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

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities

CLARIVATE ANALYTICS

BOOK CITATION INDEX

INDEXED

WEB OF SCIENCE™

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

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

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

Nonlinear Independent Component Analysis for EEG-Based Brain-Computer Interface Systems

Farid Oveisi[1], Shahrzad Oveisi[2], Abbas Efranian[3] and Ioannis Patras[1]
[1]Queen Mary University of London
[2]Azad University
[3]Iran University of Science and Technology
[1]UK
[2,3]Iran

1. Introduction

The electroencephalogram (EEG) is a complex and a periodic time series, which is a sum over a very large number of neuronal membrane potentials. Despite rapid advances of neuroimaging techniques, EEG recordings continue to play an important role in both the diagnosis of neurological diseases and understanding the psychophysiological processes. Recently, many efforts have been done to use the electroencephalogram as a new communication channel between human brain and computer (Lotte & Guan, 2011; Oveisi, 2009; Ortner et al., 2011). This new communication channel is called EEG-based brain–computer interface (BCI). Most of these efforts have been dedicated to the improvement of the accuracy and capacity of this EEG-based communication channel. One of the most important factors about the performance of BCI systems is classification system. A classification system typically consists of both a preprocessor and a classifier. Preprocessors are used to improve the performance of classifier systems. One of the preprocessors can be used to improve the performance of brain–computer interface (BCI) systems is independent component analysis (ICA) (Van et al., 2011; Oveisi, 2009). ICA is a signal processing technique in which observed random data are transformed into components that are statistically independent from each other (Oveisi et al., 2012). ICA is a useful technique for blind separation of independent sources from their mixtures. Sources are usually original, uncorrupted signals or noise sources. Linear ICA was used to separate neural activity from muscle and blink artifacts in spontaneous EEG data (Jung et al., 2000). It was verified that the ICA can separate artifactual, stimulus locked, response-locked, and non-event related background EEG activities into separate components (Jung et al., 2001). Furthermore, ICA would appear to be able to separate task-related potentials from other neural and artifactual EEG sources during hand movement imagination in form of independent components. In (Peterson et al., 2005), it has been showed that the power spectra of the linear ICA transformations provided feature subsets with higher classification accuracy than the power spectra of the original EEG signals. However, there is no guarantee for linear combination of brain sources in EEG signals. Thus the identification of non-linear dynamic of EEG signals should be taken into consideration. For non-linear mixing model, linear ICA algorithms fail to extract original signals and become inapplicable because the assumption of linear

mixtures is violated and the linear algorithm cannot compensate for the information distorted by the non-linearity.

ICA is currently a popular method for blind source separation (BSS) of linear mixtures. However, nonlinear ICA does not necessarily lead to nonlinear BSS (Zhang & Chan, 2007). Hyvarinen and Pajunen (1999) showed that solutions to nonlinear ICA always exist, and that they are highly non-unique. In fact, nonlinear BSS is impossible without additional prior knowledge on the mixing model, since the independence assumption is not strong enough in the general nonlinear mixing case (Achard & Jutten, 2005; Singer & Coifman, 2007). If we constrain the nonlinear mixing mapping to have some particular forms, the indeterminacies in the results of nonlinear ICA can be reduced dramatically, and as a consequence, in these cases nonlinear ICA may lead to nonlinear BSS. But sometimes, the form of the nonlinear mixing procedure may be unknown. Consequently, in order to model arbitrary nonlinear mappings, one may need to resort to a flexible nonlinear function approximator, such as the multi-layer perceptron (MLP) (Woo & Sali, 2002; Almeida, 2003) or the radius basis function (RBF) network (Tan et al., 2001), to represent the nonlinear separation system. In this situation, in order to achieve BSS, nonlinear ICA requires extra constraints or regularization. In (Woo & Sali, 2002), a general framework for a demixer based on a feedforward multilayer perceptron (FMLP) employing a class of continuously differentiable nonlinear functions has been explained. In this method, Cost functions based on both maximum entropy (ME) and minimum mutual information (MMI) have been used. In (Almeida, 2003), the MLP has been used to model the separation system and trains the MLP by information maximization (Infomax). Moreover, smoothness provided by the MLP was believed to be a suitable regularization condition to achieve nonlinear BSS. In (Tan et al., 2001), a blind signal separation approach based on an RBF network is developed for the separation of nonlinearly mixed sources by defining a contrast function. This contrast function consists of mutual information and cumulants matching. However, the matching between the relevant moments of the outputs and those of the original sources was expected to guarantee a unique solution. But the moments of the original sources may be unknown.

In this research, a nonlinear ICA has been used to separate task-related potentials from other neural and artifactual EEG sources. The proposed method has been tested on several different subjects. Moreover, the results of proposed method were compared to the results obtained using linear ICA, and original EEG signals.

2. Background

2.1 Mutual information

Mutual information is a non-parametric measure of relevance between two variables. Shannon's information theory provides a suitable formalism for quantifying these concepts. Assume a random variable $\mathbf{X}$ representing continuous-valued random feature vector, and a discrete-valued random variable C representing the class labels. In accordance with Shannon's information theory, the uncertainty of the class label C can be measured by entropy $H(C)$ as

$$H(C) = -\sum_{c \in C} p(c) \log p(c), \tag{1}$$

where $p(c)$ represents the probability of the discrete random variable C. The uncertainty about C given $\mathbf{X}$ is measured by the conditional entropy as

$$H(C|X) = -\int p(\mathbf{x})\left(\sum_{c \in C} p(c|\mathbf{x}) \log p(c|\mathbf{x})\right) d\mathbf{x}, \tag{2}$$

where $p(c|\mathbf{x})$ is the conditional probability for the variable C given X.

In general, the conditional entropy is less than or equal to the initial entropy. It is equal if and only if the two variables C and X are independent. The amount by which the class uncertainty is decreased is, by definition, the mutual information $I(X,C) = H(C) - H(C|X)$ and after applying the identities $p(c, \mathbf{x}) = p(c|\mathbf{x})p(\mathbf{x})$ and $p(c) = \int p(c, x) dx$ can be expressed as

$$I(X,C) = \sum_{c \in C} \int p(\mathbf{x}, c) \log \frac{p(\mathbf{x}, c)}{p(c)p(\mathbf{x})} d\mathbf{x} \tag{3}$$

If the mutual information between two random variables is large, it means two variables are closely related. The mutual information is zero if and only if the two random variables are strictly independent. The mutual information and the entropy have the following relation, as shown in Fig. 1:

$$I(X;Y) = H(X) - H(X|Y)$$
$$I(X;Y) = H(Y) - H(Y|X)$$
$$I(X;Y) = H(X) + H(Y) - H(X,Y) \tag{4}$$
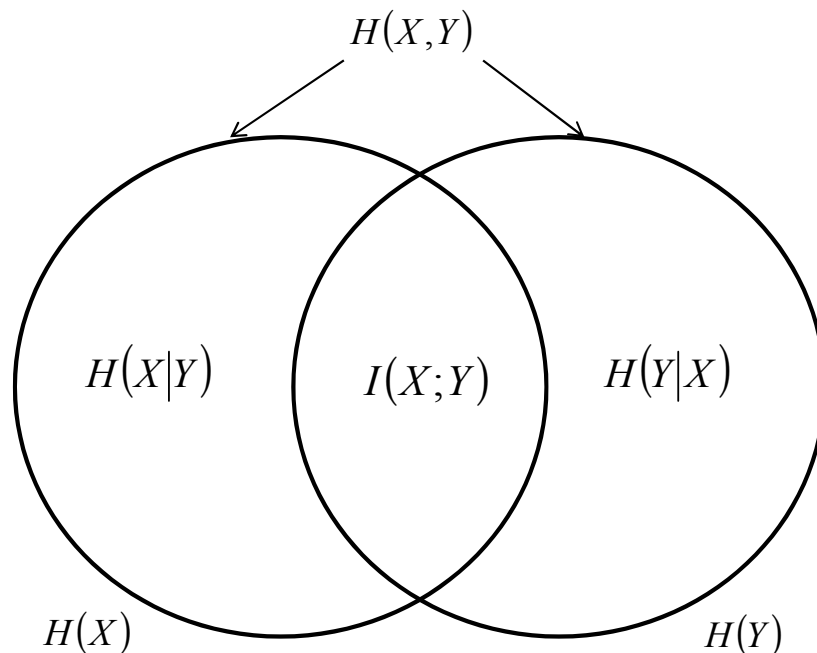$$I(X;Y) = I(Y,X)$$
$$I(X,X) = H(X).$$

Fig. 1. The relation between the mutual information and the entropy.

2.2 Genetic algorithm

In a genetic algorithm, a population of strings (called chromosomes or the genotype of the genome), which encode candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem, evolves toward better solutions. Traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible. The evolution usually starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), and modified (recombined and possibly randomly mutated) to form a new population. The new population is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population. If the algorithm has terminated due to a maximum number of generations, a satisfactory solution may or may not have been reached.

A standard representation of the solution is as an array of bits. Arrays of other types and structures can be used in essentially the same way. The main property that makes these genetic representations convenient is that their parts are easily aligned due to their fixed size, which facilitates simple crossover operations. Variable length representations may also be used, but crossover implementation is more complex in this case. Tree-like representations are explored in genetic programming and graph-form representations are explored in evolutionary programming.

The fitness function is defined over the genetic representation and measures the quality of the represented solution. The fitness function is always problem dependent. For instance, in the knapsack problem one wants to maximize the total value of objects that can be put in a knapsack of some fixed capacity. A representation of a solution might be an array of bits, where each bit represents a different object, and the value of the bit (0 or 1) represents whether or not the object is in the knapsack. Not every such representation is valid, as the size of objects may exceed the capacity of the knapsack. The fitness of the solution is the sum of values of all objects in the knapsack if the representation is valid or 0 otherwise. In some problems, it is hard or even impossible to define the fitness expression; in these cases, interactive genetic algorithms are used.

Once the genetic representation and the fitness function are defined, a GA proceeds to initialize a population of solutions (usually randomly) and then to improve it through repetitive application of the mutation, crossover, inversion and selection operators.

Initially many individual solutions are (usually) randomly generated to form an initial population. The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. Traditionally, the population is generated randomly, allowing the entire range of possible solutions (the search space). Occasionally, the solutions may be "seeded" in areas where optimal solutions are likely to be found (Akbari & Ziarati, 2010).

During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a fitness-based process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. Certain selection methods rate the fitness of each solution and preferentially select

the best solutions. Other methods rate only a random sample of the population, as the latter process may be very time-consuming.

The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover (also called recombination), and/or mutation.

For each new solution to be produced, a pair of "parent" solutions is selected for breeding from the pool selected previously. By producing a "child" solution using the above methods of crossover and mutation, a new solution is created which typically shares many of the characteristics of its "parents". New parents are selected for each new child, and the process continues until a new population of solutions of appropriate size is generated. Although reproduction methods that are based on the use of two parents are more "biology inspired", some research suggests more than two "parents" are better to be used to reproduce a good quality chromosome.

These processes ultimately result in the next generation population of chromosomes that is different from the initial generation. Generally the average fitness will have increased by this procedure for the population, since only the best organisms from the first generation are selected for breeding, along with a small proportion of less fit solutions, for reasons already mentioned above.

Although Crossover and Mutation are known as the main genetic operators, it is possible to use other operators such as regrouping, colonization-extinction, or migration in genetic algorithms.

This generational process is repeated until a termination condition has been reached. Common terminating conditions are:

- A solution is found that satisfies minimum criteria
  - Fixed number of generations reached
  - Allocated budget (computation time/money) reached
  - The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results
  - Manual inspection
  - Combinations of the above

Simple generational genetic algorithm procedure:

1. Choose the initial population of individuals
2. Evaluate the fitness of each individual in that population
3. Repeat on this generation until termination (time limit, sufficient fitness achieved, etc.):
   1. Select the best-fit individuals for reproduction
   2. Breed new individuals through crossover and mutation operations to give birth to offspring
   3. Evaluate the individual fitness of new individuals
4. Replace least-fit population with new individuals

3. Independent Component Analysis (ICA)

3.1 Linear ICA

We assume that we observe n linear mixtures $x_1, x_2, \ldots, x_n$ of n independent components:

$$x_j = a_{j_1} s_1 + a_{j_2} s_2 + \ldots + a_{j_n} s_n \tag{5}$$

In this equation the time has been ignored. Instead, it was assumed that each mixture x_j as well as each independent component s_i are random variables and $x_j(t)$ and $s_i(t)$ are samples of these random variables. It is also assumed that both the mixture variables and the independent components have zero mean (Oveisi et al., 2008).

If not subtracting the sample mean can always center the observable variables x_i. This procedure reduces the problem to the model zero-mean:

$$\hat{x} = x - E(x) \tag{6}$$

Let x be the random vectors whose elements are the mixtures $x_1, x_2, \ldots, x_n$ and let s be the random vector with the components $s_1, s_2, \ldots, s_n$. Let **A** be the matrix containing the elements a_{ij}. The model can now be written:

$$x = As \ \text{ or } \ x = \sum_{i=1}^{n} a_i s_i \tag{7}$$

The above equation is called independent component analysis or ICA. The problem is to determine both the matrix A and the independent components s, knowing only the measured variables x. The only assumption the methods take is that the components s_i are independent. ICA looks a lot like the "blind source separation" (BSS) problem or blind signal separation: a source is in the ICA problem an original signal, so an independent component. In ICA case it is also no information about the independent components, like in BSS problem.

Whitening can be performed via eigenvalue decomposition of the covariance matrix:

$$VDV^T = E\left\{ \hat{x}\hat{x}^T \right\} \tag{8}$$

where **V** is the matrix of orthogonal eigenvectors and **D** is a diagonal matrix with the corresponding eigenvalues. The whitening is done by multiplication with the transformation matrix **P**:

$$\tilde{x} = P\hat{x} \tag{9}$$

$$P = VD^{\frac{1}{2}}V^T \tag{10}$$

The matrix for extracting the independent components from $\tilde{x}$ is $\tilde{W}$, where $W = \tilde{W}P$

3.2 Nonlinear ICA

Conventional linear ICA approaches assume that the mixture is linear by virtue of its simplicity. However, this assumption is often violated and may not characterize real-life signals accurately. A realistic mixture needs to be non-linear and concurrently capable of

treating the linear mixture as a special case (Lappalainen & Honkela, 2000; Gao et al., 2006; Jutten & Karhunen, 2004). Generally, a non-linear ICA problem can be defined as follows: given a set of observations, $x(t) = \left[x_1(t), x_2(t), \ldots, x_n(t) \right]^T$ which are random variables and generated as a mixture of independent components $s(t) = \left[s_1(t), s_2(t), \ldots, s_n(t) \right]^T$ according to
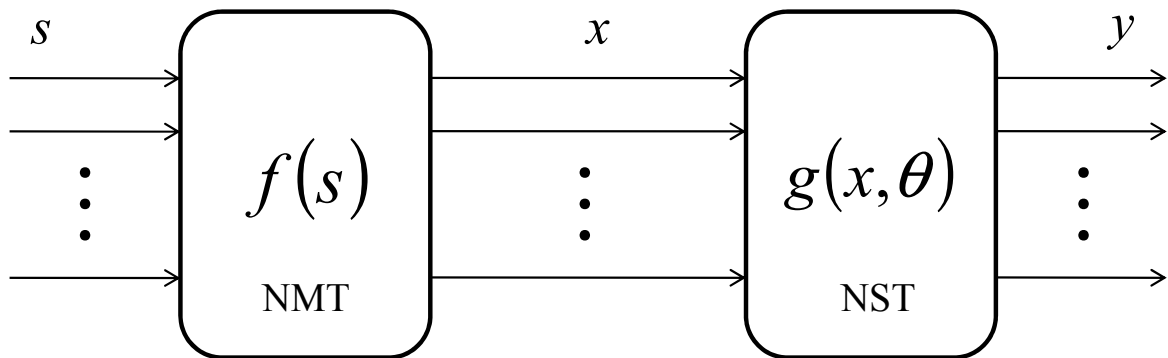
s   x   y

$$f(s)$$ $$g(x,\theta)$$

NMT   NST

Fig. 2. Nonlinear mixing and separating systems for independent component analysis.

$$x(t) = f\left[s(t) \right] \tag{11}$$

where f is an unknown nonlinear mixing transform (NMT). The block diagram of the nonlinear ICA is shown in Figure 2.

The separating system $g(.,\theta)$ in the right part of Fig. 2, called nonlinear separation transform (NST) is used to recover the original signals $x(t)$ from the nonlinear mixture without the knowledge of the source signals $s(t)$ and the mixing nonlinear function f. However, a fundamental difficulty in nonlinear ICA is that it is highly non-unique without some extra constraints; therefore, finding independent components does not lead us necessarily to the original sources (Achard & Jutten, 2005).

ICA in the nonlinear case is, in general, impossible. In (Rojas et al., 2004), it has been added some extra constraints to the nonlinear mixture so that the nonlinearities are independently applied in each channel after a linear mixture. As figure 3 shows, the proposed algorithm in (Rojas et al., 2004) needs to estimate two different mixtures: a family of nonlinearities g which approximates the inverse of the nonlinear mixtures f and a linear unmixing matrix W which approximates the inverse of the linear mixture A. For the demixing system, first we need to approximate g_i, which is the inverse of the nonlinear function in each channel, and then separate the linear mixing by applying W to the output of the g_i nonlinear function:

$$y_i(t) = \sum_{j=1}^{n} w_{ij} g_i \left(x_j(t) \right) \tag{12}$$

In order to develop a more general and flexible model of the function g_i, it can be used a M th order odd polynomial expression of nonlinear transfer function (g_i):

$$p_j(x_j) = \sum_{k=1}^{M} p_{jk} x_j^{2k-1} \tag{13}$$

where $p_j = \left[p_{j1}, p_{j2}, \ldots, p_{jM} \right]$ is a parameter vector to be determined. By using relations (12) and (13), we can write the following criterion for the output sources y_i:
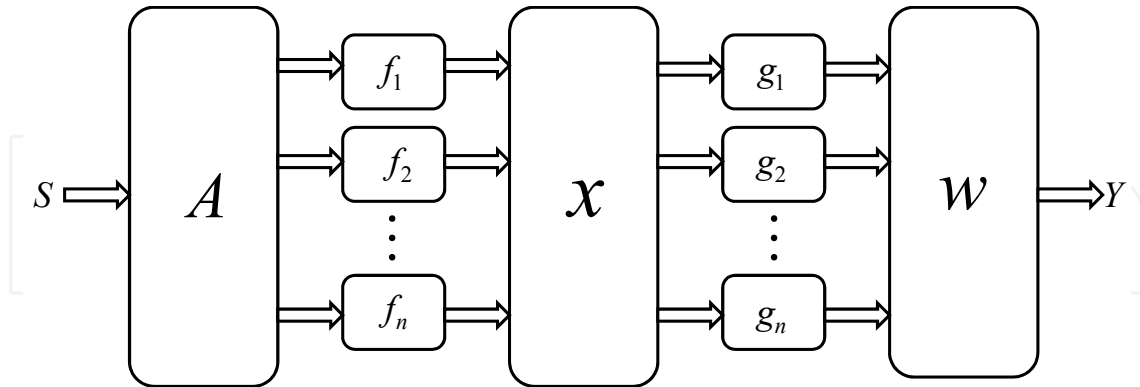

Fig. 3. Post-nonlinear mixing and demixing models for independent component analysis.

$$y_i(t) = \sum_{j=1}^{n} w_{ij} \sum_{k=1}^{M} p_{jk} . x_j^{2k-1} \qquad (14)$$

The parameter vector p_j should be determined so that the inverse of the mutual information of the output sources y_i is maximized. To achieve this objective, can be defined the following criterion (Rojas et al., 2004):

$$eval\_function(y) = \frac{1}{I(y)} \qquad (15)$$

Nevertheless, computation of the parameter vectors p_j is not easy, as it presents a problem with numerous local minima when the usual BSS cost functions are applied. Thus, we require an algorithm that is capable of avoiding entrapment in such a minimum. As a solution, in this work, a genetic algorithm (GA) (Goldberg, 1989) was used for mutual information optimization. Unlike many classical optimization techniques, GA does not rely on computing local first- or second-order derivatives to guide the search process; GA is a more general and flexible method that is capable of searching wide solution spaces and avoiding local minima (i.e., it provides more possibilities of finding an optimal or near-optimal solution). To implement the GA, we use genetic algorithm and direct search toolbox for use in Matlab (The Mathworks, R2007b).

The linear demixing stage has been performed by the well-known Infomax algorithm (Hyvarinen et al., 2001). To be precise, Infomax has been embedded into the GA in order to approximate the linear mixture.

In this application, the genetic algorithm is run for 30 generations with population size of 20, crossover probability 0.8, and uniform mutation probability of 0.01. The number of individuals that automatically survive to the next generation (i.e., elite individuals) is selected to be 2. The scattered function is used to create the crossover children by creating a random binary vector and selects the genes where the vector is a 1 from the first parent, and the genes where the vector is a 0 from the second parent.

4. Experimental setup

The EEG data of healthy right-handed volunteer subjects were recorded at a sampling rate of 256 from positions Cz, T5, Pz, F3, F4, Fz, and C3 by Ag/AgCl scalp electrodes placed according to the International 10-20 system that has been shown in Fig. 4. The eye blinks were recorded by placing an electrode on the forehead above the left brow line. The signals were referenced to the right earlobe.

Data were recorded for 5 s during each trial experiment and low-pass filtered with a cutoff 45 Hz. There were 100 trails acquired from each subject during each experiment day. At $t = 2$ s, a cross ("+") was displayed on the monitor of computer as a cue visual stimulus. The subjects were asked to imagine the hand grasping in synchronization with the cue and to not perform a specific mental task before displaying the cue. In the present study, the tasks to be discriminated are the imaginative hand movement and the idle state. The experimental setup has been shown in Fig. 5.
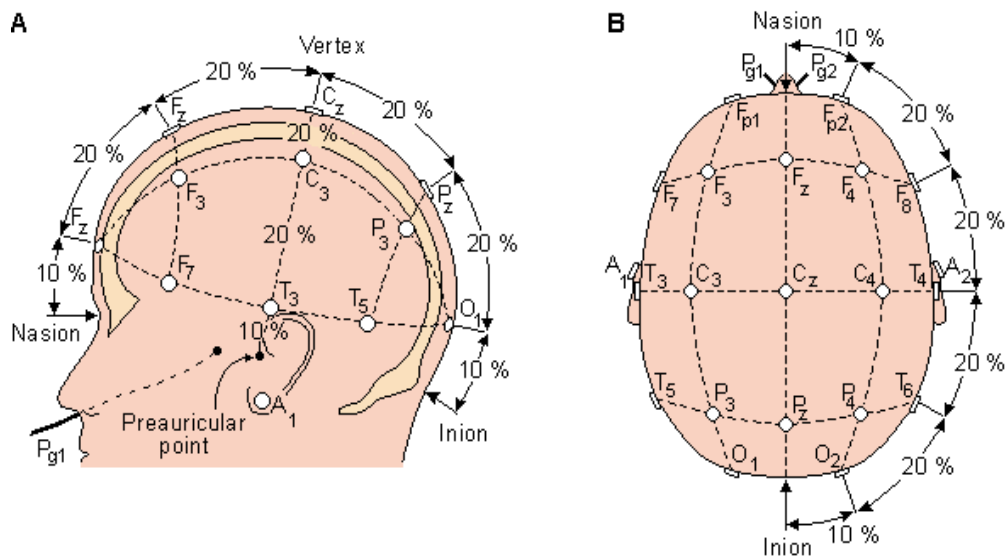
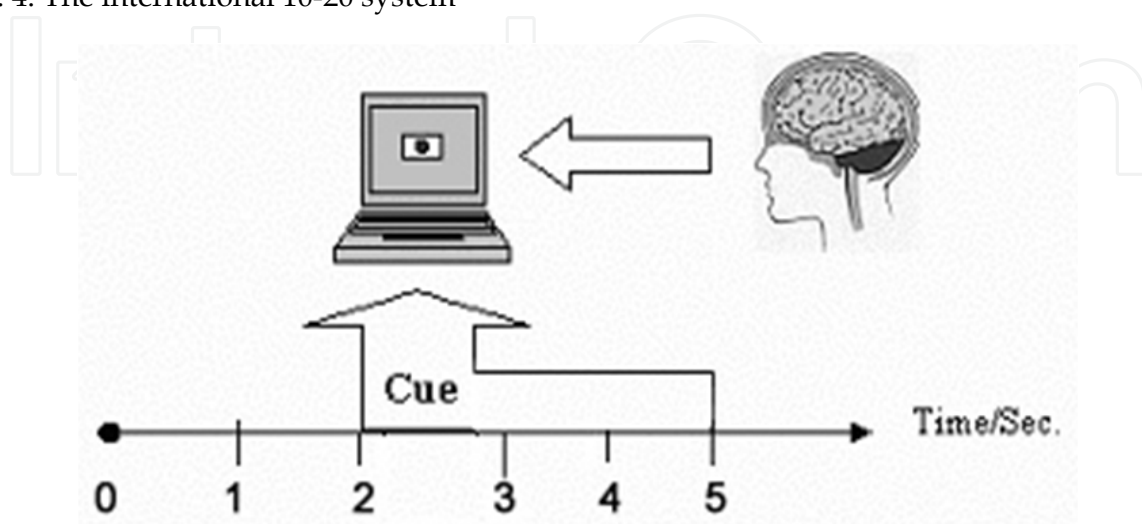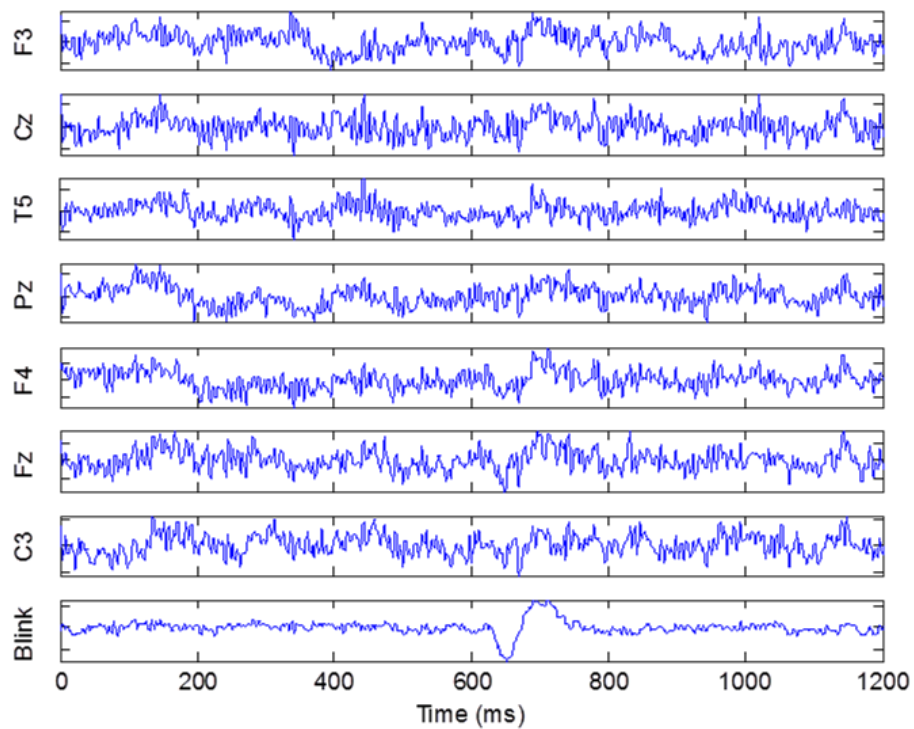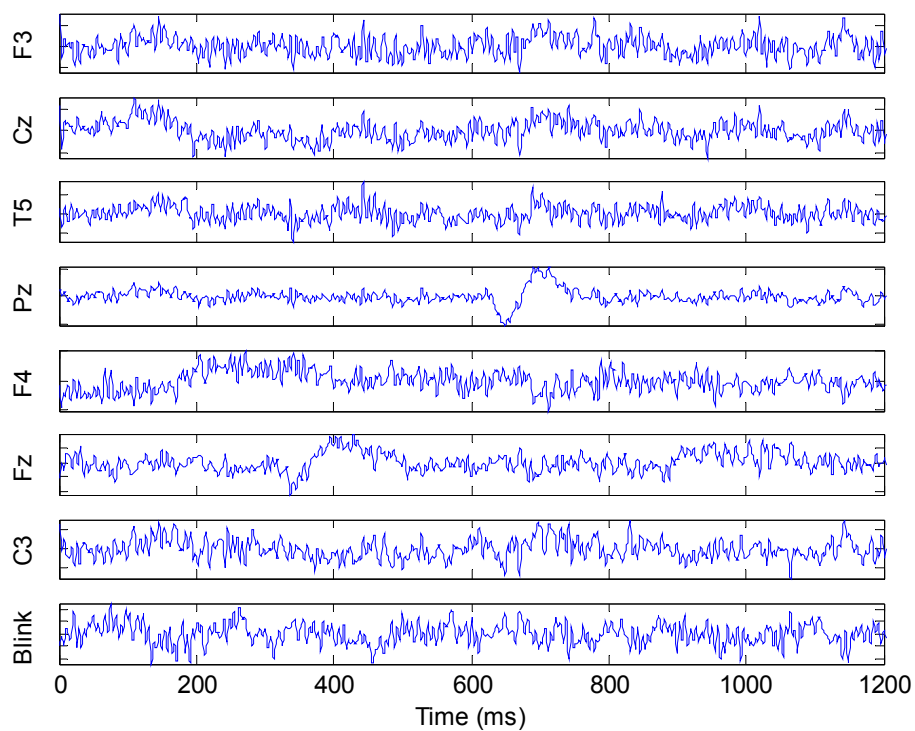Fig. 4. The international 10-20 system

Fig. 5. Experimental Setup

Eye blink artifact was suppressed by using independent component analysis. The artifactual independent components were visually identified and set to zero. This process has been shown in Fig. 6.
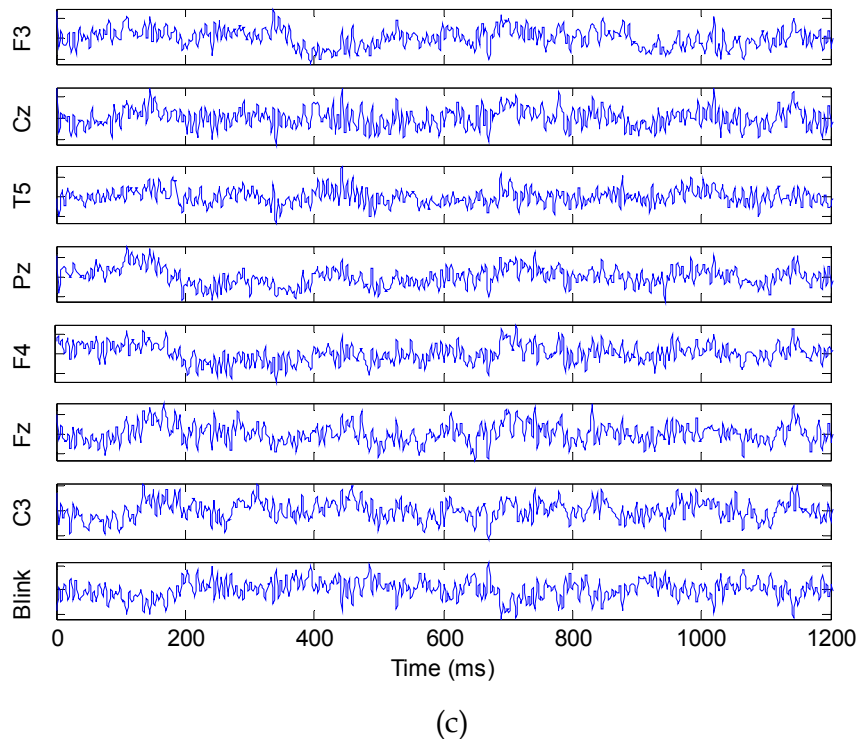

(a)

(b)

(c)

Fig. 6. (a) Raw EEG signals, (b) ICA components, (c) EEG signals after eye blink removal.

5. Results

The nonlinear ICA algorithm, proposed in (Rojas et al., 2004), was applied to given training 7-channel EEG data sets associated to the hand movement imagination and resting state. Original features are formed from 1 second interval of each component, in the time period 2.3–3.3 seconds, during each trial of experiment. The window starting 0.3 seconds after cue presentation is used for classification. The number of local extrema within interval, zero crossing, 5 AR parameters, variance, the mean absolute value (MAV), and 1Hz frequency components between 1 and 35Hz constitute the full set of features with size 44. The classifier is trained to distinguish between rest state and imaginative hand movement. The imaginative hand movement can be hand closing or hand opening. From 200 data sets, 100 sets are randomly selected for training, while the rest is kept aside for validation purposes. Training and validating procedure is repeated 10 times and the results are averaged.

Multiple classifiers are employed for classification using extracted components obtained by linear and nonlinear ICA. The Multiple Classifiers are used if different sensors are available to give information on one object. Each of the classifiers works independently on its own domain. The single classifiers are built and trained for their specific task. The final decision is made on the results of the individual classifiers. In this work, for each component, separate classifier is trained and the final decision is implemented by a simple logical majority vote function. The desired output of each classifier is −1 or +1. The output of classifiers is added and the *signum function* is used for computing the actual response of the classifier. The diagonal linear discrimination analysis (DLDA) (Krzanowski, 2000) is here considered as the classifier. The classifier is trained to distinguish between rest state and imaginative hand movement. The block diagram of classification process is shown in Fig. 7.
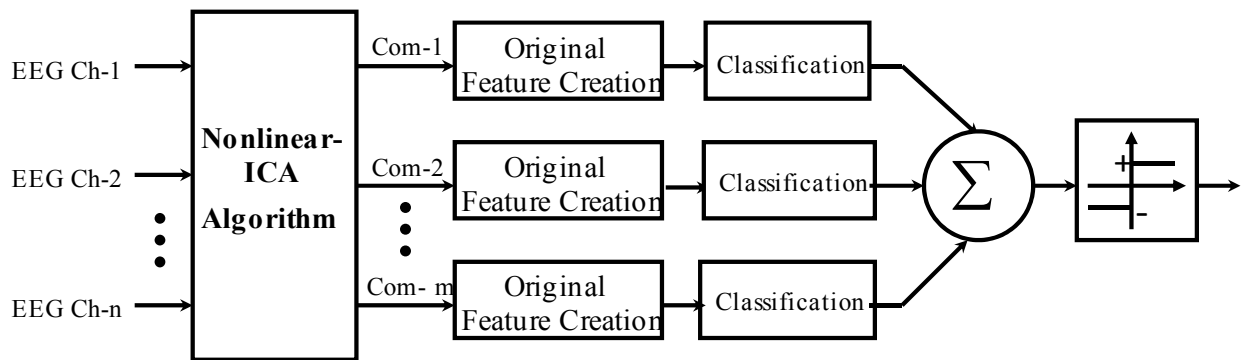
Fig. 7. The block diagram of classification process.

The results have been recorded for four subjects (AE, ME, BM, SN) for different experiment days. Table 1 summarizes the results of classification accuracy of the original EEG signals. The average classification accuracy is 73.84%.

Table 2 summarizes the results of classification accuracy for different subjects by using linear ICA. For these experiments, the Infomax algorithm (Hyvarinen et al., 2001) as a linear ICA has been used. The average classification accuracy over all subjects is 74.61% which 1% better than that obtained original EEG signals. An average classification rate of 77.95% is achieved by using nonlinear ICA. As can be observed, components which are obtained by nonlinear ICA improved the EEG classification accuracy compared to the linear ICA and original EEG signals. These results are 4 percent higher than average classification results by using the raw EEG data. Fig. 8 shows the classification accuracy rate obtained by nonlinear ICA (NICA), linear ICA (LICA), and original EEG signals (channel).
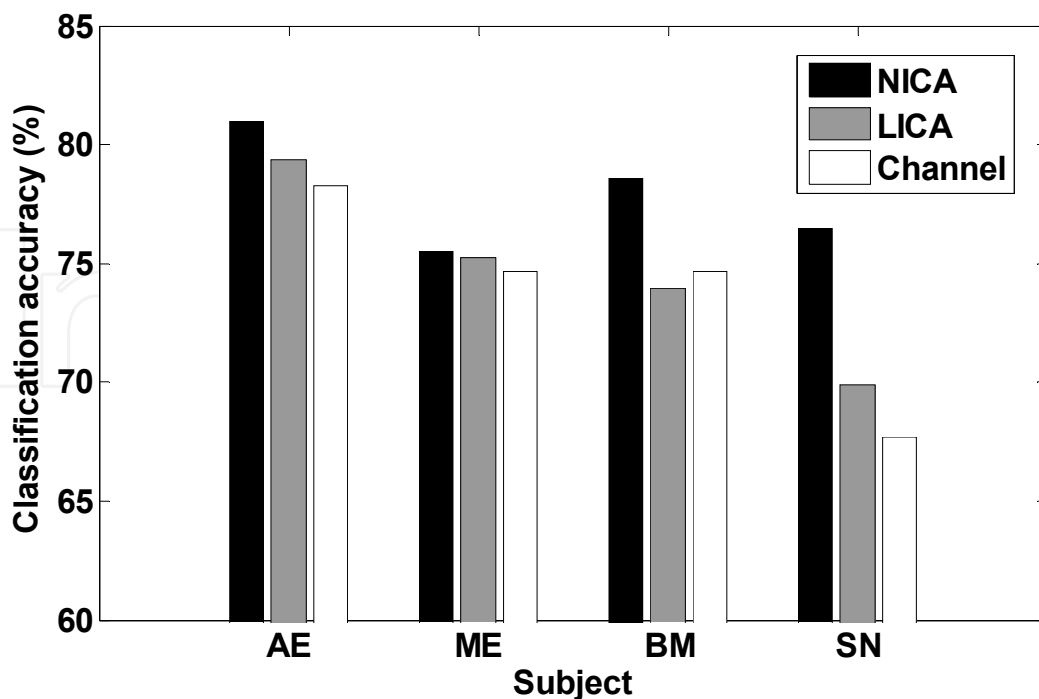

Fig. 8. Mean classification accuracy of EEG patterns for different subjects using nonlinear ICA (NICA), linear ICA (LICA), and original EEG signals (channel).

Subject	Day1	Day2	Day3	Day4	Day5	mean
AE	77.3	76.4	75.5	83.9	-	78.27
ME	65.3	84.9	74.6	73.8	-	74.65
BM	67.2	90.6	66	75.2	-	74.75
SN	77.4	66.1	61.6	69.4	64.1	67.7
mean	71.8	79.5	69.42	75.57	64.1	73.84

Table 1. Classification Accuracy Rate of Original EEG Signals During Hand Movement Imagination.

Subject	Day1	Day2	Day3	Day4	Day5	mean
AE	76.3	81.9	77.9	81.4	-	79.37
ME	68.7	84.1	77.2	71.1	-	75.27
BM	67.1	93.3	63	72.5	-	73.97
SN	78.9	71.1	64.1	67.6	67.6	69.86
mean	72.75	82.6	70.55	73.15	67.6	74.61

Table 2. Classification Accuracy Rate of Extracted Components During Hand Movement Imagination using Linear ICA.

Subject	Day1	Day2	Day3	Day4	Day5	mean
AE	77.6	81	80.1	85.3	-	81
ME	72.8	80.5	76.6	72	-	75.47
BM	76.2	93	69	76.2	-	78.6
SN	78.5	79	81.7	72.5	72	76.74
mean	76.28	83.38	76.85	76.5	72	77.95

Table 3. Accuracy Rate of Extracted Components During Hand Movement Imagination using Nonlinear ICA.

6. Conclusion

Preprocessing plays an important role in the performance of BCI systems. One of the preprocessors can be used to improve the performance of BCI systems is independent component analysis (ICA). ICA would appear to be able to separate task-related potentials from other neural and artifactual EEG sources during hand movement imagination in form of independent components. However, there is no guarantee for linear combination of brain

sources in EEG signals. Therefore, in this research a novel method was proposed for EEG signal classification in BCI systems by using non-linear ICA algorithm. The results of applying this method on four subjects have demonstrated that the proposed method in this research has improved the mean classification accuracies in relation to raw EEG data and linear ICA. The analysis of variance (ANOVA) shows that the mean classification accuracies achieved by using non-linear ICA are significantly different ($p < 0.01$).

7. Future directions

There are a number of directions in which the research described in this chapter can be extended. One area currently under investigation is to design a non-linear model for separation of nonlinearly mixed sources in the brain. As clarified in this chapter, a fundamental difficulty in nonlinear ICA is that it is highly non-unique without some extra constraints. Now, the question is which extra constraints can more compensate the information distorted by the non-linear combination of brain sources. Answering to this question will be our future work.

8. References

Achard, S.; Jutten, C. (2005). Identifiability of Post-Nonlinear Mixtures, *IEEE Trans. Signal Processing Letters*, vol. 12, no. 5, pp. 423-426.

Akbari, R. & Ziarati, K. (2010). A multilevel evolutionary algorithm for optimizing numerical functions, International Journal of Industrial Engineering Computations, Vol. 2, no. 2, pp. 419– 430.

Almeida, L.B. (2003). MISEP – Linear and Nonlinear ICA Based on Mutual Information, Journal of Machine Learning Research 4, pp. 1297-1318.

Gao, P.; Woo, W.L. & Dlay, S.S. (2006). Non-Linear Independent Component Analysis Using Series Reversion and Weierstrass Network, IEE Proc., *Vis. Image Process*, Vol. 153, no. 2, pp.115– 131.

Goldberg, D. E. (1989). *Genetic Algorithms in Search , Optimization and Machine Learning*. Addison-Wesley.

Hyvarinen, A.; Karhunen, J. & Oja, E. (2001). *Independent Component Analysis,* John Wiley & Sons.

Hyvarinen, A. & Pajunen, P. (1999). Nonlinear Independent Component Analysis Existence and Uniqueness Results, *Neural Network*, Vol. 12, no. 3, pp.429-439.

Jung, T.; makeig, S.; humphiers, C.; Lee, T.; Mckeown, M. & Sejnowski, T. (2000). Removing Electroencephalographic Arifacts by Blind Source Separation, *Psychophysiology*, pp. 163- 178.

Jung, T.; makeig, S.; Westerfield, M.; Townsend, J. ; Courchesne, E. & Sejnowski, T. (2001). Analysis and Visualizion of Single-Trial Event-Related Potentials, *Human Brain Mapping,* vol. 14, pp. 166-185.

Jutten, C. & Karhunen, J. (2004). Advances in Blind Source Separation (BSS) and Independent Component Analysis (ICA) for Nonlinear Mixtures, International Journal of Neural Sysytems, vol. 14, no. 5, pp. 267-292.

Krzanowski, W. J. (2000). *Principles of Multivariate Analysis: A User's Perspective*, Oxford University Press, Oxford.

Lappalainen, H. & Honkela, A. (2000). Bayesian Nonlinear Independent Component Analysis by Multi-Layer Perceptrons, *In Advances in Independent Component Analysis*, pp. 93-121.

Lotte, F. & Cuntai Guan (2011). Regularizing Common Spatial Patterns to Improve BCI Designs: Unified Theory and New Algorithms, *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 2, pp. 355-362.

Ortner, R.; Allison, B.Z.; Korisek, G.; Gaggl, H. & Pfurtscheller, G. (2011). An SSVEP BCI to Control a Hand Orthosis for Persons With Tetraplegia, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 1, pp. 1-5.

Oveisi, F. (2009). EEG Signal Classification Using Nonlinear Independent Component Analysis, *in Proceedings of the 34th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

Oveisi, F. (2009). Information Spectrum and Its Application to EEG-Based Brain-Computer Interface, *in Proceedings of the 4th International IEEE EMBS Conference on Neural Engineering*, Antalya, Turkey.

Oveisi, F. & Erfanian, A. (2008). A Minimax Mutual Information Scheme for Supervised Feature Extraction and Its Application to EEG-based brain-computer Interfacing, *EURASIP Journal on Advances in Signal Processing*, vol. 2008.

Oveisi, F.; Oveisi, S.; Erfanian, A.; Patras, I. (2012). Tree-Structured Feature Extraction Using Mutual Information, *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 23, no. 1, pp.127-137.

Peterson, D. A.; Knight, J. N.; Kirby, M. J.; Anderson, C. & Thaut, M. (2005). Feature Selection and Blind Source Separation in an EEG-Based Brain-Computer Interface, *EURASIP Journal on Applied Signal Processing*, vol. 19, pp. 3128–3140.

Rojas, F.; Puntonet, C.; Alvarez, M.; Rojas, I. & Clemente, R. (2004). Blind Source Separation in Post-Nonlinear Mixtures Using Competitive Learning, Simulated Annealing, and a genetic algorithm, *IEEE Trans. Systems, Man, and Cybernetics.*, vol. 34, no. 4, pp. 407-416.

Singer, A. & Coifman, R. R. (2007). Non-linear independent component analysis with diffusion maps, *Applied and Computational Harmonic Analysis*, vol. 25, pp. 226-239.

Tan, Y.; Wang, J. & Zurada, J. M. (2001). Nonlinear Blind Source Separation Using a Radial Basis Function Network, *IEEE Trans. on Neural Networks*, vol. 12, no. 1, pp. 124-134.

Van, L.; Wu, D. & Chen, C. (2011). Energy-Efficient FastICA Implementation for Biomedical Signal Separation, *IEEE Trans. on Biomedical Engineering*, vol. 58, no. 6, pp. 1865-1873.

Woo, W.L. & Sali, S. (2002). General Multilayer Perceptron Demixer Scheme for Nonlinear Blind Signal Separation, IEE Proc., *Vis. Image Process*, Vol.149, no. 5, pp.253–262.

Zhang, K. & Chan, L. (2007). Nonlinear Independent Component Analysis with Minimal Nonlinear Distortion, *24th International Conference on Machine Learning*, New York, USA.