Early seizure onset prediction in clinical recordings of EEG

OLIVER GIBSON

MSc by Research in Pattern Analysis and Neural Networks

ASTON UNIVERSITY

August 2002

ASTON UNIVERSITY

Early seizure onset prediction in clinical recordings of EEG

OLIVER GIBSON

MSc by Research in Pattern Analysis and Neural Networks, 2002

Thesis Summary

We apply and evaluate a new variation of Independent Component Analysis (ICA) to the problem of seizure onset analysis in electroencephalographic (EEG) signals in epilepsy. This constrained ICA (CICA) algorithm takes a reference signal as input, along with observed multichannel data, and extracts the independent source which is closest, in some sense, to the reference. We present an implementation of this algorithm whose behaviour has been initially tested using a simple data set. The performance of the algorithm for this application is then assessed using synthetic seizure waveform mixed with real-world EEG epochs, including some common artifacts. The application of the algorithm to both artifact rejection and the extraction of seizure waveform from real EEG recordings of seizures is shown to be possible. For seizure onset analysis, the required *a priori* reference signal can be based on the frequency and phase of a period of known seizure waveform. The CICA algorithm can then be used to extract seizure waveform from time periods prior to the clinical onset of the seizure. The algorithm implicitly calculates an estimated spatial distribution of the extracted source, which can indicate the focus of a seizure.

Keywords: Constrained Independent Component Analysis, Electroencephalogram, seizure onset, Epilepsy

Acknowledgements

I am very grateful to the Neural Computing Research Group, Aston University, for financial support during the period of this work. Many thanks to my fellow MSc students Mani, Manu, Rémi, Nicolas and Borémi and all members of the NCRG for their encouragement over the past year. I am indebted to my supervisor, Dr Christopher James, for his immense support and advice throughout the project.

Contents

List of Figures

List of Tables

Publications

The following refereed journal papers and conference papers have been produced along with the work described in this thesis:

- Gibson, O. and James, C. J., *Constrained ICA for Seizure Onset Analysis in the EEG.* Proceedings of IEEE EMBS UK & RI Postgraduate Conference in Biomedical Engineering and Medical Physics, July 2002, Birmingham, UK. ISBN 0-9543157-0-7

- James, C. J. and Gibson, O., *Temporally Constrained ICA: an application to Artifact Rejection in Electromagnetic Brain Signals.* Submitted to IEEE Transactions on Biomedical Engineering, July 2002.

- James, C. J. and Gibson, O., *ICA with a Reference: Extracting Desired Electromagnetic Brain Signals.* Accepted for the IEE Medical Applications of Signal Processing seminar, London, UK, October 2002.

- Gibson, O. and James, C. J., *On the analysis of seizure onset in the EEG: the application of Constrained ICA.* Accepted for 2nd European Medical and Biological Engineering Conference, December 2002, Vienna, Austria.

- James, C. J. and Gibson, O., *Electromagnetic Brain Signal Analysis using Constrained ICA.* Accepted for 2nd European Medical and Biological Engineering Conference, December 2002, Vienna, Austria.

Chapter 1

Introduction

1.1 Epilepsy and electromagnetic brain signals

Epilepsy is a condition characterised by the recurrence of epileptic seizures, which are categorised as partial (involving only part of the brain at the outset) or generalised [2]. The most common treatment is by anti-epilepsy drugs [3]. If these drugs prove ineffective then, in some cases, surgery may be performed but this requires knowledge of the source of the seizures within the brain. Such knowledge of the "focus" of seizures can only be obtained from analysis of the electromagnetic brain signals. Ictal recordings (that is, those made during a seizure) generally show characteristic seizure waveforms such as those shown later on, in Chapter 6, while during an inter-ictal recording (made between seizures) epileptiform discharges are often observed. These discharges are short-duration "spikes", an example of which is shown in Chapter 4.

A common technique for studying brain activity is the electroencephalogram (EEG). Work on the EEG commenced in 1929 when the electrical activity of the brain was first studied using electrodes placed on the scalp [4]. From 1935, this activity was recordable using ink charts, but clinical electroencephalography emerged with the development of electronics from 1940. This was because of the high-gain amplification required - the potentials at the scalp are typically between 10 and 300 microvolts. A modern EEG recording is generally sampled at 200 Hz using 25 electrodes placed in standard positions on the head (Figure 1.1) [5]. Non-invasive scalp electrodes are most commonly used, despite the "smearing" effect of the skull on the signals. (A more accurate recording of the brain potentials can be obtained using intra-cranial electrodes, but the risk of medical complications makes them less widely used for clinical EEG recordings.)

Alternative modern methods of recording brain activity include the magnetoencephalogram (MEG) which records variations in the brain's magnetic field. The MEG requires a magnetically-shielded room and the patient must remain still during the recording. A MEG installation is also much more expensive than an EEG system, which may also be portable [6]. The EEG therefore remains a popular technique for analysis of brain signals in order to diagnose epilepsy or study the source of seizures for possible surgical treatment.

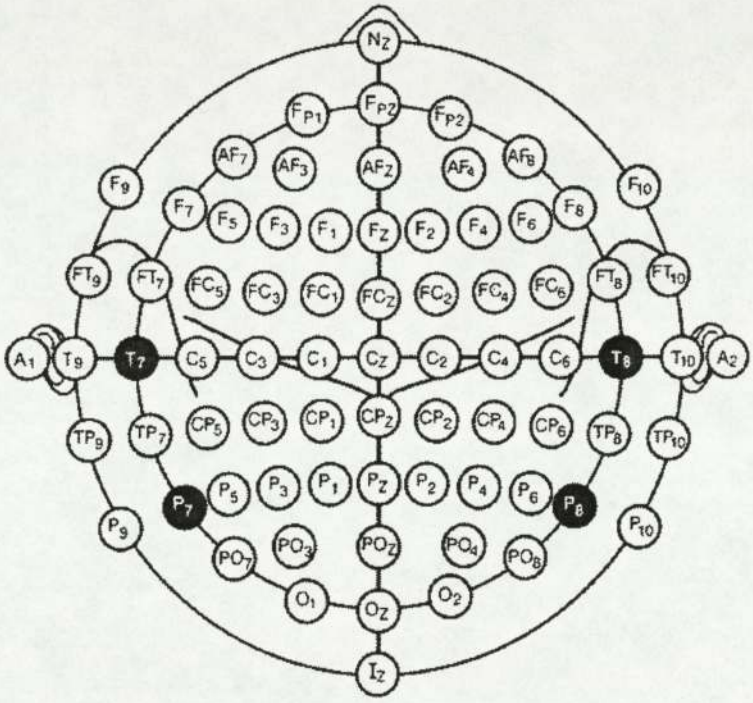


Figure 1.1: The standard 10-20 system for EEG electrode placement, reproduced from Sundaram *et al* [1]. The electrodes labelled T7, T8, P7 and P8 may also be labelled as T3, T4, T5 and T6 respectively in the recordings described in this thesis. Not all electrode positions are necessarily used; a typical EEG recording uses 25 electrodes. Other electrodes may be placed, for example, close to the eye to capture eye movement activity (an electro-oculogram or EOG).

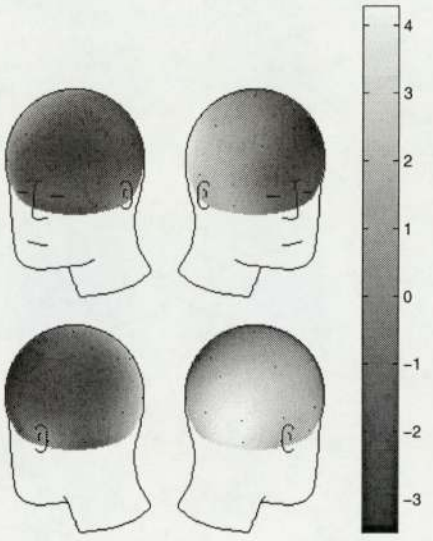


Figure 1.2: Example of topographic map, obtained by interpolating between values of potential at each electrode position. The colour bar shows how the extremes of potential are scaled to extremes of white and black. The polarity of the plot is indeterminate: the greyscale may be reversed for the same topography.

1.2 Electroencephalogram recording and analysis

The scalp potentials are typically recorded using one of two referencing systems ("montages") [7]. A referential montage uses a single electrode as a common reference, against which the potentials at other parts of the head are measured. A bipolar montage involves the recording of differences in potential between different electrodes, normally forming a "chain". The EEG recordings discussed in this thesis all use a referential montage and have the sampled data stored in a matrix. The standard preprocessing practice of subtracting row and column means from the EEG data matrix was followed, which removes any d.c. bias on each channel and cancels the referencing to a single electrode.

The use of multiple scalp electrodes provides some information on the spatial distribution of observed waveforms. Software is available [8] which interpolates between the potentials observed at each electrode position, using a spherical head model, to produce a topographic map of the waveforms. An example of a topography plotted in this way is shown in Figure 1.2. If an algorithm produces spatial information, this can be plotted in the same way, as shown in Chapter 4 and later chapters. Similarly, it is possible to generate simulated EEG data using a spherical model of the head. For this, the electrical current from a group of neurons is commonly modelled as an equivalent current dipole[1]. One or more current dipoles can be modulated with a waveform which approximates seizure onset (e.g. a sine wave whose amplitude increases with time). The potentials at each scalp electrode can then be calculated. This procedure was used to generate the simulated seizure waveform shown in Chapter 4.

[1] A current dipole has dimensions of current multiplied by length and has infintesimally small length. A dipole is described by six parameters: three defining its position, two defining its orientation and one defining its strength.

1.3 Seizure onset analysis

For some time, it was thought that the transition between inter-ictal EEG and ictal EEG was sudden; a two-state model was assumed. More recently it was demonstrated that an intermediate state exists in which "neuron recruitment" takes place, leading to a seizure [9][2]. This work used non-linear methods to compare time windows of EEG data from periods of clearly-identifiable seizure to pre-seizure periods. (Another study which did not use specially selected data sets showed that linear methods frequently performed at least as well as non-linear methods [12].)

The existence of this intermediate state allows analysis of seizure onset, with two broad aims:

- Investigation of the "source" of the seizure in the brain may lead to possible surgical treatment.
- A robust seizure onset prediction algorithm could be used in a real time system to warn patients of an impending seizure in time for precautions to be taken.

However, extraction of seizure waveforms from EEG data has hitherto proved difficult because of the nonstationarity of seizure waveforms and corruption of the EEG signals by artifacts, for example due to muscle movement. This project investigated the novel use of a Constrained ICA algorithm for the extraction of seizure waveform from time periods prior to the visible onset of the seizure. This algorithm offers advantages over the non-ICA methods mentioned above - the assumption that a small number of statistically independent sources underlie EEG observations seems physiologically reasonable. Indeed, conventional ICA, with a square mixing matrix, has previously been used to extract seizure waveforms from pre-seizure EEG data [13]. This work demonstrated the usefulness of ICA as a tool for investigation of biomedical electromagnetic signals, including analysis of seizure onset. The principal drawback of this method was that manual identification of the waveforms of interest among all the extracted components was necessary. This identification was subjective and time-consuming. A principal aim in using a constrained form of ICA was to avoid such manual intervention.

1.4 Thesis overview

The aim of this project is therefore to address the problem of manual source identification in conventional ICA by the use of a Constrained ICA (CICA) algorithm. In effect, the source of interest can then be selected automatically, if a suitable reference signal is provided to the algorithm. Although several papers have been published on the CICA algorithm no source code for the algorithm was publically available at the time of writing. The algorithm was therefore implemented as part of this project, as discussed in the following chapter. The algorithm was then tested using a simple "toy" data set in which all the true underlying sources were known (Chapter 3). The effect of several important algorithm parameters was investigated. The problem of extraction of EEG seizure waveform is addressed in Chapter 4, where trials are described with a synthetic seizure waveform added to real-world EEG epochs. In this work, the desired underlying source is therefore known, allowing automatic assessment of the trial results. Chapter 5 shows how the CICA algorithm can be applied to the slightly different problem of artifact rejection. The removal of EEG artifacts is an important research area in which the CICA algorithm appears to offer a novel method. Furthermore, artifact removal is relevant to

[2]The work described in this paper used recordings from intra-cranial electrodes; scalp electrode recordings, such as those used for this project, have been shown to give similar information [10] [11].

seizure onset analysis because artifacts may otherwise obscure the seizure waveforms. The use of the algorithm for analysis of the onset of real-world seizures is then discussed in Chapter 6, while Chapter 7 draws conclusions from the work described in this thesis.

Chapter 2

Implementation of the Constrained ICA algorithm

The problem of seizure onset analysis and the idea of addressing it by extracting seizure waveforms from EEG data were described in the previous chapter. This chapter shows how the Constrained ICA algorithm is a further development within the well-established field of Independent Component Analysis. Since, at the time of writing, there was no publically-available implementation of CICA, the algorithm was implemented in Matlab as part of this project. This implementation, for which the complete code is shown in Appendix A, is described and aspects pertinent to its application to seizure onset analysis are highlighted.

2.1 Independent Component Analysis

Most ICA algorithms [14] take as inputs N observations $\mathbf{x}(t)$ which are assumed to be the result of linear, noiseless mixing of M sources: $\mathbf{x}(t) = \mathbf{A}\mathbf{c}(t)$. The sources are assumed to be statistically independent so that they can be extracted, for example, by minimizing the mutual information between estimated sources $\mathbf{k}(t)$. A separating matrix $\mathbf{W}$ is found such that $\mathbf{k}(t) = \mathbf{W}\mathbf{x}(t)$.

The problem can be viewed in an information-maximisation framework [15] in which information is transmitted through a mapping $\mathbf{f} : \mathbf{x} \rightarrow \mathbf{y}$. Using a single layer neural network implementation, with non-linear outputs:

$$\mathbf{k} = \mathbf{W}\mathbf{x} \tag{2.1}$$

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

where $\mathbf{g}$ is the neural network's nonlinear function, which is applied separately to each element of $\mathbf{k}$.

The mutual information between the input and output is then:

$$I[\mathbf{x};\mathbf{y}] = H[\mathbf{x}] + H[\mathbf{y}] - H[\mathbf{x},\mathbf{y}] = H[\mathbf{y}] - H[\mathbf{y}|\mathbf{x}] \tag{2.3}$$

Writing the joint entropy of the outputs $H[\mathbf{y}]$ in terms of the mutual information between outputs $I[\mathbf{y}]$:

$$I[\mathbf{x};\mathbf{y}] = \sum_{m=1}^{M} H[y_m] - I[\mathbf{y}] - H[\mathbf{y}|\mathbf{x}] \tag{2.4}$$

Given low noise the uncertainty in $\mathbf{y}$ given $\mathbf{x}$, $H[\mathbf{y}|\mathbf{x}]$, can be ignored. The transmitted information can then be maximised by simultaneously maximising the individual output entropies while minimising the mutual information between them. The mutual information is invariant under the monotone transformation $\mathbf{g}$. Separation can therefore be achieved by optimising $\mathbf{g}$ and $\mathbf{k}$ to maximise the information transmitted through $\mathbf{f}$.

An alternative approach is based on the fact that independent components are maximally non-Gaussian, since, by the Central Limit Theorem, sums of non-Gaussian random variables are nearer Gaussian than the original variables [16]. Hence, an output $k_i = \mathbf{w}_i^T\mathbf{x}$ will be maximally non-Gaussian if it equals an independent source. A measure of non-Gaussianity, such as absolute kurtosis, could therefore be used to separate sources. However, kurtosis is not a robust measure since a small number of erroneous or noisy outliers may significantly alter the practical estimate of kurtosis. A more robust, though computationally more expensive, measure of non-Gaussianity is *negentropy*. A result from information theory is that a Gaussian variable has the largest entropy among all random variables of equal variance. Entropy can therefore be used as a measure of non-Gaussianity. Negentropy is a normalised version of differential entropy which is zero for Gaussian variables and always non-negative. It is defined as:

$$\text{Negentropy } J(\mathbf{y}) = H(\mathbf{y}_{Gauss}) - H(\mathbf{y}) \tag{2.5}$$

where $\mathbf{y}_{Gauss}$ is a Gaussian random variable with the same correlation and covariance matrix as $\mathbf{y}$. A more flexible and reliable approximation of negentropy [16] is:

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

where ν is a Gaussian variable with zero mean and the same variance as y and G may be almost any non-quadratic function. While the choice of G is not critical [15], some practical functions have been suggested [17]:

- General purpose: $G_1(y) = \log\cosh(a_1 y)/a_1$
- More suited to super-Gaussian signals: $G_2(y) = \exp(-a_2\frac{y^2}{2})/a_2$
- More suited to sub-Gaussian signals: $G_3(y) = y^4/4$

One of these contrast functions is also used in the Constrained ICA algorithm to separate independent components. (The performance of this algorithm with different contrast functions is described in Chapter 3.) Since, in general, EEG signals are sub-Gaussian, the "sub-Gaussian" contrast function might seem most appropriate for the work in this project. However, a paper analysing the statistical properties of these contrast functions [18] warns that the "sub-Gaussian" contrast function is not robust to noise and outliers. The "general" contrast function was therefore used for the work described in this thesis.

The usefulness of this method of approximating negentropy is shown by the FastICA algorithm [19], which has previously been used for EEG seizure onset analysis [13]. A disadvantage of the FastICA algorithm for seizure onset analysis is the use of a square mixing matrix, which results in the

same number of extracted components as input channels. For example, if a 25 channel EEG epoch is analysed with FastICA, the assumption is implicitly made that there are 25 independent underlying sources. If this assumption does not hold, this can lead to "splitting" of actual independent sources between components. Such "splitting" poses a problem for automatic analysis of the components. The 25 components must therefore be analysed manually to find, for example, the seizure waveform. The Constrained ICA algorithm, which is discussed in the following section, offers a similar method of separating components as for FastICA (although a Newton-like learning algorithm is used rather than the FastICA fixed point algorithm). However, only one independent component is extracted, the choice of which is governed by a given reference signal.

2.2 The Constrained ICA algorithm

Constrained ICA is a recently presented variation on the negentropy maximisation method of ICA which allows *a priori* information to be incorporated into the learning algorithm [20]. Following Lu and Rajapakse [20], we give a summary of the derivation of the algorithm in this section. In the following, only one-unit ICA is considered, as for our applications one extracted source is currently sufficient. (The algorithm can be easily extended to extract multiple sources, each with a supplied reference.) As for conventional ICA, the observations $\mathbf{x}(t) = (x_1(t)...x_n(t))^T$ are assumed to be a linear mixture $\mathbf{x} = \mathbf{Ac}$ where $\mathbf{c}(t) = (c_1(t)...c_m(t))^T$ are independent components and $\mathbf{A}$ is an $N \times M$ matrix. For one-unit ICA, using a negentropy contrast function as above, we would expect the contrast function to have M optimum solutions $\mathbf{w}_i, i = 1, ..., M$, each giving an output equal to an independent source $c_i, i = 1, ..., M$.

For this algorithm a reference signal $r(t)$ is also supplied *a priori*. The closeness of an estimated output y to the reference is measured using a norm $\epsilon(y, r)$, which is minimum for the closest source. If $\mathbf{w}^{*T}$ gives the source uniquely closest to the reference, we obtain:

$$\epsilon(\mathbf{w}^{*T}\mathbf{x}, r) < \epsilon(\mathbf{w}_1^T\mathbf{x}, r) \leq ... \leq \epsilon(\mathbf{w}_{m-1}^T\mathbf{x}, r) \tag{2.6}$$

We can therefore define a threshold parameter $\xi \in [\epsilon(\mathbf{w}^{*T}\mathbf{x}, r), \epsilon(\mathbf{w}_1^T\mathbf{x}, r))$ such that $g(\mathbf{w}) = \epsilon(y, r) - \xi \leq 0$ only when $y = \mathbf{w}^{*T}\mathbf{x}$. $g(\mathbf{w})$ is used as a constraint on the one-unit contrast function $J(y)$. Another constraint, $h(\mathbf{w}) = E\{y^2\} - 1 = 0$, is included to bound $J(y)$ and $\mathbf{w}$.

The augmented Lagrangian function for this problem is:

$$\mathcal{L}_1(\mathbf{w}, \mu, \lambda) = J(y) - \frac{1}{2\gamma}[\max^2\{\mu + \gamma g(\mathbf{w}), 0\} - \mu^2] - \lambda h(\mathbf{w}) - \frac{1}{2}\gamma\|h(\mathbf{w})\|^2 \tag{2.7}$$

The Lagrange multipliers for constraints $g(\mathbf{w})$ and $h(\mathbf{w})$ are μ and λ respectively; γ is the scalar penalty parameter which influences the effect of the reference signal on the algorithm. A Euclidean norm $\|.\|$ is used and the penalty term $\frac{1}{2}\gamma\|.\|^2$ ensures that this optimisation problem is held at the condition of local convexity assumption.

A Newton-like learning method is used on $\mathbf{w}$ to maximise $\mathcal{L}_1$:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta(\mathcal{L}''_{1_{w_k^2}})^{-1}\mathcal{L}'_{1_{w_k}} \tag{2.8}$$

Here, k is the iteration index and η is a positive learning rate. $\mathcal{L}'_{1_{\mathbf{w}}}$ is the first derivative of $\mathcal{L}_1$ with respect to $\mathbf{w}$:

$$\mathcal{L}'_{1_{\mathbf{w}}} = \bar{\rho}E\{\mathbf{x}G'_y(y)\} - \frac{1}{2}\mu E\{\mathbf{x}g'_y(\mathbf{w})\} - \lambda E\{\mathbf{x}y\} \tag{2.9}$$

$G'_y(y)$ and $g'_y(\mathbf{w})$ are the first derivatives, with respect to y, of $G(y)$ and $g(\mathbf{w})$. $\bar{\rho} = \pm\rho$ such that the sign is the same as $E\{G(y)\} - E\{G(\nu)\}$.

The Hessian matrix $\mathcal{L}''_{1_{\mathbf{w}^2}}$ is approximated:

$$\mathcal{L}''_{1_{\mathbf{w}^2}} = s(\mathbf{w})\mathbf{R}_{\mathbf{xx}} \tag{2.10}$$

where $s(\mathbf{w}) = \bar{\rho}E\{G''_{y^2}(y)\} - \frac{1}{2}\mu E\{g''_{y^2}(\mathbf{w})\} - \lambda$, the covariance matrix $\mathbf{R}_{\mathbf{xx}} = E\{\mathbf{x}\mathbf{x}^T\}$ and G''_{y^2} and $g''_{y^2}(\mathbf{w})$ are the second derivatives.

The learning update is then obtained from Equation 2.8 with the approximation in Equation 2.10:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta\mathbf{R}^{-1}_{\mathbf{xx}}\mathcal{L}'_{1_{\mathbf{w}_k}}/s(\mathbf{w}_k) \tag{2.11}$$

The multipliers are updated as follows:

$$\mu_{k+1} = \max\{0, \mu_k + \gamma g(\mathbf{w}_k)\} \tag{2.12}$$

$$\lambda_{k+1} = \lambda_k + \gamma h(\mathbf{w}_k) \tag{2.13}$$

After convergence (by the weight vector difference criterion described below), the algorithm output is therefore an estimated source $y = \mathbf{w}^T\mathbf{x}$. The output will satisfy the following conditions:

- y is one of the independent components which was mixed to generate the observations, separated by a negentropy measure from the other components.
- Of the IC's, y is the source closest, by some distance measure, to r.

In calculating the demixing vector $\mathbf{w}$, the algorithm implicitly estimates the mixing vector $\mathbf{a}$. This vector gives spatial information, when EEG data is used such as for seizure onset analysis, which can be plotted as a topographic map.

The algorithm is deemed to have converged when the logarithm of the difference in 2-norm of the weight vector, between two consecutive iterations, is less than a threshold. A logarithmic measure was used in order to increase the effect of small changes in the weight vector, such as when the algorithm is close to convergence, so that the algorithm does not halt prematurely. A threshold of one was used for all work described in this thesis.

Different distance measures have been suggested for assessing the closeness of the algorithm output at each iteration to the reference signal. All the work described in this thesis used a correlation estimate $\epsilon(y_i, r_i) = -E\{y_i r_i\}$. If a different reference type is used, for example the spatial reference discussed in Chapter 7, a different closeness measure may be required.

A schematic diagram of the operation of the CICA algorithm is shown in Figure 2.1.

2.3 Whitening as a preprocessing step

Whitening (or sphering) is the process of linearly transforming a set of random variables into uncorrelated variables [16]. In ICA, whitening is a common preprocessing step because the assumed mixing matrix $\mathbf{A}$ is made orthogonal. The complexity of the problem is then reduced since the mixing

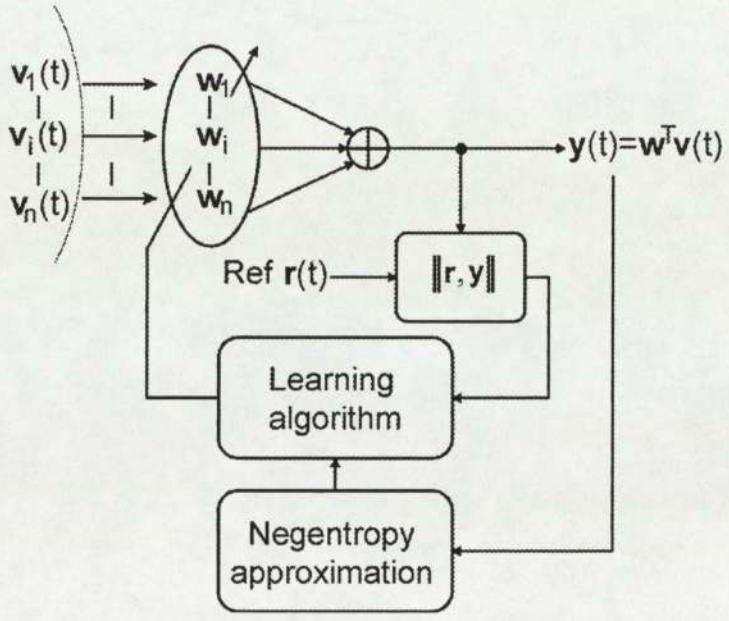


Figure 2.1: Schematic diagram of the CICA algorithm. A weight vector $\mathbf{w}$ is updated such that the output $\mathbf{y}(t) = \mathbf{w}^T\mathbf{v}(t)$ is maximally statistically independent (using the negentropy approximation) and also similar, by a correlation measure, to reference $\mathbf{r}(t)$.

matrix has fewer degrees of freedom. For the CICA algorithm, where only a mixing vector $\mathbf{a}$ is to be estimated, this reduction in parameters may be less important, but whitening was nonetheless performed using an eigenvalue decomposition. The code of our implementation of the CICA algorithm (Appendix A) shows this process, along with the dewhitening of the algorithm's output. Whitening can be specified as an optional step in the CICA function options; for all work described in this thesis the whitening step was included.

2.4 The CICA algorithm and seizure onset analysis

For seizure onset analysis as described in this thesis, the input data matrix for the CICA algorithm is the whitened EEG data. Typically such data consists of 25 channels, sampled at 200 Hz. The reference signal must be chosen to give a high correlation with the required source, but need not be a close match: square waves of similar frequency are used for much of the work in this project. Indeed, it may be undesirable to provide too close a reference if one is unsure of the exact underlying source, since the algorithm might "mix" the true underlying sources to achieve a better match to the reference. In some applications, such as seizure prediction, the EEG data may not contain any seizure waveform. The supplied reference does not then correspond to any underlying source. (This problem of "false" references is addressed in Chapters 3 and 4.)

The assumptions inherent in using the CICA algorithm seem to be reasonable when it is used for seizure onset analysis. The underlying electrical sources in the brain are mixed in a near-linear fashion to produce the signals observed at the scalp. While there is undoubtedly significant noise in the recorded data, this is likely to be treated as separate sources by the algorithm and, since only one source is to be extracted, should not adversely affect the performance. Noise in the assumed mixing matrix might be caused by interference between channels, despite electrical shielding between the conductors. However, this interference is likely to be stationary for the short EEG epochs used

(typically 2 seconds in length). Finally, the assumption that there is no more than one Gaussian underlying source present may not hold if Gaussian noise appears in the recorded data. The signals of interest are generally sub-Gaussian, so the inability of the algorithm to separate multiple Gaussian sources should not prevent the extraction of, for example, seizure waveform.

Chapter 3

Algorithm evaluation: "toy" data set

Our implementation of the CICA algorithm, which was described in Chapter 2, was tested using a simple data set before it was applied to the problem of seizure onset analysis. We look at how two of the algorithm's internal parameters (contrast function and penalty parameter) affect its behaviour. The three contrast functions recommended for this algorithm (sub-Gaussian, general and super-Gaussian) were implemented. The choice of contrast function may alter the algorithm's selection of an underlying source to extract, especially if the given reference is not close to any of the underlying sources. The penalty parameter controls the emphasis placed on the reference in the algorithm's extraction. The effect of large values of this parameter when the given reference is dissimilar to any of the underlying sources is crucial to applications in electromagnetic brain signals: does a large penalty parameter "force" the algorithm to produce an output similar to the reference, at the expense of the output's statistical independence? If the algorithm frequently mixes sources in this way to match the reference, it cannot be used to determine the presence or absence of, for example, seizure waveform in the observed data.

The CICA algorithm is initialised with a random weight vector for each trial. Useful general conclusions could only therefore be drawn from a large number of trials, giving reasonable coverage of the initial weight vector space. Manual assessment of the output of each trial was therefore impossible. Instead, a mean squared error (MSE) test was used to assess each trial automatically. The applicability of MSE for this purpose is discussed later in this chapter.

3.1 Use of a "toy" data set

A simple data set of four known sources (originally published to demonstrate the FastICA algorithm [21]) was used for these trials. The sources are generated in MATLAB as follows:

```
N = 500;
v=[0:N-1];

sig(1,:)=sin(v/2); %sinusoid
sig(2,:)=((rem(v,23)-11)/9).^5; %tan-like curve
```

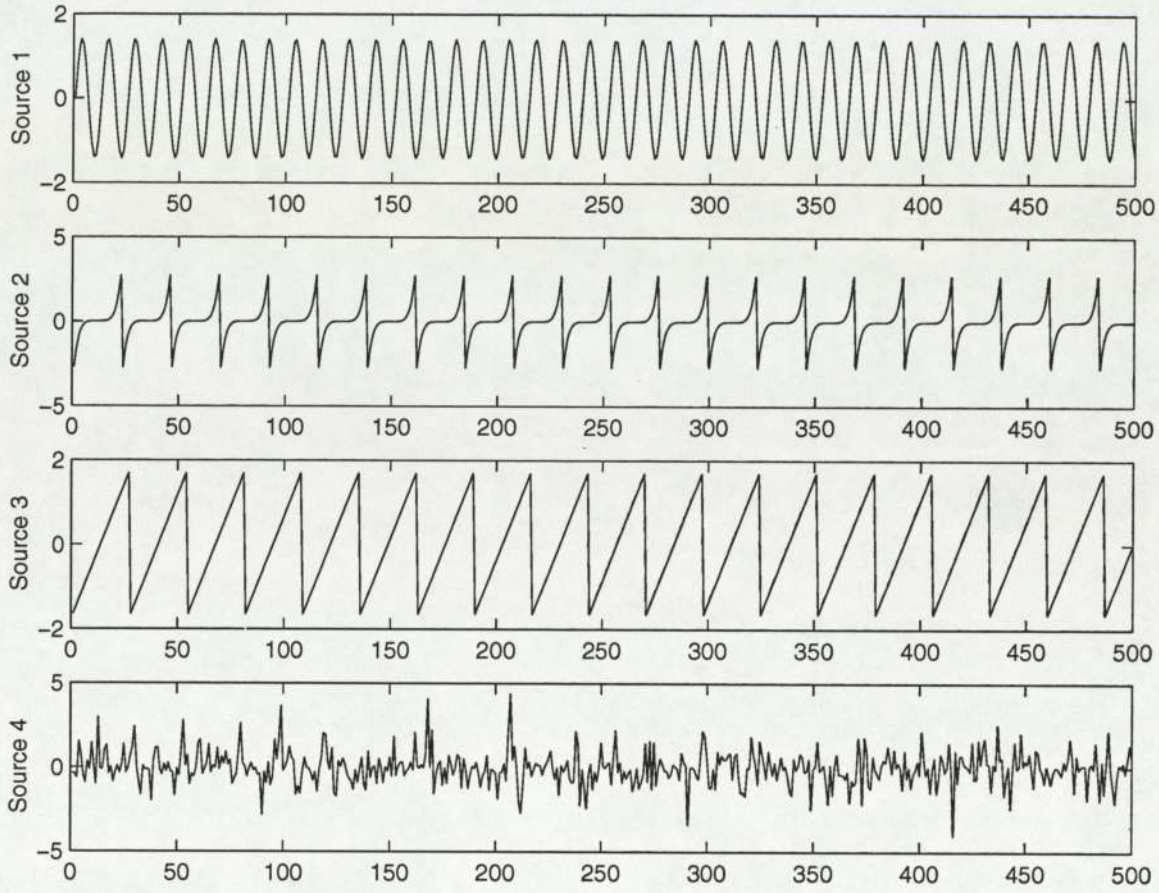


Figure 3.1: The four source signals used for the trials described in this chapter: a sine wave, a tan-like wave, a sawtooth wave and a noise source.

```
sig(3,:)=((rem(v,27)-13)/9); %saw-tooth
sig(4,:)=((rand(1,N)<.5)*2-1).*log(rand(1,N)); %noise source
```

These sources are shown in Figure 3.1. Their use for these trials seems appropriate because they have a variety of distributions (Figure 3.6), the number of sources is small for easier analysis and all the underlying sources are known. The algorithm's outputs can therefore be automatically compared exactly to the sources.

3.2 Mean squared error for algorithm output classification

Since a large number of trials were performed, an automatic assessment of each trial was necessary. Specifically, it was required that each output should be classified as being close to one of the four actual sources, or a mixture. The choice of an MSE measure for this purpose is justified below. It was not appropriate to use a correlation measure because correlation is used as a closeness measure between the output signal at each iteration and the reference signal by the CICA algorithm. When the closeness of the output signal to the desired source is to be assessed for the investigation of "false" extractions, as in the following chapter, a different closeness measure would give a better assessment. Another method of assessing each trial would be to compare the normalised dewhitened mixing vector $\mathbf{a}$ to the normalised corresponding row of the known mixing matrix A_{orig}. However, this method

would not take account of similarities between the sources. Some mixing of two similar sources is more acceptable, since the output will be close to a pure source, than mixing of two dissimilar sources, producing an output which is not close to any pure source. Assessment of only the morphology of the algorithm's output signal does not penalise the algorithm excessively for mixing similar sources and was therefore used for these trials.

Hence, for each trial the MSE between the output signal and each of the known sources was calculated. Comparison of these four values to a threshold allowed the algorithm's output to be classified as follows:

1. *All four MSE values are above the threshold:* the algorithm's output does not match any of the four sources. From the derivation of the algorithm, the output must always be some linear mixture of the observation time series. Hence if the output does not match any of the true sources, it must be some mixture of sources.

2. *One of the four MSE values is below the threshold:* the algorithm has selected a single source, with sufficient accuracy to satisfy the MSE test.

3. *Multiple MSE values are below the threshold:* the algorithm's output is not a close match to a single source, but may be a mixture of sources. In practice this situation never occurred because the true sources were very different. An output signal could not therefore simultaneously match more than one of them, for the thresholds used.

The discrimination of MSE for this application was assessed using Receiver Operating Characteristic (ROC) curves. For each of the four sources shown in Figure 3.1 a reference signal was generated by taking the sign of the signal (Figure 3.2). The CICA algorithm was executed 5000 times using each reference, with the random mixture of the sources (Figure 3.3) as input data. (For these tests the penalty parameter was fixed at one and the "general" contrast function was used. Additionally, the input data was whitened.) After each trial the MSE between the algorithm's output and each source was calculated and stored in a (4 x 20000) results matrix. Thresholds of MSE ranging from 0 to 3 in increments of 0.01 were applied to the matrix in turn. The results of the thresholding were classified as follows:

1. A **true positive** occurred if the MSE between the output and the 'correct' source (from which the reference was derived) was less than the MSE threshold.

2. A **false positive** condition was defined as for a true positive, but with one or more other sources also giving an MSE less than the threshold.

Each value of MSE threshold between 0 and 3 was used as a 'cut point' on the ROC curves in Figure 3.4. The numbers of true positives and false positives were counted and normalised. The curves demonstrate that MSE is a reasonably accurate test for this application, since the area under the curves is large. MSE measures were therefore used for comparing CICA algorithm outputs to the known input sources. An MSE threshold of 0.5 was chosen for later work because this value gives as many true positives as possible without introducing false positives, for all four ROC curves.

Figure 3.2: Reference signals used for tests in this chapter, obtained by taking the sign of the corresponding source signal.

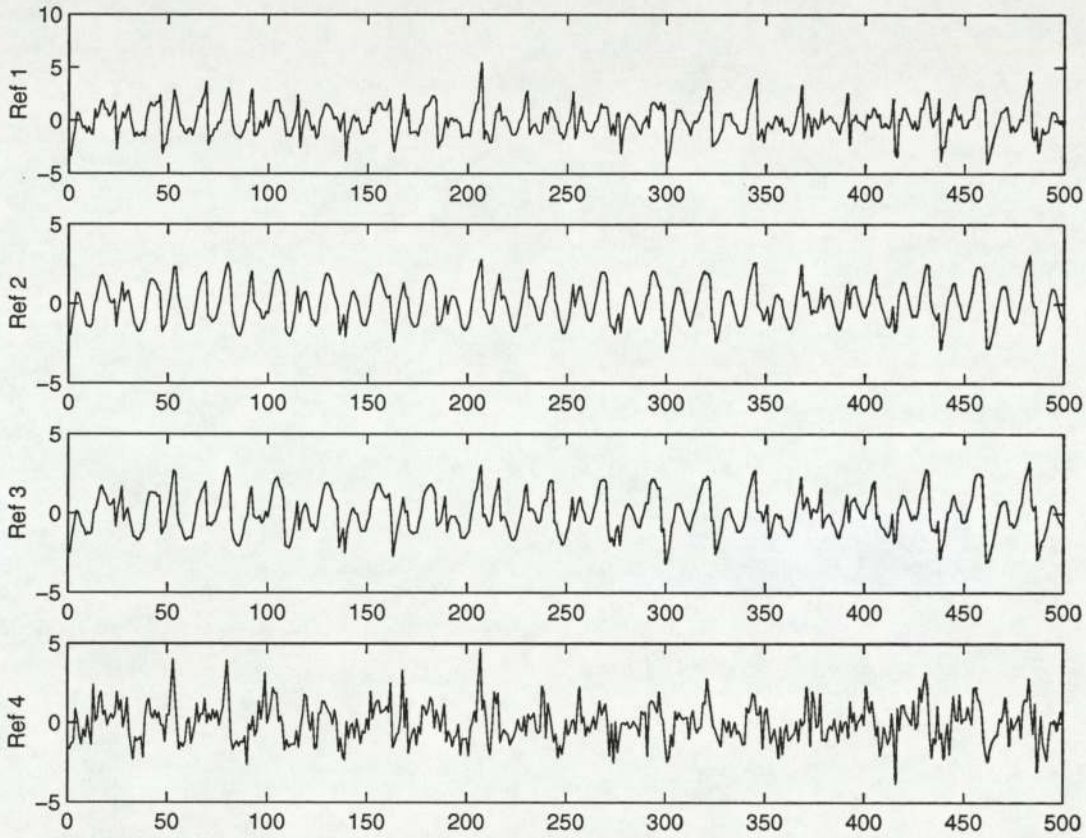


Figure 3.3: The input data to the algorithm for trials described in this chapter, generated from the sources shown in Figure 3.1 with a random mixing matrix.

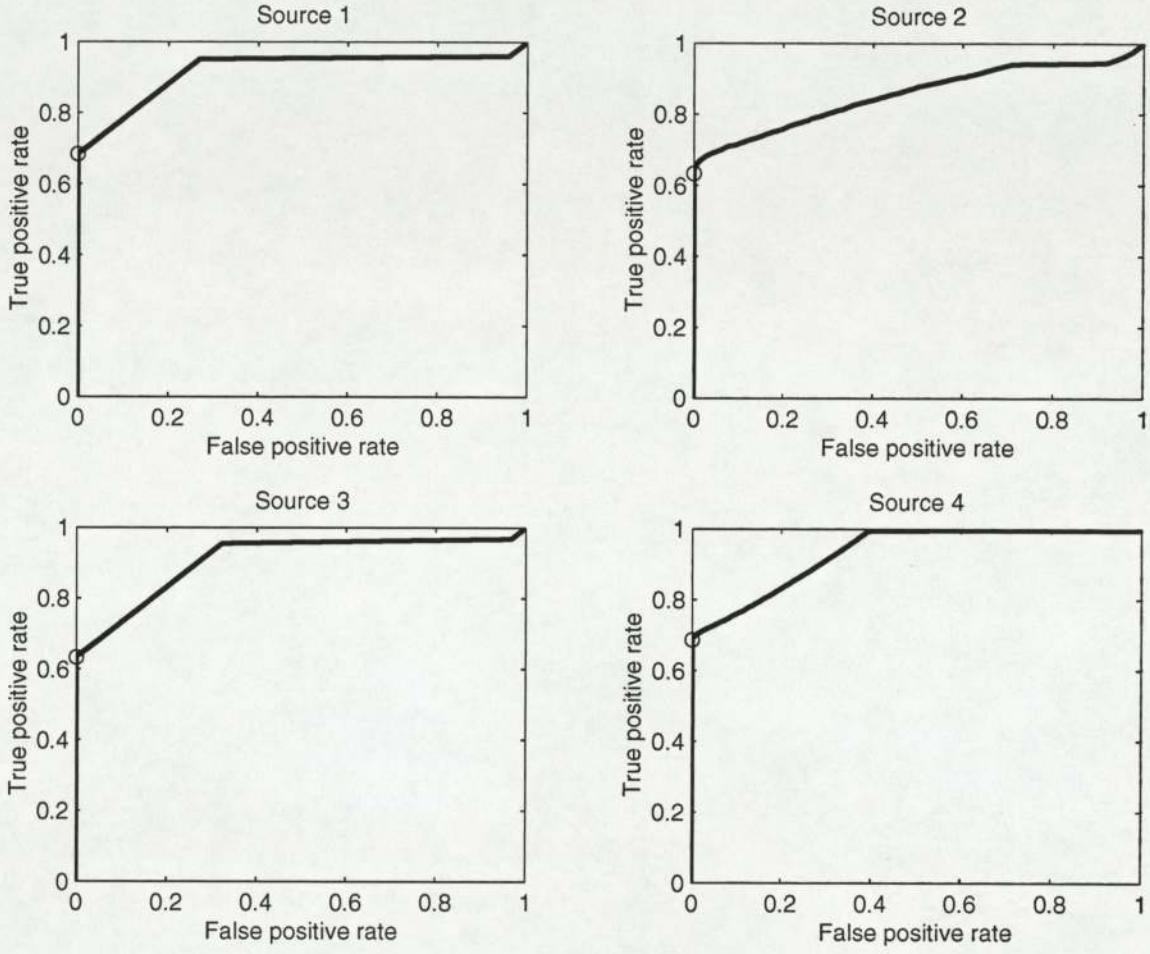


Figure 3.4: ROC curves demonstrating the validity of an MSE difference measure for assessing the accuracy of the algorithm output. For each source, 5000 trials were performed using the sign function of the source as a reference. The MSE between the output signal and each of the four sources was calculated. An MSE threshold was varied between 0 and 3 to obtain cut points for these curves. The large area under the curves indicates that MSE is a good (although not perfect) measure for assessing the algorithm's output against known sources. A threshold of 0.5 (shown as a circle on each graph) was chosen for later work with this "toy" data set since it gives the largest true positive rate without introducing false positives for all four ROC curves.

3.3 Effect of contrast function on source selection

Three negentropy approximation contrast functions were coded into the CICA algorithm, as discussed in Chapter 2. Apart from the 'general' function, the functions were claimed to exert a bias towards selection of a super-Gaussian or sub-Gaussian source [20]. The effect of changing between these functions was tested by setting the penalty parameter to a low value (0.00001)[1], which causes the reference signal to have almost no effect on the algorithm's behaviour. For each of the four test sources, 5000 trials were performed with randomly-selected starting weight vectors. This procedure was repeated for the "general", "sub-Gaussian" and "super-Gaussian" contrast functions. The results of these trials were classified with the MSE threshold described above, and are shown in Figure 3.5.

The sub-Gaussian contrast function (a kurtosis measure) is not recommended by its proposers [18] except in circumstances where the independent components are sub-Gaussian and there is no noise present. In fact it appeared to perform well in these tests, giving a fairly uniform selection of sources and few results where the algorithm failed to extract any source. There is a slight bias towards the extraction of source 1, which is highly sub-Gaussian (Figure 3.6).

The general contrast function also seemed to perform well, although the algorithm failed to converge to a single source more often than with the sub-Gaussian contrast function.

The super-Gaussian contrast function gave relatively poor results, with a large number of trials where the algorithm failed to converge to a unique source. It might be expected that extractions with this contrast function would show some bias towards source 2, which has a super-Gaussian distribution (Figure 3.6). These results, however, show a bias against this source. The large number of extractions of mixtures may have distorted the results, making conclusions about the bias of this contrast function harder to draw. In any case, regardless of its poor performance for extracting unique sources, a super-Gaussian contrast function would not be appropriate for the other work described in this thesis since EEG signals are generally considered to be sub-Gaussian.

In conclusion, since EEG signals are generally sub-Gaussian it might seem appropriate to use the sub-Gaussian contrast function for this project. However, since other work has highlighted the poor performance of this kurtosis approximation, especially in the presence of noise, the general contrast function was used instead for all other work described in this thesis. The results shown in Figure 3.5 show that this decision did not cause great loss of accuracy.

3.4 Effect of penalty parameter

The effect of varying the penalty parameter (that is, the influence of the reference signal on the algorithm's behaviour) was studied using the same sources and references as used previously. For each of the four references, 5000 trials were performed for each penalty parameter between 0 and 5 in increments of 0.5. Figure 3.7 shows that as the penalty parameter increases the accuracy of the algorithm in selecting the correct source increases. Penalty parameter values larger than approximately one have little further effect: the normalising effect of scaling factors in the algorithm appears tc absorb increases above one.

The behaviour of the algorithm given a reference which did not correspond to any true source was of interest. Although the algorithm is intended to extract a single source, it was thought that,

[1]It was not possible to set the penalty parameter to zero for these trials because, for the super-Gaussian contrast function, this caused a calculation error.

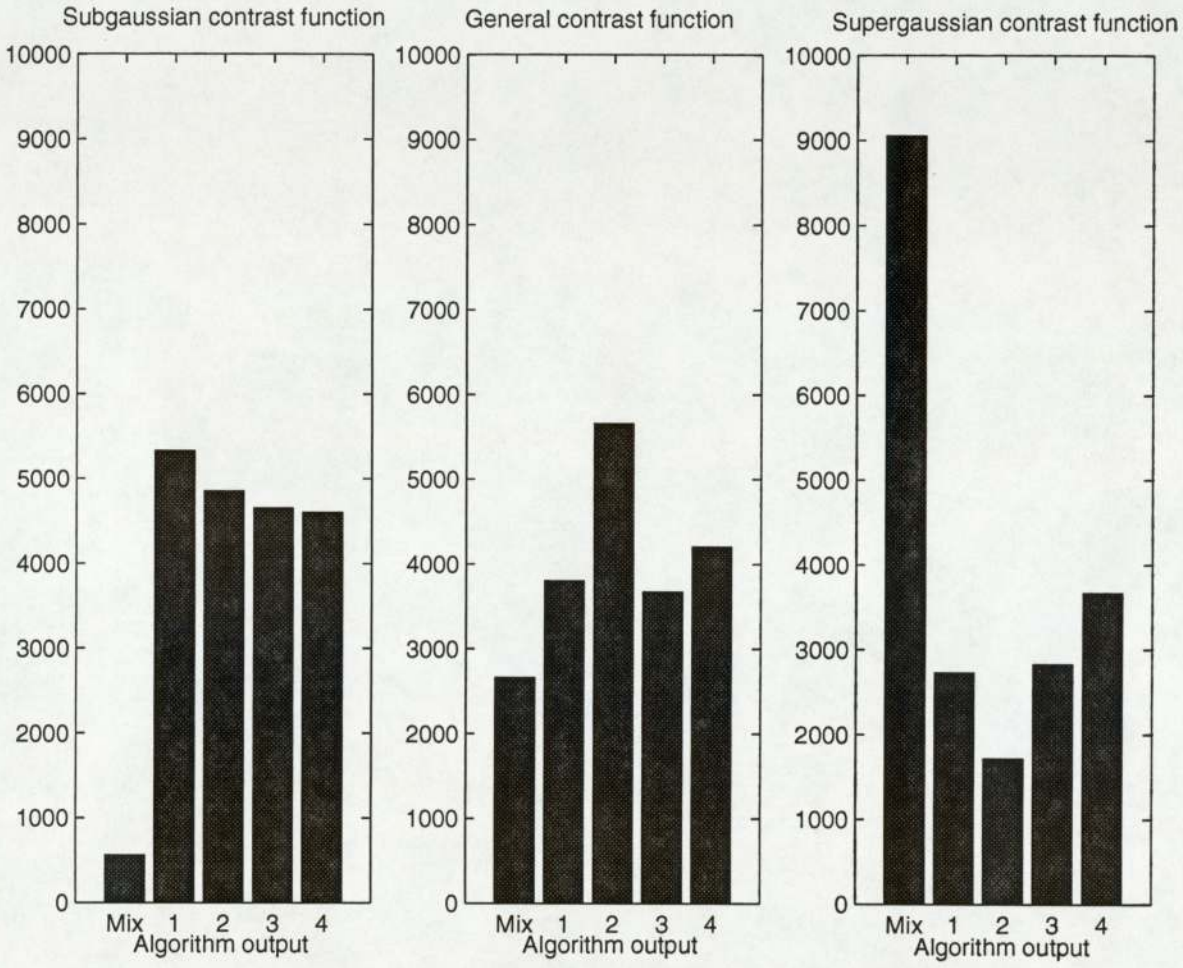


Figure 3.5: Sources selected by the CICA algorithm with different contrast functions. The influence of the reference signal was minimised by setting the penalty parameter to a small value (0.00001). The algorithm's output was compared to each of the four sources with an MSE measure. If all four MSE values were above the threshold of 0.5, the output was assumed to be match no source uniquely. These outputs are labelled "Mix".

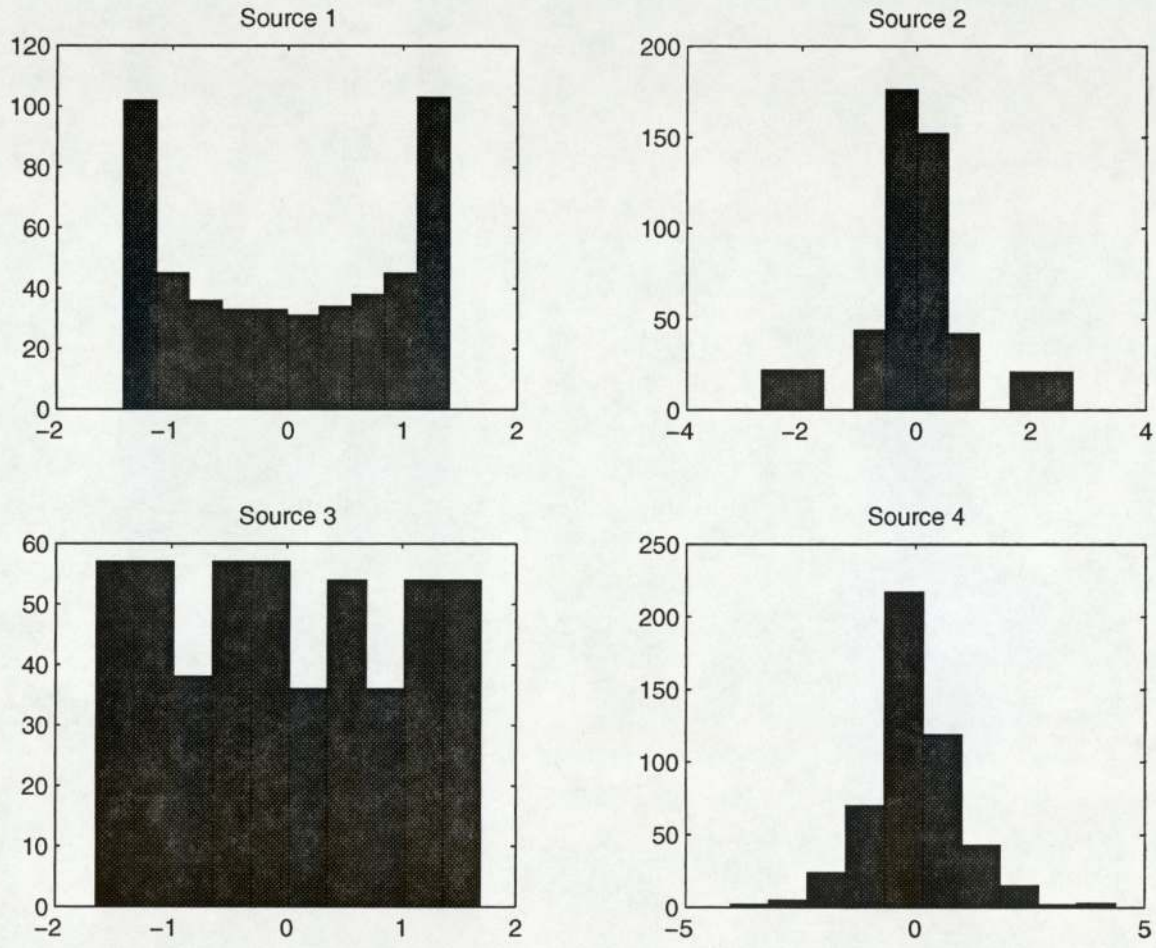


Figure 3.6: Distributions of the four sources shown in Figure 3.1

Actual source	Correlation with "false" reference
1	-0.40
2	1.7
3	5.36
4	-0.50

Table 3.1: Correlation of each of the four actual sources with the "false" reference (Figure 3.8). Although all the correlations are relatively low, since this reference was chosen to be unlike any of the actual sources, the highest correlations are for sources 2 and 3. This is reflected in the bias shown towards these two sources in trials with this reference (Figure 3.9).

with a large penalty parameter, the algorithm might be "forced" to generate an output closer to the reference by mixing sources. To test this, a "false" reference (Figure 3.8) was used in trials similar to those described above: 5000 trials were performed for each value of penalty parameter between 0 and 5 in increments of 0.5. This "false" reference has a morphology unlike any of the four actual sources. The results of trials with this "false" reference, which were classified using MSE as matches to the four true sources, are shown in Figure 3.9. With a penalty parameter of zero, the "false" reference has no effect on the algorithm, which therefore extracts one of the true sources. The choice of extracted source is then dependent on the starting weight vector and the positions of negentropy maxima. For this data set, source 2 seems to be particularly favoured. As the penalty parameter is increased to one, the "false" reference influences the algorithm's behaviour. The strong bias towards sources 2 and 3 is likely to be because the "false" reference has the largest correlation with these sources, as shown in Table 3.1. Figure 3.10 shows that the number of trials for which a mixture of sources was extracted, given the "false" reference, was reasonably uniform for different values of penalty parameter. For all the graphs shown, values of penalty parameter above one have little additional effect since they are absorbed into scaling factors in the algorithm.

These results suggest that larger values of penalty parameter do not "force" the algorithm to extract a mixture of sources to fit the reference, where no underlying source corresponding to the reference exists. This is an important result for the application of this algorithm to seizure onset analysis, where the "false" extraction of seizure-like waveform must be avoided.

3.5 Effect of increasing penalty parameter during the first iterations

The previous section describes the effect of fixing the penalty parameter at different values which were kept constant for each run of the algorithm. It was thought that the algorithm's performance might be improved by increasing the penalty parameter from zero to its final value during the first iterations - this might cause the algorithm to first seek statistical independence, before "homing in" on a source closest to the reference. This section presents some experiments to verify this suggestion. The implementation of this increase in penalty parameter has two effects:

- The penalty parameter increases over the first iterations of the algorithm.
- A minimum number of iterations is imposed on the algorithm.

These two effects were studied separately by running 5000 trials, using each of the references in turn. For the first set of trials, no minimum number of iterations was imposed. The algorithm was

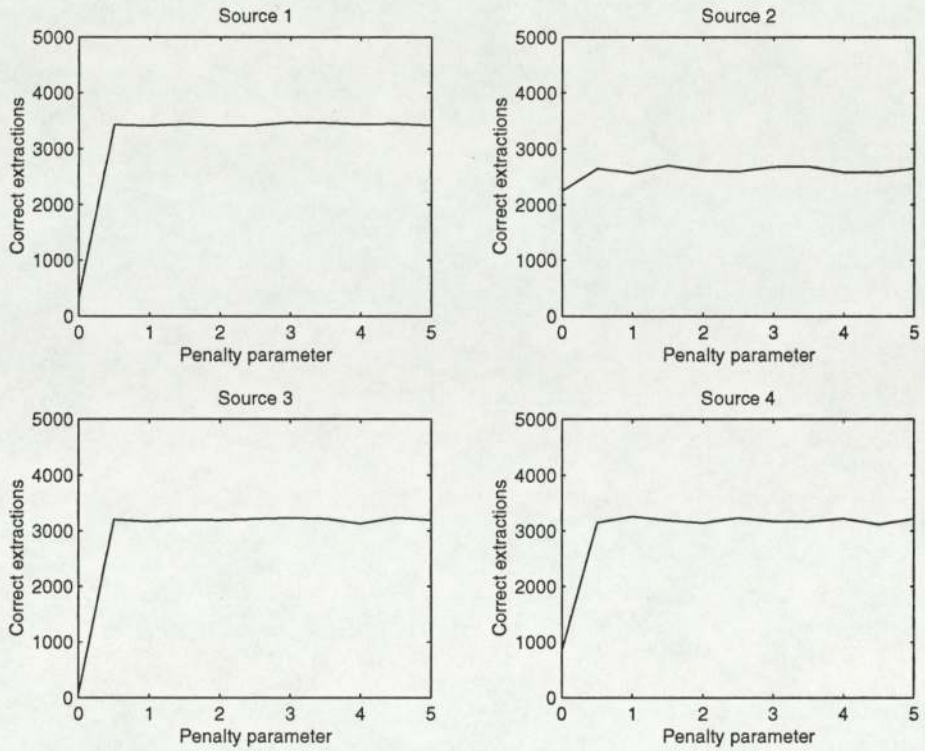


Figure 3.7: Graphs showing the effect of varying penalty parameter. The accuracy of the algorithm improves as the penalty parameter is increased up to one. Values greater than one appear to be absorbed into scaling factors in the algorithm.

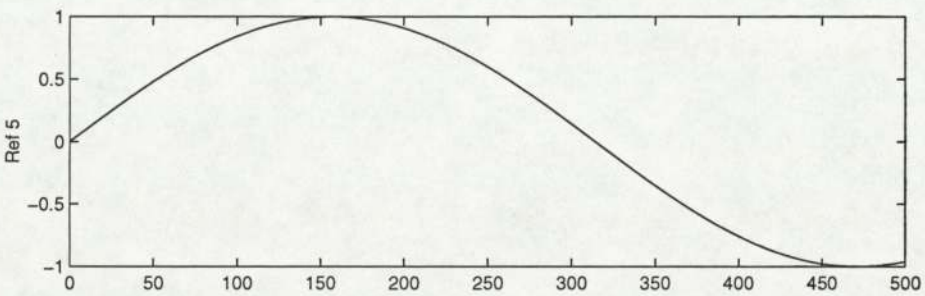


Figure 3.8: This reference does not correspond to any of the true sources (Figure 3.1). It was used to determine whether the algorithm would estimate a non-existent source to match it, given a large penalty parameter.

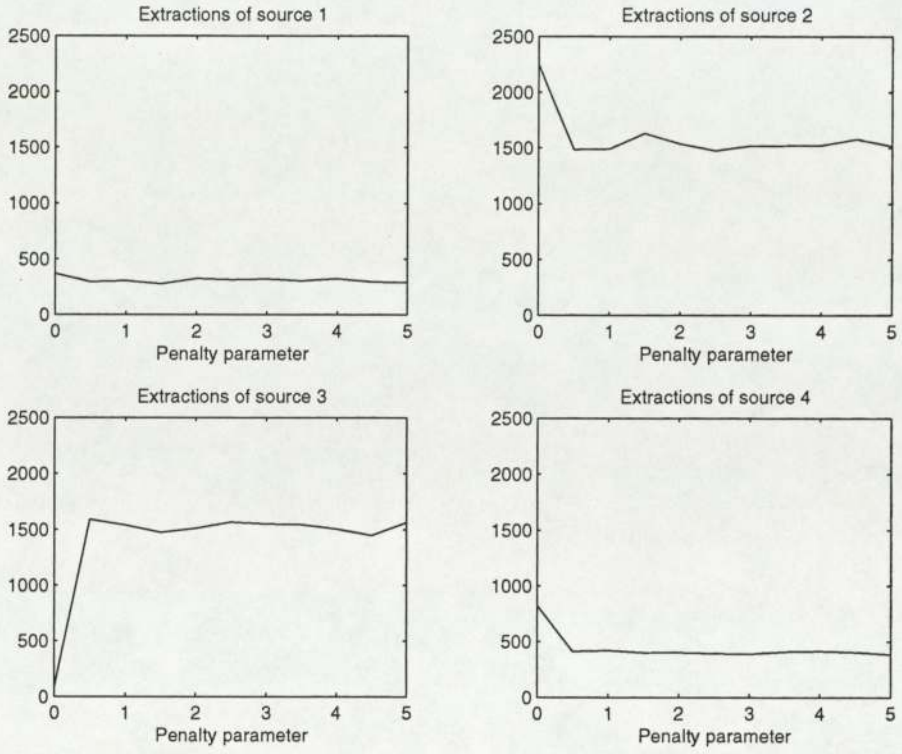


Figure 3.9: Extractions of each true source, given the "false"reference, for increasing penalty parameter.

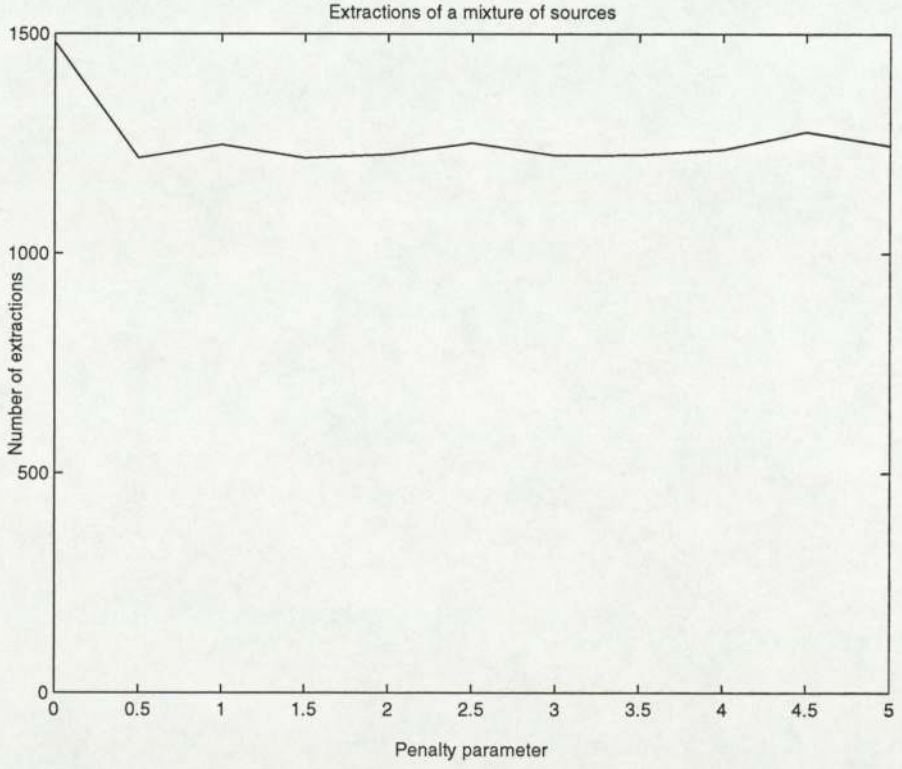


Figure 3.10: Number of trials for which a mixture was extracted (according to the MSE test)

simply terminated when a minimum change in weight vector was achieved. The second set of trials required a minimum of 10 iterations for each trial; the algorithm then continued to run until the minimum change in weight vector was detected. The penalty parameter for these trials was kept constant at 0.5. The third set of trials was the same as the second set, except that the threshold was linearly increased over the first 10 iterations from 0 to 0.5. The outputs of the algorithm were classified by MSE against each of the known sources, as described previously.

The results of these trials are shown in Figures 3.11, 3.12 and 3.13. With no minimum number of iterations, the algorithm always converged (according to the weight vector difference stopping criterion described in Chapter 2) within 3 iterations. These results are very poor compared to the other trials, suggesting that the algorithm becomes "stuck" in local minima within a few iterations, regardless of the reference given, if it is not forced to continue for at least 10 iterations. It may be that the stopping criterion used here (that the log of the norm of the difference between consecutive weight vectors should be less than 1) is not optimum for this data set. However, it may be difficult to adjust this criterion for "real-world" data where the underlying sources are not known and hence the algorithm's performance is more difficult to quantify. Imposing a minimum number of iterations improves results for this data set (Figure 3.12), even if the penalty parameter is held constant. Such an imposition may offer a means of avoiding the necessity of "tuning" the weight vector stopping criterion.

The additional improvement gained by varying the penalty parameter is negligible for these trials (Figure 3.13) once a minimum number of iterations has been imposed. For trials where one of the four "true" references was supplied, the differences in the algorithm's accuracy are all less than 3%. For trials where the "false" reference was given, use of the varying threshold made the algorithm's performance appear somewhat worse, in that the number of "false" extractions of source 3 increased by 18%. However, the other "false" extraction classifications show a reasonably uniform reduction which compensates for this; the number of "mixture" outputs has not increased. A varying threshold therefore had a reasonably small effect, which is likely to be specific to this data set. A more important improvement can be gained by imposing a minimum number of iterations before convergence.

It was thought that the reverse situation - a decrease of the penalty parameter from 0.5 to 0 over the first ten iterations - might have a different effect on performance. The algorithm might be expected to start with outputs close to the reference and move to a close statistically independent source once the penalty parameter reaches zero. In fact, as shown in Figure 3.14, the results are very similar to the other trials.

From these trials it can be concluded that the penalty parameter does not have a large effect on algorithm performance with this data set. Of greater importance is the imposition of a minimum number of iterations before the convergence test is applied. The increasing penalty parameter gave good results, perhaps because it allowed the threshold parameter to increase as recommended by the CICA algorithm authors [20]. All other trials described in this thesis therefore used an increasing threshold between 0 and 0.5, applied over the first 10 iterations, during which the algorithm could not terminate.

3.6 General performance of the algorithm

The algorithm's overall performance on this data set, using the contrast function and increasing penalty parameter previously determined, was evaluated using a larger number (500,000) of trials.

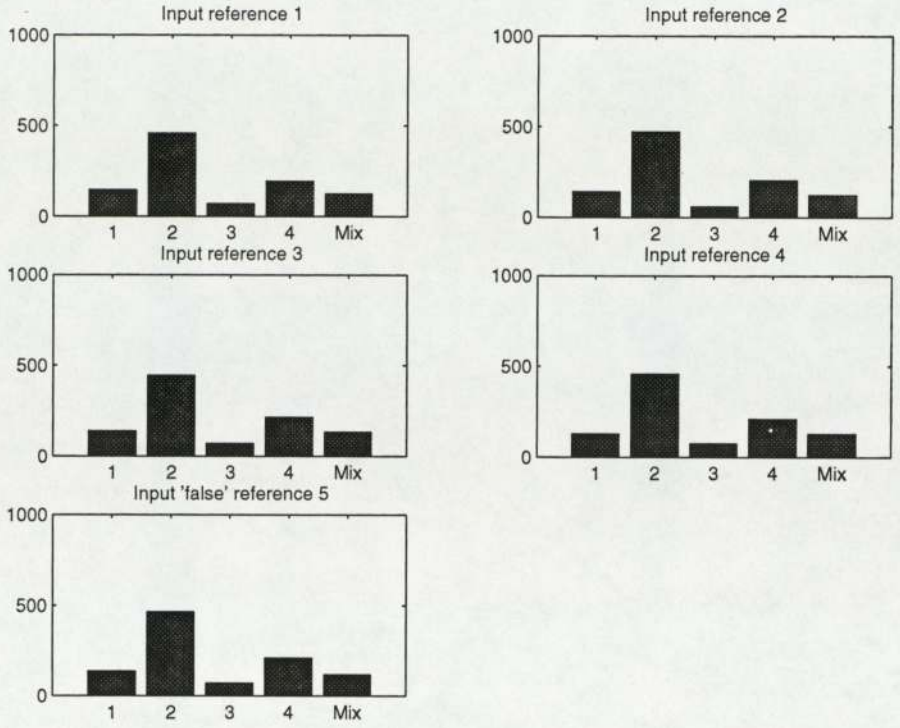


Figure 3.11: Results of 5000 trials with penalty parameter constant at 0.5. No minimum number of iterations was imposed, and the actual number of iterations taken to converge was between 1 and 3.

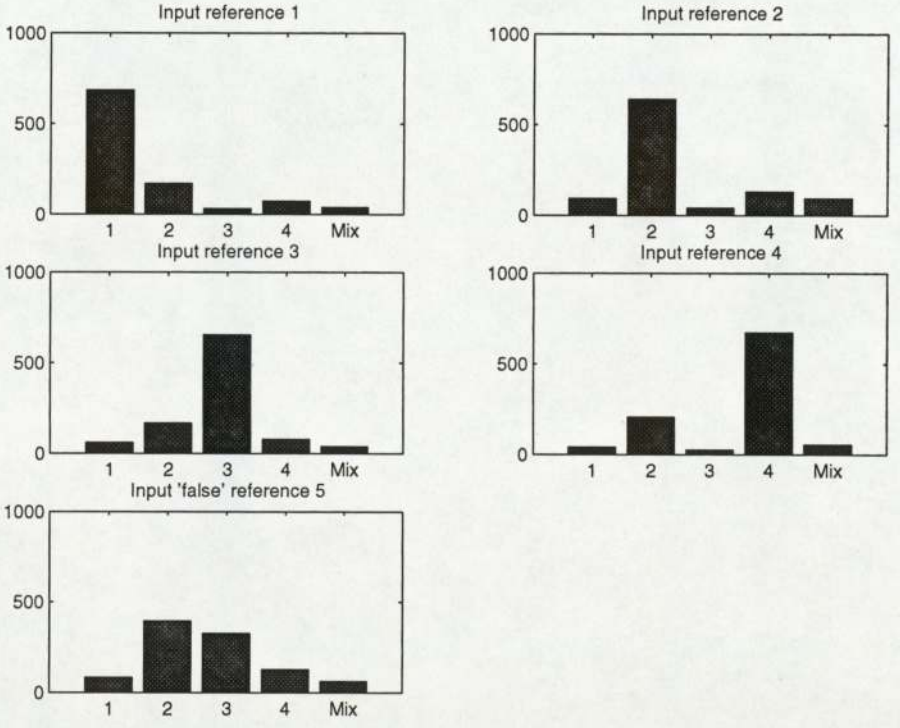


Figure 3.12: Results of 5000 trials with penalty parameter constant at 0.5, but with a minimum of 11 iterations imposed before the convergence check terminates the algorithm. In fact the number of iterations taken to converge was between 11 and 13.

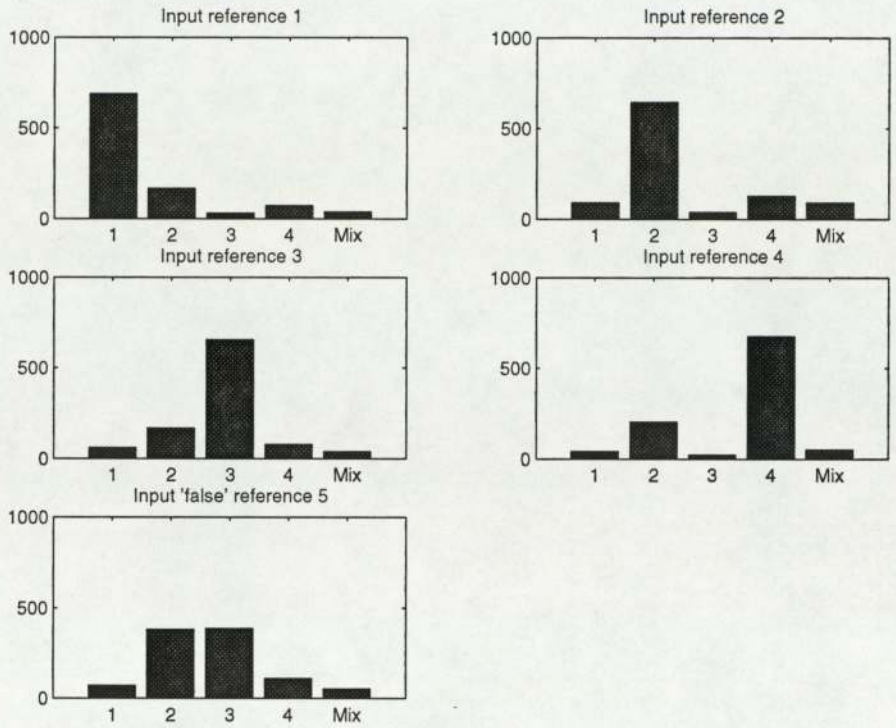


Figure 3.13: Results of 5000 trials with penalty parameter increased from 0 to 0.5 over the first ten iterations, after which termination was determined by the weight vector convergence check. The actual number of iterations taken to converge was between 11 and 13.

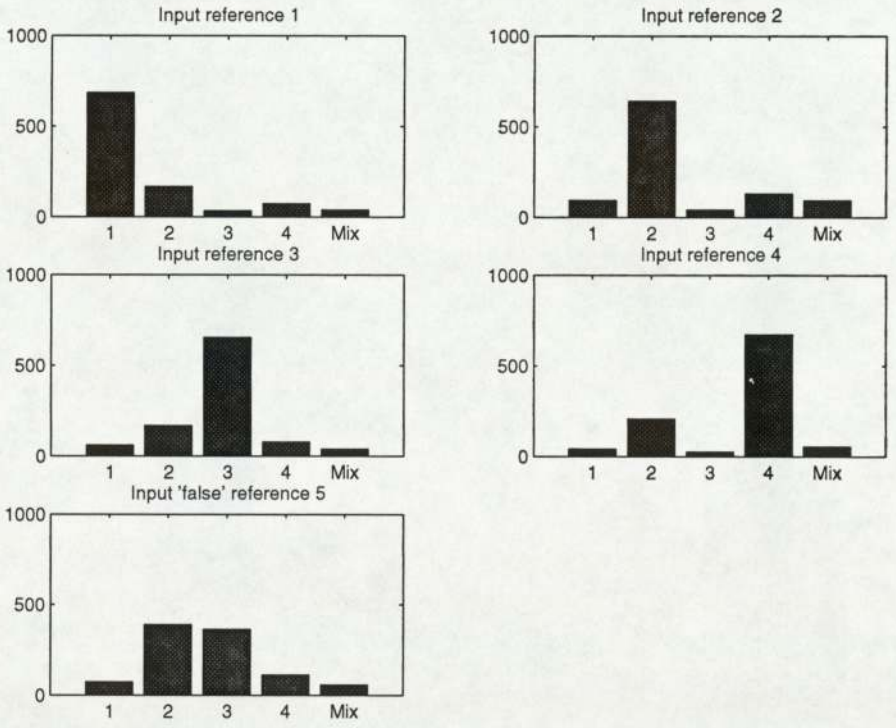


Figure 3.14: Results of 5000 trials with penalty parameter decreased from 0.5 to 0 over the first ten iterations, after which termination was determined by the weight vector convergence check. The actual number of iterations taken to converge was between 11 and 16.

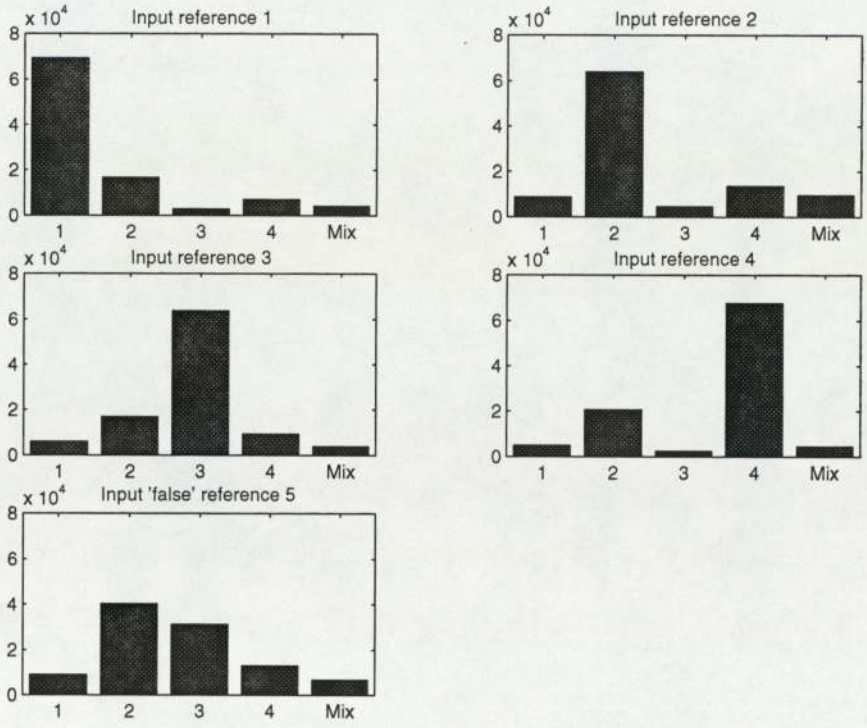


Figure 3.15: Results of 500,000 trials on the test data set, classified by mean squared error. References 1 to 4, which are similar to actual underlying sources, cause the algorithm to extract the corresponding source with reasonable accuracy. When a "false" reference was given, which was unlike any of the underlying sources, the algorithm still most often extracted pure sources. The extraction of mixtures of sources, labelled as "Mix", occurred relatively infrequently.

Figure 3.15 shows the results, classified using the MSE test described above. In the trials where one of the four "true" references was given, the algorithm extracted the corresponding source with reasonable accuracy (above 65% in all cases). Furthermore, for the application to seizure onset analysis it was encouraging that, when the "false" reference was given, the algorithm still generally extracted one of the pure sources, even though it was unlike the reference signal. In seizure onset analysis, it is undesirable for the algorithm to extract a mixture of sources which matches the reference, since this could falsely indicate the presence of seizure waveform in the EEG. However, there was still a significant number of "Mixture" outputs which could cause problems in applications, such as seizure onset analysis, where the true underlying sources are not known. The work described in the following sections attempted to find indicators of the extraction of such a "mixture".

3.7 Indicators of false extractions

3.7.1 Number of iterations

When running the CICA algorithm it might be expected that when the algorithm produces a false extraction (that is, an output which is a mixture of underlying sources rather than a pure source) it takes on average more iterations to converge. Since the true underlying sources were known for this data set, it was possible to test the use of the number of iterations taken to converge as an indicator of false extraction using a chi-squared independence test [22].

We state as a null hypothesis

H_0 : *The number of iterations is not associated with the occurrence of false extractions*

and as an alternative hypothesis:

H_a : *These two variables are associated.*

Each trial output was classified as a pure source (if the MSE of the output compared to any of the sources was less than 0.5) or a mixture (if every MSE was greater than or equal to 0.5). The accuracy of the algorithm in selecting the source which matches the reference was not considered; the independence of the output was the only concern. (In no instance did the output "match" more than one source by the MSE criterion.) This is the same method of classifying results as was used for the other work in this chapter.

The number of iterations taken to converge on every output varied between 11 and 15 iterations for these trials. (The threshold was linearly increased over the first ten iterations, as previously described, and so a minimum of ten iterations was imposed.) For each trial the number of iterations was compared to a threshold of 12 iterations, producing the following observed data contingency table:

	Iterations $<$ threshold	Iterations $\geq$ threshold	Total
Extracted pure source	471012	586	471598
Extracted mixture of sources	28402	0	28402
Total	499414	586	500000

The expected frequencies were then calculated as follows:

	Iterations $<$ threshold	Iterations $\geq$ threshold
Extracted pure source	471050	550
Extracted mixture of sources	28370	30

The chi-squared test requires a sufficiently large sample size; it is normally required that all expected frequencies must be greater than or equal to one and at most 20% of these frequencies must be less than 5. These assumptions are satisfied by the table above. The test statistic is calculated as $\chi^2 = 35.3$. This system has one degree of freedom, so that if the test is to be performed at the 5% significance level, the critical value is $\chi^2_{0.05} = 3.841$. Since the test statistic is larger than the critical value, we reject the null hypothesis, implying that the two variables are associated for this particular threshold.

The above chi-squared test was repeated for values of threshold between 11 and 16, encompassing all the numbers of iterations which occurred for these trials. The results, shown in Table 3.2, show that thresholds of 12 or 13 iterations appear to give useful tests. An insight into why only these two thresholds are useful can be gained from a table of the frequencies with which these numbers of iterations occur (Table 3.3). All the extractions of mixtures in this trial occurred for 11 iterations. Extractions of pure sources involved a wider range of iterations, from 11 to 15. This is contrary to the behaviour one might expect; it was at first thought that extractions of mixtures would involve a larger number of iterations than extractions of pure sources, since the algorithm would tend to "hunt" for a solution for longer. It may be that the algorithm can become "stuck" in a local minimum with a small number of iterations. This effect could be specific to this data set.

The relatively small numbers of extractions involving other than 11 iterations suggest that the number of iterations may not be very useful as a practical indicator of extraction of mixtures. In other words, although the number of iterations is associated with the incidence of such "false" extractions (as the chi-squared test showed), the false positive rate would be very large. Figure 3.16 demonstrates

Threshold of number of iterations	χ^2	Test result
11	Not a valid test - no trials below threshold	
12	35.3	Association
13	15.9	Association
14	2.23	No association
15	0.241	No association
16	Not a valid test - no trials above threshold	

Table 3.2: Results of chi-squared independence tests on association between number of iterations taken to converge (classified by different thresholds) and false extractions. The test result is determined in each case by comparing the χ^2 value to $\chi^2_{0.05} = 3.841$.

Number of iterations taken to converge	11	12	13	14	15
Extractions of a pure source	471012	322	227	33	4
Extractions of a mixture	28402	0	0	0	0

Table 3.3: Frequencies of extraction of pure sources and mixtures for different numbers of iterations taken to converge. All extractions of a mixture occurred for 11 iterations, while extractions of pure sources take a wider range of iterations.

this problem with an ROC curve. Although thresholds of 12 and 13 iterations give a true positive rate of 1, the false positive rates at these cutpoints are both 0.999. The number of iterations therefore appears to be a poor indicator of "false" extractions, based on the trials with this synthetic data set.

3.7.2 Dewhitened output signal power

It might be expected that the power of the dewhitened output of the algorithm may be lower for trials where significant mixing of sources has occurred. A chi-squared test was used to investigate a possible association between output power and uniqueness, as follows.

Our null hypothesis is:

H_0 :*Dewhitened output signal power is not associated with the occurrence of false extractions.*

and the alternative hypothesis is:

H_a :*The 2 variables are associated.*

Each trial output was classified as a pure source or a mixture as stated above. The signal power (calculated as the variance) of each output was then compared to a power threshold, chosen to be 0.05 for the example test shown here. This gave the following contingency table of observed data:

	Output power $<$ threshold	Output power $\geq$ threshold	Total
Extracted pure source	367914	32134	400048
Extracted mixture of sources	0	99952	99952
Total	367914	132086	500000

From this table, the expected frequencies were as follows:

	Output power $<$ threshold	Output power $\geq$ threshold
Extracted pure source	294370	105680
Extracted mixture of sources	73550	26400

These expected frequencies clearly satisfy the assumptions made for the chi-squared test: all expected frequencies are greater than or equal to one, and at most 20% of the expected frequencies are less than 5. The test statistic is calculated as $\chi^2 = 347967$. As for the previous section, this

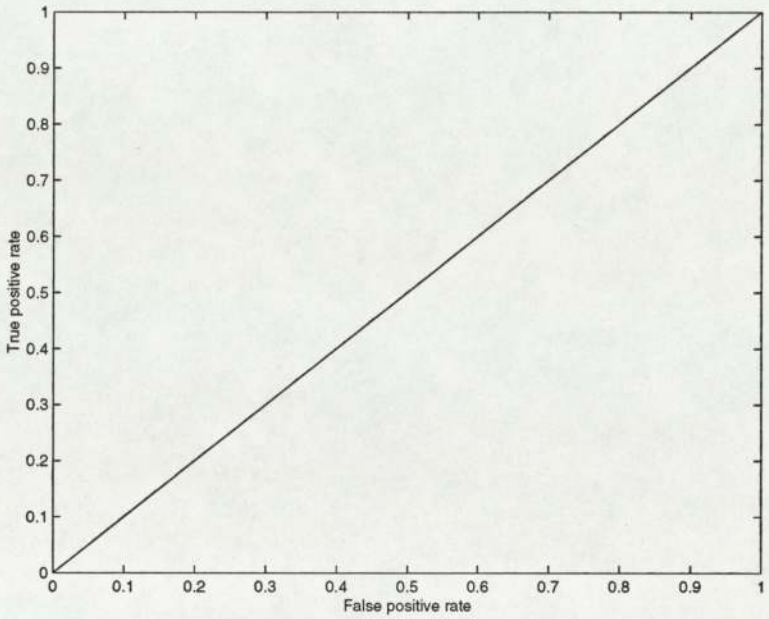


Figure 3.16: ROC curve with cutpoints for different values of threshold for number of iterations. There are only two "useful" cutpoints, at 12 and 13 iterations. The area under the curve is very close to 0.5, showing that the discrimination of this test is very poor.

system has one degree of freedom, so that if the test is to be performed at the 5% significance level, the critical value is $\chi^2_{0.05} = 3.841$. Since the test statistic is much larger than the critical value, we reject the null hypothesis, implying that the two variables are associated. The same test was repeated for other power thresholds and a range of thresholds from 10^{-23} to 10 also indicated association.

Since the power of the dewhitened algorithm output appeared to be useful as an indicator of the extraction of mixtures, an optimum value of power threshold for future use was determined using an ROC curve (Figure 3.17). This curve was generated using a range of power thresholds from 0 to 50, encompassing all the output signal powers for these trials, which were between 3×10^{-25} and 47. The area under the curve is significantly greater than 0.5, which further implies that this is a useful test. In choosing the optimum power threshold, and hence the cutpoint on the ROC curve, a tradeoff must be made between the number of true positives and false positives. In general, the consequences of a false negative are more serious than for a false positive. (For example, if the algorithm is used to extract seizure waveform from EEG given a reference resembling seizure, it is highly undesirable that the algorithm generates an output that resembles seizure by mixing underlying sources, when no actual seizure source exists. However, the consequences of a failure to extract a seizure source which is present are potentially more serious.) The chosen threshold of 10^{-20} is indicated in Figure 3.17. With this threshold the test has a true positive rate (sensitivity) of 0.986 and a false positive rate (1 - specificity) of 0.389. With this test we therefore expect almost complete identification of false outputs, but with a large number of false positives. Other cutpoints might be preferable, depending on the application of the CICA algorithm.

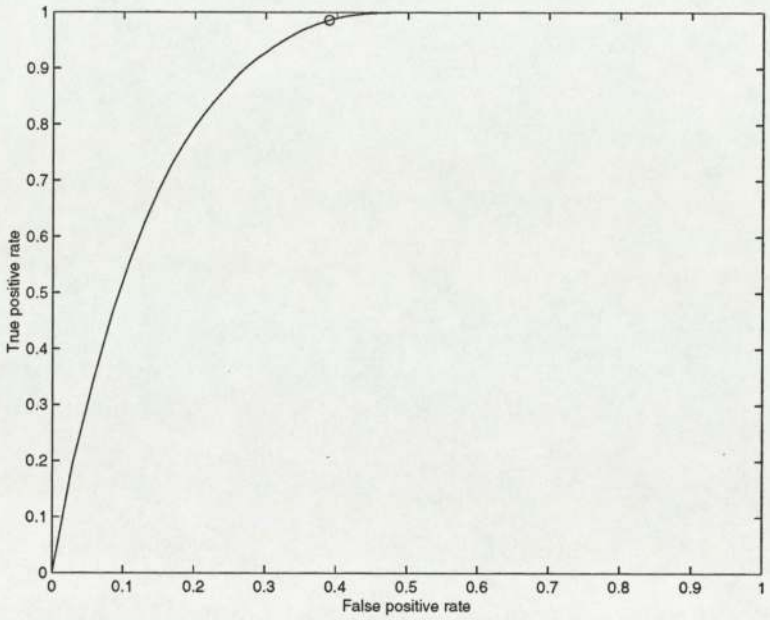


Figure 3.17: ROC curve for output power as a performance measure. The chosen optimum threshold of 10^{-20} is marked with a circle. This threshold gives a true positive rate of nearly one but at the expense of a false positive rate of nearly 0.4.

3.8 Summary

The results in this chapter provided some information on the CICA algorithm's performance, albeit based on a relatively simple example data set. This data set was used because the four underlying sources were known, so that the algorithm's performance could be automatically assessed. In general, the algorithm performs well at selecting the independent component which is closest (by correlation in this case) to the given reference. Other work described in this thesis used the "general" contrast function, which has been shown to have reasonable performance. The idea of increasing the penalty parameter over the first ten iterations, during which the algorithm cannot terminate, was adopted for all other experiments. The problem of "false" extractions, where the algorithm's output is a mixture of underlying sources, has been addressed in a preliminary fashion by a measure based on the power of the dewhitened output signal.

The next chapter describes trials which are closer to the real-world application of the CICA algorithm to seizure onset analysis, while retaining knowledge of the true source which should be extracted. The idea of output power as a "false" extraction indicator, which is likely to be very important for real-world EEG analysis, is further pursued.

Chapter 4

Extraction of simulated seizure from real background EEG

In the previous chapter, our implementation of the CICA algorithm was tested in several respects and two possible indicators of "false" extraction (that is, of mixtures rather than pure sources) were investigated. A simple data set was used, whose sources had very different distributions, so that the algorithm's output could be assessed manually if necessary. The work in this chapter moves closer to the extraction of seizure waveform from real-world data, while still keeping knowledge of the desired source so that the accuracy of the algorithm can be judged. However, the other underlying sources are not known for the work in this chapter, since they are from real-world EEG signals. A set of 2-second EEG epochs selected from EEG recordings of patients with epilepsy is described. The epochs do not contain obvious seizure waveform, and are not taken from time periods near to a patient's seizure, but some epochs contain typical EEG artifacts or anomalies, such as eye-blinks. Simulated seizure-like waveforms with different powers are added to these epochs. The CICA algorithm is then run in an attempt to extract the seizure-like waveform, given an approximate temporal reference.

The CICA algorithm options were the "standard" options tested in Chapter 3, that is, using a "general" contrast function with the input EEG data (a background epoch mixed with the synthetic seizure) whitened before each trial.

4.1 Real EEG epochs: identification of anomalies

A set of EEG recordings from patients was used to provide "background" EEG to which the simulated seizure-like waveform could be added. A 32-electrode referential montage was used, typically with Fz or FCz as the reference channel. (The effect of the reference was removed by a standard preprocessing: subtraction of EEG row and column means. This removed any constant offset from each channel, as well as any referencing to a single channel.) The first 25 channels of each recording conform to the standard 10-20 system, and only these channels were used for the CICA tests. The remaining 7 channels have different positions for different epochs, but sometimes provide extra information for manual analysis of the epochs. Although these patients had epilepsy, these 2-second epochs were extracted from time periods distant from seizures, to avoid confusion with the synthetic seizure waveform to be added. Ten epochs were selected because they had relatively "quiet" EEG, that is,

Epoch filename	Patient/date ID	Epoch start time
ongoing 1	LAM03DEC	16:00:03.290
ongoing 2	LAM03DEC	16:02:40.290
ongoing 3	LAM28NOV	14:43:12.830
ongoing 4	LAM28NOV	02:01:52.979
ongoing 5	LAM29NOV	18:01:10.900
ongoing 6	CHO22AUG	16:02:20.830
ongoing 7	CHO21AUG	16:02:01.540
ongoing 8	CHO22AUG	15:12:49.870
ongoing 9	CHO22AUG	17:01:53.685
ongoing 10	CHO22AUG	17:32:56.870

Table 4.1: Summary of two-second ongoing EEG data epochs which were used for trials

Epoch filename	Patient/date ID	Epoch start time	Anomaly interpretation
anomaly 1	LAM03DEC	16:00:15.290	Eye blink
anomaly 2	LAM03DEC	16:01:48.290	Mains electrical interference (60 Hz)
anomaly 3	LAM03DEC	16:00:45.290	Eye blink
anomaly 4	LAM03DEC	16:01:27.290	Eye blink
anomaly 5	LAM03DEC	16:03:54.720	Eye movement
anomaly 6	CHO19AUG	18:31:02.599	Epileptiform discharge
anomaly 7	CHO19AUG	18:31:42.599	Muscle movement
anomaly 8	CHO21AUG	16:02:43.540	Eye movement
anomaly 9	CHO19AUG	21:01:57.680	Mains electrical interference (60 Hz)
anomaly 10	CHO22AUG	17:31:10.870	Muscle movement

Table 4.2: Summary of two-second EEG data epochs featuring anomalies which were used for trials

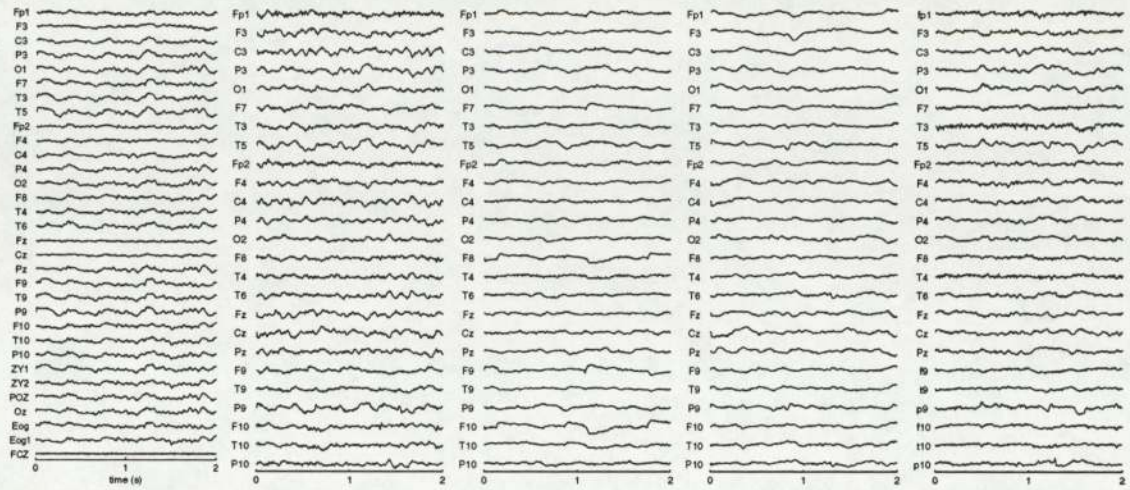
there were no severe artifacts. These ten epochs, labelled as "**ongoing** n", for $n = 1 \ldots 10$, are shown in Figure 4.1. The trial identifiers and time codes for each of these epochs are listed in Table 4.1.

Another 10 epochs which featured significant anomalies were selected, labelled as "**anomaly** n". The criteria used for selecting these was that they should be taken from the same original data as the **ongoing** n epochs described above, but should feature common EEG anomalies which are not present in ongoing EEG. The manual analysis of the anomalies in these 10 epochs is described below and summarised in Table 4.2.

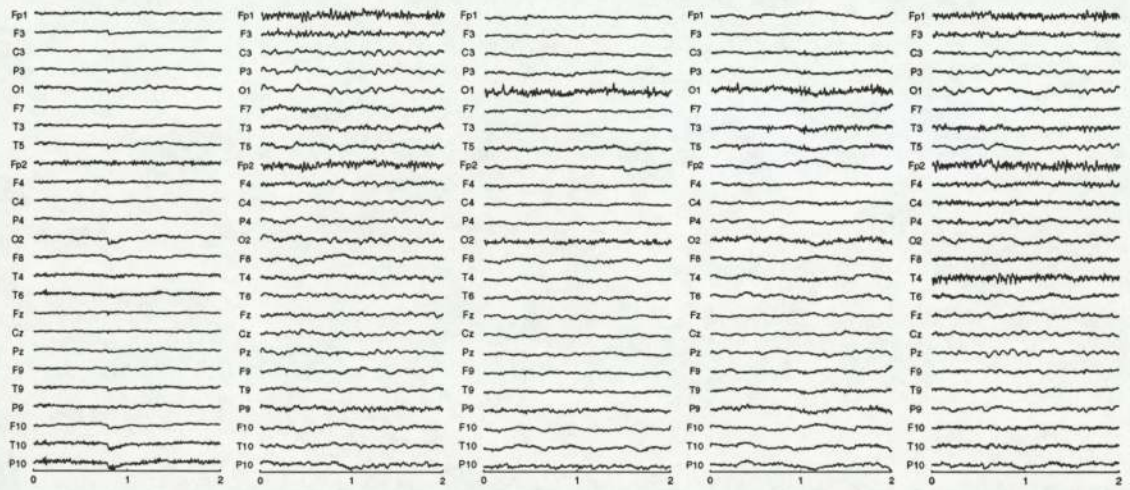
4.1.1 Ocular artifacts

Eye blinks produce short-duration artifacts in EEG recordings with a focus over the eyes. The artifacts should be especially prominent in the additional electro-oculogram electrode channels, if used, since these electrodes are positioned close to the eyes specifically to monitor eye movements. Among the epochs studied here, `anomaly 1`, `anomaly 3` and `anomaly 4` feature eye blink artifacts.

The electro-oculogram channels Eog and Eog1 show the artifact very clearly in `anomaly 1`. Figure 4.2(a) shows how the focus of the artifact can be determined by performing a Singular Value Decomposition (SVD) on the range of data encompassing the artifact. For data arranged in columns, as here, SVD is equivalent to Principal Component Analysis (PCA) [14]. The purpose of this operation is to find the best possible approximation to the data in a lower dimension. In this application, this dimension reduction will "clean" the artifact by removing other components. Since, in the range chosen, the artifact dominates the EEG, the first principal component is expected to be a good approximation to the artifact. (Only the first 25 (standard) channels were used for this analysis, for compatibility with the topographic mapping software.)



(a) ongoing 1 (b) ongoing 2 (c) ongoing 3 (d) ongoing 4 (e) ongoing 5



(f) ongoing 6 (g) ongoing 7 (h) ongoing 8 (i) ongoing 9 (j) ongoing 10

Figure 4.1: Selected two-second "quiet" EEG epochs with no obvious artifacts. The row and column means have been subtracted, as for all epochs discussed in this chapter.

An SVD operation on the (25x71) artifact range A yields:

$$A = USV^T \tag{4.1}$$

where U contains the principal components in columns. S is a diagonal matrix with singular values in decreasing order. We take the first column of U as a reasonable approximation of the artifact's topographic map. A software package [8] can then be used to map this onto a spherical head model (Figure 4.2(b)). This mapping shows a focus over the eyes, which strengthens the interpretation of this artifact as an eye blink.

The same analysis gave similar results for `anomaly 3` and `anomaly 4` (Figure 4.2), suggesting that these epochs also feature eye blink artifacts.

Eye movements, rather than blinks, produce slightly different artifacts. The focus is equivalent to a dipole located horizontally across the eyes, as shown in the topographies in Figure 4.3.

4.1.2 Epileptiform discharges

The waveform in the `anomaly 6` epoch (Figure 4.4) seems to originate in the right temporal lobe. This focus and the morphology of the anomaly suggest that it is a epileptiform discharge (spike) frequently found in the inter-ictal EEG of patients with epilepsy [2]. This anomaly is not an artifact, but such waveforms are not a feature of ongoing EEG. They may appear quite frequently in the inter-ictal EEG of patients with epilepsy, however, so the effect of these discharges on the algorithm's ability to extract seizure waveform is of particular interest.

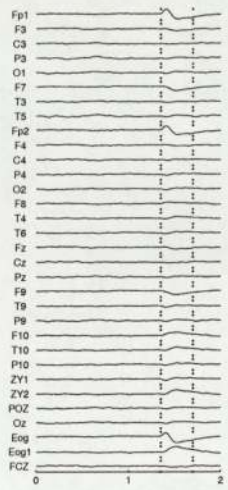
4.1.3 Electrical interference

The artifact in the `anomaly 2` epoch appears to be due to external electrical interference, since it affects almost all the channels to the same extent. (Additionally, an eye movement artifact occurs during the period marked.) A time-frequency representation of a typical channel's signal shows that, when the high frequency waveform is present, there is significant power at around 60 Hz. This corresponds to the mains frequency in Canada, where the recordings were made. The interference could be caused by nearby mains cables or equipment. Such common-mode noise should be rejected by the process of subtracting the signal from the reference electrode from the other EEG signals. However, the reference electrode (FCz) appears to be just as much affected as the other channels. This suggests that there could have been a fault with this reference electrode, or the artifact may have only affected FCz, whose signal, when subtracted from the other channels, caused the artifact to appear generally. A fault in the reference connection is another possible cause [4].

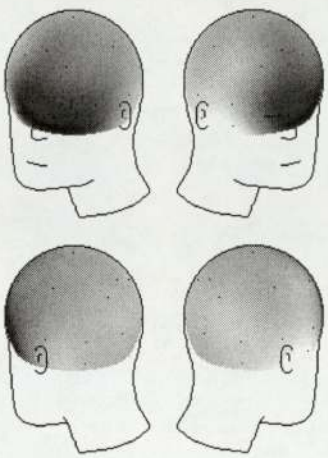
A similar artifact appears for the whole of the `anomaly 9` epoch, but only on channel T4. The earth connection for this particular electrode may have been faulty. The concentration of the power in this channel at mains frequency is shown by the corresponding spectrogram in Figure 4.5.

4.1.4 Muscle artifacts

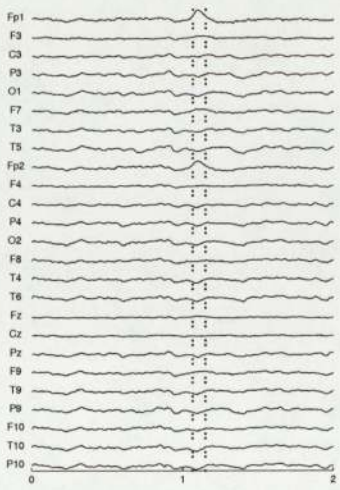
The `anomaly 7` and `anomaly 10` epochs each have several artifacts which seem to be high frequency interference patterns from muscle movements (electromyograms: EMG) made by the patient (Figure 4.6). The waveforms are typical of "chewing" jaw movements, since the foci are close to the positions



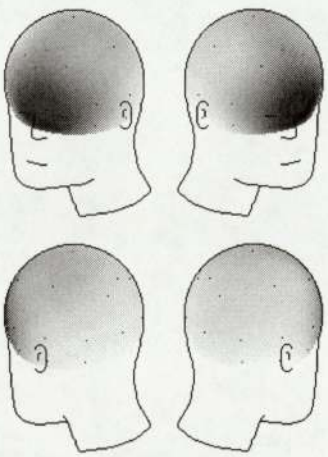
(a) `anomaly 1`, artifact marked



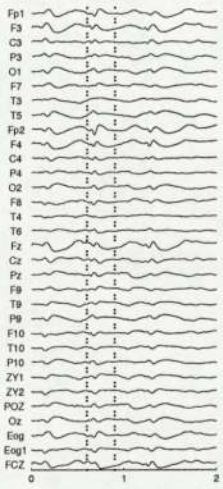
(b) `anomaly 1` artifact topography



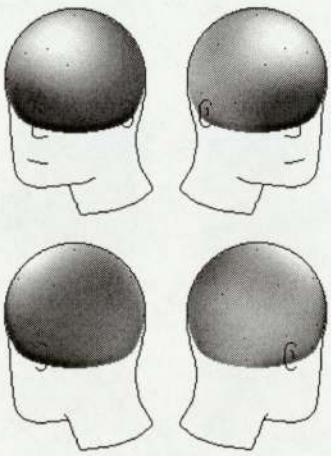
(c) `anomaly 3`, artifact marked



(d) `anomaly 3` artifact topography

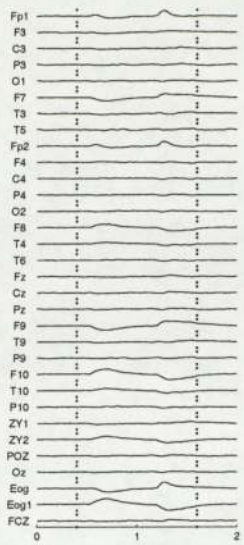


(e) `anomaly 4`, artifact marked

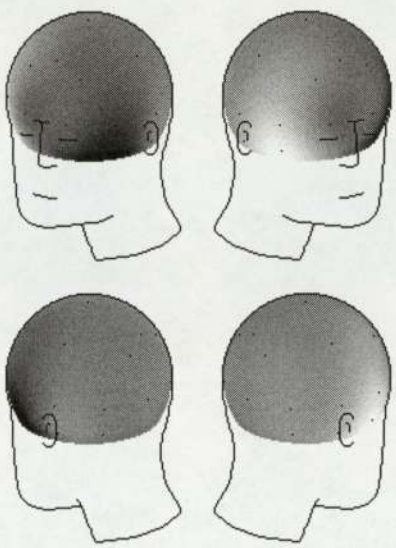


(f) `anomaly 4` artifact topography

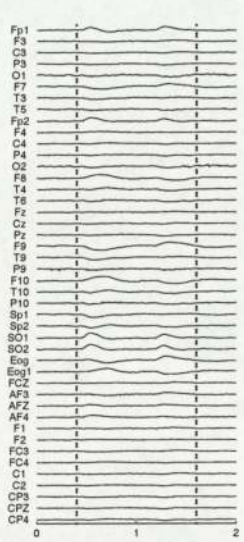
Figure 4.2: Analysis of eye blink artifacts using SVD to find the first principal component, which is projected onto a spherical head model. The time period of each artifact is marked. The topographies indicate the foci by extremes of light and dark. The clear foci over the eyes in each case suggest that these artifacts are eye blinks.



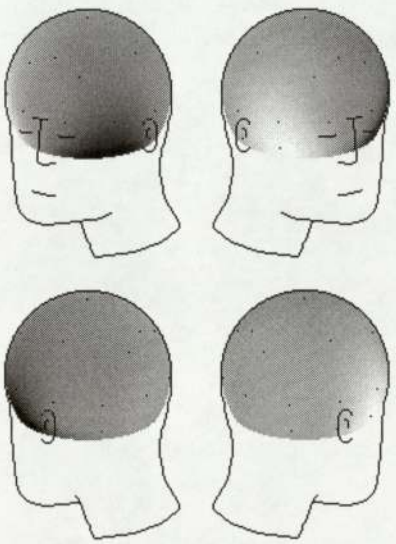
(a) `anomaly 5` with artifact range marked



(b) `anomaly 5` artifact topography, showing a dipole across the eyes. This suggests that eye movements caused the artifact.

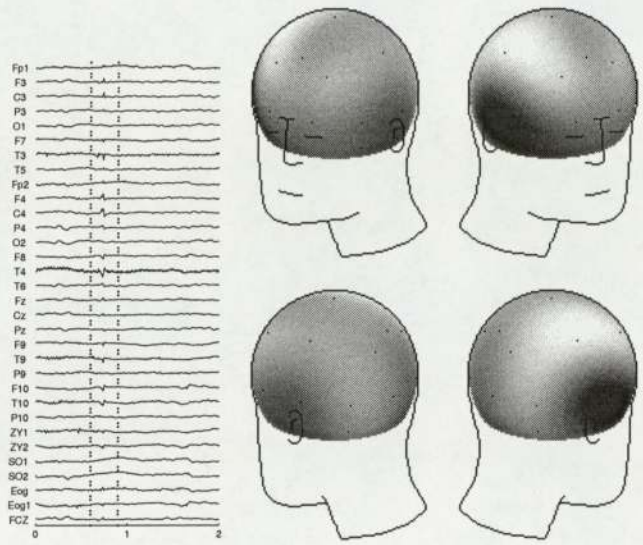


(c) `anomaly 8` with artifact range marked



(d) `anomaly 8` artifact topography, showing a dipole across the eyes, which is likely to be due to eye movement.

Figure 4.3: Identification of eye movement artifacts using SVD to determine the first principal component, which is projected onto a spherical head model in the same procedure as for Figure 4.2.



(a) `anomaly 6` with anomaly marked

(b) `anomaly 6` anomaly topography

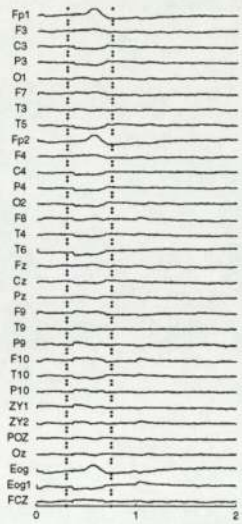
Figure 4.4: The waveform in the `anomaly 6` epoch resembles an epileptiform discharge focused in the right temporal lobe.

of the jaw muscles. Furthermore, it is reasonable to assume that the patient might be eating a meal at the time codes of these extracted epochs (approximately 1830 and 1730 respectively).

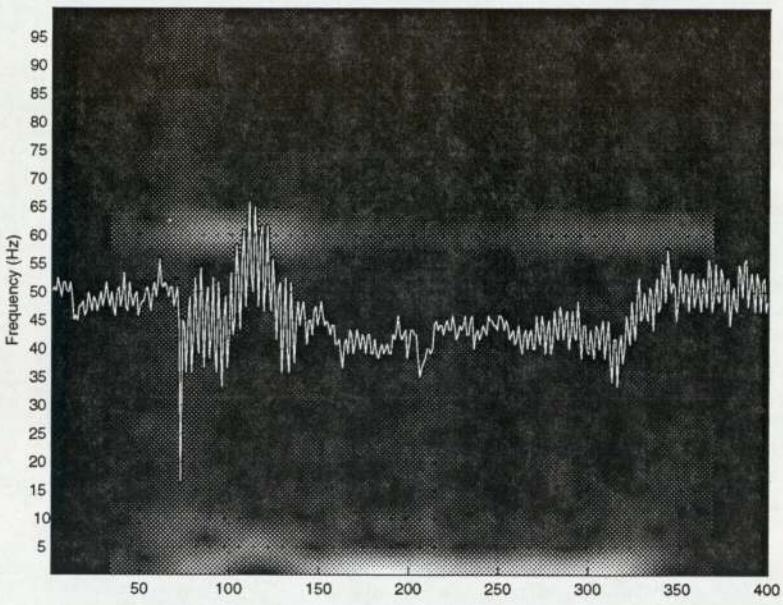
4.2 Simulated seizure waveform

The previous section describes the twenty EEG epochs, featuring both "quiet" EEG and anomalies, which were selected from real-world data. A simulated seizure waveform was generated from a 5 Hz sine wave, sampled at 200 Hz for 2 seconds to match the real EEG epochs. A sine wave of this frequency was considered a reasonable approximation to real seizure waveforms, of which examples are shown in Chapter 6. The sine wave doubles in amplitude over the 2 second duration, modelling a growing seizure. This signal was used to modulate a current dipole whose electric field could be projected onto a spherical head model (Figure 4.7), allowing the potential at each electrode to be calculated (Figure 4.8(a)).

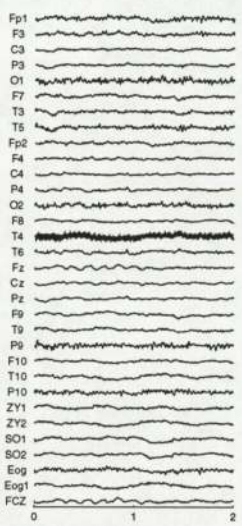
The data set for these trials was generated by adding this simulated seizure waveform (the "signal") to each of the real EEG epochs (the "noise") at different signal-to-noise ratios (SNRs). In this way, the ability of the CICA algorithm to extract seizure waveform "buried" to varying extents in other EEG signals could be assessed. There is no standard way of measuring the power in a multi-channel EEG epoch, in order to calculate the SNR. The variance of the samples from each channel can be used to estimate the power in that channel, but the difficulty lies in combining these channel powers to give an overall power estimate, since one channel with unusually high or low power may distort the overall estimate. The following EEG power measure was adopted for this work:



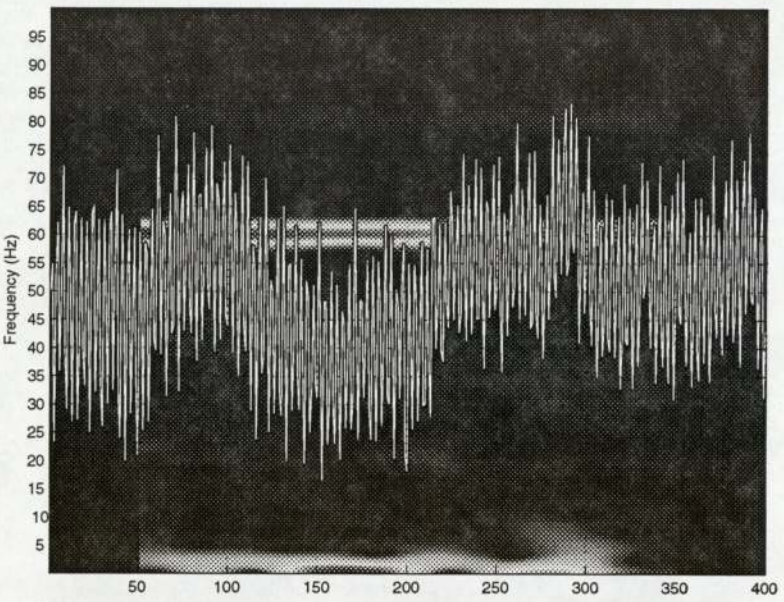
(a) **anomaly 2** with artifact marked



(b) A spectrogram of channel F3 of **anomaly 2**, during the marked time period, with the signal superimposed.

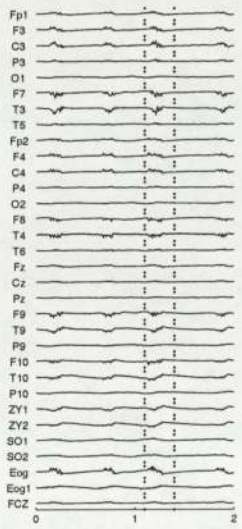


(c) **anomaly 9** data

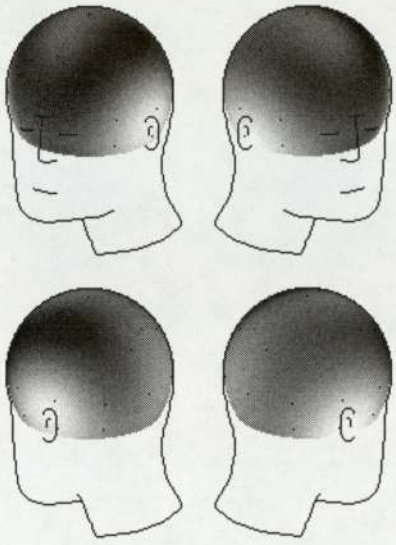


(d) A spectrogram of channel T4 of **anomaly 9**, for the whole epoch, with the signal superimposed.

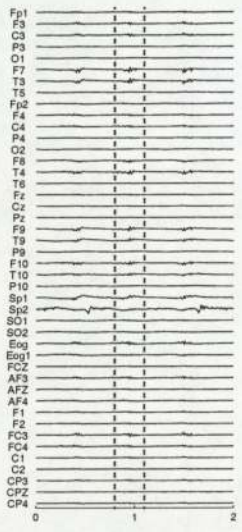
Figure 4.5: The artifacts in epochs `anomaly 2` and `anomaly 9` appear to be due to mains interference. (An eye movement artifact is also present in `anomaly 2`.) The frequency spectra of the affected channels were analysed. At times when the high frequency artifact is apparent in the EEG data, the power is greatest at around 60 Hz. This is the mains frequency in Canada where the recordings were made.



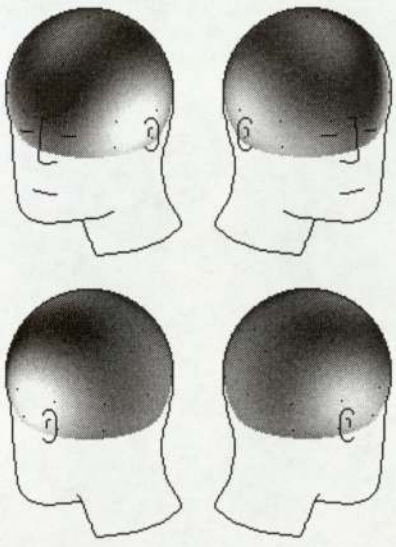
(a) `anomaly 7` with one of the artifacts marked



(b) Topography of `anomaly 7` artifact, showing foci in the left and right temporal lobes



(c) `anomaly 10` with one of the artifacts marked



(d) Topography of `anomaly 10` artifact, showing foci in the left and right temporal lobes

Figure 4.6: The `anomaly 10` artifacts appear to be characteristic of "chewing" jaw movements. The foci for these artifacts are close to the jaw muscles for both epochs.

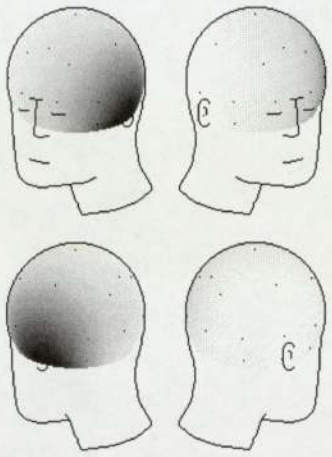


Figure 4.7: Topographic map of the focus of the simulated seizure EEG. The spatial focus of the simulated seizure was chosen to be in the left temporal lobe, a reasonable choice since temporal lobe epilepsy is the most common form of epilepsy. The amplitude-modulated sine wave (Figure 4.8(a)) was used to modulate a simulated current dipole with this focus, which was projected onto the 25 electrode positions to give a simulated EEG.

$$P_{EEG} = \sum_{c=1}^{25} \sigma_c^2 \tag{4.2}$$

where c ranges over all the EEG channels.

The SNR measure is then defined as:

$$SNR = 10 \log_{10} \frac{P_{signal}}{P_{noise}} = 10 \log_{10} \frac{\sum_{c=1}^{25} \sigma_{cs}^2}{\sum_{c=1}^{25} \sigma_{cn}^2} \tag{4.3}$$

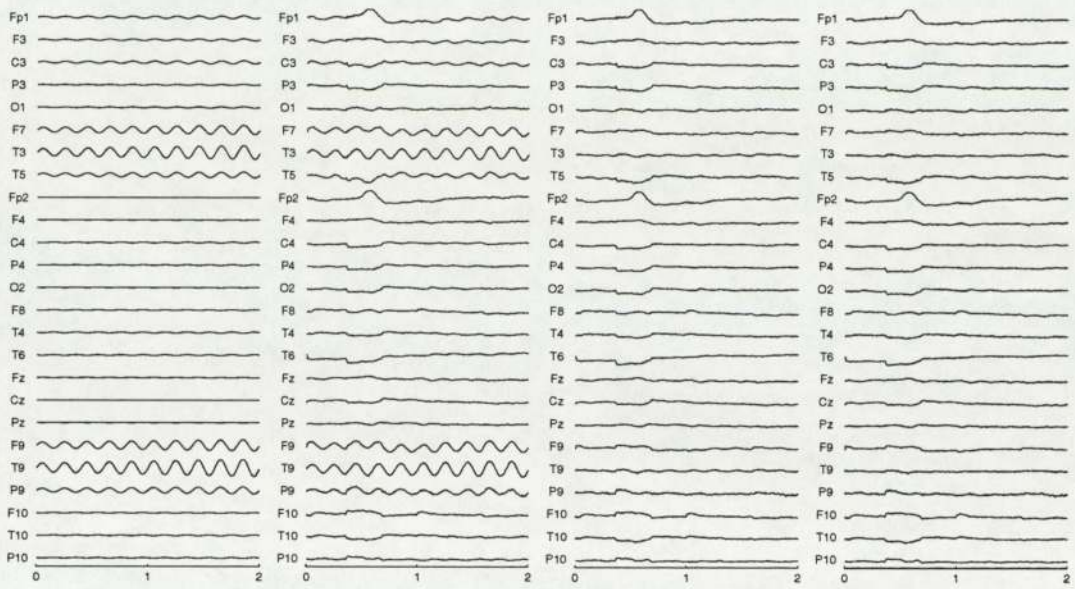
Rearranging this expression shows that a desired SNR S can be achieved by scaling the real EEG epoch data by

$$\sqrt{\frac{P_{signal}}{P_{noise} \times 10^{\frac{S}{10}}}} \tag{4.4}$$

before adding the simulated seizure data. Figure 4.8 shows the effect of adding the simulated seizure waveform with one of the real EEG epochs at the SNRs used for these trials (0 dB, -20 dB, -40 dB and -60 dB).

4.3 Tests of extraction of seizure waveform

The procedure shown in Figure 4.8 was repeated for all the real `ongoing` n and `anomaly` n epochs. The CICA algorithm was run 5000 times, with random starting vectors, on each of the combinations of real EEG epoch and SNR. The reference signal given was a 5 Hz square wave and the phase of this reference exactly matched the known seizure waveform. (The problem of matching the phase of the reference to an unknown underlying seizure waveform is addressed in Chapter 6.) For each of these trials the accuracy of the extraction was assessed by an MSE threshold, as for the work in the previous chapter. The MSE between the CICA output signal and the growing sine wave was calculated and compared to a threshold value of 0.5. The output signal power and number of iterations taken for each trial were recorded, along with the estimated mixing vector **a**.

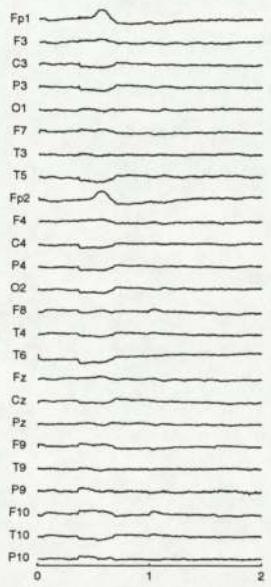


(a) Simulated seizure waveform

(b) Summation of **anomaly 2** and simulated seizure waveform with SNR 0 dB

(c) Summation of **anomaly 2** and simulated seizure waveform with SNR -20 dB

(d) Summation of **anomaly 2** and simulated seizure waveform with SNR -40 dB



(e) Summation of **anomaly 2** and simulated seizure waveform with SNR -60 dB

Figure 4.8: Addition of a simulated seizure waveform, at different powers, to a real-world EEG epoch (`anomaly 2`): (a) the simulated seizure waveform, (b) a mixture with SNR 0 dB, (c) a mixture with SNR -20 dB, (d) a mixture with SNR -40 dB, (e) a mixture with SNR -60 dB.

The results of these trials were first assessed by comparing each output signal, using an MSE threshold, to the seizure waveform. An MSE threshold of 0.5 was used, as in the previous chapter, to classify the results. These results are shown in Figures 4.11 and 4.12. It appears that using different real background EEG epochs did not have a consistent effect on the results and the presence of the artifacts used has not greatly affected the algorithm's ability to extract the simulated seizure waveform. Furthermore, while it might be expected that a lower SNR would reduce the algorithm's accuracy by "burying" the seizure waveform among the other EEG signals, no such relationship is shown in the results. This may be due to the EEG power measure used, which was described earlier. The artifacts do not in general appear at the same magnitude on all EEG channels. (Indeed, the foci of the artifacts were previously used to identify some of them.) The simulated seizure waveform is also stronger on some channels than others. The "sum of variances" power measure does not take account of the effects of mixing these unevenly balanced signals. For example, if an artifact appears only on a channel where the simulated seizure does not appear, the artifact will not greatly affect the morphology of the seizure waveform when the two EEG epochs are mixed. However, this SNR measure will take account of the artifact by increasing the power of the mixed simulated seizure waveform. Hence the algorithm appears to have performed better on epochs where an artifact was present (Figure 4.11), because the power of these epochs was over-estimated. A better measure of SNR might divide the power of each signal channel by the power of the corresponding noise channel. The largest ratio would then be the SNR estimate. This measure would still be imperfect because several channels with low SNRs might give the CICA algorithm more information on the signal source than a single channel with a low SNR.

4.4 Algorithm behaviour when given a "false" reference

Since the SNR, when varied between 0 dB and -60 dB, seemed to have no consistent effect on the algorithm's performance, it might be thought that the algorithm is producing false extractions. In other words, regardless of the power of the seizure waveform, the algorithm might mix the sources from the real EEG data to produce an output similar to the reference. This suggestion was refuted by a second execution of the trials described above, with no simulated seizure waveform at all mixed with the real EEG epochs. The same random number generator seeds were used to give the same algorithm starting vectors. When the output signals were compared to the sinusoidal seizure waveform, the MSE was always above the threshold of 0.5, showing that the algorithm never "falsely" extracted seizure waveform by this criterion. Furthermore, a comparison against the square wave reference signal also gave no matches by the MSE criterion. These results imply that, given a suitable reference, the algorithm can extract seizure waveform which is "buried" within other EEG signals, but will not produce a seizure-like output if no such source is actually present.

In the previous chapter, the power of the dewhitened output signal from the CICA algorithm seemed to be a useful indicator of the extraction of a mixture of sources rather than one pure source. For the trials described in this chapter, only one of the true underlying sources was known (the simulated seizure source). It was not therefore possible to determine whether the algorithm's output was a mixture of the sources occurring in the real-world EEG data. However, it is reasonable to assume that the occurrence of "mixture" outputs might be associated with a "false" reference: if the given reference is unlike any of the true underlying sources, the algorithm is more likely to mix these sources in attempting to match the reference. The possible association between output power and

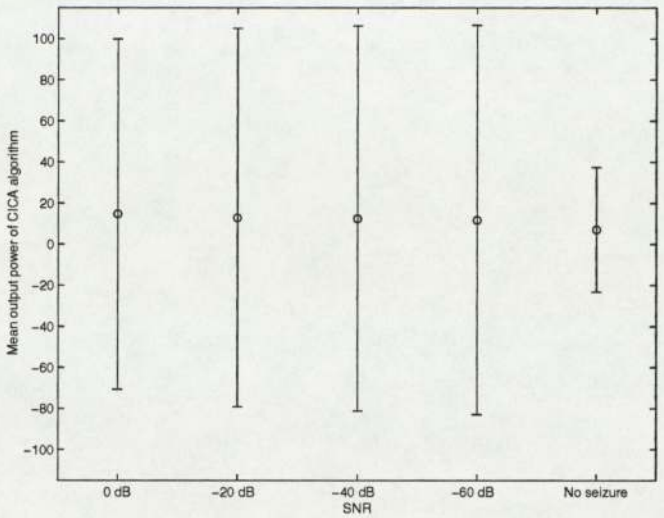


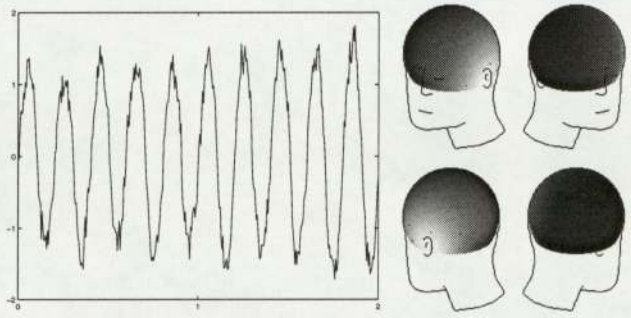
Figure 4.9: Mean power of output of CICA algorithm, with the variance shown as error bars. The mean power decreases as the SNR increases, for trials where some seizure waveform was present. The trials with no seizure waveform present have a lower mean power output, but the large variances show that use of output power as an indicator of this situation would have a high error rate.

the use of a "false" reference was studied by finding the mean and variance of the output powers for the different SNRs used in these trials (Figure 4.9). For trials where some seizure waveform was present in the data supplied to the CICA algorithm, the mean output power decreased as the SNR decreased. When no seizure waveform was mixed with the real data, the mean power was lower still and the variance was lower. This suggests that, although the mean output power might be used as an indicator of the use of a "false" reference, this indicator would have a large false positive rate due to the large variance in output power when seizure waveform is present.

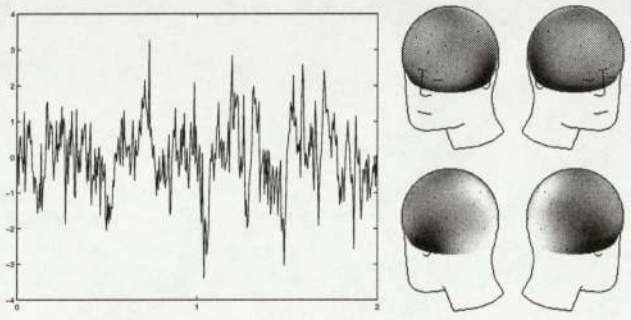
4.5 Spatial closeness for classification of results

The results described above were obtained by assessing the algorithm's output signals in the time domain. This MSE criterion compares the morphology of each output signal to the known simulated seizure signal. Alternatively, it was thought that the closeness of the spatial topography of the output compared to the known spatial distribution of the seizure signal could be used to classify the results. As Figure 4.10 shows, the spatial foci for accurate extractions of seizure waveform match the true spatial distribution (Figure 4.7) well, but the incorrect extractions have a different spatial map. To attempt to quantify this difference, the estimated mixing vector **a** was normalised and compared to the normalised known mixing vector. In this way the focus of the algorithm's output is compared to the actual focus of the simulated seizure (Figure 4.7).

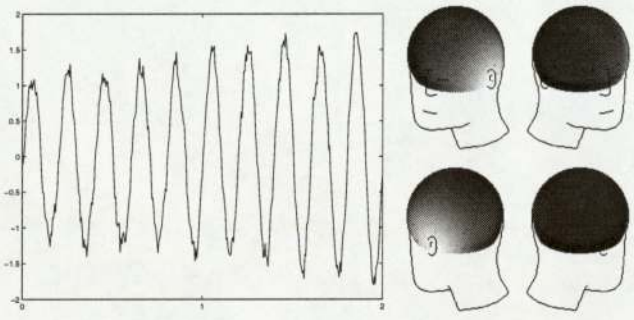
However, this spatial closeness test did not give useful results for these trials. The horizontal axis of the scatter plot in Figure 4.13 shows values of this spatial MSE test. The test does not clearly separate the points into correct and incorrect extractions and was not therefore used to classify the results. By contrast, the vertical axis of this plot shows the time domain MSE test, as used for classification of results in this chapter and the previous chapter. There is good separation between the outputs which match the simulated seizure waveform (those with low time domain MSE) and non-matches (the cluster of points with larger MSE values). The threshold of 0.5 for this MSE test



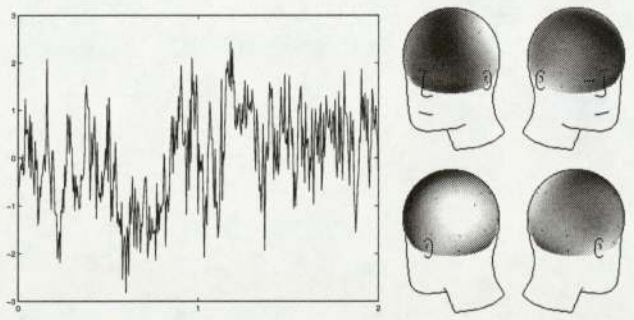
(a) Match, with MSE 0.0391.



(c) No match, with MSE 1.57.



(e) Match, with MSE 0.0173.



(g) No match, with MSE 1.90.

Figure 4.10: Typical outputs of the CICA algorithm when extracting simulated seizure waveform which had been mixed with different real background EEG epochs. All trials were classified by the MSE between the algorithm output and the known seizure waveform. If this MSE was less than 0.5, the result was classified as a "match". The topographic map from the estimated mixing vector **a** for each extraction is also shown.

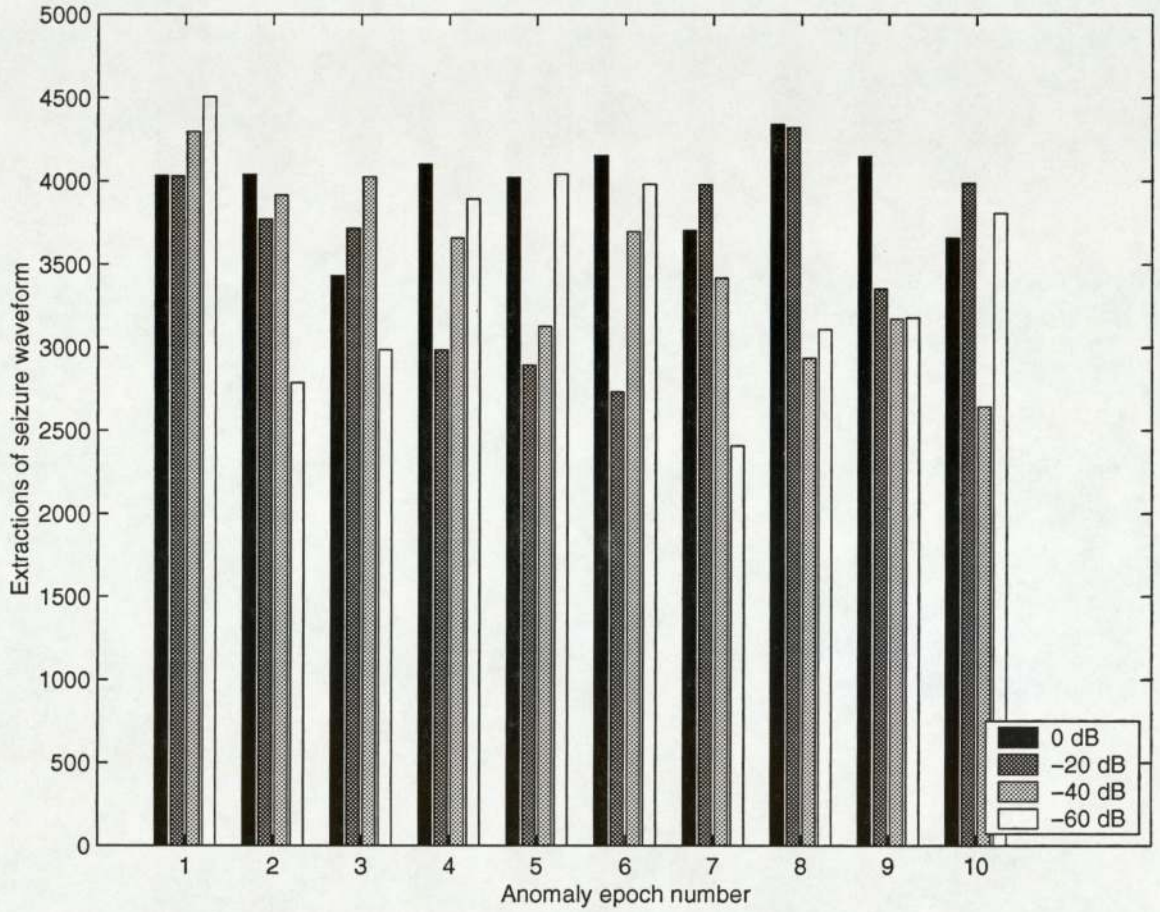


Figure 4.11: Results of trials with the anomaly EEG epochs, with simulated seizure waveform mixed at different SNRs. 5000 trials were performed for each combination of anomaly epoch and SNR, and the output signals were compared to the seizure waveform. The numbers of matches by an MSE threshold (of 0.5) are shown. In general, the algorithm's performance is not strongly dependent on the SNR or epoch used.

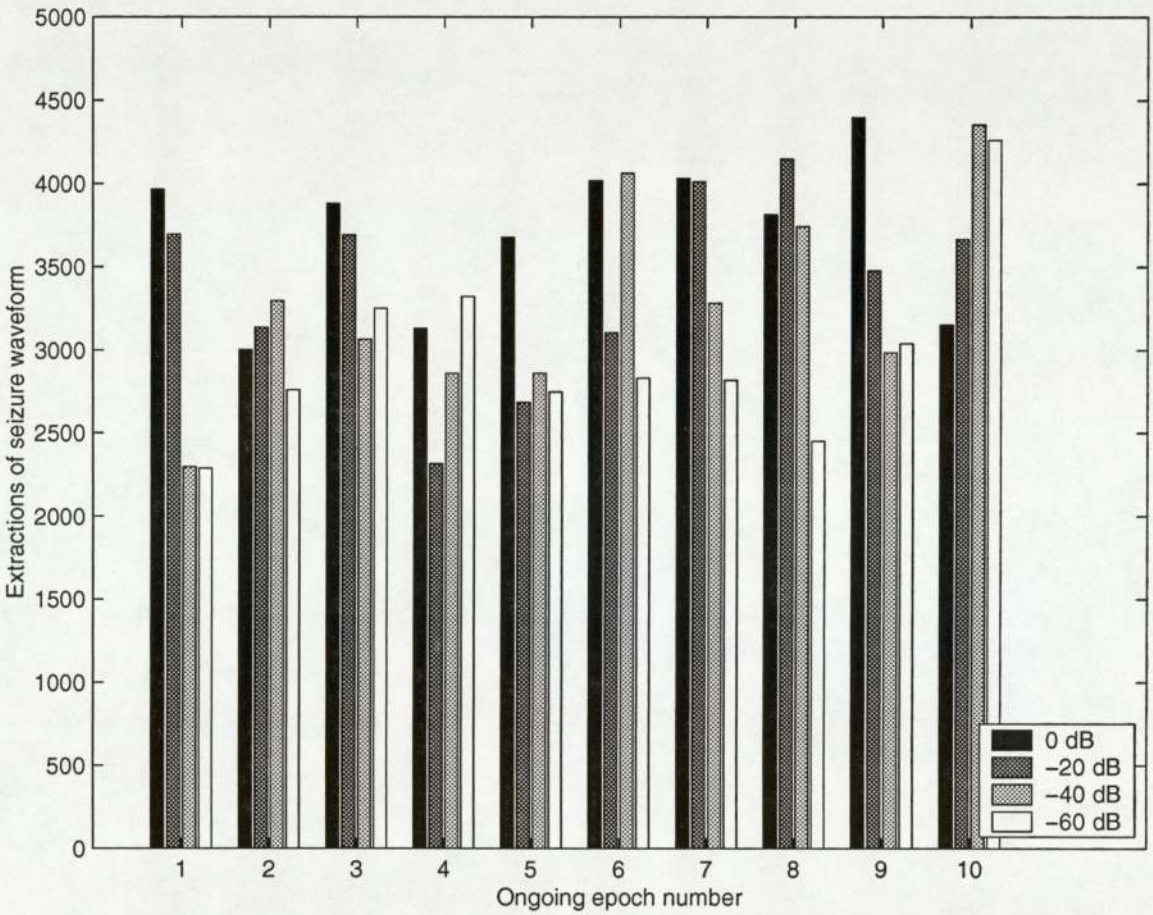


Figure 4.12: Results of trials with the ongoing EEG epochs, executed as described for Figure 4.11. Again, the algorithm's performance shows no obvious dependence on the SNR or epoch used.

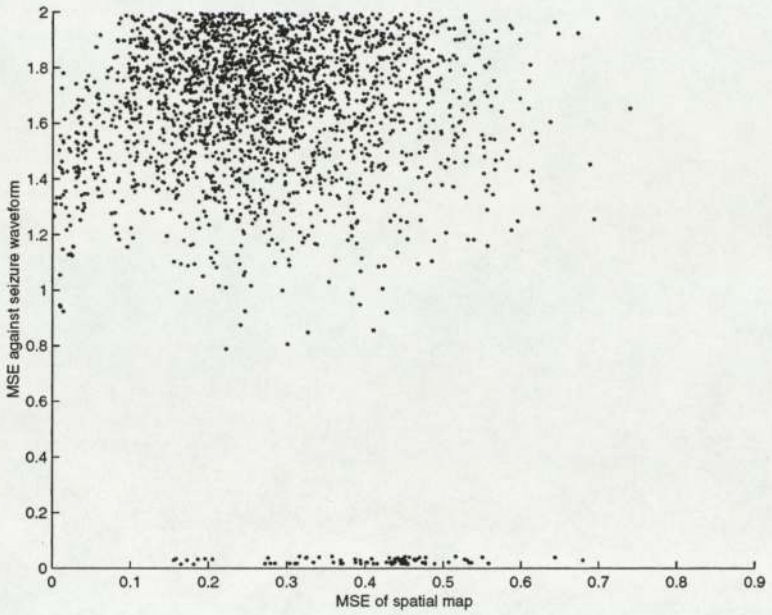


Figure 4.13: Scatter plot of 8000 output signals (randomly selected from the 400,000 trials) showing the MSE of the time domain output signal against the known seizure waveform and the MSE of the estimated mixing vector **a** against the known spatial map. The good separation of the time domain MSE test can be seen and the threshold of 0.5 is appropriate for this test. However, the spatial domain MSE test shows poor separation with this data set.

could not be tested with an ROC curve, as in the previous chapter, because only one of the underlying sources is known for this data set. Even so, this threshold seems reasonable for this data set because it separates the two clusters of points shown in this scatter plot.

4.6 Summary

The data set described in this chapter was obtained from a set of two-second real EEG epochs to which a simulated seizure waveform was added at different SNRs. This required that a measure of power in a multi-channel EEG epoch was defined. The ability of the CICA algorithm to extract the seizure waveform was then tested. Some of the real EEG epochs contained artifacts or anomalies commonly found in EEG recordings, which were identified manually. The algorithm was generally robust to these artifacts and performed reasonably well with different SNR values. Furthermore, when no seizure waveform was present, but the same reference was used, the algorithm did not extract any signals matching either the seizure waveform or the reference, by the MSE criterion. These results imply that the algorithm can be used to extract real seizure waveforms from EEG where none of the underlying sources is known; this application of the algorithm to real epileptiform EEG epochs is described in Chapter 6. The output power of the CICA algorithm seemed to be related to the occurrence of "false" extractions, when no underlying source resembled the reference signal, but the use of output power as an indicator of this situation would have given a large false positive rate for this data set.

Chapter 5

Artifact rejection using the CICA algorithm

Since the focus of this project is on the use of the CICA algorithm for seizure onset analysis, EEG artifacts have so far only been considered as possible sources of error in the extraction of seizure waveform. (Chapter 4 describes the use of selected artifacts added to a synthetic seizure waveform for tests of CICA.) However, the problem of artifact removal from EEG signals is of great interest, even if the signals are to be analysed manually rather than, as here, automatically. The CICA algorithm appears to be a useful tool for this purpose, if a suitable reference signal is provided. If the artifact can be accurately extracted as an independent component, along with its estimated mixing vector **a**, it can be removed by projecting the component to the measurement space using **a** and subtracting from the EEG signals. This chapter describes some examples of this process for the rejection of EEG eye blink artifacts. Although the time available for this project did not permit a digression into automated trials of CICA for artifact rejection, some comments on algorithm performance and the choice of reference signals for this application are made.

5.1 Rejection of eye blink artifacts

Figure 5.1 shows how eye blink artifacts can be removed using the CICA algorithm. A 20 second EEG epoch was used, during which a seizure develops. A threshold was applied to channel Fp1, where the eye blink artifacts are prominent, to generate a reference signal which has a narrow square pulse at each artifact. Figure 5.1(d) shows the output of the CICA algorithm, given all 25 channels of EEG data as the input mixture, along with the reference signal. This output does not contain the seizure waveforms which increase in amplitude through the epoch. The topographic map of the mixing vector **a** for this extraction is shown, with a clear focus over the eyes.

The extracted waveform was then projected back to the data space and subtracted from the original EEG data. As Figure 5.1(e) shows, the artifacts have been almost completely removed from Fp1, where they were previously prominent.

In the previous chapter, the EEG epochs **anomaly 1**, **anomaly 3** and **anomaly 4** were shown to contain eye blink artifacts. Figure 5.2 shows the process of artifact rejection on **anomaly 3**. A threshold was applied to an EEG channel where the artifact is prominent to obtain a reference

signal. The CICA algorithm extracted the artifact, although with some other signals present at lower amplitude. The artifact was projected to the data space and subtracted from the original EEG epoch, removing the artifact.

Another example of eye blink artifact removal using this process is shown in Figure 5.3. The two-second epoch used for this example is from a seizure recording described in Chapter 6 as Seizure 5, taken from between t=1s and t=3s. The algorithm again extracts the eye blink and its spatial distribution, so that it can be removed from the EEG recording.

5.2 Algorithm behaviour given a "false" reference

In the trials described in the previous chapters, the behaviour of the algorithm given a "false" reference was studied. An example of this situation was performed for artifact rejection on the **anomaly 3** epoch shown in the previous section: the square wave reference shown as a dashed line in Figure 5.2(d) was moved through time in increments of one sample. The power of the dewhitened CICA algorithm output was calculated for each reference position (Figure 5.4). The lack of structure in this graph suggests that output power would not be a reliable indicator of "false" extractions - there is no clear difference in power when the reference position is between 200 and 250 samples, from where the genuine artifact was extracted. An example of the algorithm's output when the reference was moved to a time period where there was little obvious EEG activity is shown in Figure 5.5. Since the algorithm's output has a morphology similar to the reference, it might appear that the algorithm has mixed the true underlying sources to achieve a high correlation with the reference. However, the true underlying sources are not known in this case, so conclusions about the statistical independence of the output cannot be drawn. A promising result is that, when this output is projected into the data space and subtracted from the original epoch, there is very little difference between the "cleaned" and "original" EEG data. In this case the "false" reference has not caused visible distortion or the introduction of a new artifact.

5.3 Summary

The examples in this chapter show how the CICA algorithm can be used for EEG artifact rejection. Since this application was not the main focus of this project, a fuller analysis of the algorithm in this application is not presented due to time constraints. It is apparent, however, that the CICA algorithm appears to offer a novel method in the field of artifact rejection. This field is relevant to seizure onset analysis since in some cases artifacts might obscure the seizure waveforms. A two-stage analysis system in which artifacts are removed before seizure waveform is extracted might be envisaged.

The references used for the examples in this chapter were obtained by a manually-chosen threshold which was applied to an EEG channel where the artifact was prominent. However, the morphologies of the references were so coarse (square waveforms) that automatic generation of reference signals should, in principle, be possible. Further work on the CICA algorithm's application to artifact rejection should address issues such as the choice of reference signal, and test the algorithm on a wider range of EEG artifacts.

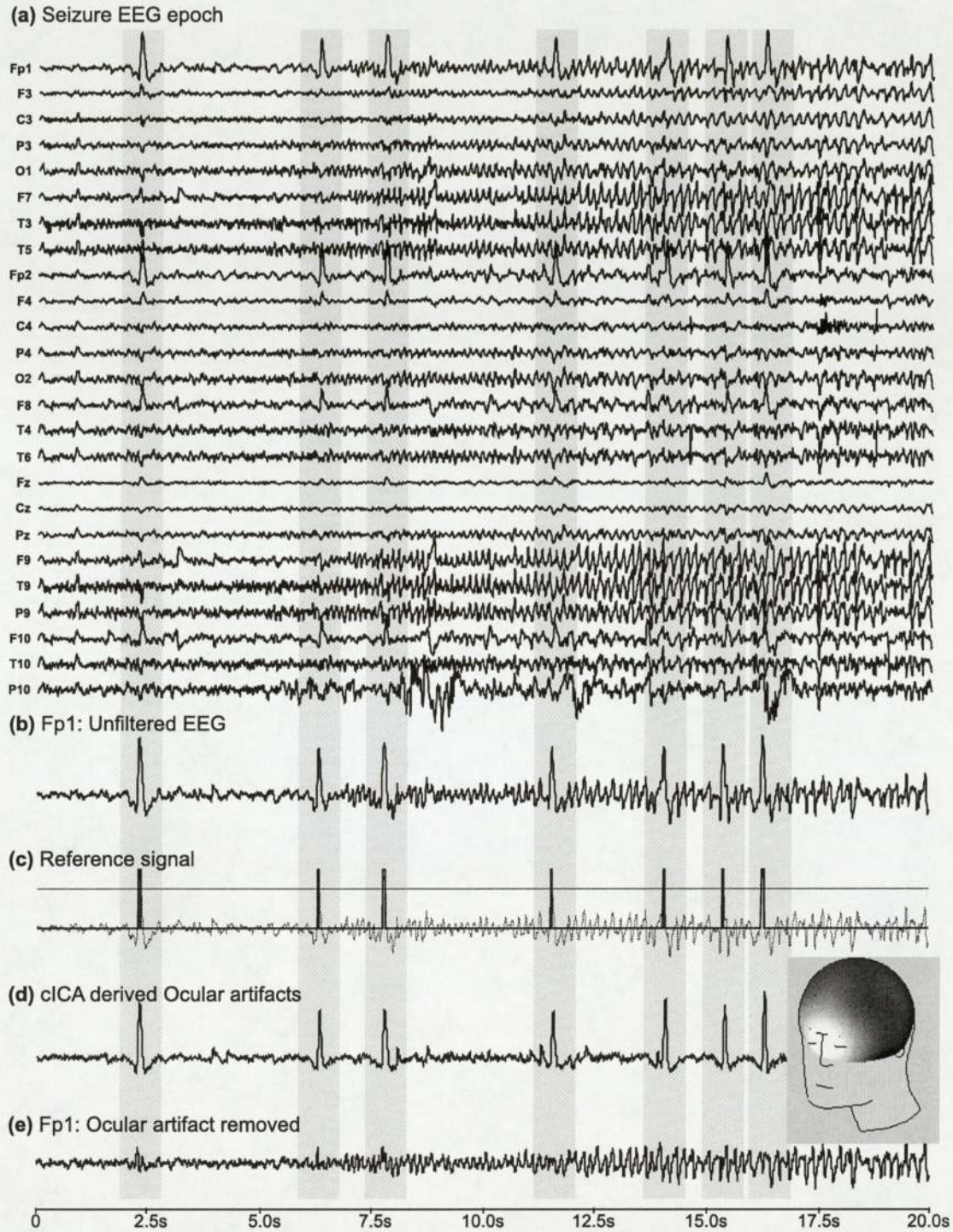
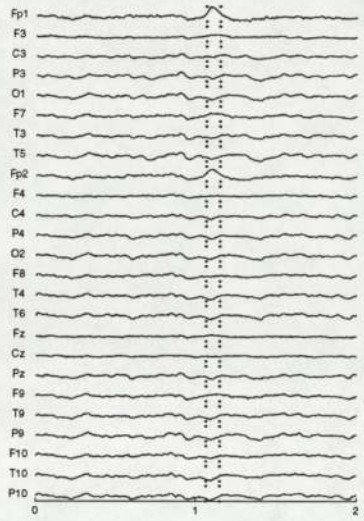
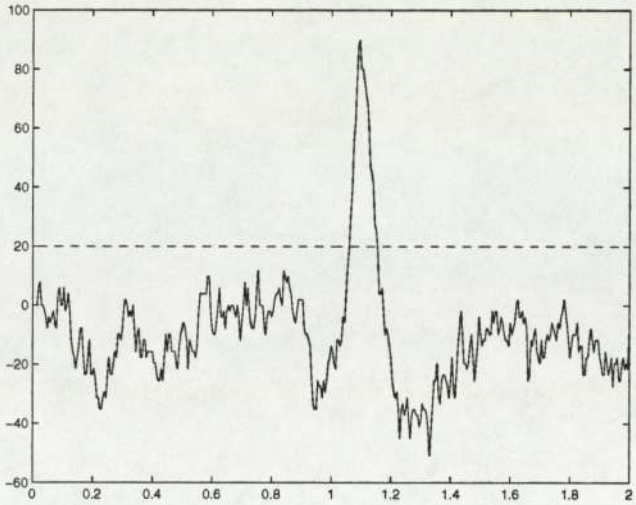


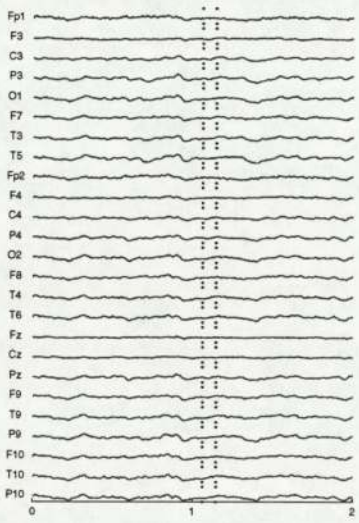
Figure 5.1: Removal of eye blink artifacts from an EEG epoch. Channel Fp1 of the original data is used to provide a reference (c) for the algorithm, which extracts the artifacts as an independent component (d). The topography of the extracted component has a focus over the eyes. When the component is projected to the data space and subtracted from the original data, the artifacts are largely removed.



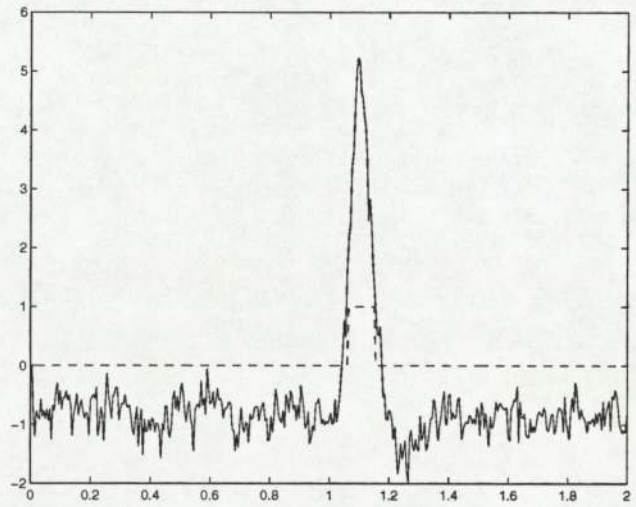
(a) Original **anomaly 3** epoch, with the eye blink artifact marked.



(b) Channel Fp1, with a threshold placed at 20 (dashed line).

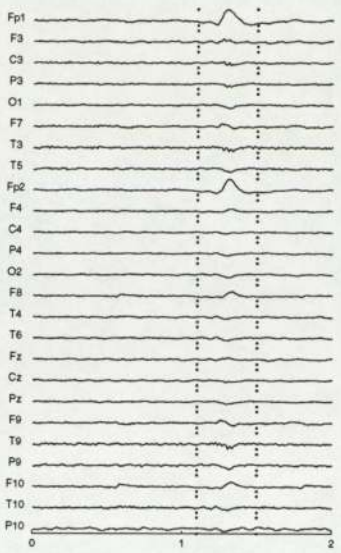


(c) **anomaly 3** with artifact subtracted, plotted at the same scale as (a) with the same section marked.

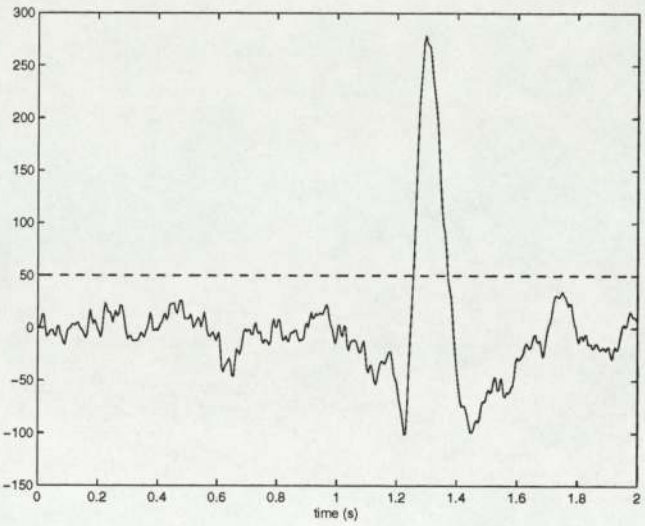


(d) Component extracted by CICA, with reference superimposed as dashed line.

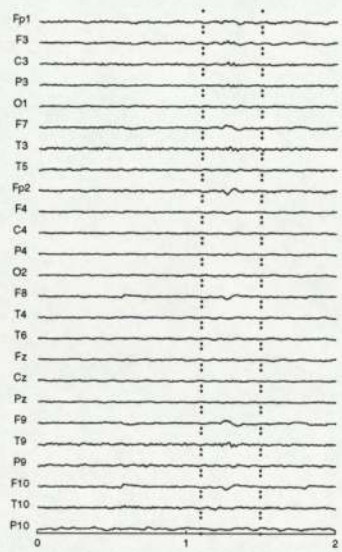
Figure 5.2: Rejection of an eye blink artifact from the **anomaly 3** epoch. A reference for the CICA algorithm was obtained from a threshold applied to the Fp1 channel (where the artifact is prominent). The extracted component was projected to the data space and subtracted from the epoch, removing the artifact.



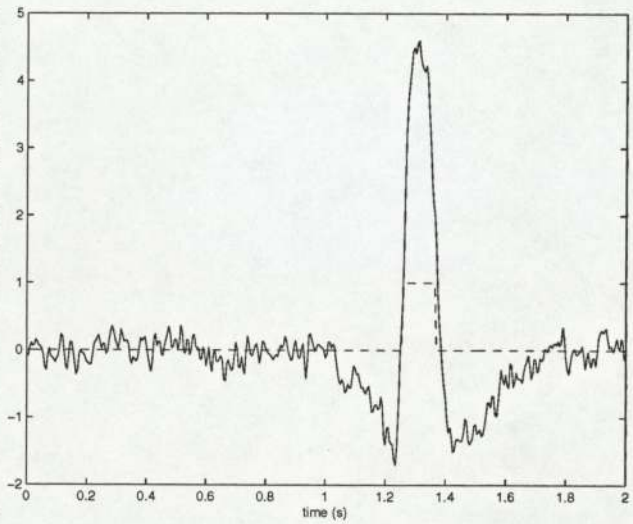
(a) Original two-second epoch featuring an eye blink



(b) Channel Fp1, with a threshold placed at 50 (dashed line.



(c) The same epoch plotted to the same scale, with the artifact subtracted.



(d) Component extracted by CICA, with reference superimposed as dashed line.

Figure 5.3: Rejection of an eye blink artifact from an EEG seizure recording (the complete recording is shown in Figure 6.26). A threshold was applied to a channel (Fp1) where the artifact was prominent, giving a reference signal. With this reference the CICA algorithm extracted the eye blink component along with its spatial distribution. When this extracted artifact was subtracted from the original EEG, the eye blink was largely removed, although some of the artifact is still visible.

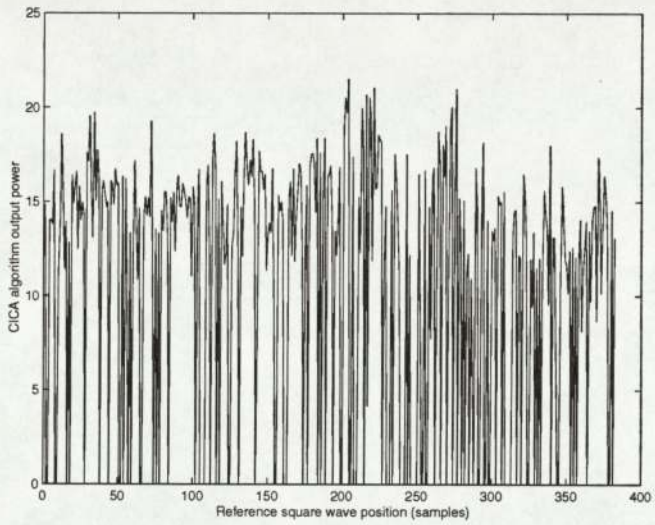
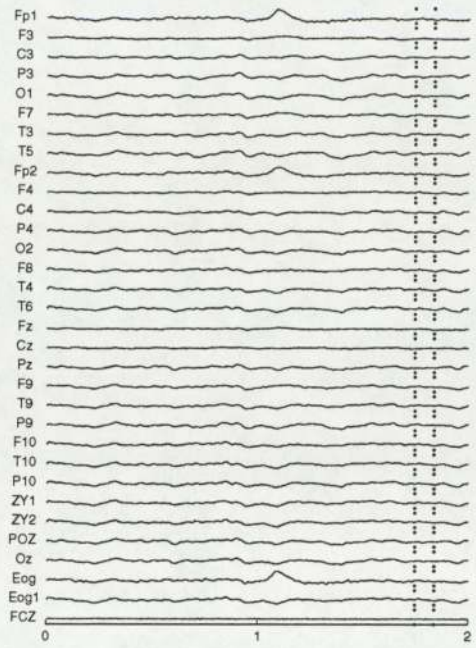
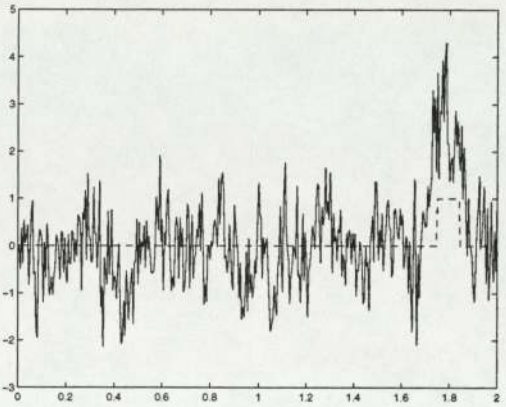


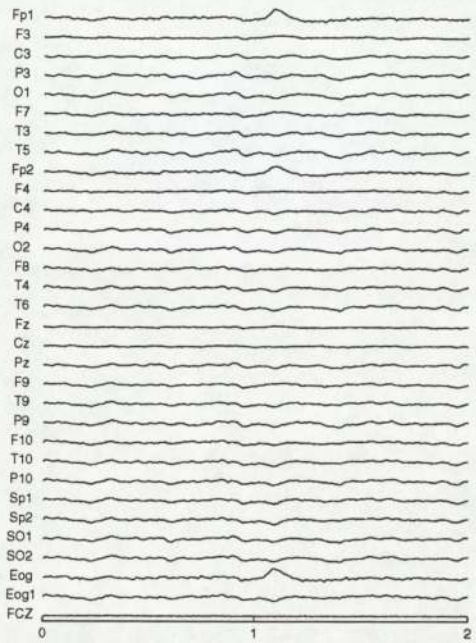
Figure 5.4: Output power of the CICA algorithm as the square wave reference (shown as the dashed line in Figure 5.2(d)) is shifted through time to create "false" references. A genuine artifact was extracted when the reference was non-zero between 200 and 250 samples (Figure 5.2) but there is no clear difference in output power between this section and other, "false" reference positions. It therefore seems that output power would not be a reliable indicator of "false" extractions.



(a) The **anomaly 3** epoch with the "false" reference period marked.



(b) The reference signal used (dashed line) and component extracted by the CICA algorithm.



(c) The epoch with the extracted component subtracted.

Figure 5.5: An example of the algorithm's behaviour when a "false" reference is given for artifact rejection. The same EEG epoch was used as for Figure 5.2 but the reference signal was moved in time. The reference signal is now non-zero for a period close to 1.75 s, where there is no artifact present. The algorithm extracted a component which is much noisier than the genuine extraction. Furthermore, when projected into the data space, the subtraction of this component barely affected the original epoch. This example is a promising result since the "false" reference has not caused corruption of the EEG epoch.

Chapter 6

Application of algorithm to real-world seizure onset analysis

In Chapter 4 the performance of the CICA algorithm at extracting simulated seizure waveform from real background EEG was tested. Since the desired source was known exactly, a large number of tests could be executed and the results automatically classified. In this chapter the algorithm is used in a preliminary application to extract seizure waveform from a set of real-world EEG recordings. Since obviously none of the underlying sources were known, automatic classification of the results was not possible. A quantitative assessment of the algorithm's performance could therefore only be obtained by the use of seizures already classified by an expert electroencephalographer. This process did not seem justified for this project, where time constraints permit only a preliminary study of the CICA algorithm in this application. Instead, therefore, a small set of seizures is used to demonstrate the CICA algorithm's performance. The practical problem of obtaining a suitable reference signal (that is, with suitable frequency and phase) is discussed. Finally, the behaviour of the algorithm when given a "false" reference is tested.

6.1 Generation of a reference signal from seizure waveform

Work described previously in this thesis has shown that a simple square wave reference is suitable for use with the CICA algorithm. In real-world situations, such as for the work described in this chapter, the frequency and phase of the desired underlying source are not known. For the examples shown here, it is assumed that the frequency and phase of the seizure waveform do not change during the time periods considered (typically less than 20 seconds). While this assumption often holds true (for example, seizure waveform is generally assumed to have relatively low frequencies) one of the examples in this chapter demonstrates how the morphology of the seizure waveform may change during a seizure (Figure 6.20). For our purposes this variation in morphology is not important unless it causes the seizure waveform to differ so much from the reference that the algorithm does not extract it.

Based on the above assumptions, therefore, the reference signal can be a square wave with frequency and phase based on a section of known seizure waveform. In the examples, this seizure waveform is a two-second epoch taken from a time period after the onset of the seizure, from an EEG channel where the seizure waveform is relatively uncorrupted by other sources or artifacts.

6.1.1 Selection of reference frequency

In the examples shown, a two-second section of seizure waveform from a channel which shows it clearly is displayed as a "spectroplot". This is a spectrogram showing a time-frequency estimate of the signal, with the temporal signal superimposed. (This "spectroplot" display was previously used in Chapter 4.) If the seizure waveform has a relatively high power in the chosen signal epoch, the approximate frequency can be seen from the spectroplot. This process of finding the fundamental frequency of the seizure waveform could be performed automatically, but the selection of a clear section of seizure waveform might still require manual intervention. (A seizure detection algorithm might however allow full automation of this process, but is beyond the scope of this project.)

6.1.2 Phase matching between reference and underlying source

Having found the approximate frequency of the seizure waveform, which will be used as the frequency of the square wave reference signal, the phase difference between the reference signal and the seizure waveform must be minimised. Correlation was used as a closeness measure between the reference signal and the algorithm output signal at each iteration for all the work described in this thesis, but the correlation between two similar signals is much lower if the phases of the signals do not match. The algorithm therefore produces inaccurate results if the reference does not have similar phase to the desired source, even if the two signals are morphologically similar. In the work involving extraction of simulated seizure waveform (Chapter 4) the phase of the seizure waveform was known and hence the phase of the reference signal could be matched to it. For the real-world seizures discussed in this chapter, none of the true underlying sources are known. A different method of matching the phase of the reference and underlying source is therefore required. (The phase of the seizure waveform could be observed manually at time periods where it is clearly visible. However, such manual intervention is undesirable and impractical for an automatic system.)

Our first solution to this problem was to perform a series of extractions with the CICA algorithm from the period of known seizure waveform, with references progressively shifted in time through a cycle. The known seizure waveform was a two-second epoch selected from an EEG channel where the seizure can be clearly seen. For example, the seizure shown in Figure 6.1 generalises to most of the EEG channels after about 7 seconds. The seizure waveform has a frequency of about 5 Hz, so a square wave reference of this frequency was used. A two second time period, between 13 and 15 seconds, was used to establish the phase of the seizure waveform since the seizure is fully generalised at this time. The square wave reference was shifted by one sample at a time, for the duration of one cycle. For each reference position, 200 executions of the CICA algorithm were performed. The final correlation between the algorithm output and the reference for each execution were stored and the mean of these values taken for each reference position. The standard deviation of the correlation values was used as an error bar. As shown in Figure 6.3, this method does not give very clear results since there appear to be two maxima of correlation. Another example of this method, shown in Figure 6.10, has no obvious maxima. These poor results may be due to the random initialisation of the algorithm's weight vector - a very large number of trials may be required to cover the possible initialisation space.

A simpler method is to find the correlation between the reference signal and the known seizure waveform directly, for a range of phase shifts. The results of this approach are shown in Figure 6.4. The maximum correlation indicates the best phase for the reference signal (Figure 6.5). This method of phase matching was used in the examples described below.

6.2 Examples of seizure onset analysis

6.2.1 Seizure 1

Seizure 1, shown in Figure 6.1, originates in the right temporal lobe; the earliest clear appearance of seizure waveform is on channel T10 from approximately t=3s. The nearby electrodes F10 and F8 also show some seizure waveform at this point. The seizure generalises at about t=7s since at this point the seizure waveform appears on most of the EEG channels. This point also seems to be the clinical onset of the seizure, since the high frequency muscle artifacts imply that the patient is moving. The aim of this work was to move backwards in time through the recording from this point, assessing the algorithm's ability to extract seizure waveform and finding the topographic maps of these extractions.

The frequency and phase of the reference signal were determined, as described in the previous section, from a selected epoch of clear seizure waveform. The seizure waveform was taken from channel T6, between t=13 and t=15 seconds. A spectroplot of this data showed that the fundamental frequency of the seizure waveform was approximately 6 Hz (Figure 6.2), while the optimum phase of the reference was determined from its correlation, at different phase shifts, with the seizure waveform (Figure 6.4). The best reference signal for this seizure was thereby established (Figure 6.5).

Using this reference signal, the CICA algorithm was used to extract the closest underlying source from a series of two-second epochs, moving backwards in time from the "known" seizure at approximately t=13 s, in 1 second decrements. One hundred executions of the CICA algorithm were performed for each epoch. The mean and standard deviation of the 100 dewhitened output signal power values for each time point were calculated and are shown in Figure 6.6. It is reasonable to assume that there will be more "correct" extractions of seizure waveform as the epoch studied is moved closer to the onset of generalised seizure, at around t=7s. Ideally, if a higher output power is a good indicator of such "correct" extractions, this graph should show an increasing trend up to this point. In fact this graph does not show any obvious trend, although this may be because the number of data points shown is too small, or the 100 trials of CICA for each time point may not give adequate coverage of the random initial weight vector space. Alternatively, the algorithm may in fact be extracting genuine seizure waveform over this period; for this seizure the time period studied before the seizure onset is relatively short.

Some examples of outputs of the CICA algorithm are shown in Figure 6.7. These examples (and the other example outputs shown in this chapter) have been selected to be typical of the outputs of the CICA algorithm. The outputs for this seizure recording appear to contain noisy seizure waveform back until the start of the recording (t=0s). The topographic maps for each extraction were generated by interpolating the values of the mixing vector $\mathbf{a}$, which correspond to electrode positions on the spherical head model used. Bearing in mind that the polarity is indeterminate (for example, the topographies in Figures 6.7(e) and (f) are very similar despite the reversal of extremes of white and black), the foci of the extracted components are clearly in the right temporal lobe for Figures 6.7(e) to (h). This matches the manual interpretation of the seizure focus since, for these epochs close to the generalisation of the seizure, the assumption of spatial stationarity of the seizure is more likely to hold and the power of the seizure waveform is sufficient to avoid "false" extractions of other similar sources. The results for epochs earlier in time (Figures 6.7(a) to (d)) show less clearly focused topographies. This may be because the seizure itself has changed over this time (the spatial focus, morphology or frequency may be different), or the algorithm's extractions may contain a greater mixture of other

sources, leading to less focused topographies.

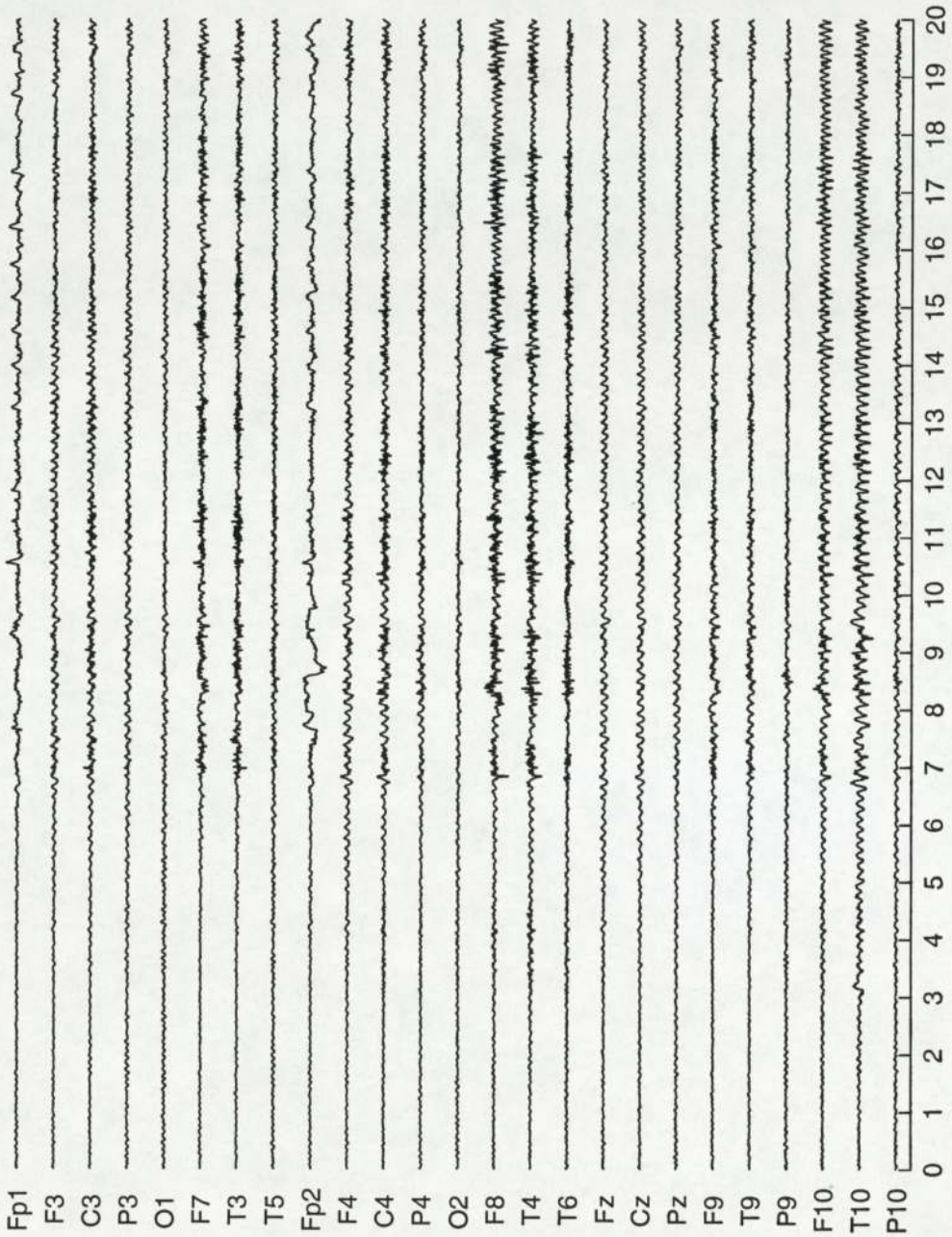


Figure 6.1: Seizure 1, an EEG recording sampled at 200 Hz, with 25 electrodes in the standard positions. A referential montage was used with FCz as the reference electrode. The visual onset of the seizure is close to t=7s, where some muscle artifact can be seen. For the previous seven seconds, a growing seizure waveform can be seen on some channels, such as T10. The seizure appears to originate in the right temporal lobe, since the seizure is most clearly seen in channels from this region.

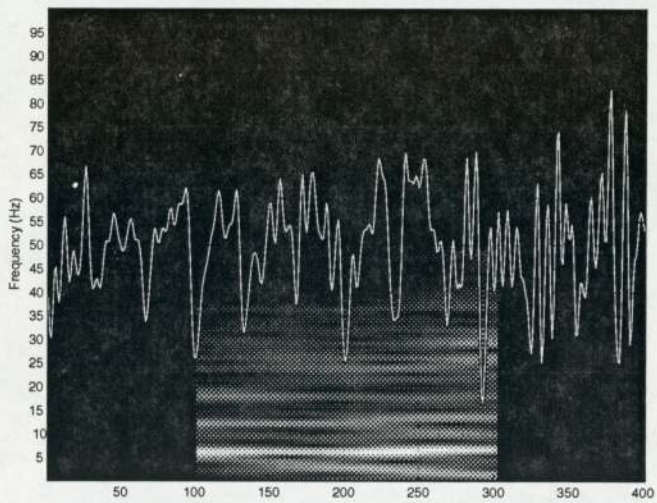


Figure 6.2: Seizure 1: Spectroplot of channel T6 between t=13 and t=15 seconds. The fundamental frequency of the seizure waveform is approximately 6 Hz.

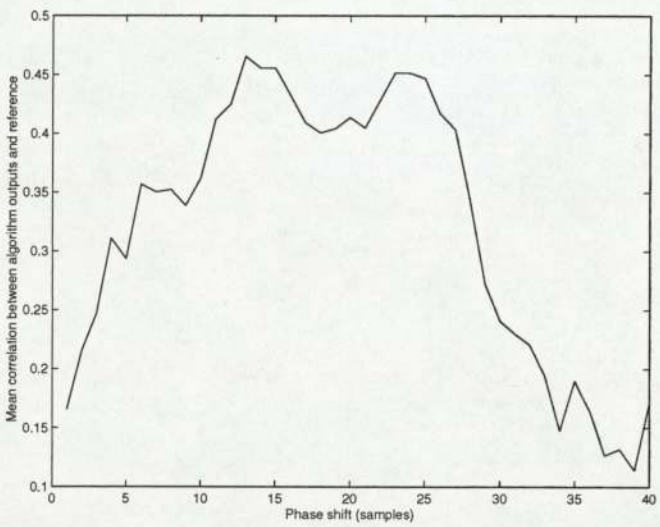


Figure 6.3: Seizure 1: Variation of mean correlation between algorithm outputs and square wave reference, as the reference is shifted through time. Data from Seizure 1 between 13 and 15 seconds was used as the input mixture and 200 executions of the CICA algorithm were performed for each phase shift. The expectation was that this would allow the phase of the reference to be matched to that of the underlying seizure waveform. However, identification of the correct peak correlation is not straightforward in this case. In general this method did not perform as well as a simple maximisation of correlation between the reference and selected seizure waveform (Figure 6.4).

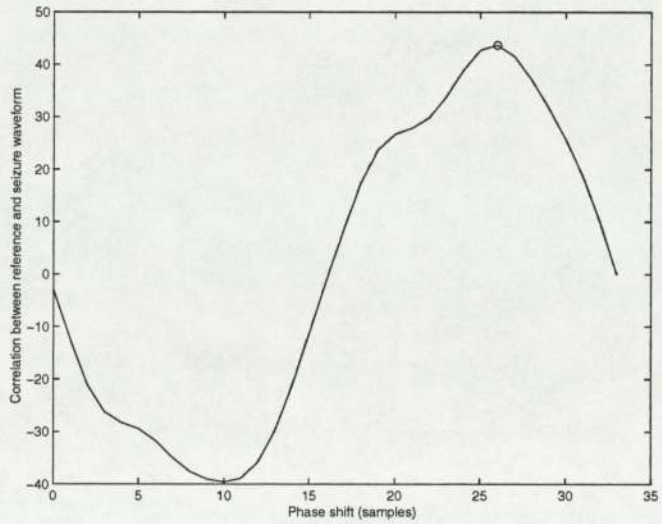


Figure 6.4: Seizure 1: Correlation between 6 Hz square wave reference signal and the selected seizure waveform (as shown in Figure 6.2) for different phase shifts. The peak correlation (at an offset of 26 samples) shows the best phase for the reference signal.

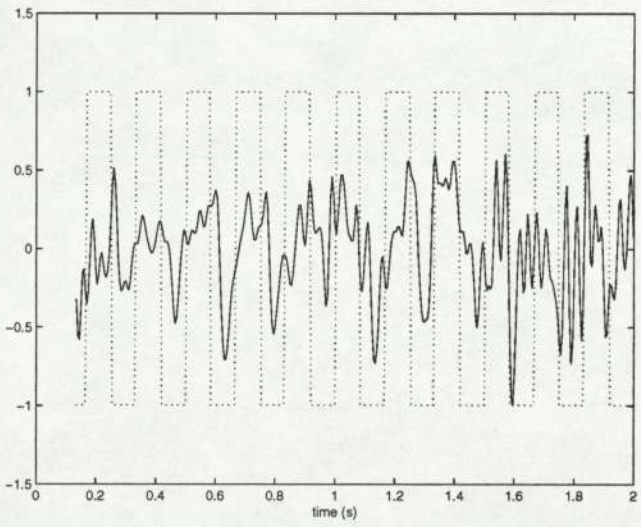


Figure 6.5: Seizure 1: Reference signal (dotted) matched in frequency and phase to the selected seizure waveform (solid line), from Figure 6.2. This reference signal was then used with the CICA algorithm at different two-second EEG epochs before the seizure onset, to extract seizure waveform.

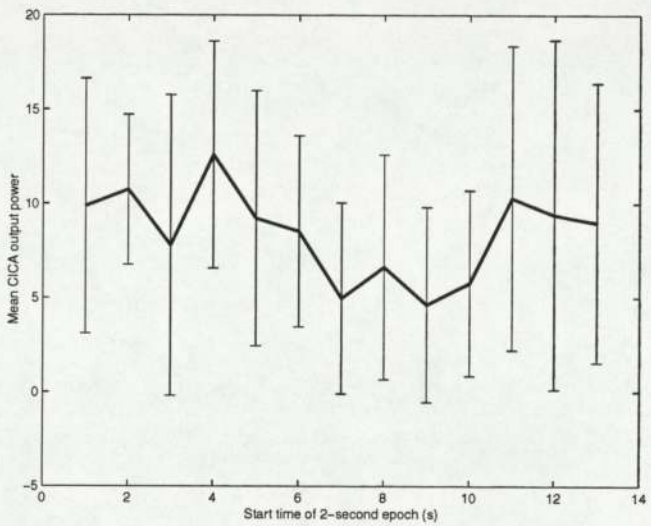
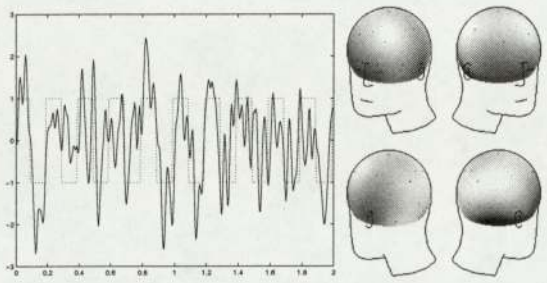
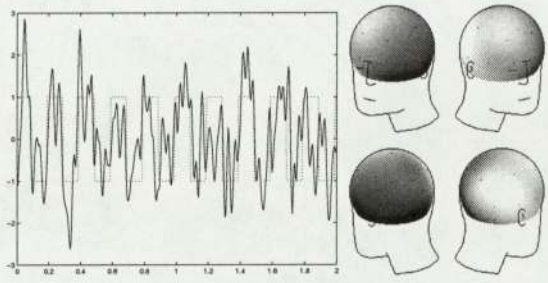


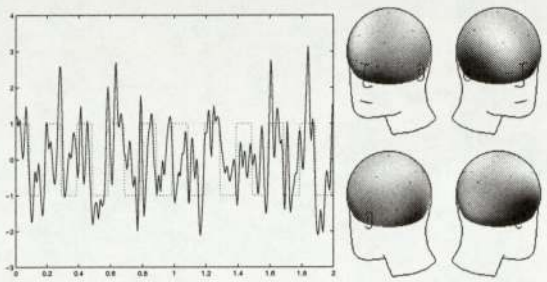
Figure 6.6: Seizure 1: Mean output power for a range of two-second epochs, with error bars of one standard deviation. One might expect the power to increase for epochs closer to the generalised seizure (that is, larger time values), but this graph shows no particular trend.



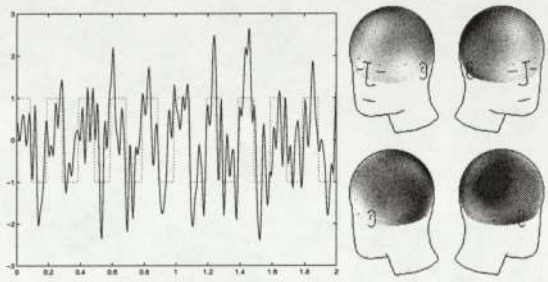
(a) Start point: 0 s



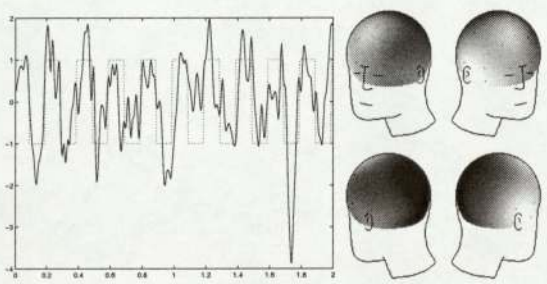
(b) Start point: 1 s



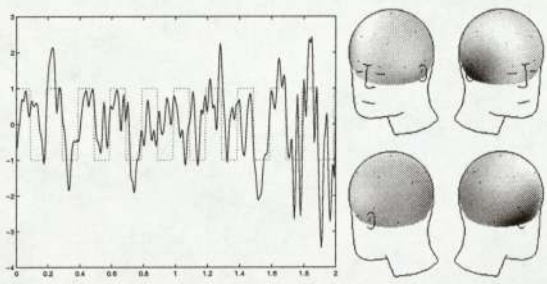
(c) Start point: 2 s



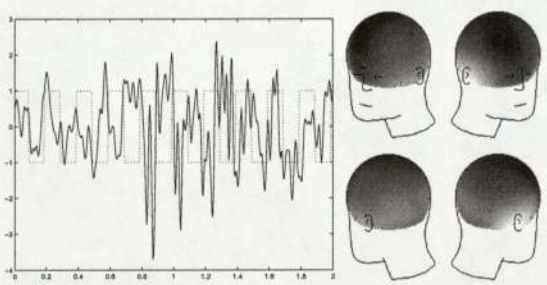
(d) Start point: 3 s



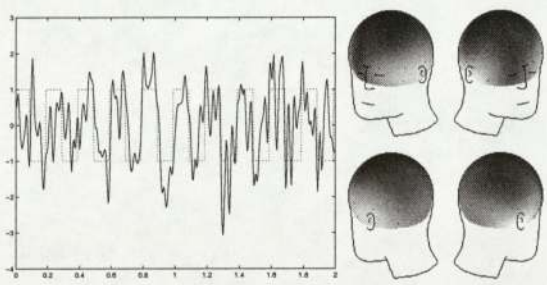
(e) Start point: 4 s



(f) Start point: 5 s



(g) Start point: 6 s



(h) Start point: 7 s

Figure 6.7: Seizure 1: Results of CICA algorithm applied to two-second epochs from Seizure 1 (Figure 6.1) at different points. The algorithm output is shown superimposed on the reference signal (dotted line). The topographic map of each output is also shown. In general, the extracted seizure waveform becomes more noisy as the epoch is moved backwards in time from the generalisation of the seizure. The topographic maps show a general seizure focus in the right temporal lobe.

6.2.2 Seizure 2

Seizure 2, like Seizure 1, is first visually identifiable in the right temporal lobe (Figure 6.8). The seizure waveforms which are first clearly visible at t=21s are strongest on channels F10, T10, F8 and T4, which are located close to this region of the brain. The EEG signals prior to t=21s are relatively "quiet", except for an artifact (perhaps an eye blink or movement) between t=3s and t=6s.

A square wave reference signal was generated following the same procedure as for Seizure 1. A section of clear seizure waveform was first isolated (from channel F10 between t=24s and t=26s) and a spectroplot of this signal shows a fundamental frequency of approximately 4 Hz. The problem of matching the phase of the reference signal to that of the seizure waveform was addressed by the two methods discussed earlier. The method based on the CICA algorithm output signal correlation with the reference for a series of extractions with different reference phase shifts was performed (Figure 6.10) but gave very poor results - there is no clear peak in the mean correlation. This method was not therefore used for the other experiments described below. The simpler method based on the correlation of the reference signal with the selected seizure waveform (Figure 6.11) gave a much clearer indication of the optimum phase for the reference signal. The final reference signal with the chosen frequency and phase is shown, superimposed on the selected seizure waveform, in Figure 6.12.

Using this reference signal, the CICA algorithm was run on a series of two-second epochs in time periods prior to the visual onset of the seizure. For each time position, 100 executions of the CICA algorithm were performed with random initial weight vectors. The mean and standard deviation of the output powers at different time positions are shown in Figure 6.13. As might be expected, there is a general trend for the output power to decrease as the EEG time window is moved backwards in time from the visual seizure onset. The seizure waveform is likely to decrease in power until it is no longer present. There is, however, a large variance in output power (as shown by the wide error bars) which makes it problematic to use by itself as a measure of "false" extractions. This is discussed further in the trials with a "false" reference, later on in this chapter.

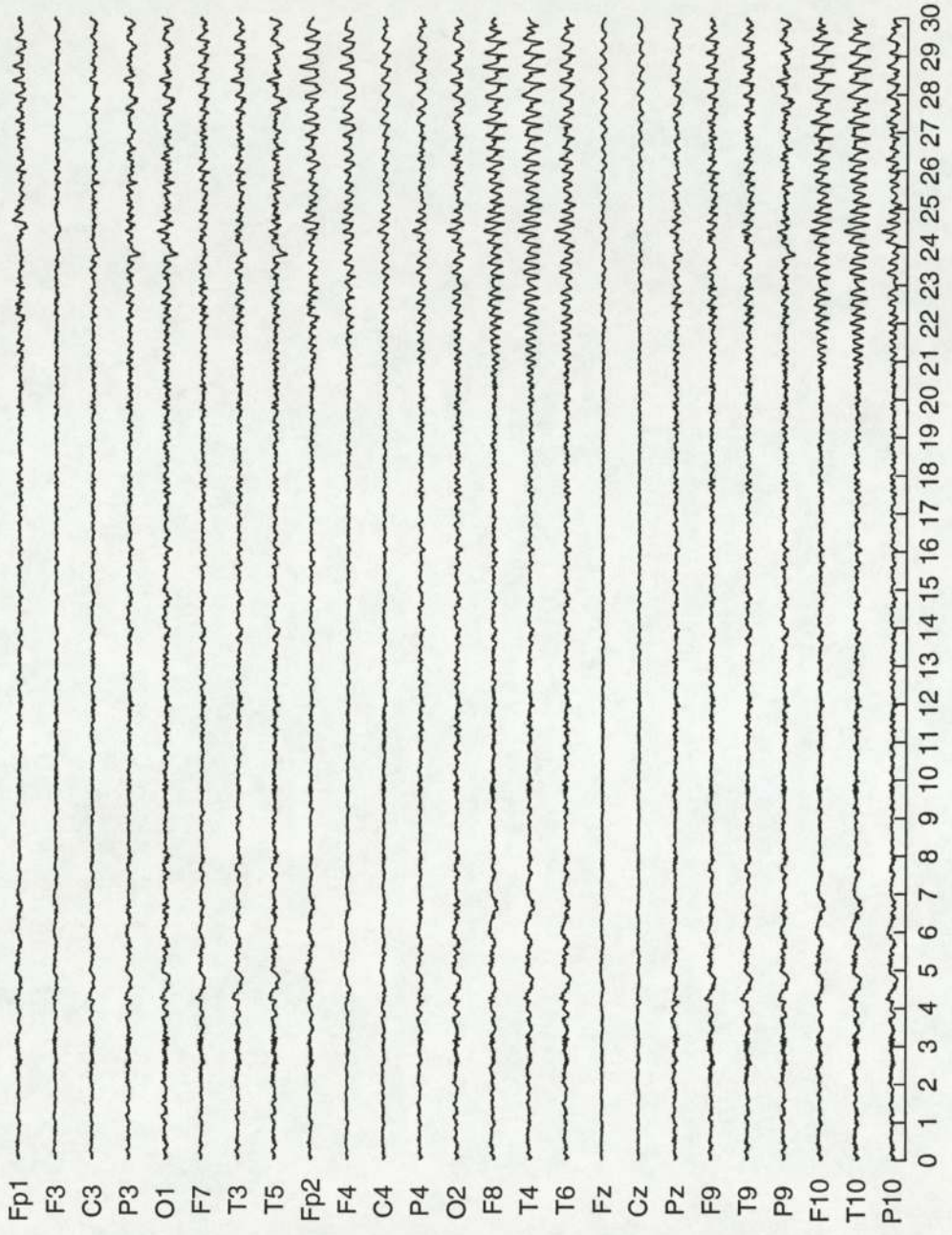


Figure 6.8: Seizure 2: 30 second EEG epoch with developing seizure, sampled at 200 Hz with 25 electrodes in the standard positions. A referential montage was used with FCz as the reference electrode. The visible onset of the seizure is at around t=21s and is strongest in channels from the right side of the head, such as F8, T4, F10 and T10.

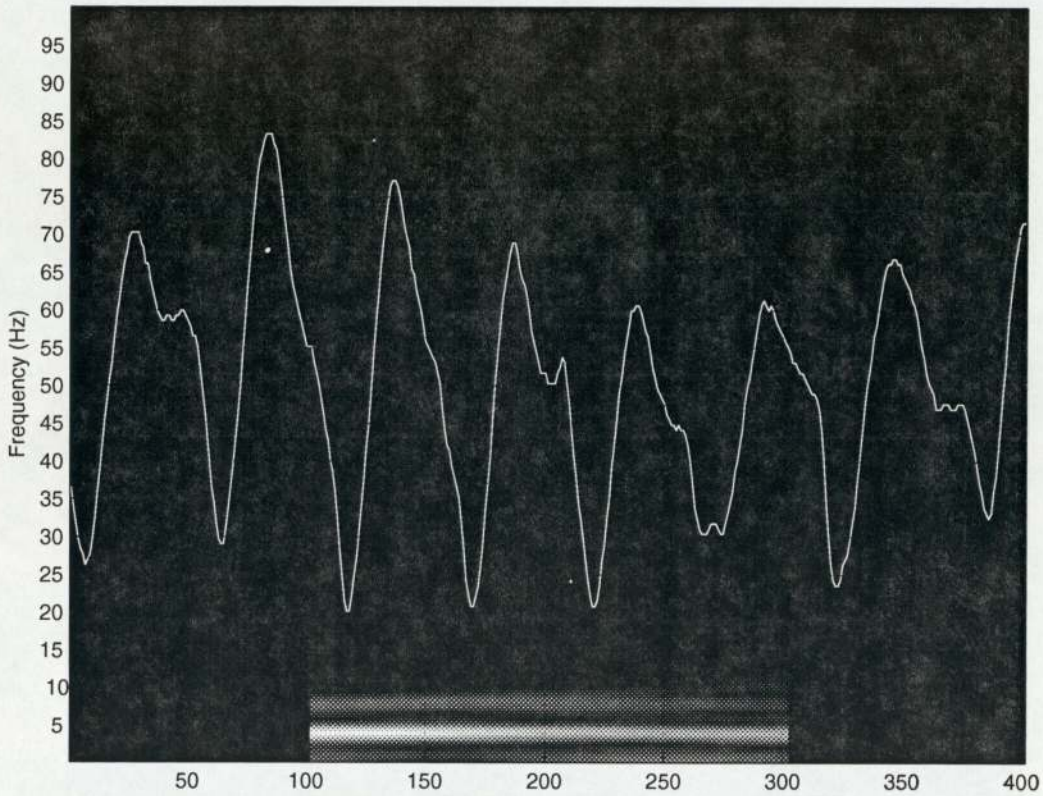


Figure 6.9: Seizure 2: Spectroplot of channel F10 between t=24s and t=26s showing prominent seizure waveform at approximately 4 Hz. This frequency was taken as the reference frequency for trials with this seizure.

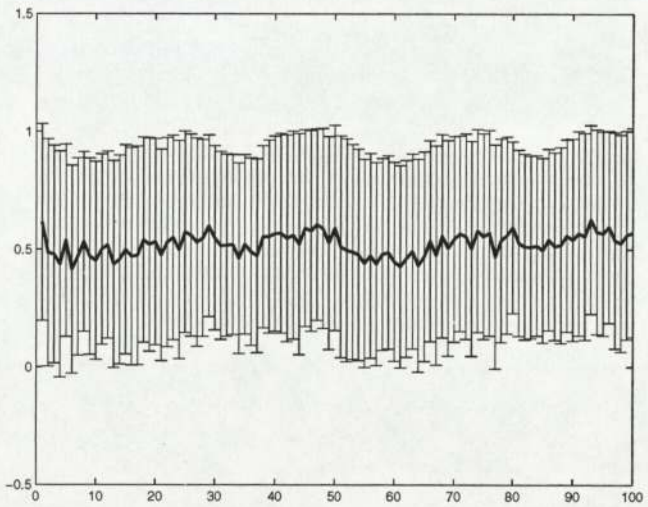


Figure 6.10: Seizure 2: Mean correlation of CICA algorithm output with reference signal for a reference signal at different phase shifts. One hundred executions of the algorithm were performed for each data point, in attempting to establish the optimum phase match between the reference signal and the underlying seizure waveform. This method proved unsuccessful for these trials, since no clear peak in correlation was observed. Instead, the correlation between the reference signal and a section of selected seizure waveform was used (Figure 6.11).

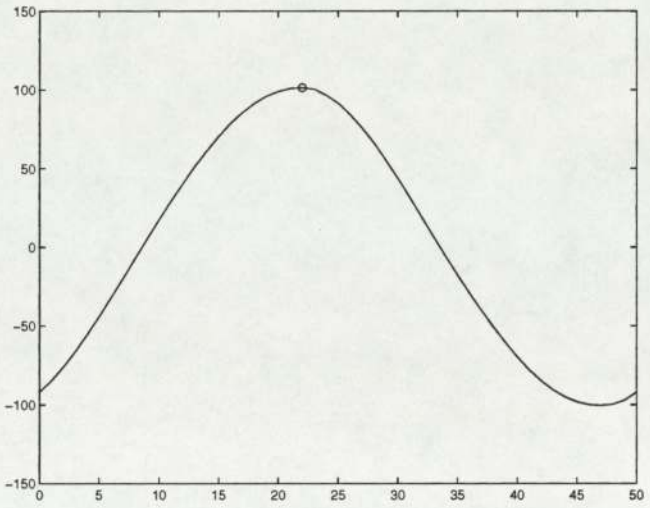


Figure 6.11: Seizure 2: Correlation between selected seizure waveform and a 4 Hz square wave reference signal which is swept through one cycle. Both the seizure waveform and the reference signal were first normalised. The maximum correlation, for an offset of 22 samples, is shown as a circle.

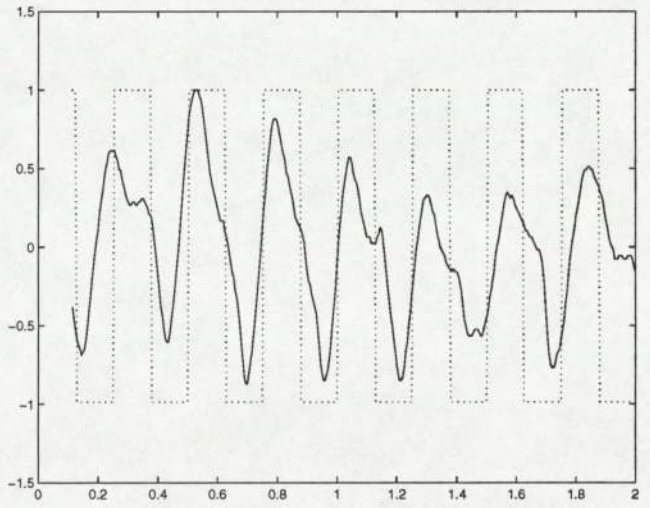


Figure 6.12: Seizure 2: The best phase for the 4 Hz square wave reference. This phase corresponds to the maximum correlation in Figure 6.11. (The reference signal and seizure waveform were both normalised for this purpose.)

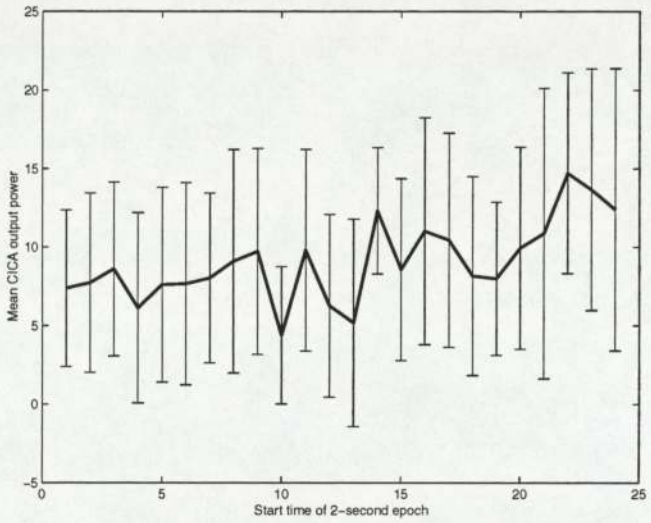
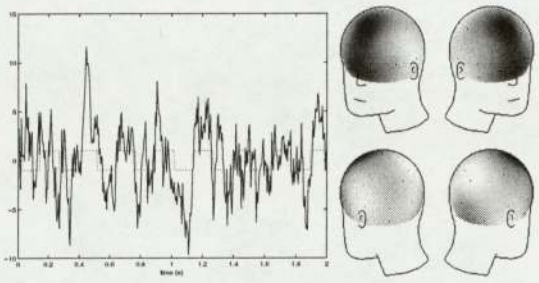
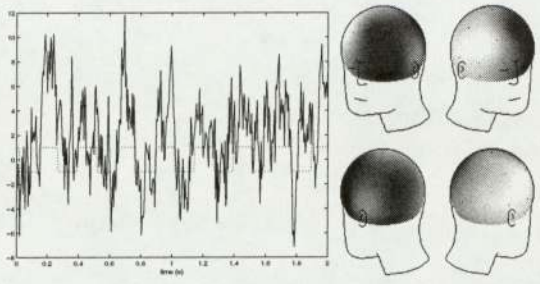


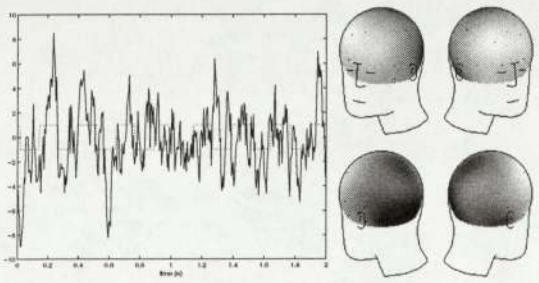
Figure 6.13: Seizure 2: Mean output power for a range of epochs, with error bars of one standard deviation. The CICA algorithm was run 100 times for each epoch. The times correspond to those on Figure 6.8. There is a general trend for the output power to fall as the epoch studied moves further back in time from the generalisation of the seizure.



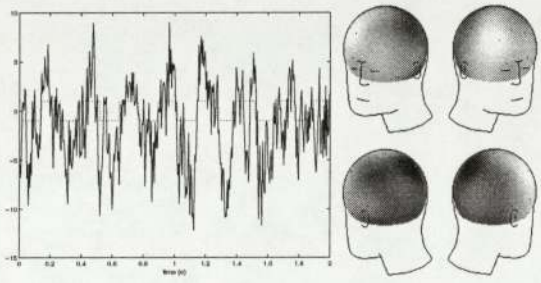
(a) Start point: 2 s



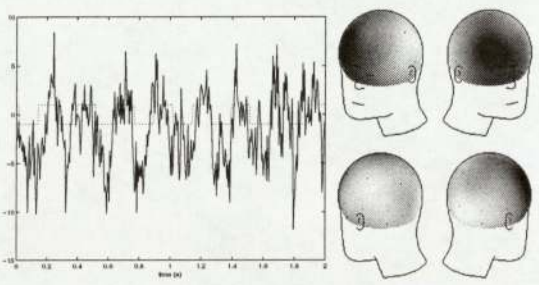
(b) Start point: 6 s



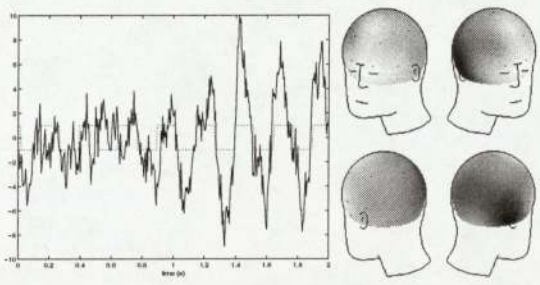
(c) Start point: 10 s



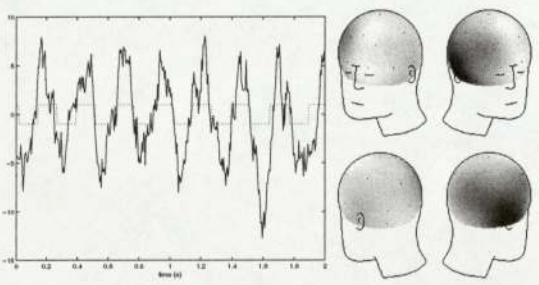
(d) Start point: 14 s



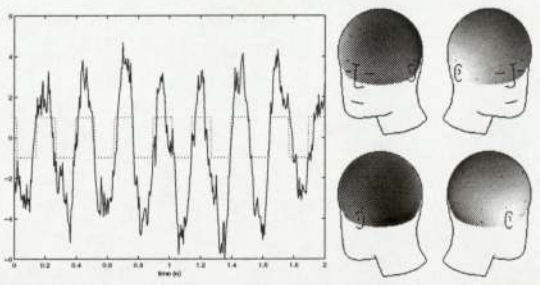
(e) Start point: 18 s



(f) Start point: 20 s



(g) Start point: 22 s



(h) Start point: 24 s

Figure 6.14: Seizure 2: Example outputs of the CICA algorithm applied to different time periods. The start point times correspond to the time axis of Figure 6.8. The topographic map of each example is also shown. In general, the extracted seizure waveform becomes noisier as the epoch studied is moved backwards in time from the visually-identified generalisation of the seizure, at approximately 24 seconds.

6.2.3 Seizure 3

Seizure 3 is very similar in morphology and spatial focus to Seizure 2; in fact the two seizures were recorded from the same patient on consecutive days. There is again a focus in the right temporal lobe when the seizure becomes clearly visible at about t=17s (Figure 6.15). The seizure generalises to almost all the EEG channels at t=20s.

The reference signal for trials with Seizure 3 was generated using the same process as for Seizure 2. A section of clear seizure waveform was taken from channel T4 between t=20s and t=22s. As might be expected, the seizure waveform has the same frequency as for Seizure 2, that is, approximately 4 Hz. The phase of the square wave reference was matched to the seizure waveform using the process of maximising correlation between the two signals (Figure 6.16). The signals are shown with optimum phase shift in Figure 6.17.

The CICA algorithm was run 100 times, using this reference, for each window position between t=1s and t=20s. The mean and standard deviation of the dewhitened output signal powers were taken, as for the previous seizures, and are shown in Figure 6.18. Here there is no clear trend shown in the graph, perhaps due to the low number of data points available.

Some example outputs are shown in Figure 6.19. An extraction from the section of generalised seizure used for the reference generation gives, as might be expected, a clean seizure waveform as output (Figure 6.19(f)). The topography for this extraction shows the focus of the seizure in the right temporal lobe. As the two-second window is moved to positions further back in time, the extracted seizure waveform becomes noisier and the topography less focused. The result shown in Figure 6.19(c) has a different topography, but this may be due to the artifact visible near this time period (t=14s) in the raw EEG (Figure 6.15). This artifact, which resembles an eye blink or movement, appears in the CICA algorithm output and causes the topography to show a focus over the eyes.

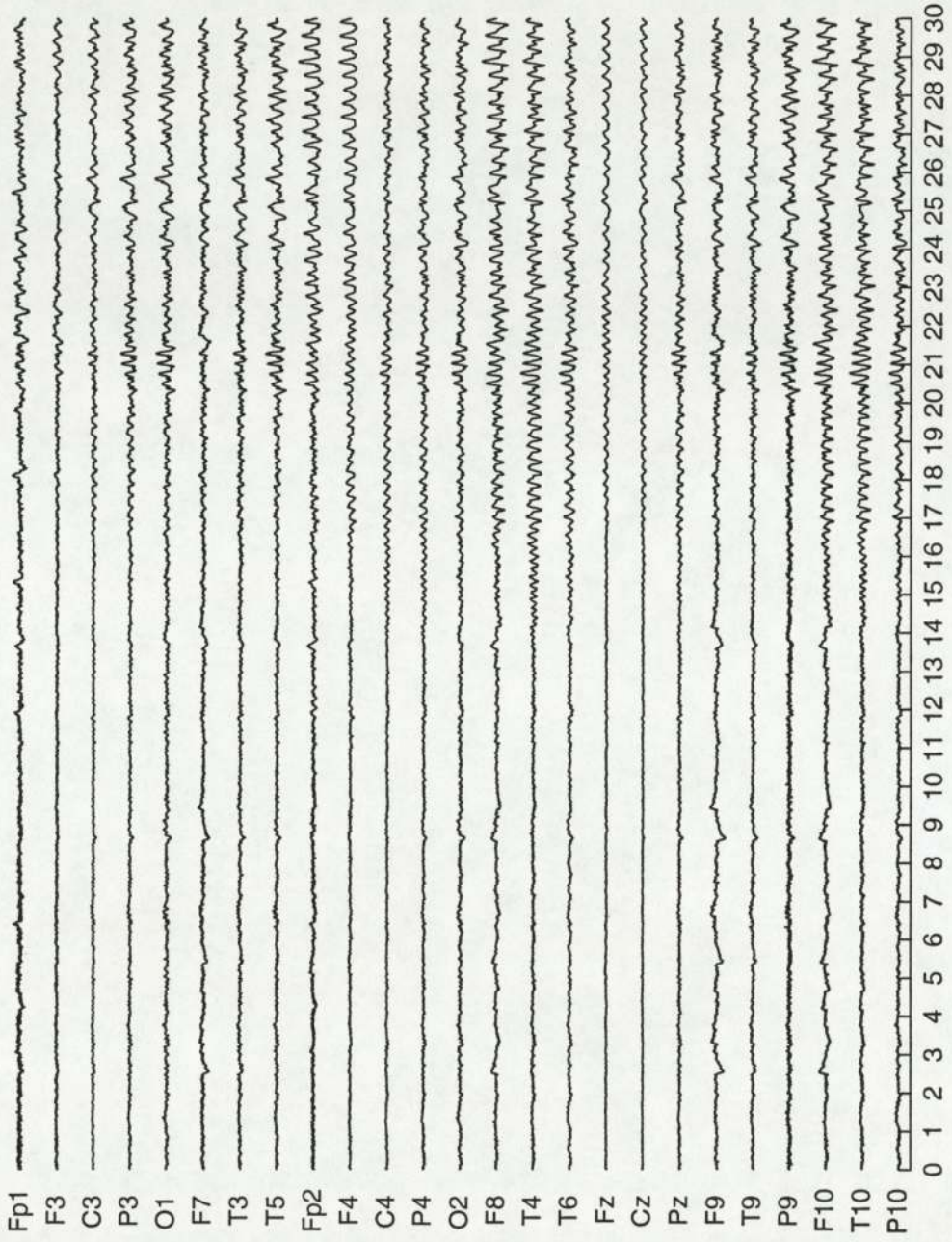


Figure 6.15: Seizure 3: 30 second EEG epoch with developing seizure, sampled at 200 Hz with 25 electrodes in the standard positions. A referential montage was used with FCz as the reference electrode. The visible onset of the seizure is at around t=17s and is strongest in channels from the right side of the head, such as F8, T4, F10 and T10. The morphology and frequency of the seizure waveform changes at about t=26s. This epoch was recorded from the same patient as Seizure 2, on the following day.

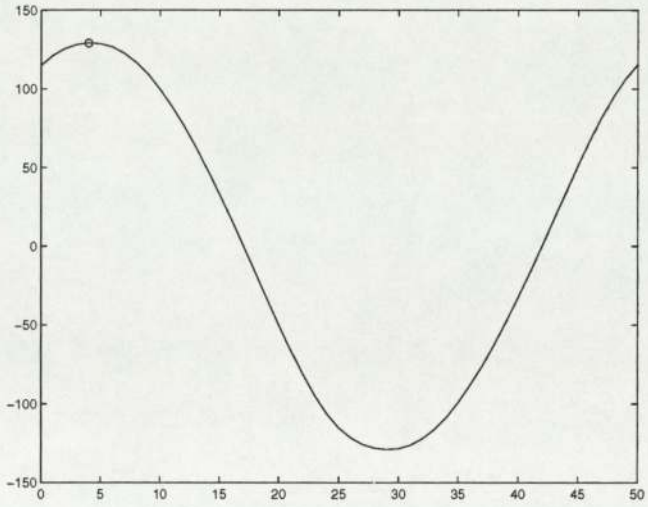


Figure 6.16: Seizure 3: Correlation between a period of selected seizure waveform and a 4 Hz square wave reference signal which is swept through one cycle. Both the seizure waveform and the reference signal were first normalised. The maximum correlation, for an offset of 4 samples, is shown as a circle.

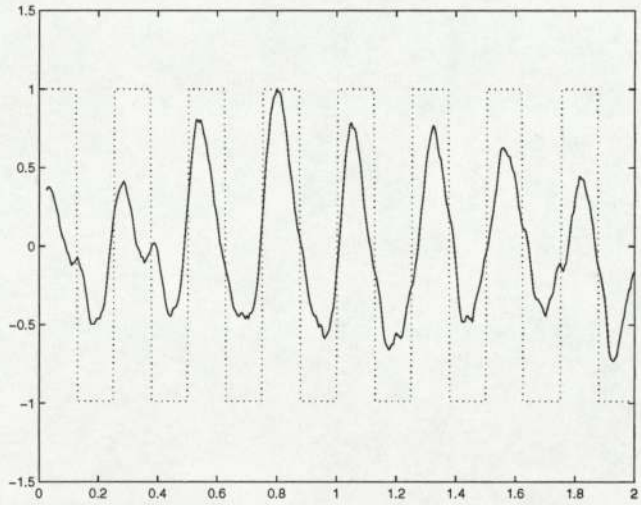


Figure 6.17: Seizure 3: The best phase for the 4 Hz square wave reference. This phase corresponds to the maximum correlation in Figure 6.16. (The reference signal and seizure waveform were both normalised for this purpose.)

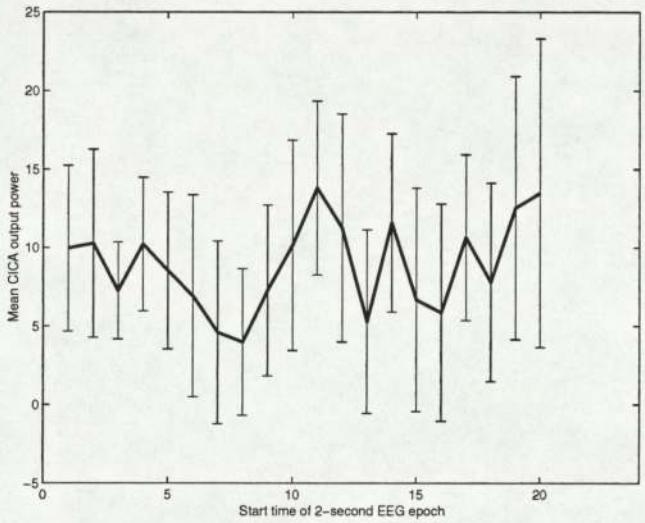
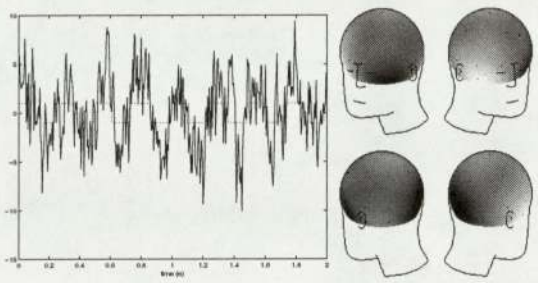
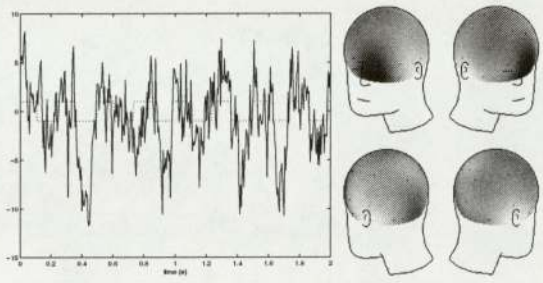


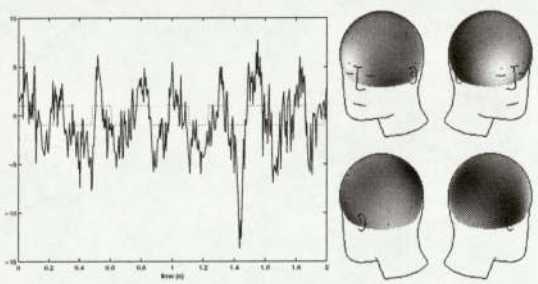
Figure 6.18: Seizure 3: Mean output power for a range of epochs, with error bars of one standard deviation. The CICA algorithm was run 100 times for each epoch. The times correspond to those on Figure 6.15.



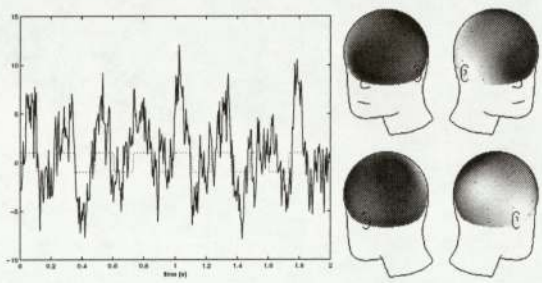
(a) Start point: 2 s



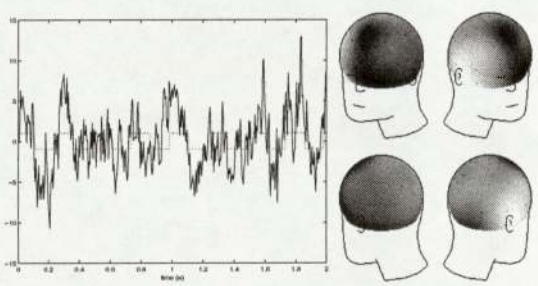
(b) Start point: 6 s



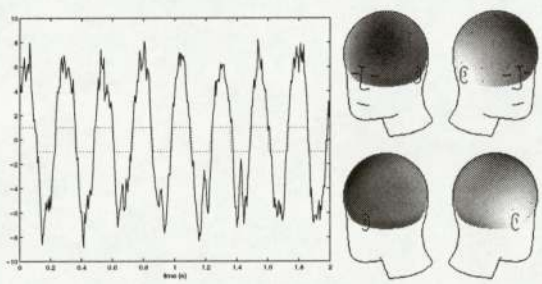
(c) Start point: 14 s



(d) Start point: 16 s



(e) Start point: 18 s



(f) Start point: 20 s

Figure 6.19: Seizure 3: Example outputs of the CICA algorithm applied to different time periods. The start point times correspond to the time axis of Figure 6.15. The topographic map of each example is also shown. In general, the extracted seizure waveform becomes noisier as the epoch studied is moved backwards in time from the visually-identified generalisation of the seizure, at approximately 20 seconds.

6.2.4 Seizure 4

Seizure 4 (Figure 6.20) has a focus in the left temporal lobe when seizure waveforms are first visible in the raw EEG at t=7s. The seizure appears quite suddenly in the raw data, unlike in the seizures previously described where there was some gradual increasing in amplitude at the onset of the seizure. The waveforms are most prominent on channels close to this part of the brain, such as F9, T9, P9 and F7. At t=11s the seizure generalises to most of the EEG channels and at this point the morphology of the seizure waveforms changes, including a decrease in frequency. This change provided an opportunity to test the CICA algorithm in a situation where the assumption of a stationary morphology of seizure waveforms did not hold.

The reference signal was therefore taken from the period of slower seizure waveform. A section of seizure waveform was taken from channel F8 between t=11s and t=13s and was used to provide the frequency (Figure 6.21) and phase (Figure 6.22). The square wave reference generated in this way is shown, superimposed on the selected seizure waveform, in Figure 6.23.

With this reference, the CICA algorithm was run on a two-second window which was positioned at different times prior to the visible seizure onset. For each window, the CICA algorithm was executed 100 times and the mean and standard deviation of the output powers calculated (Figure 6.24). The alternating nature of this graph suggests that the phase matching might not have been ideal, or the selected seizure waveform was not representative of the whole seizure. A better alignment with the underlying seizure waveform seems to have been obtained with even-numbered start times.

The example outputs shown in Figure 6.25 show the difficulty of seizure onset analysis with a suddenly-starting seizure such as this. Figure 6.25(f) shows a relatively good seizure waveform extraction, with a topography resembling the left temporal lobe focus suggested by the manual analysis above. Figure 6.25(d) shows a similar result. (The reversal of polarity of the topography is not significant.) Earlier window start points gave results with similar topographies, though with noisy seizure waveform.

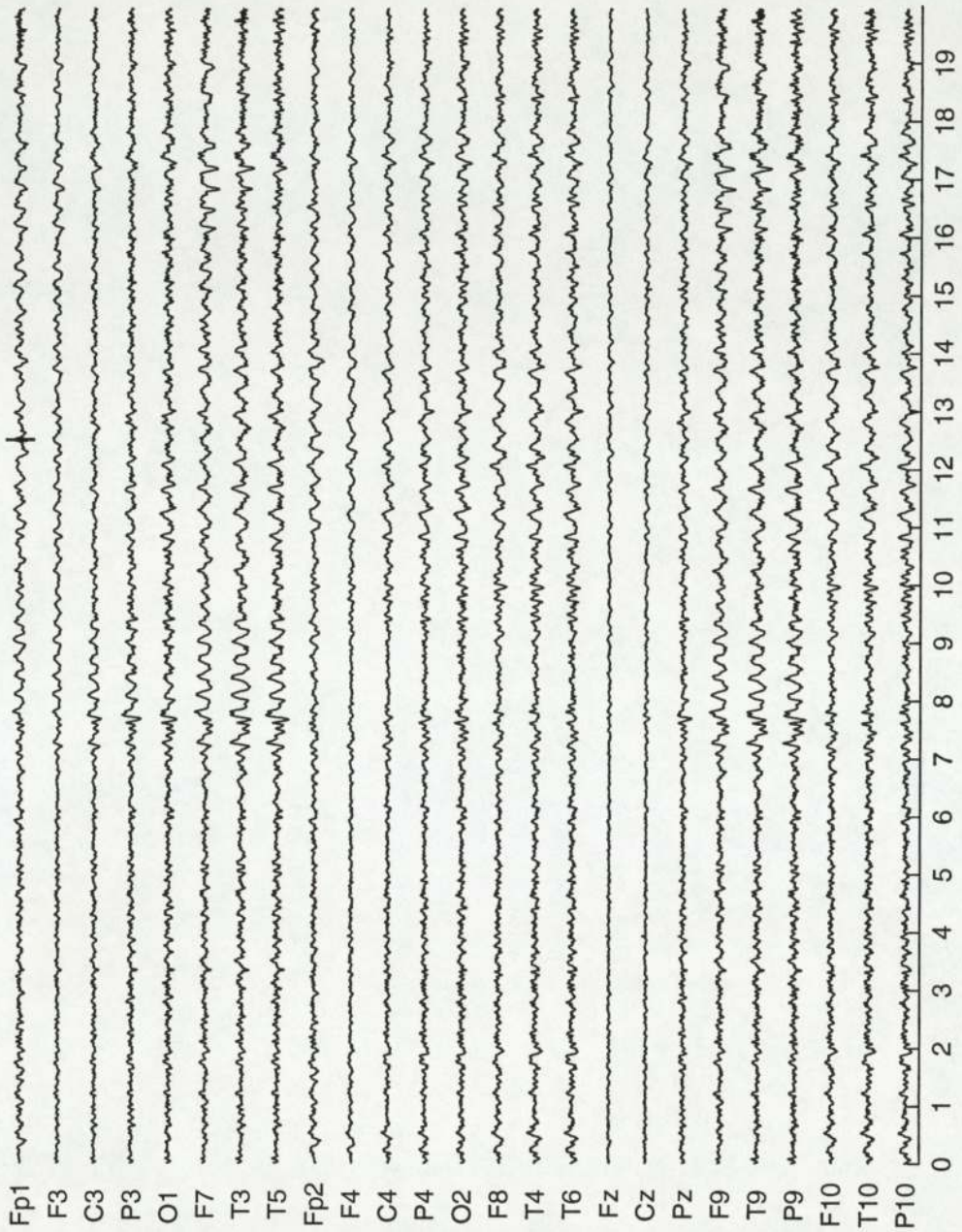


Figure 6.20: Seizure 4: 20 second EEG epoch with a seizure which develops suddenly. The morphology of the seizure waveforms changes at about t = 11 s. This data was sampled at 200 Hz with 25 electrodes in the standard positions and using a referential montage with FCz as the reference electrode. The visible onset of the seizure is at around t=7s and is strongest in channels from the left side of the head, such as F9, T9 and P9.

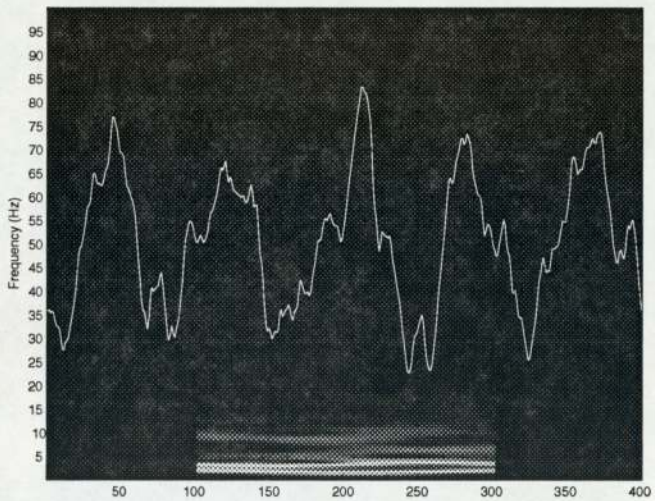


Figure 6.21: Spectroplot of channel F8 of Seizure 4, between 11 and 13 seconds. The seizure waveform is prominent and has a fundamental frequency of approximately 3 Hz. This was used as the frequency of the reference signal.

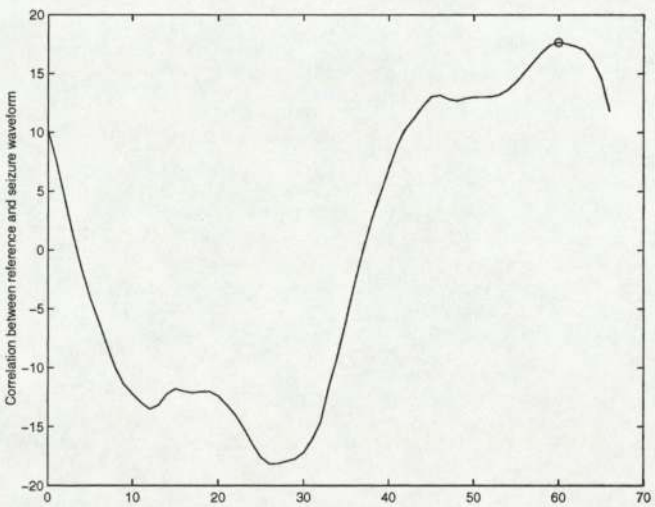


Figure 6.22: Seizure 4: Correlation between selected seizure waveform and a 3 Hz square wave reference signal which is swept through one cycle. Both the seizure waveform and the reference signal were first normalised. The maximum correlation, for an offset of 60 samples, is shown as a circle.

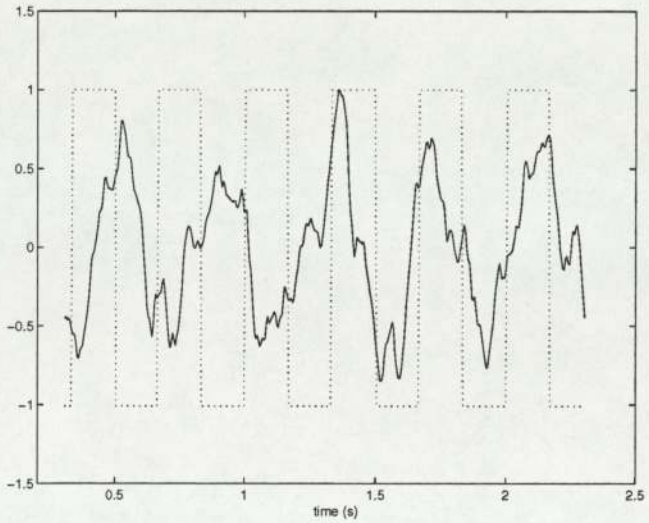


Figure 6.23: Seizure 4: The best phase for the 3 Hz square wave reference. This phase corresponds to the maximum correlation in Figure 6.22. (The reference signal and seizure waveform were both normalised for this purpose.)

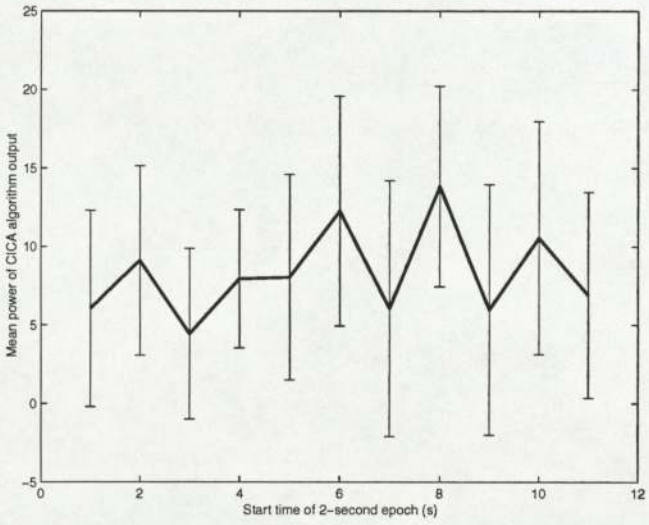
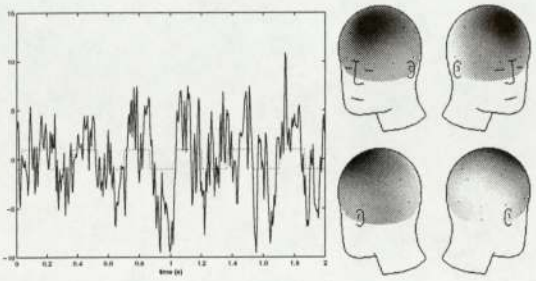
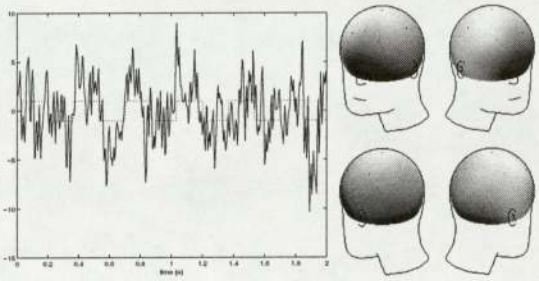


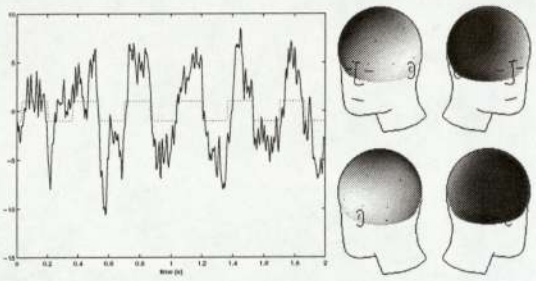
Figure 6.24: Seizure 4: Mean output power for a range of epochs, with error bars of one standard deviation. The CICA algorithm was run 100 times for each epoch. The times correspond to those on Figure 6.20.



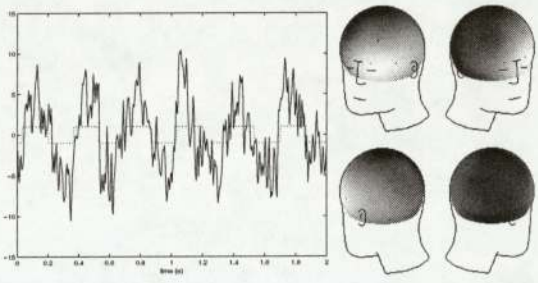
(a) Start point: 1 s



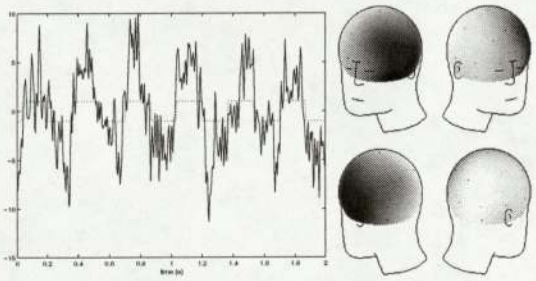
(b) Start point: 4 s



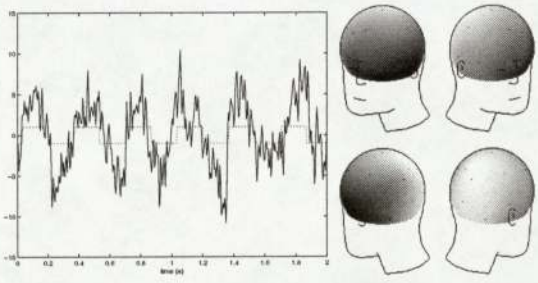
(c) Start point: 7 s



(d) Start point: 8 s



(e) Start point: 9 s



(f) Start point: 10 s

Figure 6.25: Seizure 4: Example outputs of the CICA algorithm applied to different time periods. The start point times correspond to the time axis of Figure 6.20. The topographic map of each example is also shown.

6.2.5 Seizure 5

Seizure 5, for which the raw EEG recording is shown in Figure 6.26, is from the same patient as Seizure 4. This seizure, however, is much slower to develop than the one previously discussed. It is difficult to identify the first appearance of seizure waveform but the seizure seems to generalise beyond the electrodes close to the left temporal lobe (such as T9, F9, P9 and F7) at about t=11s. There are seven eye blink artifacts in the recording which are most prominent from the electrodes close to the eyes (Fp1 and Fp2).

The reference signal was based on a section of seizure waveform taken from channel F7 between t=16s and t=18s. The fundamental frequency of this waveform is approximately 5 Hz (Figure 6.27) and the phase of the reference was established by correlation with the seizure waveform (Figure 6.28), as described previously. The square wave reference signal chosen is shown, superimposed on the selected seizure waveform, in Figure 6.29.

The results of running the CICA algorithm on a time window placed at different positions before the seizure (Figure 6.30) show a clear upward trend. This is due to the very gradual onset of this seizure compared to the seizures previously described. The eye blink artifacts do not appear to have impeded the algorithm, since they are of short duration and only appear on a few channels.

Some example extractions from this seizure recording are shown in Figure 6.31. Although the majority of these extracted signals resemble noisy seizure waveform, the topographies show foci over the eyes. This may be due to the eye blink artifacts, if they are not sufficiently separated from the extracted signals. The seizure may not have a particularly focused topography - the seizure waveform appears on a large number of channels - and so the focus of other signals may dominate the topographies shown. The software which plots the topographic maps rescales the topographies to cover the full range of shades of grey. Hence a reasonably uniform mixing vector for the seizure waveform could be noticeably biased by an artifact's spatial map, if the algorithm did not fully separate the seizure and artifact signals.

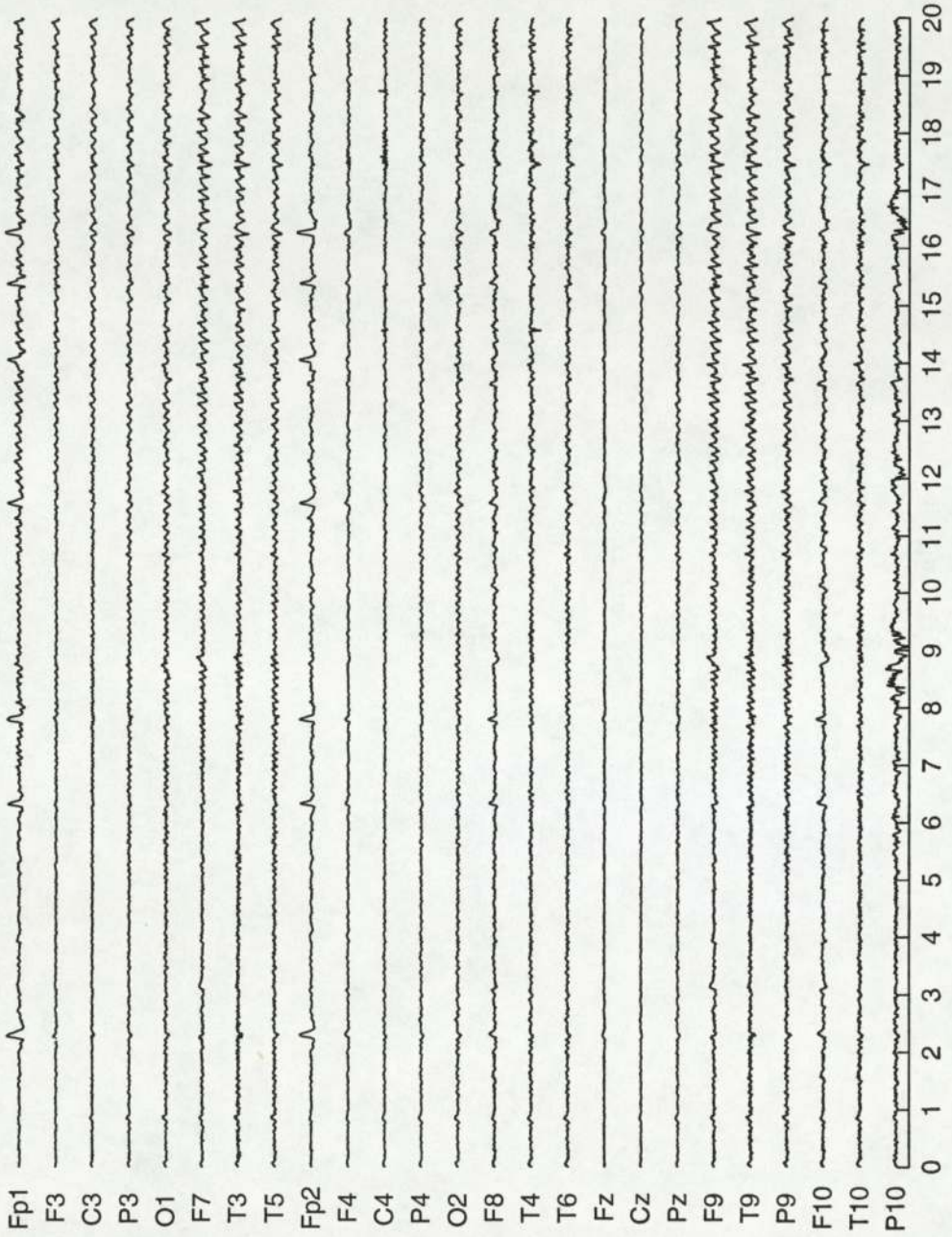


Figure 6.26: Seizure 5: 20 second EEG epoch with a seizure, sampled at 200 Hz with 25 electrodes in the standard positions. A referential montage was used with FCz as the reference electrode. Seizure waveforms of growing amplitude can be seen for most of the epoch prior to the seizure, but the seizure appears to generalise at around t=11s and first appears in channels close to the left temporal lobe (such as T9, F9, P9 and F7). Seven eye blink artifacts are visible during this epoch, most clearly on channels Fp1 and Fp2 which are closest to the eyes.

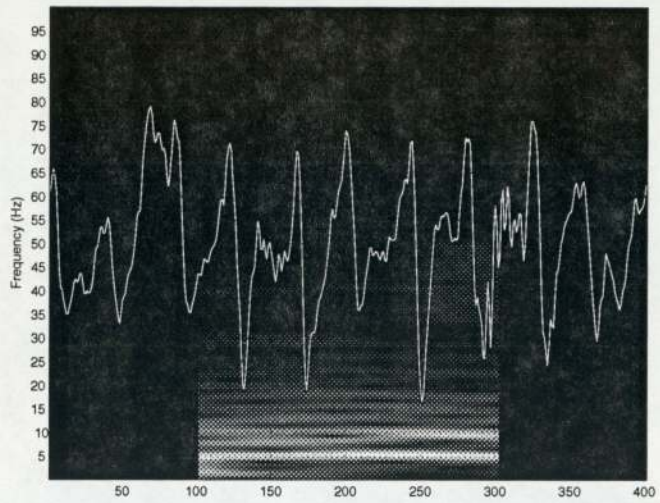


Figure 6.27: Seizure 5: Spectroplot of channel F7 of Seizure 5, between 16 and 18 seconds. The seizure waveform is prominent and has a fundamental frequency of approximately 5 Hz. This was used as the frequency of the reference signal.

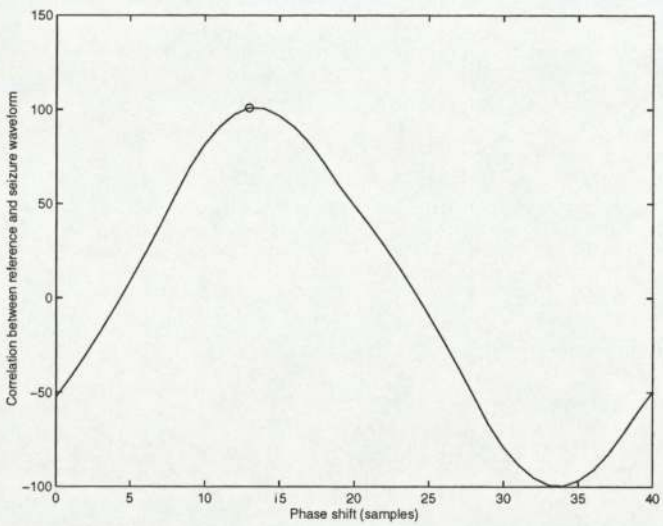


Figure 6.28: Seizure 5: Correlation between selected seizure waveform and a 5 Hz square wave reference signal which is swept through one cycle. Both the seizure waveform and the reference signal were first normalised. The maximum correlation, for an offset of 13 samples, is shown as a circle.

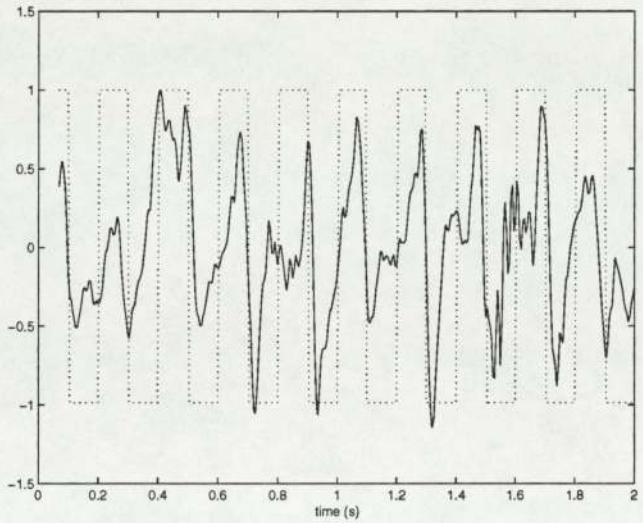


Figure 6.29: Seizure 5: The best phase for the 5 Hz square wave reference. This phase corresponds to the maximum correlation in Figure 6.28. (The reference signal and seizure waveform were both normalised for this purpose.)

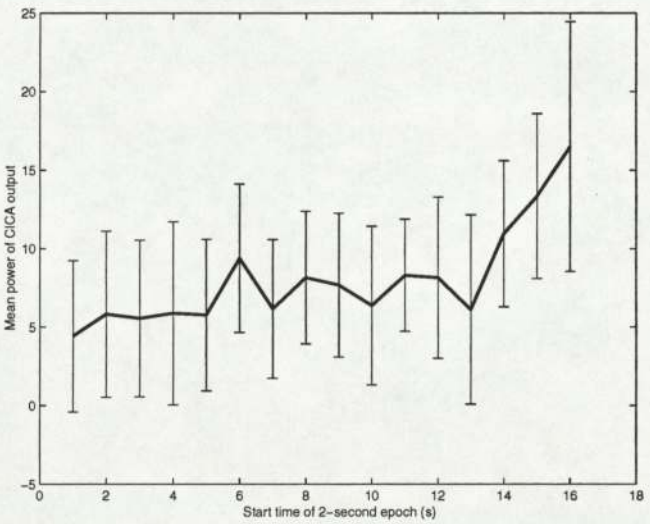
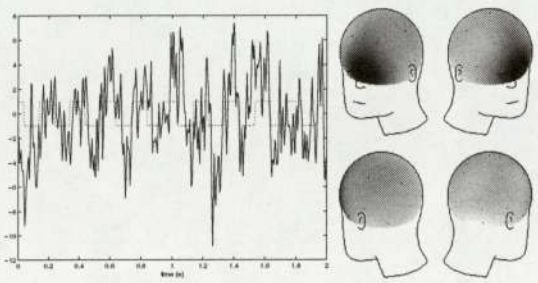
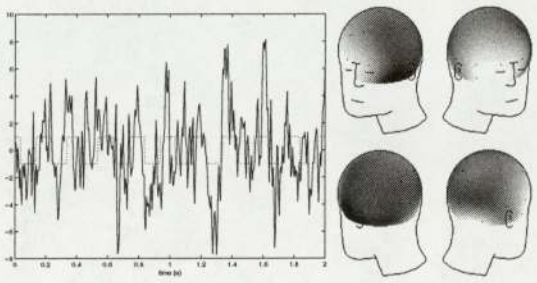


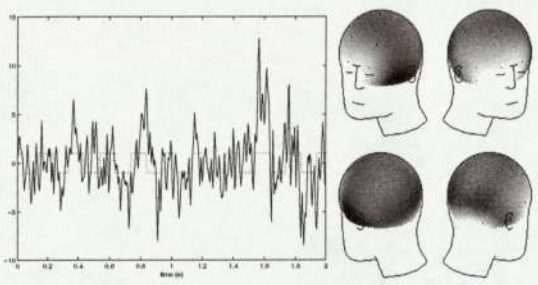
Figure 6.30: Seizure 5: Mean output power for a range of epochs, with error bars of one standard deviation. The CICA algorithm was run 100 times for each epoch. The times correspond to those on Figure 6.26.



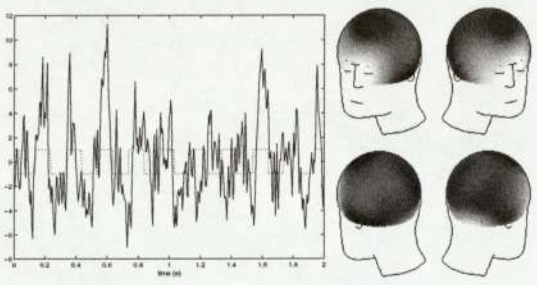
(a) Start point: 1 s



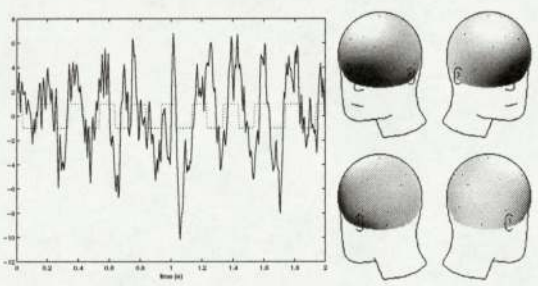
(b) Start point: 5 s



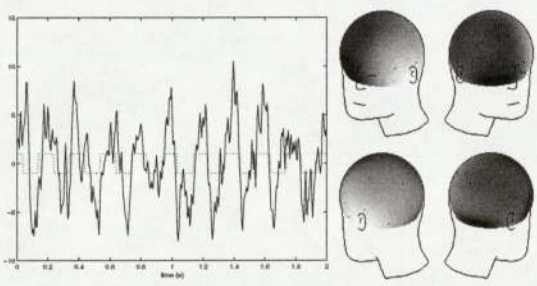
(c) Start point: 7 s



(d) Start point: 11 s



(e) Start point: 13 s



(f) Start point: 15 s

Figure 6.31: Seizure 5: Example outputs of the CICA algorithm applied to different time periods. The start point times correspond to the time axis of Figure 6.26. The topographic map of each example is also shown.

6.3 Algorithm behaviour given a "false" reference

The behaviour of the CICA algorithm when given a "false" reference has been studied in previous chapters. In this section the algorithm is applied to two recordings with no visible seizure waveform, but with the square wave reference signal used for Seizure 3. In choosing the EEG recordings to use, it was not possible to obtain recordings from patients without epilepsy, to ensure that no seizure waveform could be present. Instead, time periods distant from seizures were chosen to minimise the chance of seizure waveforms being present in the data.

6.3.1 "False" reference applied to ongoing EEG 1

The reference used for Seizure 3 was first tested with the ongoing EEG recording shown in Figure 6.32. There is no visible seizure waveform in the recording, although several artifacts are present. Channels Fp1, Fp2, O1 and O2 show a continuous high frequency artifact.

Using the same procedure as before, the mean and standard deviation of the CICA algorithm output power were stored for different two-second time windows throughout the recording. The results, shown in Figure 6.33, show that the mean powers are not greatly different from the powers obtained from seizure recordings. This suggests that, assuming that there is no seizure waveform actually present, the power of the output is not a reliable indicator of the presence or absence of seizure waveform. However, more useful information seems to be provided by the spatial maps shown in Figure 6.34. Although the extracted waveforms shown as examples could be interpreted as resembling very noisy seizure waveform, albeit with little temporal consistency, the topographies (with the exception of Figure 6.34(e)) seem to be less focused than those for "good" extractions of seizure waveform shown previously. Furthermore, the foci of the topographies do not, in general, match foci from genuine seizures. This raises the possibility of a spatial constraint to reduce the occurrence of false positives, which is discussed in the following chapter.

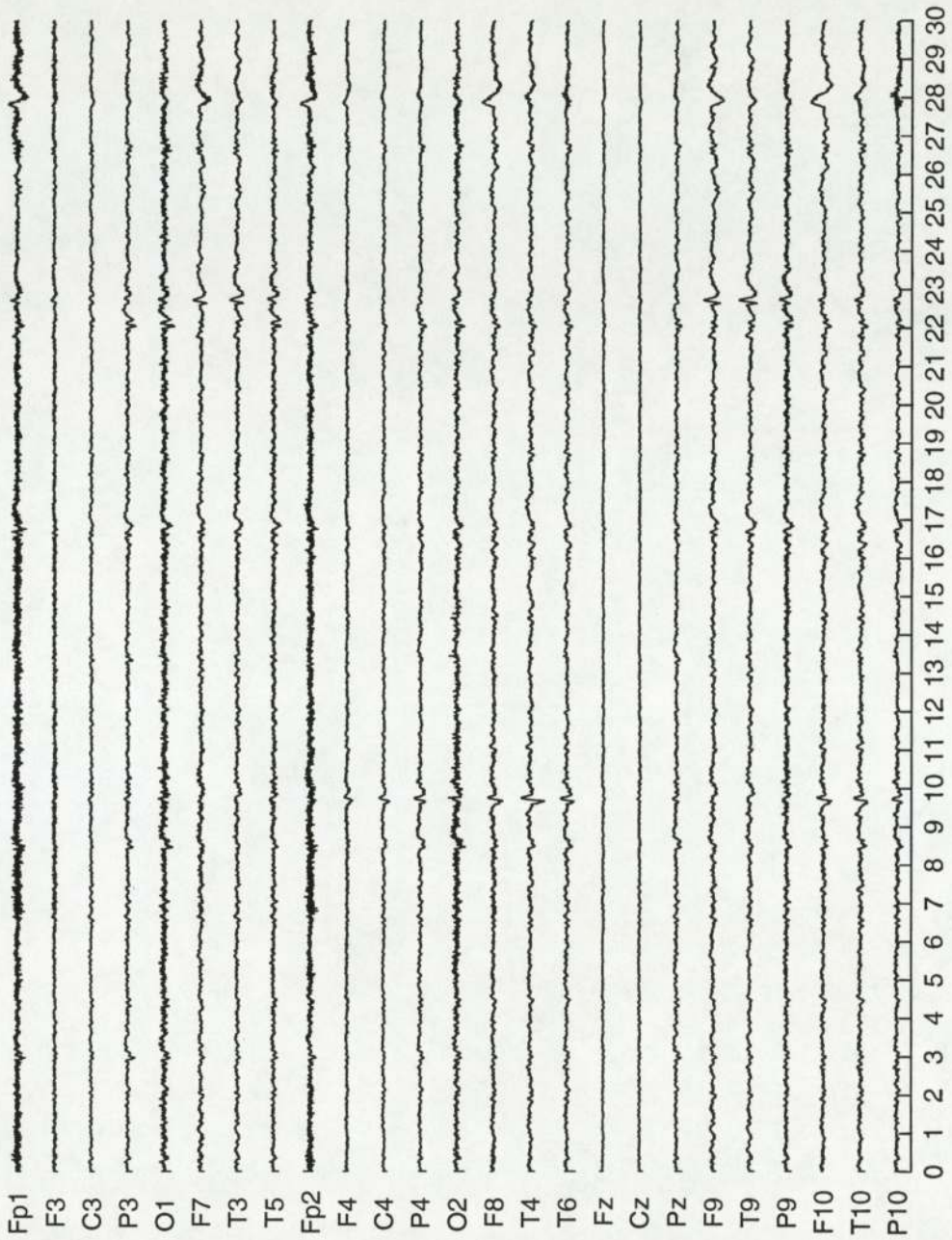


Figure 6.32: "False" reference 1: 30 second EEG epoch distant in time from a seizure, sampled at 200 Hz with 25 electrodes in the standard positions. A referential montage was used with FCz as the reference electrode. Some electrical noise seems to be present on channels Fp1 and Fp2 (at the front of the head) and O1 and O2 (at the back of the head). This artifact is not at mains frequency but might be caused by the recording process or by other nearby equipment.

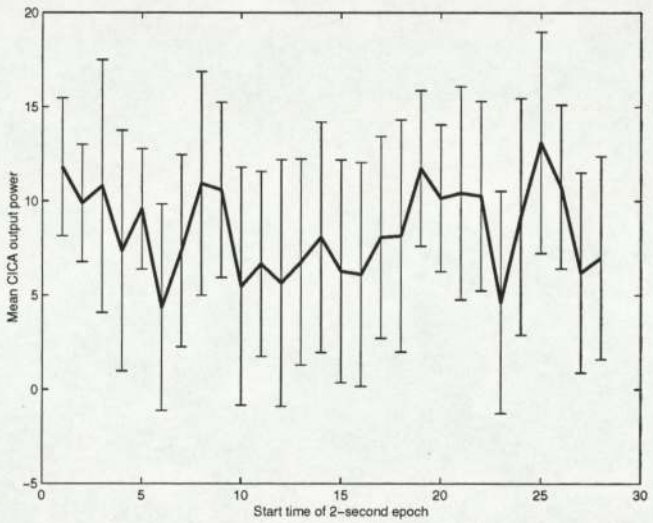
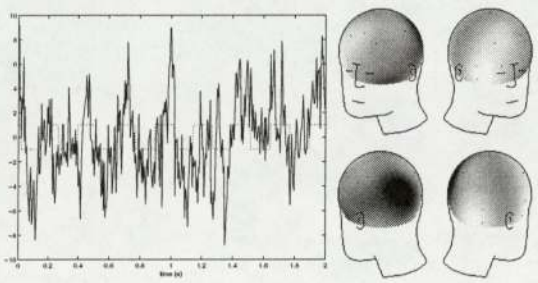
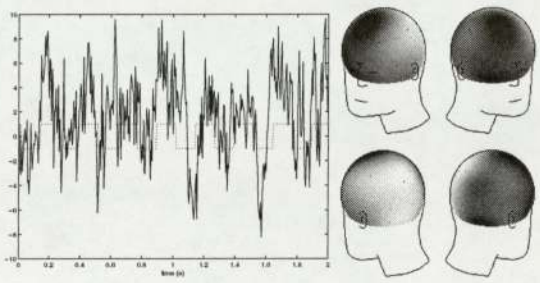


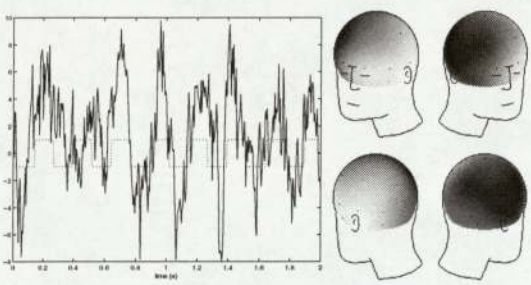
Figure 6.33: "False" reference: Mean output power for a range of epochs. The CICA algorithm was run 100 times for each epoch. The times correspond to those on Figure 6.32.



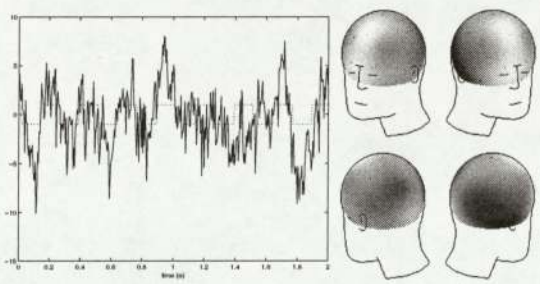
(a) Start point: 2 s



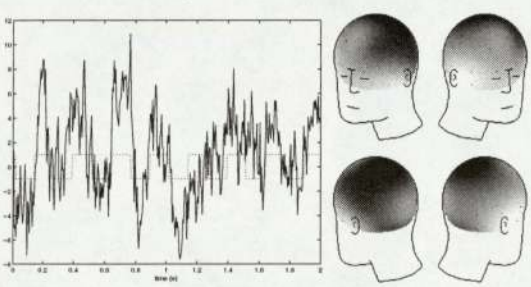
(b) Start point: 5 s



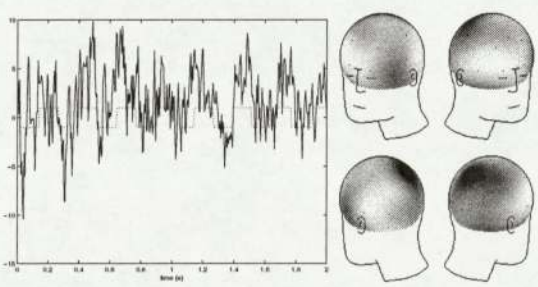
(c) Start point: 12 s



(d) Start point: 17 s



(e) Start point: 22 s



(f) Start point: 28 s

Figure 6.34: "False" reference 1: Example outputs of the CICA algorithm applied to different time periods. The start point times correspond to the time axis of Figure 6.32. The topographic map of each example is also shown.

6.3.2 "False" reference applied to ongoing EEG 2

The second "ongoing" EEG recording (Figure 6.35) is similar to the one previously described. It again features high-frequency activity, but on a larger number of channels. There appears to be no visible seizure waveform, however.

The same procedure as described above was performed to test the CICA algorithm with a "false" reference. The mean output signal power (Figure 6.36) again shows similar power levels to those for the extraction of genuine seizure waveform. The example outputs (Figure 6.37) are again very noisy and have little temporal consistency, but the majority of the topographies show a focus over the eyes. This might seem to pose a problem for the future use of a spatial map as a CICA constraint.

6.4 Summary

This chapter has shown how the CICA algorithm can be applied to the analysis of real-world seizure onset periods. Unlike the work with simulated seizure waveform (Chapter 4), the true underlying sources in this real-world EEG data are all unknown. Large numbers of automatically-assessed trials could not therefore be performed. Instead, the CICA algorithm was applied to time periods moving backwards from the clinical onset of each seizure. Reference signals were obtained by selecting a section of clear seizure waveform and finding its fundamental frequency. The problem of phase matching between the reference signal and the seizure waveform was addressed by considering the correlation between the reference and a section of selected seizure waveform, for a range of phase shifts. The maximum correlation between the seizure waveform and the reference signal indicated the best phase match.

The results for extractions within 10 seconds of a visible seizure onset were generally good, with relatively clean seizure waveform extracted and a topography closely focused on the seizure origin. The extraction of seizure waveform from more distant periods proved problematic, especially in the presence of artifacts such as eye blinks which were able to influence the topography.

Trials with a "false" reference, where a reference approximating seizure waveform was used with a recording containing no visible seizure waveform, showed that the algorithm produced outputs which could be construed as similar to the given reference. The algorithm may have extracted an underlying independent source resembling the reference, or it may have mixed actual sources to match the reference. Since, for this real-world data, the actual underlying sources are not known, it is not possible to distinguish between these cases. These "false" extractions could not be distinguished from the extraction of genuine seizure waveform by the output power, which was generally similar for both types. However, the "false" extractions showed less temporal consistency than genuine extractions of seizure waveform. Furthermore, the spatial distribution of the outputs could be used to assess the reliability of the extracted waveform, since the topographies of the extractions with a "false" reference did not resemble focused seizures. The possibilities of future work using a spatial, rather than temporal, constraint, are discussed in the following chapter.

As a proof of principle of the application of the CICA algorithm to seizure onset analysis, in a small set of example epochs, these results are very promising: the algorithm has been capable of extracting seizure waveforms with their topographies from time periods prior to the visible onset of a seizure. Further trials with expert knowledge of the seizures (effectively giving more insight into the actual underlying sources) are suggested for future work.

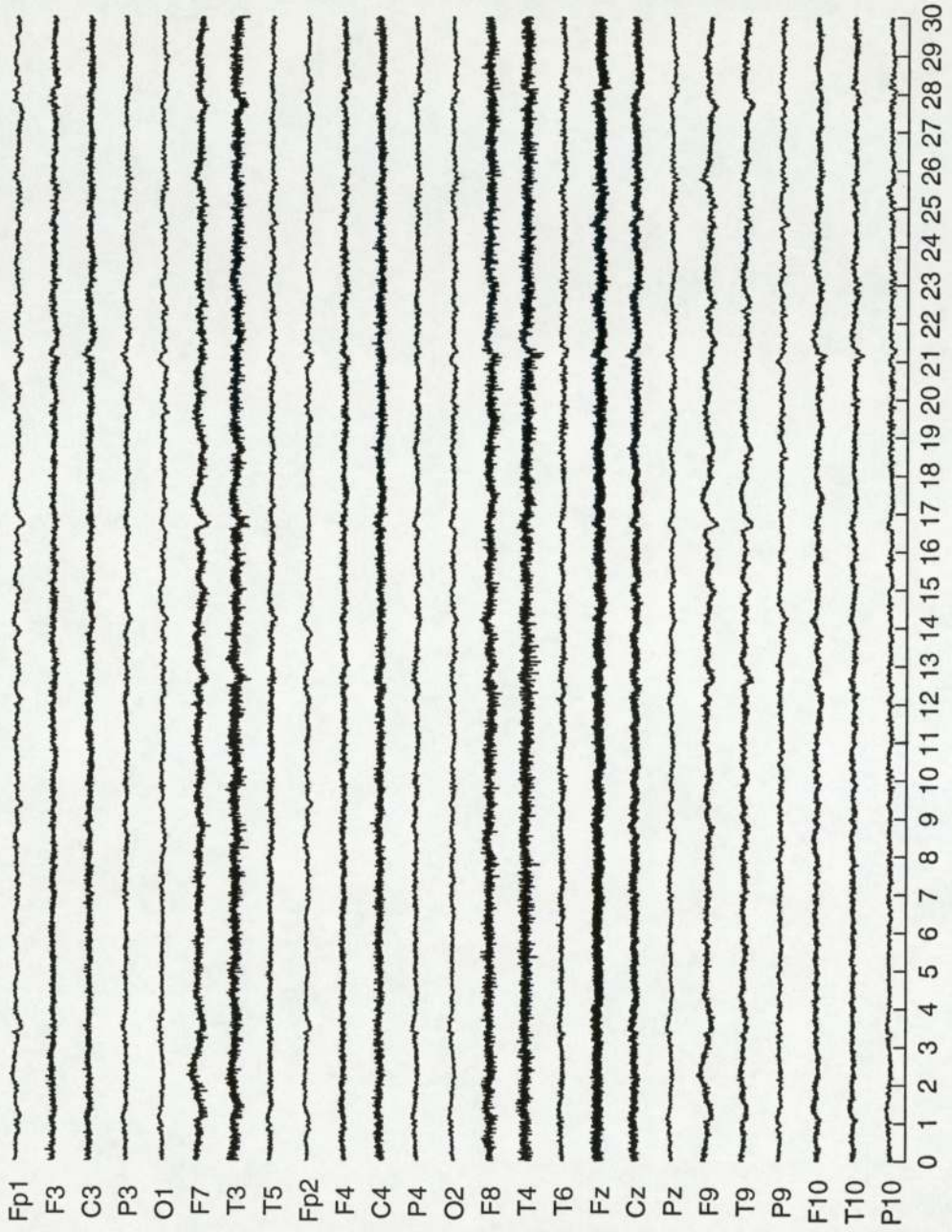


Figure 6.35: "False" reference 2: 30 second EEG epoch distant in time from a seizure, sampled at 200 Hz with 25 electrodes in the standard positions. A referential montage was used with FCz as the reference electrode. Some noise is present, which is likely to be caused by interference from nearby equipment or the recording process.

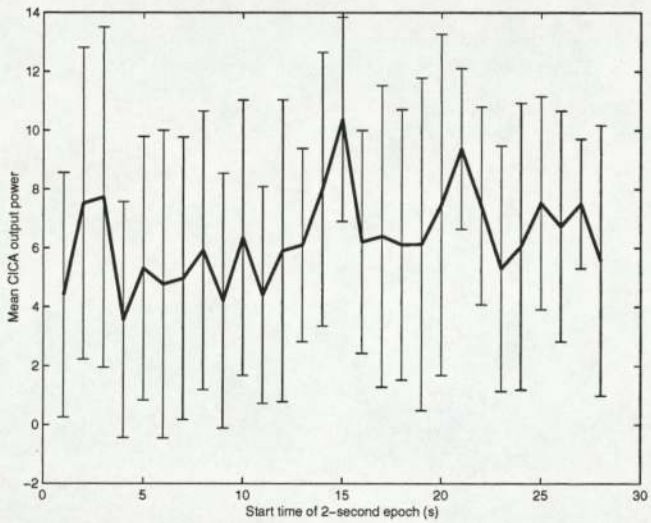
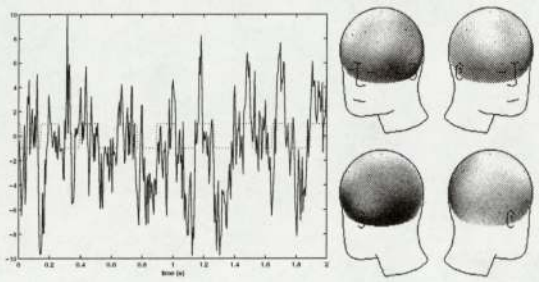
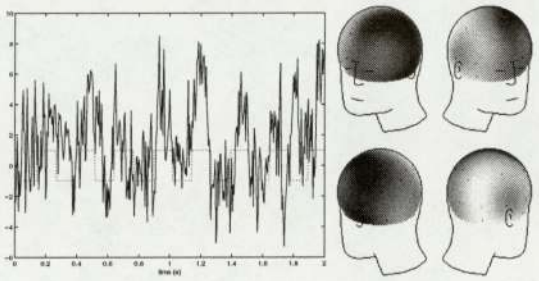


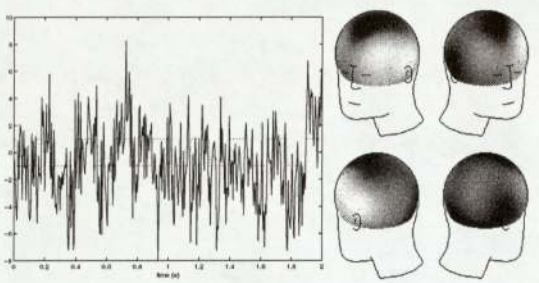
Figure 6.36: "False" reference 2: Mean output power for a range of epochs. The CICA algorithm was run 100 times for each epoch. The times correspond to those on Figure 6.35.



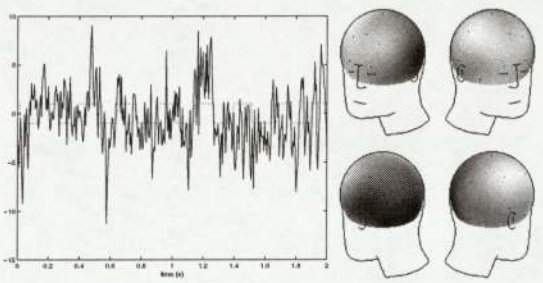
(a) Start point: 2 s



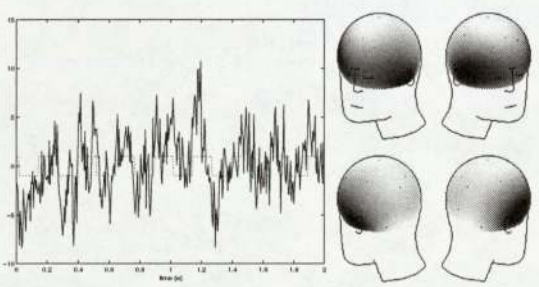
(b) Start point: 5 s



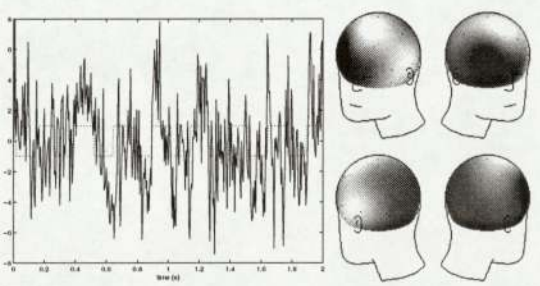
(c) Start point: 9 s



(d) Start point: 16 s



(e) Start point: 20 s



(f) Start point: 27 s

Figure 6.37: "False" reference 2: Example outputs of the CICA algorithm applied to different time periods. The start point times correspond to the time axis of Figure 6.35. The topographic map of each example is also shown.

Chapter 7

Conclusions and future work

This thesis has studied the application of the recently-proposed CICA algorithm to the problem of seizure onset analysis in EEG signals. A related application in EEG artifact rejection has also been demonstrated.

The advantages of analysing EEG signals in an ICA framework have been shown in previous work [13]. The key assumption of linear mixing is valid for brain signals and the sub-Gaussianity of EEG signals allows negentropy approximations to be used for separating underlying sources. However, the assumption of a square mixing matrix used in conventional ICA algorithms results in the extraction of the same number of components as there are observation channels. This can result in large numbers of components which are not of interest, or the splitting of components if there are more channels than actual underlying sources. The CICA algorithm [23] discussed in this thesis uses negentropy approximations to extract a single independent source, the choice of which is dictated by a supplied reference signal. Since only one source is extracted, the actual number of underlying sources is unimportant for this algorithm. The use of the *a priori* reference avoids any manual intervention to select the desired component from the multiple, unordered, components produced by conventional ICA.

At the time of writing, no implementation of the CICA algorithm was publically available. The first stage of this project was therefore to implement this algorithm in Matlab (Appendix A) and test it on a simple data set (Chapter 3). The initial use of a data set with few sources and complete knowledge of the sources allowed the algorithm's performance to be quantified. The use of a temporal MSE closeness measure for this purpose was discussed. The algorithm performed well at extracting the pure source closest by correlation to a coarse reference signal, given a mixture of sources. The choice of contrast function was addressed, along with the performance improvement gained by imposing a miniumum number of iterations before convergence. In these tests, an association was shown between low output powers of the dewhitened algorithm output and the extraction of mixtures: the dewhitened output power offered an indication of the extraction of a mixture of sources for this data set. The use of this measure in EEG data proved more problematic, perhaps because the assumption that the desired underlying source has relatively high power may not hold, especially for the pre-seizure extraction of seizure waveform.

The ability of the algorithm to extract synthetic seizure waveform which had been mixed with real-world EEG epochs (Chapter 4) was tested. The need for a suitable multi-channel SNR measure was highlighted; the measure used caused difficulties when used with real-world EEG epochs containing

anomalies. However, in general the algorithm performed well at extracting the synthetic seizure waveform at a range of SNR values. The presence of a range of common EEG anomalies did not significantly affect the results. A set of trials with a "false" reference was also performed, where no simulated seizure waveform was present in the input data. In these circumstances the algorithm did not appear to mix sources to match the reference. The use of a spatial, rather than temporal, MSE measure for comparing algorithm outputs to the simulated seizure was discussed but was found to give poor results.

The application of the CICA algorithm to the problem of EEG artifact rejection was demonstrated in Chapter 5. While artifact rejection is not the main focus of this project, it is relevant to seizure onset analysis because artifacts must sometimes be removed to allow analysis of the EEG. In this application the CICA algorithm has produced some promising results.

Finally, the CICA algorithm was used to analyse real-world seizures (Chapter 6) with some success. The algorithm can extract seizure waveform from time periods before the clinical onset of a seizure. The topographic map of the extracted component provides information on the origin of the seizure.

7.1 Future work

A key area for further development of the CICA algorithm is the investigation of alternatives to the temporal constraint used in this work. While useful results were obtained with this constraint, it has several drawbacks. For example, with the correlation measure used here, periods where the reference signal is zero are treated as having no prior information. As Figure 5.2 showed, the algorithm is not penalised for producing a non-zero output during these periods. Better results might be obtained if there was discrimination between periods where no prior knowledge applies and periods during which the output signal ought to be close to zero. For seizure onset analysis where the approximate frequency of the seizure waveform is not known (for example, in a real time seizure prediction system), an alternative constraint based on the rhythmicity of the algorithm's output might be appropriate. A rhythmicity measure has been proposed [24] as a method for separating sources. It might therefore replace the negentropy method used for this work, or it could be used as a constraint. Another possibility is a frequency-based reference which uses Fourier analysis to specify the expected seizure frequency. This might, however, be subject to the same problems which have impeded the direct use of Fourier analysis for seizure waveform extraction: other signals commonly found in the EEG have similar frequencies to seizure waveform. For seizure onset analysis, a spatial constraint would cause the algorithm to extract sources which have a similar spatial distribution to the known seizure. A more sophisticated system could incorporate prior knowledge of a "realistic" spatial distribution and penalise the algorithm if unrealistic topographies are produced.

In all the work described in this thesis, the algorithm's initial weight vector was randomly generated. Better results might be obtained by generating the starting vector based on the reference - in other words, prior to the algorithm's attempt to achieve statistical independence, the weight vector would be optimised to give the closest output to the reference signal. However, in the case of "false" references which do not match any underlying source, this technique could cause erroneous results.

If seizure recordings are to be analysed offline, the time taken for the algorithm to converge is not important, within reason. However, another aim of seizure onset analysis is the prediction of seizures in real time in order to give as much warning as possible to the patient. In principle the CICA algorithm should be suitable for real time operation since it normally converges within 15 iterations.

On a modern computer this takes a few seconds, which is close to real time for the two-second epochs used. Furthermore, the implementation used for this work was not optimised for speed.

If the algorithm was developed to a stage where real-time seizure prediction was feasible, the problem of portability would need to be addressed. The algorithm could run on a hand-held computer, for example, but the 25 scalp electrodes would restrict patient movement. It would be preferable to use only one or two electrodes positioned close to the origin of seizure, if known. In this case, it might be necessary to use techniques such as dynamical embedding to reconstruct the system dynamics from a small number of channels [25].

The performance of the algorithm itself could be compared to similar algorithms such as a recently-published algorithm using sparse decomposition for source separation with signal dictionaries to provide a template for the desired output [26]. Alternatively, a non-gradient-based algorithm might be used, to avoid problems with global maximisation which may have caused some of the "wrong" extractions described in this thesis.

Appendix A

Source code of CICA algorithm implementation

The Matlab source code of the CICA algorithm, as implemented for this project, is listed below. This implementation is discussed in Chapter 2.

```
function [cicasig, W, A, ic_corr, errorlevel, niters] = cica(mix, ref, cicaoptions, othercmds)
% "cica.m"
%
% Constrained ICA based on NIPS 2000 paper by Wei Lu & Jagath C. Rajapakse
%
% USAGE: [cicasig, W, A, ic_corr, errorlevel, niters] = cica( mix, ref, cicaoptions, othercmds)
%
% (o/p) cicasig : Constrained IC
% (o/p) W       : The weight vector of the IC such that; y = W'*mix
% (o/p) A       : The estimated mixing vector of the IC such that; mix = A*y                    10
% (o/p) ic_corr : The correlation of the IC with the reference signal
% (o/p) errorlevel: 0 if OK, 1 if failed to converge
% (o/p) niters : Number of iterations taken to converge
%
% (i/p) mix     : The input data matrix n x l
% (i/p) ref     : Reference vector 'l' samples long
% (i/p) cicaoptions: String of options, as follows:
%
% {general supergaussian subgaussian} - Contrast function to be used
%                              verbose - Displays warnings during run                           20
%                           showgraphs - Displays graphs during run
%                              whiten - Whiten the input data and dewhiten the output
%                         varythreshold - Gradually increase threshold over iterations
%
% (i/p) othercmds: Extra Matlab commands to be executed before the main loop
%
% Christopher James & Oliver Gibson

errorlevel = 0; % Default return value is no error

                                                                                               30

% Check input arguments
if nargin<3
```

```
   error('Wrong number of input arguments entered');
end

% Find size of input matrix
[n, data_len] = size(mix);
new_n         = n;

if length(ref)~=data_len                                                    40
   error('The reference vector must be the same length as each measurement vector');
end


% Whiten the data
if(findstr(cicaoptions,'whiten'))
    % Remove mean from each row of data
    for i = 1:n
        mix_no_mean(i,:) = mix(i,:) - mean(mix(i,:));
    end                                                                     50

    covarianceMatrix = cov(mix_no_mean', 1);  % Find eigenvalues/vectors
    max_num_eig      = rank(covarianceMatrix);  % Indicates the max num of eigenvectors possible
    [E, D]           = eig(covarianceMatrix);

    if max_num_eig<n                           % This adjusts the matrices if data matrix is not full rank

        [eig_val, eig_index] = sort(diag(D));
        eig_val              = flipud(eig_val);
        eig_index            = flipud(eig_index);                           60

        D = diag(eig_val(1:max_num_eig));
        E = E(:,eig_index(1:max_num_eig));

        new_n = max_num_eig;

    end

    whiteningMatrix = inv (sqrt (D)) * E';      % Obtain (de)whitening matrices
    dewhiteningMatrix = E * sqrt (D);                                       70
    new_mix         = whiteningMatrix * mix_no_mean; % Finally whiten the data
else
    new_mix = mix;
end


% Normalise the reference to zero mean, unit variance
ref = ref - mean(ref);
ref = ref ./ std(ref);
                                                                            80

% Given the input data, find covariance matrix
Rxx = new_mix*new_mix';


% Initialise variables
learning_rate    = 0.9;  % "nabla"
```

```
penalty_parameter = 0.5; % "gamma"
threshold         = 0.5;  % "large Epsilon"
rho               = 0.01; % Negentropy constant                                    90


% Parameters for variable threshold (the 'varythreshold' option)
if(findstr(cicaoptions,'varythreshold'))
    thresh_param_init = 0.0; % Initial Threshold value
    thresh_param_fin = 0.5; % Final Threshold value
    thresh_param_range = 10; % Range of iterations over which threshold changes
end

% Display parameters                                                             100
number_of_retries = 1;  % Max number of retries if converges but not with a good correlation
number_of_iters = 200;  % Max number of iterations per try allowed
display_epoch   = 2;    % Display every "display_epoch"
w_log_thresh    = 1;    % Used to check for convergence, was 1
ic_corr_thresh  = 0.15; % Used to see if desired component could be found


% Give the algorithm 'number_of_retries' tries to converge to the desired solution
while number_of_retries>0
                                                                                   110

    number_of_retries = number_of_retries - 1; % One less try available

    % Perform any user-specified initialisation
    if(nargin>3)
        cmd_error_msg = sprintf('Error evaluating additional commands');
        eval(othercmds,'disp(cmd_error_msg);pause;');
    end

        % Initialise Lagrange Multipliers                                        120
        lambda          = 0;
        mu              = 0;

        % Initialise the weights
        w    = randn(new_n,1)-0.5;
        w_old = zeros(size(w)); % for convergence check

        % Initialise the Graphical display

        if(findstr(cicaoptions,'showgraphs'))                                      130
          figure(1);
          clf;
          subplot(2,2,1);
          plot(ref);
          title('Reference Signal');
          drawnow;
        end


        % Storage for variables stored at each iteration                         140
        estore    = zeros(1,number_of_iters); % Preallocate epsilon storage
        corrstore = zeros(1,number_of_iters); % Preallocate abs correlation storage
```

```
wconv    = zeros(1,number_of_iters); % Preallocate weight convergence info storage


converged_flag = 0; % - This will be set when algorithm converges


% Now start the iterations
for niters=1:number_of_iters;                                                    150

% Increase the threshold linearly
if(findstr(cicaoptions,'varythreshold'))
   threshhold = min([(thresh_param_fin-thresh_param_init)*((niters-1)/thresh_param_range)+thresh_param_init, ...
                                                                    thresh_param_fin]);
end

   % Calculate latest output
   y = w' * new_mix;
                                                                                 160

   % Update rhobar sign
   if mean(y) < 0
      rhobar = -rho;
   else
      rhobar = rho;
   end


   % ---------------------->                                                    170
   % - Calculate "closeness" to ref, and hence g(w) ->
   % - NB: Use correlation as "closeness" measure ->
   % -    g(w)  = E{y*ref'} - threshold      ->
   % -    g'(w) = E{new_mix*ref'}            ->
   % -    g''(w) = 0                         ->

   % - Temporarily normalise y
   ynorm = y - mean(y);
   ynorm = ynorm ./ std(ynorm);
                                                                                 180
   epsilon    = ynorm*ref';
   g          = epsilon - threshold;
   g_dash     = new_mix*ref';
   g_dashdash = 0;

   % - Store the values for later
   estore(niters)    = epsilon;
   corrstore(niters) = (ynorm*ref')/(ynorm*ynorm');

   % - Calculate h(w)                                                          190
   h = mean(y.^2) - 1; % eqn 5


% Calculate first and second derivatives of contrast function
if(findstr(cicaoptions,'subgaussian')) % G3 = (y.^4)/4
   Gd = y.^3; %  <- 1st derivative
   Gdd = 3*(y.^2);% <- 2nd derivative
```

```
    end
if(findstr(cicaoptions,'supergaussian')) % G2 = exp(-0.5*a*y.^2)/a
    a = 1; % Parameter a should be approximately 1                                    200
    Gd = -y.*exp(-0.5*a*(y.^2)); % <- 1st derivative
    Gdd = ((a.*(y.^2))-1).*exp(-0.5*a*(y.^2)); % <- 2nd derivative
    end
if(findstr(cicaoptions,'general')) % G1 = log cosh(a*y)/a
    a = 1.5; % Allowed range: 1 <= a <= 2
    Gd = tanh(a*y); % <- 1st derivative
    Gdd = a*(sech(a*y).^2); % <- 2nd derivative
    end


                                                                                      210
    % == Update dL: The 1st Derivative of the Augmented Lagrange Equation (eqn 12) ==>
    % - ... first term: rhobar*E{new_mix*G3'(y)} ->
    dL1 = rhobar * new_mix*(Gd)';
    % - ... second term: 0.5*mu*E{g'_w(w)} ->
    dL2 = 0.5 * mu * g_dash;
    % - ... third term: lambda*E{new_mix*y} ->
    dL3 = lambda * (new_mix*y');
    dL = dL1 + dL2 - dL3;

    % - Calculate: alpha(w) = rhobar*E{G''(w*new_mix')} - lambda ->                   220
    alpha = (rhobar*mean(Gdd)) - lambda;

% Now update the Lagrange multipliers: lambda and mu (eqns 10 & 11)
    lambda = lambda + (penalty_parameter * h);
    mu     = max([0 (mu + (penalty_parameter * g))]);

    % Now update weights (eqns 14 and 15)
if(alpha == 0)
 errorlevel = 1; % Divide by zero may occur with supergaussian contrast function and small penalty parameter
 disp('Divide by zero in CICA');                                                      230
 break;
end

    w = w - (learning_rate*inv(Rxx)*dL/alpha);
    wconv(niters) = (norm(w - w_old));


    % - Now display where we are at in the calculations ->

    if findstr(cicaoptions,'showgraphs') & (mod(niters,display_epoch)==0 | niters==number_of_iters)   240

      figure(1);
      subplot(2,2,2);
      plot(log(wconv(1:niters)),'b');
      hold on;
      plot(log(wconv(1:niters)),'ro');
      hold off;
      title(sprintf('Convergence of log(:) weight-w (threshold: %05.4f)',w_log_thresh));

      subplot(2,2,4);                                                                 250
      plot(abs(corrstore(1:niters)),'b');
      hold on; plot(abs(corrstore(1:niters)),'m*');hold off;
```

```
    axis([1 niters 0 1]);
    title('Absolute Correlation progress');

    figure(1);
    subplot(2,2,3);
    plot(ref,'r');
    hold on;
    plot(ynorm,'b');                                                    260
    hold off;
    title(sprintf('Output at iteration %d (CORR: %f)',niters,epsilon));

    drawnow;

  end


  % - Check to see if we have converged
  % - (Only allow convergence after a minimum number of iterations)    270

  if (log(norm(w - w_old)) < w_log_thresh )

  if(findstr(cicaoptions,'varythreshold')) % Let the threshold variation finish in any case
    if(niters>thresh_param_range)
      % Converged!
          converged_flag = 1;
        break;
    end
  else                                                                  280
    % Converged! (no threshold variation selected)
    converged_flag = 1;
    break;
  end
  end

  w_old = w;

end
                                                                        290

% - Have we converged? ->

if ~converged_flag

    disp_str = sprintf('WARNING: CICA did not converge to below %05.4f after %d iterations ', ...
                                          w_log_thresh, number_of_iters);

    if number_of_retries==0
        if findstr(cicaoptions,'verbose')                               300
            disp(sprintf('%s- giving up (Max number of retries exceeded)',disp_str));
    errorlevel = 1;
        end
        break;
    else
        if findstr(cicaoptions,'verbose')
            disp(sprintf('%s- trying again (%d left)',disp_str,number_of_retries));
```

```
                end
            end
                                                                    310
        else

            % - Finished, now return the mixing and de-mixing vectors ->

            w = w/norm(w); % - Place this here to normalise the amplitude of the cIC ->
        y = w' * new_mix; % - "y=w'*new_mix" is now 'equivalent' to "cicasig=(new_mix'*w)" ->

    if(findstr(cicaoptions,'whiten'))
            A       = dewhiteningMatrix*w;
            W       = w'*whiteningMatrix;                           320
            cicasig = W*mix;
    end



            ic_corr = corrstore(niters);

            % - Issue a warning if the resulting correlation between reference
            % - and cica is very poor
                                                                    330
            if (ic_corr)<ic_corr_thresh

                disp_str=sprintf('WARNING:Output has abs correlation with reference of less than %05.4f' ...
                                                                ,ic_corr_thresh);

                if number_of_retries==0
                    if findstr(cicaoptions,'verbose')
                        disp(sprintf('%s- giving up (Max number of retries exceeded)',disp_str));
                    end
                    break;                                          340
                else
                    if findstr(cicaoptions,'verbose')
                        disp(sprintf('%s- trying again (%d left)',disp_str,number_of_retries));
                    end
                end
            else

                number_of_retries==0;
                break;
            end                                                     350

        end

    end
```

Bibliography

[1] M. Sundaram, R.M. Sadler, G.B. Young, and N. Pillay. EEG in epilepsy: Current perspectives. *Canadian Journal of Neurological Sciences*, 26(4), November 1999.

[2] A. Hopkins, S. Shorvon, and G. Cascino, editors. *Epilepsy.* Chapman and Hall Medical, 2nd ed. edition, 1995.

[3] Website of Epilepsy Action, formerly the British Epilepsy Association. http://www.epilepsy.org.uk.

[4] M.-L. Hector. *EEG Recording.* Butterworths, 1980.

[5] H.H. Jasper. The ten-twenty system of the international federation. *EEG and Clin. Neuro.*, 10(371), 1958.

[6] John S. Ebersole. *Ambulatory EEG Monitoring.* Raven Press, 1989.

[7] C. J. James. *Detection of epileptiform activity in the electroencephalogram using artificial neural networks.* PhD thesis, University of Canterbury, Christchurch, New Zealand, February 1997.

[8] C. J. James. 3D topography MATLAB package. Available from Biomedical Information Engineering Research Group, Aston University. http://www.bierg.aston.ac.uk.

[9] J. Martinerie, C. Adam, M. Le Van Quyen, M. Baulac, S. Clemenceau, B. Renault, and F.J. Varela. Epileptic seizures can be anticipated by non-linear analysis. *Nat. Med.*, 4(10):1173–76, October 1998.

[10] M. Le Van Quyen, J. Martinerie, V. Navarro, P. Boon, M. D'Havé, C. Adam, B. Renault, F. Varela, and M. Baulac. Anticipation of epileptic seizures from standard EEG recordings. *The Lancet*, 357:183 – 188, January 2001.

[11] P. Verdes, G. Deco, D. Obradovic, L. Dubé, R. Hopfengaertner, and H. Stefan. Detection and prediction of epileptic seizures: A patient's case study. In *Proceedings of CACIC 2000, the VI Argentine Congress on Computer Science, Ushuaia, Argentina*, pages 1493–1503, 2000. ISBN 950-763-033-3.

[12] D. Kugiumtzis and P. G. Larsson. Linear and nonlinear analysis of EEG for the prediction of epileptic seizures. In *Proceedings of the 1999 Workshop "Chaos in Brain?"*, pages 329–333. World Scientific, Singapore, 2000.

[13] C. J. James and D. Lowe. Isolating seizure activity in the EEG with independent component analysis. In *Artificial Neural Networks in Medicine and Biology - Proceedings of the ANNIMAB-1 Conference*, 2000.

[14] S. Roberts and R. Everson, editors. *Independent Component Analysis: Principles and Practice.* Cambridge University Press, 2001.

[15] A. J. Bell and T. J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159, 1995.

[16] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis.* Adaptive and learning systems for signal processing, communications and control. John Wiley and Sons, 2001.

[17] W. Lu and J. C. Rajapakse. Extracting a desired source in ICA. In *International Conference on Neural Information Processing (ICONIP2001)*, Shanghai, November 2001.

[18] A. Hyvärinen. One-unit contrast functions for independent component analysis. In *Neural Networks for Signal Processing VII Proc. IEEE NNSP Workshop '97, Amelia Island, Florida*, pages 388–397, 1997.

[19] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492, 1997.

[20] W. Lu and J. C. Rajapakse. ICA with reference. In *Proceedings of Third International Conference on Independent Component Analysis and Blind Signal Separation: ICA2001*, 2001.

[21] J. Hurri, H. Gävert, J. Särelä, and A. Hyvärinen. FastICA MATLAB package [online]. Available: http://www.cis.hut.fi/project/ica/fastica.

[22] N. A. Weiss. *Introductory Statistics.* Addison-Wesley, 6th edition, 2002.

[23] W. Lu and J. C. Rajapakse. Constrained independent component analysis. In *Advances in Neural Information Processing Systems 13 (NIPS2000)*, pages 570 – 576, December 2000.

[24] J. V. Stone. Blind source separation using temporal predictability. *Neural Computation*, pages 1559–1574, 2001.

[25] C. J. James and D. Lowe. Extracting multisource brain activity from single EM channel. *Special issue of Artificial Intelligence in Medicine*, 2002.

[26] M. Zibulevsky and B. A. Pearlmutter. Blind source separation by sparse decomposition in a signal dictionary. *Neural Computation*, 13(4):863–882, 2001.