.2      Preprocessing- Discussion**

Brain signals often contain unwanted signals which may bias the analysis of the signals, and may lead to wrong conclusions. Reviewson several modern approaches were carried to reduce such artifacts; each of those approaches has its own pros and cons. On a more fundamental level, however, it is clear that in order to reliably extract artifacts, one needs to know how brain signals generally look like, and what information content they encode. Therefore, as the understanding of brain signals improves, it should become less difficult to detect and remove artifacts.

## **4.9           OCULAR ZONE IDENTIFICATION**

### **4.9.1      Haar Transform for Analyzing EEG Signal**

The Haar transform is the simplest of the wavelet transforms. This transform cross-multiplies a function against the Haar wavelet with various shifts and stretches, like the Fourier transform cross-multiplies a function

against a sine wave with two phases and many stretches. The Haar transform can be thought of as a sampling process in which rows of the Transform for noise removal. It provides the shortest path and the time consumption is less. In mathematics, the Haar wavelet is a certain sequence of rescaled "square-shaped" functions which together form a wavelet family or basis (Venkata Ramanan et al 2009). Wavelet analysis is similar to Fourier analysis in and allows a target function over an interval to be represented in terms of an orthonormal function basis. The Haar sequence is now recognized as the first known wavelet basis and extensively used as a teaching example in the theory of wavelets. The Haar sequence was proposed in 1909 by Alfred Haar.

Haar used these functions to give an example of a countable orthonormal system for the space of square-integrable function on the real line. The study of wavelets, and even the term "wavelet", did not come until much later. As a special case of the Daubechies wavelet, it is also known as D2. The Haar wavelet is also the simplest possible wavelet. The technical disadvantage of the Haar wavelet is that it is not continuous, and therefore not differentiable. This property can, however, be an advantage for the analysis of signals with sudden transitions, such as monitoring of tool failure in machines.

The Haar wavelet's mother wavelet function  $\psi(t)$  can be described as in Equation (4.4) and Equation (4.5).

$$\psi(t) = \begin{cases} 1 & 0 \leq t < 1/2 \\ -1 & 1/2 \leq t < 1 \\ 0 & \text{otherwise} \end{cases} \quad (4.4)$$

Its scaling function  $\varphi(t)$  can be described as

$$\varphi(t) = \begin{cases} 1 & 1 \leq t < 1 \\ 0 & \text{otherwise} \end{cases} \quad (4.5)$$

#### **4.9.2 Automatic Identification of ocular Artifacts Zones Using Haar Wavelet**

By analyzing the frequency spread of the EEG data that contained the Ocular Artifacts, researchers found that the difference in the frequency of the spikes caused due to rapid eye blink and the EEG signal could be used along with a simultaneous recording of the EOG to detect and remove these artifacts. But correlation of the EEG and EOG is futile, especially because of the inherent corruption of EEG data by the restraint on the user's eye movements and blinks (Senthilkumar et al2008). The accurate detection of these artifacts by singular observation of the time or frequency domains fails and hence the wavelet transform can be used to study the time-frequency maps of the EOG contaminated EEG. Haar wavelet is used to decompose the recorded EEG Signal to detect the exact moment when the state of the eye changes from open to close and vice versa.

Decomposition of the EEG data with the Haar wavelet results in a step function with a falling edge for a change in the state of the eyes from open to close and a step function with a rising edge for a change in state of the eyes from close to open. The same technique is used to detect the ocular artifacts zones in the contaminated EEG.

## **4.10 DECOMPOSITION USING DISCRETE WAVELET TRANSFORM**

The transform of a signal is just another form of representing the signal. It does not change the information content present in the signal. The Wavelet Transform provides a time-frequency representation of the signal. It was developed to overcome the short coming of the Short Time Fourier Transform (STFT), which can also be used to analyze non-stationary signals. While STFT gives a constant resolution at all frequencies, the Wavelet Transform uses multi-resolution technique by which different frequencies are analyzed with different resolutions. A wave is an oscillating function of time or space and is periodic. In contrast, wavelets are localized waves that have their energy concentrated in time or space and are suited to analysis of transient signals.

While Fourier Transform and STFT use waves to analyze signals, the Wavelet Transform uses wavelets of finite energy. The wavelet analysis is done similar to the STFT analysis. The signal to be analyzed is multiplied with a wavelet function just as it is multiplied with a window function in STFT, and then the transform is computed for each segment generated. However, unlike STFT, in Wavelet Transform, the width of the wavelet function changes with each spectral component. The Wavelet Transform, at high frequencies, gives good time resolution and poor frequency resolution; while at low frequencies, the Wavelet Transform gives good frequency resolution and poor time resolution.

## **4.11 SOFT THRESHOLDING**

The wavelet coefficients calculated by a wavelet transform represent change in the time series at a particular resolution. By looking at the time series in various resolutions it should be possible to filter out noise.

However, the definition of noise is a difficult one. Figure 4.4 shows the threshold families.

### The algorithm is

1. Calculate a wavelet transform and order the coefficients by increasing frequency. This will result in an array containing the time series average plus a set of coefficients of length 1, 2, 4, 8... The noise threshold will be calculated on the highest frequency coefficient spectrum (this is the largest spectrum).
2. Calculate the median absolute deviation on the largest coefficient spectrum (Iman Elyasi and SadeghZarmehi2009). The median is calculated from the absolute value of the coefficients. The Equation for the median absolute deviation is shown in Equation (4.7).

$$\delta(\text{mad}) = \frac{\text{median} \{ |c_0|, |c_1|, \dots, |c_{2^{n-1}-1}| \}}{0.6745} \quad (4.7)$$

Here  $c_0, c_1$ , etc... are the coefficients.

The factor 0.6745 in the denominator rescales the numerator so that  $\delta(\text{mad})$  is also a suitable estimator for the standard deviation for Gaussian white noise (Wavelet Methods for Time Series Analysis).

3. For calculating the noise threshold, used a modified version of the Equation in Wavelet Methods for Time Series Analysis. This Equation is shown in Equation (4.8).

$$\tau = \delta_{\text{mad}} \sqrt{\ln(N)} \quad (4.8)$$

In this Equation  $N$  is the size of the time series.

4. Apply a thresholding algorithm to the coefficients. There are two popular versions:

1. Hard thresholding: Hard thresholding sets any coefficient less than or equal to the threshold to zero.

```
if (coef [i] <= thresh)
```

```
coef [i] = 0.0;
```

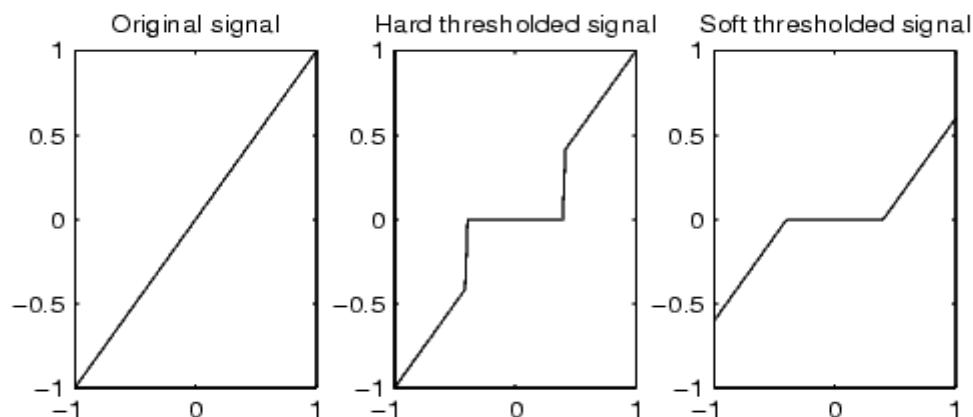
2. Soft thresholding: Hard thresholding sets any coefficient less than or equal to the threshold to zero. The threshold is subtracted from any coefficient that is greater than the threshold. This moves the time series toward zero.

```
if (coef[i] <= thresh)
```

```
coef [i] = 0.0;
```

```
else
```

```
coef [j] = coef [i] - thresh;
```



**a) Original signal    b) Hard threshold signal    c) Soft threshold signal**

**Figure 4.4 Threshold Families**

## 4.12 POWER SPECTRAL DENSITY

The above definitions of energy spectral density requires that the Fourier transforms of the signals exist, that is, that the signals are integrable/summable or square-integrable/square-summable. Often more useful alternative is the power spectral density(PSD), which describes how the power of a signal or time series is distributed with frequency. Here power can be the actual physical power, or more often, for convenience with abstract signals, can be defined as the squared value of the signal, that is, as the actual power dissipated in a purely resistive load if the signal were a voltage applied across it( Paulchamy and IlaVennila 2012). This instantaneous power (the mean or expected value of which is the average power) is then given by in Equation (4.9).

$$P(t) = s(t)^2 \quad (4.9)$$

for a signal  $s(t)$ .

### 4.12.1 Properties of Power Spectral Density

- Spectrum of a real valued process is symmetric:  $S(-f)=S(f)$
- It is continuous and differentiable on  $[-1/2, +1/2]$
- Derivative is zero at  $f = 0$
- Auto-Covariance can be reconstructed by using the Inverse Fourier transform
- It describes the distribution of variance across time scales given in Equation (4.10).

$$Var(X_t) = \gamma_o = 2 \int_0^{1/2} S(\omega) d\omega \quad (4.10)$$

- It is a linear function of the auto-covariance function

- If  $\gamma$  is decomposed into two functions  $\gamma(\tau) = \alpha_1\gamma_1(\tau) + \alpha_2\gamma_2(\tau)$

Then  $S(f)$  is given in Equation (4.11).

$$S(f) = \alpha_1 S_1(f) + \alpha_2 S_2(f) \quad (4.11)$$

where  $S_i(f) = F\{\gamma_i\}$

- The power spectrum  $G(f)$  is defined as in Equation (4.12).

$$G(f) = \int_{-\infty}^f S(f') df' \quad (4.12)$$

### 4.13 SIGNAL-TO-NOISE RATIO

Signal-to-noise ratio (often abbreviated SNR or S/N) is a measure used in science and engineering that compares the level of a desired signal to the level of background noise. It is defined as the ratio of signal power to the noise power. A ratio higher than 1:1 indicates more signal than noise. While SNR is commonly quoted for electrical signals, it can be applied to any form of signal. The signal-to-noise ratio, the bandwidth, and the channel capacity of a communication channel, is connected by the Shannon–Hartley theorem.

Signal-to-noise ratio is defined as the power ratio between a signal (meaningful information) and the background noise (unwanted signal). It is given in Equation (4.13).

$$SNR = \frac{P_{signal}}{P_{noise}} \quad (4.13)$$

$P_{signal}$  = Average power of the signal

$P_{noise}$  = Average noise of the signal

Power of the output signal is calculated as shown in Equation (4.14).

$$P_x \approx \frac{1}{M-1} \sum_{n=0}^{M-1} x^2(n) \quad (4.14)$$

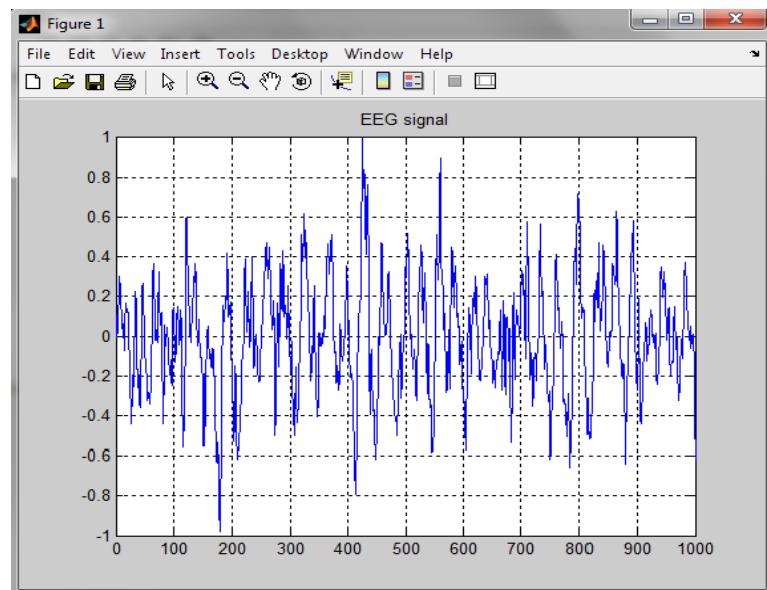
M=Number of samples

$x(n)$ =Input signal

## 4.14 RESULT&DISCUSSION

### 4.14.1 EEG Signal

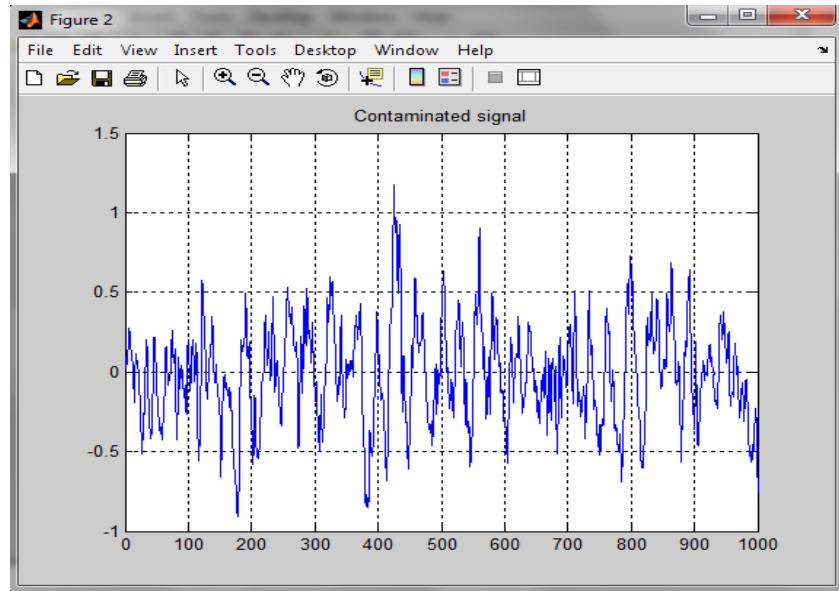
The Figure 4.5 Shows original EEG signal taken from human brain.



**Figure 4.5 EEG signal**

### 4.14.2 Contaminated Signal

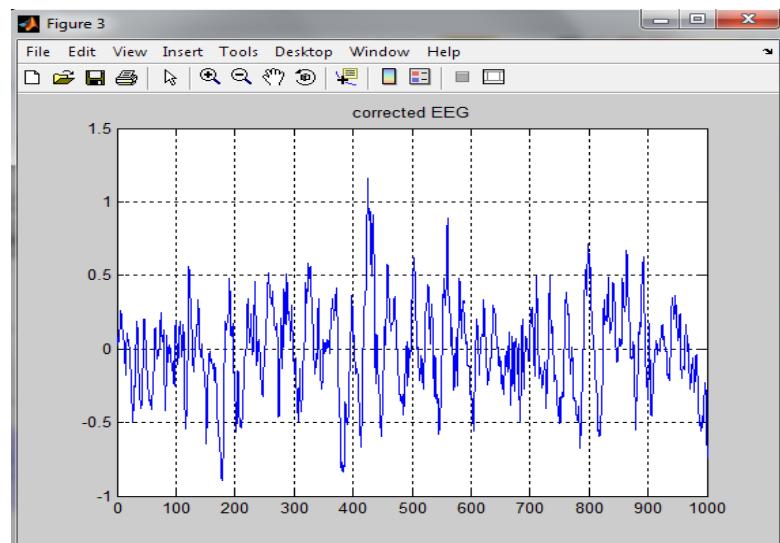
The Figure 4.6 shows contaminated signal (i.e.) combination of original signal and ocular artifacts taken from eye. It is ade-noisy signal. In this the task is to remove noisy signal (EOG) from the below signal.



**Figure 4.6 Contaminated Signal**

#### 4.14.3 Corrected EEG

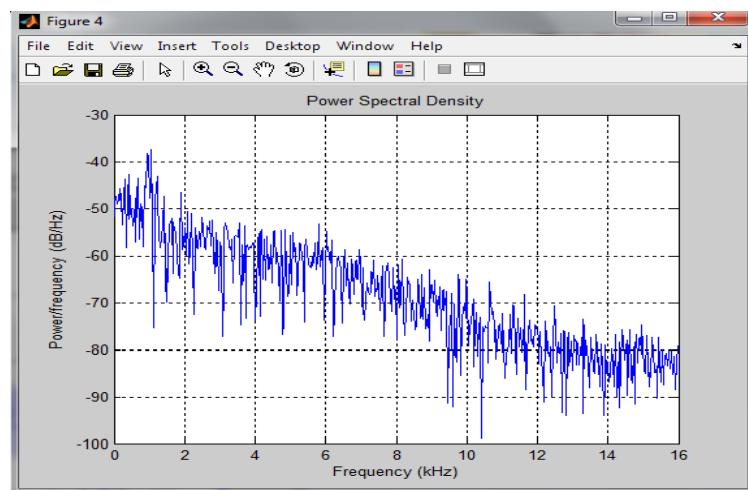
Figure 4.7 shows de-noisy signal. The noisy signal is removed from Figure 4.6 using Haar wavelet transform and soft thresholding value. This corrected signal gives original signal as taken from human brain.



**Figure 4.7 Corrected EEG Signal**

#### 4.14.4 Power Spectral Density

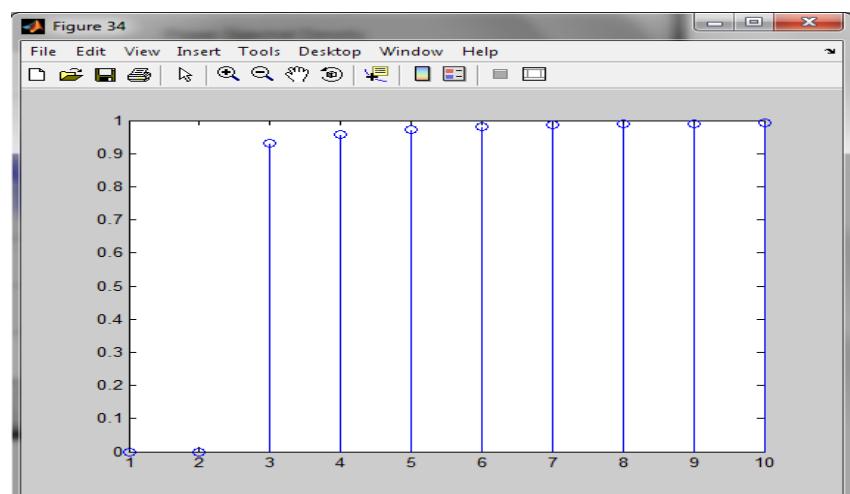
Figure 4.8 Shows power spectral density of de-noised signal and it had been carried out for different trials.



**Figure 4.8 Power Spectral Density**

#### 4.14.5 Correlation Plot

Figure 4.9 shows the correlation plot for de-noised signal. This correlation plot gives the relation between noise and de-noised signal.



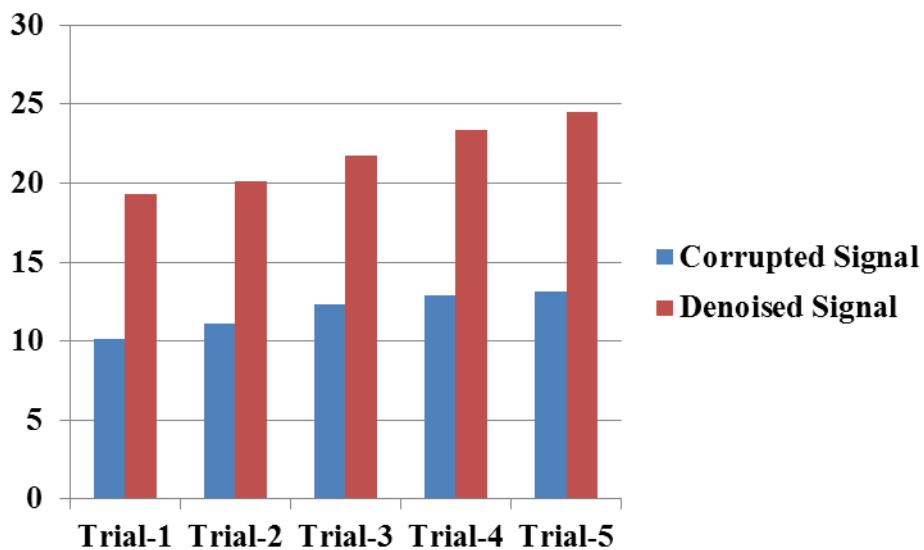
**Figure 4.9 Correlation Plot**

#### **4.15 SNR COMPARISON FOR NOISY AND DENOISY SIGNAL**

EEG Data with ocular Artifacts are taken for testing the Proposed Method. The Data is sampled at a rate of 128 Samples/Second. The effect of Ocular Artifacts will be dominant in the frontal and front polar channels. Hence it is Sufficient to apply the algorithms to these channels. Table 2.Summarises the Signal to Noise ratio Comparison of 5 trials of Noisy signal (EEG +EOG Artifact) and De-noisedSignal (Corrected EEG). Obviously the signal to Noise Ratio of de-noised signal is higher than the Noisy Signal. Table 4.1 shows the SNR Comparison between Noisy Signal and Demised Signal

**Table 4.1 SNR Comparison between corrupted signal and De-noised Signal**

<b>Trails</b>	<b>SNR for Corrupted Signal</b>	<b>SNR for Denoised Signal</b>
Trial-1	10.1040	19.3050
Trial-2	11.1020	20.1050
Trial-3	12.3010	21.7030
Trial-4	12.9210	23.4030
Trial-5	13.1020	24.5040



**Figure 4.10 SNR Curve for corrupted and denoised signal**

Figure 4.10 shows the signal to Noise ration curve for corrupted and denoised signal. The SNR of denoised signal is higher than the corrupted signal.

#### 4.16 SUMMARY

This chapter presents an effective approach for the diagnosis of brain signal using Haar wavelet transform. A method to identify the ocular artifact through Haar wavelet transform is proposed and soft-like thresholding is applied to the ocular artifact zones. An adaptive thresholding applied only to the ocular artifact zone does not affect the low frequency components but preserves the shape of the EEG signal in the non-artifact zones which is of great importance in the clinical diagnosis. Power Spectral Density and correlation values are used as performance metrics in this research. In all cases, artifacts are adequately attenuated, without removing significant and useful information.

It is concluded that the proposed method gives less complexity and provide easier technique for the removal of artifacts with the help of the wavelet decomposition and is an efficient technique for improving the quality of EEG signals in biomedical analysis. The proposed method minimizes the amplitude of the ocular artifact, preserving the magnitude and phase of the high frequency back ground EEG activity compared to the proposed method.

Efforts should be directed towards designing Haar and other similar discontinuous wavelets for highly artifact selective detection and de-noising. The decomposition and reconstruction of the signal is performed effectively by using discrete wavelet transform and inverse discrete wavelet transform. But it gives only an estimate in providing an inference relating to therelative superiority of the algorithms used for removing ocular artifacts from EEG. Further, it is the considered opinion that a suitable performance metric for validating the de-noised EEG signals should be devised for quantitatively comparing the signal to noise ratio of the signal. The signal to noise ratio has taken for different trials in the entire length of the EEG signal.

## CHAPTER 5

### MULTIWAVELET TRANSFORM

The wavelet transform is effectively implemented in signal denoising. The multiwavelet transform which is a newer alternative of wavelet, comprises some important differences compared to wavelets. Wavelets can be effectively utilized in multiresolution analysis with scaling function  $\phi(t)$  and wavelet function  $\psi(t)$ . The wavelets allow the usage of multiple scaling and wavelet functions. Multiwavelets which have been evolved behind this idea are considered as a natural extension of wavelets. The multi wavelets hold significant advantages over standard wavelets for solving more complicated problems and hence are of great interest.

#### **5.1 INTRODUCTION**

In the earlier techniques, multiresolution analysis was done using functions. The analysis of functions can only express the functions belonging to a certain space as a linear combination of basis functions. Wavelets have preceded a step further in multiresolution analysis. In the analysis using wavelets, the wavelets consider every basis function be a dilation and translation of a single scaling function of unit norm, denoted  $\phi(t)$  ( Coifman et al1992). In wavelet analysis, the integer translations should be linearly independent and produce an orthonormal basis for the subspace  $V_0$ . For fixed integer scale  $j$ , the translations  $\{2^{-j/2}\phi(2^{-j}t-k)\}|k \in \mathbb{Z}$  form an orthonormal basis for the subspace  $V_j$ , such that the subspaces satisfy the function  $\dots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \subset \dots$

The Equation (5.1) shows that relation.

$$\overline{\bigcup_{j=-\infty}^{\infty} V_j} = L^2(R), \quad \bigcap_{j=-\infty}^{\infty} V_j = \{0\} \quad (5.1)$$

This allows each function  $f(t) \in L^2(R)$  to be written as a linear combination of these basis functions with weights  $\alpha_{j,k}$ . The  $f(t)$  can be defined as follows in Equation (5.2) and Equation (5.3).

$$f(t) = \sum_{j=-\infty}^{\infty} 2^{-j/2} \sum_{k=-\infty}^{\infty} \alpha_{j,k} \phi(2^{-j}t - k) \quad (5.2)$$

$$\alpha_{j,k} = \int_{-\infty}^{\infty} f(t) 2^{-j/2} \phi(2^{-j}t - k) dt$$

where,

(5.3)

The scaling coefficients are represented by the weights  $\alpha_{j,k}$ . One of the important point to be taken into consideration is that the orthogonal expansion of  $f(t)$  in the  $V_j$  subspaces are not mutually disjoint. The functions  $\{2^{-j/2}\phi(2^{-j}t - k)\}_{j,k \in \mathbb{Z}}$  are not linearly independent across scales, so they do not represent a basis for  $L^2(R)$ .

The nesting property of the subspace satisfies the two scale dilation equation. It infers that the scaling function  $\phi \in V_0$  also belongs to  $V-1$ . The two-scale dilation is represented by Equation (5.4).

$$\phi(t) = \sqrt{2} \sum_{k=-\infty}^{\infty} h_k \phi(2t - k) \quad (5.4)$$

where the sequence  $\{h_k\}$  will become important in the next section. In this section will consider the set  $\{h_k\}$  as the coefficients of the orthogonal projection of  $\phi(t)$  onto the basis  $\{\sqrt{2} \phi(2t - k)\}_{k \in \mathbb{Z}}$  of  $V-1$ , but later we will consider them as low pass filter coefficients in a filter bank. When the scale  $j$

decreases, the support of each basis function  $2^{-j/2}\varphi(2^{-j}t-k)$  lessens as well. This clearly shows that the smaller scales generate a much finer and detailed resolution. The multiresolution analysis can be used to produce graphical interpretation.

Though the larger values of  $j$  indicate basis functions with wider support, the coarser scales provide fewer details. This makes sense, since  $V_j \subset V_{j-1}$  implies that  $V_{j-1}$  represents a space of functions at a finer resolution than  $V_j$  (Ingrid Daubechies 1992.). Since orthogonal decomposition of  $f(t)$  is required, the “difference” space  $W_j$  is denoted as the complement of  $V_j$  in  $V_{j-1}$ , as below in Equation (5.5)

$$V_{j-1} = V_j \oplus W_j, V_j \cap W_j = \emptyset \quad (5.5)$$

$L^2(\mathbb{R})$  is decomposed into mutually orthogonal subspaces by the subspace  $W$  as follows, in Equation (5.6).

$$W_j \perp W_{j'} \text{ if } j \neq j', \text{ and } W_{j'} \oplus W_j = L^2(\mathbb{R}) \quad (5.6)$$

The basis for the subspace  $W_0$  consists of translations of a new function,  $\psi(t)$  form. The basis for  $W_j$  will be  $\{2^{-j/2}\psi(2^{-j}t-k)\}|k \in \mathbb{Z}$  since  $W$  spaces inherit the scaling properties of the  $V$  spaces. Therefore similar to  $V$  spaces, each function  $f(t) \in L^2(\mathbb{R})$  is written as a linear combination of these basis functions with weights  $\beta_{j,k}$ . It is shown in Equation (5.7).

$$f(t) = \sum_j 2^{-j/2} \sum_{k=-\infty}^{\infty} \beta_{j,k} \psi(2^{-j}t - k) \quad (5.7)$$

$$\beta_{j,k} = \int_{-\infty}^{\infty} f(t) 2^{-j/2} \psi(2^{-j}t - k) dt \quad (5.8)$$

where

The wavelet function is represented the function  $\psi(t)$ . The wavelet coefficients are denoted by the numbers  $\beta_{j,k}$ . Equation (5.8) indicates the orthogonal expansion of  $f(t)$ . A basis for  $L^2(\mathbb{R})$  is represented by the functions  $\{2^{-j/2}\psi(2^{-j}t-k)\}_{j,k \in \mathbb{Z}}$ . Since  $\psi \in W_0$  and  $W_0 \subset V-1$ ,  $\psi(t)$  value is given in Equation (5.9).

$$\psi(t) = \sqrt{2} \sum_{k=-\infty}^{\infty} g_k \phi(2t - k) \quad (5.9)$$

with orthogonal projection coefficients  $\{g_k\}$ . In Equation (5.10) each subspace  $V_M$  is obtained as the result of direct sum of  $V_N$  and some  $W$  subspaces, for  $N > M$ ,

$$V_M = V_N \oplus W_N \oplus W_{N-1} \oplus W_{N-2} \oplus \dots \oplus W_{M+1} \quad (5.10)$$

There is still another way of representing the subspace  $V_M$  as a linear combination of basic functions of the mutually orthogonal subspaces  $V_N, W_N, W_{N-1}, \dots, W_{M+1}$ . For a given the function  $f(t) \in V_M$  write as follows in Equation (5.11).

$$f(t) = 2^{-N/2} \sum_{k=-\infty}^{\infty} \alpha_{N,k} \phi(2^{-N}t - k) + \sum_{j=M+1}^N 2^{-j/2} \sum_{k=-\infty}^{\infty} \beta_{j,k} \psi(2^{-j}t - k). \quad (5.11)$$

The next section describes the filter bank representation of the wavelet transform. Here, the selection of subspaces will correspond to the tree-shaped octave-band iteration of the analysis bank.

## 5.2 MULTIWAVELET

The name multiwavelets itself suggests that multiwavelets can be implemented using several wavelets having several scaling functions (Geronimo and Assopust 1994). Multiwavelets offer several advantages such as compact support, Orthogonality, symmetry, and high order approximation

compared to scalar wavelets (Strang and Strela 1995). A scalar wavelet can never inherit all these properties simultaneously (Strang and Nguyen 1995). Further a multiwavelet system provides perfect reconstruction while preserving length, good performance at the boundaries, and a high order of approximation coincidentally. Thus, multiwavelets play a significant role in signal processing.

In image processing applications, the multiwavelets enhance the possibility of superior performance and high degree of freedom compared to scalar wavelets. A multiwavelet is said to have multiplicity  $r$  when it contains  $r$  scaling functions and  $r$  wavelet functions is said to have multiplicity  $r$ . The multiwavelet system reduces to a scalar wavelet system when  $r=1$ . Multiwavelets can possess two or more scaling functions and wavelet functions. In order to provide notational difference for a multiwavelet system, the set of scaling functions can be written using the vector notation is given in Equation (5.12)

$$\phi(t) = [\phi_1(t), \phi_2(t), \dots, \phi_r(t)]^T \quad (5.12)$$

where,  $\phi(t)$  is called the multi-scaling function. The Multiwavelet function is defined from the set of wavelet function is given in Equation (5.13).

$$\psi(t) = [\psi_1(t), \psi_2(t), \dots, \psi_r(t)]^T \quad (5.13)$$

When  $r=1$ ,  $\psi(t)$  corresponds to a scalar wavelet. The Multiwavelet system requires two or more input streams the Multiwavelet filter bank. The theory of Multiwavelets also has its basis in multiresolution analysis (MRA) when compared to scalar wavelets. However, the multiwavelets possess several scaling functions. For Multiwavelets, the notion of MRA is the same except that now a basis for  $V_0$  and  $V_1$  is generated by translates of  $N$  scaling functions given in Equation (5.14).

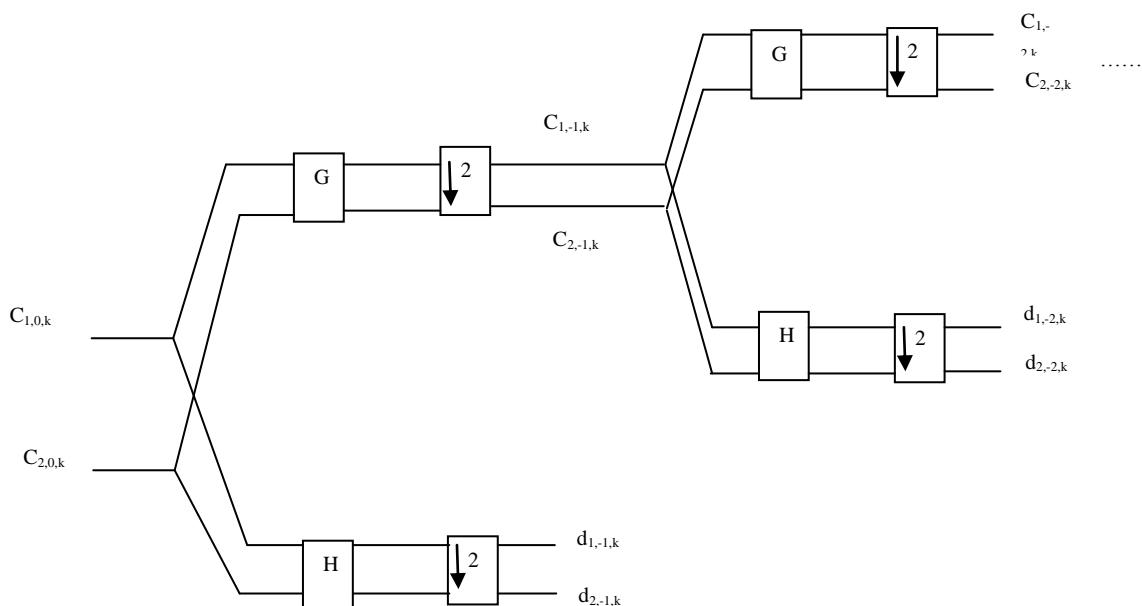
$$\phi_1(t-k), \phi_2(t-k), \dots, \phi_N(t-k) \quad (5.14)$$

The matrix dilation is satisfied by the multi scaling function and the Multiwavelet functions as in the following Equation (5.15) and Equation (5.16).

$$\Phi(t) = \sqrt{2} \sum_{k=-\infty}^{\infty} H_k \phi(2t - k) \quad (5.15)$$

$$\psi(t) = \sqrt{2} \sum_{k=-\infty}^{\infty} G_k \phi(2t - k) \quad (5.16)$$

The filter coefficients  $H_k$  and  $G_k$  are  $N$  by  $N$  matrices instead of scalar. Each of the multiwavelets comprises of a matrix-valued multi-rate filter bank. The Multiwavelet filter bank is composed of “taps” that are  $N \times N$  matrices. A symmetric Multiwavelet filter bank with 4-coefficients possesses a low pass filter which is denoted by the four  $N \times N$  matrices and is named C. The high pass filter which is named as D, cannot be obtained directly as an “alternating flip” of the low pass filter as in the case of scalar 2-band Para unitary filter bank. In multiwavelet filter bank, the wavelet filters D has to be designed. The result obtained is an  $N$  channel,  $N \times N$  matrix filter bank that operates on  $N$  input data. These inputs are filtered into  $2N$  output streams, each of which is down sampled by a factor of 2. This is shown in Figure 5.1.



**Figure 5.1 Decomposition of the Signal**

Each row of the multi filter is derived as a combination of N ordinary filters. Each of them operates on a separate data stream.

Major reasons behind utilizing multi wavelets are listed below:

1. The multi wavelets can be used to restrictions on the filter properties because they provide extra degrees of freedom.
2. Symmetric signal extension necessitates the usage of symmetric filters.
3. The multi wavelets have the capability to possess all these properties simultaneously.
4. The property of orthogonality can be effectively used to design and implement the transforms in a simpler way.

The amount of energy compaction attained by a transform is another important feature to be considered in signal denoising. Only a filter with good energy compaction property can decorrelate a fairly uniform input signal into a small number of scaling coefficients, retaining most of the energy and a large number of sparse wavelet coefficients. The wavelet coefficients are represented with significantly fewer bits on average than the scaling coefficients. Therefore, energy compaction becomes much important during quantization(Xiang-Gen Xia et al 1995). In order to obtain better performance and reduce quantization noise, the wavelet coefficients must have values clustered about zero with little variance. The multiwavelets have the capability to provide better reconstruction quality at the same bit rate.

### **5.3 PREPROCESSING FOR MULTIWAVELETS**

One of the important issues to be handled in multiwavelets which are used in transform process is that Multiwavelet filter banks require a vector-valued input signal. Such an input signal can be easily produced from

a 2-D image using several ways. However, the most excellent method is to use the adjacent rows and columns of the image data (Vasily Strela 1996). But this method does not suit well for general multiwavelets because it generates reconstruction artifacts in the lowpass data after coefficient quantization. This problem can be handled by constructing “Constrained” multiwavelets, which possess certain key properties (Vasily Strela 1996). But the Constrained multiwavelets are somewhat restricted which degrade their performance compared to some other multiwavelets. There is a better approach that allows to split each row and column into two half-length signals and then feed these two half signals as the channel inputs into the multifilter(Strela et al1998). There is yet approach that allows taking the odd samples for one signal and the even samples for the second signal. But the main disadvantage of this approach is that it destroys the assumed characteristics of the input signal and does not work well. It is a well known fact that image data can be locally well-approximated by low-order polynomials, usually constant, linear, or quadratic. When the input takes this form, the high pass filters are designed in such a way that they produce uniformly zero output but when alternating data points are taken as filter inputs, the character of the input signal is altered by the filter inputs so the output cannot be forced to remain zero. This reduces the compression performance. However, this problem can be handled by pre-filering the half-length signals before passing them into the multifilter.The properties of the input signal are adjusted this prefilter step.

That splits one scalar signal into two half-length signals in such a way that the orders of approximation built into the multifilter are utilized (Vasily Strela 1996,Strela and Walden1998 , Jo Yew Tham et al 1998). The two signals that are supposed to be prefilterd must be in the form of  $2 \times N$  matrix (where the original 1-D signal had length  $2N$ ). These signals are then

multiplied by one or more  $2 \times 2$  prefilter matrices. The major limitation in the earlier methods is that these methods were tied to a specific multifilter or required more than one prefilter matrix. Tham et al. (Jo Yew Tham et al 1998) proposed a method which made use of a single orthogonal prefilter matrix for any given multifilter. Further, this method could provide some optimization of the prefilter properties to match any given multifilter. When classes of symmetric and antisymmetric multiwavelets were subjected to the above method, it produced a prefilter matrix with entries of equal magnitude (Xiang-Gen Xia et al 1995, Vasily Strela 1996, Strela and Walden 1998). If the overall constant were absorbed into the multifilter itself, then the preprocessing operation would require no multiplications and only two additions for each input vector. The synthesis stage is provided with a matching post filter operation that neutralizes the effects of prefilter.

#### **5.4 MOTIVATION FOR MULTIWAVELETS**

In image denoising, the algorithms based on scalar wavelets were found to work quite well. Therefore, the multiwavelets must yield better results compared to scalar wavelets. The major reasons which make multiwavelets more advantageous when compared to scalar wavelets are summarized below. First and foremost, the multiwavelets possess the inherent properties of extra degrees of freedom. This reduces the restrictions on the filter properties (Gilbert Strang and Truong Nguyen 1996). Scalar wavelets possess some restrictions. It cannot have both orthogonality and a symmetric impulse response simultaneously. In order to design and implement the transforms easily, orthogonality is essential. Symmetric filters are indispensable for symmetric signal extension. Further, in scalar wavelets, the support length and the number of vanishing moments depend directly on the filter length. Therefore, in scalar wavelets, higher order of approximation can

be achieved only by the use of longer filter lengths. But longer filter lengths can be attained only at the cost of wavelet's increased interval of support (in the time domain). Better coding gain can be achieved only with higher order of approximation but shorter wavelet support is generally desired for better localized approximation of the input function. Compared to scalar wavelets, the multiwavelets possess all these properties simultaneously. For example consider the GHM multiwavelet (Geronimo et al 1994).

It contains all the above said properties like orthogonality, second order of approximation, symmetric scaling and wavelet functions (and thus symmetric filters), and short support for both of its scaling functions ([0,1] and [0,2], respectively). This combination cannot be attained with scalar wavelets. Consider the scalar filter, 4-tap Daubechies filter. It contains all the important properties like orthogonality, second order of approximation, and scaling function support on [0,3] but lacks the important property of symmetry. Similarly, the bi-orthogonal 9/7 wavelet has symmetric filters, fourth order of approximation (in both analysis and synthesis filters), and scaling function support on [0,9], but lacks orthogonality.

The previous literature reveals the fact that multiwavelets serves as a better tool in signal denoising but the signal de-noising results revealed by Strela and Walden (Strela and Walden1998) using the Bi9/7 scalar wavelet show that it gives better results when compared to the older multiwavelets on signals like EEG. However, newer multifilters have provided results which are much better and convincing when compared to some of the better previous scalar filters. A paper by Strela et al. (Strela et al 1998)presents results in which at least one multiwavelet dramatically outperformed scalar wavelets on a synthetic test signal.

Finally, considering the computational complexity, the scalar wavelets have a clear advantage over multi wavelets. Each of the branches in a multiwavelet filter bank has two channels and 2-input, 2-output filters. But the symmetric bi-orthogonal scalar wavelets are efficient because each of the scalar filters in a symmetric-antisymmetric multifilter has the same kind of symmetry. In a multifilter system, each filter processes less data compared to the scalar filter at the same level. The Multi wavelets require twice as much computational complexity when compared to scalar wavelet, provided all the other factors are equal. But the multi wavelets have the capability to provide performance comparable to scalar wavelets with shorter filters. For example, the length-4 multifilters SA4 and ORT4 require 4 multiplications and 7 additions per sample. Xia et al (Tao Xia and Qingtang Jiang 1998) shows that the Bi9/7 scalar wavelet (with M1=7 and M2=9) requires 4.5 multiplications and 7 additions per input sample. This clearly indicates that their performances are comparable. With regards to the by Wei et al (Dong Wei et al 1998), the Bi22/14 scalar filter requires 9 multiplications and 17 additions per input sample. But the longest multifilter BSA9/7 requires only 8 multiplications and 14 additions per input sample. Also, the EEG signal artifacts removal results clearly indicate that the multiwavelets have the capability to achieve the same level of performance as scalar wavelets with similar computational complexity.

## **5.5 MULTIWAVELET PACKETS**

Similar to scalar wavelets, this procedure involves repeating the filtering operation on the lowpass channel of the filter bank. By repeating the filtering operation on the highpass channels, new basis functions can be formed. This approach is named as multiwavelet packets because it combines the wavelet packet decomposition with multiwavelet filters. Though this idea

is much simple, it provides lesser computational complexity. Table 5.1 provides a comparison of computational complexities of symmetric wavelets and antisymmetric multiwavelets for one level of analysis. M1 and M2 are the low pass and high pass filter lengths and L is the length in samples of the scalar-valued input signal.

**Table 5.1 Comparison of computational complexities of wavelets and Multiwavelet**

FILTER TYPE	MULTIPLICATIONS	ADDITIONS
Scalar Wavelet, odd length	<u>L(M1 +M2 + 2)</u> 4	<u>L(M1 +M2 - 2)</u> 2
Scalar Wavelet, even length	<u>L(M1 +M2)</u> 4	<u>L(M1 +M2 - 2)</u> 2
Multiwavelet, odd length	<u>L(M1 +M2)</u> 2	<u>L(M1 +M2 - 2)</u>
Multiwavelet, even length	<u>L(M1 +M2)</u> 2	<u>L(M1 +M2 - 1)</u>

Let  $U_0(t) \equiv \Phi(t)$  and  $U_1(t) \equiv \Psi(t)$ , given in Equation (5.17) and Equation (5.18).

$$U_{2n+1}(t) = \sqrt{2} \sum_{k=-\infty}^{\infty} H_k U_n(2t - k) \quad (5.17)$$

$$U_{2n+1}(t) = \sqrt{2} \sum_{k=-\infty}^{\infty} G_k U_n(2t - k) \quad (5.18)$$

The multiwavelet packets are much similar to scalar wavelet packets. Even the basis selection algorithm and cost functions used to build the tree structure are much similar in both the cases with only one exception. In multiwavelet packets, each branching in the tree structure creates four new channels (assuming  $r=2$ ) instead of two channels in scalar wavelet packets. This is due to the dual-channel nature of multiwavelet filter banks. Thus, in multiwavelet packets, each of the parents has four children which increase the computational complexity.

The methods based on cost functions operate on all the pixels corresponding to each node. Therefore, they remain unaffected. In case of multiwavelet packets, there are four nodes instead of two but each node represents half as much data so the net effect is zero. Some of the methods make use of the rate-distortion optimization techniques. However, the increased number of nodes increase the computational complexity of these methods.

The motivation to use multiwavelet packets still holds as it does for scalar wavelets so as to better capture high frequency content and oscillations in the original image data, while retaining the benefits of multiwavelet filters. Since no published literature has yet tested multiwavelet packets, we do so here.

## **5.6 MULTIWAVELET DENOISING TECHNIQUE**

EEG is nothing but a signal super imposed on a noisy signal. Let  $S(t)$  represent the true signal and  $\varepsilon(t)$  represent the external noise. The measured signal is written in the form as Equation (5.19).

$$X(t) = S(t) + \varepsilon(t) \quad (5.19)$$

Assume that that  $S(t)$  and  $\varepsilon(t)$  are uncorrelated and are stationary processes. When a signal is decomposed using wavelet transform, a set of wavelet coefficients that correlates to the high frequency sub-bands are obtained. The details in the data set are present within these high frequency sub-bands. The details might be neglected affecting the main features of the data set if they are negligible. Thresholding provides promising results in signal and image de-noising. The proper selection of threshold limit, thresholding function and window sizes is considered to be much important in denoising procedure because the original signal coefficients which contain the critical information in the analyzed data should not be removed. In this chapter, the following thresholding (statistical empirical) formula is used for calculating the thresholding limits. This formula produces better de-noised results than (Krishnaveni et al 2004), which is applied to the entire length of the signal.

Threshold based on Statistics of the signal

Threshold Value is given in Equation (5.20).

$$T_k = N \times \left( \frac{\bar{x} - \sigma}{\bar{x} + \sigma} \right) \quad (5.20)$$

Window Length=10 Seconds

where  $N$  is a Positive Integer, ranging from 100 to 150

$\bar{x}$  - Mean of all samples

$\sigma$  - Standard deviation of all samples

## 5.7 ASSESSMENT CRITERIA

- **Signal to Noise Ratio**

Signal-to-noise ratio is defined as the power ratio between a signal (meaningful information) and the background noise (unwanted signal). It is given in Equation (5.21).

$$SNR = \frac{P_{signal}}{P_{noise}} \quad (5.21)$$

where P is average power.

- **Correlation**

Correlation is nothing but similarity between two signals. Cross correlation is known as similarity between two different signals.

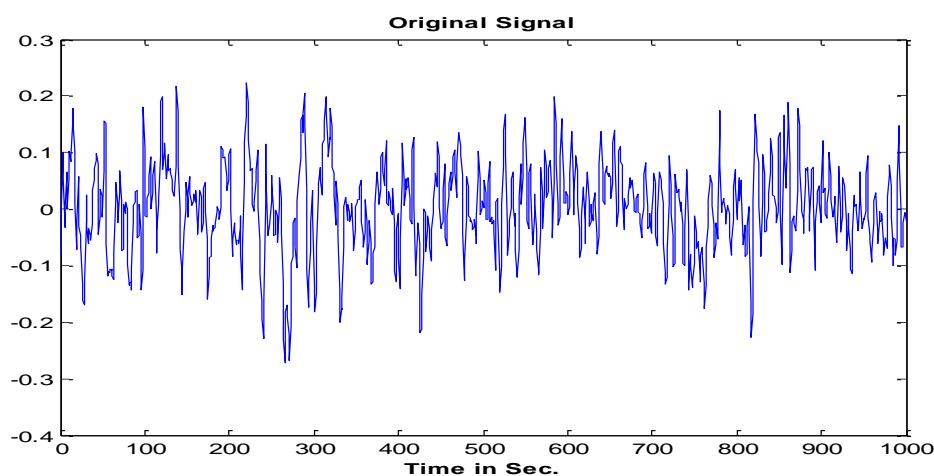
The CC which is used to these similarities is measured as follows, in Equation (5.22).

$$CC = \frac{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2} \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \bar{y})^2}} \quad (5.22)$$

where  $x_i$  and  $y_i$  are the samples of the original signal and its reproduced version,  $x$  and  $y$  represents their averages respectively.

## 5.8 RESULTS & DISCUSSION

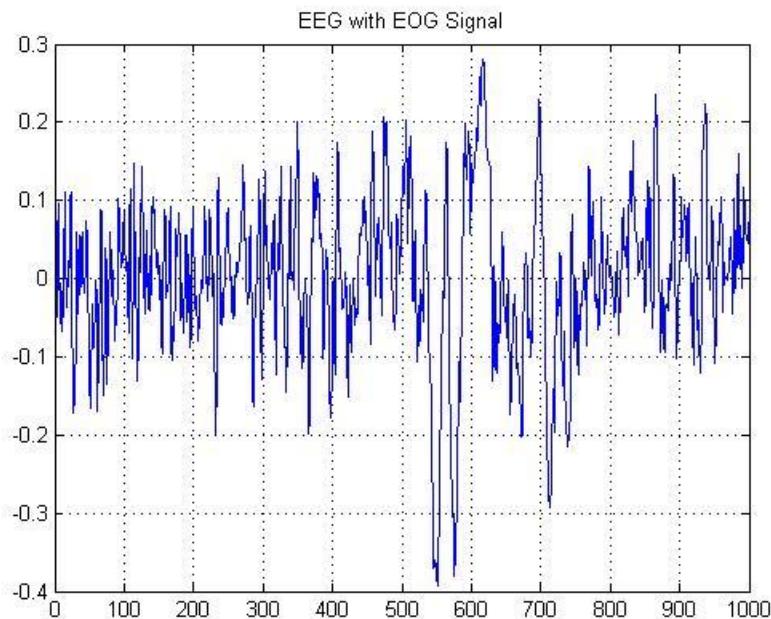
EEG data with artifacts are taken from the website [http://www.sccn.ucsd.edu/~arno/famzdata/publicly\\_avalble\\_EEG\\_data.html](http://www.sccn.ucsd.edu/~arno/famzdata/publicly_avalble_EEG_data.html)(Schlogal et al 2007) for testing the proposed methods. The effect of artifacts is dominant in the Frontal and fronto -polar channels like Fp1, Fp2, F7, F8. Hence it is sufficient to apply the method to these channels. In the case of multiwavelet transform the denoising of EEG signals is carried out by using threshold limit, threshold function and window Size. Choice of threshold limit and thresholding function is a crucial step in the de nosing procedure, as it should not remove the original signal coefficients leading to loss of critical information in the analyzed data(Amari and Cichocki 1996), (Bell and Sejnowski 1995). The Figure 5.2 shows the original EEG.



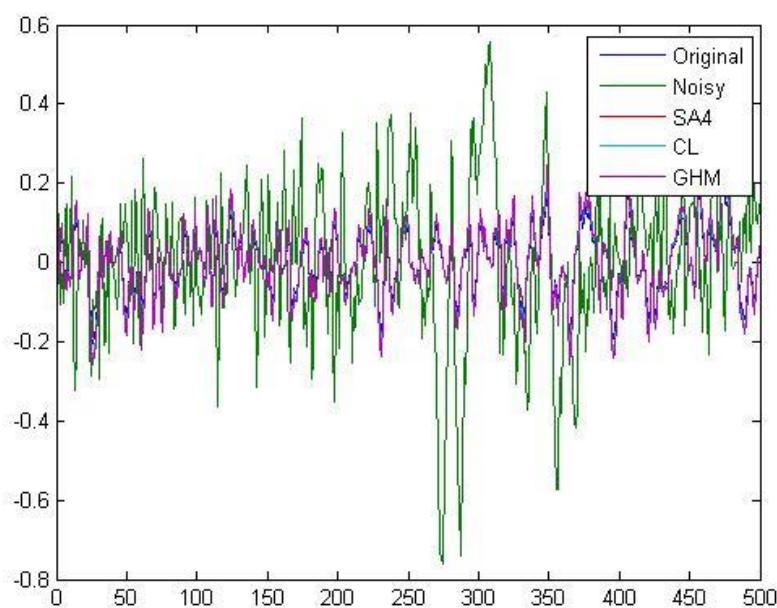
**Figure 5.2 Original EEG**

The figure 5.3 to 5.12 describes the empirical results for five trials of EG contaminated with EOG artifacts and power spectral density plot for original and denoised EEG using Multiwavelets.

➤ **Trial 1**

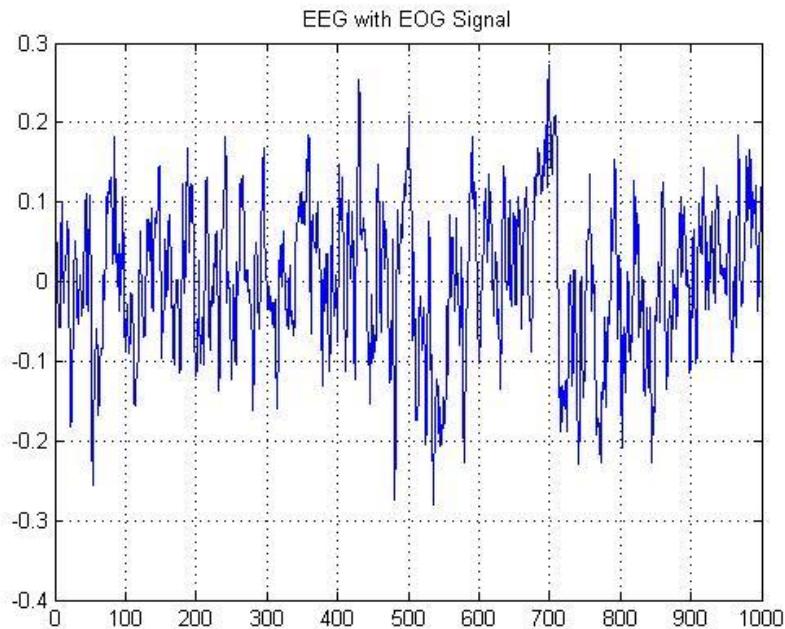


**Figure 5.3 EEG with EOG signal**

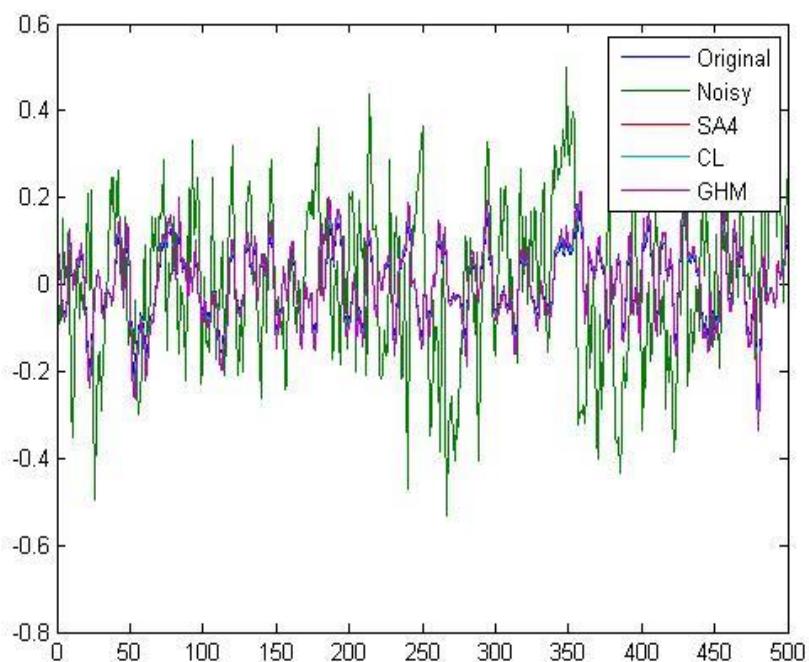


**Figure 5.4 Power spectra plot**

➤ **Trial 2**

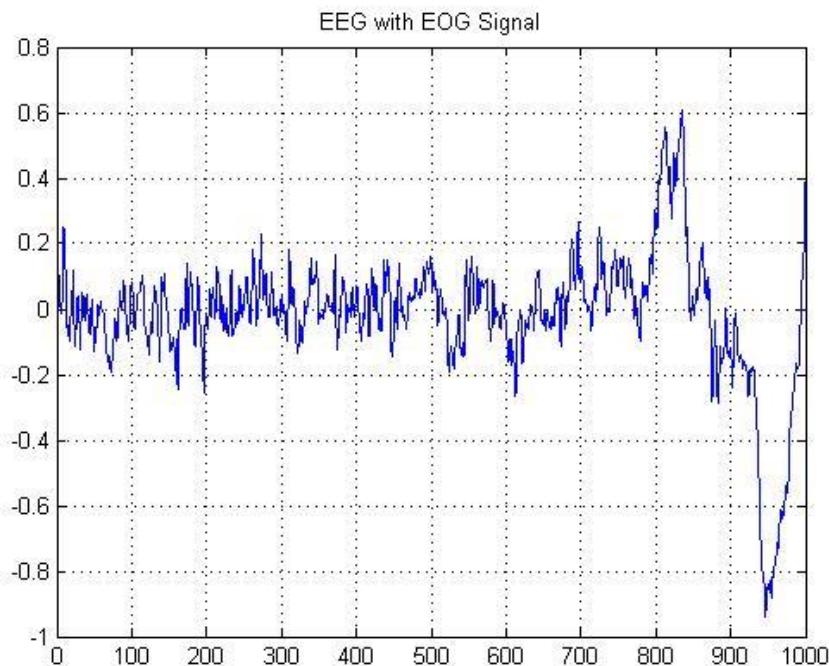


**Figure 5.5 EEG with EOG signal**

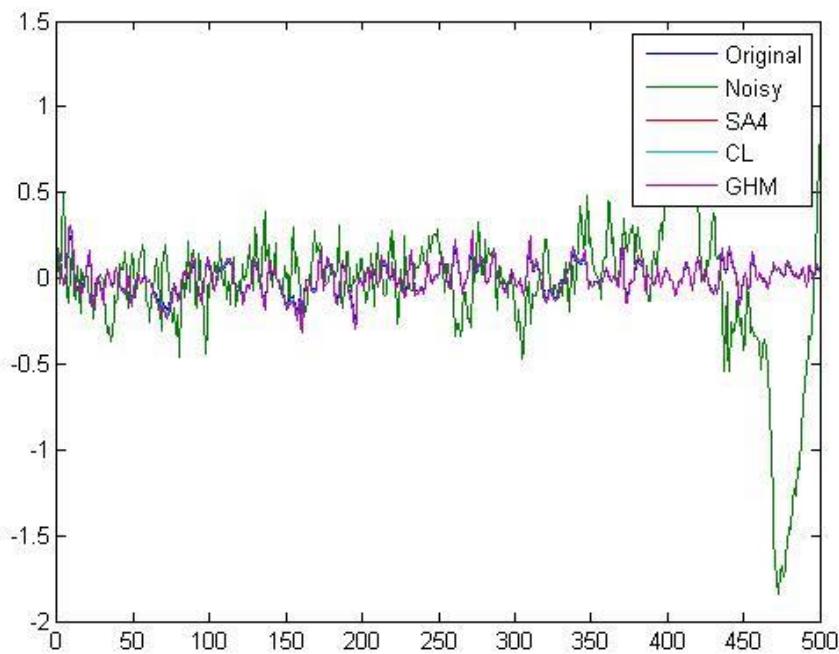


**Figure 5.6 Power spectra plot**

➤ **Trial 3**

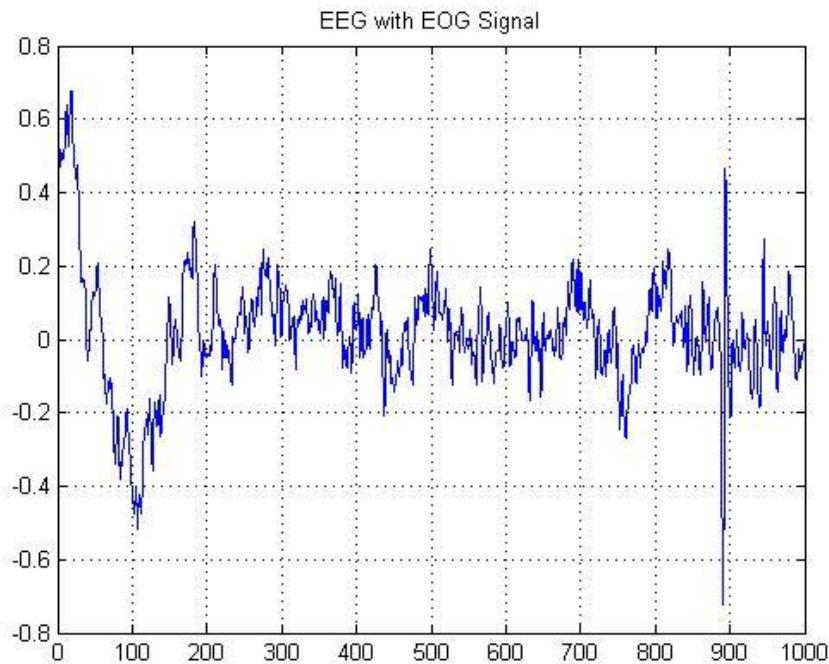


**Figure 5.7 EEG with EOG signal**

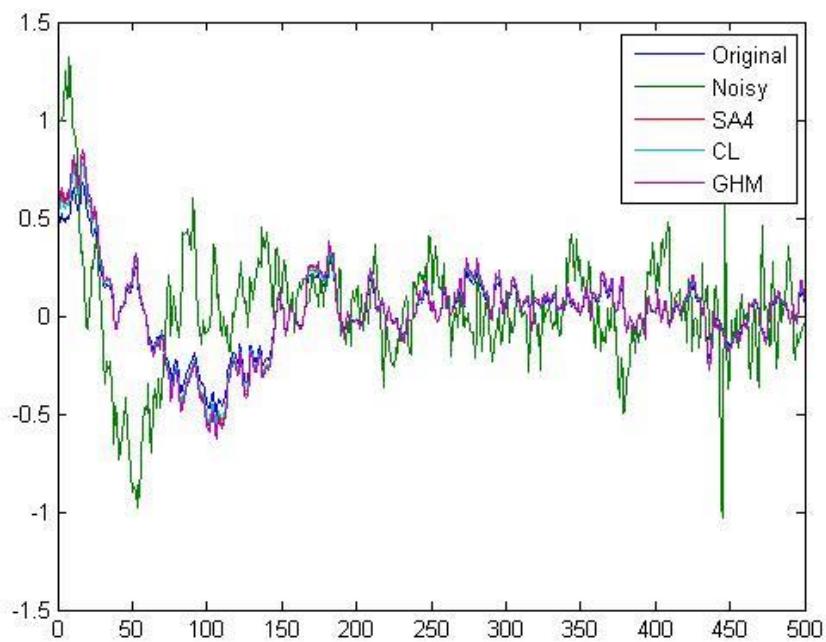


**Figure 5.8 Power spectra plot**

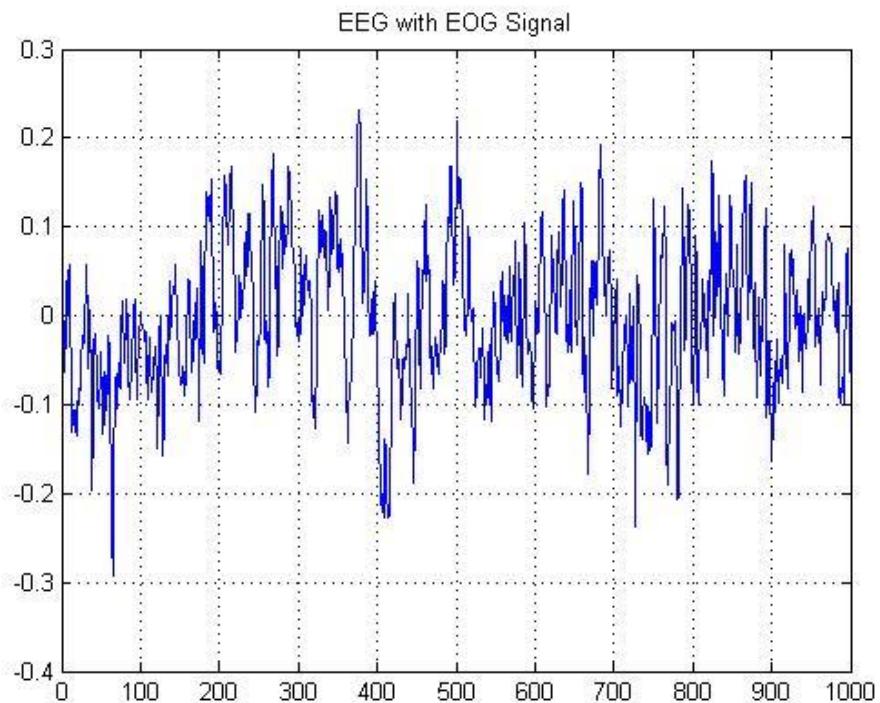
➤ **Trial 4**



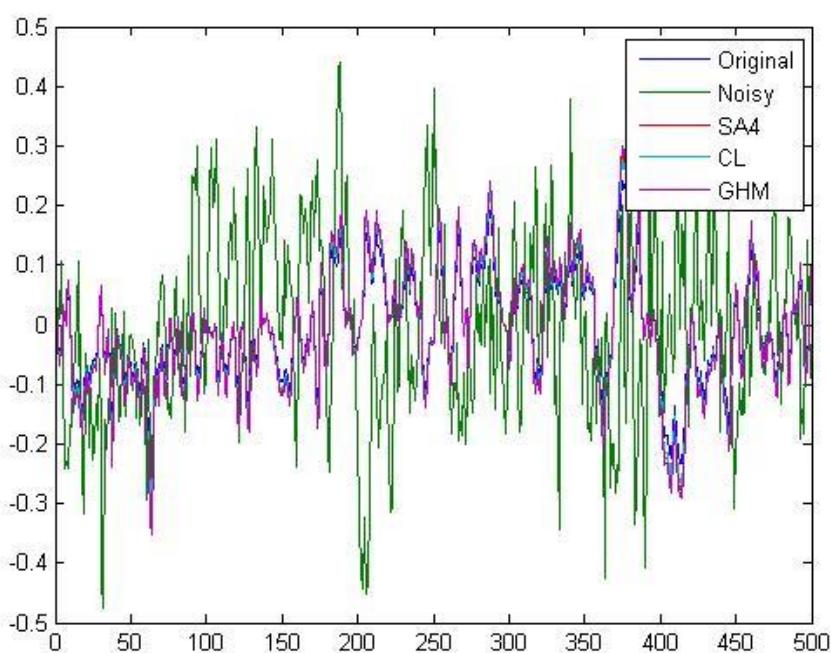
**Figure 5.9 EEG with EOG signal**



**Figure 5.10 Power spectra plot**

➤ **Trial 5**

**Figure 5.11 EEG with EOG signal**

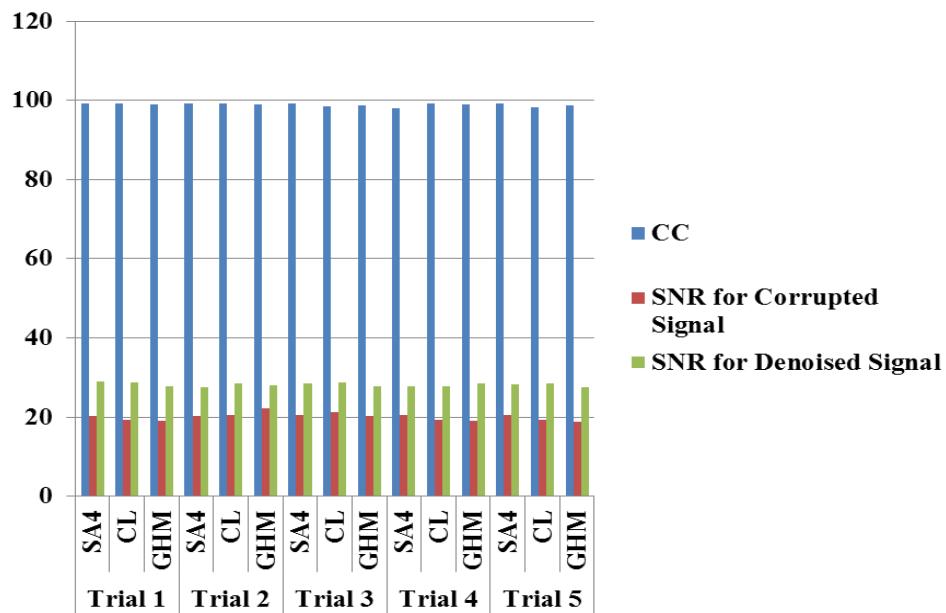


**Figure 5.12 Power Spectra Plot**

Signal to Noise ratio and cross correlation is the important technical evaluation parameter in signal. Table 5.2 compares SNR and Cross correlation values of various Multiwavelet filters.

**Table 5.2 Summary of five trials Artifacts removal results (CC in percent)**

Trials	Multiwavelet filters	CC	SNR(dB)	
			Corrupted Signal	Denoised Signal
1	SA4	99.13	20.03	28.9
	CL	99.20	19.2	28.7
	GHM	98.98	19.0	27.9
2	SA4	99.11	20.20	27.6
	CL	99.26	20.05	28.4
	GHM	98.99	22.1	28.1
3	SA4	99.23	20.5	28.5
	CL	98.42	21.2	28.8
	GHM	98.78	20.3	27.8
4	SA4	98.13	20.4	27.9
	CL	99.20	19.2	27.9
	GHM	98.98	19.0	28.6
5	SA4	99.16	20.5	28.2
	CL	98.20	19.2	28.4
	GHM	98.69	18.8	27.6



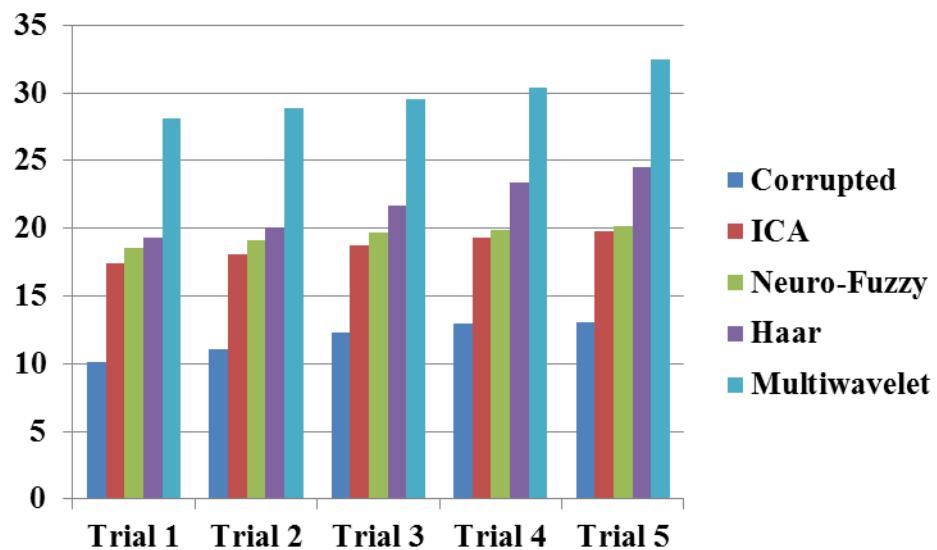
**Figure 5.13 SNR and CC curve for five trial using multiwavelet**

Figure 5.13 shows the SNR and cross correlation of five using multiwavelet transform. From the figure, the cross correlation achieve for denoised signal is nearabout 98%. The SNR value of denoised signal is higher than the corrupted signal.

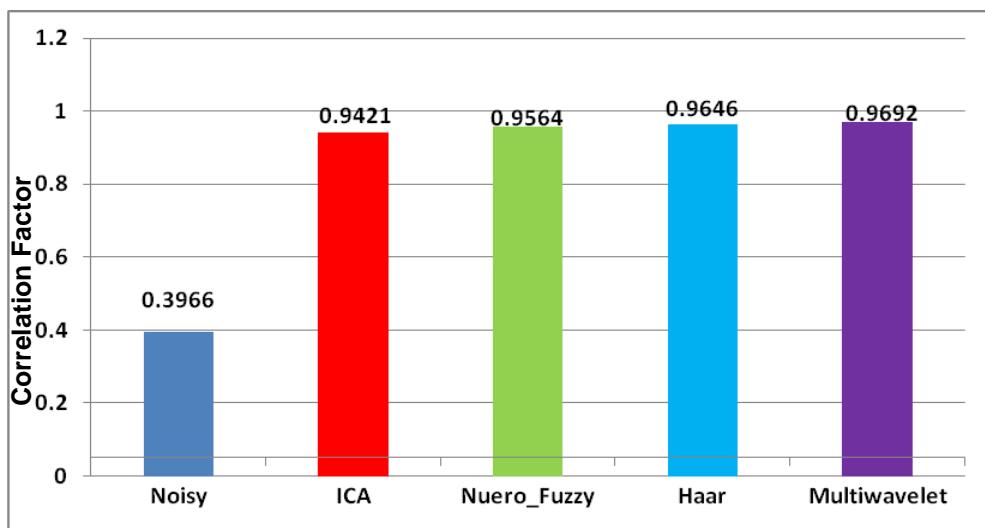
**Table 5.3SNR values for proposed methods**

Trials	Corrupted Signal SNR ( dB)	SNR in dB for proposed Methods			
		Using ICA (IJADE)	Neuro fuzzy Filter	Haar wavelet Transform	Multi wavelet Transform (SA4 Filter)
Trial 1	10.1040	17.4210	18.5040	19.3040	28.1010
Trial 2	11.1020	18.1010	19.1020	20.1050	28.9040
Trial 3	12.3010	18.7020	19.7030	21.7030	29.5040
Trial 4	12.9210	19.3030	19.9040	23.4030	30.4020
Trial 5	13.1020	19.8040	20.2060	24.5040	32.5010

The Table 5.3 shows the SNR value in dB for the proposed methods. The IJADE SNR value is higher than corrupted signal. Neuro-fuzzy filter SNR is higher than IJADE .Haar wavelet SNR is higher than Neuro fuzzy filter. Finally Multiwavelet achieve high SNR value compare with Other methods.

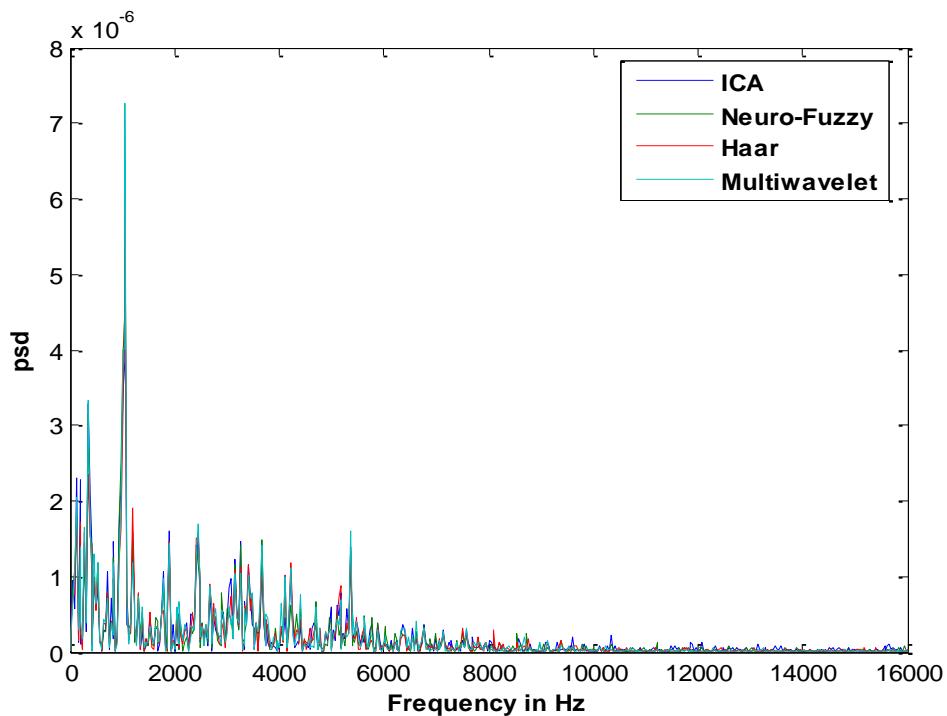


**Figure 5.14** SNR curve for corrupted signal and proposed artifacts removal methods



**Figure 5.15** Correlation plot for noisy signal and proposed artifacts removal methods

Figure 5.15 shows the correlation plot for corrupted signal and various artifacts removal method. This correlation plots gives the relation between noisy and de-noised signal.



**Figure 5.16 Power spectral density plot for proposed methods**

Figure 5.16 shows the power spectral density plot for proposed methods namely IJADE, neuro fuzzy filter, Haar wavelet and multiwavelet transform amongs this methods multiwavelet transform only achieve high power spectral density.

## 5.9 SUMMARY

In this chapter multiwavelets filters are used to remove the artifacts from electroencephalogram (EEG) signals. They are defined using several wavelets with several scaling functions. Multiwavelet has several advantages in comparison with scalar wavelet. The features such as compact support, orthogonality, symmetry and higher order approximation are known to be

important in signal processing. Multiwavelets provide one alternative to the wavelet transform. Multiwavelets are very similar to wavelets but have some important differences. In particular, where as wavelets have an associated scaling function and wavelet function, multiwavelets have two or more scaling and wavelet function. The technique has been checked on five trials of artifacts Signals. Using multiwavelet filters SA4, GHM, CL to remove EOG, artifacts by a new threshold formula is discussed. The proposed Multiwavelet method gives a better result without any complexity and also retains the original information contained in the EEG signal. Power Spectral Density plot, correlation values table & SNR values table are used as performance matrices. Multiwavelet method has outperformed ICA, Neuro fuzzy filter and WT as far as SNR and cross correlation concerned. This chapter conclude that the Statistical method gives lesser complexity and easier to remove the Artifacts with help of Multiwavelet Decomposition. It is an efficient technique for improving the Quality of EEG Signals in Biomedical Analysis.

## CHAPTER 6

### CONCLUSION AND FUTURE WORK

#### 6.1 CONCLUSION

This chapter summarizes the outcome of the research work developed for artifacts removal in EEG. The details of various methods are analysed and present their comparative evaluation of interms of various technical parameters like signal to noise ratio, correlation factor and power spectral density. A combination of approaches using all methods for de-noising evaluation is identified. Also; several directions for research in this area are discussed. Artifacts removal is the process of identifying and removing artifacts from brain signals. An artifacts removal method should be able to remove the artifacts as well as keep related neurological phenomenon intact. The main objective of the artifacts removal is to remove the artifacts from original signal without loss of information. In this research four novel artifacts removal methods and its output performances are discussed. The major contributions of this research work are:

- Improvement in power spectral density of de-noised signal
- High signal to noise ratio value
- Reduction in the time in removal process
- Correlation achievement
- Improvement of information content

The concluding remarks of the each artifacts removal methods are stated as follows.

Chapter 2 presented independent component analysis technique is discussed. Observed from the literature that JADE performs better than the other ICA Algorithms. In this chapter Improved JADE Algorithm is presented. The implementation of the Infomax algorithms, Fast ICA will have adjustable parameters, whereas IJADE had no parameters. Separation quality of Infomax, Extended Infomax and Fast ICA highly depends on tuning of its parameters. Time required for JADE is to separate the source signals is very less when compared with other algorithms, so that JADE is faster in execution for the 7 channel data. IJADE uses cumulant matrices which are equal to square of number of channels, if number of channels are increasing then the cumulants matrices also increases, the diagonalization process of cumulant matrices becomes complex and time required for separation also increases and memory requirement also increases. Infomax algorithm cannot separate the sub-gaussian sources such as line noise; whereas Extended Infomax can do better for all types of sources. The five trial datas are considered for artifacts removal. The Improved JADE obtained results have higher signal to noise ratio value and less execution time compared to the existing technique such as fast ICA, infomax and extended infomax.

Chapter 3 discusses Neuro-fuzzy filter, which are an integration of neural networks and fuzzy logic. The computational process envisioned for neuro-fuzzy systems starts with the development of a “fuzzy neuron” based on the understanding of biological neuronal morphologies, followed by learning mechanisms. This leads the functions of a fuzzy neural computational process. Development of fuzzy neural models motivated by biological neurons. Models of synaptic connections which incorporates fuzziness into neural network. Development of learning algorithms (the method of adjusting

the synaptic weights). In this work Neuro fuzzy filter gives the better Results in terms of signal to Noise ration and correlation factor compare with ICA.

Chapter 4 describes the Haar transform which serves as a prototype for all other wavelet transforms is the simplest of all the transforms. A discrete signal is decomposed into two sub signals of half its length using Haar transform. One of the sub signals denotes the running average of trend while the other denotes the running difference or fluctuation. The selection of threshold and thresholding function plays a vital role in signal denoising .In order to discard the noise coefficients efficiently, a proper thresholding function is essential. The thresholding function can be implemented to retain the wavelet coefficients. The retained the wavelet coefficients represent the denoised signal while the discarded coefficients represent the noise signals. Inverse wavelet transform is implemented to obtain the denoised EEG signal. In this work, Ocular artifact zones are identified by the use of applied wavelet based adaptive thresholding algorithm, which prevents the removal of background EEG information.Haar wavelet transform gives the better result in terms of signal to Noise ratio compare with Neuro fuzzy filter.

Chapter 5 presents maultiwavevelet transform, it has several advantages in comparison with scalar wavelet. The features such as compact support, orthogonally, symmetry, and higher order approximation are known to be important in signal processing. In this method thresholding technique is used for signal de-noising. Decomposing a signal using the wavelet transform, a set of wavelet coefficients that correlates to the high frequency sub bands. These high frequency sub bands consist of the details in the data set. If these details are small enough, they might be omitted without substantially affecting the main features of the data set. The de-noising of EEG signal is carried out by using different combinations of threshold limit, thresholding function and window sizes. Choice of threshold limit and

thresholding function is a crucial step in the de-noising procedure, as it should not remove the original signal coefficients leading to loss of critical information in the analysed data. Because of using this transform the artifacts in the EEG signal could be removed without loss of information. Compare to previous methods, Multiwavelet transform has outperformed improved JADE, Neuro fuzzy filter and wavelet transform as far as SNR and correlation factor concerned.

From the experimental results, the following conclusions can be drawn:

- It can be seen from the experiments that it can successfully separate noise from EEG signals.
- Multiwavelet method has outperformed ICA, Neuro fuzzy filter and Wavelet Transform as far as SNR concerned.
- Multiwavelet method has outperformed ICA, Neuro fuzzy filter and Wavelet Transform as far as correlation factor concerned.
- It can be seen from the experiments the powers of the spectral components have been fully retained.
- Based on these results it can be concluded that Multiwavelet transform has an overall performance which is better than all three methods like IJADE, Haar transform and Neuro fuzzy filter, i.e. Multiwavelet is the most consistent and robust artifacts removal method.

## 6.2 FUTURE ENHANCEMENTS

Some of the suggestions towards extension and/or future related works are identified.

- Adaptive filter algorithm can be used to minimize the impact of noise in EEG signal
- Single objective and multiple objective genetic algorithms can be used to optimize the neuro-fuzzy filter coefficients.
- Multidirectional transforms like curvelet and contourlet transform can be investigated to denoise the EEG signal.

## REFERENCES

1. Abdulhamit Subasi and Ergun Ercelebi " Classification of EEG Signals using Neural Network and Logistic Regression", Computer Methods and Programs in Biomedicine, Vol. 78, pp. 87- 99, 2005.
2. Acharya, D.P. "A review of independent component analysis techniques and their applications", IETE Technical Review, Vol. 25, No. 6, 2008.
3. Ali, H.S. "Fundamentals of Adaptive Filters", Complex systems, 2003.
4. Amari and Cichocki "A new learning algorithm for blind source separation", Advances in Neural Information Processing, MIT press, Vol.8, pp.757-763, 1996.
5. Amble, T. "Logic Programming and Knowledge Engineering", Addison Wesley, Fourier Transform", International Journal of Biological and Life Sciences, Vol. 1, No. 2, pp. 81-89, 2005.
6. Ashish Raj, Akanksha Deo, Mangesh, S., Tomar and Manoj Kumar Bandil "Analysing the Inclusion of Soft Computing Techniques in Denoising EEG Signal", International Journal of Soft Computing and Engineering, Vol. 2, No. 4, 2012.
7. Bell, A.J. and Sejnowski, T.J. "An information- maximization approach to blind separation and blind deconvolution", Neural Computation, Vol.7, pp.1129-1159, 1995.
8. Bell, A.J. and Sejnowski, T.J. "An information-maximization approach to blind separation and blind deconvolution", Neural Computation, vol.7, pp.1129-1159, 1995.
9. Berg, P. and Scherg, M. "Dipole model of eye activity and its application to the removal of eye artifacts from the EEG and MEG," Clinical Physics and Physiological Measurements, Vol. 12, pp.49-54, 1991.
10. Bickford, R.D. "Electroencephalography", Adelman G.ed. Encyclopedia of Neuroscience, Birkhauser, Cambridge USA.

11. Carrie Joyce, A., Irina Gorodnitsky, F. and Marta Kutas "Automatic removal of eye movement and blink artifacts from EEG data using blind component separation", *Psychophysiology*, Vol. 41, No. 2, pp. 313-325, 2009.
12. Cichoki, A. and Vorobyov, S. "Application of ICA for automatic noise and interference cancellation in multisensory biomedical signals" In Proceedings of Second International Workshop on ICA and BSS, pp.621-626, June 2000.
13. Coifman, R.R., Meyer, Y. and Wickerhauser, M.V. "Wavelet analysis and signal processing", *Wavelets and their Applications*, Jones and Bartlett, Boston MA, pp.153–178, 1992.
14. Croft, R.J. and Barry, R.J. "Removal of ocular artifact from the EEG: a review", *Clinical Neurophysiology*, Vol. 30, No. 1, pp. 5-19, 2000.
15. Dong Wei, Hung Ta Pai and Alan Bovik, C. "Antisymmetric biorthogonal coiflets for image coding. Proceedings of the IEEE International Conference on Image Processing", ICIP, October 1998.
16. Fakhreddine Karray, Milad Alemzadeh, Jamil Abou Saleh and Nours Arab "Human-Computer Interaction: Overview on State of the Art", international journal on smart sensing and intelligent systems, Vol. 1, No.1, pp. 13-23, 2008.
17. Garces Correa, A., Laciar, E., Patino, H.D. and Valentinuzzi, M.E. "Artifacts removal from EEG signal using adaptive filters in cascade", 16<sup>th</sup> Argentine bioengineering congress and the 5<sup>th</sup> Conference on Clinical Engineering, Vol. 90, 2007.
18. Garces, A., Laciar, E., Patino, H.D. and Valentinuzzi, M.E. "Artificat removal from EEG signals using adaptive filters in cascade", 16th Argentine Bioengineering Congresss and the 5th Conference on Clinical engineering, Vol. 90, pp.1-10, 2007.
19. García, J., Medel, J. and Guevara, L. "RTFD Description for ARMA Systems", WSEAS International Conferences, Canada, 2007.
20. García, J., Medel, J. and Sánchez "Real-time neuro-fuzzy digital filtering Approach", Computer and Simulation in Modern Science, WSEAS press selected papers, Vol.1, No. 1, pp. 23-28, 2008.

21. Geronimo, J., Hardin and Massopust, P. "Fractal functions and wavelet expansions based on several scaling functions", *J. Approx Theory*, Vol. 78, pp.373–401, 1994.
22. Geronimo, J., Hardin, D. and Massopust, P. "Fractal functions and wavelet expansions based on several scaling functions", *J. Approx. Theory*, Vol.78, pp. 373–401, 1994.
23. Gilbert Strang and Truong Nguyen "Wavelets and Filter Banks", Wellesley-Cambridge Press, Wellesley M.A, first edition, 1996.
24. Girolami and Fyfe "Independent Component Analysis: Principles and Practice", 1997.
25. Hillyard, S.A. and Galambos, R. "Eye-movement artifact in the CNV", *Electroencephalography and Clinical neurophysiology*, Vol. 28, pp.173-182, 1970.
26. Huang, G., Zhu, K. and Siew, C. "Real-Time Learning Capability of Neural Networks", *IEEE Transactions on Neural Networks*, Vol. 17, 2006.
27. Hyvarinen, A., Karhunen, J. and Oja, E. "Independent Component Analysis", Wiley-Interscience, May 2001.
28. Hyvarinen, N. "Survey on Independent component analysis" *Neural Computing Surveys*, Vol.2, pp.94-128, 1999
29. Iman Elyasi and Sadegh Zarmehi "Elimination of noise by Adaptive wavelet threshold", *World academy of science engineering and technology*, Vol. 7, No. 1, pp.142-148, 1997.
30. Ingrid Daubechies "Ten Lectures on Wavelets", SIAM, Philadelphia PA, First Edition, 1992.
31. James, C.J. and Hesse, C.W. "Independent component analysis for biomedical signals", *Physiological Measurement*, Vol. 26, No.1, pp. 1-5, 2005.
32. Jo Yew Tham, Li-Xin Shen, Seng Luan Lee and Hwee Huat Tan, A. "General approach for analysis and application of discrete multiwavelet transforms. Preprint", 1998.

33. Jonathan, R., Wolpaw, Niels Birbaumer, Dennis McFarland, J. Gert Pfurtschelle and Theresa Vaughan, M. "Brain-Computer Interfaces for Communication and Control", Clinical Neurophysiology, Vol. 113, pp. 767-791, 2002.
34. Jung, T.P. and Humphries, C. "Removing electroencephalographic artifacts by blind source separation", Psychophysiology, Vol. 37, pp.163-178, 2000.
35. Jutten, C. and Karhunen, J. "Advances in blind source separation (bss) and independent component analysis (ica) for nonlinear mixtures", Int J. Neural Syst., Vol.14, No. 5, pp. 267–292, October 2004.
36. Krishnaveni, V., Jayaraman, S., Malmurugan, N., Kandaswamy, A. and Ramadoss, K. "Non adaptive thresholding methods for correcting ocular artifacts in EEG", Acad Open Internet Journal, Vol. 13, 2007
37. Krishnaveni, V., Jayaraman, S., Malmurugan, N., Kandasamy, A. and Ramadoss, D. "Non adaptive thresholding methods for correcting ocular artifacts in EEG", Academi Open Internet Journal, Vol.13, 2004.
38. Krishnaveni, V., Jayaraman, S., Aravind, S. and Ramadas, K. "Automatic identification and removal of ocular artifacts from EEG using wavelet transforms", Measurement Science Review, Vol. 6, No.2-4, 2011.
39. Lagerlund, T.D., Sharbrough, F.W. and Busacker, N.E. "Spatial filtering of multichannel electroencephalographic recordings through principal component analysis by singular value decomposition", Clinical Neurophysiology, Vol. 14, No. 1, pp.73– 82, 2009.
40. Lee, T.W. "Blind separation of delayed and convolved sources", pp.758–764, 1997.
41. Lee, T.W. and Sejnowski, T.J. "Independent component analysis for sub-gaussian and super-gaussian mixtures" Proceedings of 4<sup>th</sup> joint symposiums on Neural computations, Vol. 7, pp.132-139, 1997.
42. Lin, C.T. and Lu, Y.C. "A neural fuzzy system with linguistic teaching signal Y.C", IEEE Transactions on Fuzzy Systems, Vol. 3, pp. 169-189, 1995.

43. Lin, C.T. and Lee, C.S.G. "Neural-network-based fuzzy logic control and decision system", IEEE Transactions on Computers, Vol. 40, pp. 1320-1336, 2001.
44. Lin, Y. and Cunningham, G.A. "A new approach to fuzzy-neural system modelling", IEEE Transactions on Fuzzy systems, Vol. 3, pp. 190-198, 1995.
45. Maan Shaker, M. "EEG Waves Classifier using Wavelet Transform Fourier Transform", International Journal of Biological and Life Sciences, Vol. 1, No. 2, 2005.
46. Mackay, D.J.C. "Maximum likelihood and covariant algorithms for independent component analysis", University of Cambridge, London, Tech. Rep, 1996.
47. Murugappan, M., Rizon, M., Nagarajan, R., Yaacob, S., Hazry, D., and Zunaidi, L. "Time-Frequency Analysis of EEG Signals for Human Emotion Detection", Biomed 2008, Proceedings 21, pp. 262–265, 2008.
48. Paulchamy Balaiah and Ilavennila "Comparative Evaluation of Adaptive Filter and Neuro-Fuzzy Filter in Artifacts Removal from Electroencephalogram Signal", American Journal of Applied Sciences, Vol. 9, No. 10, pp. 1583-1593, 2012.
49. Paulchamy, B. and Ila Vennila "A Certain Exploration on EEG Signal for the Removal of Artifacts using power Spectral Density Analysis Through Haar Transform", International Journal of Computer Applications, Vol.42, No.3, 2012.
50. Pierre Comon "Independent Component Analysis: a new concept" ELSEVIER, Signal Processing, Vol. 36, pp. 287-314, 1992.
51. Pierre, C. "Independent Component Analysis: a new concept" ELSEVIER, Signal Processing, Vol. 36, pp.287-314, 1992.
52. Sarah Hosni, M. and Mahmoud Gadallah, E. "Classification of EEG signals using different feature extraction techniques for mental-task BCI" Ain Shams university, Cairo, Egypt.
53. Say Song Goh, Qingtang Jiang and Tao Xia "Construction of biorthogonal multiwavelets using the lifting scheme preprint", 1998.

54. Schlogal, A., Keinrath, C., Zimmermann, D., Scherer, R., Leeb, R. and Pfurtscheller, G. "A fully automated correction method of EOG artifacts in EEG recordings", Clin Neurophysiol, Vol. 118, No. 1, pp. 98-104, 2007.
55. Scott Makeig. "Developments in Neural Human-Systems Interface Design" downloaded from (<Ftp://www.cnl.salk.edu/pub/> ).
56. Senthilkumar, P., Arumuganathan, R., Sivakumar, k. and Vimal, C. "A Wavelet based Statistical Method for De-Noising of Ocular Artifacts in EEG Signals", IJCSNS International Journal of Computer Science and Network Security, Vol.8, No.9, 2008.
57. Senthilkumar, P., Arumuganantham, R., Sivakumar, K. and Vimal, C. "Removal of Artifacts from EEG Signals using Adaptive Filter through Wavelet Transform", ICSP2008 Proceedings.
58. Sornmo, L. and Laguna, P. "Bioelectrical Signal Processing in Cardiac and Neurological", 2005.
59. Stone, J.V. "Independent Component Analysis : A Tutorial Introduction (Bradford Books)" The MIT Press, September 2004.
60. Strela, V. and Walden, A.T, "Orthogonal and biorthogonal multiwavelets for signal denoising and image compression", Proc. SPIE, Vol. 3391, pp. 96–107, 1998.
61. Strela, V., Heller, P.N., Strang, G., Topiwala, P. and Heil, C. "The application of multiwavelet filter banks to image processing. Preprint", 1998.
62. Tao Xia and Qingtang Jiang "Optimal multifilter banks: Design, related symmetric extension transform and application to image compression. Preprint", 1998.
63. Tatjana Zikov, Stephane Bibian, Guy Dumont, A. and Mihai Huzmezan "A wavelet based de-noising technique for ocular artifact correction of the Electroencephalogram", 24th International conference of the IEEE Engineering in Medicine and Biology Society, Huston, Texas, 2010.
64. Teplan, M. "Fundamentals of EEG measurement", Measurement and science review, Vol. 2, pp.1-11, 2002.

65. Vasily Strela, "Multiwavelets: Theory and Applications", PhD thesis, Massachusetts Institute of Technology, 1996.
66. Venkata Ramanan, S., Sahambi, J.S. and Kalpakam, N.V. "A Novel Wavelet Based Technique for Detection and De-Noising of Ocular Artifact in Normal and Epileptic Electroencephalogram", BICS, 2009.
67. Verleger, R. and Gasser, T. "Correction of EOG artifacts in event related potentials of EEG: Aspects of reliability and validity" Psychophysiology, Vol. 19, pp. 472-480, 1982.
68. Verobyov, S. and Cichocki, A. "Blind noise reduction from multisensory signals using ICA and subspace filtering, with application to EEG analysis", Biological Cybernetics, Vol.86, pp. 293-303, 2002.
69. Widrow, B. and Stearns, S.D. "Adaptive Signal Processing", New jersey, Prentice Hall, 1985.
70. Woestengurg, J.C., Verbaten, M.N. and Slangen, J.L. "The removal of the eye movement artifact from the EEG by regression analysis in the frequency domain", Biological Physiology, Vol.16, pp.127-147, 2007
71. Wu, E.H. and Yu, P.L. "Independent component analysis for clustering multivariate time series data", pp.474–482, 2005.
72. Xiang Gen Xia, Jeffrey Geronimo, S. Douglas Hardin, P. and Bruce Suter, W. "Computations of multiwavelet transforms", Proc. SPIE, Vol. 2569, pp. 27–38, 1995.

## LIST OF PUBLICATIONS

### **International Journal**

1. **Paulchamy, B.** and IlaVennila “Qualitative Analysis of ICA Based Algorithms for the Removal of Artifacts from EEG Signals”, in the Journal of Computer Science, Vol.8 No.3, Page: 287-295, 2012.
2. **Paulchamy, B.** and IlaVennila “A Certain Exploration on EEG Signal for the Removal of Artifacts using power Spectral Density Analysis Through Haar Transform”, in the International Journal of Computer Applications, Vol. 42, No. 3, pp. 9-14, 2012.
3. **Paulchamy, B.** and IlaVennila “A proficient Design of Hybrid Synchronous and Asynchronous Digital FIR Filter Using FPGA”, in the International Journal of Computer Science Issues, Vol. 9, No. 1, pp. 395-404, 2012.
4. **Paulchamy, B.** and IlaVennila “Comparative evaluation of Adaptive filter and Neuro fuzzy filter in Artifacts removal from EEG Signal”, in the American Journal of Applied Sciences, Vol.9, No.10, pp. 1583-1593.
5. **Paulchamy, B.** and IlaVennila “A proficient system for preventing and Acknowledging about the drunken drive by analyzing the Neuronal – Activity of the Brain”, in the Journal of Engineering and Applied Science, Vol. 7, No. 8, pp. 1029-1036, 2012.
6. **Paulchamy, B.** and IlaVennila “Certain Investigation on EEG Signals for the Removal of Artifacts Using Soft Computing Technique”, International Journal of Communication and Engineering, Vol.2, pp. 01-10, 2012.

### **Communicated**

1. **Paulchamy, B.** and IlaVennila “Certain Investigation on EEG Signals for the Removal of Artifacts Using Multi wavelet Transform’, in the International Journal of Engineering and Innovative Technology (IJEIT), Vol. 2, 2012. (**Accepted**).

- 1 **Paulchamy, B.** and IlaVennila “A Novel Approach to communicate and to detect Emotion of Differentialy –Abled persons Using Sensors And Electroencephalogram Signal”, in the International Journal of IEEJ Transactions on Electrical and Electronics Engineering, **(Accepted Anexxure-I, International Journal)**
- 2 **Paulchamy, B.** and IlaVennila “Comparative Evaluation of Various Techniques for the Removal of Artifacts from EEG Signal”, in the International Journal of Fuzzy Systems.( **Under Review**)

### **International Conferences**

1. Presented Paper titled “Independent Component Analysis for Separation of Epileptic Seizure Signals”, in the International Conference on Intelligent Systems and control ISCO2010 at Karpagam College of Engineering on 4-5<sup>th</sup> February 2010.
2. Presented Paper titled “Analysis & Separation of Artifacts in Newborn EEG”, in the International Conference on Emerging trends in engineering Technologies-2010 (ICETES-2010) at Nooral Islam University Thuckalay on 25<sup>th</sup> & 26<sup>th</sup> March-2010.

### **National Conference**

1. Presented Paper titled “Image data Noise Cancellation Using Intelligent Softwares” in the National Conference on “System on Chip”, at Oxford Engineering College, Prittaiyur Trichy on 27<sup>th</sup> – 29<sup>th</sup> August 2009.
2. Presented paper titled “Brain Tumor Detection using EM Algorithm”, in the National conference on “Electronics, control and Automation” at SRM University Kattankulathur, Chennai-2013 on 17<sup>th</sup> & 18<sup>th</sup> September 2009.
3. Presented Paper titled “Minimization of Eye Movement Artifacts in EEG Recordings”, in the national conference on “Advances in Communication and Computing” at Karppagam College of Engineering Coimbatore-32 on 18<sup>th</sup> September 2009.
4. Presented paper titled “Detection and Analysis of Seizure in new born EEG”, in the national conference on VLSI Multimedia Communication at R.V.College of Engineering Bangalore on 23<sup>rd</sup> – 24<sup>th</sup> October 2009.

5. Presented paper titled “Removing Electroencephalographic artifacts by Blind Source Separation”, in the National Conference on Network, Computer & Communication (NCNCC,10) at Hindusthan Institute of Technology, held on 7<sup>th</sup> october-2010.
6. Presented paper titled “An improved Algorithm for Enhancement of Image Morphological Functions”, in the National Conference on Industrial Technology(NCIT’11)at Bannari Amman Institute of Technology, held on 4<sup>th</sup> & 5<sup>th</sup> March-2011.

## CURRICULUM VITAE

**B. Paulchamy** was born in 1975. He received his Bachelor's degree in Electronics Communication Engineering from National Engineering College, Kovilpatty, Manonmanium Sundaranar University, in 1997. He received Master's degree in Applied Electronics from PSG College of Technology, Anna University, Chennai, in 2006.

He is currently working as an Assistant Professor in the Department of Electronics and Communication Engineering, Hindusthan Institute of Technology, Coimbatore. His areas of interest include Digital Signal Processing and Digital Image Processing. He has 13 years of experience in teaching and industry.

He has published 9 papers in International Journals and 25 papers in International and National Conferences. He has published a book titled “Digital Signal Processing” along with Dr. J. Jaya.