

Independent component analysis: comparison of algorithms for the investigation of surface electrical brain activity

Matthias Klemm · Jens Haueisen · Galina Ivanova

Received: 25 March 2008 / Accepted: 21 January 2009 / Published online: 13 February 2009
© International Federation for Medical and Biological Engineering 2009

Abstract We compared the performance of 22 algorithms for independent component analysis with the aim to find suitable algorithms for applications in the field of surface electrical brain activity analysis. The quality of the separation is assessed with four performance measures: a correlation coefficient based index, a signal-to-interference ratio, a signal-to-distortion-ratio and the computational demand. Artificial data are used consisting of typical electroencephalogram and evoked potentials signal patterns, e.g. spikes, polyspikes, sharp waves and spindles. We evaluate different noise scenarios and the influence of pre-whitening. The comparisons reveal considerable differences between the algorithms, especially concerning the computational load. Algorithms based on the time structure of the data set seem to have advantages in separation quality especially for sine-shaped signals. Derivates of FastICA and Infomax also attain good results. Our results

can serve as a reference for selecting a task-specific algorithm to analyze a large number of signal patterns occurring in the surface electrical brain activity.

Keywords Independent component analysis · Electrical brain activity · EEG simulations · EEG processing · Blind source separation

1 Introduction

The methodology of independent component analysis (ICA) was first introduced in the early 1980s in the context of neural networks [10]. Since then a large number of methods were developed for application in, e.g. feature extraction, brain imaging, telecommunications and finance. Makeig et al. [3] utilized ICA as an important method for brain signal analysis. The goal of ICA as well as of the related blind source separation (BSS) is to separate instantaneously mixed signals into their independent sources without knowledge of the mixing process. It is possible to recover original sources from mixtures if they are independent of each other. There is no straightforward mathematical principle available to solve this problem. However, there are a large number of different algorithmic approaches. Algorithms are either based on higher order statistics or estimation of statistical parameters like negentropy, maximum likelihood or mutual information or rely on the temporal structure of the signals. To achieve separation, nearly all algorithms follow an iterative scheme until a stop criterion is reached.

One practical application of ICA decomposition is the electroencephalogram (EEG) and the evoked potential (EP) analysis. The acquired EEG/EP can be considered to result

Electronic supplementary material The online version of this article (doi:10.1007/s11517-009-0452-1) contains supplementary material, which is available to authorized users.

M. Klemm · J. Haueisen
Biomedical Engineering Department, Faculty of Computer Science and Automation, Institute of Biomedical Engineering and Informatics, Technische Universität Ilmenau,
P. O. Box 100565, 98684 Ilmenau, Thuringia, Germany

G. Ivanova (✉)
Department of Signal Processing and Pattern Recognition,
Biomedical Signal Processing, Faculty of Mathematics and Natural Sciences II, Institute of Computer Science, Humboldt-Universität zu Berlin, Unter den Linden 6,
10099 Berlin, Germany
e-mail: givanova@informatik.hu-berlin.de;
galina.ivanova@gmx.de

from a number of different sources in the brain and various artifact sources that simultaneously generate electrical signals. Thus, appropriate source localization is not trivial. The electrodes placed on the scalp measure a superposition or mixture of the underlying original sources. However, many clinical applications do not require a reconstruction of these original sources but instead a decomposition of the EEG/EP into spatio-temporal components (features). The ICA is predestined for such decomposition with possible applications for both feature extraction (selection of only interesting sources, e.g. heartbeat detection in electrocardiogram (ECG) data [24] or automated classification of epileptiform activity in EEG data [2]) and artifact removal (elimination of disturbing sources, e.g. removal of eye movement artifacts from the electroretinogram (ERG) [6] or removal of motion artifacts from electrocardiographic signals [17]).

The objective of our study is to determine the algorithms best suited for extensive EEG/EP signal analysis.

2 Materials and methods

2.1 Simulation premises

The investigations, done, were performed under the following premises:

- All ICA algorithms available to us at the time of the project were compared. Preconditions for algorithms are free availability and to be written in Matlab (or having a framework that allows integration into Matlab) with the open source code. We studied the following ICA algorithms (in no specific order): FastICA, efficient FastICA (EFICA), WASOBI, COMBI, MULTI-COMBI, JADE, SOBI, Acsobero, Kernel-ICA, TCA, RADICAL, MILCA, Infomax, SNICA, OGWE, SHIBBS, TDSEP, CUBICA, EGLD, Pearson-ICA, 1FICA and Block EFICA (see Supplementary material for author and source, respectively, reference information).
- Comparisons were made using a data set simulated, close to the real characteristics, and consequently consisting of signals with typically occurring groups of EEG/EP patterns and artifacts.
- Two simulations were performed concerning the influence of different noise models. In addition, another simulation was carried out concerning the influence of whitening as preprocessing prior to application of the ICA algorithms.
- “Sources” in the context of this paper refer to ICA sources and not underlying EEG/EP sources.

2.2 Comparison criteria

2.2.1 General considerations

To compare different ICA algorithms, characteristics and criteria of interest have to be defined. The general decision factors within this study are: separation quality and computational demand. The separation quality is certainly of top priority for the performance evaluation of the ICA algorithms. The computational requirements may also be an important criterion for algorithm selection, especially in the case of ICA methods with comparable separation quality, e.g. for conducting clinical studies with very large data sets. Eventually, a compromise between separation quality and computational demand has to be met.

2.2.2 Quality of source separation

In the context of this paper, quality is defined as the accuracy of the algorithm in the demixing of the sources in terms of the signal shape. The perhaps biggest problem and thus most influential factor to quality is that the ICA model does not entirely fulfill the assumption of independence. In practice, sources are often not absolutely statistically independent, prohibiting perfect separation. Over-determination of the ICA model, i.e. more sensors than sources, can pose another problem, however, this issue can be solved by applying, e.g. principal component analysis (PCA). If the opposite is the case and the ICA model is under-determined, i.e. more sources than sensors, only the strongest sources or superposition of sources can be separated. Ultimately, quality is influenced by noise since most ICA algorithms are designed and tested for noiseless cases.

Since the order of the sources can be altered, the quality quantification of an ICA algorithm is not trivial. An algorithm matching the original source with corresponding independent components is therefore necessary for direct comparison of the original source and the demixed source. This poses a difficult task as independent components might be heavily distorted. An additional complication is the loss of the original amplitude of the sources due to varying values of the scaling factor between sources and independent components.

2.2.2.1 Correlation based criterion The Spearman correlation coefficient r [9] proves to be a good choice to compare the original source and the independent component because it is not dependent on the absolute amplitude but instead on the (relative) shape of the signal and shows normalized results between 0 and 1 (the sign is not of importance here). The formula is:

$$r \equiv 1 - 6 \sum \frac{d^2}{N(N^2 - 1)} \tag{1}$$

whereby d is the difference in statistical rank of the corresponding variables and N is the number of pairs of values.

We choose the nonparametric Spearman correlation coefficient instead of the Pearson correlation coefficient [9] since the latter shows correct results only for normal (Gaussian) distributions which are not given in this case as ICA requires nongaussianity. Correlation coefficients are calculated against all independent components for each source. The independent component with the highest absolute coefficient is the sought match. We refer to this quality as ‘‘Correlation based Index’’ (CBI). It is computed for each ICA algorithm by summing up the correlation coefficients for each independent component and then computing its mean value. To some extent, this allows compensation of badly detected sources with very good separations. A problem arises if an ‘‘independent component’’ is a mixture of two or more sources. Consequently, the correlation coefficient might turn out to be maximal at the same ‘‘independent component’’, thus compromising the CBI since its value would be too high. This is prohibited by removing matched independent components from the pool.

2.2.2.2 Criterion based on signal-to-interference ratio

Another measure for the separation quality of an ICA algorithm is the signal-to-interference ratio. Based on Xu et al. [25] we define the SIR for the k th original source as:

$$P = W \times B \tag{2}$$

$$SIR_k = 10 \log_{10} \left(\frac{\max(P(:, k))^2}{P(j, :) \cdot P(j, :)^T - \max(P(:, k))^2} \right) \tag{3}$$

whereby B is the original mixing matrix and W is the estimated demixing matrix. If B equals W^{-1} then P becomes the identity matrix, otherwise P is roughly a permutation matrix. Here j is the row where $\max(P(:, k))^2$ occurs. This SIR is the ratio of signal power of the estimated source and total power of the interfering signals, measured in decibels (dB).

The SIR criterion can only be used for theoretical comparison, as the true mixing matrix is required for calculation. On the other hand, the correlation based CBI criterion is solely based on a comparison of true and estimated sources. A similar approach is very often used in practical ICA applications where one or more reference signals are known and the corresponding estimated sources should be determined, making our CBI criterion more significant for an application oriented evaluation.

2.2.2.3 Criterion based on source-to-distortion ratio

Vincent et al. [22] introduced a number of measures

to evaluate blind audio source separation (BASS) algorithms. One of the considered BASS methods is the independent component analysis. Therefore, the performance measures, described in their paper, can be applied for the purposes of our study too. Vincent et al. implemented their measures in a freely available Matlab toolbox, which allows for a straightforward integration into our benchmark framework. We compute performance criteria in case that the only allowed distortions are time-invariant gains and chose therefore the source-to-distortion ratio (SDR) for our investigations.

The matching of original and estimated sources is done similarly to the correlation based criterion. The BASS criterions for each estimated source are computed against all original sources. The one that gives the best results, is the sought ‘‘true source’’ as described in [22].

Furthermore and differentially from [22] we compute the SNR using the following definition:

$$SNR = 20 \log_{10} \left(\frac{\max(signal) - \min(signal)}{\max(noise) - \min(noise)} \right) \tag{4}$$

whereby with *signal* is the original source, and with *noise* is the noise vector defined for the specific simulation, denoted. To obtain the SNR for a complete dataset, the SNR is computed for each source and averaged afterwards.

2.2.3 Computational demand

As stated above, the demand of computational resources is another very important criterion for the application of an ICA algorithm. Considering the present attributes of typical computer systems we determine the required CPU time as the only crucial factor of computational resource for algorithms since all other aspects, like hard disk space or size of random access memory (RAM), do not pose any limitations in this study. The required execution time, though, can vary by factors of more than one thousand for the different algorithms. Measuring the execution time proves to be the more realistic and useful measure for practical application since counting the required floating point operations per second (FLOPS) only shows part of the computational resources an ICA algorithm consumes. Memory bandwidth, for example, is another important resource because ICA algorithms work on large data matrices and typically perform a lot memory reading, writing, allocation and de-allocation, etc. Operations may require a considerable amount of execution time but not many FLOPS. Furthermore execution time serves as an estimate of required analysis time for comparable data sets. Then again, FLOPS are a system independent measure while measured execution time is only valid for the specific test system. Tests with different Intel and AMD based systems reveal no significant differences in relative

execution times of algorithms. As expected, absolute execution times vary.

The required CPU time is of equal importance when considering online application of algorithms. In this context, “online” could stand for trial based signal processing during the interval between recordings of particular trials. This waiting time is typically short, i.e. a few seconds. A classic procedure in EEG studies is to perform a fixed number of trials for given paradigm to improve the signal-to-noise ratio. If the SNR is determined in the very short time between trials, recording can be stopped when a pre-determined SNR is reached. This keeps the number of trials as low as possible, reducing the stress on the test person and verifying signal quality during measurement. Consequently, an ICA algorithm suitable for online application has even higher constraints concerning required execution time.

2.3 Data

True sources and mixing process of real EEG/EP data sets, particularly in pathological cases, are unknown and not very well suited for an objective comparison. Therefore, we decide to create an artificial data set including known test signals emphasizing the common structures of real EEG/EP signals, e.g. a sine with phase shifts or phase jumps, a number of graphoelements like spikes, polyspikes, sharp waves as well as spindles and signals that represent certain phenomena such as evoked potentials and event related desynchronization or synchronization (ERD/ERS). The amplitudes of typical EEG/EP signals are at rather small values between 1 and 100 μV . The synthetic data set

for our investigations is shown in Fig. 1 and described Table 1. The signals are supposed to correspond to typical, underlying EEG ICA sources in terms of shape and amplitude. The dataset consists of 16 channels, the minimum required for practical EEG recordings. Parameters and shapes of the test signals are based on Ebe et al. [5] and Rodenbeck et al. [19]. We used an identical randomly generated mixing matrix for each algorithm. Each element of the mixing matrix is a double precision floating point number between 0 (total dampening of the influence of a certain source for the specific mixture/electrode channel) and 1 (signal not dampened). The assumption of a random and thereby unstructured mixing matrix leads to a very abstract view of the mixing process. If the model should have been more realistic from a neurophysiologic point of view, the concurrent occurrence of certain sources would have to be excluded. Moreover other sources would have to be temporally and spatially aligned resulting in a structured mixing matrix. Due to the random mixing matrix we model a more challenging situation (or even the worst case), when the sources may occur randomly, occasionally overlapping in time and in close spatial proximity.

2.4 Computation

One of the intentions is to obtain representative results from our simulations. An ICA algorithm is a statistical procedure, which results may depend on its random initialization. Therefore, the data set described in Sect. 2.2 is repetitively generated and processed ten times by each algorithm. Those iterations are not related to the generation of ensemble for

Fig. 1 Visualization of the test data set. The histograms of each source of our artificial data set are shown on the right. Each histogram consists of 8 bins. The corresponding short description of each signal is given in Table 1

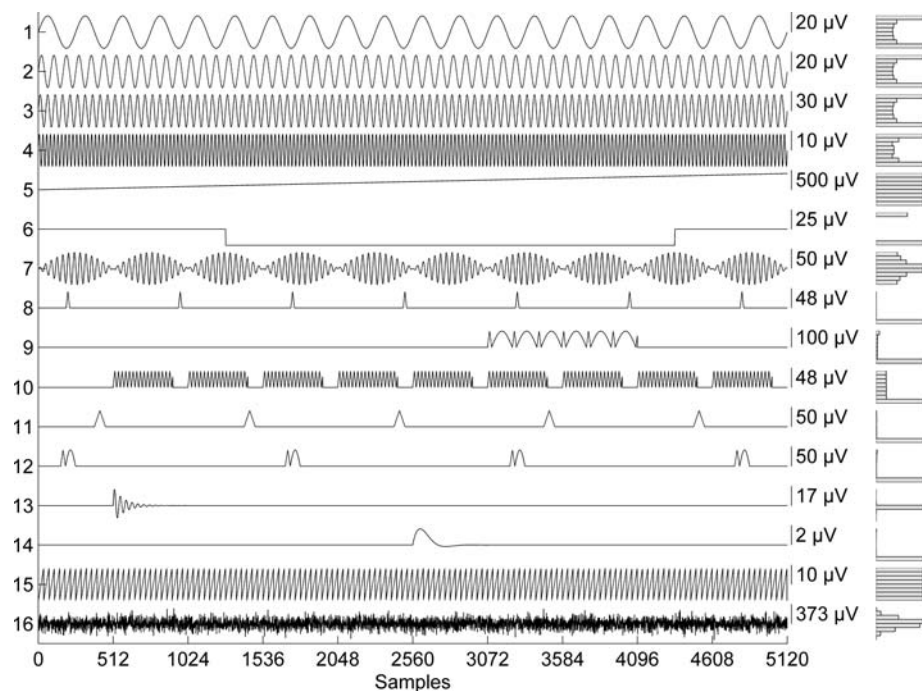


Table 1 Test data set characteristics

Signal	Description	Kurtosis
1	Delta-band sinus wave (2 Hz)	-1.500
2	Theta-band sinus wave (6 Hz)	-1.500
3	Alpha-band sinus wave (10 Hz)	-1.500
4	Beta-band sinus wave (20 Hz)	-1.500
5	Ramp (electrode drift)	-1.200
6	Contingent negative variation (CNV)-shaped	-1.834
7	Sleep spindles (13 Hz base frequency, 0.5 Hz modulation frequency)	-0.750
8	Spikes	47.687
9	Spike-wave-complex	3.079
10	Polyspikes	-1.260
11	Sharp-waves	19.421
12	Slow-sharp-waves	15.439
13	Event related desynchronization (ERD)-shaped	99.623
14	Weak evoked potential	45.291
15	Saw tooth wave	-1.200
16	Strong Gaussian noise	0.016

The kurtosis of a Gaussian distribution is 0

subsequent epoch averaging, e.g. in case of evoked potentials with negative SNR, but to attain characteristic results for each ICA algorithm. Because of the time constraints and the very long runtimes Kernel-ICA, TCA and both MILCA implementations processed the data set only twice. Finally we averaged the repetition results for the four defined criteria (Sect. 2.1), whereby CBI, SIR and SDR serve as the quality criteria and the required computation time as the computational demand benchmark. The CPU time is obtained with Matlab's built-in function that shows the consumed CPU time and not the actual running time so possible influences of simultaneously running external processes are virtually eliminated. Unfortunately, this procedure is not possible with the algorithms MILCA and SNICA because the main part of their implementations is written in C and executed as an external process. The computation time for these cases has to be measured by using the system clock. The recorded CPU times are still very accurate as the running times of both algorithms are relatively long and the test system used is a dual processor system. Hence, possible influences by other programs running in the background are very small. No other processes were started during the benchmark. The C programs of MILCA and SNICA are not able to utilize the second CPU core of the test system. This would have shortened the recorded running time. The measured CPU time is solely the computation time of the specific ICA algorithm. Preparation of the data, possible preprocessing- or display-

operations have no influence on the recorded time. MILCA and SNICA did not work under Windows Vista SP1 64 bit (32 bit version was not tested).

The computer system utilized is equipped with an Intel Core2 E6600 processor (dual-core CPU, 2×2.4 GHz, 4 MB L2 Cache), 2,048 MB of DDR2 memory (533 MHz) running Windows XP SP2 (32 bit version) and Matlab R2007b (Version 7.5.0342). The multithreaded computation was disabled, because our goal was not to benchmark Matlab's capability to distribute the ICA workload onto multiple CPUs, but the actual computational demand of the algorithms. By using the Microsoft Windows Task Manager, we verified that the amount of RAM memory did not pose a limitation to any of the ICA algorithms for the utilized data set. The parameters for each algorithm are listed in the Supplementary material.

3 Results

For sensor noise simulations, a method adding a zero-mean Gaussian random vector to each synthetic source is chosen. This is a deliberate violation of the requirement to apply ICA. We want to investigate how the ICA algorithms deal with more than one source with Gaussian signal components (only one Gaussian source is allowed for ICA per definition [10]). Thus three simulations with a standard deviation of $0.1 \mu\text{V}$ (mean SNR: 36.51 dB), $0.2 \mu\text{V}$ (mean SNR: 30.43 dB) and $0.4 \mu\text{V}$ (mean SNR: 24.57 dB) for the additive noise were performed. A standard deviation of $0.2 \mu\text{V}$ is a typical value obtained by in-house measurements of sensor noise. As mentioned in Sect. 2.2 typical EEG amplitudes are of much larger values at about $1\text{--}100 \mu\text{V}$. Although the amplitude of the additional noise vector is quite low, this scenario with many Gaussian noise sources is rather extreme for an ICA algorithm. However, in addition to the sensor noise there might be a lot of different noise sources under real life conditions as well. No PCA or whitening is done as a preprocessing step. The CBIs for each algorithm and source for the simulation with a standard deviation of $0.2 \mu\text{V}$ for the noise vectors are shown in Fig. 2. This visualization allows an investigation of the performance of each algorithm for different sources. The required CPU times as the results of the simulation for the computational demand criterion are given in Table 2. The CBIs in Fig. 2 and Table 2 make evident that SNICA achieves the lowest average results. TDSEP and Acsohiro also achieve low results. Kernel-ICA and EGLD perform better than SNICA, TDSEP and Acsohiro but are still significantly worse compared to the remaining algorithms. SOBI and WASOBI deliver the best separation quality for the CBI criterion.

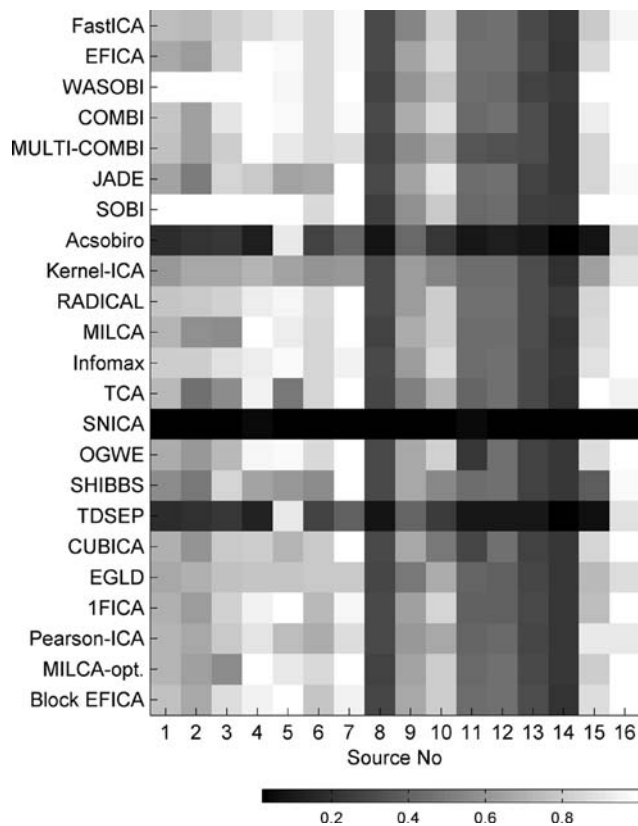


Fig. 2 Source–algorithm plot of the CBI criterion averaged, over the simulation trials with additive noise (0.2 μV standard deviation). The *white* color of a *rectangle* for a specific source (*horizontal axis*) and algorithm (*vertical axis*) reflect the maximum CBI of 1.0 and thus the perfectly matching of the original and the estimated source. Oppositely, the *black* color stands for the minimum CBI of 0.0, i.e. the ICA algorithm totally failed to extract the specific source. For example, the SOBI algorithm delivers very good results for sources 1–5, 7 and 15

Figure 3 displays the simulation results for the SIR criterion. WASOBI achieves the best average SIR of all algorithms (18.1 dB). For SNICA the SIR computation fails because of a bad demixing matrix.

Figure 4 graphs the results for the SDR performance measure. The findings are similar to the SIR results whereby SOBI attains the highest average SDR of 18.2 dB.

Furthermore the influence of power-law noise was examined. In contrast to the previous simulations, no noise is added to the sources before mixing, leaving source number 16 as the only noise origin in this case. Here, the distribution of the noise is not Gaussian but power-law ($1/f$) with mean SNR of -25.3 dB. Since the noise amplitude can be raised significantly without much of an effect, this simulation appears to be somewhat easy for the ICA. Again, the SNICA algorithm achieves the lowest average values for CBI, SIR and SDR; SOBI and WASOBI deliver the best separation quality (Table 2).

Additionally, the effect of PCA and whitening as preprocessing were studied in this simulation. Here, PCA does

not effect dimension reduction since there are no redundant mixtures which can be removed. The dataset and simulation parameters of the previous simulation were used. This is the only simulation in which TDSEP and Acsobiro generate better results. Both algorithms seem to be the only ones in the test field which rely on PCA and whitening applied to the data set. The Pearson-ICA algorithm gains a computational speed increase of about 30% from the additional preprocessing while MULTI-COMBI, Kernel-ICA, Infomax, TCA, CUBICA and EGLD attain slightly lower results in terms of CBI, SIR and SDR.

To evaluate the influence of power-law distributed noise in comparison to Gaussian noise we repeated the last simulation using a Gaussian noise source. This corresponds to the very often applied assumption that all Gaussian noise sources can be merged into a single source. The COMBI algorithm seems to benefit from this modification as it achieves for all three quality criteria the best results.

The results of the different simulations are summarized in Tables 2 and 3.

4 Discussion

4.1 Literature evaluation

A large number of different ICA algorithms have been developed. Several papers on comparisons of ICA algorithms have been published.

Usability for artifact removal from multi-channel EEG was evaluated for the ICA algorithms FastICA and Infomax in a publication by Glass et al. [8]. A stream of blinks with known spatial and temporal characteristics (blink template) was added to manually selected blink free segments. The estimated source with the largest correlation coefficient for the blink template was removed and the cleaned data sets were compared to the contaminated sets using correlation coefficients. However, this study did not take into consideration the computational requirements of the algorithms.

Another study was undertaken by Nicolaou et al. [18]. The focus of this study was to compare temporal and common ICA algorithms, using both real and simulated data sets. The data set was comprised of two EEG channels, an electromyogram (EMG) and an electrocardiogram (ECG). The only criterion for comparison used, was the signal-to-interference ratio (SIR) index as the computational demand was not investigated.

Krishnaveni et al. [14, 15] compared in two papers a large number of ICA algorithms on the basis of a mutual information estimator, using EEG data sets containing ocular artifacts but did not examine the computational demand. The algorithms Infomax, Extended Infomax,

Table 2 The outcome of all three simulations (mean values of ten iterations), the numerical rankings are given in curled brackets

Algorithm	Additive noise				Separated noise source				With preprocessing			
	Gaussian noise standard deviation: 0.2 μ V				1/f noise standard deviation: 100 μ V				1/f noise standard deviation: 100 μ V			
	CBI	SIR (dB)	SDR (dB)	t_{CPU} (s)	CBI	SIR (dB)	SDR (dB)	t_{CPU} (s)	CBI	SIR (dB)	SDR (dB)	t_{CPU} (s)
FastICA	0.67 {8}	2.69 {11}	9.70 {11}	3.05 {15}	0.67 {11}	3.51 {10}	13.01 {2}	2.17 {15}	0.66 {13}	2.72 {10}	10.49 {6}	2.76 {15}
EFICA	0.69 {6}	8.60 {5}	12.15 {5}	0.78 {9}	0.68 {10}	11.13 {5}	9.84 {7}	0.93 {11}	0.69 {8}	12.22 {5}	10.19 {7}	0.68 {8}
WASOBI	0.74 {2}	18.10 {1}	17.22 {2}	0.18 {2}	0.73 {1}	14.17 {1}	10.91 {5}	0.21 {2}	0.73 {1}	14.71 {1}	14.48 {1}	0.19 {2}
COMBI	0.71 {3}	10.40 {3}	13.81 {3}	1.02 {11}	0.72 {3}	13.60 {3}	12.27 {3}	0.92 {10}	0.69 {7}	12.68 {3}	13.65 {3}	1.14 {12}
MULTI-COMBI	0.65 {13}	7.66 {6}	7.96 {14}	2.05 {14}	0.64 {13}	8.50 {7}	9.14 {8}	1.78 {14}	0.67 {12}	9.64 {7}	9.90 {9}	2.11 {14}
JADE	0.64 {14}	-0.52 {16}	3.29 {18}	0.74 {7}	0.61 {17}	-0.49 {15}	1.71 {17}	0.90 {9}	0.61 {17}	-0.49 {15}	2.40 {19}	0.67 {7}
SOBI	0.74 {1}	13.43 {2}	18.20 {1}	0.38 {4}	0.73 {2}	13.65 {2}	16.28 {1}	0.41 {5}	0.72 {2}	13.42 {2}	14.38 {2}	0.39 {5}
Acscobiro	0.26 {22}	-11.20 {22}	-13.39 {22}	0.39 {5}	0.25 {21}	-10.12 {21}	-13.76 {21}	0.40 {4}	0.48 {21}	-15.73 {22}	-5.27 {22}	0.37 {4}
Kernel-ICA	0.56 {20}	-1.61 {19}	3.42 {17}	1,939.11 {20}	0.55 {20}	-1.55 {17}	8.25 {9}	1.07447 {20}	0.59 {19}	-0.57 {16}	5.01 {13}	1.42174 {20}
RADICAL	0.70 {5}	9.24 {4}	12.87 {4}	225.95 {18}	0.70 {4}	2.07 {11}	10.62 {6}	240.96 {19}	0.71 {3}	2.26 {11}	12.11 {4}	224.29 {18}
MILCA	0.67 {10}	2.61 {12}	9.94 {10}	21,657.06 {23}	0.69 {7}	-2.79 {18}	-3.52 {20}	24,507.06 {22}	0.68 {10}	-5.73 {19}	7.83 {10}	18,923.03 {22}
Infomax	0.71 {4}	3.17 {10}	10.38 {9}	14.08 {17}	0.70 {5}	3.72 {9}	4.65 {13}	15.87 {17}	0.70 {5}	3.68 {8}	7.41 {11}	15.26 {17}
TCA	0.62 {17}	3.65 {9}	0.81 {20}	21,341.04 {22}	0.62 {16}	4.95 {8}	0.78 {19}	28,681.11 {23}	0.60 {18}	3.65 {9}	3.15 {17}	21,794.03 {23}
SNICA	0.03 {23}	NaN {23}	-23.42 {23}	0.83 {10}	0.03 {23}	NaN {23}	-23.73 {23}	0.79 {8}	0.03 {23}	NaN {23}	-23.73 {23}	0.72 {9}
OGWE	0.67 {9}	0.51 {15}	11.13 {8}	0.25 {3}	0.69 {6}	1.88 {12}	5.55 {12}	0.23 {3}	0.68 {9}	1.83 {12}	5.89 {12}	0.24 {3}
SHIBBS	0.56 {19}	-1.48 {18}	1.46 {19}	0.78 {8}	0.55 {19}	-1.00 {16}	3.04 {15}	0.65 {7}	0.56 {20}	-0.99 {17}	3.11 {18}	0.73 {10}
TDSEP	0.27 {21}	-10.44 {21}	-13.24 {21}	0.02 {1}	0.24 {22}	-10.46 {22}	-13.98 {22}	0.01 {1}	0.48 {22}	-13.94 {21}	-3.39 {21}	0.02 {1}
CUBICA	0.62 {16}	-0.72 {17}	8.50 {13}	0.51 {6}	0.62 {15}	0.54 {14}	6.01 {11}	0.45 {6}	0.65 {14}	0.74 {14}	4.97 {14}	0.58 {6}
EGLD	0.60 {18}	-2.05 {20}	3.46 {16}	268.52 {19}	0.61 {18}	-6.57 {20}	2.67 {16}	238.05 {18}	0.61 {16}	-5.35 {18}	4.04 {15}	259.71 {19}
IFICA	0.67 {11}	6.65 {7}	11.68 {6}	1.44 {13}	0.66 {12}	11.46 {4}	11.68 {4}	1.36 {12}	0.70 {6}	12.24 {4}	10.95 {5}	1.11 {11}
Pearson-ICA	0.64 {15}	0.83 {14}	5.85 {15}	8.17 {16}	0.63 {14}	0.57 {13}	4.12 {14}	11.29 {16}	0.63 {15}	1.07 {13}	3.64 {16}	8.72 {16}
MILCA-opt.	0.66 {12}	1.96 {13}	9.49 {12}	10,486.86 {21}	0.68 {9}	-5.87 {19}	0.93 {18}	12,278.73 {21}	0.67 {11}	-6.71 {20}	0.99 {20}	11,132.11 {21}
Block EFICA	0.69 {7}	6.05 {8}	11.39 {7}	1.23 {12}	0.69 {8}	10.16 {6}	7.95 {10}	1.40 {13}	0.70 {4}	10.67 {6}	9.92 {8}	1.20 {13}

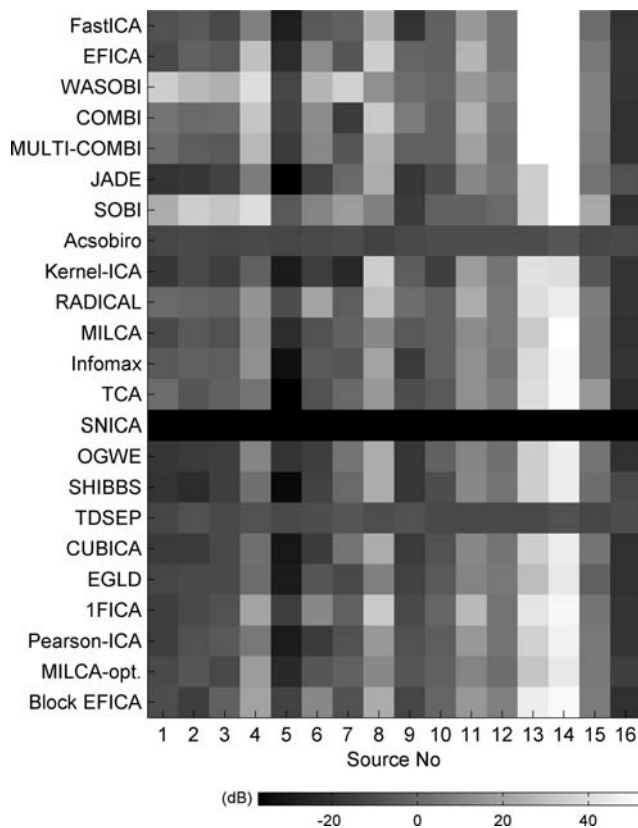


Fig. 3 Source–algorithm plot of the SIR criterion in dB, averaged over the simulation trials with additive noise ($0.2 \mu\text{V}$ standard deviation). Analog to Fig. 2 and analog to the well known SNR, *black* represents a very low ratio of wanted source signal to unwanted interference. *White* stands for a high SIR, indicating good separation quality. For example, the WASOBI algorithm delivers very good results for sources 1–4, 8, 13 and 14

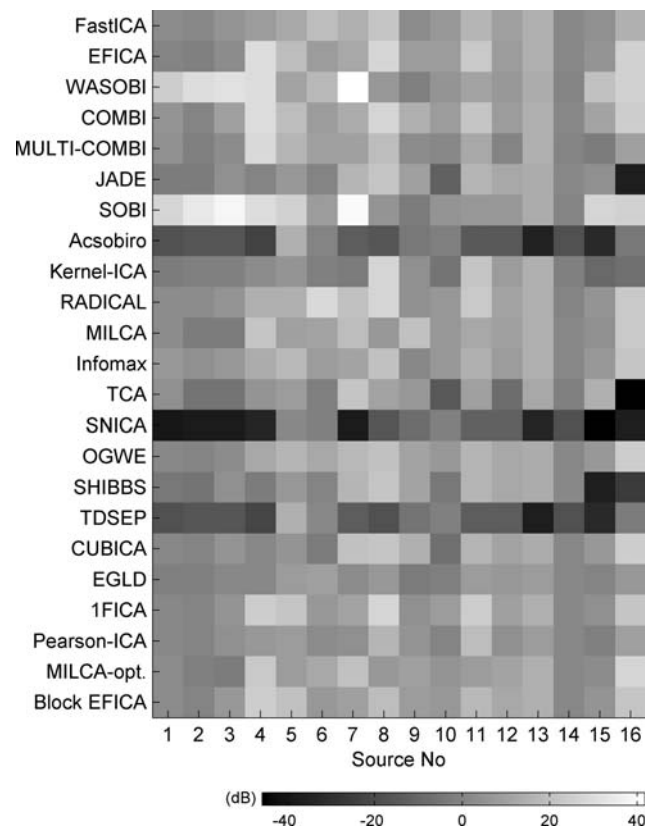


Fig. 4 Source–algorithm plot of the SDR criterion in dB, averaged over simulation trials with additive noise ($0.2 \mu\text{V}$ standard deviation). Similar to Fig. 3 *black* represents a very low ratio of wanted source signal to unwanted distortion. *White* stands for a high SDR, indicating good separation quality. Again the SOBI algorithm achieves good results for sources 1–5 and 7

FastICA, SOBI, TDSEP, JADE, OGWE, MS-ICA, SHIBBS, Kernel-ICA, RADICAL and MILCA were analyzed.

A topical review of James et al. [11] describes the technique of ICA as a method performing BSS in the context of biomedical signal processing. Although the scope of this article is not a comparison of ICA algorithms, it contains helpful information on various approaches applicable to solve the BSS problem in the field of biomedical signals. No part of this article solely investigates the application of ICA on other biomedical signals, e.g. magnetoencephalogram (MEG) data [7], sound data [16] and magnetocardiogram (MCG) data [4].

Though all the above mentioned studies show valuable results, none were carried out with the objective of finding outcomes for practical biomedical signal processing, especially for the analysis of huge amount EEG/EP data. Often, the few cases which investigated physiological EEG signals used very small data sets or a small number of channels. Furthermore, evaluation of algorithms with real EEG data sets was based on subjective criteria since the

original sources were unknown. Only very few algorithms (very popular algorithms like FastICA and Infomax) were compared in most papers and none provided a widespread overview on available implementations considering the latest developments.

4.2 Result evaluation

In the following section the results of the simulations are being analyzed. The good performance of the SOBI-algorithms is most probably due to the fact that these algorithms rely on the time structure of the signals. Given the fact that test signals used in this study focused on typical time characteristics, implementation of this approach proved to be of advantage here. Performance results should be applicable in a practical EEG analysis as well. Likewise, OGWE, RADICAL, EFICA, COMBI, Infomax and CUBICA are only slightly inferior to SOBI and WASOBI by a difference of about 0.05 or less for the CBI criterion. Figures 2, 3 and 4 depict those specific sources

Table 3 Kendall's tau for the quality measures

	Kendall's tau	Additive noise Gaussian noise standard deviation: 0.2 μV	Separated noise source 1/f noise standard deviation: 100 μV	With preprocessing 1/f noise standard deviation: 100 μV
A tau of 1.0 indicates a strong correlation between the measures while 0 denotes their independency	CBI–SIR	0.76	0.61	0.63
	CBI–SDR	0.83	0.49	0.71
	SIR–SDR	0.74	0.67	0.73

for which SOBI and WASOBI achieve better separation than the remaining algorithms and thus could be used by researchers requiring a suitable algorithm for a given signal decomposition problem (e.g. epileptic spikes). Sources 1–3, sine waves at dissimilar frequencies, illustrate this case. SOBI and WASOBI can be used for a near perfect estimation. One reason for the inferior performance of the other algorithms might be the chosen nonlinearity (source prior). Sine waves have multimodal probability density functions (see Fig. 1, histograms on the right side) while for ICA typically high-kurtosis source priors are used, as pointed out by Knuth [12].

Required CPU times differ greatly. Surprisingly, one of the two C implementations (MILCA) belongs to the slowest algorithms in the test-field, performing slower than most other algorithms of comparable quality (roughly by a factor of ten thousand). Therefore, we recompiled the source code using Microsoft Visual Studio.NET 2005 and the latest Intel C(++) compiler with all optimization flags turned on and also utilizing SSE2 processor instructions. The source code was not altered. We wanted to briefly investigate achievable performance improvements of those very easily attainable optimizations. We refer to the recompiled version of the MILCA algorithm as “MILCA-optimized”. Compared to the original binary, our recompiled binary performed up to 100% faster but is still too slow to be considered for practical application in EEG.

Kernel-ICA's separation results (average CBI of about 0.5–0.6) and especially the required CPU time (about 1900s) make this ICA variant rather inapplicable for our purposes. A relatively slow algorithm is RADICAL (about 200 s). It delivers good results in terms of quality but the execution time is too long to be of use for biomedical signal processing of large data sets. Virtually the same is the case with Infomax which is considerably faster than RADICAL (less than 15 s execution time), but still roughly ten to fifty times slower than other algorithms. OGWE is able to accomplish separation on par with most algorithms of the test-field and is even quicker than the already very fast SOBI algorithm (0.16 vs. 0.38 s). The fastest algorithm in all three simulations is TDSEP (about 0.01 s). However, achieved separation quality is comparatively low (at best an achieved CBI of 0.5) and this algorithm can only be recommended if the computational demand is of top priority and PCA and whitening are applied as preprocessing.

The degree of correspondence between the rankings, based on the three separation quality criteria, calculated using the Kendall tau coefficient [1] is strong (Table 3). The only exception is the correspondence between CBI and SDR in the simulation of separated noise and without preprocessing. The high decision correspondence can be interpreted as verification of the performance results of the different ICA algorithms.

Figure 5 shows the results of the SOBI ICA algorithm for simulation with a separated noise source. It indicates that signals are very close to original source signals (Fig. 1). The signs of some signals are reversed and some signals have minimal influences from other sources. As already mentioned, SOBI performs particularly well in separating sine waves of dissimilar frequencies. Most other ICA algorithms are not as successful.

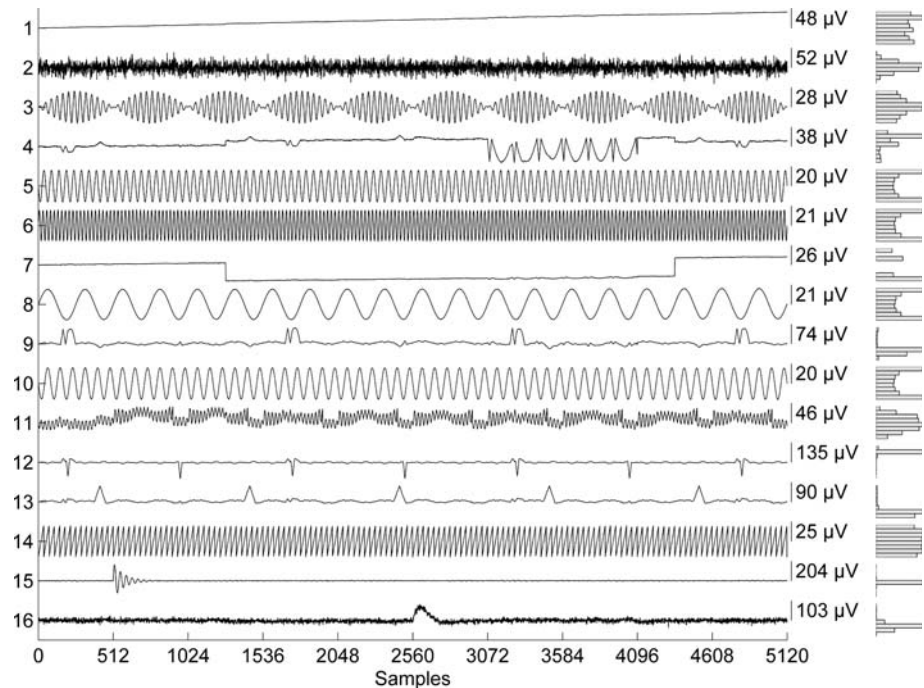
It also becomes clear that the order of the signals is different in both figures, i.e. the estimated sources are permuted. The question is how successfully the estimated sources are matched to the corresponding original sources by the different separation quality criteria as each criterion uses its own methodology. An inspection of the detected matches confirmed that all performance measures found the correct sources in this example.

4.3 Limitations of the simulations

It is important to point out that the evaluations consider only the special case of scalp-recorded EEG and EP. We are not considering intracranial EEG or magnetoencephalography data as the signal-levels, noise-levels and noise features are distinctly different in these recording modalities. Further on, the physical layout of the acquisition arrays compared to the physical distribution of the neural sources is distinctly different between scalp-recordings and intracranial EEG.

A compromise between a pure statistical comparison of ICA algorithms and a comparison based on real EEG data using our approach of simulation and an artificial dataset was sought. As already stated, the problem with the former is the difficult predictability and validation of the actual outcome for practical applications. The problem with the latter is that the true sources and the mixing process are unknown so that the results of the comparison cannot be objectively evaluated. This compromise also proved to be a

Fig. 5 Visualization of estimated sources from the SOBI algorithm. The result is very good as all original sources (Fig. 1) can be clearly recognized. Two limitations of the ICA become evident: the order and the absolute amplitude (including the sign) of the sources are lost. Similar to Fig. 1, the histograms of each source of our artificial data set are shown on the right side of this figure. Eight bins were used to create the histograms



limitation of our evaluation. Furthermore, real EEG is more complex than test signals used in this study and consists of more than 16 sources.

Another limitation of the benchmark is related to the CBI criterion but not exclusively. Sources with large parts of zero activity (e.g. sources 8, 11, 12, 13 and 14 in Fig. 1) are problematic for computing the correlation coefficient. Very small deviations of the estimated sources in those zero valued signal parts cause a low correlation coefficient although relevant (non-zero) parts of the signal are estimated accurately. Figures 2, 3 and 4 illustrate this effect. All algorithms attain quite low results for these mentioned sources. The problem is further revealed when comparing an original source, e.g. source 8 in Fig. 1 and the corresponding estimated source of the SOBI algorithm in Fig. 5 (source 12). Although both sources match quite well visually, the achieved correlation coefficient is only about 0.26. Due to the facts that the performance measures rely on averaged statistical information and our simulations are epoch-oriented, shorter epochs would probably improve the values of the measures in the case of short-time sources with seldom occurrence. Overall, this limitation does not lead to false results. However, sensitivity of the measures for specific sources is lowered. Conclusively, inter-algorithm differences are crucial for comparison, not absolute values.

In accordance with Giannakopoulos et al. [7], we found that fixed-point algorithms (e.g. FastICA and its derivatives) deliver good performance. We can also confirm that algorithms based on a temporal structure, such as the SOBI and WASOBI algorithms as well as TDSEP used in our study, are beneficial for EEG analysis [18].

Krishnaveni et al. [14, 15] compared ICA algorithms on the basis of mutual information [23], an approach that does not make assumptions about data used, to measure the dependence of components of a random vector. This makes mutual information a perhaps suitable measure for separation quality, provided that independence of source signals is assumed.

There are a number of estimators available to help obtain mutual information, e.g. Stogbauer et al. [20]. Unfortunately, there are remarkable differences between the results of the different findings. This made it difficult to choose a single estimator for our study and also complicated interpretation of the results. Since the aim of this paper is not the comparison of mutual information estimators and due to imposed time constraints, we decided not to implement mutual information as a quality criterion.

Finally, specialized source separation algorithms, e.g. dVCA [13, 21] as it relies on multiple (non-averaged epochs) of single-trial data, are beyond the scope of this article.

5 Conclusion

Despite the limitations discussed above, the study carried out shows that certain ICA algorithms appear to be more applicable for EEG/EP analysis than others.

SOBI and WASOBI showed the best results for separation quality in most cases. EFICA, which is an improved version of FastICA, nearly matched the performance of SOBI and is clearly superior to its original version in our

simulations. OGWE and CUBICA, both fourth order statistics based algorithms, also attain good results. COMBI, an approach to unite EFICA and WASOBI, should combine the advantages of FastICA and a SOBI derivative and thus extend its field of applications. Although the algorithm falls slightly behind in terms of quality and computational demand (the algorithm is nevertheless very fast), it should be considered for practical application, and especially since the combination approach might show advantages in select cases. RADICAL and Infomax delivered very good results but might be too slow for many applications. Both algorithms might be an option if required computation time is not a factor of consideration.

The detailed results for the spectrum of EEG/EP signal patterns can serve as a reference when selecting a particular algorithm for a specific purpose, such as the identification of certain artifact types.

Acknowledgments This work was in part supported by the German Ministry of Science (03IP605) and the German Research Council (DFG Ha 2899/7-1).

References

- Abdi H (2007) Encyclopedia of measurement and statistics. Sage Publications Inc., Thousand Oaks
- De Lucia M, Fritschy J et al (2008) A novel method for automated classification of epileptiform activity in the human electroencephalogram-based on independent component analysis. *Med Biol Eng Comput* 46(3):263–272. doi:10.1007/s11517-007-0289-4
- Delorme A, Makeig S (2004) EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *J Neurosci Methods* 134(1):9–21. doi:10.1016/j.jneumeth.2003.10.009
- DiPietroPaolo D, Müller H et al (2006) Noise reduction in magnetocardiography by singular value decomposition and independent component analysis. *Med Biol Eng Comput* 44(6):489–499. doi:10.1007/s11517-006-0055-z
- Ebe M, Homma I (1994) Leitfaden für die EEG-Praxis. G. Fischer Verlag, Stuttgart
- Fisher AC, El-Deredy W et al (2007) Removal of eye movement artefacts from single channel recordings of retinal evoked potentials using synchronous dynamical embedding and independent component analysis. *Med Biol Eng Comput* 45(1):69–77. doi:10.1007/s11517-006-0123-4
- Giannakopoulos X, Karhunen J et al (1999) An experimental comparison of neural algorithms for independent component analysis and blind separation. *Int J Neural Syst* 9(2):99–114. doi:10.1142/S0129065799000101
- Glass K, Frishkoff G et al (2004) A framework for evaluating ICA methods of artifact removal from multichannel EEG. *Lect Notes Comput Sci* 3195/2004:1033–1040
- Göhler W (1986) Höhere Mathematik. VEB Deutscher Verlag für Grundstoffindustrie, Leipzig
- Hyvarinen A, Karhunen J et al (2001) Independent component analysis. Wiley, New York
- James CJ, Hesse CW (2005) Independent component analysis for biomedical signals. *Physiol Meas* 26(1):R15–R39
- Knuth KH (1998) Maximum entropy and Bayesian methods. Springer, Boise
- Knuth KH, Shah AS et al (2006) Differentially variable component analysis: identifying multiple evoked components using trial-to-trial variability. *J Neurophysiol* 95(5):3257–3276. doi:10.1152/jn.00663.2005
- Krishnaveni V, Jayaraman S et al (2005) Comparison of independent component analysis algorithms for removal of ocular artifacts from electroencephalogram. *Meas Sci Rev* 5(2):67–78
- Krishnaveni V, Jayaraman S et al (2006) Application of mutual information based least dependent component analysis (MILCA) for removal of ocular artifacts from electroencephalogram. *Meas Sci Rev* 1(1):63–74
- Li Y, Powers D et al (2000) Comparison of blind source separation algorithms. World Scientific and Engineering Society Press, pp 18–21
- Milanesi M, Martini N et al (2008) Independent component analysis applied to the removal of motion artifacts from electrocardiographic signals. *Med Biol Eng Comput* 46(3):251–261. doi:10.1007/s11517-007-0293-8
- Nicolaou N, Nasuto SJ (2003) Comparison of temporal and standard independent component analysis (ICA) algorithms for EEG analysis. In: Tenth international conference on neural information processing (ICANN/ICONIP'03), pp 157–160
- Rodenbeck A, Binder R et al (2006) A review of sleep EEG patterns. Part I: A compilation of amended rules for their visual recognition according to Rechtschaffen and Kales. *Somnol Sleep Res Sleep Med* 10(4):159–175
- Stogbauer H, Kraskov A et al (2004) Least-dependent-component analysis based on mutual information. *Phys Rev E Stat Nonlin Soft Matter Phys* 70(6 Pt 2):66–123
- Truccolo W, Knuth KH et al (2003) Estimation of single-trial multicomponent ERPs: differentially variable component analysis (dVCA). *Biol Cybern* 89(6):426–438. doi:10.1007/s00422-003-0433-7
- Vincent E, Rémi G et al (2006) Performance measurement in blind audio source separation. *IEEE Trans Audio Speech Lang Process* 14(4). doi:10.1109/TSA.2005.858005
- Weber H (1992) Einführung in die Wahrscheinlichkeitsrechnung und Statistik für Ingenieure. Teubner, Stuttgart
- Wiklund U, Karlsson M et al (2007) Adaptive spatio-temporal filtering of disturbed ECGs: a multi-channel approach to heart-beat detection in smart clothing. *Med Biol Eng Comput* 45(6):515–523. doi:10.1007/s11517-007-0183-0
- Xu W, Erdogmus D et al (2004) Independent component analysis and blind signal separation. Springer, Heidelberg