Nonlinear Non-Gaussian Algorithms for Signal and Image Processing

A THESIS SUBMITTED TO
DEPARTMENT OF ELECTRONIC AND ELECTRICAL ENGINEERING
AND THE COMMITTEE FOR POSTGRADUATE STUDIES
OF THE UNIVERSITY OF STRATHCLYDE
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Gordon Morison

June 2007

Copyright

Declaration

I declare that this thesis embodies my own research work and that it was composed by myself. Where appropriate, I have made acknowledgments to the work of others.

Gordon Morison

"If you think a thing is impossible, you'll make it impossible."

Bruce Lee (1940-1973)

Acknowledgments

I would like to begin first and foremost by thanking Professor Tariq S. Durrani for his supervision and guidance during the duration of this research, and for the confidence he gave me in pursuing my own ideas within the field.

This work was funded by University of Strathclyde and by Epson Scotland Design Centre. For this support I am hugely thankful.

I was greatly aided during the initial stages of this research by Dr. Mounir Ghogho. I would like to thank him for his guidance, and for sharing both his extensive Signal Processing expertise and enthusiasm with me.

During my PhD I had the great honour to visit the Advanced Brain Signal Processing Group at the Brain Science Institute in Riken Japan. For this I wish to thank the Laboratory Head, Professor Andrzej Cichocki for allowing me to experience such an amazing research environment, and for providing some of the diagrams for use within this thesis.

At various times throughout the development of this research I have benefited from the fruit full discussions that take place within the coffee room. For allowing me to participate I must thank Professor John J. Sorraghan, Professor Steve Marshall, Dr. William Nailon, Dr. David Hamilton, Mr. Roy Chapman and Ms Sheila M Forbes.

The PhD process would have been a lonely journey without that various interesting interactions and distractions that arose during the period. For this I wish

to thank everyone in the Signal Processing laboratory. I would like to specifically thank Dr. Gavin Paterson, Dr. Alan Green and Dr. Stefan Martin for their numerous discussions on Signal Processing, Mathematics, Optimization, Neural Networks and Dilbert, generally over a beer or two.

I spent a large amount of my PhD downtime skydiving and I would like to the thank both the staff and customers of Skydive St Andrews for making my time there so enjoyable. Two of the instructors at the centre, Jim Hood and Andrew Hilton have given me a great deal of support both in the Skydiving world and continuing with me PhD, for that I kindly thank them. Specifically, I would like to thank Alan Wilkinson for all the good times we had, especially in TGI Fridays.

During my time back in Academia I have again met some exceptional people. In the Communications Division in Strathclyde I wish to specifically thank Dr. Robert Atkinson for his help reviewing this thesis and his general assistance whenever it was required. I also wish to thank Christos Tachtatzis and Colin Arthur for their friendship during this period. In the Psychology department in Glasgow University I wish to thank Professor Phillipe G. Schyns, Dr. Klaus Kessler and Dr. Marie Smith for inviting me to work with them, and to Dr. Fraser Smith for our numerous interesting chats on ICA, Statistics and Psychology.

I would like to thank both of my parents for their support during the duration of this research work. I would also like to thank my brother Alan for interesting discussions and the occasional online game of Checkers.

Finally, I would like to thank Cat, Mimi and Otis for putting up with all the weekends and evenings I spent writing, for listening to my sagas and for providing me with the motivation to finish this thesis, for that I am eternally grateful.

Gordon Morison

Glasgow, June 2007

To my parents

Abstract

This thesis is initially concerned with solving the Blind Source Separation (BSS) problem. The BSS problem has been found to occur frequently in problems existing in various Scientific and Engineering application areas. The basic idea of the BSS problem is to separate a collection of mixed data into its underlying information components.

To tackle the BSS problem two related methodologies have been utilized extensively throughout the literature. The first approach is by utilizing the statistical technique Independent Component Analysis (ICA). This method utilizes a transformation that maximizes the statistical independence of the mixed data components. The second approach is based on the Approximate Joint Diagonalization (AJD) of a set of target matrices, either the time delayed correlation matrices or matrix slices of the fourth order cumulant tensor. This approximate diagonalization results in matrices which are maximally diagonal. Within this thesis both of the above approaches are utilized within an adaptive gradient descent setting to tackle the BSS problem.

The first contribution within this thesis is the novel application of the Matrix Momentum optimization framework to perform ICA, via the optimization of a Mutual Information based cost function. The algorithm is shown to give Newton like performance with low computational cost.

The second contribution within this thesis is the first application of the Simulta-

neous Perturbation Stochastic Approximation (SPSA) algorithm to jointly diagonalize a set of time delayed correlation matrices.

As a result of the above work it was also found that the SPSA algorithm could also be applied to the problem of Image Registration. Currently one of the most popular methods of solving the Image Registration problem is based on the maximization of the Mutual Information between the images. The final contribution within this thesis is the application of the SPSA algorithm to other novel Information Theoretic cost functions to perform Image Registration.

Contents

List of Figures

List of Tables

List of Symbols

$\mathbf{s}$	Source signal vector
$\mathbf{A}$	Mixing matrix
$\mathbf{x}$	Mixed signal vector
$\mathbf{y}$	Unmixed signals
$\mathbf{W}$	Unmixing matrix
$\mathbf{P}$	Permutation matrix
$\mathbf{D}$	Diagonal scaling matrix
$\mathbf{z}$	Whitened observation vector
$\mathbf{B}$	Whitening matrix
$\mathbf{Q}$	Orthogonal transformation matrix
$\mathbf{u}$	Orthogonally transformed whitened vector
$\mathbf{I}$	Identity matrix
$\mathbf{R}_{ss}$	Source signal covariance matrix
$\mathbf{R}_{xx}$	Mixed signal covariance matrix
$\mathbf{R}_{zz}$	Whitened signal covariance matrix

$\mathbf{R_{uu}}$	Orthogonally transformed whitened signal covariance matrix
μ	Mean value
σ	Variance
κ_3	Third cumulant of a distribution
κ_4	Fourth cumulant of a distribution
$p_x(x)$	Marginal probaility density function
$p_{x,y}(x,y)$	Joint probablity density function
$D(p\|\|q)$	Divergence measure between p and q
$J(\mathbf{W})$	Contrast function
$I(\mathbf{y})$	Mutual Information of vector $\mathbf{y}$
$H(\mathbf{y})$	Joint Entropy of vector $\mathbf{y}$
$H(y)$	Marginal Entropy of scalar y
$\mathbf{W}_{ML}$	Maximum Likelihood estimator for $\mathbf{W}$
$\|\|\mathbf{w}\|\|^2$	Norm of $\mathbf{w}$
$\mathbf{E}$	Global System Matrix
J_c	Jacobian
$\phi(\mathbf{y})$	Nonlinear transformation
$\mathbf{G}$	Riemannian Metric Tensor
S	Euclidean Manifold
$\nabla J(\mathbf{w})$	Derivative of the cost function $J(\mathbf{w})$
$\tilde{\nabla} J(\mathbf{w})$	Natural Gradient of the cost function $J(\mathbf{w})$

$tr(\mathbf{W})$	Trace of the matrix $\mathbf{W}$
β	Momentum parameter
$\mathbf{H}$	Hessian matrix
$\mathbf{R}\{\mathbf{w}\}$	Pearlmutter's Hessian Vector product operator
$\otimes$	Kronecker Product
$g(\mathbf{w})$	Gradient of a cost function
$\hat{g}(\mathbf{w})$	Approximation of the gradient of a cost function
a	SPSA learning rate
c	SPSA perturbation constant
$\odot$	Element by element multiplication
$\mathbf{C}^l$	Set of l matrices
$off(\mathbf{W})$	Measure of the diagonality of the matrix $mathbfW$
t_x	Image translation in the x direction
t_y	Image translation in the y direction
θ	Image rotation parameter
s	Image scaling factor
$D_{Ts}(p\|\|q)$	Tsallis divergence measure
I_{Ts}	Tsallis mutual information measure
H_{Ts}	Tsallis entropy
$D_{Re}(p\|\|q)$	Renyi divergence measure
I_{Re}	Renyi mutual information measure

H_{Re}	Renyi entropy
T	Rigid body transformation parameterized vector
h_n	Hermite polynomial of order n

List of Acronyms

AC-DC Alternating Columns Diagonal Centers

AMUSE Algorithm for Multiple Unknown Signal Extraction

API Amari's Performance Index

BSS Blind Source Separation

CT Computer Tomography

CAT Computed Axial Tomography

CPU Central Processing Unit

EAMUSE Extended Algorithm for Multiple Source Extraction

EASI Equivariant Adaptive Source Separation

EEG Electroencephalogram

EWASOBI Extended Weight Adjusted Second Order Blind Identification

EVD Eigenvalue Decomposition

FDSA Finite Difference Stochastic Approximation

FFDiag Fast Frobenius Diagonalization

fMRI Functional Magnetic Resonance Imaging

FPGA Field Programmable Gate Array

FOBI Fourth-Order Blind Identification

ICA Independent Component Analysis

ISR Interference to Signal Ratio

JADE Joint Approximate Diagonalization of Eigen-matrices

JD Joint Diagonalization

LMS Least Mean Square

MIMO Multiple Input Multiple Output

MEG Magnetoencephalogram

MRI Magnetic Resonance Imaging

PCA Principal Component Analysis

PET Positron-Emission Tomography

QDIAG Quadratic Diagonalization

RMSA Robbins-Monro Stochastic Approximation

SA Stochastic Approximation

SMD Stochastic Meta Descent

SNR Signal to Noise Ratio

SOBI Second Order Blind Identification

SPECT Single Photon Emission Computed Tomography

SPSA Simultaneous Perturbation Stochastic Approximation

SVD Singular Value Decomposition

TDSEP Temporal Decorrelation Source Separation

TITO Two Input Two Output

WASOBI Weight Adjusted Second Order Blind Identification

Chapter 1

Introduction

This chapter gives an introduction to the main topic of this thesis, Blind Signal Processing, and includes a background to the Blind Source Separation (BSS) problem and the statistical technique of Independent Component Analysis (ICA), that are of current interest in the field. Associated algorithms and further details of these problems will be explained further throughout this thesis. An introduction to the Image Registration problem is also given here. This problem can be solved using similar cost functions and algorithms to those used in solving the Blind Source Separation problem.

1.1 Blind Source Separation

Blind Source Separation refers to the problem of separating mixed data into its underlying information components. The problem is often described as the Cocktail Party Problem based upon the remarkable ability of humans to track and attend to an auditory source in a noisy environment, when the source is generated independently by a speech or sound signal. This phenomenon was first studied in 1953 by Cherry [1, 2, 3]. The model for the Cocktail Party Problem

is shown in Figure 1.1. Within Figure 1.1, for the Cocktail Party problem the

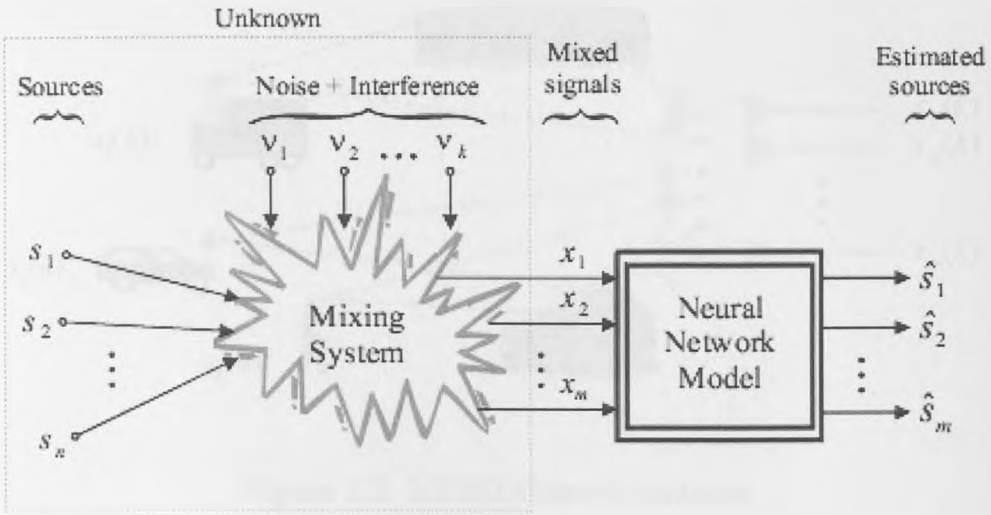


Figure 1.1: The Cocktail Party Problem Model

input sources s_i represent the sound sources and the mixing system represents the room reverberation. The neural network model utilized to create the unmixing system required to select the sound of interest, represents the human brain [3, 4]. Extending the Cocktail Party Problem model to the separation of Multiple Input Multiple Output (MIMO) digital communication signals is shown in Figure 1.2, where in this context the sources to be separated s_i represent the transmission from a handset to a multi-antenna base station, the mixture process arises from the reverberation of the transmitted communication signals with the transmission environment and interference from other transmitting signals. In both of the above examples the source mixtures **s** arriving at the sensors **x** are the result of a convolution between the sources and the transmission environment. For the case where the propagation delays are negligible then this convolutive model reduces to an instantaneous mixture of the source signals **s**, further described in chapter 2. This case arises for example when separating out artifacts from signals of interest in EEG experiments [5, 6, 7, 8], MEG Source Localization [9, 10, 11], Analysis

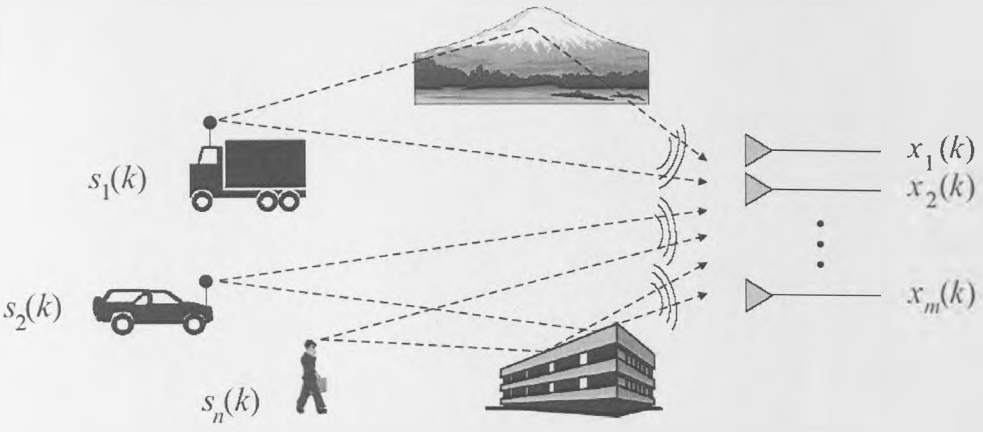


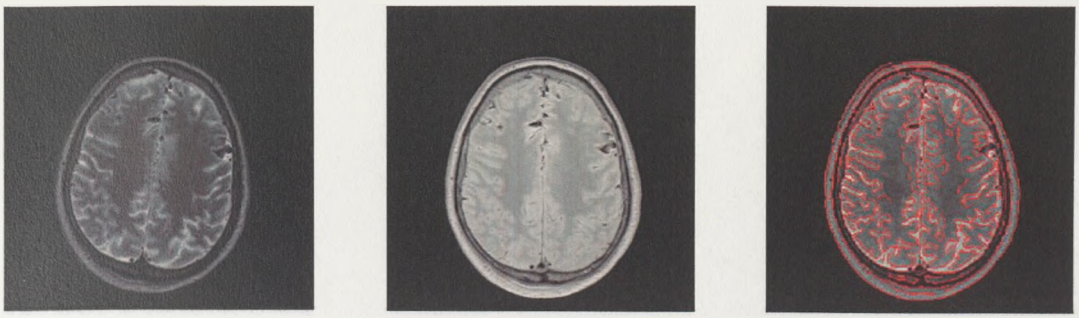
Figure 1.2: MIMO Communications

of fMRI [12, 13], Face Recognition [14, 15, 16, 17] and Image Feature Extraction [18, 19, 20] to name but a few. Therefore due to its widespread application in this thesis only the instantaneous mixing case is considered in detail. The algorithms developed within this thesis could be extended to the convolutive mixing scenario by performing the separation in the Frequency Domain [21, 22, 23], using a Linear Prediction based approach [24, 25] or by using oversampling and row stacking [26, 27, 28, 29, 30]. During the development of the BSS/ICA algorithms detailed further within this thesis it became clear that a number of the algorithms and cost functions utilized within this domain mapped directly to the Image Registration problem. This led to the work described in chapter 6. An Introduction to the Image Registration problem is given in the following section.

1.2 Image Registration

In 1895 Physicist Wilhelm Konrad Roentgen accidentally discovered x-rays. This remarkable discovery led to the birth of the field of medical imaging when Roent-

gen famously took a picture of his wife Frau Roentgen's hand [31]. From this point, x-ray projection radiography became, and continues to this day, to be the most commonly utilized imaging modality for medical diagnosis. In the 1970s there was another revolution in medical imaging with the introduction of x-ray Computer Tomography (CT) also know as Computed Axial Tomography (CAT). Since then there has been numerous advances in both Computational and Engineering methods that have resulted in several new imaging modalities, e.g Positron-Emission Tomography (PET), Single Photon Emission Computed Tomography (SPECT), Ultrasound and Magnetic Resonance Imaging (MRI). The emergence of these new imaging methods provided huge developments in clinical treatments and in medical research. It is known that each of the above modalities has its own individual strengths and weaknesses. As a method of compensating for this, practitioners will often require scans from multiple modalities when developing diagnosis and plans for treatment. This process may then require to be repeated to allow the monitoring of patient medical changes and aid subsequent diagnosis. This potentially requires the clinician to mentally integrate multimodal imagery acquired at multiple time points to extract useful patient information. To aid this process for the clinicians, Image Registration has been applied within the medical domain. The first application of Image Registration within multimodal imagery was developed in [32] where tomographic brain images were utilized for the planning of Radiotherapy treatment. The objective of Image Registration is to develop a spatial transformation that maps homologous points between a pair of images and brings them into correspondence. Most commonly intermodal imagery is applied intra-subject, where different modalities of the same subject are observed, but it is also possible to register inter or intramodal images inter-subject, over multiple subjects. This may be required for example to establish a homeomorphism between the brain images of a group of individuals and a given



(a) Source Image (b) Target Image (c) Registered Image

Figure 1.3: The objective of the registration is to find the corresponding mapping between the Source and Target images

reference image, as a method of reducing the anatomical intersubject variance [33]. To demonstrate Image Registration an example of registering two MRI images is given in Figures 1.3(a), 1.3(b) and 1.3(c). The first image in Figure 1.3(a) is a slice of a T2 contrast MR image of a subject, and the second image is the equivalent slice of a T1 contrast MR image of the same subject. The different contrast weighted MRI scans provide a clinician with additional information, as depending on the contrast some tissues become easier to see with one technique over the other. T1 contrasts are known to provided better anatomical information and provide better distinction between cystic and solid structures, whereas T2 contrast images are known to provide better pathological changes. Within this example, the Tsallis Mutual Information Image Registration algorithm developed in chapter 6 is utilized to register the T2 contrast source image shown in 1.3(a) to the T1 contrast target image shown in 1.3(b). To demonstrate the accuracy of the registration, a Canny edge detection of the target image in Figure 1.3(b) is overlaid upon the registered image. This is shown in Figure 1.3(c). The source and target images utilized in example where obtained from [34]. The above sections have given an overview to the topics that will be developed in further detail within this thesis. In the following an outline of the structure of

the thesis is given.

1.3 Thesis Outline

This chapter provided a simple introduction to the field of Blind Source Separation, the problem and some of the applications of the developed algorithms to real world problems in diverse fields ranging from Mobile Communications to Cognitive Neuroscience. Also provided was an introduction to Image Registration, a topic also discussed within this thesis as during the PhD research it was discovered that the algorithms utilized within these two fields, specifically Information Theoretic based cost functions, often overlap.

Chapter 2 gives a more detailed introduction to solving the Blind Source Separation problem utilizing the statistical technique of Independent Component Analysis. The assumptions required for utilizing the ICA technique are detailed, and it is shown that the second order Principal Component Analysis technique in its direct form is incapable of solving the BSS problem for Non-Gaussian signals. Next the Kullback-Leibler Divergence is introduced as a cost function for optimization within the BSS context. Then, to finish this chapter some of the most commonly utilized ICA techniques are described.

Chapter 3 details the development of the Information Maximization algorithm one of the first and most commonly utilized stochastic gradient descent based ICA algorithms. The cost function originally utilized within this algorithm is the Kullback-Leilber divergence. The full development of the gradient update equation is derived for this neural network based algorithm. During the remainder of this chapter, Amari's Natural Gradient is introduced and is then placed within the context of the BSS problem. The algorithm is shown to provide a dramatic improvement in the convergence properties of the InfoMax algorithm.

Chapter 4 introduces the first application of the Matrix Momentum algorithm within the BSS context. The Matrix Momentum algorithm is a gradient descent based method that utilizes a modified momentum term to create a second order Newton type method without the requirement for a matrix inversion inherent with standard Newton type approaches. The first approach undertaken was to utilize Pearlmutter's Hessian Vector product to develop the momentum term. It was found that this method consistently became unstable. The second approach to avoid this algorithm instability is to utilize the exact Hessian calculation within the algorithm development. This newly developed algorithm is shown to provide fast convergence with low computational complexity when applied within the BSS context.

Chapter 5 introduces Spall's Simultaneous Perturbation Stochastic Approximation algorithm (SPSA) and its application to the BSS problem. Stochastic approximation algorithms are introduced, specifically the Finite Difference Gradient algorithm before the SPSA algorithm is introduced. It is shown that the SPSA algorithm significantly reduces the computational complexity per iteration when compared with the FDSA algorithm. Previous applications of the SPSA algorithm to the BSS problem are discussed. The SPSA algorithm is then utilized to develop a novel matrix joint diagonalization algorithm titled SPSA-JD. The algorithm is shown to provide good performance when applied to diagonalize a set of perfectly diagonalizable matrices. The algorithm is then shown to perform well in the BSS context diagonalizing a set of time delayed correlation matrices.

Chapter 6 utilizes the SPSA algorithm detailed within the previous chapter in application to optimization for solving an Image Registration problem in medical imaging. Image registration is initially introduced, and the Information Theoretic cost functions for optimization via the SPSA algorithm are detailed. This details the Kullback-Liebler, Renyi and Tsallis divergence measures. It is then shown

that for a synthetic Image Registration problem Tsallis divergence measure results in fast and accurate convergence. This represents both the first application of Tsallis divergence and the SPSA optimization framework within medical image registration.

Chapter 7 brings this thesis to its conclusion, giving a summary of the work detailed in the previous chapters. Also within this chapter, further work for continuing the research areas developed is suggested.

1.4 Original Contributions

The original contributions presented in this thesis include the development of novel adaptive gradient descent based algorithms and their application to optimization problems in Blind Source Separation and Image Registration. These algorithms are based on Information Theoretical and Joint Diagonalization based cost functions.

The first contribution developed within this thesis is the application of the Matrix Momentum algorithm to the Blind Source Separation problem. This algorithm represents a Newton based second order gradient descent method without the requirement for the inversion of the Hessian matrix required by straight Newton methods. The algorithm is shown to provide good convergence properties with low computational complexity

The second contribution, is the utilization of the Simultaneous Perturbation Stochastic Approximation gradient descent technique for optimization of a matrix joint diagonalization based cost function. This algorithm is then utilized to diagonalize a number of time delayed correlation matrices for application within the second order based Blind Source Separation problem.

The third and final contribution developed is the application of the Simultane-

ous Perturbation Stochastic Approximation gradient descent technique to Mutual Information based cost functions in Image Registration. Tsallis, Renyi and Shannon's entropy are compared and it is shown that for a synthetic Image Registration problem the combination of the Tsallis Relative Entropy based cost function resulted in the highest performance.

1.5 Publications resulting from this work

Gordon Morison, Tariq Durrani, 'SPSA for Noisy Non-stationary Blind Source Separation' IEEE International Conference on Acoustics, Speech and Signal Processing, Hong Kong 2003 pages 285-288

Gordon Morison, Tariq Durrani, 'Blind Equalization Using Matrix Momentum and Natural Gradient Adaptation', IEEE International Workshop on Neural Networks for Signal Processing, Toulouse, France 2003 pages 439-448

Gordon Morison, Tariq Durrani, 'Blind MIMO Equalization Using Matrix Momentum and Natural Gradient Adaptation', IEE Colloquium on DSP Enabled Radio, Livingston, Scotland, UK 2003

Stephan Martin, Gordon Morison, William Nailon, Tariq Durrani, 'Fast and accurate image registration using Tsallis Entropy and Simultaneous Perturbation Stochastic Approximation', IEE Electronics Letters, Volume 40, Number 10, May 2004 pages 595-597

Chapter 2

Background

In this chapter the essential background for the statistical technique of Independent Component Analysis (ICA) is developed, and the application to the field of Blind Source Separation is described.

2.1 Independent Component Analysis

Independent Component Analysis (ICA) in its most simplistic form aims at decomposing a multivariate data into a linear sum of non-orthogonal basis vectors which have basis coefficients that are maximally statistically independent. The basis vectors and the basis coefficients are learned in an unsupervised manner. The standard ICA model representing an n-dimensional observation vector $\mathbf{x}(k) = [x_1(k), \cdots, x_n(k)]^T$ is generated as follows:

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) \tag{2.1}$$

where $\mathbf{s}(k) = [s_1(k), \cdots, s_n(k)]^T$ is an n-dimensional i.i.d. (independent identically distributed) vector known as *sources*, and $\mathbf{A} \in \mathbb{R}^{n \times n}$ is known as the linear instantaneous *mixing matrix*. The decomposition of the observed vector $\mathbf{x}(k)$ into

maximally statistically independent components is achieved via a linear transformation $\mathbf{W} \in \mathbb{R}^{n \times n}$ in the following manner:

$$\mathbf{y}(k) = \mathbf{W}\mathbf{x}(k) \tag{2.2}$$

where the matrix $\mathbf{W} \in \mathbb{R}^{n \times n}$ represents the inverse of the mixing matrix $\mathbf{A}^{-1}$, and the system output vector $\mathbf{y}(k)$ is an estimate of the independent source vector $\mathbf{s}(k)$. As the linear transformation in ICA is learned in an unsupervised manner the problem would of course be ill-posed if assumptions on the nature of the system were not made. Some of these assumptions have already been mentioned but in the interests of clarity they will be stated explicitly here.

2.1.1 Assumptions

The assumptions made for the standard ICA model for the case of linear instantaneous mixture:

1. The components $s_i(k)$ of the source vector $\mathbf{s}(k)$ are statistically independent.

2. The mixing matrix $\mathbf{A}$ is non-singular and full column rank

3. The observed vector $\mathbf{x}(k)$ is zero mean.

4. At most one of the sources has a Gaussian distribution.

Assumption 2 may be stated as the columns of the matrix are linearly independent. For simplicity it is generally assumed that the mixing matrix is square, although a number of algorithms have been developed that allow for this assumption to be removed [35, 36]. Assumption 3 arises due to the fact that a non zero mean observed random variable x can be written as a combination of a zero

mean stochastic process and a constant factor

$$x = \tilde{x} + c \tag{2.3}$$

where c is a constant. Using the fact that the random variable x and the constant c are independent then c is viewed by a separating system as another independent observed signal, where c is a constant non zero variable. It will be seen later that due to the constant nature of this variable its identifiability would not be possible. Fortunately this may be thought of as a soft assumption as any source vector $\mathbf{x}(k)$ not meeting this condition can be replaced by a centred version of itself as shown:

$$\mathbf{x}(k+1) = \mathbf{x}(k) - E[\mathbf{x}(k)] \tag{2.4}$$

where $E[.]$ represents the expectation operator. The fourth assumption will be explained later in the chapter.

2.1.2 Ambiguities

Taking into account the ICA model and the above assumptions the following ambiguities arise in the model.

1. The variances of the source signals s_i cannot be determined.

2. The mixing matrix $\mathbf{A}$ can only be determined up to a $(n \times n)$ permutation matrix $\mathbf{P}$.

The first ambiguity arises due to the fact that both the source signal $\mathbf{s}(k)$ and the mixing matrix $\mathbf{A}$ are unknown then a fixed scalar ζ between a source signal $\mathbf{s}(k)$ and the corresponding column of the mixing matrix $\mathbf{A}$ does not effect the observation vector $\mathbf{x}(k)$ as shown:

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) = \sum_{i=1}^{n} \frac{\mathbf{a}_i}{\zeta_i} \zeta_i s_i \tag{2.5}$$

where $\mathbf{a}_i$ represents the i-th column of the mixing matrix $\mathbf{A}$. Using this result the source signal variance may be then normalized to unity without loss of generality.

$$E[s_i(k)] = 1 \tag{2.6}$$

The second ambiguity again arises due to the fact that the source signal $\mathbf{s}(k)$ and the mixing matrix $\mathbf{A}$ are unknown therefore any ordering placed on the signals is essentially meaningless. Therefore using the above ambiguities the ICA model may now be written as:

$$\mathbf{y}(k) = \mathbf{W}\mathbf{x}(k) = \mathbf{PDWA}\mathbf{s}(k) = \mathbf{PD}\mathbf{s}(k) \tag{2.7}$$

where as above, $\mathbf{P}$ is a permutation matrix and $\mathbf{D}$ is a diagonal scaling matrix.

2.2 Statistical Independence

The key concept of ICA is the statistical independence assumption of the measured signals. Two scalar random variables x and y are said to be independent if knowledge of the value of one of the random variables gives no information on the value of the other. Statistical independence is defined in terms of the probability density functions of the random variables. Two random variables are said to be statistically independent if their joint probability density function factors into the product of the marginal probability density functions of the random variables. This may be stated mathematically for x and y as follows:

$$p_{x,y}(x, y) = p_x(x)p_y(y) \tag{2.8}$$

where $p(x, y)$ is the joint probability density function of the random variables x and y, and $p_x(x)$ and $p_y(y)$ represent their marginal probability densities. Generalising this result to the vector case:

$$p_{\mathbf{x},\mathbf{y},\mathbf{x},\ldots}(\mathbf{x}, \mathbf{y}, \mathbf{z}, \ldots) = p_{\mathbf{x}}(\mathbf{x})p_{\mathbf{y}}(\mathbf{y})p_{\mathbf{z}}(\mathbf{z}) \ldots \tag{2.9}$$

Referring to the original ICA model Equation 2.1 it may be intuitively seen that after the linear transformation of the source signals $\mathbf{s}(k)$ by the mixing matrix $\mathbf{A}$ the joint probability function of the observation vector $p_{\mathbf{x}}(\mathbf{x})$ will not be equal to the product of the marginal densities $p_x(x_i)$.

$$p_{\mathbf{x}}(\mathbf{x}) \neq \prod_{i=1}^{n} p_x(x_i) \tag{2.10}$$

Also in the standard ICA model no assumptions are made regarding the pdf of the input signals, other than at most one signal is drawn from a Gaussian distribution. Therefore information contained in the signals must be used to develop a linear transformation that maximizes the independence of the signals.

2.3 Principal Component Analysis

Principal Component Analysis (PCA) is a method for removing the second order dependence from a set of observed random variables, this results in a set of variables that are uncorrelated. This may be thought of as a weak form of independence as the resulting random variables are independent up to second order. From that point of view ICA may be thought of as a refinement of PCA, that decorrelates non-Gaussian data for all statistical orders. There are a number of methods available for performing PCA [37, 38]. These methods may generally be split into matrix methods and data type methods. The matrix based methods will utilize all of the available data, the matrix is then decomposed to reveal more detail about the principle directions of the variances. This will involve the diagonalization of the matrix using for example the Householder transformation, Eigenvalue Decomposition (EVD) or Singular Value Decompositions (SVD). For completeness creating a Whitening transformation using the SVD is described within Appendix B due to its widespread use within ICA. The data type methods use the data directly, often in an adaptive manner, such that the principal

component directions are updated as each data sample arrives. Two examples of these methods are using a neural network with Hebbian learning [39, 40, 41] or multilayer perceptron trained using backpropagation [42, 43]. Although it must be noted that the convergence of the adaptive based methods rely heavily on the intelligent selection of the learning parameters for the algorithm. Parameter selection for adaptive algorithms for application to the ICA problem are discussed in chapters 3, 4 and 5 within this thesis. For now it is sufficient to describe the outcome of the PCA process independently of a specific algorithm, making the assumption that the orthogonalization is performed correctly. Utilizing this, the output of the PCA process generates a matrix $\mathbf{B} \in \mathbb{R}^{n \times n}$ that decorrelates the observation vector $\mathbf{x}(k)$. The resulting output from the Whitening process is the uncorrelated vector $\mathbf{z}(k)$ generated as follows:

$$\mathbf{z}(k) = \mathbf{B}\mathbf{x}(k) = \mathbf{B}\mathbf{A}\mathbf{s}(k) \tag{2.11}$$

The resulting covariance matrix of the whitened observations is as follows:

$$\mathbf{R}_{\mathbf{zz}} = E[\mathbf{z}(k)\mathbf{z}(k)^H] = \mathbf{I} \tag{2.12}$$

where $\mathbf{I}$ represents the identity matrix. It can be seen that if an orthogonal transformation $\mathbf{Q}$ is applied to the resulting whitened outputs $\mathbf{z}(k)$, the output of which will be defined as $\mathbf{u}(k)$:

$$\mathbf{u}(k) = \mathbf{Q}\mathbf{z}(k) \tag{2.13}$$

Then the resulting covariance matrix of the newly defined vector $\mathbf{u}(k)$ is as follows:

$$\begin{aligned} \mathbf{R}_{\mathbf{uu}} &= E[\mathbf{Q}\mathbf{z}(k)\mathbf{z}(k)^H\mathbf{Q}^H] \\ &= \mathbf{Q}\mathbf{R}_{\mathbf{zz}}(k)\mathbf{Q}^H \\ &= \mathbf{Q}\mathbf{Q}^H = \mathbf{I} \end{aligned} \tag{2.14}$$

It can be seen from the above that the Whitening transform obtained using PCA can only separate the observation vector up to an orthogonal transformation. This arises as the transformation restores independence only up to second order. An orthogonal transformation has eigenvalues equal to one and has the property that it provides an isometry between two spaces in which distance between points is preserved under the transformation. This may be thought of in a vector space as a rotation without scaling. Therefore, it can be seen that decorrelation is a necessary but not sufficient condition to maximize the independence of the observation vector $\mathbf{x}(k)$. In order to resolve the remaining rotation, further information is required, the orthogonal matrix must be found. This restricts the search space to the space of orthogonal matrices. An orthogonal matrix has $n(n-1)/2$ degrees of freedom, therefore the whitening process reduces the complexity of the ICA problem. In order to resolve the orthogonal ambiguity that is left after the whitening process further information from the signal is required. There are numerous techniques available to resolve this ambiguity, the most popular methods in the field are described in the following sections.

2.3.1 Non-Guassianity

The Central Limit Theorem states that under certain conditions the distribution of the arithmetic mean of a number of independent random variables will tend towards a Gaussian distribution as the number of variables tends to infinity [44]. As the observation vector $\mathbf{x}(k)$ in the Blind Source Separation and ICA problems represents a linear combination of random variables, based on the Central Limit Theorem the distribution of the observation vector $\mathbf{y}(k)$ will tend to a Gaussian distribution. Therefore to obtain independent components at the system output $\mathbf{y}(k)$ then a linear transformation $\mathbf{W}(k)$ is required that results in components

at the output that have a distribution that is maximally far from a Gaussian. That is disributions that have a positive or negative kurtosis, described as super-Gaussian or sub-Gaussian distributions. This is described in further detail in Appendix A.

2.3.2 Non-Gaussian Assumption

The above may be used to explain the fourth assumption that at most one source has a Gaussian distribution. This will be demonstrated for the case of two sensors and two sets of two sources each containing 1000 data samples. The first source vector is generated from a Uniform distribution with zero mean and unit variance, the second source vector is generated from a Gaussian distribution with zero mean and unit variance. The joint probability density function of the above vectors is plotted on the following bidimensional plot know as a scatter diagram [45, 46] shown in Figures 2.1 and 2.2 respectively.

The signals are mixed using the following randomly chosen mixing matrix:

$$\mathbf{A} = \begin{pmatrix} 2 & -3 \\ 2 & -1 \end{pmatrix} \tag{2.15}$$

The resulting joint distributions of the Uniform and Gaussian distributed sources are as shown in Figures 2.3 and 2.4 respectively. Both the above sets of signals are spatially whitened using PCA, specifically using the SVD as described in Appendix B. The resulting joint distributions for the Uniformly distributed sources and the Gaussian distributed sources are shown in Figures 2.5 and 2.6 respectively. As was stated previously in section 2.3 it can be seen clearly for the Uniformly distributed signals in Figure 2.5 that the PCA stage separates the signals up to a rotation. For non-Gaussian signals this rotation can then be resolved by either implicitly or explicitly utilizing the higher order statistics of the observation vector $\mathbf{x}(k)$. Yet for the Gaussian case the distributions are rotationally

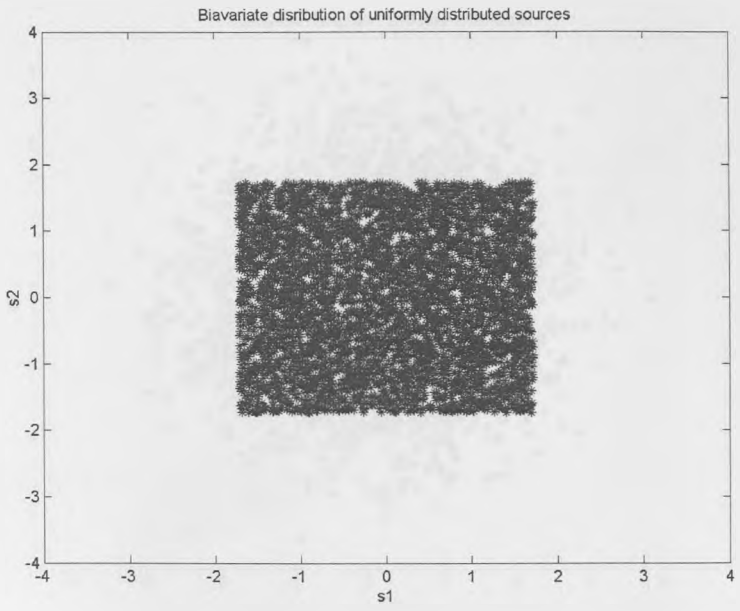


Figure 2.1: Joint distribution of two Uniformly distributed sources

symmetric, therefore an orthogonal mixing matrix has not altered the pdf of the observation vector $\mathbf{x}(k)$ as shown in Figure 2.6. This gives no information as to the nature of the mixing matrix. Unlike non-Gaussian distributions theoretically a Gaussian random variable has zero higher order statistics, that is statistics of order greater than 2, as a Gaussian distribution can be completely characterized by its mean μ and variance σ. An introduction to higher order statistics is given in Appendix A.

2.4 Independence Measures

As discussed in the previous chapter the fundamental assumption in Blind Source Separation is the statistical independence of the input sources. The basis of any source separation algorithm is to restore the statistical independence of the sources at the output of the sensor array. The process of transforming the data at

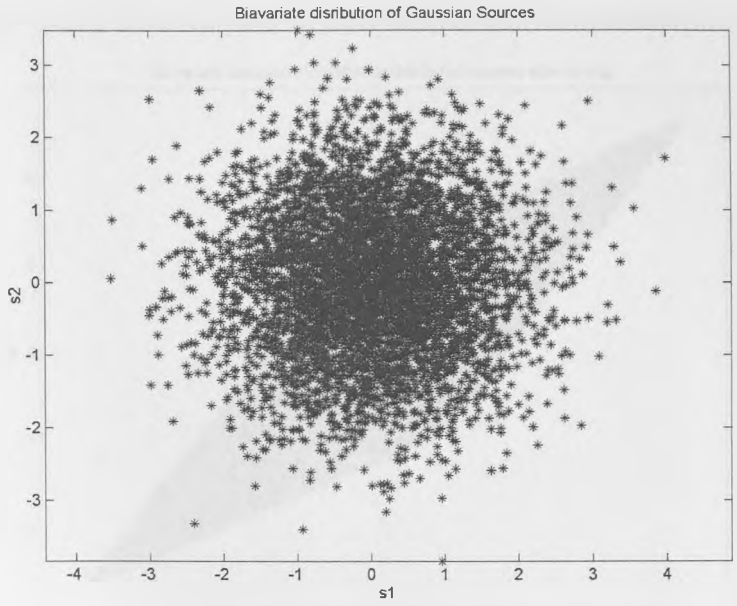


Figure 2.2: Joint distribution of two Gaussian distributed sources

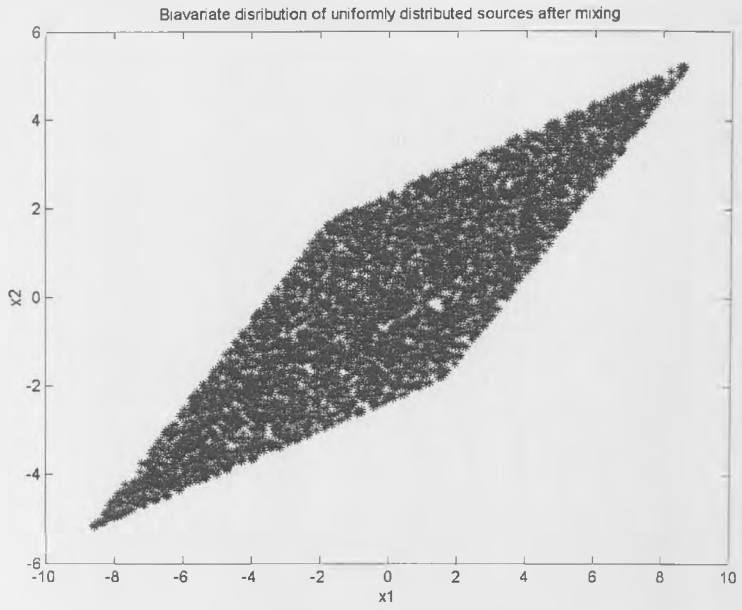
the output of the sensor array into statistically independent sources is known as Independent Component Analysis (ICA) [47, 48]. Independent Component Analysis extends Principal Component Analysis (PCA) by the additional inclusion of higher order statistics. Thus ICA may be defined as a linear decomposition that minimizes the dependence measure between the sources. A number of criterion can be used for measuring the statistical dependence between the output sources. These criterion are often referred to as cost functions or contrasts [48] and their properties are defined in the following sections.

2.5 Contrast Functions

A contrast function first defined in [48] is mapping J from the set of densities $\{p(\mathbf{x})\}$ to $\mathbb{R}$, where the mapping J has the following properties:

- The mapping $J(p(\mathbf{x}))$ stays constant if the components of the vector $\mathbf{x}$ are

Figure 2.3: Joint distribution of two mixed Uniform distributed sources



permuted.

$$J(p(\mathbf{Px})) = J(p(\mathbf{x})), \forall \mathbf{P}$$

where $\mathbf{P}$ is a permutation matrix.

- The mapping $J(p(\mathbf{x}))$ is invariant to changes in scale

$$J(p(\mathbf{Dx})) = J(p(\mathbf{x})), \forall \mathbf{D} \tag{2.16}$$

where $\mathbf{D}$ is a diagonal matrix.

- If the components of $\mathbf{x}$ are independent

$$J(p(\mathbf{Ax})) \geq J(p(\mathbf{x})), \forall \mathbf{A} \tag{2.17}$$

where $\mathbf{A}$ is any invertible matrix. Therefore the minimum value of the contrast function arises when the components of the vector $\mathbf{x}$ are independent.

Some of the most commonly used contrast functions used to generate independent components from a mixture of sources are given in the following subsections.

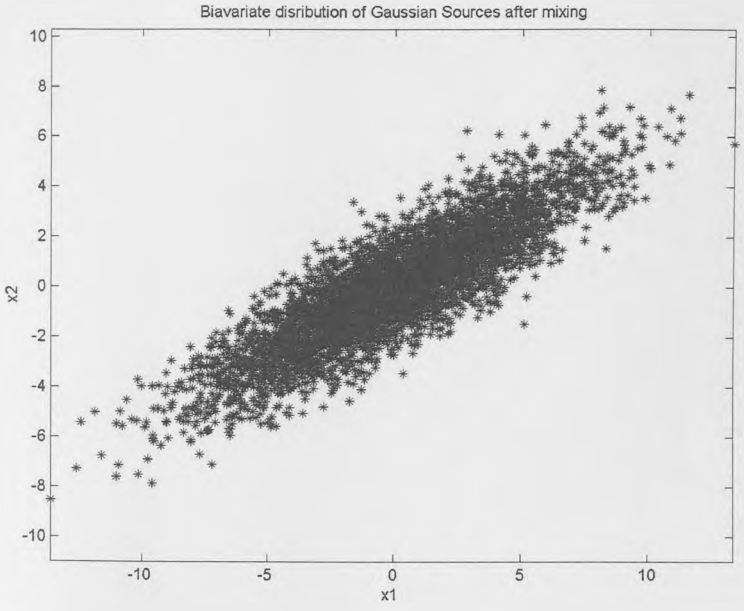


Figure 2.4: Joint distribution of two mixed Gaussian distributed sources

2.5.1 Information Theoretic Independence Measures

A popular class of contrast functions for the Blind Source Separation problem are based upon Shannon's Information Theory [49, 50], specifically the concept of Relative Entropy as the stochastic independence measure. The contrast function used in this case is the Kullback-Leibler divergence (Relative Entropy) [51]. The Kullback-Leibler divergence between two density functions $p(\mathbf{y})$ and $q(\mathbf{y})$ is defined as follows.

$$D(p||q) = \int_{-\infty}^{\infty} p(\mathbf{y}) \log \left(\frac{p(\mathbf{y})}{q(\mathbf{y})} \right) d\mathbf{y} \tag{2.18}$$

This divergence may be thought of as a distance measure between probability density functions, the result of the above equation is always non-negative and zero if and only if the two distributions $p(\mathbf{y})$ and $q(\mathbf{y})$ are the same distribution.

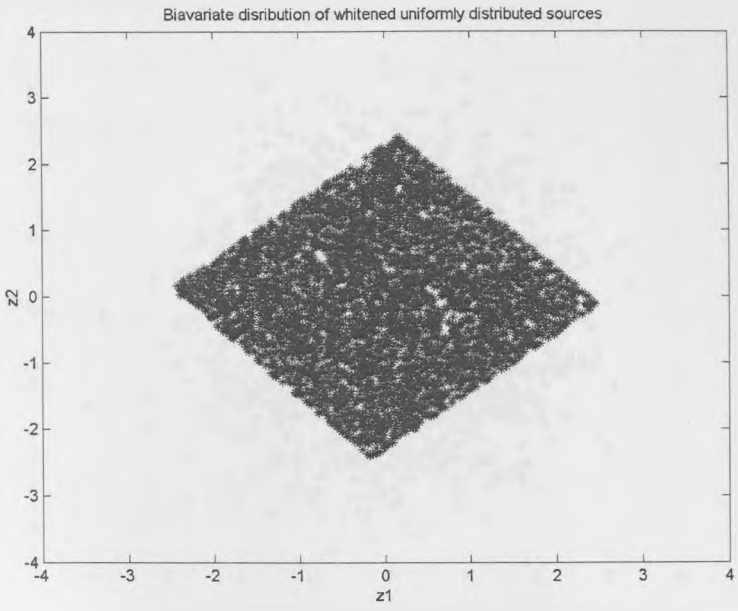


Figure 2.5: Joint distribution of two unmixed Uniform distributed sources

1. The Kullback-Leibler divergence is always non-negative

$$\int_{-\infty}^{\infty} p(\mathbf{y}) \log \left(\frac{p(\mathbf{y})}{q(\mathbf{y})} \right) d\mathbf{y} \geq 0,$$

with equality if and only if

$$p(\mathbf{y}) \equiv q(\mathbf{y})$$

2. The Kullback-Leibler divergence is invariant under an invertible transformation $g(.)$

$$D(p||q) = \int_{-\infty}^{\infty} p(\mathbf{y}) log \left(\frac{p(\mathbf{y})}{q(\mathbf{y})} \right) d\mathbf{y} = \int_{-\infty}^{\infty} p(\mathbf{y}) \log \left(\frac{p(g(\mathbf{y}))}{q(g(\mathbf{y}))} \right) d\mathbf{y} = D(g(p)||g(q))$$

3. The Kullback-Leibler divergence is non-symmetrical.

$$D(p||q) \neq D(q||p)$$

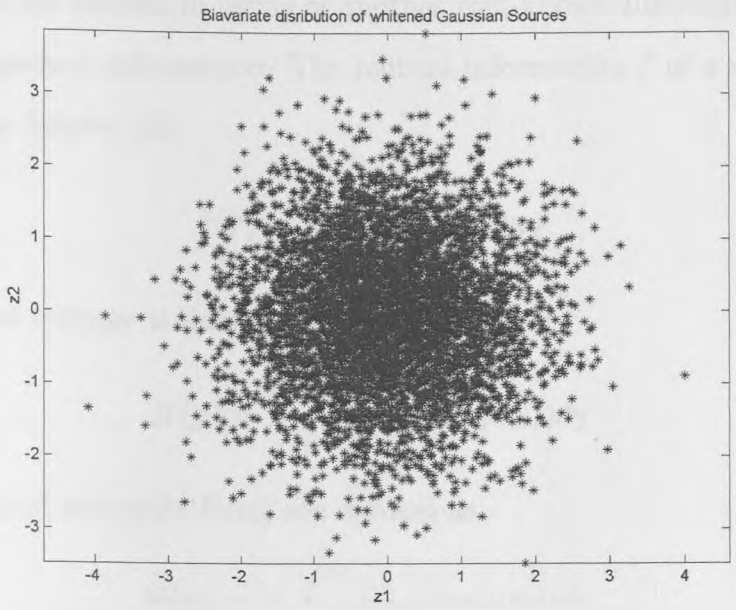


Figure 2.6: Joint distribution of two unmixed Gaussian distributed sources

Although the term distance measure is used this is a slight misnomer, as the Kullback-Leibler divergence is not a true distance measure or metric in the sense of a divergence measure in calculus as it is not a symmetric measure, and therefore does not obey the triangle inequality [37]. A large number of algorithms for the Blind Source Separation problem are based upon either minimization or maximization of Equation 2.18, with the differences stemming from the choice of the distribution $q(\mathbf{y})$. Ideally $q(\mathbf{y})$ would be chosen such as the true distribution of the input sources, but commonly the true distribution may be unknown, thus either a hypothesized distribution is assumed or the distribution is estimated. One approach is the use of a marginalized probability density function. The Kullback-Leibler divergence for an n-dimensional vector $\mathbf{y}$ can then be written as follows:

$$J(\mathbf{W}) = \int_{-\infty}^{\infty} p(\mathbf{y}) log\left(\frac{p(\mathbf{y})}{\prod_{i=1}^{n} p_i(y_i)}\right) d\mathbf{y} \tag{2.19}$$

This may then be written in terms of another well-known Information Theoretic quantity the mutual information. The mutual information I of a random vector $\mathbf{y}$ is defined as follows [52].

$$I(\mathbf{y}) = \sum_{i=1}^{n} H(y_i) - H(\mathbf{y}) \tag{2.20}$$

where the joint entropy is defined as

$$H(\mathbf{y}) = -\int_{-\infty}^{\infty} p(\mathbf{y}) \log(p(\mathbf{y}))d\mathbf{y} \tag{2.21}$$

and the marginal entropies $H(y_i)$ are defined as

$$H(y_i) = -\int_{-\infty}^{\infty} p(y_i) \log(p(y_i))dy_i \tag{2.22}$$

The mutual information has the property that it is non-negative and zero if and only if the random vectors are statistically independent. Therefore the mutual information makes a specifically attractive contrast function for the ICA problem. Using the transformation of random variables [44] and taking into the definition of $\mathbf{y}(k)$ in Equation 2.2, the mutual information may now be written as follows:

$$I(\mathbf{y}) = \sum_{i=1}^{n} H(y_i) - H(\mathbf{x}) - \log|\det \mathbf{W}| \tag{2.23}$$

A closely related information measure to mutual information is Negentropy, this is defined in the following subsection.

2.5.2 Negentropy

It is know from Information Theory that Gaussian random variables have the highest entropy or are the most random of all random variables of equal variance [52, 44]. Using this result entropy can be interpreted as a measure of non-Gaussianity. It was stated in subsection 2.3.1 that independent components have

a distribution that is maximally far from a Gaussian distribution. Negentropy of a random variable is defined as follows:

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y}) \tag{2.24}$$

where $\mathbf{y}_{gauss}$ is a n-dimensional Gaussian random vector with the same mean and covariance as the output vector $\mathbf{y}(k)$ given as follows:

$$p_{gauss}(\mathbf{y}) = \frac{1}{(2\pi)^{\frac{n}{2}} \det(\mathbf{R}_{yy})^{\frac{1}{2}}} exp\left(-\frac{1}{2}\mathbf{y}^T\mathbf{R}_{\mathbf{yy}}^{-1}\mathbf{y}\right) \tag{2.25}$$

where $\mathbf{R}_{\mathbf{yy}} = E[\mathbf{y}\mathbf{y}^T]$ is the covariance matrix of the random vector. Taking into account the inherent scaling ambiguity, the mean and variance of the random vector $\mathbf{y}$ will in practice be normalized to zero mean and unit variance, therefore a Gaussian random vector with zero mean and unit covariance can be used. Negentropy has the following properties inherited from the Kullback-Liebler divergence:

1. It is invariant for any linear invertible transformation.
2. It is zero if and only if the output vector $\mathbf{y}(k)$ has Gaussian distribution.
3. It is always non-negative.

The relationship between mutual information and Negentropy is discussed in the following subsection.

2.5.3 Mutual Information and Negentropy

Comparing Equations 2.20 and 2.24, it can be observed that minimization of the mutual information is equivalent to maximization of the Negentropy. The Negentropy may also be defined as the Kullback-Liebler divergence between the

density of the output vector $\mathbf{y}(k)$ and its normal equivalent both with zero mean and unit variance.

$$J(\mathbf{y}) = \int_{-\infty}^{\infty} p(\mathbf{y}) log\left(\frac{p(\mathbf{y})}{p_{gauss}(\mathbf{y})}\right) d\mathbf{y} \tag{2.26}$$

The Maximum Likelihood contrast function is described in the following subsection.

2.5.4 Maximum Likelihood

Another popular contrast function utilized extensively in Blind Source Separation is based on the principle of Maximum Likelihood estimation. For a given model the probability of a data set as a function of the parameters of the model is termed the likelihood. In the Blind Source Separation situation, the parameters of the model given in Equation 2.1 are the transformation $\mathbf{A}$ and the pdf's of the sources $p(\mathbf{s})$. Therefore the likelihood for the Blind Source Separation problem is given as:

$$p_{\mathbf{x}}(\mathbf{x}|\mathbf{A}, p_{\mathbf{s}}) = \frac{p_s(\mathbf{A}^{-1}\mathbf{x})}{|\det \mathbf{A}|} \tag{2.27}$$

Due to the fact that in the source separation context we are interested in finding a transformation that unmixes the source signals $\mathbf{s}$, then we substitute the demixing parameter $\mathbf{W}$ for $\mathbf{A}^{-1}$ giving the following likelihood:

$$p_{\mathbf{x}}(\mathbf{x}|\mathbf{A}, p_{\mathbf{s}}) = |\det \mathbf{W}| p_s(\mathbf{W}\mathbf{x}) \tag{2.28}$$

Assuming that the estimation will be based upon a set of N i.i.d. samples

$$\mathbf{x}_N = \{\mathbf{x}(1), \ldots, \mathbf{x}(N)\} \tag{2.29}$$

then the pdf of $\mathbf{x}_N$ is given as

$$p_{\mathbf{x}}(\mathbf{x}_N) = \prod_{n=1}^{N} p_{\mathbf{x}}(\mathbf{x}(n)) \tag{2.30}$$

Normalizing and taking the logarithm of the above equation results in the normalized log-likelihood, giving the Maximum Likelihood estimator for $\mathbf{W}$ based upon $\mathbf{x}_N$ as

$$\begin{aligned}\mathbf{W}_{ML} &= \arg\max_{\mathbf{W}} \frac{1}{N} \log \prod_{n=1}^{N} p_{\mathbf{x}}(\mathbf{x}(n)) \\ &= \arg\max_{\mathbf{W}} \frac{1}{N} \sum_{n=1}^{N} \log p_{\mathbf{x}}(\mathbf{x}(n)) \end{aligned} \tag{2.31}$$

It can be seen clearly that the above equation represents the sample average of $\log p_{\mathbf{x}}(\mathbf{x}(n))$. As $N \to \infty$ the above equation may be written as:

$$\begin{aligned}\mathbf{W}_{ML} &= \arg\max_{\mathbf{W}} \int_{-\infty}^{\infty} p_{\mathbf{x}}(\mathbf{x}) \log p_{\mathbf{x}}(\mathbf{x}|\mathbf{W}, p_s) d\mathbf{x} \\ &= \arg\max_{\mathbf{W}} \int_{-\infty}^{\infty} p_{\mathbf{x}}(\mathbf{x}) \log \Big(|\det \mathbf{W}| p_s(\mathbf{W}\mathbf{x}) \Big) d\mathbf{x} \\ &= \arg\max_{\mathbf{W}} \int_{-\infty}^{\infty} p_{\mathbf{x}}(\mathbf{x}) \log p_s(\mathbf{W}\mathbf{x}) d\mathbf{x} + \log |\det \mathbf{W}| \end{aligned} \tag{2.32}$$

The following term can be subtracted from the above equation without altering the likelihood function as it is independent of demixing matrix $\mathbf{W}$

$$\int_{-\infty}^{\infty} p_{\mathbf{x}}(\mathbf{x}) \log p_{\mathbf{x}}(\mathbf{x}) d\mathbf{x} \tag{2.33}$$

Giving the function to be optimized for the Maximum Likelihood solution as:

$$\begin{aligned}\mathbf{W}_{ML} &= \arg\max_{\mathbf{W}} \int_{-\infty}^{\infty} p_{\mathbf{x}}(\mathbf{x}) \log \left(p_s(\mathbf{W}\mathbf{x}) d\mathbf{x} - \log \frac{p_{\mathbf{x}}(\mathbf{x})}{|\det \mathbf{W}|} \right) d\mathbf{x} \\ &= \arg\max_{\mathbf{W}} - \int_{-\infty}^{\infty} p_{\mathbf{x}}(\mathbf{x}) \log \left(\frac{p_{\mathbf{y}}(\mathbf{y})}{p_{\mathbf{s}}(\mathbf{y})} \right) d\mathbf{x} \end{aligned} \tag{2.34}$$

Now that the most common contrast functions to be optimized for ICA have been introduced, some alternative second order methods are described, that when combined with additional assumptions on the data can also be used to solve the Blind Source Separation problem.

2.6 Second Order Methods

So far all of the independence measures described have been based on Information Theoretic ideas, which are based on the assumptions that the signals are non-Gaussian i.i.d random variables, as these have tended to be the more popular approaches to the ICA problem. It was shown in section 2.3 that second order techniques such as PCA can separate sources only up to an orthogonal rotation $\mathbf{Q}$ for the case of i.i.d. sources. If it is known that the sources have a temporal structure, e.g. the sources have non-vanishing correlations and the sources have different power spectrum then the statistical independence condition can be relaxed, and second order statistics can be sufficient to separate the sources and correctly estimate the mixing matrix [53, 54, 55]. Some of the most popular second order blind source separation techniques will be described later in this chapter.

2.7 Approaches to the BSS problem

Having laid the basic foundations for the linear instantaneous blind source separation problem, it is now time to review some of the previous solutions to the problem. Over the last decade there have been numerous differing approaches to the BSS problem arising from the different communities that are currently working in the area. These are classified based on the nature of the approach. During the rest of this chapter some of the most popular algorithms for solving the BSS problem will be described.

2.8 Closed form methods

The statistical signal processing community has also had a huge impact in the development of ICA and approaches to the blind source separation problem. These techniques in general are referred to as batch methods, will normally involve explicit calculation of the higher order statistics of the observation vector $\mathbf{x}(k)$ and are closed form methods. Some of the original techniques developed in this field are described in the following subsections.

2.8.1 Comon's Method

One of the early and seminal works in this field was Comon's minimization of mutual information method [48]. It was in this paper that the idea of contrast functions was introduced and defined. This work extended the well known field of Principal Component Analysis (PCA) with the addition of higher order information. The specific contrast function used in this seminal paper was the maximization of the Negentropy. The algorithm adopted a two stage procedure consisting of a PCA prewhitening stage followed by an orthogonal rotation stage exploiting higher order statistics of the output vector $\mathbf{y}(k)$. For simplicity the two input two output (TITO) scenario will be described first, then the extension to higher dimensionality will be shown. A prewhitening matrix $\mathbf{B}$ is generated by the PCA stage and the resulting whitened output vector is given as follows:

$$\mathbf{z}(k) = \mathbf{B}\mathbf{x}(k) \tag{2.35}$$

For the TITO case, the required orthogonal matrix can be parameterized as a Given's rotation matrix [56]. A Given's rotation matrix is a plane rotation matrix

that can be parameterized as follows:

$$\mathbf{Q} = \begin{pmatrix} cos(\theta) & sin(\theta) \\ -sin(\theta) & cos(\theta) \end{pmatrix} \tag{2.36}$$

Where θ represents the angle of rotation. The contrast function utilized was the Negentropy of the prewhitened vector $\mathbf{z}(k)$. This is given as follows:

$$I(\mathbf{z}(k)) = J(\mathbf{z}(k)) - \sum_{i=1}^{n} J(z_i) + \frac{1}{2} \log \frac{\prod diag(\mathbf{R_{zz}})}{\det(\mathbf{R_{zz}})} \tag{2.37}$$

Where *diag* represents the diagonal elements of the covariance matrix $\mathbf{R_{zz}}$ of the prewhitened vector $\mathbf{z}(k)$. Taking into account the diagonal nature of the covariance matrix $\mathbf{R_{zz}}$ due to the prewhitening stage, the resulting contrast function for the algorithm is given as follows:

$$\begin{aligned} I(\mathbf{z}(k)) &= J(\mathbf{z}(k)) - \sum_{i=1}^{n} J(z_i) + \frac{1}{2} \log \frac{\prod diag(\mathbf{R_{zz}})}{\det(\mathbf{R_{zz}})} \\ &= J(\mathbf{z}(k)) - \sum_{i=1}^{n} J(z_i) \end{aligned} \tag{2.38}$$

The contrast function is now the maximization of the marginal negentropies of the prewhitened vector $\mathbf{z}(k)$. Utilizing the fact that both Negentropy and multivariate entropy are invariant to orthogonal transformation $\mathbf{Q}$ the mutual information for the prewhitened non-Gaussian data may now be written as follows:

$$\begin{aligned} I(\mathbf{y}) &= J(\mathbf{z}) - \sum_{i=1}^{n} J(z_i) \\ &= J(\mathbf{z}) - \sum_{i=1}^{n} J(y_i) \end{aligned} \tag{2.39}$$

By definition this involves the calculation of the marginal probability densities y_i, which are currently unknown and require to be estimated. Comon estimated the required marginal probability densities using an Edgeworth expansion [57] of the system outputs y_i. An Edgeworth expansion as stated in Appendix A is a

series expansion about a Gaussian pdf. Utilizing the Edgeworth expansion the following approximation to the Negentropy is given, where κ_n is the n^th order cumulant of the system output y_i as defined in Appendix A.

$$J(y_i(k)) = \frac{1}{12}(\kappa_3(y_i))^2 + \frac{1}{48}(\kappa_4(y_i))^2 + \frac{7}{48}(\kappa_3(y_i))^4 - \frac{1}{8}(\kappa_3(y_i))^2(\kappa_4(y_i)) \quad (2.40)$$

In generating the system of equations to be solved to find the rotation angle θ, Comon made the assumption that majority of signals naturally occurring will have a symmetric probability density function and therefore the third order statistics need not be considered. This is a commonly made assumption in the ICA and BSS literature. Removing the third order cumulant terms from the above equation, the following equation is given as the contrast function to be optimized.

$$J(y_i(k)) = \frac{1}{48}(\kappa_4(y_i))^2 \quad (2.41)$$

As was stated in the above paragraph, the multivariate entropy $J(\mathbf{z})$ is invariant under an orthogonal transformation, therefore only the above equation is required in the optimization procedure. The contrast function to be maximized for the algorithm can now be written as:

$$I(\mathbf{y}) = -\frac{1}{48}(\kappa_4(\mathbf{y}))^2 \quad (2.42)$$

It turns out that the optimization of the above equation may be carried out by taking the root of a fourth order polynomial, where the unknown variable within the optimization is the angle θ. The above method is applicable only for the TITO case. In order to extend the algorithm to the more useful scenario of more than two signals Comon introduced a pairwise Jacobi like iteration. Comon also details within the paper the extension of the algorithm to complex valued signals. This is of specific importance in the communications domain as a number of commonly used modulation schemes employ complex valued signals

[58]. Comon's algorithm has been extended in [59] to deal with the case that the sources have differing fourth order cumulant signs, and a simplified version of the algorithm was described in [45, 46] where the algorithm was utilized for the extraction of the fetal electrocardiogram from the maternal cutaneous potential recordings. The algorithm was further extended to the case that the third and fourth order cumulant tensor was simultaneously diagonalized in [60, 61, 62]. This algorithm results in a more intuitive cost function for optimization, yet has the limitation that it can only separate real valued signals.

2.8.2 FOBI

Around the time that Comon was working on his method [48], Cardoso had also developed an algebraic method for solving the source separation problem that exploited the fourth order moments of the observation vector $\mathbf{y}(k)$ [63]. This algorithm was creatively titled FOBI (Fourth-Order Blind Identification) by Cardoso. The algorithm utilizes a two step approach in a similar manner to Comon's algorithm, where the first stage of the algorithm is a prewhitening performed using a PCA, as in Comon's method creating the prewhitened output vector $\mathbf{z}(k)$. In the second stage of the algorithm, a quadratically weighted covariance matrix is formed, using a slight abuse of standard notation for a covariance matrix in the following equations:

$$\mathbf{R}_{\mathbf{zz}}(k) = E(|\mathbf{z}(k)|^2 \mathbf{z}(k)\mathbf{z}(k)^T) \tag{2.43}$$

Using the definition of the prewhitened vector $\mathbf{z}(k)$ the above equation may be rewritten as follows:

$$\mathbf{R}_{\mathbf{zz}}(k) = E(|\mathbf{BAs}(k)|^2 \mathbf{BAs}(k)\mathbf{s}(k)^T(\mathbf{BA})^T) \tag{2.44}$$

Utilizing the independence of the input source vector $\mathbf{s}(k)$, and the knowledge that after the whitening transformation $\mathbf{B}$ has been applied the resulting matrix

BA is orthogonal, then the following equation is obtained.

$$\mathbf{R}_{\mathbf{zz}}(k) = \mathbf{W} diag(E[s_i^4(k) + n - 1])\mathbf{W}^T \tag{2.45}$$

The demixing matrix **W** in this instance is equivalent to $\mathbf{BA}^T$ due to the orthogonality after the whitening process **B**. It can be seen that the above equation has the form of an Eigenvalue Decomposition (EVD), the demixing matrix can be obtained from the output of the EVD of Equation 2.45 providing the elements of the diagonal matrix above that represent the kurtosis of the source vector are unique. The FOBI algorithm was extended to operate in the presence of noise by Tong et al. in [64], though one of the problems with the FOBI approach is it can only separate sources that have differing kurtosis values, if this is not the case the algorithm fails completely. To alleviate this problem Cardoso extended the FOBI algorithm to utilize the tensor structure of the fourth order cumulant in [65, 66]. This resulted in one of the most heavily utilized algorithms within the ICA field, the JADE algorithm [67]. This algorithm is described in the following subsection.

2.8.3 JADE

One of the most popular algorithms used in ICA was developed by Cardoso in [67]. The algorithm is known as the JADE algorithm (Joint Approximate Diagonalization of Eigen-matrices) as it utilizes the joint approximate diagonalization algorithm developed by Cardoso in [67] as a method of diagonalizing the fourth order cumulant tensor. The JADE algorithm was initially developed for the separation of complex signals [68, 69] for application to the separation of communications signals. The method is a natural extension of the FOBI algorithm described above. As with the FOBI algorithm above, the JADE algorithm begins with a prewhitening stage, the orthogonal transformation that still remains

is found by diagonalizing the fourth order cumulant tensor. The fourth order cumulant tensor described in Appendix A can be thought of as a four way array, the JADE algorithm performs the separation by utilizing the joint diagonalization algorithm developed by Cardoso and Souloumiac [70] on matrix parallel slices of this array. If all cumulant matrices are utilized for the diagonalization, then JADE equivalently minimizes the following cost function, similar to Comon's cost function detailed above in Equation 2.41.

$$J(\mathbf{W}) = \sum_{ijkl \neq iikl} cum(\mathbf{y}_i \mathbf{y}_j \mathbf{y}_k \mathbf{y}_l)^2 \tag{2.46}$$

In [71] Cardoso gives a comparison of the JADE algorithm with gradient based methods. The JADE algorithm has been utilized extensively in application to the BSS problem [72, 73, 74], although the algorithm is very computationally demanding due to the requirement to calculate the full fourth order cumulant tensor. This is especially problematic in high dimensional spaces. In the following sections some gradient based algorithms are described that provide more computationally efficient approaches to solving the BSS problem.

2.9 Fixed Point Methods

These methods were first proposed by Hyvärinen et al. in their original work [75, 76] and extended in subsequent papers [77, 78, 79, 80, 81]. The original FastICA method operates by finding a single source at a time, then repeating the process to find additional sources. The algorithm to find a single source is described in the following section along with some subsequent extensions to the algorithm.

2.9.1 FastICA

The FastICA method is a fixed point method originally proposed by Hyvärinen in [75, 76]. The optimization criterion described in this paper was the maximization of the kurtosis. As is mentioned in Appendix A, the normalized kurtosis value of a Gaussian variable is zero in the case of infinite samples, thus positive or negative values for kurtosis can be used as a measure of non-Gaussianity. It is known from the Central Limit Theorem that the joint pdf of the output vector $\mathbf{x}$ will tend towards a Gaussian distribution, as a result of the mixing process. Therefore a transformation $\mathbf{W}$ that maximizes the value of the kurtosis will also maximize the deviation from Gaussianity, this will result in independent signals at the output $\mathbf{y}$. Typically the non-Gaussianity is measured as either the squared value or the absolute value of the kurtosis. In order to simplify the optimization space for the algorithm, the data is transformed to be zero mean and then the data is prewhitened. The output of the prewhitening stage is given as in Equation 2.11. Prewhitening the data using PCA as described in Appendix B constrains the vector $\mathbf{w}$ to the unit circle. Thus the norm of the vector $\mathbf{w}$ is given as follows:

$$||\mathbf{w}||^2 = 1 \tag{2.47}$$

In order to find the direction in which the absolute value of the kurtosis traverses gradient techniques are employed. The gradient of the kurtosis of the output vector $\mathbf{y}$ is given as follows:

$$\frac{\partial \kappa_4(\mathbf{y})}{\partial \mathbf{w}} = 4[E(\mathbf{z}(\mathbf{w}^T\mathbf{z})^3)] - 3\mathbf{w}E[(\mathbf{w}^T\mathbf{z})^3] \tag{2.48}$$

The squared value of the kurtosis could also be used in the above equation [75, 48]. The following gradient algorithm can now be constructed that includes the

whiteness constraint reducing the degrees of freedom of the problem.

$$\begin{aligned} \mathbf{w}(k+1) &= \mathbf{w}(k) \pm \mu[E(\mathbf{z}(\mathbf{w}^T\mathbf{z})^3) - 3\mathbf{w}E[(\mathbf{w}^T\mathbf{z})^3] \\ \mathbf{w}(k+1) &= \frac{\mathbf{w}(k+1)}{||\mathbf{w}(k+1)||^2} \end{aligned} \tag{2.49}$$

In order to prevent the weights from possibly converging to the same maxima the output of the system is decorrelated at each iteration. This may be achieved using a Gram-Schmidt orthogonalization [77]. Due to the magnitude and sign ambiguity inherent with the source separation problem, convergence can be determined when the demixing vector $\mathbf{w}$ direction no longer changes or flips in magnitude. Thus convergence can be determined as follows:

$$|\mathbf{w}^T(k-1)\mathbf{w}(k)| \sim 1 \tag{2.50}$$

This allows the introduction of a fixed point algorithm with the absolute or squared value of the kurtosis as the contrast function. In a fixed point algorithm a solution to an equation of the following form is required:

$$y(k+1) = f(y(k)) \tag{2.51}$$

The solution is found by making an initial estimate y, then making the iterative step given in Equation 2.51, and repeating the process until some stopping criterion is met. For the case when the absolute value of the kurtosis is used as the optimization criterion, the following fixed point algorithm was suggested in [75, 78]:

$$\mathbf{w} = E[\mathbf{z}(\mathbf{w}^T\mathbf{z})^3] - 3\mathbf{w} \tag{2.52}$$

This above algorithm introduced fixed point algorithms to the source separation problem, yet using a kurtosis based cost function has the limitations that the estimated kurtosis value is very sensitive to outliers within the data. For some problems, where the underlying source data is bounded this problem is of little

concern (e.g audio or image separation). Yet when the data is potentially unbounded other cost functions measuring non-Gaussianity are preferred. In [82] Hyvärinen changed the cost function of the algorithm to using Negentropy as the optimization criterion to perform the separation, thus escaping the problems associated with kurtosis based algorithms. In [83, 84, 85] the algorithm was extended to complex valued signals. This represented a significant development in the field, as the algorithm could then be applied within the digital communications context. This was tackled by Zarzoso in [86] where the FastICA algorithm based on kurtosis optimization was successfully applied to the MIMO communications. A disadvantage of the FastICA algorithm, is due to the batch nature of the algorithm, a large amount of data must be stored in memory simultaneously. This can be reduced by computing the expectation $E[]$ over a finite number of samples in an online manner, while keeping the columns of the mixing matrix $\mathbf{w}_i$ fixed, updating the mixing matrix columns $\mathbf{w}_i$ once the average has been calculated. The FastICA algorithm to this point represents one of the most highly used algorithms for application to the BSS problem.

2.10 Second Order Methods

In the cases where the sources have non-vanishing temporal correlations then as previously mentioned in section 2.6 it is possible to use alternative separation techniques to methods based either implicitly or explicitly on higher order statistics, like the methods described in the sections above. In these cases it is sufficient to utilize only the second order statistics as the optimization criterion to perform the source separation. A number of algorithms exist in the literature that exploit only second order statistics, the most well published of these algorithms is a second order variant of the JADE algorithm and is described within

this section. To begin of the initial second order methods is described.

2.10.1 AMUSE

The Algorithm for Multiple Unknown Signal Extraction (AMUSE) algorithm was developed in 1991 by Tong et al. [53] and represents one of the first approaches to the BSS problem utilizing only second order statistics of the mixed source signal vector $\mathbf{x}(k)$. The AMUSE algorithm begins by utilizing a Whitening transformation to diagonalize the zero lag covariance matrix of the input vector $\mathbf{x}(k)$, this results in the whitened output vector $\mathbf{z}(k)$. The second stage within the algorithm is to calculate the symmetrized covariance matrix as follows.

$$\mathbf{R}_{\mathbf{zz}} = \frac{1}{2}\left(\mathbf{R}_{\mathbf{zz}} + \mathbf{R}_{\mathbf{zz}}^{T}\right) \tag{2.53}$$

Where the covariance matrix in the above equation is calculated as follows for a given time lag τ.

$$\mathbf{R}_{\mathbf{zz}} = E[\mathbf{z}(k)\mathbf{z}(k-\tau)^{T}] \tag{2.54}$$

The second stage given in Equation 2.53 ensures the symmetry of the covariance matrix in the presence of estimation error. The final stage of the algorithm is to take the Eigenvalue Decomposition of the covariance matrix defined in Equation 2.53, the rows of the separating matrix $\mathbf{W}$ are given as the resulting eigenvectors of this transformation. A similar algorithm to the AMUSE algorithm described was also detailed by Molgedey et al. in [87]. The algorithm was extended by Liang in [54] to allow for the scenario where the additive noise at each of the inputs has a different noise covariance. An extension to the AMUSE algorithm is described in the following subsection which includes diagonalization of multiple covariance matrices at various time delays.

2.10.2 SOBI

The Second Order Blind Identification (SOBI) algorithm was introduced in [55] and is a second order variant of Cardoso's JADE algorithm described previously in subsection 2.8.3, the algorithm represents an extension of Tong's AMUSE algorithm [53]. The algorithm exploits the time coherence of the source signals $s_i(k)$. The algorithm requires the following assumption on the nature of the source signals.

1. $\mathbf{R}_{\mathbf{ss}}(\tau) = E[\mathbf{s}(k+\tau)\mathbf{s}^T(k)] = diag[\rho_1(\tau)\dots\rho_n(\tau)]$

This assumption implies that the sources $s_i(k), 1 \leq i \leq n$ are mutually uncorrelated and $\rho_i(\tau) = E[s_i(k+\tau)s_i(k)]$ represents the auto-covariance of the source $s_i(k)$. The SOBI algorithm calculates the covariance matrices of the input vector $\mathbf{x}(k)$ as follows for multiple time lags τ:

$$\begin{aligned}
\mathbf{R}_{\mathbf{xx}}(\tau) &= E[\mathbf{x}(k)\mathbf{x}(k-\tau)^T] \\
&= E[(\mathbf{A}\mathbf{s}(k))(\mathbf{A}\mathbf{s}(k-\tau))^T] \\
&= \mathbf{A}E[\mathbf{s}(k)\mathbf{s}(k-\tau)^T]\mathbf{A}^T \\
&= \mathbf{A}\mathbf{R}_{\mathbf{ss}}(\tau)\mathbf{A}^T
\end{aligned}$$

From the above assumption the cross correlation terms, that are given by the off-diagonal elements of the covariance matrices $\mathbf{R}_{\mathbf{ss}}(\tau)$ for each time lag τ are zero for independent signals. Hence the demixing matrix $\mathbf{W} = \mathbf{A}^{-1}$ can be found as the solution to a matrix diagonalization problem. That is, to find the matrix that simultaneously jointly diagonalizes the set of covariance matrices $\mathbf{R}_{\mathbf{xx}}(\tau)$. To perform the diagonalization of the above covariance matrices at multiple values of time delay τ, the authors used the joint diagonalization of Cardoso and Souloumiac described in [70]. Joint diagonalization algorithms are discussed in greater detail in chapter 5. As was previously mentioned for the

explicit fourth order based methods, the performance of the SOBI algorithm [55] can be poor if the number of data samples available to the algorithm is small. This arises for small sample sizes, the cross terms of the correlation matrix do not become precisely zero, thus the correlation matrices $\mathbf{R}_{\mathbf{xx}}(\tau)$ may not be exactly jointly diagonalizable. Nonetheless, for the case of small sample sizes the AMUSE and SOBI algorithms can provide an improvement in performance over HOS based methods due to the rapid convergence of second order cumulants to their asymptotic values, when compared to their higher order counterparts. The computational cost for the SOBI algorithm may also be less than required for the explicit HOS based algorithms, yet this is sometimes countered by the large number of correlation matrices required to be jointly diagonalized in order to obtain good convergence from the algorithm, and that the number of matrices must be estimated. Adaptive and Neural network based methods which can be run in both online and offline mode are introduced in the next section and described in greater detail later in this thesis.

2.11 Adaptive filtering and Neural Network based methods

A number of researchers have tackled the BSS problem using neural network approaches [88, 41, 76]. As these techniques are essential to the development of the algorithms described in this thesis, an initial overview of one of the original neural network based techniques is described here while more in-depth description is left for later chapters. The Herault-Jutten network is described in the following subsection.

2.11.1 Herault-Jutten Network

In the mid 80s Herault first began studying the following problem in Computational Neuroscience: How is the central nervous system able to recover or separate the mixtures of signals that are transmitted along the neuronal fibres? This work was first developed within [89, 90]. This work then led onto one of the first approaches to the BSS problem within a Signal Processing context, this was given by Herault and Jutten [47, 91] where a neural network implementation that implicitly introduced higher order moments of the output by cancellation of two non-linear odd functions of the separator output. It was in this paper that the term Independent Component Analysis was coined as a descriptive term for a method used to maximize the statistical independence of a set of mixed sources. This term arises due to the similarity of ICA and Principal Component Analysis (PCA). In the Herault-Jutten paper [47] a recursive neural network as shown

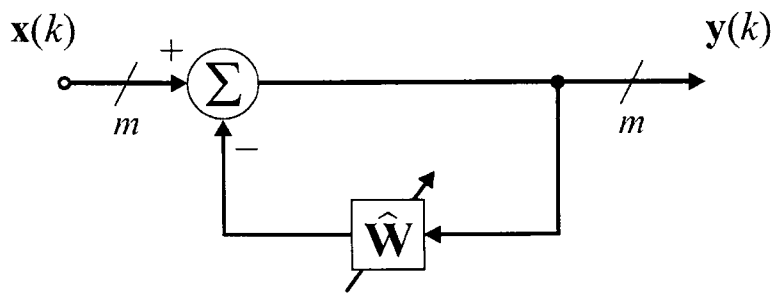


Figure 2.7: Herault-Jutten Recursive Neural Architecture

in Figure 2.7 is utilized where the weights or the network are updated using a steepest descent algorithm. The output of the Herault-Jutten recursive neural network representing the unmixed signals is given as follows.

$$\mathbf{y} = (\mathbf{I} + \mathbf{W})^{-1}\mathbf{x} \tag{2.55}$$

The stochastic gradient equation for the system is given as follows:

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \mu\nabla\mathbf{W} \tag{2.56}$$

Where as stated above the update equation $\nabla\mathbf{W}$ is two non-linear odd functions of the separator output $f(y_i)$ and $g(y_j)$, given as follows.

$$\nabla w_{ij} = f(y_i)g(y_j), i \neq j \tag{2.57}$$

The Herault-Jutten algorithm has been implemented extensively in hardware throughout the literature [92, 93, 94, 95] due to its performance and easy implementation. The algorithm has been further extended to incorporate the case of ill conditioned mixing by Cichocki et al. in [96]. The algorithm has been extended to the convolutive environment in [97, 98], a FPGA implementation of the convolutive extension described in [97] has recently been proposed in [99]. Thus the Herault-Jutten algorithm still remains a popular method within the literature to this day. In the following section some methods for assessing the performance of algorithms in the BSS context is introduced.

2.12 Performance Measures

As a method of comparing the performance of ICA and BSS algorithms a number of measures have been proposed throughout the literature. The most commonly used performance measure is known as Amari's performance measure described in the following subsection. Other measures have been proposed for specific application areas e.g. audio separation in [100], yet these measures are less commonly cited.

2.12.1 Amari's Performance Measure

As had been described throughout the course of this chapter the Blind Source Separation problem is to create a transformation $\mathbf{W}$ that inverts the mixing matrix $\mathbf{A}$. Therefore a natural performance measure for a source separation algorithm is to measure the distance of the global system matrix $\mathbf{E} = \mathbf{WA}$ to the identity matrix. Although this would be ill posed if the ambiguities described in subsection 2.1.2 were not also taken into account. The standard performance measure Amari's Performance Index (API), used for linear instantaneous Blind Source Separation was first give by Amari in [101]. It provides a measure of the distance of the global system matrix $\mathbf{E}$ from the identity matrix incorporating a potential permutation and scaling.

$$API = \sum_{i=1}^{n}\left(\sum_{j=1}^{n}\frac{|e_{ij}|}{max_k|e_{ik}|} - 1\right) + \sum_{j=1}^{n}\left(\sum_{i=1}^{n}\frac{|e_{ij}|}{max_k|e_{ki}|} - 1\right) \tag{2.58}$$

where $\mathbf{E} = (e_{ij}) = \mathbf{WA}$. It can be seen that the above equation has the following two properties.

- $0 \leq API \leq 1$ for all global mixing matrices $\mathbf{E}$
- $API = 0$ if and only if $\mathbf{E} = \mathbf{PD}$ where $\mathbf{P}$ is a permutation matrix and $\mathbf{D}$ a diagonal scaling matrix

Another performance measure that can be used is the signal to noise ratio described in the following subsection.

2.12.2 Signal to Noise Ratio

Another possible separation criteria detailed in [102, 103] is the signal to noise ratio (SNR) of the output sources. This is given by the following equation:

$$SNR = 10\log_{10}\left(\frac{E[\mathbf{s}(t)^2]}{E[\mathbf{n}(t)^2]}\right) \tag{2.59}$$

Where $\mathbf{n}(t) = \mathbf{y}(t) - \mathbf{s}(t)$, with the permutation manually resolved, represents the undesired or noise vector. This performance measure assumes that the permutation ambiguity has been solved before its application.

2.12.3 Gradient Norm

As the algorithms detailed within this thesis concentrate on the gradient based approaches to the BSS problem then the norm of the gradient of the cost function is a commonly used performance measure. This is based upon the idea that for a convex cost function the usual stopping criterion for an optimization algorithm is the nullity of the gradient, in other words the required solution is found once the gradient vanishes or decreases beyond a given threshold. This is given as follows for a given BSS cost function $J(\mathbf{W})$

$$\left|\left|\frac{\partial J(\mathbf{W})}{\partial \mathbf{W}}\right|\right|^2 \leq \epsilon \tag{2.60}$$

Where in the above context, the value ϵ is a suitably chosen small value in the vicinity of zero. This performance metric will be used for a number of the algorithms developed within this thesis. Another important factor utilized in the selection of signal processing algorithms is the computational complexity of a chosen algorithm. This is discussed in the following subsection.

2.12.4 Computational Complexity

Computational complexity represents an important consideration in algorithm development as more and more devices are mobile, relying on low power usage. Yet this, although not exclusively, is predominantly based upon implementation and hardware considerations. A number of authors have utilized FLOPS (MATLAB floating point operations) as a method of assessing the computational

complexity of their algorithms, yet this method has been subsequently made redundant. Therefore, where appropriate within this thesis, purely for a measure of relative comparison the computational time consumption on a consistent microprocessor architecture will be utilized.

2.13 Summary

This chapter has given an overview of the basic theory required for the Blind Source Separation problem, and has given a review of the current state of the art methods, specifically concentrating on the instantaneous linear mixture case for as many sensors as sources. As the Natural Gradient algorithm plays an important part in the development of this thesis a more detailed description of the algorithm, its development and extensions is given in the following chapter.

Chapter 3

Information Maximization

3.1 Introduction

The Information Maximization algorithm (InfoMax) has been one of the most influential algorithms in solutions to the BSS problem. The algorithm was first introduced in [104, 105] by Bell and Sejnowski. An example of the setup for the algorithm is shown in Figure 3.1. In this paper the chosen cost function

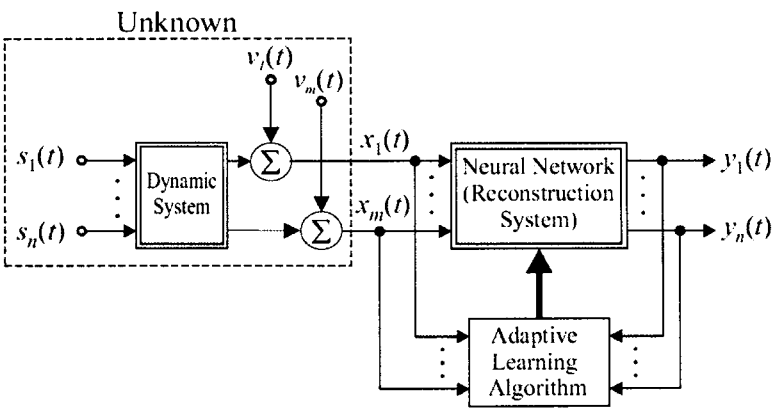


Figure 3.1: Neural Architecture for the BSS problem

$J(\mathbf{W})$ was the maximization of the information between the input and the output

of a feedforward neural network. Mutual Information provides a measure of the amount of information one variable tells about another. Thus, utilizing the definition for the mutual information defined in Equation 2.20 for the input vector $\mathbf{x}$ and the output vector $\mathbf{y}$ we obtain the following Equation.

$$I(\mathbf{y},\mathbf{x}) = H(\mathbf{y}) + H(\mathbf{x}) - H(\mathbf{y},\mathbf{x}) \tag{3.1}$$

Within this equation the entropy at the neural network output $H(\mathbf{y})$ represents the uncertainty at the output $\mathbf{y}$ that can be explained by $\mathbf{x}$, which is the mutual information $I(\mathbf{y},\mathbf{x})$, and the uncertainly at the output $\mathbf{y}$ that cannot be explained by the input $\mathbf{x}$. This can be written as $H(\mathbf{y}|\mathbf{x})$. Thus an alternative definition for the mutual information between the input and output is obtained [52, 104, 105].

$$I(\mathbf{y},\mathbf{x}) = H(\mathbf{y}) - H(\mathbf{y}|\mathbf{x}) \tag{3.2}$$

The second term in Equation 3.2 is a noise term since is represents the information at the output that is not related to the input. Thus in the no noise case the term $H(\mathbf{y}|\mathbf{x}) = 0$. Therefore when the weight matrix of the feedforward neural network represents an invertible continuous deterministic mapping, the maximization of the above equation for the mutual information with respect to the weight matrix of the feedforward neural network $\mathbf{W}$ is equivalent to the maximization of the entropy at the output. This is given as follows:

$$\frac{\partial}{\partial \mathbf{W}} I(\mathbf{y},\mathbf{x}) = \frac{\partial}{\partial \mathbf{W}} H(\mathbf{y}) \tag{3.3}$$

Straight maximization would be inappropriate as the entropy of the output of the demixing system $\mathbf{y} = \mathbf{W}\mathbf{x}$ would tend to infinity for arbitrarily large demixing matrices $\mathbf{W}$ [106]. Therefore to perform separation the output data $\mathbf{y}$ is transformed via a nonlinear transformation $\phi(\mathbf{y})$, that acts component-wise on $\mathbf{y}$. For ease of notational simplicity and to remain consistent with notation used for the

mixing and demixing system throughout this thesis the output from the demixing system will be defined as

$$\mathbf{u} = \mathbf{W}\mathbf{x} \tag{3.4}$$

Therefore allowing the standard output vector $\mathbf{y}$ to be written as

$$\mathbf{y} = \phi(\mathbf{u}) = \phi(\mathbf{W}\mathbf{x}) \tag{3.5}$$

Thus the cost function maximized with respect to the demixing matrix W used in the Bell and Sejnowski algorithm is as follows:

$$\frac{\partial}{\partial \mathbf{W}} I(\mathbf{y}, \mathbf{x}) = \frac{\partial}{\partial \mathbf{W}} H(\mathbf{y}) = \frac{\partial}{\partial \mathbf{W}} H(\phi(\mathbf{u})) \tag{3.6}$$

When the vector $\mathbf{u}$ is transferred through the nonlinear transformation $\phi(\mathbf{u})$ maximization of the mutual information and therefore the maximization of the entropy at the output vector $\mathbf{y}$ is achieved when high density parts of the pdf of $\mathbf{x}$ are aligned with sloping parts of the function $\phi(\mathbf{u})$. When the transformation $\phi(\mathbf{u})$ represents a monotonically increasing invertible function that maps the input to the interval [0,1]. The transformation is also known as a squashing function in neural network literature [43] and has the requirement that the function has a unique inverse. If the transformation $\phi(\mathbf{u})$ meets the above criterion then the linear transformation $\mathbf{W}$ performed on the observation vector $\mathbf{x}$ results in a transformation of the probability density function [44]. This is given as follows:

$$p(\mathbf{y}) = \frac{p(\mathbf{x})}{|J_c|} \tag{3.7}$$

where $|J_c|$ is the absolute value of the Jacobian of the transformation. The Jacobian is defined as the determinant of the matrix of partial derivatives given as follows:

$$J_c = \det \begin{pmatrix} \frac{\partial y_1}{\partial x_1} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_n}{\partial x_1} & \cdots & \frac{\partial y_n}{\partial y_n} \end{pmatrix} \tag{3.8}$$

For an invertible transformation $\phi(\mathbf{u})$ the Jacobian may be written as:

$$J_c = \det(\mathbf{W}) \prod_{i=1}^{n} \frac{\partial y_i}{\partial u_i} \tag{3.9}$$

As was described in chapter 2 the joint entropy of the output vector $\mathbf{y}$ is given as follows:

$$\begin{aligned} H(\mathbf{y}) &= -\int_{-\infty}^{\infty} p(\mathbf{y}) \log(p(\mathbf{y})) d\mathbf{y} \\ &= E[-\log(p(\mathbf{y}))] \end{aligned} \tag{3.10}$$

Using this definition in the maximization of the mutual information the following analysis for the maximization of the cost function $J(\mathbf{W})$ is obtained:

$$\begin{aligned} \frac{\partial J(\mathbf{W})}{\partial \mathbf{W}} &= \frac{\partial}{\partial \mathbf{W}} I(\mathbf{y}, \mathbf{x}) = \frac{\partial}{\partial \mathbf{W}} E[-\log(p(\mathbf{y}))] \\ &= -\frac{\partial}{\partial \mathbf{W}} E\left[\log\left(\frac{p(\mathbf{x})}{|J_c|} \right) \right] \\ &= -\frac{\partial}{\partial \mathbf{W}} E\left[\log(p(\mathbf{x})) - \log|J_c| \right] \\ &= \frac{\partial}{\partial \mathbf{W}} E[\log|J_c|] \\ &= \frac{\partial}{\partial \mathbf{W}} E\left[\log\left(\det(\mathbf{W}) \prod_{i=1}^{n} \frac{\partial y_i}{\partial u_i} \right) \right] \\ &= E\left[\frac{\partial}{\partial \mathbf{W}} \log(\det(\mathbf{W})) + \frac{\partial}{\partial \mathbf{W}} \sum_{i=1}^{n} \log\left(\frac{\partial y_i}{\partial u_i} \right) \right] \end{aligned}$$

In order to calculate the derivative in Equation 3.14 it is required to choose the nonlinearity $\phi(.)$. The nonlinear transformation $\phi(\mathbf{u})$ should be picked such that the transformation matches as closely as possible the cumulative distribution function (cdf) of the input in an attempt to match the pdf, therefore the transformation should ideally be given as:

$$\phi(\mathbf{u}) \simeq \int_{-\infty}^{u} p(u) du \tag{3.11}$$

In the blind case this is not completely possible as the underlying pdf's of the input data are assumed to be unknown. Therefore an approximation of the pdf

of the data is made, obviously at this point if a priori information is known about the data then it can be included to provide a better solution, this idea will be expanded later within this thesis. In the Bell and Sejnowski paper [104] the logistic sigmoidal function was used.

$$\phi(\mathbf{u}) = \frac{1}{1+e^{-\mathbf{u}}} \tag{3.12}$$

This function has the advantage that its derivative has a simple form, this is given as follows:

$$\frac{\partial}{\partial \mathbf{u}}\left(\frac{1}{1+e^{-\mathbf{u}}}\right) = \frac{\partial}{\partial \mathbf{u}}\phi(\mathbf{u}) = \phi(\mathbf{u})(1-\phi(\mathbf{u})) \tag{3.13}$$

Utilizing the logistic function given in Equation 3.12 and its derivative given above within Equation 3.13 the following cost function is obtained[1].

$$\begin{aligned}
\frac{\partial J(\mathbf{W})}{\partial \mathbf{W}} &= E\left[\frac{\partial}{\partial \mathbf{W}}\log(\det(\mathbf{W})) + \frac{\partial}{\partial \mathbf{W}}\sum_{i=1}^{n}\log\left(\frac{\partial y_i}{\partial u_i}\right)\right] \\
&= E\left[\mathbf{W}^{-T} + \frac{\partial}{\partial \mathbf{W}}\log\Big(\phi(\mathbf{y})(1-(\phi(\mathbf{y})))\Big)\right] \\
&= E\left[\mathbf{W}^{-T} + \Big(\phi(\mathbf{y})(1-(\phi(\mathbf{y})))\Big)^{-1}\frac{\partial}{\partial \mathbf{W}}\Big(\Big(\phi(\mathbf{y})(1-(\phi(\mathbf{y})))\Big)\Big)\right] \\
&= E\left[\mathbf{W}^{-T} + (1-2\phi(\mathbf{y}))\mathbf{x}^T\right]
\end{aligned} \tag{3.14}$$

In the paper this cost function is maximized via the standard Steepest Descent algorithm [108, 109]. The following update equation is developed for the Bell and Sejnowski algorithm[2] [104, 105].

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \mu[\mathbf{W}^{-T} - (1-2\phi(\mathbf{y}))\mathbf{x}^T] \tag{3.15}$$

[1]The derivative of the determinant with respect to a matrix is calculated using Jacobi's Formula [107]

[2]As is common notation in the adaptive ICA literature the k index is only used for the first two terms of the update equation, this format is used within this thesis unless the index is required for clarity

In the derivation of the algorithm given in [104] additional unit inputs were included that facilitate dealing with biased inputs, as this has largely been ignored in the literature, they have been omitted here and as the data can be centred to be zero mean, the function $\phi(.)$ and the expected value of the data will be zero mean rendering any bias term superfluous. Of specific importance in the above algorithm is the nonlinearity at the output of the neural implementation. This will be developed further in the following subsection.

3.1.1 Maximum Entropy

In the InfoMax algorithm the nonlinearity $\phi()$ plays an essential part in the minimization of the mutual information required to perform the separation of the mixed source vector $\mathbf{x}$, it will be shown in the following subsection that the maximization of the entropy between the input and output of the network is equivalent to the minimization of the mutual information. From an information theoretic point of view this is explained initially by observing the joint entropy of the output vector $\mathbf{y}$, rearranging the equation for the mutual information given in Equation 2.20, the joint entropy at the output of the network is given as follows:

$$H(\mathbf{y}) = \sum_{i=1}^{n} H(y_i) - I(\mathbf{y}) \tag{3.16}$$

The marginal entropy as shown in Equation 2.22 can be written as an expectation as follows:

$$\begin{aligned} H(y_i) &= -\int_{-\infty}^{\infty} p(y_i)\log(p(y_i))dy_i \\ &= -E[\log(p(y_i))] \end{aligned} \tag{3.17}$$

Using the transformation of random variables for the nonlinearity $\phi()$ that has been previously introduced [44] in section 3.1, the marginal entropy is written

$$H(y_i) = -E\left[\log\left(\frac{p(u_i)}{|\frac{\partial y_i}{\partial u_i}|}\right)\right] \tag{3.18}$$

Taking the derivative of the joint entropy $H(\mathbf{y})$ with respect to the demixing matrix $\mathbf{W}$ gives the following:

$$\frac{\partial H(\mathbf{y})}{\partial \mathbf{W}} = -\frac{\partial}{\partial \mathbf{W}} \sum_{i=1}^{n} E\left[\log \left(\frac{p(u_i)}{|\frac{\partial y_i}{\partial u_i}|} \right) \right] - \frac{\partial}{\partial \mathbf{W}}(I(\mathbf{y})) \tag{3.19}$$

It can be seen from the above equation that there exists a relationship between the maximization of the joint entropy between the input and the output of the neural network and the minimization of the output components $y_i = \phi(u_i)$. If the nonlinear activation function $\phi()$ is chosen such that it matches the cumulative distribution function of the input u_i, then the first term in Equation 3.19 is equal to zero, and the maximization of the entropy between the input and output of the neural network is equal to the minimization of the mutual information between the output components. In the context of the BSS problem the exact pdf of the inputs is generally unknown, therefore the above term acts as an error term. In the next subsection it is shown that the InfoMax algorithm described above is in fact equivalent to the Maximum Likelihood contrast function [106].

3.1.2 Maximum Likelihood

In subsection 2.5.4 the Maximum Likelihood principle was introduced as a potential cost function for use in solving the Blind Source Separation problem. In [106] Cardoso showed that the Maximum Likelihood approach does in fact coincide with the Bell and Sejnowski algorithm developed above in [104] provided that the demixing matrix $\mathbf{W}$ is identified to $\mathbf{A}^{-1}$. In the next section Amari's Natural gradient algorithm is introduced which both improves the convergence and the computational complexity of the InfoMax algorithm.

3.2 Natural Gradient Adaptation

When performing optimization within a manifold a metric is required to give a concept of distance within the manifold. For the case of a Euclidean manifold $S = \{\mathbf{w} \in \mathbb{R}^n\}$ with a cost function $J(\mathbf{w})$ defined on S, $\mathbf{w}$ representing an orthonormal coordinate system, the squared length of a small incremental vector $d\mathbf{w}$ joining $\mathbf{w}$ and $\mathbf{w} + d\mathbf{w}$ is given as follows:

$$|d\mathbf{w}|^2 = d\mathbf{w}^T d\mathbf{w} = \sum_{i=1}^{n} (dw_i)^2 \tag{3.20}$$

where dw_i represents the components of the vector $d\mathbf{w}$. If the coordinate system is not orthonormal or the space represents a Riemannian space S then the above equation is inappropriate and a measure incorporating the local curvature of the space is required. This is given by the following quadratic form:

$$|d\mathbf{w}|^2 = \mathbf{w}^T \mathbf{G} \mathbf{w} = \sum_{i,j} g_{i,j}(\mathbf{w}) dw_i dw_j \tag{3.21}$$

In the above equation the matrix $\mathbf{G}$ is known as the Riemannian metric tensor. This matrix introduces curvature of the cost surface. For a Euclidean manifold the matrix $\mathbf{G}$ reduces simply to the identity matrix $\mathbf{I}$. It can be seen that the above equation represents a weighted distance, induced by a weighted inner product [37]. The above Riemannian metric will be included within the standard Steepest Descent algorithm in the next subsection.

3.2.1 Steepest Descent Directions

The standard Steepest Descent algorithm [108, 110, 109] as utilized in the above InfoMax algorithm [104] utilizes the following strategy. For a cost function $J(\mathbf{w})$ at $\mathbf{w}$, the direction of Steepest Descent is given as the vector dw that minimizes $J(\mathbf{w} + d\mathbf{w})$ under the constraint:

$$|d\mathbf{w}|^2 = \epsilon^2 \tag{3.22}$$

for a sufficiently small ϵ, and an assumed fix step size for the learning algorithm. Based on the above analysis it can be seen clearly that the standard Steepest Descent algorithm is appropriate only in the case where the underlying optimizations space exists within a Euclidean manifold. If this is not the case then the Steepest Descent algorithm may take incorrect descent steps. To improve this Amari introduced the local curvature of the cost surface within the cost function [111]. This is shown in the following subsection.

3.2.2 Natural Gradient Descent

It was shown above that if the underlying optimization manifold is not Euclidean then the ordinary gradient does not in fact represent the direction of steepest descent. If the underlying cost surface is Riemannian, then to obtain the direction of steepest descent the local curvature of the cost surface must be included within the algorithm. This is achieved utilizing the steps detailed in [112, 88], in the following manner, setting the direction vector $d\mathbf{w}$ as follows:

$$d\mathbf{w} = \epsilon \mathbf{a} \tag{3.23}$$

where $\mathbf{a}$ is a vector that satisfies the constraint.

$$||\mathbf{a}||^2 = \mathbf{a}^T \mathbf{G} \mathbf{a} = \sum_{i,j} g_{i,j} \mathbf{a}_i \mathbf{a}_j = 1 \tag{3.24}$$

and $\mathbf{G}$ represents the Riemannian metric tensor introduced in section 3.2. It is required to find $\mathbf{a}$ to minimize

$$J(\mathbf{w} + d\mathbf{w}) = J(\mathbf{w} + \epsilon \mathbf{a}) \tag{3.25}$$

Taking the first two terms of a Taylor series expansion of the above equation, the following equation is obtained:

$$J(\mathbf{w} + \epsilon \mathbf{a}) = J(\mathbf{w}) + \epsilon \nabla J(\mathbf{w})^T \mathbf{a} \tag{3.26}$$

where $\nabla J(\mathbf{w})$ is the standard gradient vector. To minimize the function under the given constraint the following Lagrangian equation is utilized:

$$J(\mathbf{w}+\epsilon\mathbf{a}) = J(\mathbf{w}) + \epsilon\nabla J(\mathbf{w})^T\mathbf{a} - \lambda\mathbf{a}^T\mathbf{G}\mathbf{a} \tag{3.27}$$

Taking the derivative of the above equation and setting the result to zero:

$$\frac{\partial}{\partial a_i}\left\{\epsilon\nabla J(\mathbf{w})^T\mathbf{a} - \frac{\lambda}{2}\mathbf{a}^T\mathbf{G}\mathbf{a}\right\} = 0 \tag{3.28}$$

the following equation is obtained:

$$\nabla J(\mathbf{w}) = 2\lambda\mathbf{G}\mathbf{a} \tag{3.29}$$

solving for $\mathbf{a}$ gives

$$\mathbf{a} = \frac{1}{2\lambda}\mathbf{G}^{-1}\nabla J(\mathbf{w}) \tag{3.30}$$

where λ is chosen to normalize the direction vector $\mathbf{a}$ without changing its direction. The following equation

$$\tilde{\nabla} J(\mathbf{w}) = \mathbf{G}^{-1}\nabla J(\mathbf{w}) \tag{3.31}$$

is know as the Natural Gradient of the cost function J in the Riemannian space representing the direction of steepest descent. Amari suggested in [112, 88] a Natural Gradient descent algorithm of the form

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu\tilde{\nabla} J(\mathbf{w}) \tag{3.32}$$

where as in the standard steepest descent algorithm μ represents a suitably chosen step size. In [101, 113, 114, 112, 88] Amari addressed the BSS problem in the context of Natural Gradient adaptation. This is shown in the following subsection.

3.2.3 BSS via Natural Gradient Adaptation

Amari introduced the concept of the Natural Gradient algorithm within the BSS/ICA problem [101]. The development of the algorithm was based on the fact that the optimization space of the problems takes place within the space of $n \times n$ invertible matrices $\mathbf{W}$. The space of $n \times n$ invertible matrices forms a Lie Group. Some properties of matrix groups, and Lie groups are addressed in the following subsection as a necessary introduction for the application of the Natural Gradient algorithm within the ICA and BSS problems.

3.2.4 Groups Theory

It was stated above that in [101] Amari introduced the Natural Gradient algorithm to the BSS problem. The development of the algorithm is based upon the group structure of the space of $n \times n$ matrices. A group in mathematics, is a non-empty set together with a binary operation that satisfies certain axioms. The axioms for a group G are as follows, where . denotes a binary operation:

1. Closure - $\forall a, b \in G, \quad a.b \in G$
2. Associativity - $\forall a, b, c \in G, \quad a.(b.c) = (a.b).c$
3. Identity $\forall g \in G, \quad e.g = g = g.e$
4. Inverse - $\forall g \in G, \exists h \in G, \quad h.g = e = g.h$

It can be seen clearly that the space of square nonsingular matrices forms a group, when the binary operation is matrix multiplication and the identity for the group is the matrix identity $\mathbf{I}$. It was mentioned above that in the derivation of the Natural Gradient algorithm Amari utilized the Lie Group structure of the space of matrices. Lie groups are defined in the following subsection.

3.2.5 Lie Groups

Lie groups were introduced by Sophus Lie in [115]. A Lie group obeys the same properties as a standard group but has the additional condition that the operations of the group are differentiable. A Lie group represents a differentiable manifold that obeys the group properties [116]. To define the Riemannian metric for the BSS problem it is require to define the following operator $K()$ that maps matrix to a matrix:

$$K(\mathbf{W}) : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{n \times n} \tag{3.33}$$

Utilizing this operator the cost function for the BSS problem may be written as:

$$\tilde{J}(\mathbf{W}) = K(\mathbf{W})J(\mathbf{W}) \tag{3.34}$$

Changing the constraint on the cost function from above slightly for the development of the algorithm, the additional constraint required to achieve equilibrium is added that:

$$E\left[\frac{\partial \tilde{J}(\mathbf{W})}{\partial \mathbf{W}}\right] = 0 \tag{3.35}$$

For $\mathbf{W} = \mathbf{A}^{-1}$, as no further adaptation is required once the correct demixing system has been obtained. Combining this new constraint with Equation 3.34, then the equation satisfies the above constraint, when the cost function $J(\mathbf{W})$ does. Therefore the functions have the same equilibrium, although their stability may be different. The Natural Gradient algorithm will not effect the stability of an equilibrium provided the matrix $\mathbf{G}^{-1}$ is a positive definite matrix [3]. Repeating the analysis above for the BSS problem as defined by Amari in [101, 117, 114, 112], we begin with the demixing matrix $\mathbf{W}$. We wish to extend this matrix by an infinitesimally small perturbation $d\mathbf{W}$ such that:

$$\mathbf{W} \rightarrow \mathbf{W} + d\mathbf{W} \tag{3.36}$$

[3] A positive definite matrix has the property that the determinant is always positive

The tangent space $T_{\mathbf{W}}$ of the Lie Group at W is a linear vector space, it is known as a Lie Algebra and is spanned by all small perturbations dW_{ij}. Defining an inner product at $\mathbf{W}$ that will give a distance metric within the weight space $\mathbf{W}$, and defining the tangent vector at $\mathbf{W}$ as $d\mathbf{W}$, the small perturbation results in the following inner product at $\mathbf{W}$:

$$\begin{aligned} |d\mathbf{W}|^2 &= \langle d\mathbf{W}, d\mathbf{W} \rangle_{\mathbf{W}} \\ &= tr(d\mathbf{W}^T d\mathbf{W})_{\mathbf{W}} \end{aligned} \tag{3.37}$$

Taking into account that to solve the BSS problem we require to find the inverse of the transformation $\mathbf{x} = \mathbf{As}$, such that the demixing matrix equals $\mathbf{W} = \mathbf{A}^{-1}$. Therefore the following mapping exists:

$$\mathbf{WA} = \mathbf{WW}^{-1} = \mathbf{I} \tag{3.38}$$

The perturbation $d\mathbf{W}$ at $\mathbf{W}$ is mapped to a perturbation at the Identity matrix by multiplying the above mapping by W^{-1}:

$$(\mathbf{W} + d\mathbf{W})\mathbf{W}^{-1} = \mathbf{I} + d\mathbf{W}\mathbf{W}^{-1} \tag{3.39}$$

Creating the term for the right multiplied tangent vector

$$d\mathbf{X} = d\mathbf{W}\mathbf{W}^{-1} \tag{3.40}$$

and comparing the above equations we can see that the tangent vector $d\mathbf{W}$ at $\mathbf{W}$, corresponds to the tangent vector $d\mathbf{X}$ at $\mathbf{I}$. In this case both the tangent vectors must have the same length [101]. Therefore we can equate the inner products at both $\mathbf{W}$ and the identity $\mathbf{I}$ as follows:

$$\langle d\mathbf{W}, d\mathbf{W} \rangle_{\mathbf{W}} = \langle d\mathbf{X}, d\mathbf{X} \rangle_{\mathbf{I}} \tag{3.41}$$

Following the analysis for the Riemannian metric tensor vector component given in section 3.2 we can equate the weighted inner products above to the Riemannian

metric tensor for the matrix case as follows:

$$\langle d\mathbf{W}, d\mathbf{W}\rangle_{\mathbf{W}} = \langle d\mathbf{X}, d\mathbf{X}\rangle_{\mathbf{I}} = \sum_{ij}(dX_{ij})^2 = \sum_{ijkl} G_{ijkl}(\mathbf{W})d\mathbf{W}_{ij}d\mathbf{W}_{kl} \tag{3.42}$$

where the Riemannian metric tensor $\mathbf{G}$ is defined as follows:

$$\mathbf{G}_{ijkl}(\mathbf{W}) = \sum_{m} \delta_{ik}\mathbf{W}^{-1}_{jm}\mathbf{W}^{-1}_{lm} \tag{3.43}$$

Extending again the analysis in section 3.2 the first two terms of the Taylor series expansion:

$$J(\mathbf{W} + d\mathbf{W}) = J(\mathbf{W}) + \epsilon\nabla J(\mathbf{W})^T d\mathbf{W} \tag{3.44}$$

Utilizing the above equation the matrix differential can be given as:

$$J(\mathbf{W} + d\mathbf{W}) - J(\mathbf{W}) = \epsilon\langle\nabla J(\mathbf{W}), d\mathbf{W}\rangle_I \tag{3.45}$$

Interpreting the Natural Gradient $\tilde{\nabla}J(\mathbf{W})$ of J as a vector applied at $\mathbf{W}$ and the standard gradient $\nabla J(\mathbf{W})$ of J as a vector applied at the identity $\mathbf{I}$, then from the above analysis these two may be equated as follows:

$$\langle\tilde{\nabla}J(\mathbf{W})^T, d\mathbf{W}\rangle_{\mathbf{W}} = \langle\tilde{\nabla}J(\mathbf{W})\mathbf{W}^{-1}, d\mathbf{W}\mathbf{W}^{-1}\rangle_{\mathbf{W}\mathbf{W}^{-1}} = \langle\nabla J(\mathbf{W})^T, d\mathbf{W}\rangle_I \tag{3.46}$$

Rewriting the above weighted inner products in trace form:

$$tr(\mathbf{W}^{-T}\tilde{\nabla}J(\mathbf{W})^T d\mathbf{W}\mathbf{W}^{-1}) = tr(\nabla J(\mathbf{W})^T d\mathbf{W}) \tag{3.47}$$

Utilizing the commutative properties of the trace function and equating terms:

$$tr((\mathbf{W}^{-1}\mathbf{W}^{-T}\tilde{\nabla}J(\mathbf{W}) - \nabla J(\mathbf{W}))d\mathbf{W}) = 0 \tag{3.48}$$

Rearranging terms in the above equation, gives the following equation for the Natural Gradient algorithm within the space of matrices for application to the BSS problem [101, 117, 114, 112]:

$$\tilde{\nabla}J(\mathbf{W}) = \nabla J(\mathbf{W})\mathbf{W}^T\mathbf{W} \tag{3.49}$$

The gradient directions for a hypothetical manifold are shown diagrammatically in Figure 3.2.5 for both the standard stochastic gradient $\nabla J(\mathbf{W})$ and for the Natural Gradient $\tilde{\nabla} J(\mathbf{W})$.

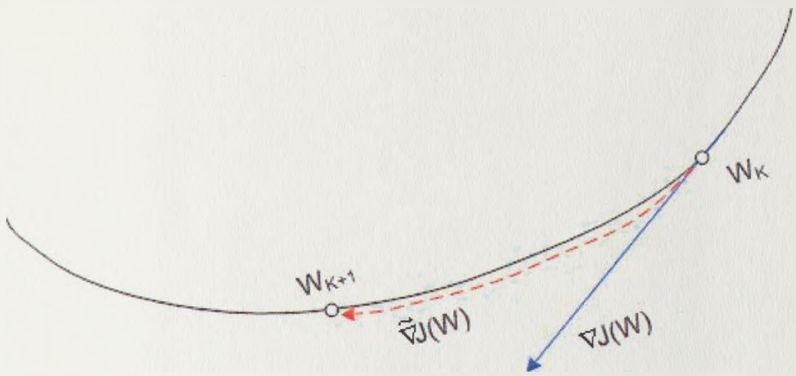


Figure 3.2: Comparison of Natural Gradient and Stochastic Gradient descent directions

Thus, the update equation including the Natural Gradient cost function modification is given as follows:

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \mu[\mathbf{I} - \phi(\mathbf{y})\mathbf{y}^T]\mathbf{W} \tag{3.50}$$

The nonlinearity $\phi(\mathbf{y})$ is chosen such that it models implicitly the higher order statistics of the underlying source signal. It was shown in [118, 119, 120, 121] that the exact form of the nonlinearity used with the Natural Gradient and InfoMax algorithms is not crucial to the separation success providing that the sign of the kurtosis of the underlying signals is unchanged. The performance of the InfoMax and Natural Gradient algorithms is compared in the following section.

3.3 Simulation Example

In order to demonstrate the performance of the Natural Gradient algorithm in on-line mode, and to show the improvement in performance over the InfoMax

algorithm the following simulation is employed. The source signals are three artificially generated sub-Gaussian signals, consisting of a Uniformly distributed signal, and two Binary signals. They are shown in Figure 3.3 The signals are

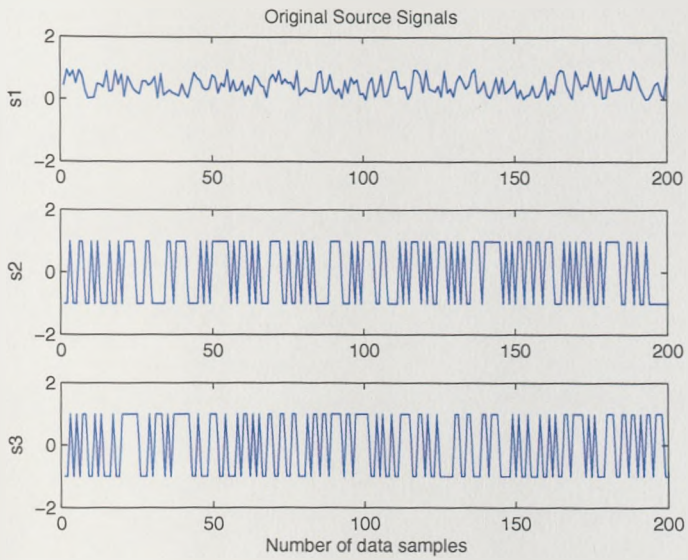


Figure 3.3: Sub-Gaussian source signals

then mixed using a random mixing matrix, generated from a Uniform distribution. The mixed signals are shown in Figure 3.4. An example of the unmixed signals generated by the Natural Gradient algorithm is chosen randomly from the output of 10 simulation runs and is shown in Figure 3.5: Within this example the source signals were sub-Gaussian in nature, therefore the score function $\phi(\mathbf{y}) = \mathbf{y}^3$ was chosen. The score function has been shown to provide good performance for separation of sub-Gaussian signals [101, 113, 114, 112, 88]. The performance of the algorithm shown in Figure 3.6, represents the average value of two hundred simulation runs of the above described example. The performance metric used to show the convergence of the algorithm is Amari's performance metric, described in the previous chapter. It can be seen clearly from Figure 3.6 that the Natural

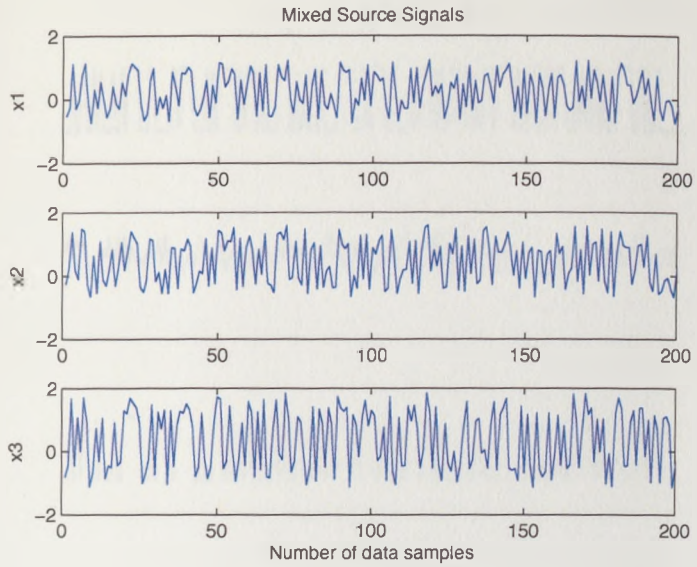


Figure 3.4: Signals mixed using a uniformly distributed random mixing matrix

Gradient extension to the InfoMax algorithm vastly improves the convergence of the InfoMax algorithm. The removal of the matrix inversion in the update equation also represents a large reduction in the computation per iteration required by the algorithm. Later in this thesis it will be shown for the communications context that convergence improvements can be achieved by better modeling the distributions of the underlying signals.

3.4 EASI

At the same time that Amari was developing the Natural Gradient method Cardoso and Laheld had independently developed a similar method they termed the gradient update within the paper the Relative Gradient method [122, 123], the complete algorithm within the paper is known as the Equivariant Adaptive Source Separation (EASI) algorithm. As the development of this algorithm results in a similar function in nature to the Natural Gradient algorithm its development

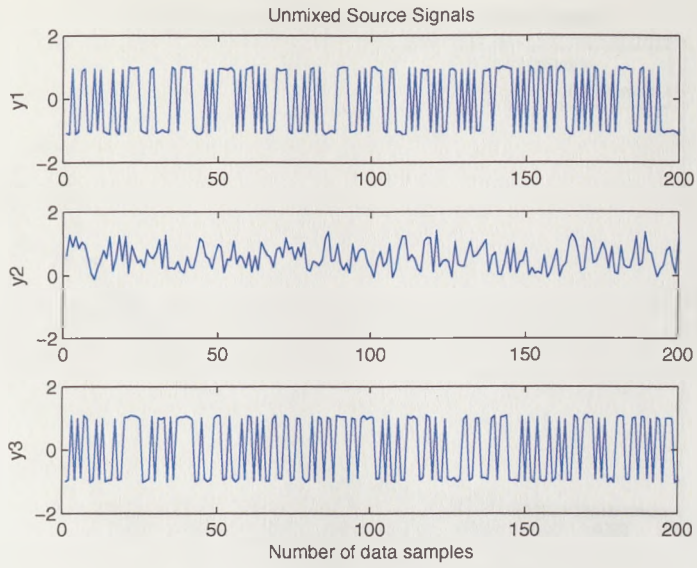


Figure 3.5: Unmixed signals using the Natural Gradient algorithm

is left described in the references. The algorithm does differ in the inclusion of a prewhitening stage as part of the gradient update equation. The equation is given as follows:

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \mu[\mathbf{y}\mathbf{y}^T - \mathbf{I} + \phi(\mathbf{y})\mathbf{y}^T - \mathbf{y}\phi(\mathbf{y})^T]\mathbf{W} \tag{3.51}$$

For all of the above algorithms the convergence of the algorithm is dependent on correct choice of the adaptation parameter μ. A fixed value for μ is often used, but if fast convergence speed is required then larger adaptation rates become necessary, this can lead to algorithm instability. A number of adaptive step size selection methods have been introduced for use with the Natural gradient algorithm [124, 125, 126] that offer improvements in both algorithm convergence and misadjustment reduction. In the EASI algorithm a normalized version of the algorithm was introduced similar to the normalized LMS algorithm [127] that provides extra stabilization for the algorithm. The normalized EASI algorithm

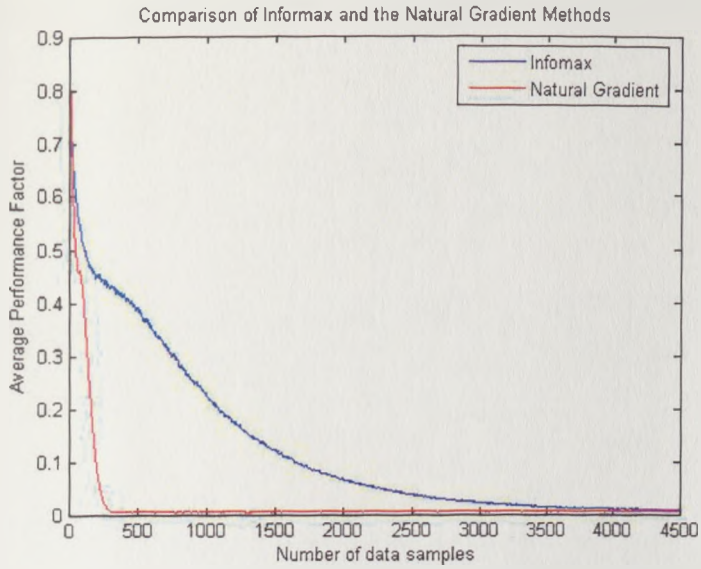


Figure 3.6: Comparison of the Natural Gradient and the InfoMax algorithms

is given as follows:

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \mu[\frac{\mathbf{y}\mathbf{y}^T - \mathbf{I}}{1 + \mu\mathbf{y}\mathbf{y}^T} + \frac{\phi(\mathbf{y})\mathbf{y}^T - \mathbf{y}\phi(\mathbf{y})^T}{1 + \mu|\mathbf{y}^T\phi(\mathbf{y})|}]\mathbf{W} \tag{3.52}$$

In [128] an algorithm is developed that provides additional stability to the Natural Gradient algorithm [101, 113, 114, 112, 88]. This algorithm was extended to the EASI algorithm and was then applied to the separation of communication signals in [129].

3.5 Conclusion

In the above analysis it can be seen clearly that incorporating the Natural Gradient extension into the InfoMax algorithm reduces the computational complexity of the algorithm greatly due to the removal of the matrix inversion within the cost function, and increases the convergence of the algorithm due to better modeling of the underlying cost surface [101, 113, 114, 112, 88]. The EASI algorithm was

introduced due to its similarity to the Natural Gradient method. The Matrix Momentum algorithm is introduced within the next chapter. This algorithm represents a second order gradient based method, similar in nature to the Newton method without the requirement for a matrix inversion of the Hessian matrix. This method is shown to provide fast convergence and low complexity. The Matrix Momentum method is fully explained within the following chapter and its performance is compared with the Natural Gradient algorithm.

Chapter 4

Matrix Momentum

In this chapter the Matrix Momentum algorithm is introduced, which is known to improve the performance of the standard Steepest Descent gradient algorithm. The algorithm is then utilized within the Blind Source Separation (BSS) context as an extension to the Information Maximization algorithm described in the previous chapter. The performance of the algorithm is demonstrated via the separation of a mixture of speech signals. It is shown that the algorithm achieves fast convergence and low computational complexity when compared with the Natural Gradient and Relative Newton algorithms.

4.1 Introduction

In the previous chapter Amari's Natural Gradient algorithm was discussed, and it was shown to provide dramatic increases in convergence speed and computational complexity when compared with the InfoMax algorithm [104]. The Newton Method is a well known technique in Optimization Theory offering quadratic convergence properties that significantly outperform the standard Steepest Descent technique. One of the well known problems with the Newton Method is the

requirement of a matrix inversion at each iteration step. A number of Newton Methods have been applied to the BSS problem each avoiding the matrix inversion in a novel fashion [130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140]. In this chapter the Matrix Momentum algorithm developed by Orr in [141, 142], is utilised as a second order optimization technique applied to the BSS problem, resulting in a Newton type method without the requirement for a matrix inversion. This provides improved convergence relative to the standard Steepest Descent methods without significant increase in computational cost. The Matrix Momentum algorithm will be described fully within this chapter, before being applied within the framework of the BSS problem. The Matrix Momentum algorithm is demonstrated to further increase the convergence with respect to the Natural Gradient algorithm [101, 113, 114, 112, 88] and is shown to provide equal performance in convergence with low computational complexity when compared with alternative Newton based BSS methods [134, 135, 136, 136]. The development of the above algorithms is shown within the following sections. To begin the development of the Matrix Momentum algorithm it will be important to first revisit the Steepest Descent technique, before continuing to the Newton Method.

4.2 Gradient Learning Algorithms

It was mentioned in the previous chapter that the Steepest Descent algorithm [108, 110, 109] was utilized in the development of the InfoMax algorithm [104]. The algorithm dates back to 1847 [108] and represents one of the most time served algorithms known in Optimization Theory. The format of the Steepest Descent algorithm for a matrix valued cost function $J(\mathbf{W})$ is given by the following equation.

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \mu C(\mathbf{W}(k)) \frac{\partial J(\mathbf{W}(k))}{\partial \mathbf{W}(k)} \tag{4.1}$$

where as previously described μ represents the learning rate parameter for the algorithm and additionally added is $C(\mathbf{W})$ a suitably chosen positive definite matrix. For the standard Steepest Descent algorithm the matrix is chosen as the identity matrix $C(\mathbf{W}) = \mathbf{I}$. It was shown in the previous chapter that if the matrix $C(\mathbf{W})$ is set equal to the inverse of the Riemannian metric tensor $\mathbf{G}^{-1}$ then we find the direction of steepest descent in the Riemannian space, this algorithm developed by Amari is known as the Natural Gradient [101, 113, 114, 112, 88], and provides rapid convergence and reduced computational complexity when compared with the Steepest Descent approach [104] . For the case that the matrix $C(\mathbf{W})$ is set equal to the inverse Hessian matrix, this represents another well known algorithm within Optimization Theory: the Newton Method. As the Newton Method is important to the understanding of the Matrix Momentum algorithm it will be described in the following subsection.

4.2.1 The Newton Method

Newton method is often described initially in terms of root finding. As this explanation can naturally be extended to function minimization or maximization, this explanation will be used here. Beginning with the Taylor series expansion of the scalar valued function $J(w)$ about the point $w = w(k) + \triangle w$ we find the roots of the function as follows.

$$J(w(k) + \triangle w) = J(w(0)) + (\triangle w)J'(w(0)) + O(n^2) \tag{4.2}$$

Truncating the above expansion at first order, setting the equation equal to zero such that $J(w(0) + \triangle w) = 0$, then solving for $\triangle w = \triangle w(0)$, results in the following equation for an initial guess $w(0)$.

$$\triangle w(0) = -\frac{J(w(0))}{J'(w(0))} \tag{4.3}$$

Letting $\triangle w_1 = w_0 + \triangle w_0$, and repeating this results in the following update equation.

$$\triangle w(k) = -\frac{J(w(k))}{J'(w(k))} \tag{4.4}$$

We can use the above to develop an iterative update equation for finding the root. This is given as follows.

$$w(k+1) = w(k) - \frac{J(w(k))}{J'(w(k))} \tag{4.5}$$

Trivially, it can be seen that the above analysis can be extended to function minimization by observing that if a variable w is a stationary point of a cost function $J(w)$ then the variable w is a root of the cost function's derivative. Therefore the value of the input variable w can be found by applying the Newton method to $J'(w)$. This results in the following update equation.

$$w(k+1) = w(k) - \frac{J'(w(k))}{J''(w(k))} \tag{4.6}$$

This method has the constraint that it requires the cost function $J(w)$ to be twice differentiable. Extending this analysis to a vector valued twice differentiable cost function $J(\mathbf{w})$ we obtain the following update equation for the Newton method.

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mathbf{H}^{-1}\nabla J(\mathbf{w}) \tag{4.7}$$

Where $\mathbf{H}$ represents the $n \times n$ Hessian matrix, the matrix of second partial derivatives, and $\nabla J(\mathbf{w})$ represents the $n \times 1$ Jacobian vector, the vector of partial derivatives. Extending this result to a matrix valued twice differentiable cost function as required in BSS the following Newton update equation is obtained.

$$\mathbf{W}(k+1) = \mathbf{W}(k) - mat\left[\mathbf{H}^{-1} vec[\nabla J(\mathbf{W})]\right] \tag{4.8}$$

Where *mat* is an operator that transforms a $n^2 \times 1$ vector into a $n \times n$ matrix, and the *vec* operator turns a $n \times n$ matrix into a $n^2 \times 1$ vector. These operators

are required as the Hessian matrix $\mathbf{H}$ of a matrix valued cost function results in a $n^2 \times n^2$ matrix. The above equation more generally will be written including a step size parameter μ which is generally defined in the range $0 < \mu < 1$. To ensure algorithm convergence the step size μ should be chosen to ensure the Wolfe conditions are satisfied for each iteration of the method [143, 144].

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \mu\left(mat\left[\mathbf{H}^{-1}vec[\nabla J(\mathbf{W})]\right]\right) \tag{4.9}$$

It can be seen clearly in the above equation that the inversion of a $n^2 \times n^2$ matrix represents a significant cost overhead in the implementation of a straight Newton method applied to the BSS problem. In the following section the Matrix Momentum algorithm developed by Orr in [141, 142] is introduced, which provides an algorithm with Newton-like performance avoiding the high computational cost of the straight Newton method. In [26, 27] the first application of the Matrix Momentum algorithm to the BSS problem was introduced, as detailed further within this thesis.

4.3 The LMS algorithm with Momentum

The idea of incorporating previous values of the weight vector along with the standard Steepest Descent update term was first utilized by Proakis in [145, 58] for high speed adaptive equalization in digital communications. The idea was then revisited by Roy in [146, 147] and within these references, the algorithm was named the Momentum LMS (MLMS) algorithm. At the same point the addition of a momentum parameter was further analysed by Tugay in [148, 149, 150]. The momentum parameter addition to the Steepest Descent update equation is shown as follows:

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \mu\frac{\partial J(\mathbf{W}(k))}{\partial \mathbf{W}(k)} + \beta(\mathbf{W}(k) - \mathbf{W}(k-1)) \tag{4.10}$$

Where to ensure stability the momentum parameter β is defined such that $|\beta| < 1$. It was found by Roy and Shynk in [147] that while the algorithm remains stable if the momentum parameter β is negative within the above range this can cause decreased algorithm performance. Therefore, generally the parameter is chosen to be positive. Expanding the right hand side of the above steepest descent equation with momentum, it can be seen that the above equation can be written as an exponential averaging of the weight update equation $\frac{\partial J(\mathbf{W}(k))}{\partial \mathbf{W}(k)}$ for a constant value for β. Rewriting the above equation as an exponential average the following equation is obtained.

$$\begin{aligned}
\mathbf{W}(k+1) &= \mathbf{W}(k) - \mu\frac{\partial J(\mathbf{W}(k))}{\partial \mathbf{W}(k)} + \beta\left(\mathbf{W}(k-1) - \mu\frac{\partial J(\mathbf{W}(k-1))}{\partial \mathbf{W}(k-1)}\right) - \beta(\mathbf{W}(k-1)) \\
&= \mathbf{W}(k) - \mu\left(\frac{\partial J(\mathbf{W}(k))}{\partial \mathbf{W}(k)} - \beta\frac{\partial J(\mathbf{W}(k-1))}{\partial \mathbf{W}(k-1)}\right) \\
&= \mathbf{W}(k) - \mu\left(\beta^0\frac{\partial J(\mathbf{W}(k))}{\partial \mathbf{W}(k)} - \beta^1\frac{\partial J(\mathbf{W}(k-1))}{\partial \mathbf{W}(k-1)}\right) \\
&= \mathbf{W}(k) - \mu\sum_{i=0}^{1}\beta^i\left(\frac{\partial J(\mathbf{W}(k-i))}{\partial \mathbf{W}(k-i)}\right)
\end{aligned} \tag{4.11}$$

For the case where the momentum parameter β lies within the range defined in [147] as detailed above, and making the assumption that within the significant terms within the above exponential average equation the gradient terms are not changing rapidly, then the finite summation term can be replaced by an infinite summation as follows:

$$\mu\sum_{i=0}^{1}\beta^i\left(\frac{\partial J(\mathbf{W}(k-i))}{\partial \mathbf{W}(k-i)}\right) \approx \mu\frac{\partial J(\mathbf{W}(k))}{\partial \mathbf{W}(k)}\sum_{i=0}^{\infty}\beta^i \tag{4.12}$$

Using the above equation it can be seen from the convergence of the Geometric series that the momentum addition has the effect of rescaling the learning rate parameter,

$$\mu\frac{\partial J(\mathbf{W}(k))}{\partial \mathbf{W}(k)}\sum_{i=0}^{\infty}\beta^i \approx \frac{\mu}{1-\beta}\left(\frac{\partial J(\mathbf{W}(k))}{\partial \mathbf{W}(k)}\right) \tag{4.13}$$

This is shown as follows where the learning rate μ_{eq} is defined as the equivalent learning that would be required to obtain the same rate of convergence for the case that $\beta = 0$:

$$\mu_{eq} = \frac{\mu}{1-\beta} \tag{4.14}$$

It was shown in [141] that the addition of a momentum term does not improve convergence in excess of what can be achieved by simply utilizing the equivalent learning rate μ_{eq}. This is an unsurprising result, yet the addition of a momentum term has been utilized extensively in the neural network literature [151, 152, 153, 154] and the references therein. Specifically in the context of the Backpropagation algorithm [42]. In the following subsection the Matrix Momentum algorithm is introduced.

4.3.1 Matrix Momentum

It has been shown previously that although the Newton method provides improved convergence with respect to the Steepest Descent algorithm, the required matrix inversion per iteration can be computationally expensive. As a method of removing the requirement for this matrix inversion Orr introduced the Matrix Momentum algorithm [141, 142]. Beginning with the momentum equation given in Equation 4.10 it was shown that the addition of the momentum term had the effect of rescaling the learning parameter. It was shown in subsection 4.2.1 that for the Newton Method the learning rate parameter is equal to the inverse of the Hessian of the cost function. In [141] Orr posed the question: "What was the momentum parameter required such that the equivalent learning rate parameter μ_{eq} was equal to the inverse of the Hessian matrix?". This is shown as follows for the matrix valued case:

$$\mu_{eq} = \mathbf{H}^{-1} = \mu[\mathbf{I} - \beta]^{-1} \tag{4.15}$$

Where $\mathbf{I}$ in the above equation represents the identity matrix. Solving the above equation for the momentum parameter β we obtain the following result such that the effective learning rate for the system is equal to the inverse of the Hessian matrix of the cost function:

$$\beta = \mathbf{I} - \mu\mathbf{H} \tag{4.16}$$

Placing this β momentum parameter in the context of the steepest descent equation with the momentum update as shown in Equation 4.10 we obtain the following update equation.

$$\begin{aligned} \mathbf{W}(k+1) &= \mathbf{W}(k) - \mu\frac{\partial J(\mathbf{W}(k))}{\partial \mathbf{W}(k)} \\ &+ \Big((\mathbf{W}(k) - \mathbf{W}(k-1)) \\ &- \mu\Big(mat\Big[\mathbf{H}vec(\mathbf{W}(k) - \mathbf{W}(k-1))\Big)\Big]\Big) \end{aligned} \tag{4.17}$$

Rewriting the final section of the above equation for simplicity as $\Delta\mathbf{W}(k) = \mathbf{W}(k) - \mathbf{W}(k-1)$, the above equation takes the following form.

$$\begin{aligned} \mathbf{W}(k+1) &= \mathbf{W}(k) - \mu\frac{\partial J(\mathbf{W}(k))}{\partial \mathbf{W}(k)} \\ &+ \Big(\Delta\mathbf{W}(k) - \mu\Big(mat\Big[\mathbf{H}vec(\Delta\mathbf{W}(k))\Big]\Big)\Big) \end{aligned} \tag{4.18}$$

Thus it can be seen from the above equation that it is required to calculate the product of the Hessian matrix and the vector of previous weights $mat[\mathbf{H}vec(\Delta\mathbf{W}(k))]$. This product can be calculated in two ways, the full Hessian matrix $\mathbf{H}$ of the cost function $J(\mathbf{W})$ can be calculated as in Newton based approaches [134, 135, 136, 138, 117, 112, 88] and multiplied by the vector $vec(\Delta\mathbf{W}(k))$ or alternatively the product of the Hessian $\mathbf{H}$ and the vector can be found in one calculation. The latter method will be explored within this chapter. The product of the Hessian and an arbitrary vector was introduced by Pearlmutter in [155] and at the same time a similar method was developed

independently by Moller in [156, 157]. This method is described in the following section initially for functions of a vector, as this can then be further extended to functions of a matrix utilizing the *mat* and *vec* operators, transforming the matrices to vectors, calculating the Hessian vector product, then transforming again the corresponding vectors to matrices.

4.4 Pearlmutter's Hessian Vector product

It is a well known fact that the Hessian (the matrix of second order derivatives) and higher order derivatives appear in the Taylor series expansion of the gradient perturbed around a point in the parameter space of the a given vector $\mathbf{w}$, this may be shown as follows using prime notation for simplicity:

$$J'(\mathbf{w} + \Delta\mathbf{w}) = J'(\mathbf{w}) + \mathbf{H}\Delta\mathbf{w} + O(||\Delta\mathbf{w}||^2) \tag{4.19}$$

where $J(\mathbf{w})$ represents the cost function, $\mathbf{H}$ is the Hessian matrix and $\Delta\mathbf{w}$ represents a perturbation of the vector $\mathbf{w}$. Setting this perturbation $\Delta\mathbf{w} = r\mathbf{v}$ where $\mathbf{v}$ represents an arbitrary vector and r a small number. Manipulating the above equation to compute the product of the Hessian $\mathbf{H}$ with vector $\Delta\mathbf{w}$ the following equation is obtained:

$$\begin{aligned} \mathbf{H}\Delta\mathbf{w} &= \mathbf{H}(r\mathbf{v}) = r\mathbf{H}\mathbf{v} \\ r\mathbf{H}\mathbf{v} &= J'(\mathbf{w} + r\mathbf{v}) - J'(\mathbf{w}) + O(r^2) \end{aligned} \tag{4.20}$$

dividing the above equation by r

$$\mathbf{H}\mathbf{v} = \frac{J'(\mathbf{w} + r\mathbf{v}) - J'(\mathbf{w})}{r} + O(r) \tag{4.21}$$

The above equation has been used for calculation of the Hessian vector product; however this method has the drawback that it is very susceptible to round off

errors. This arises due to the the fact that the constant term r must be small enough that the $O(r)$ term is insignificant, thus the precision of the vector $\mathbf{v}$ is affected. A loss of precision is also experienced in the gradient calculation of the perturbed gradient minus the original one, as the values are almost identical. To alleviate the numerical difficulties associated with the above method, Pearlmutter used the following elegant solution to compute $\mathbf{Hv}$ exactly:

$$\mathbf{Hv} = \lim_{r \to 0} \frac{J'(\mathbf{w} + r\mathbf{v}) - J'(\mathbf{w})}{r} = \frac{\partial}{\partial r} J'(\mathbf{w} + r\mathbf{v}) \Big|_{r=0} \tag{4.22}$$

Taking the limit of the above Equation 4.22 as $r \to 0$ gives the definition of a gradient on the right hand side of the equation, leaving the left hand side as the Hessian vector product $\mathbf{Hv}$. Pearlmutter defined the following transformation to convert a gradient calculation algorithm into a Hessian vector product calculation [155]. This transformation was achieved by defining the following operator:

$$\mathbf{R}\{J(\mathbf{w})\} = \frac{\partial}{\partial r} J(\mathbf{w} + r\mathbf{v}) \Big|_{r=0} \tag{4.23}$$

The above operator is then applied to each of the equations of the procedure for calculating the gradient. As $\mathbf{R}\{.\}$ is a differential operator, it follows the standard rules for differential operators, these are written for the $\mathbf{R}\{.\}$ case as follows:

$$\mathbf{R}\{c\} = 0 \tag{4.24}$$

$$\mathbf{R}\{\mathbf{w}\} = \mathbf{v} \tag{4.25}$$

$$\mathbf{R}\{f(g(\mathbf{w}))\} = f'(g(\mathbf{w}))\mathbf{R}\{g(\mathbf{w})\} \tag{4.26}$$

$$\mathbf{R}\{cf(\mathbf{w})\} = c\mathbf{R}\{f(\mathbf{w})\} \tag{4.27}$$

$$\mathbf{R}\{f(\mathbf{w})g(\mathbf{w})\} = \mathbf{R}\{f(\mathbf{w})\}g(\mathbf{w}) + f(\mathbf{w})\mathbf{R}\{g(\mathbf{w})\} \tag{4.28}$$

The initial aim was to embed the Natural Gradient algorithm within the Matrix Momentum algorithm. This was implemented utilizing the above rules in Equations 4.24-4.28, and combining with the Natural Gradient update equations the following equation is developed for the Hessian vector product required for the Matrix Momentum upgrade equation.

$$\begin{aligned} mat\left[\mathbf{H}vec(\Delta\mathbf{W})\right] &= \mathbf{R}\left\{\frac{\partial J(\mathbf{W})}{\partial \mathbf{W}}\mathbf{W}^T\mathbf{W}\right\} \\ &= \mathbf{R}\{[\mathbf{I}-\phi(\mathbf{y})\mathbf{y}^T]\mathbf{W}\} \end{aligned} \tag{4.29}$$

As previously the *vec* operator amalgamates into a single column vector the columns of a matrix, and the *mat* operator reverses this operation. Applying Pearlmutter's $\mathbf{R}\{.\}$ operator to Equation 4.29 gives the following result:

$$\begin{aligned} \mathbf{R}\{[\mathbf{I}-\phi(\mathbf{y})\mathbf{y}^T]\mathbf{W}\} &= \{[-diag(\phi'(\mathbf{y}))\mathbf{y}^T]\Delta\mathbf{W} \\ &+ \phi(\mathbf{y})(\Delta\mathbf{W}\mathbf{y})^T\}\mathbf{W} \\ &+ [\mathbf{I}-\phi(\mathbf{y})\mathbf{y}^T]\Delta\mathbf{W} \end{aligned} \tag{4.30}$$

As is standard in Natural Gradient based algorithms the choice of the score function $\phi(.)$ is critical to the algorithm performance. It was shown in [104, 88] that the score function should be chosen as follows:

$$\phi(\mathbf{y}) = \frac{\partial}{\partial \mathbf{W}}\log(p(\mathbf{y})) = \frac{p'(\mathbf{y})}{p(\mathbf{y})} \tag{4.31}$$

For the separation of super-Gaussian sources the hyperbolic tangent function tanh(.) is commonly utilized as the nonlinearity for the above equation. Placing this equation in the context of the Hessian vector product of the gradient update equation, including the Natural gradient update equation and the momentum term $\triangle\mathbf{W}$ for insertion within the Matrix Momentum algorithm, we obtain the

following equation.

$$\begin{aligned}\mathbf{R}\{[\mathbf{I}-\tanh(\mathbf{y})\mathbf{y}^T]\mathbf{W}\} &= \{[-diag(1-\tanh^2(\mathbf{y}))\mathbf{y}^T]\Delta\mathbf{W} \\ &+ (1-\tanh^2(\mathbf{y}))(\Delta\mathbf{W}\mathbf{y})^T\}\mathbf{W} \\ &+ [\mathbf{I}-\phi(\mathbf{y})\mathbf{y}^T]\Delta\mathbf{W} \end{aligned} \tag{4.32}$$

Utilizing this equation the following gradient update equation is obtained for the Matrix Momentum algorithm, including the Hessian vector product created from the Natural Gradient update equation.

$$\begin{aligned}\mathbf{W}(k+1) &= \mathbf{W}(k)-\mu[\mathbf{I}-\tanh(\mathbf{y})\mathbf{y}^T]+\Delta\mathbf{W}(k) \\ &- \mu\Big\{\{[-diag(1-\tanh^2(\mathbf{y}))\mathbf{y}^T]\Delta\mathbf{W} \\ &+ (1-\tanh^2(\mathbf{y}))(\Delta\mathbf{W}\mathbf{y})^T\}\mathbf{W} \\ &+ [\mathbf{I}-\phi(\mathbf{y})\mathbf{y}^T]\Delta\mathbf{W}\Big\} \end{aligned} \tag{4.33}$$

In the next section the performance of the above algorithm is discussed.

4.5 Simulations

The above Hessian vector product including the tanh(.) nonlinearity was utilized to apply the Matrix Momentum algorithm to the separation of super-Gaussian sources. A similar approach to the above had been utilised previously in application to the BSS problem by Schraudolph in [158] in the context of the Stochastic Meta Descent (SMD) algorithm. As stated by Schraudolph in [159, 160, 161] the Matrix Momentum algorithm is prone to instability when applied to nonlinear problems. It was found that the above algorithm caused the system to diverge. As a method of stabilizing the algorithm, Schraudolph proposed an adaptive term in [162]. To combat the above stability problems, in utilizing Pearlmutter's Hessian vector product within the Matrix Momentum algorithm, there are two key

modifications that could be made to improve the performance of the algorithm for application to the separation of super-Gaussian signals. The first modification was to utilize the exact Hessian calculation of the underlying cost function $J(\mathbf{W})$. The second was to utilize an improved nonlinearity specific to super-Gaussian signals proposed in the development of the Relative Newton method described in [135, 134, 138]. It is also believed that one of the problems associated with this approach is that the calculation of the gradient is made using the modified differential $d\mathbf{X} = d\mathbf{W}\mathbf{W}^{-1}$ as described in chapter 3, while the Hessian vector product is not calculated in the Riemannian space. This approach could be altered such that the second order gradient is calculated in the Riemannian space as shown by Elsabrouty in [163]. Instead an approach similar to the Relative Newton method [135, 134, 138] is undertaken that utilizes the full Hessian calculation. This approach is detailed within the following section and applied within the context of the Matrix Momentum algorithm.

4.6 Matrix Momentum with Full Hessian

As was shown in the above section the application of the Matrix Momentum [141, 142] method combined with Pearlmutter's Hessian vector [155] is prone to instability when applied to the BSS problem. To stabilize the Matrix Momentum algorithm within the BSS context it was decided to begin with the Maximum Likelihood cost function shown in chapter 3 to be equivalent to the InfoMax formulation. The Maximum Likelihood cost function is given as follows.

$$J(\mathbf{W}) = \log|\det(\mathbf{W})| + \sum_{i=1}^{n} \log(p_i(y_i)) \tag{4.34}$$

Taking the derivative of the above equation results in the gradient update equation seen previously in the development of the InfoMax algorithm [104, 105].

$$\frac{\partial J(\mathbf{W})}{\partial \mathbf{W}} = \mathbf{W}^{-T} - \phi(\mathbf{y})\mathbf{x}^T \tag{4.35}$$

The approach taken by Zibulevsky in the development of the Relative Newton method [134, 135, 136, 136], at this point is to calculate the Hessian of the cost function $J(\mathbf{W})$ by first utilizing the *vec* operator to transform the $n \times n$ matrices in the above equation into a vector of length $n^2 \times 1$. This results in the following update equation

$$vec\left(\frac{\partial J(\mathbf{W})}{\partial \mathbf{W}}\right) = vec(\mathbf{W}^{-T}) - vec(\phi(\mathbf{y})\mathbf{x}^T) \tag{4.36}$$

The calculation of the differential of the above equation can be broken into two distinct components, the first component $\mathbf{H}_1$ is calculated as the differential of the $vec(\mathbf{W}^{-T})$ term in Equation 4.36. To calculate the differential of this term it is useful to first begin by differentiating both sides of the identity $\mathbf{W}^{-T}\mathbf{W}^T = \mathbf{I}$.

$$\begin{aligned} d(\mathbf{W}^{-T}\mathbf{W}^T) &= d(\mathbf{I}) \\ d(\mathbf{W}^{-T})\mathbf{W}^T + \mathbf{W}^{-T}d\mathbf{W}^T &= 0 \\ d(\mathbf{W}^{-T}) &= -\mathbf{W}^{-T}d\mathbf{W}^T\mathbf{W}^{-T} \end{aligned} \tag{4.37}$$

Thus the first component $\mathbf{H}_1$ of the Hessian calculation can be written as a Kronecker product as follows.

$$\mathbf{H}_1 = -\mathbf{W}^{-T} \otimes \mathbf{W}^{-T} \tag{4.38}$$

Continuing with the calculation of the differential of the second term $vec(\phi(\mathbf{y})\mathbf{x}^T)$ of Equation 4.36, we obtain the following block diagonal matrix equation for the

second part of Hessian H_2.

$$\mathbf{H}_2 = \begin{bmatrix} \phi_1''(\mathbf{y})\mathbf{x}\mathbf{x}^T & 0 & \cdots & 0 \\ 0 & \phi_2''(\mathbf{y})\mathbf{x}\mathbf{x}^T & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \phi_n''(\mathbf{y})\mathbf{x}\mathbf{x}^T \end{bmatrix} \tag{4.39}$$

Thus the Hessian matrix can be calculated by summing the above two equations. Thus the Matrix Momentum equation is given as follows.

$$\begin{aligned} \mathbf{W}(k+1) &= \mathbf{W}(k) + \mu[\mathbf{W}^{-T} - \phi(\mathbf{y})\mathbf{x}^T] \\ &+ \Big(\triangle \mathbf{W}(k) + \mu(mat[(\mathbf{H}_1 + \mathbf{H}_2)vec(\triangle \mathbf{W})]) \Big) \end{aligned} \tag{4.40}$$

As stated in the previous section to improve the performance of the algorithm for the separation of super-Gaussian sources we require to use a model for the logarithm of the pdf $\log(p(\mathbf{y}))$ as given in Equation 4.34. While studying the sparse image coding within the visual cortex, Olshausen and Field introduced in [164, 165] a number of functions appropriate for the modeling of sparse sources, examples of these functions are $-e^{-x^2}$, the absolute function $|x|$ and $\log(1+x^2)$. Another approximation utilized within the development of the Relative Newton [134, 135, 136, 136] method is implemented. This permits fair performance comparison with the Matrix Momentum algorithm. The function is given as follows to approximate the logarithm of the pdf $\vartheta(\mathbf{y}) \approx \log(p(\mathbf{y}))$.

$$\begin{aligned} \vartheta_1(\mathbf{y}) &= |\mathbf{y}| - \log(1 + |\mathbf{y}|) \\ \vartheta_\lambda(\mathbf{y}) &= \lambda \vartheta_1\Big(\frac{\mathbf{y}}{\lambda}\Big) \end{aligned} \tag{4.41}$$

The above function has the property that it tends towards the absolute value function $|.|$ as the parameter λ tends to zero. The above analysis is utilised in the following section for the separation of three speech signals, which have a super-Gaussian distribution.

4.7 Simulations

To demonstrate the performance of the above algorithm an example is given in the following section that demonstrates the performance of the algorithm in the separation of two mixed speech and one music signal.

4.7.1 Separation of speech signals

The algorithm is utilized to separate the mixture of two speech signals and one music signal, these are shown in Figure 4.1, each signal consists of 10000 sample points. The mixing matrix is a randomly generated 3×3 matrix drawn from a uniform distribution. An example of the mixed signals is shown in Figure 4.2, and an example of the unmixed signals using the Matrix Momentum algorithm is shown in Figure 4.3.

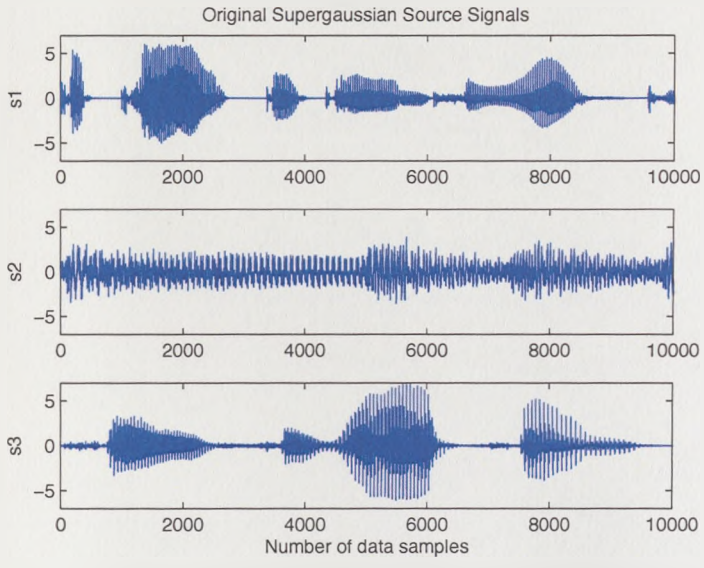


Figure 4.1: 2 speech 1 music signals before mixing

Three measures have been utilized to show the performance of the algorithm, the norm of the gradient, Amari's performance metric, and the CPU utilization

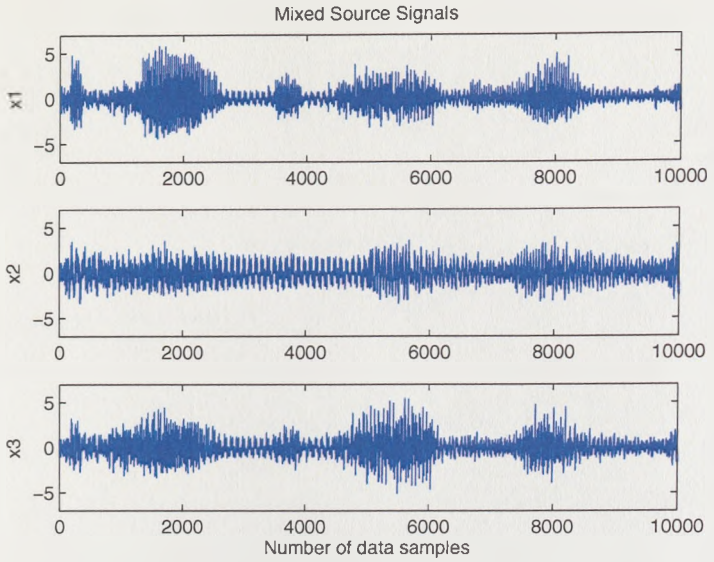


Figure 4.2: 2 speech 1 music signals mixed

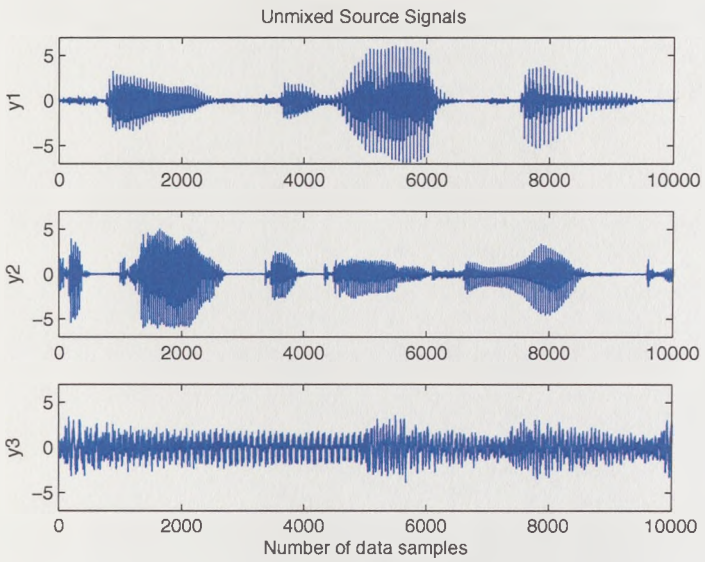


Figure 4.3: 2 speech 1 music signals unmixed

of the system, as described in chapter 2. The algorithm was run for 50 independent simulation trials in offline mode. The Matrix Momentum algorithm is

compared with both the Natural Gradient algorithms and the Relative Newton method, with the original Relative Newton Method and the fast version introduced by Zibulevsky in [136]. For the examples shown the λ parameter used in the nonlinearity shown in Equation 4.41 is set to a value of 0.01, the value was chosen heuristically. The gradient norm was chosen randomly for one of the 50 simulation trials that were run. This is shown in the following diagram. Amari's

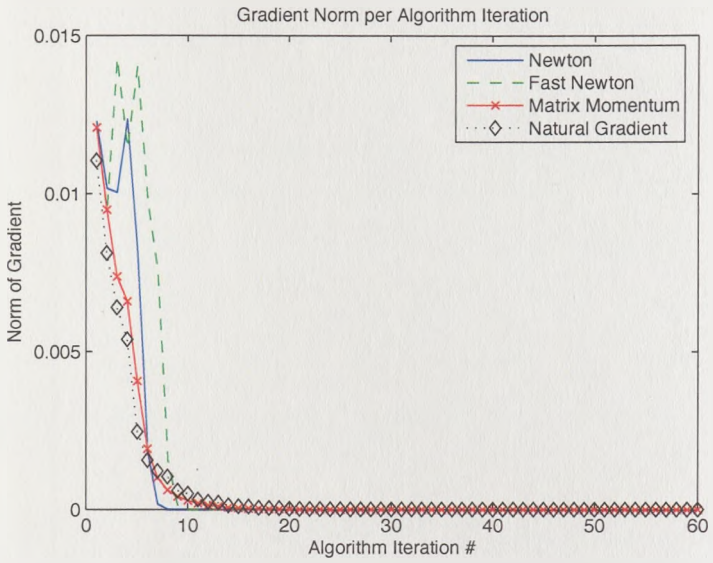


Figure 4.4: Gradient norm per algorithm iteration

performance metric as described in chapter 2 gives a measure of the separation capability of the algorithm. The average of this value for each of the algorithms over 50 independent simulation trials is given in Figure 4.5. It can be seen from Figures 4.4 and 4.5 that the Newton based methods and the Natural Gradient method converge to a similar solution. The CPU utilization is given to compare the algorithms, the simulations were performed on identical hardware. This is shown in Figure 4.6.

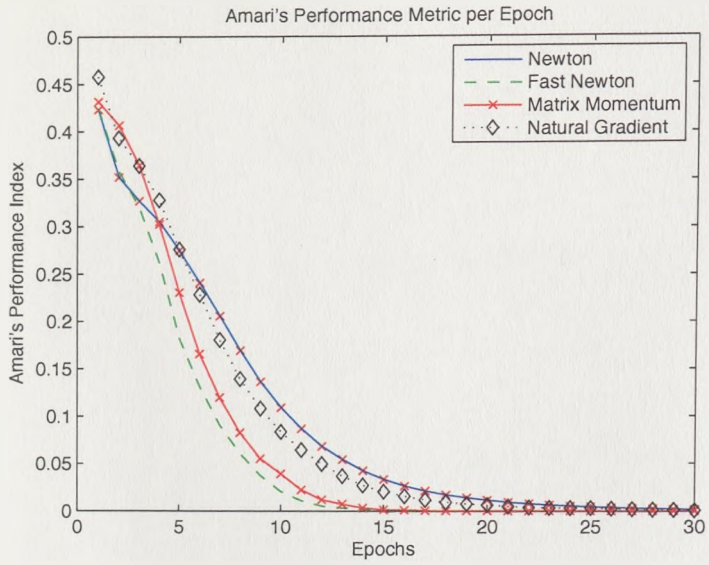


Figure 4.5: Average Amari Performance Index for 50 independent simulation trials

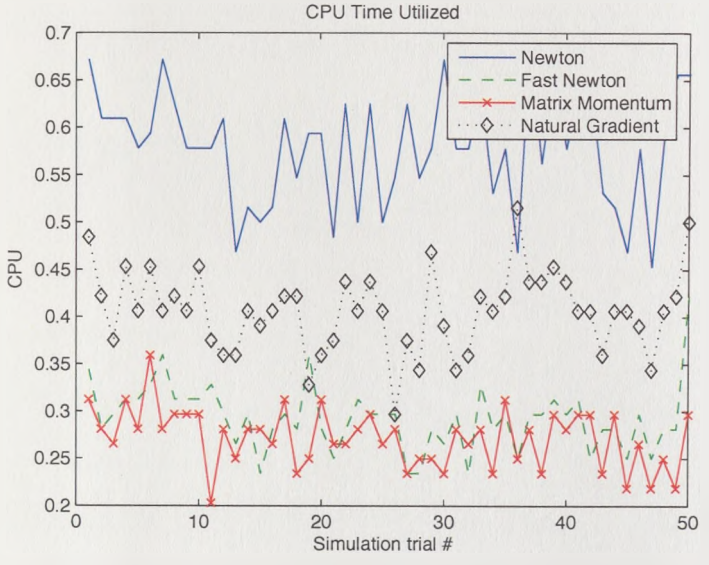


Figure 4.6: CPU time taken for each independent trial

It can be seen from the above diagrams that the convergence properties of the algorithms give similar performance. However, the CPU utilization required for the Matrix Momentum algorithm is vastly reduced when compared with the Relative Newton and Natural Gradient methods, and offers similar convergence with a slight improvement in computational complexity when compared to the Fast Newton method. Thus it has been shown that the Matrix Momentum algorithm combined with the derived Hessian provides good separation performance in the BSS case. The following section concludes the Matrix Momentum portion of this thesis.

4.8 Conclusions

In this chapter the Matrix Momentum algorithm originally introduced by Orr [141] has been described in the context of the BSS problem. The Newton Method was introduced and it was shown that the Matrix Momentum method does in fact converge to the Newton method without the requirement for a matrix inversion as needed by the standard Newton method. It was described that utilizing Pearlmutter's Hessian vector product within the Matrix Momentum algorithm as shown by Orr in [141] was not suitable for the Hessian calculation within the BSS context for the separation of super-Gaussian sources. To overcome these problems with this approach the full Hessian was calculated. This, combined with a nonlinearity specifically chosen for super-Gaussian or sparse sources, was shown to provide good performance for real and synthetic signal separation. To further improve the Matrix Momentum algorithm the following approaches are suggested.

1. Placing the Matrix Momentum upgrade equations within the context of a trust region optimization framework as described by Choi in [166, 167, 168]

and extended again by Choi for the Trust Region Relative Newton method [169, 170]. This would prevent the algorithm from diverging and resort to the standard Steepest Descent algorithm in the event that the Newton step moves too far from the cost function.

2. Another approach would be to repeat the Matrix Momentum algorithm development in the context of the Riemannian development framework. This would involve the calculation of the Hessian of the cost function $J(\mathbf{W})$ with respect to the modified differential $d\mathbf{X} = d\mathbf{W}\mathbf{W}^{-1}$ in a similar manner to the Newton type method described by Elasbrouty in [163, 171] but utilizing the Matrix Momentum algorithm to avoid the matrix inversion required by a standard Newton method.

3. Another potential solution would be to combine the above approaches to develop a Riemannian Trust Region Matrix Momentum based BSS solution.

In this chapter an algorithm has been introduced that gives similar performance to current second order methods with reduced computational complexity compared to standard gradient methods. In the following chapter an algorithm is introduced based on its simplicity from an implementation point of view.

Chapter 5

Simultaneous Perturbation Stochastic Approximation

5.1 Introduction

It has been seen throughout this thesis that optimization methods have been essential to solving the BSS problem. This will be extended further within this chapter with the introduction of Spall's Simultaneous Perturbation Stochastic Approximation (SPSA) method. The term Stochastic Approximation (SA) has become a standard term for techniques that try to either minimize or maximize a function observed in the presence of noise, or to find an approximation to the solution of an equation that has been observed in the presence of noise. This technique was first introduced by Herbert Robbins and his student, Sutton Monro in 1951 in their seminal paper [172] and extended further by Kiefer and Wolfowitz in [173]. The SPSA technique has been shown by Spall [174, 175, 176, 177] to provide a number of benefits when compared to these classical SA methods, including faster convergence and reduced computation. The SPSA algorithm has been applied extensively in a number of fields including optimal control [178,

179], neural network training [180], optimization of particle filters [181], traffic management [182, 183] and recently in application to the BSS problem [184]. The SPSA method applied within the context of the BSS problem as detailed within [184] is described within this chapter, and in the following chapter the SPSA algorithm is applied with a novel cost function in the context of Medical Image Registration as detailed in [185]. Therefore, within this chapter the classical SA methods will be introduced. This will be followed by an full explanation of the SPSA method. At this point the SPSA algorithm will be utilized within the context of the BSS problem and will be shown to be capable of approximately jointly diagonalizing a set of time delayed covariance matrices. First, the classical SA methods are introduced in the following section.

5.2 Stochastic Approximation Methods

As is standard with optimization methods, SA algorithms are required to find either the minimum or maximum value of some vector valued cost function $J(\mathbf{w})$. More specifically the algorithms are required to find the vector $\mathbf{w}$ that minimize or maximize this cost function. In the context of SA algorithms the cost function will generally be corrupted by an additive noise source. Skipping slightly ahead it can be seen that for application to the BSS scenario we require to extend the above SA framework to the matrix case, such that it is required to find the matrix $\mathbf{W}$ that will minimize or maximize a given cost function $J(\mathbf{W})$. The SA algorithms will initially be described for vector valued cost functions and then will subsequently be extended to matrix valued cost functions when utilized within the BSS context. The seminal algorithm in the field was the Robbins-Monro algorithm described in the following subsection.

5.2.1 Robbins-Monro Stochastic Approximation

In 1951 Herbert Robbins and his student, Sutton Monro developed an algorithm to estimate the roots of a regression equation with the benefit that the algorithm had guaranteed convergence properties [172]. This algorithm is known as the Robbins-Monro Stochastic Approximation (RMSA) algorithm. The algorithm makes the assumption that the objective or cost function is a differentiable function $J(\mathbf{w})$ with respect to the vector parameter $\mathbf{w}$. Noting that the roots of an equation can be found by taking the derivative of the objective function and setting the result equal to zero. Assuming that $J(\mathbf{w})$ is a differentiable function with respect to the matrix $\mathbf{w}$. This can be written as follows:

$$g(\mathbf{w}) = \frac{\partial J(\mathbf{w})}{\partial \mathbf{w}} = 0 \tag{5.1}$$

Assuming that the available measurement will be a noisy version of the above gradient $\mathbf{y}(\mathbf{w})$ at each iteration k will be given as follows:

$$\mathbf{y}(\mathbf{w})(k) = g(\mathbf{w})(k) + \nu(k) \tag{5.2}$$

Where ν represents an additive i.i.d. zero mean noise term. Placing the above equation in the context of a stochastic gradient equation we obtain the following equation for the weight vector update $\mathbf{w}$.

$$\mathbf{w}(k+1) = \mathbf{w}(k) - a(k)\mathbf{y}(\mathbf{w}) \tag{5.3}$$

The $a(k)$ in the above equation represents the step size parameter which, within this thesis, generally is denoted by μ but due to the preference within the Stochastic Approximation literature the term $a(k)$ will be utilized within this and the following chapter. The above method has the potential disadvantage for a stochastic approximation algorithm that an analytical gradient $g(\mathbf{w})$ must be known in advance. An alternative method that utilizes an approximation to calculate the gradient is defined in the following subsection.

5.2.2 Finite Difference Stochastic Approximation

To avoid the requirement that the analytical gradient be known in advance as described in the RMSA algorithm described above, Kiefer and Wolfowitz developed a method in [173], refined subsequently in [186], that calculates an approximate gradient value utilizing values of the cost function $J(\mathbf{w})$. This method was titled the Finite Difference Stochastic Approximation (FDSA) method as it utilizes a finite difference method to approximate the gradient of the given cost function. This update equation for the FDSA method is shown as follows.

$$\mathbf{w}(k+1) = \mathbf{w}(k) - a(k)\hat{g}(\mathbf{w}) \tag{5.4}$$

Where $\hat{g}(\mathbf{w})$ is an approximation of the gradient calculated from potentially noisy measurements of the cost function. The following equation represents a one sided finite difference gradient approximation. Within this equation the vector $\mathbf{e}$, represents a vector containing n elements, where the i^{th} element is set to one and all remaining elements are zero. The value $c(k)$ is a small constant that is annealed throughout the duration of the learning process.

$$\hat{g}_i(\mathbf{w}(k)) = \frac{J(\mathbf{w}(k) + c(k)e_i) - J(\mathbf{w}(k))}{c(k)} \tag{5.5}$$

Another alternative gradient approximation is the two sided version as given in the following equation.

$$\hat{g}_i(\mathbf{w}(k)) = \frac{J(\mathbf{w}(k) + c(k)e_i) - J(\mathbf{w}(k) - c(k)e_i)}{2c(k)} \tag{5.6}$$

For the vector valued case the chosen gradient approximation from the above equations requires one calculation of the cost function $J(\mathbf{w})$ for each of the n elements of the input vector for the one sided gradient approximation, and two cost function calculations for the two sided version. It can then be seen that extending the above gradient approximation equations to the case of a $n \times n$

matrix would subsequently require a minimum of n^2 calculations of the cost function $J(\mathbf{W})$ per algorithm iteration. Thus this approach has the disadvantage that the algorithm becomes computationally very complex as the dimensionality of the problem scales. This problem was the motivation for the development of the Simultaneous Perturbation Stochastic Approximation algorithm.

5.2.3 Simultaneous Perturbation Stochastic Approximation

As a method of reducing the computational cost and of improving the convergence speed of the above FDSA algorithm without requiring knowledge of the underlying analytical gradient as with the RMSA algorithm, Spall developed the Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm [174]. Spall presents general introduction to Stochastic Approximation algorithm and a complete coverage of SPSA in [177]. The development of the algorithm as with the previous algorithms begins with a potentially noise version of the cost function. The update equation for the vector valued case is given by the following equation.

$$\hat{g}(\mathbf{w}(k)) = \frac{J(\mathbf{w}(k) + c(k)\xi(k)) - J(\mathbf{w}(k) - c(k)\xi(k))}{2c(k)} \odot \begin{bmatrix} \xi_1^{-1}(k) \\ \vdots \\ \xi_m^{-1}(k) \end{bmatrix} \tag{5.7}$$

Where $\odot$ represents element by element multiplication and $\xi(k)$ is a random vector generated from a Bernoulli ± 1 distribution. The random vector has the following property.

$$E[\xi_i] = 0 \tag{5.8}$$

Utilizing the above properties and taking the Taylor series expansion of Equation 5.7, we obtain the following expansion.

$$\begin{aligned}\hat{g}_i(\mathbf{w}(k) &= \frac{1}{2c(k)\xi_i(k)}\Bigg[\Bigg[J(\mathbf{w}_i(k)) + c(k)\xi_i(k)\frac{\partial J(\mathbf{w}_i(k))}{\partial \mathbf{w}_i} \\ &+ \frac{(c(k)\xi_i(k))^2}{2!}\frac{\partial^2 J(\mathbf{w}_i(k))}{\partial \mathbf{w}_i^2} + O(n^3)\Bigg] \\ &- \Bigg[J(\mathbf{w}(k)) - c(k)\xi_{ki}\frac{\partial J(\mathbf{w}_i(k))}{\partial \mathbf{w}_i} \\ &+ \frac{(c(k)\xi_i(k))^2}{2!}\frac{\partial^2 J(\mathbf{w}_i(k))}{\partial \mathbf{w}_i^2} + O(n^3)\Bigg]\Bigg] \end{aligned} \tag{5.9}$$

Observing the above expansion it can be seen that Equation 5.7 gives an approximation of the gradient as with the above SA algorithms with a huge reduction in the computational cost when compared to the FDSA algorithm as the gradients for each element are calculated simultaneously, instead or requiring an iteration per element. This removes the requirement for knowledge of the analytical gradient as needed with the RMSA method. Extending the above Equation 5.7 to the matrix valued cost function as required in the BSS problem we obtain the following equation for each matrix element.

$$\nabla W_{ij}(k) = \frac{J(W_{ij}(k) + c(k)\xi_{ij}(k)) - J(W_{ij}(k) - c(k)\xi_{ij}(k))}{2c(k)\xi_{ij}(k)} \tag{5.10}$$

The above equation has the benefit that only two values of the cost function require to be calculated at each iteration, independent of the size of the input vector. This offers a huge computational improvement over the FDSA algorithm. As with the FDSA algorithm the value $c(k)$ is a small constant that is annealed throughout the learning duration, in [174, 177] the equation for the learning rate is given as follows.

$$c(k) = \frac{c}{(k+1)^{\gamma}} \tag{5.11}$$

A practically effective value for the γ parameter is given by Spall in [176] as 0.101. This value will be utilized within this thesis. The SPSA algorithm utilizes the same gradient update equation as the previous FDSA method given in 5.4. To improve the accuracy of the gradient calculation given in Equation 5.7 when utilized within the update Equation 5.4 an expectation of the gradient calculations is often taken. Yet as with standard gradient algorithms, the expectation is replaced by a sample average. This is given in the following equation.

$$\mathbf{W}(k+1) = \mathbf{W}(k) - a(k)\Big\{N^{-1}\sum_{n=0}^{N-1}(\nabla\mathbf{W})_n\Big\} \tag{5.12}$$

As with the perturbation constant $c(k)$ the learning rate parameter is also annealed during the learning process as given in [174, 177]. This is given as follows.

$$a(k) = \frac{a}{(k+1)^{\alpha}} \tag{5.13}$$

A practically effective value for the α parameter is given by Spall in [176] as 0.602, which will be utilized within this thesis. Now the SPSA algorithm and its associated parameters have been discussed, the application of the algorithm to the BSS problem is described in the next section.

5.3 BSS using SPSA Optimization

The first application of the SPSA algorithm within the context of the BSS problem was introduced by Ding et al. in [187, 188, 189] where SPSA was utilized to optimize a Mutual Information based cost function. The same authors demonstrated the application of the algorithm with a differing cost function in [190], where the diagonality of the nonlinear correlation function was utilized as the measure for optimization. Independently Maeda, at the same time been considering the application of the SPSA technique within the context of the BSS

problem, his first work [191] in this area was similar to Ding et al. in which he applied the SPSA framework to a Mutual Information based cost function. This work was subsequently extended by Maeda in [192] where a Natural Gradient based framework was embedded within the SPSA algorithm. In this thesis the SPSA algorithm is utilized to create a gradient based approximate joint diagonalization algorithm [184]. The approximate joint diagonlization cost function and its application within the BSS context is described within the following section.

5.4 Joint Diagonalization

Joint diagonalization of matrices is an extremely well utilized technique in the fields of Numerical Computation, Multivariate Statistics and Signal Processing, specifically in the context of ICA and the BSS problem. Utilizing the measure originally defined by Cardoso in [67] the joint diagonalization of a given set of N matrices $\mathbf{C}^l$ may be written in cost function format $J(\mathbf{W})$ as follows:

$$J(\mathbf{W}) = \sum_{l=1}^{N} off(\mathbf{W}\mathbf{C}^l\mathbf{W}^T) \tag{5.14}$$

Where the function $off(.)$ gives a measure of the diagonality of the resulting matrix, that is the sum of the squares of the off-diagonal elements of a given matrix $\mathbf{B}$. This is given by the following matrix equation

$$off(\mathbf{B}) = \sum_{i \neq j} b_{ij}^2 = ||\mathbf{B}||^2 - \sum_{i \neq j} b_{ii}^2 \tag{5.15}$$

The problem with utilizing Equation 5.14 as the minimization cost function for joint diagonalization is the trivial solution $\mathbf{W} = 0$, results in a minimum of the cost function. In order to avoid the trivial solution, constraints must be placed upon the matrix $\mathbf{W}$. Numerous techniques exist within the literature for constraining the optimization space when utilized in the BSS context, this is described within the following subsection.

5.4.1 Application of Joint Diagonalization to BSS

As described initially in chapter 2, Joint Diagonalization algorithms play an important role in two of the most heavily utilized methods within the BSS literature, that is the SOBI [55] and JADE [67] algorithms. Joint diagonalization is an extensively utilized technique within the BSS field. For the case of two matrices, the problem can be solved using a Generalized Eigenvalue Decomposition as described in [193]. Other methods will often require the diagonalization of multiple matrices. The following list details a number of the areas so far that have utilized Approximate Joint Diagonalization to solve the BSS problem.

1. Multiple time delayed correlation matrices [194, 195, 139, 55, 196, 197, 198, 199, 200]
2. Fourth order cumulant matrices [201, 67]
3. Second characteristic function [202, 203, 204]
4. Spatial time-frequency distribution matrices [205, 206]

One of the original methods for constraining the search space for the joint diagonalization algorithms was to first prewhiten the data as described in Appendix B, this is the approach taken in the SOBI [55] and its extensions WASOBI [197, 198], EWASOBI [199, 200], TDSEP [196], JADE [67]. Prewhitening is known to be efficient from a computational point of view for approximate joint diagonalization as this addition enables Jacobi based algorithms to estimate the required orthogonal factor of the matrix process. A problem that exists when using a prewhitening based approach is that errors introduced at this stage cannot be subsequently removed by higher order stages. This introduces a lower bound on the achievable estimation error between the true and estimated mixing matrices, this was initially shown by Cardoso in [207], and later and in further detail by DeLathauwer

et al. in [208]. To avoid the potential introduction of errors via the prewhitening stage a number of Approximate Non Orthogonal Joint Diagonalization algorithms were introduced. One of the initial approaches, utilizes the constraint that the matrices within the set of matrices to be diagonalized are positive definite [209]. Later this constraint was subsequently relaxed in [201, 210, 211, 212]. An alternative approach to joint diagonalization from the previous mentioned batch based close form approaches is to introduce a gradient based cost function. A comprehensive introduction to gradient based approximate joint diagonalization was introduced by Joho et al in [194, 195], where a number of constraint methods were introduced for avoiding the trivial solution for minimization of Equation 5.14. This idea was further extended by the same authors utilizing a constrained Newton based approach in [139]. The problem was approached from the context of Riemannian geometry by Ziehe et al. in [213] where a Natural Gradient based approach is undertaken. This Riemannian approach was continued via Asfari in [214, 215, 216] where Riemannian based gradient algorithms were shown for both the orthogonal and non-orthogonal joint diagonalization cases.

As has been demonstrated within this subsection, there exists a number of approaches for both limiting the optimization space required when implementing joint diagonalization algorithms and for avoiding the trivial solutions $\mathbf{W} = 0$. In the following subsection the procedure utilized within this thesis is described.

5.4.2 Joint Diagonalization using SPSA

It was described in the above subsection that the search space for the optimization algorithm must be constrained such that the trivial solution is avoided, numerous methodologies have been adopted to achieve this goal. Within this thesis the joint diagonalization Equation 5.14 is combined with a penalty term described within

[214], this is shown in the following equation.

$$J(\mathbf{W}) = \sum_{l=1}^{N} off(\mathbf{W}\mathbf{C}^l\mathbf{W}^T) - \log(\det(\mathbf{W})) \tag{5.16}$$

The term $\det(\mathbf{W})$ penalizes the cost function when the weights $\mathbf{W}$ are very small, therefore providing a method of avoiding the trivial solution, without the requirement that the matrix $\mathbf{W}$ be orthogonal, or the columns of the matrix are normalized at each algorithm iteration reducing the complexity of the resulting algorithm. The performance of the algorithms is demonstrated in the following section.

5.5 Simulations

In order to demonstrate the performance of the algorithm, the SPSA-JD algorithm will first be utilized to diagonalize a set of perfectly diagonalizable matrices. The algorithm will then be applied to the diagonalization of a set of time delayed correlation matrices of speech signals for application to the instantaneous BSS problem.

5.5.1 Separation of perfectly diagonalizable matrices

Before demonstrating the application of the SPSA-JD algorithm developed above to the BSS problem where the cost functions (either multiple correlation or cumulant matrices) are not exactly diagonalizable, it is first shown that for the case where the set of matrices is exactly diagonalizable the SPSA-JD algorithm gives good performance. To demonstrate this, the following setup originally detailed in the simulation of the FFdiag [217, 218] and further utilized within the QDIAG algorithm detailed in [219] is implemented. A set of N diagonal matrices $\mathbf{C}^l$ is created such that the elements on the diagonal are produced from a standard zero

mean unit variance Gaussian distribution for each matrix $\mathbf{C}^l$ where $1 \leq l \leq N$. The procedure to generate a set of N perfectly diagonalizable matrices $\mathbf{R}^l$ is to premultiply the set of diagonal matrices $\mathbf{C}^l$ by a randomly generated matrix $\mathbf{A}$, and postmultiply by its transpose $\mathbf{A}^T$. This matrix has again been produced from a standard zero mean unit variance Gaussian distribution, this is shown as follows.

$$\mathbf{R}^l = \mathbf{A}\mathbf{C}^l\mathbf{A}^T \tag{5.17}$$

The above generated set of matrices $\mathbf{R}^l$ can be jointly diagonalized by any matrix that differs from the matrix $\mathbf{A}^{-1}$ by a row permutation or a row scaling. To demonstrate the performance of the algorithm the algorithm is compared with the FFdiag [217, 218], QDIAG [219] and ACDC [220] algorithms. The performance measure utilized is a normalized version of the cost function given in Equation 5.14, where the normalization factor is the number of off-diagonal elements given by $n^2 - n$ where n represents the size of the $n \times n$ matrices. The performance of the algorithm is given in the following diagram.

It can be see that for the above task the SPSA-JD algorithm is outperformed by the FFdiag [217, 218] algorithm and the QDIAG [219], but performs well against the ACDC [220] algorithm. It was not shown in the above diagram for space reasons but the ACDC algorithm was found to converges after approximately 3000 iterations. A similar result to the above analysis was shown in [213] where the Natural Gradient is applied within the joint diagonalization cost function space. Therefore it can be judged from this that while gradient based joint diagonalization offer good performance when applied to perfectly diagonlizable matrices they are outperformed by close form methods such as FFdiag [217, 218] and the recently developed QDIAG [219]. It was shown within this subsection that the SPSA-JD algorithm developed above is capable of jointly diagonalizing a set of perfectly diagonalizable matrices with good performance. In the following

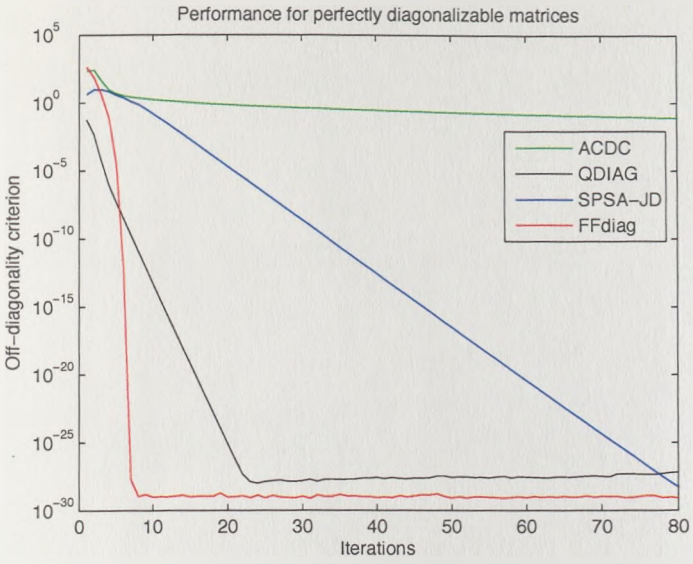


Figure 5.1: Algorithm performance for perfectly diagonlizable matrices

subsection this concept is extended to the BSS problem.

5.5.2 Separation of a mixture of speech signals

In order to test the algorithm developed in this chapter in the BSS context the following simulation scenario is utilized. To solve the problem the SPSA-JD algorithm will be applied to the diagonalization of the time delayed correlation matrices. The fourth order cumulant matrices, second characteristic function and the spatial time-frequency distribution matrices could also have been utilized as the optimization criterion within this framework, yet the time delayed correlation matrices were chosen as their calculation requires the lowest computational complexity when compared with the other methods. Three speech signals sampled at 11025Hz consisting of 20000 samples are utilized as the input sources signals, these are shown in Figure 5.2. The signals are mixed using randomly mixing matrix, generated from a Uniform distribution, an example of this mixing is shown

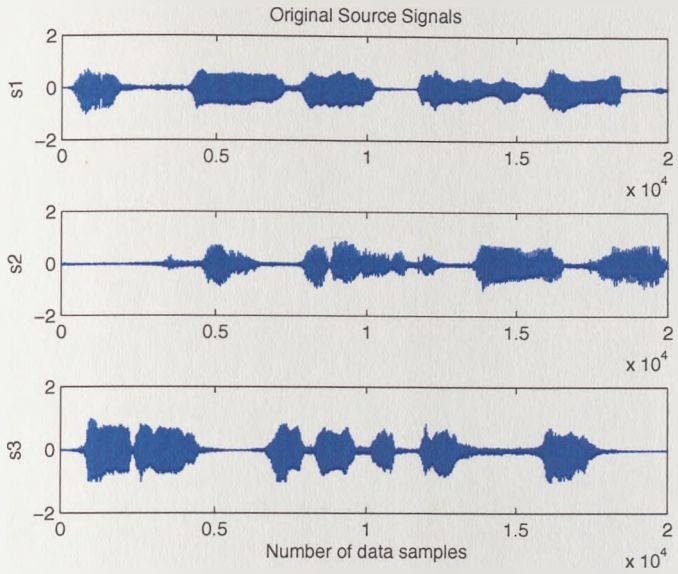


Figure 5.2: Speech source signals

in Figure 5.3. To achieve separation the algorithm approximately jointly diago-

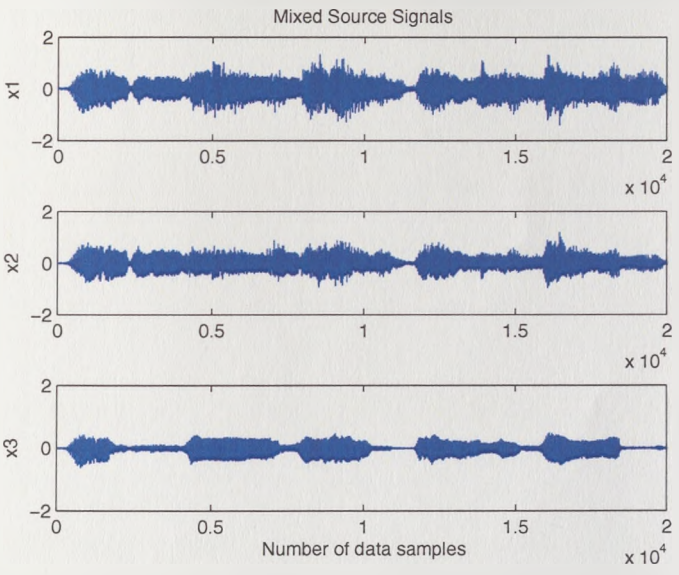


Figure 5.3: Mixed speech source signals

nalizes 10 time delayed correlation matrices. The correlation matrices calculated

over K samples are given by the following formula:

$$\mathbf{R}_{\mathbf{xx}}^{l(\tau)} = \frac{1}{N-\tau} \sum_{k=0}^{K} \mathbf{x}(k)\mathbf{x}(k-\tau) \tag{5.18}$$

The time delays for each correlation matrix $\mathbf{R}_{\mathbf{xx}}^{l(\tau)}$ given in the above Equation were chosen as $l(\tau) = [1, 3, 5, 9, 11, 13, 15, 17, 19, 21]$, these delays were chosen heuristically and many other potential combinations are available, an optimal number and the time delays of the correlation matrices to be jointly diagonalized is still an open research question. An example of the separation output of the example is shown in Figure 5.4. As the calculated correlation matrices are not

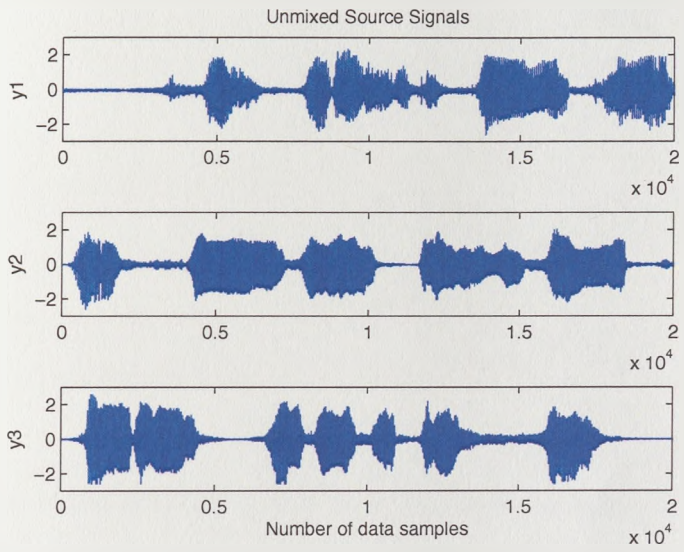


Figure 5.4: An example of the output signals

exactly diagonalizable, the joint diagonalization cost function given in Equation 5.14 is inappropriate for assessing the performance of the algorithm within the BSS context. To show the performance of the algorithm Amari's Performance Index is utilized, this is shown in Figure 5.5. The final performance metric for the algorithm is to demonstrate that when the algorithm has converged, the

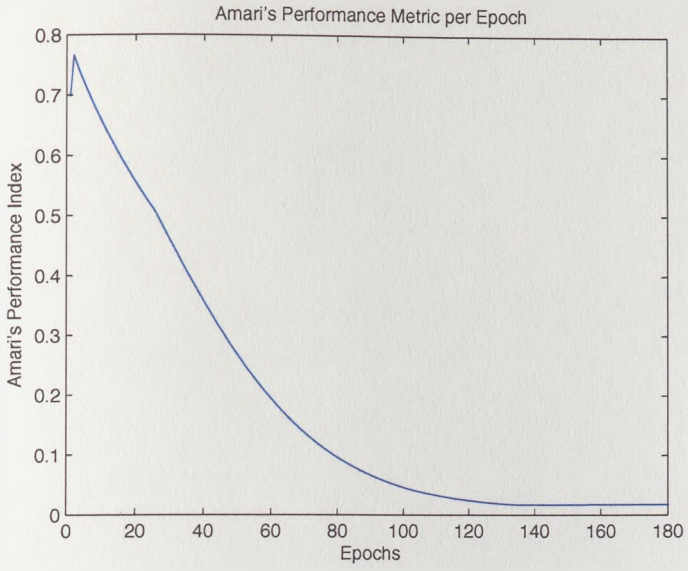


Figure 5.5: Amari's Performance Index

product of the mixing and unmixing matrices represents an identity matrix up to a permutation and a scaling. This is shown for an example run in the following Figure 5.6. Thus in this chapter the SPSA optimization framework has been utilized to develop an approximate joint diagonalization algorithm, this algorithm has been named the SPSA-JD method. It was shown that for the case of perfectly separable matrices, the algorithm provides good performance. The algorithm was then shown to perform well in the BSS context for the separation of time delayed correlation matrices.

5.6 Conclusions

In this chapter the SPSA algorithm introduced initially by Spall has been introduced and applied to the joint diagonalization of a set of matrices for application within the BSS context. This chapter began by giving an introduction to the Stochastic Approximation methods, specifically the Finite Difference method, it

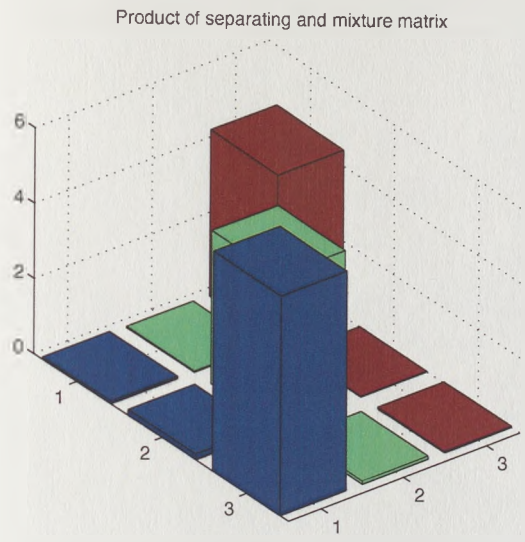


Figure 5.6: Combination of the mixing and unmixing matrices

was then shown that the SPSA method vastly reduces the computational complexity when compared with the Finite Difference method. Previous applications of the SPSA method within the BSS context were discussed. Joint diagonalization methods were then introduced, before combining with the SPSA method to create the SPSA-JD algorithm. The performance of this algorithm was then shown for perfectly diagonalizable matrices, then further demonstrated in BSS context. To further improve the SPSA-JD algorithm the following approaches are suggested.

1. One of the problems that currently exists with the SPSA method is the requirement to choose both the step size parameter $a(k)$ and the perturbation parameter $c(k)$. Utilizing the basic step size and perturbation methods described by Spall in [177] has the problem that these parameters may converge before the underlying algorithm has converged resulting in massive misadjustment, making the choice of these parameters an art within itself.

This creates huge potential for improving the SPSA algorithm by developing novel step size and perturbation algorithms.

2. To further improve the convergence of the SPSA-JD algorithm the second order SPSA algorithm, known as the adaptive SPSA algorithm, introduced initially in [221] and further described within [177, 222, 223] could be implemented in place of the standard SPSA method. This method represents the stochastic equivalent of the Newton-Raphson algorithm discussed earlier within this thesis. It is thought that this method could be utilized to drastically improve the convergence properties of the SPSA-JD algorithm.

3. One of the problems that is common to all methods that utilize the diagonalization of the time delayed correlation matrices is the selection of the number of matrices to diagonalize, and the delays at which to select these matrices. A genetic based time lag selection algorithm for the TDSEP algorithm [196] was introduced in [224], this approach provides good performance yet genetic algorithms are known to be computationally intense, therefore the proper selection of the number and time lag of associated matrices is still a difficult research question.

4. A final possibility would be to change the optimization methods utilized within this chapter. One of the potential replacement candidates would be the Complex Step Derivative optimization framework described by Martin in [225, 226]. This method provides a way of estimating the derivative of real valued function $J(\mathbf{W})$ numerically utilizing the following formula.

$$\frac{\partial J(\mathbf{W})}{\partial \mathbf{W}} \approx \frac{Im[J(\mathbf{W} + i\Delta\mathbf{W})]}{\Delta\mathbf{W}} \tag{5.19}$$

Where the i in the above equation creates a complex valued perturbation $\Delta\mathbf{W}$, and the operator $Im[.]$ returns the imaginary part of the created

complex function. This approach has the potential drawback that the step size parameter to utilize the gradient within a steepest descent algorithm has to be chosen, and the algorithm in its current form is only applicable to real valued sources.

In this chapter the SPSA algorithm has been utilized in the BSS context to develop a joint diagonalization based solution. In the next chapter the SPSA algorithm is utilized to optimize a number of Information Theoretic cost functions to provide a novel solution to a medical image registration problem.

Chapter 6

Image Registration Using SPSA

In the previous chapter the SPSA algorithm was introduced in the context of Joint Diagonalization and its application to the BSS problem. It was found during the course of this research that the Information Theoretic cost functions based on mutual information used to solve the Image Registration problem were identical to those utilized within the ICA and BSS communities [227, 228, 229, 230, 88, 118]. It was then found that the SPSA algorithm was specifically suitable for optimization within this framework. Within this chapter the first application of the SPSA algorithm to Medical Image Registration is introduced, this is also reinforced with the novel application of both Renyi's and Tsallis's mutual information measures [231, 232].

6.1 Image Registration

Image Registration as initially introduced in chapter 1 is the process utilized to apply a transformation to a pair of images that results in the maximum accuracy between the two images, where the two images have originally been captured via different imaging methods or captured at differing time points. The main

application for Image Registration is the suppression or cancellation of geometric distortions between a given Reference Image $R(i,j)$ and the Sensed Image, generally referred to as the Floating Image $F(i,j)$. This is addressed by finding a transformation T that maximizes the alignment of the images $R(i,j)$ and $F(i,j)$. Details of the transformation T will be explained later within this chapter. There exists a huge number of applications for Image Registration, including some typical applications in Computer Vision [233, 234] , Remote Sensing [235, 236, 237] and Medical Imaging [185, 238, 239]. Some common features of Image Registration algorithms are discussed in the following subsection.

6.1.1 Image Registration Approaches

A vast number of approaches for tackling the Image Registration problem, it was shown by Brown [240] and in several subsequent Image Registration survey papers [241, 238, 239, 242] that the majority of these algorithms share the four following distinct algorithm components.

1. Feature Detection - This work involves obtaining useful components from the images to be utilized within the registration process. Examples of features utilized within Image Registration algorithms are Edges [243, 244, 245], Curvature [246, 247], Corresponding Points [248, 249]. Within this thesis the features utilized within the Image Registration algorithm are the Histograms of the individual images and the Joint Image Histogram [230, 185], these are utilized to approximate the underlying pdfs of the individual images for application in the mutual information based cost functions as explained in section 6.2.

2. Search Space - This space contains the geometrical transformations that will be applied to the floating image $F(i,j)$ to register with the reference

image $R(i, j)$. There exists a number of potential transformations that can be applied, yet these can broadly be split into two categories, Rigid and Non-Rigid Transformations [250, 251, 252]. Rigid transformations are generated via a combination of image translations and rotations, and are often extended to include Affine transformations that extend this transformation to include scalings and shearings. These transformations are applied globally to the image. Non-Rigid Transformations allow local deformations of the image, these are sometimes referred to as Elastic Transformations [253, 254]. Within this thesis only Rigid Transformations are considered, these are described in subsection 6.1.2.

3. Similarity Metric - The Similarity Metric is the specific measure utilized to gauge the degree of similarity between two images to be registered. A number of measures have been proposed within the Image Registration field, including Cross Correlation [240, 255], Sum of Squared Differences (L_2 Norm) [256, 257] and mutual information based approaches [227, 228, 229, 230, 239, 185]. In this thesis the mutual information based approaches are extended by utilizing both Renyi [231] and Tsallis [232] based mutual information measures. This is explained further in section 6.2.

4. Optimization Algorithm - This component represents the algorithm utilized to maximize the similarity measure. A number of Optimization algorithms have been utilized within the context of Multiresolution based Image Registration [258], these included Steepest Descent [108, 127], Conjugate-Gradient [259, 56], Quasi-Newton [260, 261] and Levenberg-Marquardt [262, 263, 264]. In this thesis Spall's SPSA algorithm [174, 175, 176, 177] detailed within the previous chapter for application to the Blind Source Separation problem, is applied as the optimization algorithm, used to max-

imize the similarity between the two images to be registered.

As has been stated above, there exists a number of potential geometrical transformations that can be applied within the image registration context. Within this thesis only Rigid Body transformations are considered, specifically consisting only of Translation and Rotation, as this will allow an accurate comparison of the novel mutual information based cost functions presented within this thesis and current algorithms based on Shannon's mutual information. Rigid Body Transformations are described as follows.

6.1.2 Rigid Body Transformations

Rigid Body Transformations represent a subset of Affine Transformations. Affine Transformations have the property that when applied to an image, straight lines remain straight and parallel lines are preserved but rectangles within the image may potentially become parallelograms. The transformation is applied to the image globally, and no deformation of the image is performed. Affine Transformations are generated from the following Image operations.

1. Rotation - A rotation around the centre of the image
2. Translation - A displacement of every point in the image by a constant distance
3. Scaling - A magnification or shrinkage of the image
4. Shearing - A shearing transformation results in an image that appears that it has been pushed in a direction that is parallel to the coordinate axis

These image operations are demonstrated in Figure 6.1 on a synthetic image. Within this thesis the following Equation is utilized to generate the transformation T that simultaneously performs an image translation t_x in the x-axis, an

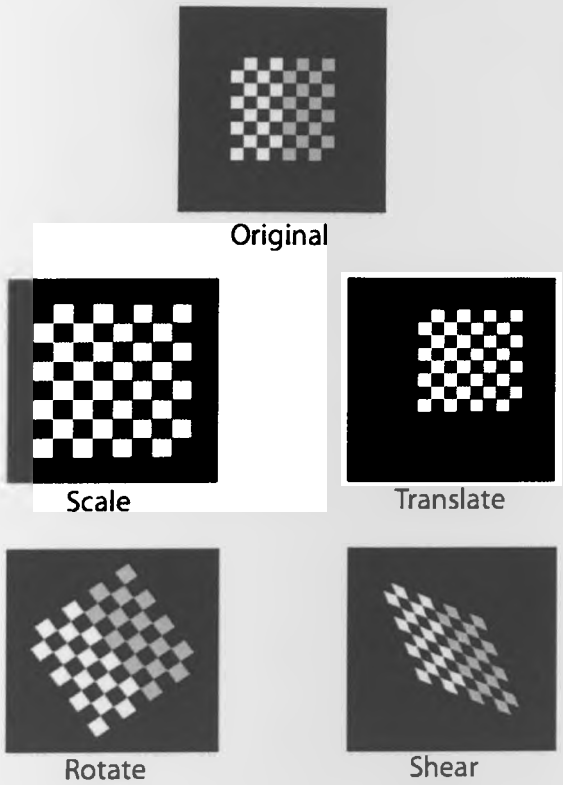


Figure 6.1: Affine Transformation Image Operations

image translation t_y in the y-axis, and a rotation of the image by angle θ. The quantities (x_F, y_F) and (x_R, y_R) represent the x and y coordinates of the Floating and Reference Image respectively.

$$\begin{bmatrix} x_F \\ y_F \end{bmatrix} = \begin{bmatrix} s.cos(\theta) & -s.sin(\theta) \\ s.sin(\theta) & s.cos(\theta) \end{bmatrix} \begin{bmatrix} x_R \\ y_R \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \tag{6.1}$$

This can be written compactly as one matrix equation as follows

$$\begin{bmatrix} x_F \\ y_F \\ 1 \end{bmatrix} = \begin{bmatrix} s.cos(\theta) & -sin(\theta) & t_x \\ sin(\theta) & s.cos(\theta) & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_R \\ y_R \\ 1 \end{bmatrix} \tag{6.2}$$

Which for simplicity we can write in parametrized form as $T = [s, t_x, t_y, \theta]$, resulting in the following:

$$\begin{bmatrix} x_F \\ y_F \end{bmatrix} = T \begin{bmatrix} x_R \\ y_R \end{bmatrix} \tag{6.3}$$

Within this thesis it is assumed that the scaling parameter is 1, and can be removed from further consideration. In Figure 6.2 a flow chart representation of the Image Registration system developed within this thesis is detailed.

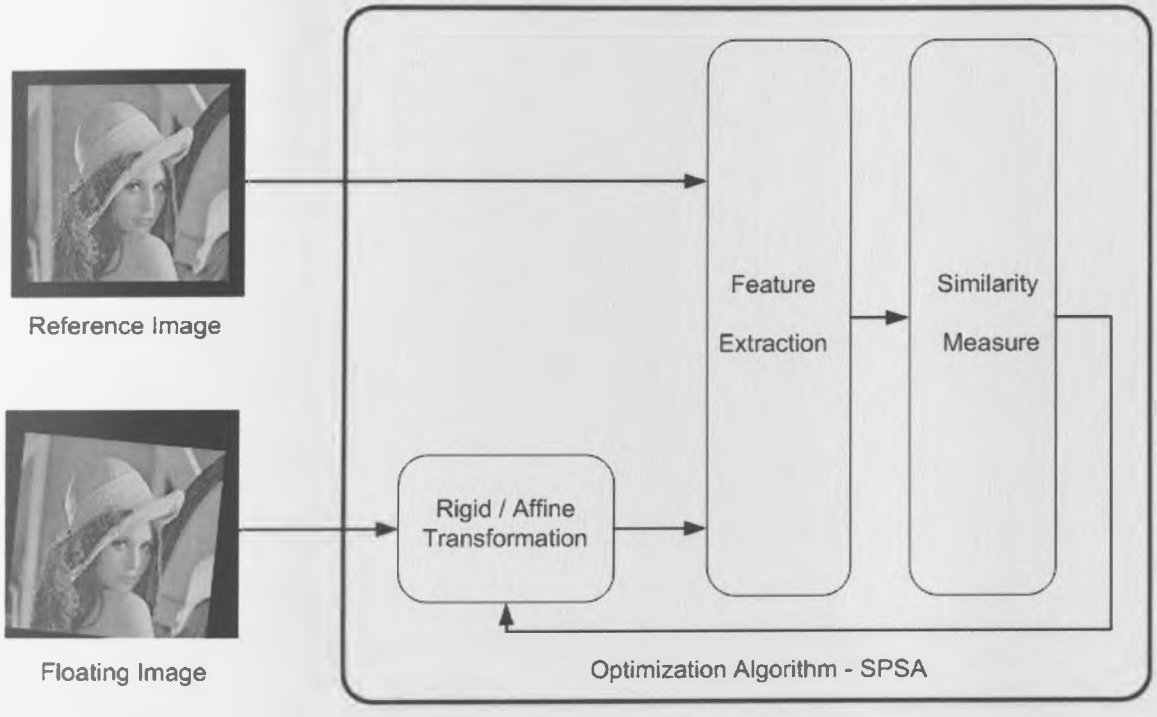


Figure 6.2: Flowchart for Image Registration system

In the following section an introduction to Image Registration via maximization of mutual information is developed.

6.2 Image Registration by Maximization of Mutual Information

Image Registration by maximization of mutual information has generated a large quantity of literature especially in medical imaging [238, 239]. This approach was introduced initially by Viola in [227, 228, 229] and independently by Collignon [265], then subsequently by Maes [230], Pluim [266] and Studholme et al. [267]. These algorithms offer an improvement in both convergence speed and computational complexity when compared with traditional correlation based Image Registration algorithms [268, 269]. Today a huge amount of research is still continuing in this field making it an ever expanding area. As stated above in section 6.1 registration consists finding the optimum transformation T, which will best align the images $R(i,j)$ and $F(i,j)$ with i and j being their coordinates. R being the reference image and F the floating image such that $F(T(i,j))$ should fit R. In this thesis Rigid Body transformations are applied to 2D images as detailed within subsection 6.1.2. In standard mutual information based Image Registration approaches [227, 228, 229, 230, 239, 185] the measure of similarity between the two images being registered is computed by finding the value of the mutual information associated to the pixel intensity distribution of the images. Within this chapter the pixel intensity distributions are calculated using the individual and joint image histograms, these could be calculated utilizing Parzen Density Estimation [270] or an alternative Probability Density function estimation algorithm [271, 272]. The definition of mutual information is based on the Relative Entropy or Kullback-Leibler distance [51], is described as:

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{6.4}$$

with X and Y are two discrete random variables representing the pixel intensity, $p(x)$ and $p(y)$ respectively, are their marginal probability density functions and $p(x, y)$ is their joint probability density. This mutual information is based on the classic Shannon definition of entropy [49, 50, 52].

6.2.1 Registration Algorithm

Different measures of divergence based on generalized entropies are proposed in the registration algorithm to produce alternative ways to process the mutual information between the images R and F. They provide a measure of the statistical dependence between the distribution of the image intensities of pixels in both images. The images are transformed according to a Rigid Body Transformation scheme as described in subsection 6.1.2. Writing the transformation utilized above in Equation 6.2 as a transformation vector $T = [t_x, t_y, \theta]$ which contains the parameters corresponding to an image translations over x and y axes and a image rotation as described in subsection 6.1.2, the parameters can now be utilized directly in an optimization algorithm framework. If the output of the transformation T results in non-integer coordinates for the Floating image $F(i, j)$, then a mapping is required to a set of integer coordinates. This interpolation procedure can be thought of as a resampling of the Floating image $F(i, j)$ [273]. The technique chosen to fit the floating image $F(T(i, j))$ to the grid of the reference $R(i, j)$ after each transformation is based on Spline Interpolation [274, 275, 276]. This method was chosen as in Lehmann et al. survey paper on interpolation in Medical Imaging it was found that Spline interpolation offered the smallest interpolation error without a great increase in computational complexity when compared with other methods such as nearest neighbour, linear, quadratic, Lagrangian and Gaussian interpolation [277].

6.2.2 Measures of Divergence and Mutual Information

Shannon's mutual information based Image Registration algorithms [227, 228, 229, 230, 239] have been shown to provide excellent performance in comparison to Correlation based measures [268, 269]. Since the development of Shannon's mutual information a number of different definitions of Entropy and measures of divergence have been investigated [52, 278]. One Entropy measure which is famous due to its popularity in the Physics domain was introduced by Tsallis in [232]. Tsallis presented a form of nonextensive entropy to describe a large class of physical phenomena. The divergence measure proposed by Tsallis is given by:

$$D_{Ts}(p \parallel q) = \frac{1}{1-\alpha}(1 - \sum_i \frac{p_i^\alpha}{q_i^{\alpha-1}}) \tag{6.5}$$

with $\alpha \in \mathbb{R} - \{1\}$. If p_i and q_i are replaced respectively by the joint probability $p(x, y)$ and by the product of the marginal distribution $p(x).p(y)$, then the mutual information based on Tsallis definition of entropy is obtained, which can also be described as:

$$\begin{aligned} I_{Ts}^\alpha(x;y) &= H_{Ts}^\alpha(x) + H_{Ts}^\alpha(y) - \\ &- (1-\alpha)H_{Ts}^\alpha(x)H_{Ts}^\alpha(y) - H_{Ts}^\alpha(x,y) \end{aligned} \tag{6.6}$$

where Tsallis's form of entropy $H_{Ts}^\alpha(x)$ of order α is given via the following equation.

$$H_{Ts}^\alpha(x) = \frac{1}{(1-\alpha)}(\sum p(x)^\alpha - 1) \tag{6.7}$$

Another form of entropy similar in form to Tsallis entropy has been proposed earlier by Renyi in [231], which results in the following divergence measure is given by the following equation:

$$D_{Re}(p \parallel q) = \frac{1}{1-\alpha}[\log \sum_i \frac{q_i^{\alpha-1}}{p_i^\alpha}] \tag{6.8}$$

with $\alpha \in \mathbb{R} - \{1\}$. As with the Tsallis divergence measure if p_i and q_i are replaced respectively by the joint probability $p(x, y)$ and by the product of the marginal distribution $p(x).p(y)$, then the mutual information based on Renyi definition of entropy is obtained, which can be described as:

$$I^{\alpha}_{Re}(x; y) = H^{\alpha}_{Re}(x) + H^{\alpha}_{Re}(y) - H^{\alpha}_{Re}(x, y) \tag{6.9}$$

where Renyi's form of entropy $H^{\alpha}_{Re}(x)$ of order α is given via the following equation.

$$H^{\alpha}_{Re}(x) = \frac{1}{(1-\alpha)}(\log \sum p(x)^{\alpha}) \tag{6.10}$$

It is important to note that when $\alpha \rightarrow 1$, using L'Hopital rule, Tsallis and Renyi definitions tend towards the Shannon expression of entropy. In [185] the first application of Tsallis Entropy as a divergence measure for optimization as an Image Registration criterion was demonstrated, this work will be further explained within this chapter. Renyi's divergence measures had been utilized in [279] in a Fourier based registration method. In the following section the Shannon [49, 50, 52], Renyi [231] and Tsallis [232] based information measures are utilized within the SPSA algorithm as the Optimization criterion to perform Image Registration.

6.3 Image Registration using SPSA

The SPSA gradient free Optimization framework introduced initially by Spall in [174, 175, 176, 177] was described in detail in the previous chapter. The SPSA algorithm was initially applied to the optimization of Shannon's mutual information for Image Registration by Cole-Rhodes in [236, 280, 237] for Image Registration applications in Remote Sensing. Cole-Rhodes then extended this work to utilize Spall's second order SPSA [177, 222, 223] in [281]. The SPSA

algorithm has subsequently been applied in [282, 283, 253] again for optimization of Shannon's mutual information as the given cost function. As stated above within this thesis the first application of Tsallis mutual information to the Image Registration problem is demonstrated, along with the novel combination of both Tsallis [232] and Renyi's [231] mutual information with Spall's SPSA algorithm [174, 175, 176, 177]. To utilize the Information measures within the SPSA algorithm the following stochastic update equation for the transformation vector T is generated, as described in the previous chapter in Equation 5.12.

$$T(k+1) = T(k) + a(k)\hat{g}(T(k)) \tag{6.11}$$

In the above equation the gradient vector $\hat{g}(k)$ for the parameter space $T = [t_x, t_y, \theta]$ is calculated using the following equation for each parameter.

$$\hat{g}_i(T(k)) = \frac{J(T + c(k)\xi_i(k)) - J(T - c(k)\xi_i(k))}{2c(k)\xi_i(k)} \tag{6.12}$$

The cost function $J(T)$ used in the optimization is either given by the Tsallis mutual information defined in Equation 6.7, or the Renyi mutual information defined in Equation 6.9. The transformation T is then applied to the image at each algorithm iteration. The parameters $a(k), c(k)$ and ξ are chosen as described previously in subsection 5.2.3. In the following subsection the performance of the algorithm is illustrated.

6.3.1 Automated Registration Algorithm

In this subsection, the three different measures of similarity were tested based on the classic Shannon mutual information and on its Tsallis [232] and Renyi [231] forms. The reference image for the experiment is a 512 by 512 pixels $t2$ Magnetic Resonance Image (MRI) with 16 bits gray scale levels. To demonstrate the performance of the algorithms and to compare between the mutual information measures the approach initially utilized by Maes in [230] is undertaken.

Figure 6.3: Reference and Floating images used for the registration algorithm. To demonstrate algorithm performance the floating image is created from the Reference image plus speckled noise.

The floating image is the same image but corrupted with uniformly distributed random multiplicative noise with mean 0 and variance equal to 0.04. The images which are utilized for the validation of the above algorithms are displayed in the Figure 6.3.

Before running the optimization process, the floating image is transformed with an initial vector $T = [t_x, t_y, \theta] = [10, -5, 15]$ and no optimization is carried out on the scaling parameter. In the case of multimodality optimization within medical imaging the pixel resolution is often a well known parameter so the pixel dimension can be fixed before the optimization of the translations and the rotation, removing the requirement to optimize the scale parameter. The algorithm is required to output a solution as close as possible to $T = [0, 0, 0]$.

6.4 Implementation Results

Table 6.1 provides a summary of the simulations, which have been carried out during the experiment. If the absolute average error between the results and the

Type of Entropy		*Results*			*Conv.*
Type	α	t_x	t_y	θ	*Iterations*
Shannon		0.0081	0.0118	0.0316	300
Tsallis	0.9	0.0069	0.0086	0.0118	100
Tsallis	0.8	0.003	0.008	0.0028	042
Tsallis	0.5	4.2386	10.8246	-99.6618	none
Renyi	0.9	0.0055	0.0046	0.0203	295
Renyi	0.5	0.0039	-0.0019	0.0075	380

Table 6.1: Registration results.

expected solution is computed, the most precise registration after 500 iterations is achieved with Tsallis mutual information set with $\alpha = 0.8$ and Renyi set with $\alpha = 0.5$. According to the results of the Table 6.1, Tsallis can achieve a sharper registration than Shannon based mutual information approaches. The registration algorithm converges to the global solution except when it is parameterized with Tsallis definition and $\alpha = 0.5$. When utilizing Tsallis mutual information it is important to take into consideration that if α is too low this may cause the algorithm to diverge, if $\alpha = 1$ then Shannon mutual information is simply applied. The α parameter needs to be set up according to a trade-off between the convergence speed and precision of the registration. Renyi's mutual information provided no improvement in algorithm convergence but resulted in slightly similar registration accuracy to the Tsallis definition. Using Renyi if α is decreased, the optimization algorithm converges slowly to the global solution, and was found not to diverge. Thus with both Tsallis and Renyi's mutual information measures the choice of the α parameter is crucial to the algorithm performance.

The different definitions for mutual information are assessed in terms of precision and accuracy but also comparing the number of iterations needed to converge

to the global solution. Figure 6.4 represents the behaviour of the θ parameter during the 500 first iterations of the optimization process. Observing the variation of this parameter can inform us on how fast the algorithm converges to the global solution. The initial transformation, which was applied to the floating image, was equal to $T = [10, -5, 15]$. It is also supposed that the three transformation parameters converge approximately at the same time to the solution, which has been practically verified. Referring to the Figure 6.4, the fastest data set to converge was observed when using Tsallis mutual information with $\alpha = 0.8$ in approximately 42 iterations. This result when compared to Shannon mutual information, which converges in 300 iterations, achieves a speed up of approximately 7 times, this represents a very significant speedup in the algorithm convergence. Tuning the α parameter results in modifying the average time of convergence. However if α is not correctly set up, the algorithm may not converge at all and result in a false registration. The following section concludes the Image

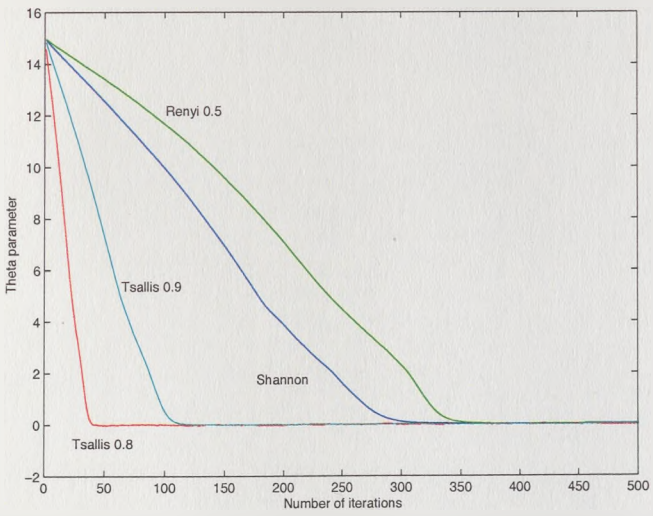


Figure 6.4: Convergence of the θ transformation parameter in function of the number of iterations of the optimization algorithm for different definitions of mutual information.

Registration component of this thesis.

6.5 Conclusion

In this chapter the first application of both Renyi's and Tsallis's entropies as an optimization criterion for solving the problem of Image Registration was introduced, utilizing Spall's SPSA algorithm. It was shown that the Tsallis entropy based information measures out-performed the convergence achieved using Shannon's mutual information, Renyi's information measure was found to demonstrate no convergence improvement compared with Shannon's measure. To further improve the algorithms developed within this chapter the following approaches are suggested.

1. As was stated in the previous chapter, one of the methods that has recently been introduced to improve the convergence properties of the SPSA algorithm is Spall's second order extension to the standard SPSA algorithm [177, 222, 223]. This approach has been successfully undertaken by Cole et al in [281] in the context of registration of multi-temporal satellite images, where Shannon's mutual information was utilized as the similarity measure. It is thought that the combination of the second order optimization criterion with either Tsallis or Renyi's mutual information measures would result in even faster performance than has been currently demonstrated.

Within this chapter the SPSA algorithm has been combined in a novel manner with Tsallis and Renyi's mutual information Measures to produce algorithms for application to problems in Medical Image Registration. In the following chapter this thesis is brought to its conclusion, and suggestions for further work are re-emphasized.

Chapter 7

Conclusion

As all things must come to an end, this chapter provides an overview of the achievements developed within this thesis, a summary of the work contained within this thesis, some future problems and outstanding issues are discussed, bringing this thesis to its conclusion.

7.1 Achievements

The aim of this thesis was to develop novel adaptive algorithms for application to the problem of Blind Source Separation (BSS). During the development of this process it was also found that there was significant overlap between the cost functions and optimization procedures utilized within BSS and the problem of mutual information based Image Registration. This led to the following resulting contributions.

- The first application of the Matrix Momentum algorithm to the BSS problem was shown [26, 27]. This combined the exact Hessian of the InfoMax cost function with the Matrix Momentum algorithm to develop an algorithm with Newton like performance.

- A novel application of the SPSA algorithm to the joint diagonalization of a set of matrices is demonstrated, it is then shown that this can be utilized to solve the BSS problem by jointly diagonalizing the time delayed correlation matrices of the observation vector $\mathbf{x}(k)$.

- The first application of Tsallis mutual information measure [232] to the Image Registration problem.

- The first combination of the SPSA algorithm with Renyi [231] and Tsallis [232] based mutual information measures for application to the Image Registration problem.

These achievements were detailed within this thesis as follows.

7.1.1 Thesis Summary

This thesis was split into 7 chapters. The first chapter introducing the topics of research, the Blind Source Separation and Image Registration problems. A review of the topics discussed within this thesis is given, and the original publications developed during the course of PhD research are detailed.
The second chapter develops the theory utilized in solving the BSS problem, specifically using ICA and contrast function optimization. The second chapter concludes with introductions to some of the most fundamental algorithms that have been developed within the ICA field. These algorithms form the basis set for the development of new algorithms and improvements of existing ones.
The third chapter details one of the most prominently used Neural Network approaches to the BSS problem, the Information Maximization (InfoMax) algorithm [104, 105]. After the development of this algorithm it was noticed by Amari that the algorithm convergence and computational complexity could be reduced

by post multiplying the gradient by the positive definite matrix $\mathbf{W}^T\mathbf{W}$. This represents the Natural Gradient operator within the parameter space of square non-singular matrices [101, 117, 114, 112]. The Mathematical description of this algorithm was detailed. At the same time Amari introduced the Natural Gradient extension to the InfoMax algorithm Cardoso and Lahleld had independently developed an equivalent gradient method with a prewhitening extension they labeled the Relative Gradient algorithm [122, 123]. This algorithm is also shown for completeness.

The fourth chapter introduces the application of the Matrix Momentum gradient optimization algorithm [141, 142] to the BSS problem is presented. The Matrix Momentum algorithm provides a Newton type method with reduced computational complexity, compared with methods requiring a direct matrix inversion. It was found that combining the Matrix Momentum algorithm with Pearlmutter's Hessian vector product given in Equation 4.22 as initially suggested by Orr [141] was found to be unsuitable within the BSS context, as the algorithm suffered from instability. To avoid these instability problems the exact Hessian was calculated and utilized within the Matrix Momentum framework. The algorithm is shown to provide fast convergence with low computational complexity.

The fifth chapter introduces the application of Spall's Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm [174, 177] to the joint diagonalization of a set of matrices. Initially, Stochastic Approximation (SA) algorithms are introduced, specifically the Finite Difference Stochastic Approximation (FDSA) algorithm. The SPSA algorithm is then compared with the FDSA algorithm in terms of calculations per iteration and is shown to require significantly less computation [174, 177]. The first application of the SPSA algorithm to the problem of joint diagonalization of matrices is then introduced. This is then utilized in the BSS context to jointly diagonalize a set of time delayed correlation matrices.

Finally the sixth chapter introduced the application of the SPSA algorithm [174, 177] described in the previous chapter to the Image Registration problem. The algorithm is combined with Shannon [49], Renyi [231] and Tsallis [232] mutual information measures. This representing the first application of Tsallis entropy to the Medical Image Registration [185], and the first combination of Renyi and Tsallis entropy with the SPSA algorithm. It was demonstrated that Tsallis entropy resulted in the fastest convergence of the three described methods.

7.2 Further Research

The aim of this research was to develop novel algorithms for application to the BSS problem. As stated in chapter 1 within this thesis only the instantaneous BSS problem was considered. The extension of the Matrix Momentum algorithm to the convolutive case would represent the next natural algorithm extension. This could be undertaken using a variety of approaches, but some initial ideas are given as follows.

- Transformation to the frequency domain [21, 22, 23], this transforms the convolutive model to a series of instantaneous ICA problems in each frequency bin. The Matrix Momentum could be applied to perform the ICA at each frequency, then applying the inverse Fourier transform. This approach has the problem that the permutation and amplitude cannot be found using standard ICA methods. To alleviate this problem the above technique could be combined with methods for resolving these ambiguities inherent to the ICA problem [284, 285].

- Temporal based methods based on oversampling and row stacking [26, 27, 28, 29] have also been utilized within the Blind Equalization field for Single

Input Multiple Output (SIMO) Digital Communications. This concept can be extended to the Multiple Input Multiple Output (MIMO) scenario using instantaneous ICA as in [30].

The SPSA algorithm introduced in chapter 5 and again utilized in chapter 6 was shown to provide fast convergence for both the BSS and Image Registration problems. It is expected that for both tasks the following would provide interesting research avenues.

- To improve the convergence properties of the SPSA algorithm Spall developed the Adaptive SPSA algorithm [177, 222, 223]. This algorithm represents a Newton type method, with appropriate parameter tuning could provide large improvements in algorithm convergence time at the expense of additional computational complexity.

- Investigation of novel non gradient based optimization algorithms such as the Complex Step Derivative [225, 226] or methods based on Algorithmic Differentiation [286] may offer new frameworks upon which to develop new BSS and Image Registration based methods.

It has been mentioned repeatedly throughout this thesis that it was discovered during the course of the work on BSS that there was significant overlap between the cost functions and optimization procedures utilized within the BSS and Image Registration fields. A final suggestion for future merging of the work developed within this thesis.

- The application of the Matrix Momentum algorithm for optimization of the Shannon [49], Renyi [231] and Tsallis [232] based mutual information to solve the Image Registration problem.

- The above algorithms could then be utilized in other applications of Image Registration than Medical Imaging, for example Super Resolution Image Reconstruction [234, 233] or Motion Estimation [287, 288, 289].

Appendix A

Higher Order Statistics

A.1 Higher Order Statistics

Due to the importance of higher order statistics in the theoretical development of the ICA problem this thesis would be incomplete without an introduction to the field. Second order processes have historically been the main topic of study in the statistical signal processing community based predominately on the assumption that the data has a Gaussian distribution. Yet these techniques are inappropriate when the data is non-Gaussian. As no assumptions as to the density function of the source signals are made, a method of characterizing the distributions, that may be computed from the data samples to give information about the nature of the source signals is required. The statistics used to further describe a non-Gaussian distribution are the moments and cumulants, these will be described further within this Appendix. The term Higher Order Statistics (HOS) refers to moments and cumulants of order greater than two. The utilization of HOS either implicitly or explicitly forms the backbone of the majority of algorithms for the BSS problem. These will be described in the following section.

A.2 Moments

The moments of a random variable x with probability density function $p_x(x)$ are given as follows:

$$m_n = E[x^n] = \int_{-\infty}^{\infty} x^n p_x(x) dx \tag{A.1}$$

where n is a non-negative integer number. Another useful set of moments of a random variable x are the central moments, which are the moments about the mean value μ, this is given as follows:

$$\mu_n = E[(x-\mu)^n] = \int_{-\infty}^{\infty} (x-\mu)^n p_x(x) dx \tag{A.2}$$

The first central moment ($n = 1$) is zero and produces no useful information.

$$\mu_1 = E[(x-\mu)] = \int_{-\infty}^{\infty} (x-\mu) p_x(x) dx = 0 \tag{A.3}$$

Yet the standard first moment gives the mean value μ of the distribution

$$\mu = E[x] = \int_{-\infty}^{\infty} x p_x(x) dx \tag{A.4}$$

The second central moment ($n = 2$) represents the variance σ^2 of the distribution, or the average deviation from the mean value of the distribution.

$$\sigma^2 = E[(x-\mu)^2] = \int_{-\infty}^{\infty} (x-\mu)^2 p_x(x) dx \tag{A.5}$$

The third central moment, known as the skewness γ provides a measure of the asymmetrical nature of a pdf.

$$\gamma = E[(x-\mu)^3] \tag{A.6}$$

Thus for symmetrical distributions this is zero, and as the majority of natural signals have symmetrical pdf's this measure is less frequently used in solving BSS problems [41, 118, 88], yet a recent application of third order statistics applied

to biomedical data is given in [290, 291]. A much more widely used statistic in solving BSS and ICA problems is the forth order moments and a related statistic known as the kurtosis. As a number of the pdf's of signals encountered in BSS and ICA problems have zero mean or can be normalized to have zero mean this will be assumed from this point. The kurtosis of a zero mean random variable has its origins in cumulants which will be described later in this appendix, but due to the relation to the fourth order moment, they will be described initially here. The kurtosis of a random variable is given as follows:

$$\kappa_4(x) = E[x^4] - 3(E[x^2])^2 \tag{A.7}$$

In the case where the data has been normalized to unit variance, as is common with a number of BSS and ICA algorithms as a result of the scaling ambiguity, then the kurtosis is given as follows:

$$\kappa_4(x) = E[x^4] - 3 \tag{A.8}$$

The above equation may be seen as a standardized version of the standard fourth order moment. Another definition of kurtosis often utilized is the normalized kurtosis.

$$\kappa_4(x) = \frac{E[x^4]}{(E[x^2])^2} - 3 \tag{A.9}$$

It can be seen clearly that if the variable has been normalized to unit variance then the normalized kurtosis is again simply a normalized version of the fourth order moment. The kurtosis of a random variable is specifically important in the fields of BSS and ICA as the kurtosis gives a measure of the non-Gaussianity of a random variable, as for a Gaussian random variable the kurtosis is zero [41, 118, 88]. In the statistical literature a distribution with zero kurtosis is called mesokurtic. Distributions that rise to a peak faster than a Gaussian distribution, and have longer tails are known as leptokurtic or super-Gaussian distributions.

Super-Gaussian distributions can be characterized by having a kurtosis value greater than zero. Distributions that have shorter tails and rise slower than a Gaussian distribution are known as platykurtic or sub-Gaussian distributions. These can be characterized by having a kurtosis value less than zero. An example of a Gaussian, super-Gaussian and sub-Gaussian distribution is shown graphically in Figure A.1.

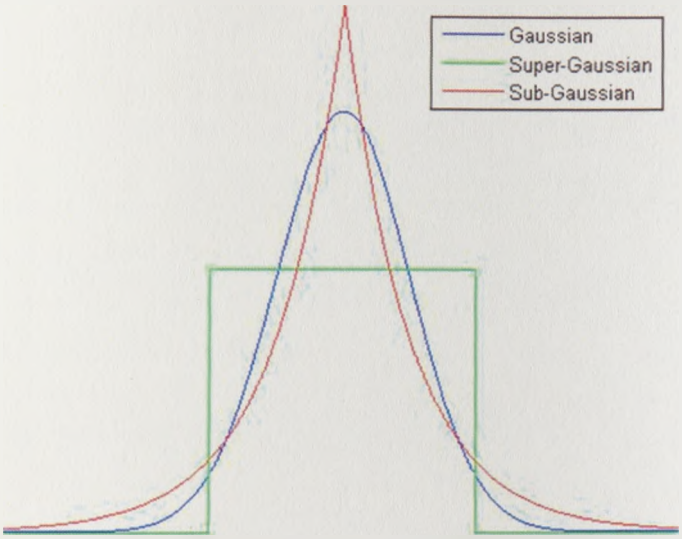


Figure A.1: Density models for the super-Gaussian, sub-Gaussian and Gaussian distributions

In the above section the scalar moments up to the fourth order have been introduced. In the following subsection the generation of moments is introduced along with their link to probability density functions via the characteristic function.

A.2.1 Characteristic Functions

For a zero mean random variable x with probability density function $p_x(x)$ the characteristic function or moment generating function is given by the following

integral [44, 292, 293].

$$\Phi(\omega) = \int_{-\infty}^{\infty} p_x(x) e^{j\omega x} dx \tag{A.10}$$

Each probability density function is uniquely defined by its characteristic function. The characteristic function represents the Fourier transform of the pdf of the random variable x. This may be written as the expectation over the density $p_x(x)$.

$$\Phi(\omega) = E[e^{j\omega x}] \tag{A.11}$$

Taking the Taylor series expansion of the above equation

$$\Phi(\omega) = \int_{-\infty}^{\infty} p_x(x)\left(1 + j\omega x + \frac{(j\omega x)^2}{2!} + \cdots\right) dx \tag{A.12}$$

Taking the n^{th} derivative of the above equation and evaluating at $j\omega = 0$ results in the following expression.

$$\begin{aligned} m_n &= \frac{d^n \Phi(j\omega)}{d\omega^n} \\ &= \int_{-\infty}^{\infty} p_x(x) \frac{d^n}{d\omega^n}\left(1 + j\omega x + \frac{(j\omega x)^2}{2!} + \cdots\right) dx \\ &= \frac{d^n}{d\omega^n}\bigg|_{\omega=0} \sum_{n=0}^{\infty} \left(\frac{x^n (j\omega)^n}{n!}\right) p(x) \qquad \text{(A.13)} \\ &= \frac{(j\omega)^n}{n!} E[x^n] \qquad \text{(A.14)} \end{aligned}$$

The coefficient terms of the above equation represents the moments of the random variable x hence the term moment generating function. If all the moments of a random variable are finite and the series converges absolutely near $\omega = 0$ then the moments uniquely define the pdf of the random variable. Extending this result to the case of multiple random variables with joint probability density function $p(x_1, x_2, \ldots, x_n)$ the following characteristic function is developed

$$\Phi(\omega) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} p_{x_1,\ldots x_n}(x_1, \ldots x_n) e^{j(\omega_1 x_1 + \cdots + \omega_n x_n)} dx_1 \ldots dx_n \tag{A.15}$$

The joint moments are calculated in the same manner as the scalar moments of a random variable. Taking the Taylor series expansion of the above equation and differentiating as done previously. the following equation for the joint moments of order $r = i_1 + \cdots + i_n$ is obtained.

$$m_{i_1,\ldots,i_n} = (-j)^r \frac{\partial^r \Phi(\omega_1,\ldots,\omega_n)}{\partial \omega_1^{i_1},\ldots,\partial \omega_n^{i_n}}\bigg|_{\omega_1=\cdots=\omega_n=0} \tag{A.16}$$

This can be written simply as follows:

$$m_{i_1,\ldots,i_n} = E[x_1^{i_1}\ldots x_n^{i_n}] \tag{A.17}$$

In the next section the cumulants of random variables and vectors are introduced.

A.3 Cumulants

Related to moments, cumulants have the interesting property that the n^{th} cumulant of a sum of independent variables is simply the sum of the n^{th} cumulants of the summands. Calculation of the cumulants is detailed in the following subsection.

A.3.1 Cumulant Generating Function

Just as the characteristic function for a random variable x can be used to generate the moments, the natural logarithm of the characteristic function can be used to generate the cumulants of the random variable [44, 292, 293]. this is known as the second characteristic function or the cumulant generating function

$$\Psi(\omega) = ln(\Phi(\Omega)) = ln(E[e^{j\omega x}]) \tag{A.18}$$

The coefficients of κ_n of the Taylor series expansion of the cumulant generating function are called the cumulants of the distribution of the random variable x

Repeating the analysis performed for the moments, the cumulants are given as follows:

$$\kappa_n = (-j)^n \frac{d^n ln(\Phi(\Omega))}{d\omega^n}\bigg|_{\omega=0} \tag{A.19}$$

Using the above equation the first four cumulants of a random variable x with non zero mean ($\mu \neq 0$) are given as follows:

$$\kappa_1 = m_1 \tag{A.20}$$

$$\kappa_2 = m_2 - m_1^2 \tag{A.21}$$

$$\kappa_3 = m_3 - 3m_2m_1 + 2m_1^3 \tag{A.22}$$

$$\kappa_4 = m_4 - 3m_2^2 - 4m_3m_1 + 12m_2m_1^2 - 6m_1^4 \tag{A.23}$$

The first and second cumulants representing the mean and the variance of the random variable x respectively. Repeating this analysis for the multivariate case, results in the following equation.

$$\kappa_{i_1,\ldots,i_n} = (-j)^r \frac{\partial^r \Psi(\omega_1,\ldots,\omega_n)}{\partial\omega_1^{i_1},\ldots,\partial\omega_n^{i_n}}\bigg|_{\omega_1=\cdots=\omega_n=0} \tag{A.24}$$

The following simplified notation is used extensively when dealing with cross cumulants, to calculate the cross cumulant for a given order $r = i_1 + \cdots + i_n$

$$\kappa_{i_1,\ldots,i_n} = cum_{\kappa_{i_1},\ldots,i_n}(\mathbf{x}_1,\ldots,\mathbf{x}_n) \tag{A.25}$$

Utilizing this equation the second, third and fourth cross cumulants for a zero mean vector $\mathbf{x}$ are given as follows:

$$cum_2(\mathbf{x}_1\mathbf{x}_2) = E[\mathbf{x}_1\mathbf{x}_2] \tag{A.26}$$

$$cum_3(\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3) = E[\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3] \tag{A.27}$$

$$cum_4(\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3\mathbf{x}_4) = E[\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3\mathbf{x}_4] - E[\mathbf{x}_1\mathbf{x}_2]E[\mathbf{x}_3\mathbf{x}_4] \tag{A.28}$$

$$- E[\mathbf{x}_1\mathbf{x}_3]E[\mathbf{x}_2\mathbf{x}_4] - E[\mathbf{x}_1\mathbf{x}_4]E[\mathbf{x}_2\mathbf{x}_3] \tag{A.29}$$

It can be seen clearly that Equation A.29 reduces to the kurtosis given in Equation A.7 for the case of a single variable, where $x_1 = x_2 = x_3 = x_4$.

Now that moments and cumulants have been introduced a probability density estimation technique, the Edgeworth expansion that is utilized within the BSS literature will be described.

A.4 Probability Density Estimation

In the Blind Source Separation and Equalization algorithms it is commonly necessary to have to estimate the probability density functions of the underlying source signals. A number of estimation methods have been utilized in the literature, for example Parzen Windowing [294], Gaussian Mixture Models [295] and methods based upon using the Pearson system [296, 291, 297]. Methods based upon Chebyshev-Hermite polynomials have been extensively used throughout the literature and are generally based upon either the Gram-Charlier [298] or Edgeworth [48] expansions. These expansions lead to very similar approximations differing based upon the ordering of the terms. The Edgeworth expansion is introduced in the following section.

A.5 Edgeworth Expansion

It is well known in Signal Processing and Mathematics that functions can be expressed as a series of terms such as trigonometric functions e.g. the Fourier and Cosine series, or as powers of the variable using the Taylor series expansion. For the case of probability density functions these expansions aren't suited ideally. Instead the Gram-Charlier and Edgeworth expansions approximate the probability density function using the moments and cumulants respectively of a

random variable [299, 300, 301]. The Edgeworth expansion utilizes heavily the Chebyshev-Hermite polynomials in the pdf expansion [302]. These are described in the following subsection.

A.5.1 Chebyshev-Hermite polynomials

The Chebyshev-Hermite polynomials $h_n(y)$ are defined as the successive derivatives of a Gaussian random variable $\phi(y)$ with zero mean and unit variance, this is given as follows:

$$(-1)^n \frac{d^n}{dy^n}\phi(y) = h_n(y)\phi(y) \tag{A.30}$$

With $\phi(y)$ given as follows .

$$\phi(y) = e^{-y^2/2} \tag{A.31}$$

A recurrence relation for calculation of the Hermite polynomials is given as follows:

$$h_{n+1}(y) = yh_n(y) - nh_{n-1}(y) \tag{A.32}$$

Where the initial coefficient in the relation $h_0 = 1$. Utilizing the above the truncated Edgeworth expansion written in terms of the n^{th} order cumulants κ_n and the Hermite polynomials h_n is given as follows:

$$\begin{aligned} p_y(y) &= \phi(y)\Big[1 + \frac{1}{3!}\kappa_3 h_3(y) + \frac{1}{4!}\kappa_4 h_4(y) + \frac{10}{6!}\kappa_4^2 h_6(y) \\ &+ \frac{1}{5!}\kappa_5 h_5(y) + \frac{35}{7!}\kappa_3\kappa_4 h_7(y) + \frac{280}{9!}\kappa_3^3 h_9(y) \\ &+ \frac{1}{6!}\kappa_6 h_6(y) + \frac{56}{8!}\kappa_3\kappa_5 h_8(y) + \frac{35}{8!}\kappa_4^2 h_8(y) \\ &+ \frac{2100}{10!}\kappa_3^2\kappa_4 h_{10}(y) + \frac{15400}{12!}\kappa_3^4 h_{12}(y)\Big] \end{aligned} \tag{A.33}$$

where $\phi(y)$ represents the Gaussian density function given in Equation A.31

Appendix B

Whitening Transformations

It was shown in chapter 2 that a whitening transformation performed by a Principal Component Analysis (PCA) stage, before performing ICA, solves the problem up to an orthogonal rotation parameter. This parameter can be resolved by incorporating either the HOS of the data vector within an ICA algorithm, or by making further assumptions upon the source signals and solving using a second order BSS algorithm. Within this Appendix the Singular Value Decomposition utilized extensively for performing the initial Whitening transformation is described.

B.1 Singular Value Decomposition

The Singular Value Decomposition (SVD) is known as one of the most powerful tools from Linear Algebra. It is used to decompose a matrix into several component matrices, this is often described as a matrix factorization. The popularity of the SVD arises due to its ability to deal robustly with over and underdetermined least squares problems [303], and ill-conditioned matrices [304, 305]. In the Blind Source Separation or Independent Component Analysis case the singular value

decomposition is used as a method of diagonalizing the covariance matrix of the observation vector $\mathbf{y}(k)$. This process is often referred to as a spatial whitening. The SVD of a matrix $\mathbf{A}$ is given in the following subsection.

B.1.1 SVD Theory

Every matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ can be factored as follows:

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \tag{B.1}$$

where $\mathbf{U} \in \mathbb{R}^{m \times m}$ is an orthogonal matrix, $\boldsymbol{\Sigma} \in \mathbb{R}^{m \times n}$ is a diagonal matrix[1], and $\mathbf{V}^T \in \mathbb{R}^{n \times n}$ is an orthogonal matrix. The columns of the matrix $\mathbf{U}$ are the eigenvectors of the matrix $\mathbf{A}\mathbf{A}^T$ and are known as the left singular eigenvectors, likewise the columns of the matrix $\mathbf{V}$ are the eigenvectors of the matrix $\mathbf{A}^T\mathbf{A}$ and are similarly known as the right singular eigenvectors. The diagonal elements σ_i of the matrix $\boldsymbol{\Sigma}$ are known as the singular values of $\mathbf{A}$, these are the square root of the eigenvalues of both $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$. The singular values from the output of the SVD will be ordered as follows:

$$\sigma_1 \geq \sigma_2 \geq \sigma_2 \geq \cdots \geq \sigma_p \geq 0. \tag{B.2}$$

where $p = min(m, n)$. The number p of non-zero singular values gives the numerical rank of the matrix $\mathbf{A}$ [56].

B.1.2 SVD as a Whitening transformation

In chapter 2 it was shown that the Whitening transformation is used to develop a matrix $\mathbf{B}$ that decorrelates the observation vector $\mathbf{x}(k)$. This transformation shown in Equation 2.11. should result in the diagonalization of the covariance matrix, and the variances of the sources on the diagonal should be unity. The

[1] If the matrix Σ is not a square matrix has non zero elements for Σ_{ii} and zeros elsewhere

covariance matrix of the observation vector $\mathbf{x}(k)$ before the Whitening transformation $\mathbf{B}$ is given as follows.

$$\begin{aligned}\mathbf{R}_{\mathbf{xx}} &= E[\mathbf{x}(k)\mathbf{x}(k)^T] \\ &= E[(\mathbf{A}\mathbf{s}(k))(\mathbf{A}\mathbf{s}(k))^T] \\ &= E[\mathbf{A}\mathbf{s}(k)\mathbf{s}(k)^T\mathbf{A}^T] \\ &= E[\mathbf{A}\mathbf{R}_{\mathbf{ss}}\mathbf{A}^T] \end{aligned} \tag{B.3}$$

Utilizing the fundamental assumption of Statistical Independence of the input sources $\mathbf{s}(k)$ described in subsection 2.1.1, the covariance matrix of the input sources $\mathbf{R}_{\mathbf{ss}} = \mathbf{I}$. Placing this into Equation B.3 we obtain the following simplified equation for the covariance matrix.

$$\mathbf{R}_{\mathbf{xx}} = E[\mathbf{A}\mathbf{A}^T] \tag{B.4}$$

The above covariance has the property that its symmetric. that is $\mathbf{R}_{\mathbf{xx}} = \mathbf{R}_{\mathbf{xx}}^T$. Real symmetric matrices have a number of unique properties [37, 107].

1. They have a unique spectral factorization.
2. Their eigenvalues are all real.
3. The eigenvectors corresponding to each unique eigenvalue are orthogonal.
4. As a consequence of the symmetry the above matrices $\mathbf{U}$ and $\mathbf{V}$ in the SVD shown in Equation B.1 are identical.

Taking the SVD of Equation B.4 and utilizing the above properties of real symmetric matrices, then in this case the SVD is equivalent to the Eigenvalue Decomposition (EVD) as a Whitening transformation.

$$\begin{aligned}\mathbf{R}_{\mathbf{xx}} &= \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T \\ E[\mathbf{A}\mathbf{A}^T] &= \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T \\ E[\mathbf{A}\mathbf{A}^T] &= \mathbf{V}\mathbf{\Sigma}^{1/2}\mathbf{\Sigma}^{1/2}\mathbf{V}^T \end{aligned} \tag{B.5}$$

From observation of Equation B.5, and the knowledge that the matrix $\mathbf{V}$ is orthogonal, such that $\mathbf{V}\mathbf{V}^T = \mathbf{V}^T\mathbf{V} = \mathbf{I}$ it can be seen that the transformation $\mathbf{B}$ required to diagonalize the covariance matrix $\mathbf{R}_{xx}$ is given by the following equation.

$$\mathbf{B} = \Sigma^{-1/2}\mathbf{V}^T \tag{B.6}$$

This is demonstrated by applying this transformation to the observation vector $\mathbf{x}(k)$, then taking the covariance matrix of the Whitened output $\mathbf{z}(k)$ results in the following covariance matrix originally shown in Equation 2.12. Utilizing the SVD transformation obtained above in Equation B.5, the Whitened output is obtained.

$$\begin{aligned}
\mathbf{R}_{zz} &= E[\mathbf{z}(k)\mathbf{z}(k)^T] \\
&= E[\mathbf{B}\mathbf{x}(k)\mathbf{x}(k)^T\mathbf{B}^T] \\
&= E[\mathbf{B}\mathbf{R}_{xx}\mathbf{B}^T] \\
&= E[\mathbf{B}\mathbf{A}\mathbf{A}^T\mathbf{B}^T] \\
&= E[\mathbf{B}\mathbf{V}\Sigma^{1/2}\Sigma^{1/2}\mathbf{V}^T\mathbf{B}^T] \\
&= E[\Sigma^{-1/2}\mathbf{V}^T\mathbf{V}\Sigma^{1/2}\Sigma^{1/2}\mathbf{V}^T\mathbf{V}\Sigma^{1/2}] \\
&= \mathbf{I}
\end{aligned} \tag{B.7}$$

Bibliography

[1] C. E. Cherry. Some experiments in the recognition of speech, with one and two ears. *Journal of the Acoustical Society of America*, 25:975–979, 1953.

[2] C. E. Cherry and W. L. Taylor. Some further experiments upon the recognition of speech, with one and two ears. *The Journal of the Acoustical Society of America*, 26:554–559, 1954.

[3] C. E. Cherry. *On human communication: A review, survey, and a criticism* MIT Press, Cambridge, MA, 1957.

[4] S. Haykin and Z. Chen. The cocktail party problem. *Neural Computation*, 17(9):1875–902, 2005.

[5] S. Makeig and A. Bell. Independent component analysis of electroencephalographic data. *Advances in Neural Information Processing Systems*, 8:145–151, 1996.

[6] R. N. Vigrio. Extraction of ocular artifacts from eeg using independent component analysis. *Electroenceph. clin. Neurophysiol.*, 103:395–404, 1997.

[7] A. Delorme and S. Makeig. Eeglab: an open source toolbox for analysis of single-trial eeg dynamics. *Journal Neurosci Methods*, 134:9–21, 2004.

[8] T-P. Jung, S. Makeig, A. Bell, and TJ. Sejnowski. *P. Poon and J. Brugge editors Auditory Processing and Neural Modeling.* Plenum Press, New York 1998.

[9] R. Vigario, J. Sarela, V. Jousmaki, M. Hamalainen, and E. Oja. Independent component approach to the analysis of eeg and meg recordings. *IEEE Trans. Biomedical Engineeering*, 47(5):589–593, 2000.

[10] A. C. Tang, B.A. Pearlmutter, N. A. Malaszenko, D. B. Phung, and B. C. Reeb. Independent components of magnetoencephalography: Localization *Neural Computation*, 14(8):1827–1858, 2002.

[11] A. C. Tang, D. Phung, B. A. Pearlmutter, and R. Christner. Localization of independent components from magnetoencephalography. In *Proc. IEEE Workshop on ICA*, pages 387–392, Helsinki, Finland, 2000.

[12] MJ. McKeown, S. Makeig, GG. Brown, T-P. Jung, SS. Kindermann, and T. Sejnowski. Analysis of fmri data by blind separation into independent spatial components. *Human Brain Mapping*, 6:160–188, 1998.

[13] V. Calhoun, T. Adali, L. K. Hansen, J. Larsen, and J. Pekar. Ica of functional mri data: an overview. In *Proc. IEEE Workshop on ICA*, pages 281–288, Nara, Japan, 2003.

[14] C. Liu and H. Wechsler. Comparative assessment of independent component analysis (ica) for face recognition. In *Proc. of the Second International Conference on Audio- and Video-based Biometric Person Authentication*, pages 211–216, 1999.

[15] M.S. Bartlett, J.R. Movellan, and T.J. Sejnowski. Face recognition by independent component analysis. *IEEE Trans. on Neural Networks*, 13(6):1450–1464, 2002.

[16] M.S. Bartlett. *Face Image Analysis by Unsupervised Learning*. Kluwer Academic Publishers, Princeton, NJ, 2001.

[17] M. S. Bartlett. *Face image analysis by unsupervised learning and redundancy reduction*. PhD thesis, University of California, 1998.

[18] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

[19] P. Hoyer and A. Hyvarinen. Independent component analysis applied to feature extraction from colour and stereo images. *Network: Computation in Neural Systems*, 11:191–210, August 2000.

[20] A. Hyvarinen, J. Hurri, P.O. Hoyer, and E. Oja. Image feature extraction by sparse coding and independent component analysis. In *Proceedings of the 14th International Conference on Pattern Recognition*, pages Vol II. 1268–1273, 1998.

[21] M. E. Davies. *Audio Source Separation, In Mathematics in Signal Processing V*. Oxford University Press, 2002.

[22] N. Mitianoudis and M. E. Davies. Audio source separation of convolved mixtures. *IEEE Trans. Speech and Audio Processing*. 11(5):489–497. 2003.

[23] P. Smaragdis. Blind separation of convolved mixtures in the frequency domain. *Neurocomputing*, 22:21–34, 1998.

[24] L. K. Hansen and M. Dyrholm. A prediction matrix approach to convolutive ica. In *Proceedings of IEEE Workshop on Neural Networks for Signal Processing*, pages 249–258, 2003.

[25] M. Dyrholm. *Independent Component Analysis in a convoluted world*. PhD thesis, Technical University of Denmark, 2005.

[26] G. Morison and T. Durrani. Blind equalization using matrix momentum and natural gradient adaptation. In *IEEE 13th Workshop Neural Networks for Signal Processing, NNSP'03*, pages 439–448, 2003.

[27] G. Morison and T. Durrani. Blind MIMO channel equalisation using matrix momentum natrual gradient adaptation. In *IEE Conference on DSP enabled radio, Livingston , Scotland*, September 2003.

[28] H. H. Yang. On-line blind equalization via on-line blind separation. *Signal Processing*, 68:271–281, 1998.

[29] S. Choi and A. Cichocki. Blind equalization via approximate maximum likelihood source separation. *Electronic Letters*, 37:61–62, 2001.

[30] X. Zhuang and A. Swindlehurst. Blind equalization via blind source separation techniques. *Signal Processing - In Review*, 1999.

[31] B. H. Kevles. *Naked to the Bone: Medical Imaging in the Twentieth Century*. Perseus Books, U.S, Cambridge, MA, 1998.

[32] T. Adair, M. Reivich, P. Karp, and A. Stein. Computer assisted analysis of tomographic images of the brain. *Journal of Computer Assisted Tomography*, 5(6):929–932, 1981.

[33] B.A. Ardekani, A.H. Bachman, S.C Strother, and Y Fujibayashi Y Yonekura. Impact of inter-subject image registration on group analysis of

fmri data. *Quantitation in biomedical imaging with PET and MRI Elsevier International Congress Series*, 1265:49–59, 2004.

[34] S. Periaswamy. *General-Purpose Medical Image Registration*. PhD thesis, Department of Computer Science, Dartmouth College, Hanover, NH, 2003.

[35] T.W. Lee, M.S. Lewicki, M. Girolami, and T.J. Sejnowski. Blind source separation of more sources than mixtures using overcomplete representations. *IEEE Signal Processing Letters*, 6(4), 1999.

[36] L. Zhang, A. Cichocki, and S. Amari. Natural gradient algorithm for blind separation of overdetermined mixture with additive noise. *IEEE Signal Processing Letters*, 6(11):293–295, 1999.

[37] W. Stirling T. Moon. *Mathematical Methods and Algorithms for Signal Processing*. Prentice Hall, 1999.

[38] M. Partridge and R.A. Calvo. Fast dimensionality reduction and simple pca. *Intelligent Data Analysis*, 2(3), 1998.

[39] E. Oja. A simplified neuron model as a principal component analyzer. *J. of Mathematical Biology*, 15:267–273, 1982.

[40] E. Oja. Neural networks, principal components, and subspaces. *Int. J. on Neural Systems*, 1:61–68, 1989.

[41] M. Girolami. *Self-Organising Neural Networks - Independent Component Analysis and Blind Source Separation*. Springer-Verlag, 1999.

[42] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. In D. E. Rumelhart, J. L. McClelland, et al., editors, *Parallel Distributed Processing: Volume 1: Foundations*, pages 318–362. MIT Press, Cambridge, 1987.

[43] S.Haykin. *Neural Networks: A Comprehensive Foundation (2nd Edition)*. Prentice Hall, 1998.

[44] A. Papoulis. *Probability, Random Variables and Stochastic Processes*. MIT Press, Cambridge, MA, 1991.

[45] V. Zarzoso. *Closed-Form Higher-Order Estimators for Blind Separation of Independent Source Signals in Instantaneous Linear Mixtures*. PhD Thesis,The University of Liverpool, Liverpool, UK, 1999.

[46] A. K. Nandi V. Zarzoso. *"Blind Source Separation"*, in: *A. K. Nandi (Ed.), Blind Estimation Using Higher-Order Statistics*. Kluwer Academic Publishers, 1999.

[47] C. Jutten and J. Hérault. Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal Processing*, 24:1–10, 1991.

[48] P. Comon. Independent component analysis—a new concept? *Signal Processing*, 36:287–314, 1994.

[49] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423 and 623-656, July and October, 1948.

[50] W. Weaver and C. E. Shannon. *The Mathematical Theory of Communication*. University of Illinois Press, Urbana, Illinois, 1949, republished in paperback in 1963.

[51] R. Leibler S. Kullback. On information and sufficiency. *Ann. Math. Stat.*, 22:79–86, 1951.

[52] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley, 1991.

[53] L. Tong, V. Soo, R. Liu, and Y. Huang. AMUSE: a new blind identification algorithm. In *IEEE International Symposium on Circuits and Systems*, New Orleans, USA, 1990.

[54] Ying-Chang Liang, Yan-Da Li, and Xian-Da Zhang. EAMUSE: an extended algorithm for multiple sources extraction. In *IEEE International Symposium on Circuits and Systems*, Seattle, WA, USA, 1995

[55] A. Belouchrani, K. Abed Meraim, J.-F. Cardoso, and E. Moulines. A blind source separation technique based on second order statistics. *IEEE Trans on Signal Processing*, 45(2):434–444, 1997.

[56] G. Golub and C. van Loan. *Matrix Computations*. The Johns Hopkins University Press, 3rd edition, 1996.

[57] Edgeworth. F. Y. On the representation of statistical frequency by a series. *J. R. Star. Soc.*, 70:102–106, 1907.

[58] J.Proakis. *Digital Communications (4th Edition)*. McGraw Hill, 2001.

[59] M. Ghogho, A. Swami, and T. Durrani. Approximate maximum likelihood blind source separation with arbitrary source pdfs. In *IEEE workshop on Statistical signal and Array Processing*, Pennsylvania, USA, 2000.

[60] Tobias Blaschke and Laurenz Wiskott. An improved cumulant based method for independent component analysis. In José R. Dorronsoro, editor, *Proc. Intl. Conf. on Artificial Neural Networks - ICANN'02*, Lecture Notes in Computer Science, pages 1087–1093. Springer, 2002

[61] Tobias Blaschke and Laurenz Wiskott. CuBICA: Independent component analysis by simultaneous third- and fourth-order cumulant diagonaliza-

tion. Computer Science Preprint Server (CSPS): Computational Intelligence/0304002, April 2003.

[62] T. Blaschke and L. Wiskott. CuBICA: Independent Component Analysis by Simultaneous Third- and Fourth-Order Cumulant Diagonalization. *IEEE Transactions on Signal Processing*, 52(5):1250–1256, May 2004.

[63] J.-F. Cardoso. Source separation using higher order moments. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'89)*, pages 2109–2112, Glasgow, UK, 1989.

[64] L. Tong, R.-W. Liu, V.C. Soon, and Y.-F. Huang. Indeterminacy and identifiability of blind identification. *IEEE Trans. on Circuits and Systems*, 38:499–509, 1991.

[65] J.-F Cardoso. *Eigen-structure of the fourth order cumulant tensor with applications to the blind soud separation problem.* Proc. IEEE International Conference on Acoustics, Speech and Signal
Processing, Alburquerque, New Mexico, pages 2655-2658. USA. 1990.

[66] J.-F. Cardoso. Super-symmetric decomposition of the fourth-order cumulant tensor. blind identification of more sources than sensors. In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'97)*, pages 3109–3112, 1991.

[67] A Souloumiac J.-F Cardoso. *Blind Beamforming for non Gaussian signals.* IEEE Proceedings-F, vol. 140, no 6, pp. 362-370, USA. 1993

[68] J.F. Cardoso and A. Souloumiac. An efficient technique for the blind separation of complex sources. In *IEEE Signal Processing Workshop on Higher-Order Statistics*, pages 275–279. South Lake Tahoe. CA. USA. 1993

[69] J.-F. Cardoso and A. Souloumiac. Blind beamforming for non Gaussian signals. *IEE Proceedings-F*, 140(6):362–370, 1993.

[70] J.-F. Cardoso and A. Souloumiac. Jacobi angles for simultaneous diagonalization. *SIAM J. Mat. Anal. Appl.*, 17(1):161–164, January 1996.

[71] J.-F. Cardoso. High-order contrasts for independent component analysis. *Neural Computation*, 11(1):157–192, 1999.

[72] A. Batra and J.R. Barry. Blind unitary source separation using a multidimensionalphase-locked loop. In *IEEE Signal Processing Workshop on Signal Processing Advances in Wireless Communications*, pages 61–64, 1997.

[73] J. Rinas and K.D. Kammeyer. Comparison of blind source separation methods based on iterative algorithms. In *5th International ITG Conference on Source and Channel Coding (SCC04)*, pages 61–64, 2004.

[74] L. De Lathauwer. *Signal Processing by Multilinear Algebra*. PhD thesis, Faculty of Engineering, K. U. Leuven, Leuven, Belgium, 1997.

[75] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492, 1997.

[76] A. Hyvärinen. *Independent Component Analysis: A Neural Network Approach*. PhD thesis, Helsinki University of Technology, Department of Computer Science and Engineering, October 1997.

[77] A. Hyvärinen. Fast and robust fixed point algorithms for independent component analysis. *IEEE Trans. on Neural Networks*, 10(3):626–634, 1999.

[78] A. Hyvärinen. A family of fixed-point algorithms for independent component analysis. *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pages 3917–3920, 1997.

[79] A. Hyvärinen. Independent component analysis by minimization of mutual information. *Technical Report A46, Helsinki University of Technology, Laboratory of Computer and Information Science*, 1997.

[80] A. Hyvärinen. One-unit contrast functions for independent component analysis: A statistical analysis. *In Neural Networks for Signal Processing VII (Proc. IEEE Workshop on Neural Networks for Signal Processing*, pages 388–397, 1997.

[81] A. Hyvärinen. Independent component analysis in the presence of gaussian noise by maximizing joint likelihood. *Neurocomputing*, pages 22:49–67, 1998.

[82] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. on Neural Networks*, 10(3) 626–634, 1999.

[83] E. Bingham and A. Hyvärinen. Fast and robust deflationary separation of complex-valued signals. *Proc. European Conf. on Signal Processing (EUSIPCO)*, I:23–26, 2000.

[84] E. Bingham and A. Hyvärinen. ICA of complex-valued signals: a fast and robust deflationary algorithm. In *Proc. Int. Joint Conf. on Neural Networks (IJCNN2000)*, pages 357–362, Como, Italy, 2000

[85] E. Bingham and A. Hyvärinen. A fast fixed-point algorithm for independent component analysis of complex-valued signals. *Int. J. of Neural Systems*, 10(1):1–8, 2000.

[86] V. Zarzoso and A. Nandi. Blind mimo equalization with optimum delay using independent component analysis. *International Journal of Adaptive Control and Signal Processing (Special Issue on Blind Signal Separation)*. 18(3):245–263, 2004.

[87] L. Molgedey and H. G. Schuster. Separation of a mixture of independent signals using time delayed correlations. *Physical Review Letters*. 72:3634–3636, 1994.

[88] A. Cichocki and S. Amari. *Adaptive Blind Signal and Image Processing*. John Wiley and Sons, 2002.

[89] J. Hérault and B. Ans. Circuits neuronaux à synapses modifiables: décodage de messages composites par apprentissage non supervisé. *C.R. de l'Académie des Sciences*, 299(III-13):525–528. 1984.

[90] J. Hérault, C. Jutten, and B. Ans. Détection de grandeurs primitives dans un message composite par une architecture de calcul neuromimétique en apprentissage non supervisé. In *Actes du Xème colloque GRETSI*. pages 1017–1022, Nice, France, 1985.

[91] C. Jutten. *Calcul neuromimétique et traitement du signal, analyse en composantes indépendantes*. PhD thesis. INPG, Univ. Grenoble. France. 1987. (in French).

[92] Ammar B., A. Gharbi, and Fathi Salam. Implementation and test results of a chip for the separation of mixed signals. *IEEE transactions on Circuits and Systems*. 42(11):748–751. November 1995.

[93] Ammar B., A. Gharbi, and Fathi Salam. Implementation and test results of a chip for the separation of mixed signals. In *Proc. Int. Symp on Circuits and Systems, Seattle, WA, USA.*, pages 271–274, 1995.

[94] Ammar B., A. Gharbi, and Fathi Salam. Implementation and experimental results of a chip for the separation of mixed and filtered signals. *Journal of Circuits, Systems and Computers*, 6(2):115–139, April 1996.

[95] A. Celik, M. Stanacevic, and G. Cauwenberghs. Mixed-signal real-time adaptive blind source separation. In *Proc. Int. Symp. on Circuits and Systems, Vancouver Canada*, pages 23–26, 2004.

[96] Cichocki A., Bogner R.E., Moszczynski L., and Pope K. Modified herault-jutten algorithms for blind separation of sources. *Digital Signal Processing*, 7:80–93, 1997.

[97] Nomura T., Eguchi M., Niwamoto H., Kokubo H., and Miyamoto M. An extension of the herault jutten network to signals including delays for blind separation. In *Proceedings IEEE Neural Networks for Signal Processing VI Piscataway, New Jersey*, pages 443–452, Helsinki, Finland, 1996.

[98] F. Berthommier and S. Choi. Several improvements of the herault-jutten model for speech segregation. In *Proceedings of the International Conference on Independent Component Analysis and Signal Separation, Nara, Japan*, pages 1089–1094, Nara, Japan, April 2003.

[99] Yong Kim and Hong Jeong. A fpga architecture of blind source separation and real time implementation. In *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach: First International Work Conference on the Interplay Between Natural and Artificial Computation.*

IWINAC 2005, pages 347–356, Las Palmas, Canary Islands, Spain, June 15-18 2005.

[100] D. Schobben, K. Torkkola, and P. Smaragdis. Evaluation of blind signal separation methods. *Proceedings Int. Workshop Independent Component Analysis and Blind Signal Separation, Aussois, France, January 11-15 1999.*, 1999.

[101] S.-I. Amari, A. Cichocki, and H.H. Yang. A new learning algorithm for blind source separation. In *Advances in Neural Information Processing Systems 8*, pages 757–763. MIT Press, 1996.

[102] A. Westner. Object based audio capture: separating acoustically mixed sounds. *MIT Masters Thesis*, pages 54–55, 1999.

[103] J. Peach Y.Li, D. Powers. Comparison of blind source separation algorithms. *Advances in Neural Networks and Applications*, pages 18–21, 2000.

[104] A.J. Bell and T.J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7.1129-1159, 1995.

[105] A. J. Bell and T. J. Sejnowski. A non-linear information maximization algorithm that performs blind separation. In *Advances in Neural Information Processing Systems 7*, pages 467–474. The MIT Press, Cambridge, MA, 1995.

[106] J.-F. Cardoso. Infomax and maximum likelihood for source separation. *IEEE Letters on Signal Processing*, 4:112–114, 1997.

[107] G. Strang. *Introduction to Linear Algebra, 3rd Edition*. Wellesley-Cambridge Press, 1998.

[108] A. Cauchy. Mémoire sur les racines des équivalences correspondantes à des modules quelconques premiers ou non premiers, et sur les avantages que présente l'emploi de ces racines dans la théorie des nombres. 25:37–54, 1847. Reprinted in *Oeuvres (1)*, Vol. 10, pp. 324–333.

[109] S.Haykin. *Adaptive Filter Theory (4th Edition)*. Prentice Hall, 2001.

[110] B. Widrow. *A Statistical Theory of Adaptation*. Perdman Press, Oxford, 1963.

[111] S. Amari. A theory of adaptive pattern classifiers. *IEEE Transactions on Systems, Man and Cybernetics*, 2:643–657, 1967.

[112] S.-I. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998.

[113] S.-I. Amari. Neural learning in structured parameter spaces—natural Riemannian gradient. In *Advances in Neural Information Processing Systems 9*, pages 127–133. MIT Press, 1997.

[114] S.-I. Amari and A. Cichocki. Adaptive blind signal processing—neural network approaches. *Proceedings of the IEEE*, 86(10):2026–2048, 1998.

[115] F.Engel S.Lie. Theorie der transformationsgruppen, 1899.

[116] S. Sternberg. *Group theory and physics*. Cambridge, 1999.

[117] S.-I. Amari, T.-P. Chen, and A. Cichocki. Stability analysis of adaptive blind source separation. *Neural Networks*, 10(8):1345–1351, 1997.

[118] T.-W. Lee. *Independent Component Analysis - Theory and Applications*. Kluwer, 1998.

[119] T.-W. Lee, M. Girolami, and T. J. Sejnowski. Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources. *Neural Computation*, 11(2):417–441, 1999.

[120] T.-W. Lee, M. Girolami, A.J. Bell, and T.J. Sejnowski. A unifying information-theoretic framework for independent component analysis. *Computers and Mathematics with Applications*, 31(11):1–12, 2000.

[121] M. Girolami and C. Fyfe. Stochastic ICA contrast maximisation using Oja's nonlinear PCA algorithm. *Int. J. Neural Systems*, 8(5 & 6):661–678, 1997.

[122] J.-F. Cardoso and B. Hvam Laheld. Equivariant adaptive source separation. *IEEE Trans. on Signal Processing*, 44(12):3017–3030, 1996.

[123] B. Laheld and J.-F. Cardoso. Adaptive source separation with uniform performance. In *Proc. EUSIPCO*, pages 183–186, Edinburgh, 1994.

[124] A. Cichocki, S.-I. Amari, and J. Cao. Blind separation of delayed and convolved sources with self-adaptive learning rate. In *Proc. Int. Symp. on Nonlinear Theory and Applications (NOLTA'96)*, pages 229–232, Kochi, Japan, 1996.

[125] M. G. Jafari, J. A. Chambers, and D. P. Mandic. A novel adaptive learning rate sequential blind source separation algorithm. *Signal Processing*, 84:801–804, 1994.

[126] J. A. Chambers, M. G. Jafari, and S. McLaughlin. A new variable step-size easi algorithm for sequential blind source separation. *IEE Electronics Letters*, 40:393–394, 2004.

[127] S. Haykin. *Adaptive Filter Theory*. Prentice Hall, 3rd edition, 1996.

[128] J.J. Murillo-Fuentes and F. J. Gonzlez-Serrano. Improving stability in blind source separation with the stochastic median gradient. *Electronic Letters*, 36:1662–1663, 2000.

[129] J.J. Murillo-Fuentes and F. J. Gonzlez-Serrano. Median equivariant adaptive separation via independence: application to communications. *Neurocomputing*, 49:389–409, 2002.

[130] Akuzawa T. and Murata N. Multiplicative nonholonomic/newton-like algorithm. *Chaos, Solitons and Fractals*, 12, Number 4, 3:785–793(9), January 2001.

[131] Akuzawa T. New fast factorization method for multivariate optimzation and its realization as ica algorithm. In *Proceedings of the International Conference on Independent Component Analysis and Signal Separation, ICA2001, San Diego, California*, pages 114–119, 2001.

[132] Akuzawa T. Extended quasi-newton method for the ica. In *Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation, ICA2000, Helsinki. Finland*, pages 521–525, 2000.

[133] Akuzawa T. Nested newton's method for ica and post factor analysis. *IEEE Trans. on Signal Processing*, 51(3):839–852, March, 2003.

[134] M. Mibulevsky. Relative newton method for signal separation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 29–32, 6-10 April 2003.

[135] Michael Zibulevsky. Blind source separation with relative newton method. In *Proceedings of the International Conference on Independent Component Analysis and Signal Separation, Nara, Japan*, pages 897–902, 2003.

[136] Alexander M. Bronstein, Michael M. Bronstein, and Michael Zibulevsky. Blind source separation using the block-coordinate relative newton method. In *Proceedings of the International Conference on Independent Component Analysis and Signal Separation,ICA 2004, Granada, Spain*, pages 406–413, 2004.

[137] H. Shen, K. Hper, and A.J. Smola. Newton-like methods for nonparametric independent component analysis. *Lecture Notes in Computer Science: Neural Information Processing*, 4232 / 2006:1068–1077, 2006.

[138] M. Zibulevsky. Blind source separation using relative newton method combined with smoothing method of multipliers. *Tech. Report CCIT,EE Dept, Technion.*, No 556, 2005.

[139] M. Joho and K. Rahbar. Joint diagonalization of correlation matrices by using newton methods with application to blind signal separation. In *IEEE Sensor Array and Multichannel Signal Processing Workshop SAM, Rosslyn, VA,*, pages 403–407, 2002.

[140] Jun Lu, T.N. Davidson, and Z.-Q. Luo. Blind separation of bpsk signals using newton's method on the stiefel manifold. In *Proceedings Acoustics, Speech, and Signal Processing, 2003 (ICASSP '03)*, pages 301–304, 2003.

[141] G. B. Orr. *Dynamics and algorithms for Stochastic Learning.* PhD thesis, Depeartment of Computer Science and Engineering, Oregan Graduate Institure, Beaverton, 1995.

[142] G. B. Orr and T. K. Leen. Using curvature information for fast stochastic search. In *Advances in Neural Information Processing Systems 9*, 1996.

[143] P. Wolfe. Convergence conditions for ascent methods. *SIAM Review*, 11:226–235, 1969.

[144] P. Wolfe. Convergence conditions for ascent methods: some corrections. *SIAM Review*, 13:185–188, 1971.

[145] J. G. Proakis. Channel identification for high speed digital communications. *IEEE Trans. on Automat Control*, 19(11):916–922, Dec 1974.

[146] J. J. Shynk and S. Roy. The LMS algorithm with momentum updating. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, pages 2651–2654, June 6–9 1988.

[147] S. Roy and J.J. Shynk. Analysis of the momentum lms algorithm. *IEEE Trans. on Signal Processing*, 38(12):2088–2098, Dec 1990.

[148] Y. Tanik. *A study of the LMS adaptive filter and two new methods for convergence rate improvement.* PhD thesis, Department of Electrical Engineering, Middle East Technical University, June 1989.

[149] M.A. Tugay and Y. Tanik. Properties of the momentum lms algorithm. In *Electrotechnical Conference, 1989. Proceedings. 'Integrating Research, Industry and Education in Energy and Communication Engineering', MELECON '89., Mediterranean*, pages 197–200, 11-13 Apr 1989.

[150] M.A. Tugay and Y. Tanik. Properties of the momentum LMS algorithm. *Signal Processing*, 18(2):117–127, October 1989.

[151] L. Fausett. *Fundamentals of Neural Networks.* Prentice Hall International, 1994.

[152] A. Cichocki and R. Unbehauen. *Neural Networks for Signal Processing and Optimization.* Wiley, 1994.

[153] S. Haykin. *Neural Networks - A Comprehensive Foundation.* Prentice Hall, 2nd edition, 1998.

[154] Moreira M. and Fiesler E. Neural networks with adaptive learning rate and momentum terms. *IDIAP Technical Report Martigny, Switzerland*, 95-04, October 1995.

[155] B. A. Pearlmutter. Fast exact multiplication by the hessian. *Neuro Computation*, 6(1):147–160, 1994.

[156] Martin F. Moller. *Efficient training of feed-forward neural networks.* PhD thesis, Computer Science Dept., Aarhus University, December 1993.

[157] M. Moller. Exact calculation of the product of the hessian matrix of feed forward error function and a vector in O(n) time. *Neuro Computation*, 14(7):147–160, 1993.

[158] N N. Schraudolph and X. Giannakopoulos. Online independent component analysis with local learning rate adaptation. In *In Advances in Neural Information Processing Systems (NIPS), Cambridge, MA*, pages 789–795, 2000.

[159] N. N. Schraudolph. Fast curvature matrix vector products for second order gradient descent. *Neuro Computation*, 14(7):147–160, 2002.

[160] N N. Schraudolph. Fast curvature matrix-vector products. In *In Proc. Intl. Conf. Artificial Neural Networks (ICANN), Berlin, Vienna, Austria.* pages 19–26, 2001.

[161] Nicol N. Schraudolph. Fast curvature matrix-vector products for second-order gradient descent. *Neural Computation*, 14(7):1723–1738, 2002.

[162] T. Graepel and N N. Schraudolph. Stable adaptive momentum for rapid online learning in nonlinear systems. In *In Proc. Intl. Conf. Artificial Neural Networks (ICANN), Madrid, Spain*, pages 450–455, 2002.

[163] M. Elsabrouty, T. Aboulnasr, and M. Bouchard. A new diagonal hessian algorithm for blind signal separation. In *International Conference on Acoustics, Speech and Signal Processing*, 2006.

[164] B. Olshausen and D. Field. Wavelet-like receptive fields emerge from a network that learns sparse codes for natural images. *Nature*, 381:607–609, 1993.

[165] B. Olshausen and D. Field. Sparse coding with an overcomplete. basis set: a strategy employed by v1? *Vision Res*, 37:3311–3325, 1997.

[166] H. Choi, S. Kim, and S. Choi. A trust-region ica algorithm. In *In Proc. Korea Information Science Society, KAIST, Taejeon*, 23-24 April, 2004.

[167] H. Choi, S. Kim, and S. Choi. A trust-region ica algorithm. In *In 4th Postech-Kyutech Joint Workshop on Neuroinformatics, Kyushu, Japan*, 23-25 August, 2004.

[168] H. Choi, S. Kim, and S. Choi. A trust-region ica algorithm. In *In Proc. Int. Joint Conf. Neural Networks (IJCNN), Budapest, Hungary*, 25-29 July, 2004.

[169] H. Choi and S. Choi. Relative trust-region learning for ica. In *in Proc. Int. Conf. Acousitcs, Speech and Signal Processing (ICASSP), Philadelphia, PA*, 19-23 March, 2005.

[170] H. Choi and S. Choi. A relative trust-region algorithm for independent component analysis. *Neurocomputing*, In Press 2007.

[171] M. Elsabrouty, M. Bouchard, and T. Aboulnasr. Blind signal separation of audio signals using an on-line reduced hessian algorithm on the riemannian manifold. In *IEEE 2nd International Computer Engineering Conference (ICENCO 2006), Cairo, Egypt*, December 2006.

[172] H. Robbins and S. Monro. A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407, 1951.

[173] J. Kiefer and J. Wolfowitz. Stochastic estimation of a regression function. *Annals of Mathematical Statistics*, 23:462–466, 1952.

[174] J. C. Spall. A stochastic approximation technique for generating maximum likelihood parameter estimates. In *Proceedings of the American Control Conference*, pages 1161–1167. IEEE Transactions on Automatic Control, 1987.

[175] J. C. Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control*, 37:332–341, 1992.

[176] J.C. Spall. Implementation of the simultaneous perturbation algorithm for stochastic optimization. *IEEE Transactions on aerospace and electronic systems*, 34(3):817–823, 1998.

[177] J.C. Spall. *Introduction to Stochastic Search and Optimization: Estimation, Simulation and Control.* Wiley-Interscience, 2003.

[178] J. C. Spall and J. A. Cristion. Nonlinear adaptive control using neural networks: Estimation with a smoothed simultaneous perturbation gradient approximation. *Statistica Sinica*, 4:1–27, 1994.

[179] Y. Maeda, H. Hirano, and Y. Kanata. A learning rule of neural networks via simultaneous perturbation and its hardware implementation. *Neural Networks*, 8:251–259, 1995.

[180] J. C. Spall. On the use of simultaneous perturbation stochastic approximation for neural network training. In *Proceedings of the American Control Conference, San Diego, CA*, pages 388–392, 2-4 June 1999.

[181] B.L. Chan, A. Doucet, and V.B. Tadic. Optimisation of particle filters using simultaneous perturbation stochastic approximation. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Hong Kong*, pages 681–684, April 2003.

[182] R. Burnett. Application of stochastic optimization to collision avoidance. In *Proceedings of the American Control Conference, Boston, MA*, pages 2789–2794, 29 June2 July 2004.

[183] Kleinman N.L., Hill S.D., and Ilenda V.A. SPSA/SIMMOD optimization of air traffic delay cost. In *Proceedings of the American Control Conference, Albuquerque, NM*, pages 1121–1125, 4-6 June 1997.

[184] G. Morison and T. Durrani. Spsa for noisy non-stationary blind source separation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 285–288, 6-10 April 2003.

[185] S. Martin, G. Morison, W. Nailon, and T. Durrani. Fast and accurate image registration using tsallis entropy and simultaneous perturbation stochastic approximation. *Electronics Letters*, 40(10):595–597, 13 May 2004.

[186] Blum J. R. Multidimensional stochastic approximation methods. *Annals of Mathematical Statistics*, 25(4):737–744, Dec. 1954.

[187] Shuxue Ding, Teruo Niitsuma, and Kazuyoshi Sugai. New algorithms for ica via simultaneous perturbation stochastic approximation method. *Technical Report of IEICE, NC2001-80*, pages 75–82, 2001.

[188] Shuxue Ding, Daming Wei, and S. Omata. An algorithm for real-time independent component analysis in dynamic environments. In *The 2004 47th Midwest Symposium on Circuits and Systems, 2004. MWSCAS '04*, pages 105–108, 25-28 July 2004.

[189] Shuxue Ding, Jie Huang, Daming Wei, and Sadao Omata. Real-time independent component analysis based on gradient learning with simultaneous perturbation stochastic approximation. In *The 8th International Conference on Knowledge Based Intelligent Information and Engineering Systems, Lecture Notes in Artificial Intelligence 3214*, pages 366–374, 22-24 Sep 2004.

[190] Shuxue Ding, Jie Huang, Daming Wei, and S. Omata. Signal extensions in independent component analysis and its application for real-time processing. In *The Fourth International Conference on Computer and Information Technology, 2004. CIT '04.*, pages 839–844, 14-16 Sept. 2004.

[191] Y. Maeda and K. Tsushio. Blind signal separation via simultaneous perturbation method. In *Proceedings of the International Joint Conference on Neural Networks, Honolulu, HI*, pages 439–443, 12-17 May 2002.

[192] Yutaka Maeda and Takayuki Maruyama. Natural gradient using simultaneous perturbation without probability densities for blind source separation. In *Proceedings of the International Conference on Independent Component Analysis and Signal Separation, Nara, Japan.* pages 439–443. 2003.

[193] L. Parra and P. Sajda. Blind source separation via generalized eigenvalue decomposition. *Journal of Machine Learning Research*, 4(7-8):1261–1269, 2004.

[194] M. Joho, R. H. Lambert, and H. Mathis. Elementary cost functions for blind separation of non-stationary source signals. In *Proc. ICASSP 2001*, pages 2793–2796, Salt Lake City, UT, May 2001.

[195] M. Joho and K. Rahbar. Joint diagonalization of correlation matrices by using gradient methods with application to blind signal separation. In *IEEE Sensor Array and Multichannel Signal Processing Workshop SAM, Rosslyn, VA,*, pages 273–277, 2002.

[196] A. Ziehe and K.-R. Müller. TDSEP—an efficient algorithm for blind separation using time structure. In *Proc. Int. Conf. on Artificial Neural Networks (ICANN'98)*, pages 675–680, Skvde, Sweden, 1998.

[197] A. Yeredor. Blind source separation based on second-order statistics with asymptotically optimal weighting. In *Invited to a special session on Statistical and Array Signal Processing at The 4th World Multiconference on Systemics, Cybernetics and Informatics (SCI2000)*, pages 247–251. Orlando. Florida, 2000.

[198] A. Yeredor. Blind separation of gaussian sources via second-order statistics with asymptotically optimal weighting. *IEEE Signal Processing Letters*, 7(7):197–200, July 2000.

[199] P. Tichavsk, Z. Koldovsk, E. Doron, A. Yeredor, and G. G. Herrero. Blind signal separation by combining two ica algorithms: Hos-based efica and time structure-based wasobi. In *In: Proceedings of the 14th European Signal Processing Conference (EUSIPCO 2006)*. Florence, Italy, September 2006.

[200] P. Tichavsk, Z. Koldovsk, A. Yeredor, G. G. Herrero, and E. Doron. A hybrid technique for blind non-gaussian and time-correlated sources using a multicomponent approach. *submitted to IEEE Trans. on Neural Networks*, January 2007.

[201] A. Ziehe. *Blind Source Separation based on joint diagonalization of matrices with applications in biomedical signal processing.* PhD thesis, University of Potsdam, April 2005.

[202] A. Yeredor. Blind source separation using the second derivative of the second characteristic function. In *Proceedings of The 2000 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'2000)*, pages 3136–3139, Istanbul, Turkey, 2000.

[203] A. Yeredor. A novel approach for blind source separation based on the characteristic function. In *Proceedings of The IEEE Sensor Array and Multichannel Processing Workshop (SAM'2000)*, pages 365–369, Boston, Massachusetts, March 2000.

[204] A. Yeredor. Blind source separation via the second characteristic function. *Signal Processing*, 69(5):897–902, May 2000.

[205] A. Belouchrani and M. Amin. Blind source separation using time-frequency distributions: Algorithm and asymptotic performance. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'97)*, pages 49–53, Munich, Germany, 1997.

[206] A. Belouchrani and M. Amin. Blind source separation based on time-frequency signal representations. *IEEE Trans. on Signal Processing.* 46(11):2888–2897, 1998.

[207] J.-F Cardoso. On the performance of orthogonal source separation algorithms. In *Proc. EUSIPCO*, pages 776–779, Edinburgh, Setember 1994.

[208] L. DeLathauwer, B. De Moor, and J. Vandewalle. A prewhitening-induced bound on the identification error in independent component analysis. *IEEE Transactions on Circuits and Systems I*, 52(3):546–554, March 2005.

[209] D. T. Pham. Joint approximate diagonalization of positive definite matrices. *SIAM J. on Matrix Anal. and Appl*, pages 1136–1152, 2001.

[210] Angelika Bunse-Gerstner, Ralph Byers, and Volker Mehrmann. Numerical methods for simultaneous diagonalization. *SIAM Journal of Matrix Analysis and Applications*, 14(4):927–949, 1993.

[211] A. Yeredor. On using exact joint diagonalization for non-iterative approximate joint diagonalization. *IEEE Signal Processing Letters*, 12(9):645–648, September 2005.

[212] A. Yeredor. Approximate joint diagonalization using non-orthogonal matrices. In *Proceedings of The International Workshop on Independent Component Analysis and Blind Source Separation (ICA2000)*, pages 33–38, Helsinki, Finland, June 2000.

[213] A. Yeredor, A. Ziehe, and K.-R. Mller. Approximate joint diagonalization using a natural-gradient approach. In *5th International Conference on ICA*. Granada, Spain, September 2004.

[214] B. Afsari. Gradient flow based matrix joint diagonalization for independent componenet analysis. *Maryland University Masters Thesis*. pages 46–47. 2004.

[215] B. Afsari and P. S. Krishnaprasad. Some gradient based joint diagonalization methods for ica. In *Fifth International Conference Independent Component Analysis and Blind Signal Separation, Granada, Spain*, pages 437–444, 2004.

[216] B. Afsari and P. S. Krishnaprasad. A novel non-orthogonal joint diagonalization cost function for ica. *Technical Report TR 2005-106, The Institute for Systems Research, University of Maryland*, 2005.

[217] A. Ziehe, P. Laskov, KR. Muller, and G. Nolte. A linear least-squares algorithm for. joint diagonalization. In *Proceedings ICA2003*, page 469474, 2003.

[218] A. Ziehe, P. Laskov, G. Nolte, and KR. Muller. A fast algorithm for joint diagonalization with non-orthogonal transformations and its application to blind source separation. *The Journal of Machine Learning Research*, pages 777–800, July 2004.

[219] R. Vollgraf and K. Obermayer. Quadratic optimization for simultaneous matrix diagonalization. *IEEE Trans. on Signal Processing*, 54(9):3270–3278, September 2006.

[220] A. Yeredor. Non-orthogonal joint diagonalization in the least-squares sense with application in blind source separation. *IEEE Trans. on Signal Processing*, 50(7):1545–1553, July 2002.

[221] J. C. Spall. Adaptive stochastic approximation by the simultaneous perturbation method. *IEEE Transactions on aerospace and electronic systems*, 45:817–823, 2000.

[222] J. C. Spall. Feedback and weighting mechanisms for improving jacobian (hessian) estimates in the adaptive simultaneous perturbation algorithm. In *Proceedings of the American Control Conference*, pages 6–12, 2006.

[223] J. C. Spall. Feedback and weighting mechanisms for improving learning in the adaptive simultaneous perturbation algorithm. In *Proceedings of the Performance Metrics for Intelligent Systems Workshop*, 2006.

[224] Zhan-Li Sun, De-Shuang Huang, Chun-Hou Zheng, and Li Shang. Optimal selection of time lags for tdsep based on genetic algorithm. *Neurocomputing*, 69(7-9):884–887, 2006.

[225] R. Joaquim, R. A. Martins, P. Sturdza, and J. J. Alonso. The connection between the complex-step derivative approximation and algorithmic differentiation. *Proceedings of the 39th Aerospace Sciences Meeting, Reno, NV*, January 2001.

[226] R. Joaquim, R. A. Martins, Peter Sturdza, and Juan J. Alonso. The complex-step derivative approximation. *ACM Transactions on Mathematical Software*, 29(3):245–262, September 2003.

[227] P. A. Viola. *Alignment by Maximization of Mutual Information.* PhD thesis, MIT, June 1995.

[228] P. A. Viola and W. M. Wells III. Alignment by maximization of mutual information. *International Journal of Computer Vision*, 22(2):137–154, 1997.

[229] W. M. Wells III, P. A. Viola, H. Atsumi, S. Nakajima, and R. Kikinis. Multi-modal volume registration by maximization of mutual information. *Medical Image Analysis*, 1:35–52, 1996.

[230] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. Multimodality image registration by maximization of mutual information. *IEEE Trans Med Imaging*, 16(2):187–198, 1997.

[231] A. Renyi. On measures of entropy and information. In *Proceedings 4th Berkeley Symp. Mathematical Statistics Probability*, pages 547–561, Berkeley, University of California, 1961.

[232] C. Tsallis. Possible generalization of Boltzmann-Gibbs statistics. *J. Stat. Physics*, 52:479–487, 1988.

[233] D. Capel and A. Zisserman. Computer vision applied to super resolution. *IEEE Signal Processing Mag*, 20(3):75–86, 2003.

[234] Y. Getian. *Image registration and super-resolution mosaicing.* PhD thesis, University of New South Wales - Australian Defence Force Academy, September 2005.

[235] J. Le Moigne, W. J. Campbell, and R. F Cromp. An automated parallel image registration technique based on the correlation of wavelet features. *IEEE Transactions on Geoscience and Remote Sensing*, 40(8):1849 – 1864, 2002.

[236] A. Cole-Rhodes, K. Johnson, and J. LeMoigne. Blind signal separation into groups of dependent signals using joint block diagonalization. In *Proc. SPIE Aerosense, Wavelet Applications IX, vol. 4738*, pages 44–85, Orlando Fl., 2002.

[237] A. Cole-Rhodes, K. Johnson, J. LeMoigne, and I. Zavorin. Multiresolution registration of remote sensing imagery by optimization of mutual informa-

tion using a stochastic gradient. *IEEE Transactions on Image Processing*, 12:1495–1511, 2003.

[238] J.B. Antoine Maintz and M. A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2:1–36, 1998.

[239] J.P.W Pluim, J.B.A. Maintz, and M.A Viergever. Mutual-information-based registration of medical images: a survey. *IEEE Transactions on Medical Imaging*, 22(8):986–1004, 2003.

[240] L. Brown. A survey of image registration techniques. *ACM Computing Surveys*, 24(4):325–376, 1992.

[241] B. Zitova and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21(11):977–1000, 2003.

[242] J. Modersitzki. *Numerical Methods for Image Registration.* Oxford University Press Series: Numerical Mathematics and Scientific Computation, 2004.

[243] K. William and Y. Chun. Active edge maps for medical image registration. In *Proc. SPIE Vol. 4322, Medical Imaging*, pages 516–526, 2001.

[244] A. Franz, I. C. Carlsen, S. Kabus, T. Netsch, V. Pekar, and S. Renisch. Modular toolbox for derivative-based medical image registration. In *Proceedings of SPIE 2005*, pages 1222–1233, 2005.

[245] J.P.W Pluim, J.B.A. Maintz, and M.A Viergever. A contour based approach to multisensor image registration. *IEEE Transactions on Image Processing.* 4(3):320–334, 1995.

[246] B. Fischer and J. Modersitzki. Curvature based image registration. *Journal of Mathematical Imaging and Vision*, 18(1):81–85, 2003.

[247] S. Henn. A full curvature based algorithm for image registration. *Journal of Mathematical Imaging and Vision*, 24(3):195–208, 2006.

[248] E. Guest, E. Berry, R. A. Baldock, M. Fidrich, and M. A. Smith. Robust point correspondence applied to two and three-dimensionalimage registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):165–19, 2001.

[249] Volume Image Registration by Template Matching. L. ding and a. goshtasby and m. satter. *Image and Vision Computing*, 19(12):821–832, 2001.

[250] W. R. Crum, T. Hartkens, and D. L. G. Hill. Non-rigid image registration: Theory and practise. *The British Journal of Radiology*, 77:140–153, 2004.

[251] H. Lester and S. Arridge. A survey of hierarchical non-linear medical. image registration. *Patter Recognition*, 32:129–149, 1999.

[252] S. Periaswamy and H. Farid. Differential elastic image registration. *Technical Report 2001-413, Dartmouth College, Computer Science*, 2001.

[253] Aloys du Bois d'Aische. *Multimodal Images Registration Constrained by Rigid Structure with Applications in Radiotherapy.* PhD thesis, Université Catholique du Louvain, August 2005.

[254] L. Zagorchev and A. Goshtasby. A comparative study of transformation functions for nonrigid image registration. *IEEE Trans. Image Processing*, 15(3):529–538, 2006.

[255] W. K. Pratt. *Digital image processing.* Wiley-Interscience Publication, New-York, tats-Unis, 1978.

[256] R. K. Sharma and M. Pavel. Multisensor image registration. In *Society for Information Display Vol. XXVIII*, pages 951–954, 1997.

[257] E. DAgostino, J. Modersitzki, F. Maes, D. Vandermeulen, B. Fischer, and P. Suetens. Free-form registration using mutual information and curvature regularization. *Lecture Notes in Computer Science: Neural Information Processing*, 2717 / 2003:11–20, 2003.

[258] F. Maes, D. Vandermeulen, P., and Suetens. Comparative evaluation of multiresolution optimization strategies for multimodality image registration by maximization of mutual information. *Medical Image Analysis*, 3(4):373–386, 1999.

[259] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Research Nat. Bur. Standards*, 49:409–436, 1952.

[260] W. C. Davidon. Variable metric method for minimization. *A.E.C. Research and Development Report*, 1959.

[261] R. Fletcher and M. J. D. Powell. A rapidly convergent descent method for minimization. *Computer Journal*, 6:163–168, 1963.

[262] K. Levenberg. A method for the solution of certain problems in least squares. *Quart. Appl. Math.*, 2:164–168, 1944.

[263] D. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM J. Appl. Math.*, 11:431–441, 1963.

[264] P. Thvenaz and M. Unser. Optimization of mutual information for multiresolution image registration. *IEEE Transactions on Image Processing*, 9(12):2083–2099, 2000.

[265] A. Collignon, F. Maes, D. Delaere. D. Vandermeulen. P. Suetens. and G. Marchal. Automated multimodality image registration based on infor-

mation theory. In *Information Processing in Medical Imaging (Y. Bizais, C. Barillot and R. Di Paola, eds.)*, pages 263–274, Orlando Fl., 1995.

[266] J.P.W. Pluim. *Multi-modality matching using mutual information, Master Thesis.* PhD thesis, University of Groningen, Groningen, The Netherlands, June 1996.

[267] C. Studholme, D. L. G. Hill, and D. J. Hawkes. Auomatic 3d registration of magnetic resonance and positron emission tomography brain images by multiresolution optimization of voxel similarity measures. *Med. Phys.*, 24(1):25–35, 1997.

[268] D. I. Barnea and H. F. Silverman. A class of algorithms for fast digital image registration. *IEEE Transactions on Computers*, 22(2):179–186, 1972.

[269] W. K. Pratt. Correlation techniques of image registration. *IEEE Transactions on Aerospace and Electronic Systems*, AES-10(3):353–358, 1974.

[270] E. Parzen. On the estimation of a probability density function and the mode. *Annals of Math. Stats*, 33:1065–1076, 1962.

[271] Arunava Banerjee Ajit Rajwade and Anand Rangarajan. Continuous image representations solve the histogram binning problem in mutual information-based image registration. In *IEEE International Symposium on Biomedical Imaging (ISBI)*, Arlington, Virginia, USA, 2006.

[272] Arunava Banerjee Ajit Rajwade and Anand Rangarajan. New method of probability density estimation with application to mutual information based image registration. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, pages 1769–1776, 2006.

[273] T.M. Lehmann, C. Gonner, and K. Spitzer. Survey: interpolation methods in medical image processing. *IEEE Transactions on Medical Imaging*, 18(11):1049–1075, November 1999.

[274] I. J. Schoenberg. Contributions to the problem of approximation of equidistant data by analytic functions. *Quart. Applied Math*, 4:45–99, 112–141, 1946.

[275] M. Unser. Splines: A perfect fit for signal and image processing. *IEEE Signal Processing Magazine*, 16(2):22–38, 1999.

[276] M. Unser. Splines: A perfect fit for medical imaging. In *Proc. SPIE Vol. 4684, Medical Imaging*, pages 225–236, 2002.

[277] T.M. Lehmann, C. Gonner, and K. Spitzer. Survey: interpolation methods in medical image processing. *IEEE Transactions on Medical Imaging*, 20(7):660–665, July 2001.

[278] I. Csiszar. Information type measures of differences of probability distribution and indirect observations. *Studia Math. Hungarica*, pages 299–318, 1967.

[279] Y. He, A. B. Hamza, and H. Krim. A generalized divergence measure for robust image registration. *IEEE Trans. Signal Processing*, 51(5):299–318, 2003.

[280] A. Cole-Rhodes and A. Adenle. Automatic image registration by stochastic optimization of mutual information. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 313–316, 2003.

[281] A. Cole-Rhodes, K. Johnson, and J. LeMoigne. Image registration using a 2nd order stochastic optimization of mutual information. *International Geoscience and Remote Sensing Symposium*, 6:4038–4040, 2003.

[282] Q. Li and I. Sato Y. Murakami. Simultaneous perturbation stochastic approximation algorithm for automated image registration optimization. In *Proceedings IEEE International Geoscience and Remote Sensing Symposium and 27th Canadian Symposium on Remote Sensing (IGARSS 2006). Remote Sensing: A Natural Global Partnership*, pages 184–187, 2006.

[283] Q. Li and I. Sato Y. Murakami. Automated image registration using stochastic optimization strategy of mutual information. In *Proceedings International Conference of Sensing, Computing and Automation*, pages 2872–2877, 2006.

[284] S. Ikeda and N. Murata. A method of ICA in time-frequency domain. In *Proc. Int. Workshop on Independent Component Analysis and Signal Separation (ICA'99)*, pages 365–370, Aussois, France, 1999.

[285] M. Kim and S. Choi. Ica-based clustering for resolving permutation ambiguity in frequency-domain convolutive source separation. In *18th International Conference on Pattern Recognition (ICPR'06)*, pages 950–954, Aussois, France, 2006.

[286] A. Griewank. *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*. Number 19 in Frontiers in Appl. Math. SIAM, Philadelphia, PA, 2000.

[287] B. Horn and B. Schunk. Determining optical flow. *Artificial Intelligence*, 17:185–204, August 1981.

[288] M. Irani and S. Peleg. Motion analysis for image enhancement: Resolution, occlusion, and transparency. *Journal of Visual Communication and Image Representation*, 4(4):324–335, December 1993.

[289] Y. Keller, Y. Shkolnisky, and A. Averbuch. The angular difference function and its application to image registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(6):969–976, June 2005.

[290] N. Mitianoudis, T. Stathaki, and M. Davies. Blind separation of skewed signals in instantaneous mixtures. *IEEE Workshop on Signal Processing Systems*, 2005.

[291] J. Karvanen and V. Koivunen. Blind separation methods based on pearson system and its extensions. *Signal Processing*, 82(4), 2002.

[292] C. L. Nikias and A. Petropulu. *Higher Order Spectra Analysis, A nonlinear Signal Processing Framework.* Prentice Hall Signal Processing Series, Cambridge, MA, 1993.

[293] M. Rosenblatt. *Stationary Sequences and Random Fields.* Birkhauser, 1985.

[294] D. Xu, J. Principe, and J. Fisher. A novel measure for independent component analysis. *in Proc. IEEE ICASSP98*, 2:1161–1164, 1998.

[295] R.A. Redner and H.F. Walker. Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review*, 26(2):195–239, 1984.

[296] J. Karvanen, J. Erikson, and V. Koivunen. Pearson system based method for blind separation. *Proceedings of Second International Workshop on Independent Component Analysis and Blind Signal Separation*, 2000.

[297] J. Karvanen, J. Erikson, and V. Koivunen. Adaptive score functions for maximum likelihood ica. *Journal of VLSI Signal Processing Systems*, 32:83–92, 2002.

[298] A. Hald. The early history of the cumulants and the gram-charlier series. *International Statistical Review / Revue Internationale de Statistique*, 68(2):137–153, August 2000.

[299] M. Kendall and A. Stuart. *The Advanced Theory of Statistics.* Griffin, 1958.

[300] M. Kendall and A. Stuart. *The Advanced Theory of Statistics, Vols. 1–3.* Macmillan, 1976–1979.

[301] M. Kendall. *Multivariate Analysis.* Griffin, 1975.

[302] S. Blinnikov and R. Moessner. Expansions for nearly gaussian distributions. *Astronomy and Astrophysics Supplement Series*, 130:193–205, 1998.

[303] G. Golub and C. Reinsch. Singular value decomposition and least squares solutions. *Numerische Mathematik*, 14:403–420, 1970.

[304] G. Golub and W. Kahan. Calculating the singular values and pseudo-inverse of a matrix. *Journal SIAM Numerical Analysis*, 22(2):205–224, 1965.

[305] E. Rothwell and B. Drachman. A unified approach to solving ill-conditioned matrix problems. *International Journal for Numerical Methods in Engineering*, 28(3):609–620, 1988.