Signal Processing 98 (2014) 102–120

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Comparative evaluation of existing and new methods for correcting ocular artifacts in electroencephalographic recordings

Murielle Kirkove*, Clémentine François, Jacques Verly

Department of Electrical Engineering and Computer Science, University of Liège, Grande Traverse 10, B 4000 Liège, Belgium

ARTICLE INFO

Article history:
Received 19 July 2013
Received in revised form
23 October 2013
Accepted 13 November 2013
Available online 22 November 2013

Keywords:
Polysomnography
Electroencephalography (EEG)
Ocular artifacts
Wavelet transform
Adaptive filtering
Independent component analysis
Performance evaluation

ABSTRACT

EEG signals are often contaminated by ocular artifacts (OAs), in particular when they are recorded for a subject that is, in principle, awake, such as in a study of drowsiness. It is generally desirable to detect and/or correct these OAs before interpreting the EEG signals. We have identified 11 existing methods for dealing with OAs. Their study allowed us to create 16 new methods. We performed a comparative performance evaluation of the resulting 27 distinct methods using a common set of data and a common set of metrics. The data was recorded during a driving task of about two hours in a driving simulator. This led to a ranking of all methods, with five emerging clear winners, comprising two existing methods and three new ones.

1. Introduction

Electroencephalographic (EEG) recordings are often contaminated by signals from other sources. The term "artifact" is used both to qualify the contaminating signals, and to refer to a local deformation of the signal of interest, here the EEG signal. One distinguishes between physiological artifacts and technical artifacts. The most frequent physiological artifacts are due to the activity of the eyes, the heart, and the muscles. The most common physiological artifacts are the ocular artifacts (OAs), due to the movements of the eyeballs and eyelids. Technical artifacts are mostly due to electrode placement problems and body movements. Although we generally record the EEG signals from several electrodes, we often consider one of these signals as a generic exemplar. For conciseness, we also use "EEG" to refer to one or more EEG signals in a recording. All artifacts result in a recorded EEG signal that may be quite different, generally locally, from the true underlying EEG signal reflecting the true brain electrical activity. It is thus critical to do something about OAs.

There are three main ways of dealing with OAs. The first ("prevention") tries to minimize the occurrence of OAs by giving proper instructions to patients. However, some OAs occur in a spontaneous, involuntary way, and are thus unavoidable. The second ("epoch rejection") first detects OAs and then rejects, in totality, any epoch affected by one or more OAs; a typical epoch is 2-s long. This approach has the disadvantage of wasting lots of potentially useful data. The third ("signal correction") modifies the OA-corrupted signal and tries to recover, as best as possible, the true, underlying EEG signal (which will, of course, never be known exactly).

The best approach for cleaning an EEG recording from its OAs is probably to detect (i.e. locate) each successive OA, to select a time interval containing it (with appropriate

* Corresponding author. Tel.: +32 4 366 37 70; fax: +32 4 366 29 50.
E-mail address: M.Kirkove@ulg.ac.be (M. Kirkove).

0165-1684/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.sigpro.2013.11.015

Table 1
Existing methods from the literature, and the tools they use.

			Existing methods										
Tool families	**Tools**	**Hypotheses**	1	2	3 .1	3 .2	4 .1	4 .2	5 .1	5 .2	6 (*)	7 (*)	8 (**)
WT	DWT	-	■									■	■
	SWT	-		■		■		■		■	■		
AF	LMS	-			■	■							
	RLS	-					■	■					
	H$^{\infty}$-TV	-							■	■			
ICA	E-INFOMAX	NG										■	
	FastICA										■		
	BGSEP	NS											■

*Uses stICAra.
**Uses weICAra.

margins on both sides), and to correct the EEG recording only in this interval. This avoids perturbing the signal in time regions where the signal is probably not, or less, corrupted; this also results in savings in computation. This approach thus consists in detection/location followed by correction/cleaning. Most published approaches apply the correction directly to the whole signal (possibly per epoch), i.e. without a preliminary detection. Of course, any correction method can be turned into a detection method by subtracting the corrected signal from the raw signal, and appropriately thresholding the resulting signal. The ideal approach for cleaning long recordings from OAs is probably to apply a fast method to locate each OA, and then a sophisticated method to correct the raw signal within the corresponding interval. This paragraph clearly indicates that one needs various types of methods to detect and/or correct OAs. This paper is precisely devoted to providing and evaluating such methods.

When dealing with OAs, it is useful to also record the electrooculographic (EOG) signals, which allows the observer (and the algorithms) to relate the OAs in the EEG and the corresponding features in the EOG.

Our interest in the handling of OAs arose from the study of drowsiness in subjects actively involved in a task, such as driving. Until they fall asleep, these subjects have their eyes mostly open. Therefore, the EEG signals recorded for studying the evolution of drowsiness are affected by OAs due to eye movements and eye blinks. This should be contrasted with the study of sleep, where subjects have their eyes closed. (However, the eyes and the eyelids can move even when the eyes are closed.)

In this paper, we start by identifying the existing methods for detecting and for correcting OAs in EEG signals, and by decomposing them into their constitutive elements. We continue by creating new methods obtained by recombining and/or complementing these elements in different ways. Finally, we quantify the performance of all existing and new methods on a common set of real-life signals and by using a common set of performance measures. Such a comparative study did not exist for the existing methods.

Section 2 describes the data used. Section 3 describes the existing and new methods for dealing with OAs. Section 4 describes the techniques we designed for evaluating and comparing the performances of all these methods. Section 5 presents the results of the comparative performance evaluation. Section 6 concludes and suggests directions for further work.

2. Description of data

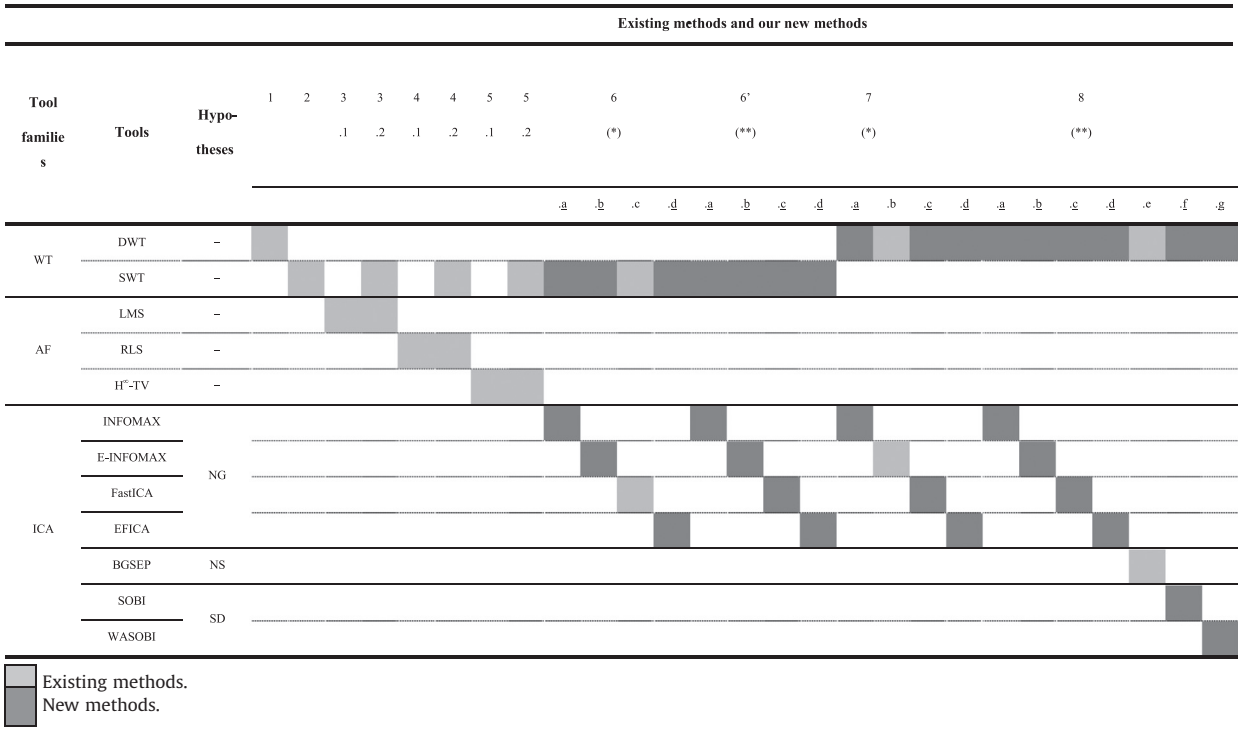
We acquired data at the “Centre d'Etudes des Troubles de l'Eveil et du Sommeil” (CETES) of the University Hospital of Liège in the context of the study of driver drowsiness. We successively placed five subjects in a driving simulator and asked them to drive, for about two hours, at a constant speed of 80 km/h, on a one-way road with no other traffic. A fixed laboratory Embla system recorded the following polysomnographic (PSG) signals, at a sampling rate of 200 Hz: EEG (for electrodes Fz, Cz, Pz, C3, C4, A1, A2), EOG (left and right), and EMG. The PSG signals were oversampled at 500 Hz and partitioned into butting (and thus non-overlapping) epochs. The size (in seconds and/or samples) of each depends upon the method used for dealing with OAs.

3. Methods for dealing with OAs

3.1. Existing methods from the literature

Starting with the reviews of [1] and [2], we identified eight existing “main” methods for dealing with OAs, with

Table 2
Existing methods and our new methods, and the tools they use.



*Uses stICAra.
**Uses weICAra.

three of them coming in two variants. We thus found a total of 11 distinct, existing methods for dealing with OAs. We denote the main methods by the integers 1 to 8, and we add ".1" and ".2" for the two variants. The 11 methods are thus denoted by 1, 2, 3.1, 3.2, etc. These labels appear in the second row in Table 1, which provides one column per method.

Our analysis of the 11 methods allowed us to identify their constitutive elements, which we refer to as "tools". One example of a tool is the discrete wavelet transform (DWT). The eight tools we identified appear in the second column of Table 1, which provides one row per tool. The abbreviations used in this paper are spelled out at the appropriate times, and many are standard. The above tools can be grouped into three families, shown in the first column of Table 1: wavelet transform (WT), adaptive filtering (AF), and independent component analysis (ICA). Some of the tools are applicable only under some hypothesis. The hypotheses appear in the third column: NG stands for "non-Gaussianity" and NS for "non-stationarity".

Table 1 shows, at a glance, the tools used by each method. Table 1 thus gives a synthetic, pictorial view of the relevant methods found in the literature for dealing with OAs.

The main Methods 1 to 7 are applied to one epoch at a time, while the main Method 8 is applied to all (concatenated) epochs in a recording. Method 1 is a detection method, whereas all others are correction methods. To remove artifacts, Methods 6 and 7 use the "standard" ICA approach for removing artifacts, denoted here by "stICAra", whereas Method 8 uses a non-standard approach called the wavelet-enhanced ICA approach for removing artifacts, denoted here by "weICAra", initially introduced in [3] for removing physiological and electrical artifacts from EEG signals[1]. Later, we give a brief description of each existing method.

3.2. Our new methods

Table 2 provides an overall view of both the existing methods and our new methods (for dealing with OAs).

The new methods are all related to the original (main) Methods 6, 7, and 8, which all involve some form of ICA. We added a new main Method 6′, so denoted because it is closely related to Method 6. Each of the main Methods 6, 6′, 7, and 8, have variants. The existing Methods (found in Table 1) are 6.c, 7.b, and 8.e. All others are new; they are identified with an underbar, e.g. as 6.a and 6′.a, as in Table 2, where the existing and new methods are also distinguished by light and dark gray, respectively.

Table 2 shows that Methods 6 and 7 use stICAra, whereas Methods 6′ and 8 use weICAra. There is one existing method under each of Methods 6, 7, and 8, but none under Method 6′. The many methods under Methods 6, 6′, 7, and 8 correspond to different ways of computing the direct and inverse ICA transforms. Each way

[1] "stICAra" stands for "standard ICA (approach) for removing artifacts", and "weICAra" for "wavelet-enhanced ICA (approach) for removing artifacts".

corresponds to a specific tool in Table 2, and some tools are applicable only under some hypotheses, i.e. NG, NS, and SD, where NG stands for "non-Gaussianity", NS for "non-stationarity", and SD for "spectral diversity". Below, we motivate the introduction of the new combination of tools leading to the 16 new methods, for a total of 27 methods.

3.2.1. Representation of signals

We generally process one epoch at a time. However, in some cases, we process the whole recording, or we process segments of an epoch. Some methods use a single PSG channel, and others, several, or all, of the available channels. In any case, the data processed by an algorithm is, for each channel, a succession of samples, which is treated either as an ordered set of samples or, equivalently, as a vector, both of length N.

In some theoretical developments, the signals are treated as random processes (RPs), and the corresponding vectors as random vectors (RVs). Of course, in such cases, the processing is applied to one particular realization of each of these random quantities.

3.2.2. Wavelet transforms (WT) tools

The wavelet transform (WT) [4] is one of the leading techniques for processing non-stationary signals, where "non-stationary" should be understood here loosely as meaning "with time-varying frequency content" (and not according to the meaning that this qualifier has in statistical signal processing). The WT is thus well suited for EEG signals, since these are non-stationary. The major feature of the WT is its capacity of decomposing a signal into components that are well localized in scale (which is essentially the inverse of frequency) and time.

The continuous WT (CWT) uses a family of wavelets by "continuously" scaling and translating a localized function called the mother wavelet. The discrete WT (DWT) is obtained by discretizing the scale and translation variables "dyadically", i.e. by using powers of two. This allows one to implement the CWT on a computer. The DWT is not translation invariant. However, such invariance is required in some applications like change detection and denoising [5]. The transform obtained by removing the down-samplers and up-samplers from the CWT is translation invariant, and naturally called the stationary WT (SWT) [5]. Just as the DWT, the SWT uses a dyadic grid. The DWT can be interpreted as (1) passing its input through a low-pass filter (LPF) and a high-pass filter (HPF), resulting in an "approximation" part (a) and a "detail" part (d), respectively, and then (2) recursively applying this double operation to the output of the LPF, i.e. to the a part. With L successive applications of the procedure, one ends up with one set of detail WT coefficients (WTCfs) at each of the L levels (of scale, from small to large), and one set of approximation WTCfs at level L. The detail WTCfs correspond to higher frequencies, and thus to smaller scales, and conversely for the approximation WTCfs.

3.2.2.1. Method 1 (based on DWT). Method 1 is the method of [6], which is designed to detect (zones of) OAs. (As alluded to earlier, this is the only detection method, all others being correction methods.) Fig. 1 shows the block diagram (BD) of the method, and, thus, its main processing steps. The input is the raw EEG signal, and the output is the OA-zone signal, i.e. a binary signal with 1's in what the method believes to be zones of OAs. The input and output signals should be understood as being ordered sets of N samples, here covering exactly 2 s.

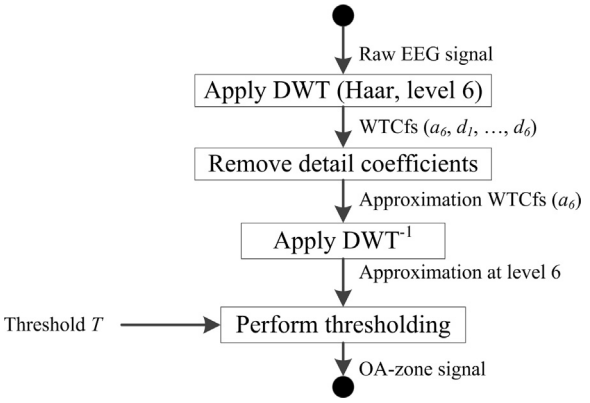


Fig. 1. Block diagram of Method 1.

Method 1 applies the DWT to the input (up to some level L), using the Haar mother wavelet, because of its resemblance to an eyeblink OA. Since the (D)WT can be viewed as performing, at each level l, i.e. at each scale s, a correlation of the input with the chosen mother wavelet scaled by s, one would expect to find strong WTCfs at some scales. Since the OAs are typically located in the lower part (defined as [0–13] Hz or as [0–16] Hz) of the EEG spectrum ([0–20] Hz), one would expect to find the strongest WTCfs mostly in the approximation WTCfs (and perhaps in some of the detail WTCfs at the larger scales). Method 1 removes all detail WTCfs, keeping only the approximation WTCfs at the last level, i.e., L, with L set to 6. The method then reconstructs the time-domain signal corresponding to the approximation WTCfs. The result is a staircase waveform. All parts of this waveform that are above some experimentally-determined threshold are declared to correspond to a zone of OAs.

3.2.2.2. Method 2 (based on SWT). Method 2 is the method of [7], which is designed to reduce the effects of OAs. Fig. 2 shows the BD of the method. The input is the raw EEG signal, and the output is a cleaned EEG signal, i.e. with the effects of OAs reduced. Here too, the input and output should be understood as being ordered sets of N samples. However, since the SWT requires that N be a power of two, we use a total of 1024 samples corresponding to duration of 2.048 s, which is close to the desired, nominal duration of 2 s.

Method 2 is a typical denoising method. Most denoising methods based on the WT use the SWT (because of its time-invariance), followed by soft or soft-like thresholding of some of the WTCfs[2]. Simplifying, the general idea is to

[2] Soft-like-thresholding functions have continuous derivatives [9], whereas soft-thresholding functions are continuous with discontinuous derivative [8].

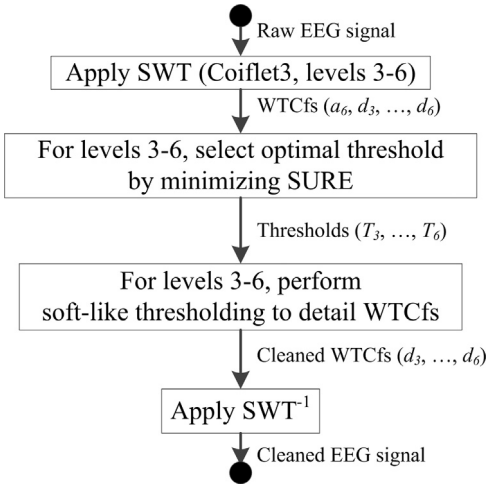


Fig. 2. Block diagram of Method 2.

eliminate the WTCfs with magnitude below some threshold T, and to reduce the magnitude of the others by T. The threshold can be different at each (scale) level.

Noise is commonly found at high frequencies, in which case the threshold is applied to all detail WTCfs. If one views OAs as low-frequency ([0–13/16] Hz) noise (as in [7]), then the thresholding should be applied to the lower frequencies ([0–13/16] Hz), but perhaps not all the way to zero frequency (DC). This explains why, in [7], the SWT is computed (using the Coiflet 3 mother wavelet) for $L=6$ levels, with thresholding being applied to the detail WTCfs for levels 3 to 6 (corresponding to larger scales and thus smaller frequencies). Carefully note that the thresholding is not applied to the approximation WTCfs at the last level.

Ref. [7] uses soft-like thresholding, and the optimum threshold is computed at each of the levels 3 to 6 by minimizing the Stein Unbiased Estimate of Risk (SURE) [8,10]. It is crucial to note that [10] requires that the "quantities", say x_i, that are thresholded, here the detail WTCfs, be (realizations of) random variables (RVs) X_i that are independent and Gaussian with equal unit variance and possibly different means. However, one can show that the above method for computing the optimum threshold is also justified by only assuming that the X_i's are independent and Gaussian with possibly different means and possibly different variances. Because of the linearity and the time-invariance of the SWT, one can show that, if the input signal is corrupted by Gaussian noise, the WTCfs of the noise part of the signal also constitutes a Gaussian noise. However, it is evident that an OA in an EEG signal is not a Gaussian noise. A major question with the method of [7] is whether the WTCfs constitute a Gaussian noise. Although the WTCfs are probably more noise-like than the corresponding OA, the issue of whether the WTCfs constitute a Gaussian noise remains open… The likely lack of satisfaction of this requirement may explain why we did not observe good performance with Method 2 (see Section 5).

3.2.3. *Adaptive filtering (AF) tools*

Fig. 3(a) shows the "most" generic form of an adaptive filter (AF): $x[n]$ is the input, $y[n]$ is the output, and $d[n]$ is

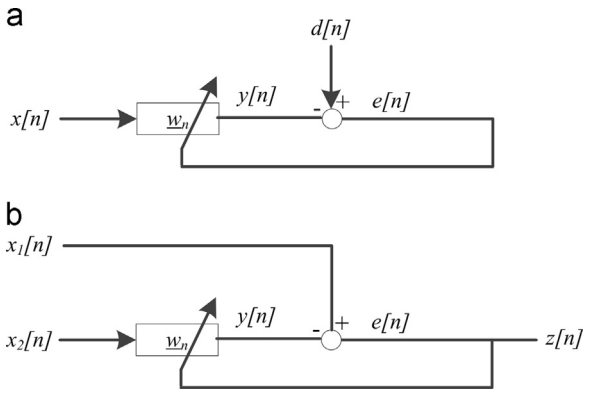


Fig. 3. Block diagram of (a) generic adaptive filter (AF) and (b) specific AF.

the "desired" signal. Notations such as $x[n]$ denote discrete-time (DT) random processes, the realizations of which are conventional DT sequences [11]. The difference between $d[n]$ and $y[n]$ defines the error signal $e[n]$. The value of this signal at any particular time, say n_0, is used to adjust the weight vector $\underline{w}_n$ of the AF at time n_0, i.e. $\underline{w}_{n0}$; $\underline{w}_n$ is the vector representation of the impulse response (IR) of a time-variant FIR filter, with length denoted by L[3].

Fig. 3(b) shows a slight adaptation of Fig. 3(a) tailored to the problem at hand. The input to the AF, here denoted by $x_2[n]$, is the raw EOG signal, and the desired signal, here denoted by $x_1[n]$, is the raw EEG signal. While the output of the generic AF is still $y[n]$, the output that is of interest is the error signal $e[n]=x_1[n]-y[n]$. The hope is that $y[n]$ will correspond to the OA artifact that affects $x_1[n]$, so that the output $e[n]$ will be a new EEG signal with the effect of OAs reduced.

For clarity, the system of Fig. 3(a) is here called "generic AF", and that of Fig. 3(b) "specific AF". Note that the (primary and secondary) inputs of the specific AF are x_1 and x_2, and its output z.

In Fig. 3(a), the value of the intermediate output (signal) $y[n]$ at $n=n_0$ is obtained as the scalar product between the filter vector $\underline{w}_{n0}$ of length L, and a vector $\underline{x}_{2n0}$ of length L containing the present sample $x_2[n_0]$ and the L-1 preceding values of $x_2[n]$, which we write as

$$y[n]=\underline{w}_n^{\dagger}\times\underline{x}_{2n}.$$

The error (signal) $e[n]$ is thus

$$e[n]=x_1[n]-y[n]=x_1[n]-\underline{w}_n^{\dagger}\times\underline{x}_{2n}.$$

The weight $\underline{w}_n$ is adapted linearly at each time step according to

$$\underline{w}_{n+1}=\underline{w}_n+e[n]\underline{g}_n,$$

where $\underline{g}_n$ is a vector of length L, the form of which depends upon the particular type of generic AF used.

We considered three types of generic AF filters, respectively based on (1) the least mean square (LMS) algorithm, which minimizes the mean squared error, (2) the recursive

[3] Not to be confused with the number of levels in a WT.

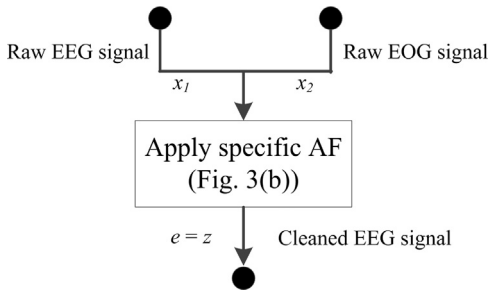


Fig. 4. Block diagram of Methods 3.1, 4.1, and 5.1.

least squares (RLS) algorithm, which minimizes a cost function that is a linear combination of the squared error, and (3) the H^∞ time-varying (H^∞–TV) algorithm, which minimizes the infinite norm of a linear combination of the squared error [12]. We implemented these three AF filters as in [13]. In particular, we set L to 3.

For LMS, $\underline{g}_n$ is given by

$$\underline{g}_n = \mu \underline{x}_{2n},$$

where μ is the step size, chosen to balance the stability and convergence of the algorithm [14]. As in [13], we take $\mu = 10^{-6}$.

For RLS, $\underline{g}_n$ is given by

$$\underline{g}_n = \frac{P_{n-1}\underline{x}_{2n}}{\lambda + \underline{x}_{2n}^T P_{n-1}\underline{x}_{2n}},$$

where the $L \times L$ matrix P_n is recursively defined by

$$P_{L-1} = \frac{I}{\sigma}, P_n = \frac{P_{n-1}}{\lambda} - \frac{\underline{g}_n \underline{x}_{2n}^T P_{n-1}}{\lambda},$$

where λ is the forgetting factor [13]. As in [13], we take $\sigma = 10^{-2}$ and $\lambda = 9999\ 10^{-4}$.

For H^∞–TV, $\underline{g}_n$ is given by

$$\tilde{P}_n^{-1} = P_n^{-1} - \frac{\underline{x}_{2n}\underline{x}_{2n}^T}{\varepsilon^2}, g_n = \frac{\tilde{P}_n \underline{x}_{2n}}{1 + \underline{x}_{2n}^T \tilde{P}_n \underline{x}_{2n}},$$

where the $L \times L$ matrix P_n is recursively defined by

$$P_{L-1} = \eta I,\ P_{n+1} = \left(P_n^{-1} + \left(1 - \frac{1}{\varepsilon^2}\right)\underline{x}_{2n}\underline{x}_{2n}^T\right)^{-1} + \rho I,$$

where η is the initial distance to the optimal solution, ρ the variation speed of the filter coefficients, and ε a positive constant [13]. As in [13], we take $\eta = 510^{-3}$, $\rho = 10^{-5}$, and $\varepsilon = 1.5$.

3.2.3.1. Methods 3.1, 4.1, and 5.1 (based on AF only). Table 1 shows that the existing Methods 3.1, 4.1, and 5.1 use, as their sole tool, an AF, based on the LMS, RLS, and H^∞–TV algorithms, respectively. Specifically, they do not use any WT tool. Fig. 4 shows the common BD of the three methods.

3.2.3.2. Methods 3.2, 4.2, and 5.2 (based on AF and SWT). Method 3.2 is the method of [15]. It applies the SWT to the input (using the Symlet3 mother wavelet, and 8 levels), LMS to the SWT coefficients (WTCfs) for each decomposition level, and the inverse SWT to the AF (LMS) outputs at all decomposition levels. Methods 4.2 and 5.2

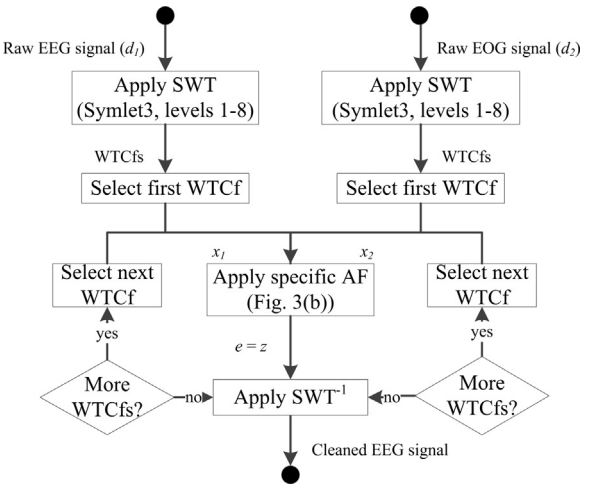


Fig. 5. Block diagram of Methods 3.2, 4.2, and 5.2.

are similar to Method 3.2, the only difference being that, instead of LMS, they use RLS and H^∞–TV, respectively. Fig. 5 describes each of the three methods.

Because of the use of the WT in Methods 3.2, 4.2, and 5.2, we use $N = 1024$ samples. We used this same number for the .1 variants.

3.2.4. Independent component analysis (ICA) tools

In the context of ICA, one assumes that one has access to, i.e. can observe, several signals that are each a linear combination of (1) some other signals that are the ones of interest, but are unknown, and (2) a noise signal that is also unknown (and of no interest). Each signal of interest is called a source (signal) or a component (signal). In a nutshell, if the sources are independent, and the noise uncorrelated with the sources, ICA allows one to recover the unknown signals (sources) from the observed signals (observations).

The fundamental ICA problem is formulated as follows. One assumes that one has access to K observations – corresponding here to the available PSG channels – and that these observations are, for simplicity, generated by the same number (i.e. K) of sources, and are each affected by noise. One models the underlying signals as random, and one deals here with random vectors (RVs), e.g. corresponding to an epoch, all of the same length N. The kth observation, source, and noise RVs are denoted by $\underline{x}_k$, $\underline{s}_k$, and $\underline{n}_k$, respectively. For $\underline{x}_k$ and $\underline{n}_k$, one can say that "k" corresponds to the kth observation channel, but this does not apply to $\underline{s}_k$! One assumes that the above RVs are linearly related by

$$\begin{pmatrix} \underline{x}_1 \\ \vdots \\ \underline{x}_K \end{pmatrix} = \begin{pmatrix} A_{11} \cdots A_{1K} \\ \vdots \\ A_{K1} \cdots A_{KK} \end{pmatrix} \begin{pmatrix} \underline{s}_1 \\ \vdots \\ \underline{s}_K \end{pmatrix} + \begin{pmatrix} \underline{n}_1 \\ \vdots \\ \underline{n}_K \end{pmatrix}, \quad (1)$$

or

$$(\underline{x}_1 \cdots \underline{x}_K)^T = \left(\sum_{l=1}^{K} A_{1l}\underline{s}_l + \underline{n}_1 \cdots \sum_{l=1}^{K} A_{Kl}\underline{s}_l + \underline{n}_K \right)^T,$$

where the block matrix of the A_{kl}, denoted by A, is the mixing matrix, and each $\underline{n}_k$ is assumed to be a white noise

(WN) RV uncorrelated with each of the $\underline{s}_k$'s. One can compactly write (1) as

$$\underline{x} = A\underline{s} + \underline{n}, \tag{2}$$

where all quantities should always be viewed as consisting of blocks of vectors or matrices, as appropriate.

ICA provides an estimate of $\underline{s}$ according to some optimization criterion if A has full rank. The ICA solution is ambiguous in the sense that $\underline{s}$ is only identifiable up to a permutation and a scaling of the sources (meaning that the order and the scales of the original sources cannot be retrieved). Matrices of the form $W = \Delta P A^{-1}$, where Δ is a diagonal matrix with nonzero diagonal elements and P is a permutation matrix, are called de-mixing matrices. By applying such a matrix W to the realization of interest of $\underline{x}$, one gets an estimate of the corresponding $\underline{s}$. Once the source vectors (or simply sources) are known, one needs to determine which of the sources are of interest. E.g., one source could be the artifactual part of the EEG signal due to the eye movements.

There are several ways to compute the above matrices A or W. We refer to these various implementations of ICA as the ICA tools. These appear in Table 2. The tools used by the existing methods are E-INFOMAX, FastICA, and BGSEP; and those introduced by the new methods are INFOMAX, EFICA, SOBI, and WASOBI.

As suggested by Table 2, each ICA tool is applicable only under some specific hypothesis. These tools are presented in [16], which is a survey of fast and accurate methods for implementing ICA. This reference distinguishes between four categories of tools depending upon the statistical model assumed for the sources: non-Gaussianity (NG), non-stationarity (NS), spectral diversity (SD), and hybrid, where "hybrid" means a combination of several models. For conciseness, we ignore here the hybrid models. Specifically, we only consider the NG, NS, and SD models (and hypotheses).

The NG model assumes that the sources are independent RPs. Some algorithms (or techniques) maximize the Kullback–Leibler divergence between (1) the joint probability density function (JPDF) of the separated sources and (2) the product of the marginals [17]. One can show that the idea behind these algorithms is equivalent to maximizing the non-gaussianity of the probability density functions (PDFs) of the sources, i.e. of the RVs $\underline{s}_k$ [18]. Some algorithms estimate the PDFs in a non-parametric way. According to [16], these algorithms are computationally complex and cannot be used when the number of sources is large, say for $K > 10$. Other algorithms model the PDFs parametrically and estimate the parameters of these PDFs. FastICA and INFOMAX are the most popular algorithms for the NG hypothesis. FastICA is a fast and accurate algorithm for minimizing the negentropy, which can be approximated by a non-linear and non-quadratic measure, such as the kurtosis[4] [18]. INFOMAX [19] is an unsupervised learning algorithm for maximizing the output entropy of a neural network with non-linear outputs. INFOMAX is effective in separating sources that have super-gaussian PDFs, but it cannot separate sources that have negative kurtosis. E-INFOMAX, i.e. extended-INFOMAX, can deal with both sub-gaussian and super-gaussian sources [20]. EFICA is sophisticated, nearly achieves the Cramér–Rao bound, and is less computationally demanding [21]. Simulations show that EFICA outperforms some non-parametric algorithms [21], including the classical JADE [22], in terms of both accuracy and computational complexity.

In the NS model, one divides the time interval of interest, typically corresponding to one epoch, into M time segments of equal durations. One can then define the $\underline{x}_k$'s, $\underline{s}_k$'s, and $\underline{n}_k$'s for each segment. One assumes that, on each segment, the $\underline{s}_k$'s are zero-mean independent RVs. The variance of each component of $\underline{s}_k$ is allowed to vary from segment to segment, and with k (the "channel" index). Under the NS hypothesis, the de-mixing matrix W can be found by an approximate joint diagonalization of the sample covariance matrices of the realizations corresponding to the observation RVs $\underline{x}_k$, $k=1,\ldots,K$ [16]. This can be done in several ways, each optimizing a different criterion. [23] suggests one criterion leading to the ML estimation of the sources, and [24] an algorithm for minimizing this criterion. [25] gives a different method of approximate joint diagonalization, called BGWEDGE, leading to an estimate that is asymptotically equivalent to that of [24], but at a lesser computational cost. When used with ICA, the method is called BGSEP, which stands for "block Gaussian separation".

In the SD model, one proceeds as follows. The vectors $\underline{x}_k$, $\underline{s}_k$, and $\underline{n}_k$ in (1) are assumed to be functions of the sample index n, and are thus denoted $\underline{x}_k[n]$, $\underline{s}_k[n]$, and $\underline{n}_k[n]$. Eq. (2) then becomes

$$\underline{x}[n] = A\underline{s}[n] + \underline{n}[n]$$

where all quantities should be viewed as consisting of blocks. The (block) cross-correlation matrix $R_{\underline{x}}(n_1, n_2)$ of $\underline{x}[n]$ is given by

$$\begin{aligned} R_{\underline{x}}(n_1,n_2) &= E\{\underline{x}[n_1]\underline{x}^{\dagger}[n_2]\} = E\{(A\underline{s}[n_1]+\underline{n}[n_1])(\underline{s}^{\dagger}[n_2]A^{\dagger}+\underline{n}^{\dagger}[n_2])\} \\ &= AR_{\underline{s}}(n_1,n_2)A^{\dagger} + AR_{\underline{sn}}(n_1,n_2) + R_{\underline{ns}}(n_1,n_2)A^{\dagger} + R_{\underline{n}}(n_1,n_2). \end{aligned}$$

As indicated above, in the ICA problem, one assumes that each $\underline{s}_k$ is uncorrelated with each $\underline{n}_l$ and that each $\underline{n}_l$ is a WN. In the SD model, one also assumes that, for each sample index n, $\underline{n}_l[n]$ is zero mean and has a variance independent of n and l, denoted by σ^2. The above equation then reduces to

$$R_{\underline{x}}(n_1,n_2) = AR_{\underline{s}}(n_1,n_2)A^{\dagger} + \sigma^2 I\delta_{n_1 n_2},$$

which we can also write as

$$R_{\underline{x}}(n+d,n) = AR_{\underline{s}}(n+d,n)A^{\dagger} + \sigma^2 I\delta_{d0},$$

where n is the (present) sample index and d the sample separation $d = n_1 - n_2$. One then assumes that $\underline{s}_k[n]$ are such that the correlation matrices of each $\underline{s}_k[n]$ for $k=1,\ldots,K$, and thus of $\underline{s}[n]$ only depend upon the difference of the

[4] The kurtosis is a measure of the peakiness of the PDF of a random variable (RV). The kurtosis of any gaussian RV is zero. Thus, the kurtosis also measures the non-gaussianity of RVs. RVs with negative kurtoses are called sub-gaussian, and those with positive kurtoses are called super-gaussian [18].

indices, i.e. d. It follows that

$$R_{\underline{x}}(n+d,n) = AR_{\underline{s}}(d)A^{\dagger} + \sigma^2 I\delta_{d0},$$

so that $R_{\underline{x}}(n+d,n)$ also depends only on d. We thus write

$$R_{\underline{x}}(d) = AR_{\underline{s}}(d)A^{\dagger} + \sigma^2 I\delta_{d0}. \quad (3)$$

Let $\underline{z}$ denote the composite vector $A\underline{s}$, and Z denote a matrix such that the $Z\underline{z}_k$'s are temporally white. We call such matrix Z a whitening matrix and the process consisting in applying such a matrix to a vector whitening. We get

$$I = E\{Z\underline{z}[d](Z\underline{z})^{\dagger}[d]\} = ZE\{A\underline{s}[d](A\underline{s})^{\dagger}[d]\}Z^{\dagger} = ZAR_{\underline{s}}(0)A^{\dagger}Z^{\dagger}.$$

Since any scalar multiplication of one source $\underline{s}_k$ and the division of the corresponding column in A by the same scalar leaves $\underline{x}$ unaffected (see Eq. (1)), one can assume (without any loss of generality) that the sources have unit variances, i.e.$R_{\underline{s}}(0) = I$, in which case the last Equation reduces to $ZA(ZA)^{\dagger} = I$. This relation means that there exists a unitary matrix U so that

$$ZA = U. \quad (4)$$

Using (3) and (4), we get the key relation [26]

$$R_{Z\underline{x}}(d) = ZR_{\underline{x}}(d)Z^{\dagger} = UR_{\underline{s}}(d)U^{\dagger}.$$

Since U is unitary and $R_{\underline{s}}(d)$ is diagonal, this relation means that, for any whitening matrix, the correlation matrices $R_{Z\underline{x}}(d)$ can be diagonalized by a unitary matrix. Let us recall that in such a diagonalization problem, the matrix U can be found only if the eigenvalues are distinct. Therefore, after whitening, one estimates A by (1) finding an index d such that the eigenvalues of $R_{Z\underline{x}}(d)$ are distinct, (2) determining the eigenvectors, (3) building the unitary matrix U with these eigenvectors as columns, and (4) solving the linear Eq. (4). It is impossible to determine a priori the index d such that the eigenvalues of $R_{Z\underline{x}}(d)$ are distinct. According to [26], the problem becomes easier when one considers the joint diagonalization of p correlation matrices$\{R_{Zx}^{d_1}, \ldots, R_{Zx}^{d_p}\}$ corresponding to p sample separations$\{d_1, \ldots, d_p\}$. The existence and the uniqueness of the solution is ensured if and only if the following condition is satisfied [25],

$$\forall k,l \exists d : (R_{\underline{s}}^{d})_{kk} \neq (R_{\underline{s}}^{d})_{ll}.$$

This relation means that the sources $\underline{s}_k$ have distinct power spectral densities (which justifies the name "spatial diversity" (SD) of the present model of sources). The first algorithm for performing such diagonalization is based on Jacobi rotations and is known as SOBI (second order blind identification) [26]. It is popular in biomedical applications. Several algorithms were independently proposed in [25,27], and [28] for solving ICA when the SD model is restricted to independent sources following the Gaussian auto-regressive (AR) model. [25] shows that these algorithms can be generalized to the framework of approximate joint diagonalization (AJD) incorporating weight matrices; [25] also derives weight matrices for the algorithm presented in [27]. The resulting algorithm is called WASOBI (for weight adjusted SOBI). According to [16], WASOBI is the most efficient algorithm for performing ICA (under the SD hypothesis) in high-dimensional space ($K > 100$).

To sum up, the efficient, non-hybrid ICA tools for biomedical applications are EFICA (under the NG hypothesis), BGSEP (NS hypothesis), and WASOBI (SD hypothesis) [16]. For memory, we do not consider hybrid models here. We used existing Matlab codes for (1) FASTICA[5], (2) INFOMAX and E-INFOMAX[6], (3) EFICA[7], (4) SOBI[8], and (5) WASOBI, BGSEP, and BGWEDGE[9].

3.2.4.1. Methods 6 and 6′ (based on SWT and ICA). Among the Methods 6.x and 6′.x[10], the only existing method is Method 6.c, which is the method of [29] (Table 2). All others are new. Fig. 6(a) and (b) show the BDs of Methods 6 and 6′, respectively. In each case, the input is the set of raw PSG signals, and the output is a set of cleaned EEG signals, where "set" refers to the channels that are available and are of interest. Fig. 6 shows that Methods 6 and 6′ call upon a single (sub)-system, i.e. stICAra and weICAra, respectively. The reason for using distinct BDs for these methods and these subsystems is that the methods deal with PSG signals, while the subsystems deal with generic (i.e. "arbitrary") signals.

3.2.4.1.1. Subsystems stICAra and weICAra. For memory, stICAra stands for "standard ICA (approach) for removing artifacts", and weICAra for "wavelet-enhanced ICA (approach) for removing artifacts". These are our own acronyms and terms. Figs. 7 and 8 show the BDs of these subsystems. Once again, in each case, the input is a set of raw signals, and the output is a set of cleaned signals. The BDs show that the general principle of operation of both subsystems is essentially the same: (1) Apply ICA to the input, (2) identify artifactual sources, (3) deal with these "in some way" (removal or replacement), (4) apply inverse ICA (i.e. apply the de-mixing matrix to the output of the previous step).

The BDs of these subsystems indicate that the artifactual sources are identified in one common way in stICAra and weICAra when these are used in Methods 6 and 6′, respectively, and in two other distinct ways when used in Methods 7 and 8, respectively. To distinguish these three subsystems for identifying artifactual sources, we denote them by "Identify artifactual sources (i)", with i=1,2,3. Since we are presently considering Methods 6 and 6′, we only describe the first of these incarnations below.

Subsystem stICAra corresponds to the standard way of dealing with (i.e. "removing") artifacts in EEG signals, whereas weICAra is not. While weICAra is a method for dealing with artifacts, it has apparently never been used as such in the context of PSG signals. Indeed, in [3], there is no equivalent to our subsystem "Identify artifactual sources (i)" since the artifacts are identified manually. In

[5] http://research.ics.aalto.fi/ica/fastica/.
[6] http://sccn.ucsd.edu/eeglab/allfunctions/runica.html.
[7] http://itakura.kes.tul.cz/zbynek/dwnld/efica.m.
[8] http://sccn.ucsd.edu/eeglab/allfunctions/sobi.m.
[9] http://si.utia.cas.cz/Tichavsky.html.
[10] The notation "Method n.x" is a compact way of denoting all methods of main Method n.

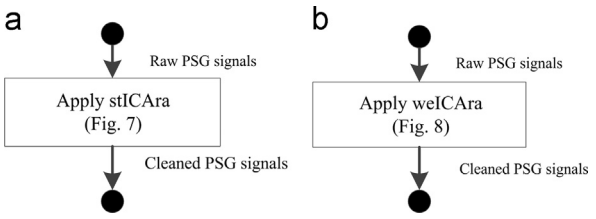


Fig. 6. Block diagrams of Methods 6 and 6′.

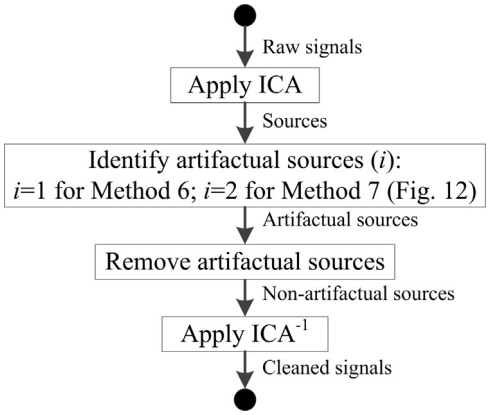


Fig. 7. Block diagram of subsystem "standard ICA (approach) for removing artifacts (stICAra)".

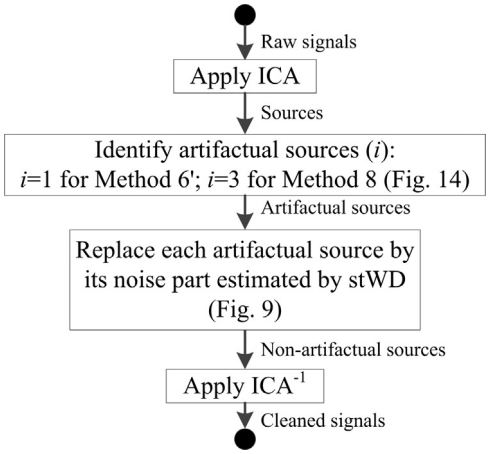


Fig. 8. Block diagram of subsystem "wavelet-enhanced ICA (approach) for removing artifacts (weICAra)".

[30], which is essentially Method 8.e, the subsystem weICAra is not as clearly identified as here, and the method was not initially developed for analyzing PSG signals but for analyzing neo-natal EEG signals.

Below, we successively describe the subsystems "Identify artifactual sources (1)" and "Replace each artifactual source by its noise part estimated by stWD". Of course, the subsystem "Remove artifactual sources" is self-explanatory.

3.2.4.1.2. Subsystem "Identify artifactual sources (1)". As alluded to above, there are three different incarnations of the operation "Identify artifactual sources (i)". Here, we describe the first of these, identified with "(1)", which is used in Methods 6 and 6′ (more specifically in the subsystems stICAra and weICAra). The present approach comes from [29]. In all incarnations, the input is a set of sources (produced by some ICA tool), and the output is the set of artifactual sources.

In the present case ("(1)"), the artifactual sources are identified as follows. The first step consists in computing ten statistical measures. These measures are computed for each of the sources $\underline{s}_k$. Some of these measures involve the left or right EOG signals, also treated as RVs (just as $\underline{s}_k$), and denoted by $\underline{e}_L$ and $\underline{e}_R$, respectively. Two measures are the mutual informations between $\underline{s}_k$ and $\underline{e}_L$ (MI_L), and between $\underline{s}_k$ and $\underline{e}_R$ (MI_R). Two measures are the projection strengths of $\underline{s}_k$ on $\underline{e}_L$ (PS_L), and of $\underline{s}_k$ on $\underline{e}_R$ (MI_R). Two measures are the correlations between $\underline{s}_k$ and $\underline{e}_L$ ($Corr_L$), and between $\underline{s}_k$ and $\underline{e}_R$ ($Corr_R$). One measure is the kurtosis of the source (Kt). Three measures are also kurtoses, but they are computed on the sets of WT components (WTCps)[11] corresponding to the approximation parts at levels 3 and 4, and to the detail part at level 5 (Kta_3, Kta_4, and Ktd_5). The WTCfs used for generating the WTCps are computed using the SWT (because of its stationarity). [29] does not give any clear indication about the mother wavelet that is used. Here, we use the Symmlet 3 mother wavelet (because of its resemblance to an OA). The ten statistical measures m_{kl} are illustrated in Table 3 for an example case of four sources $\underline{s}_k$. This table is perfectly suited for explaining and illustrating the second step of the identification. The algorithm effectively flags the largest value in each of the columns corresponding to the first six measures, as well as the two largest values in the last three columns. Any source with at least four flags (thus appearing in the corresponding row of Table 3) is deemed to be artifactual.

3.2.4.1.3. Subsystem "Replace each artifactual source by its noise part estimated by stWD" (for weICAra only). Rather than completely removing the artifactual sources, this subsystem replaces each (full) artifactual source by its noise part, which is estimated by using a standard denoising approach, called here "standard wavelet denoising", and denoted by "stWD".

The rational for this use of denoising is as follows. One can view each artifactual source (signal) as consisting of the sum of two (component) signals corresponding to OAs and to the neural activity, respectively [3]. Since the OAs correspond to lower frequencies, and the neural activity to higher frequencies, the neural activity signal can be treated as being noise, and, thus, identified using stWD. [32] appears to be the first reference using WD to separate strong artifacts from neural activity in sources obtained by applying ICA on EEG signals.

3.2.4.1.4. Subsystem "Standard wavelet denoising (stWD)". This subsystem corresponds to a standard wavelet-based technique for denoising signals. Fig. 9 shows the BD of this subsystem. The input is a raw signal and the output a cleaned version of this input. When used in the subsystem

[11] The difference between WT coefficients and WT components is perhaps best understood for the DWT. In the DWT series $\sum c_k\psi_k$, ψ_k is a basis function, c_k a coefficient, and $c_k\psi_k$ the corresponding component. The interpretation is more complex for the SWT, which is not an orthogonal transform.

Table 3
Illustration of method for identifying artifactual sources.

		Measures										Artifactual sources
		m_1 (MI_L)	m_2 (MI_R)	m_3 (PS_L)	m_4 (PS_R)	m_5 ($Corr_L$)	m_6 ($Corr_R$)	m_7 (Kt)	m_8 (Kta_3)	m_9 (Kta_4)	m_{10} (Ktd_5)	
Sources	s_1	m_{11}	m_{12}	m_{13}	m_{14}	m_{15}	m_{16}	m_{17}	m_{18}	m_{19}	m_{110}	No
	s_2	m_{21}	m_{22}	m_{23}	m_{24}	m_{25}	m_{26}	m_{27}	m_{28}	m_{29}	m_{210}	Yes
	s_3	m_{31}	m_{32}	m_{33}	m_{34}	m_{35}	m_{36}	m_{37}	m_{38}	m_{39}	m_{310}	No
	s_4	m_{41}	m_{42}	m_{43}	m_{44}	m_{45}	m_{46}	m_{47}	m_{48}	m_{49}	m_{410}	Yes

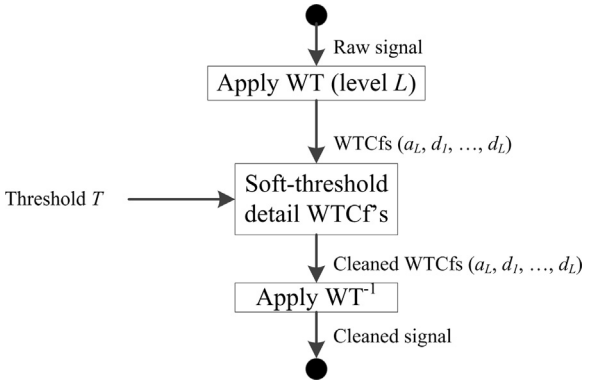


Fig. 9. Block diagram of subsystem "Standard wavelet denoising (stWD)".

"Replace …" and, thus, in weICAra, the input is a source (signal). The subsystem applies a WT (computed to level L), soft-thresholds the resulting detail WTCfs, and applies the inverse WT to the so modified WTCfs.

In our use of stWD within Method 6′ (based on weICAra), we use the same tools and parameters as in [3]. We use the SWT (which better deals with non-stationary signals than the DWT), the Haar mother wavelet (since it is simple, provides fast convergence, and its combination with SWT gives satisfactory denoising performance with minimum distortion and data redundancy [3]), and $L=5$ decomposition levels (since this gives satisfactory results for the weICAra experiments of [3]). stWD is also used by Method 8 within the subsystem weICAra, but with different tools and parameters.

The only existing method under Methods 6 and 6′ is Method 6.c (the method of [29]). The epochs used in this last method have durations of 4 s. Since Method 6′ uses stWD, and since the SWT used here by stWD requires a number of samples that is a power of two, we apply each Method 6′.x to epochs of 2048 samples corresponding to durations of 4.096 s, which is close to the duration of 4 s. Since Method 6 is a sort of variant of Method 6′, we also use 2048 samples for all Methods 6.x.

Note that, in weICAra and, thus, in Method 6′, the WT is used twice, once to identify artifactual sources, and once to replace each by its noise part.

3.2.4.1.5. Remarks on Methods 6 and 6′. As indicated above, Method 6′ is novel since weICAra has never been used as such to deal with PSG signals. In the existing Method 6.c (the method of [29]) – which uses stICAra – the direct and inverse ICA are implemented using the ICA tool called FastICA (which is furthermore valid only under the NG hypothesis).

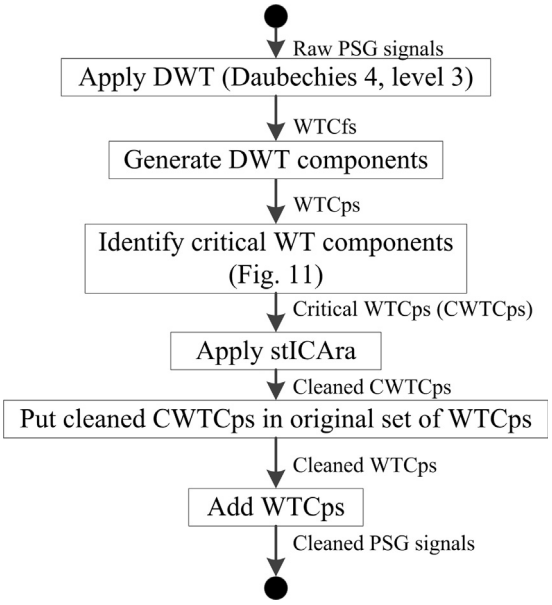


Fig. 10. Block diagram of Method 7.

Our new Methods 6.a, 6.b, and 6.d, innovate by using INFOMAX, E-INFOMAX, and EFICA, respectively, all themselves applicable under the NG hypothesis. Under the NG hypothesis, FastICA, INFOMAX, and E-INFOMAX are the most popular ICA tools, and EFICA the most powerful [16].

3.2.4.2. Methods 7 (based on DWT and ICA). Among the Methods 7.x, the only existing method is Method 7.b, which is the method of [31] (Table 2), further discussed below. All others are new. Fig. 10 shows the BD of main Method 7. The inputs and outputs are identical to those of Methods 6 and 6′, i.e. the sets of raw or cleaned PSG signals.

At a high-level, the method works as follows. For each input raw PSG signal, there are a corresponding input and a corresponding output for the fourth processing step "Apply stICAra". With the exception of this fourth step, all processing steps can be described on a per-input-signal basis. The method starts by applying, to each raw PSG signal, the DWT (with the Daubechies 4 mother wavelet, up to level 3), which results in four WT coefficients (WTCfs) and the four corresponding WT components (WTCps). It then identifies, among these components, the ones that are deemed to contain information about the OAs. They are called here "critical WTCps (CWTCps)". This identification is performed by the subsystem "Identify

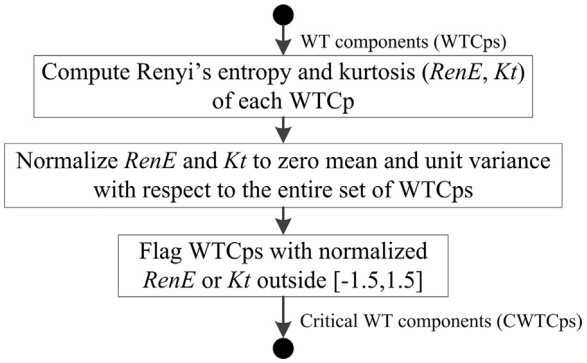


Fig. 11. Block diagram of subsystem "Identify critical WT components".

critical WT components", described later. The input to "Apply stICAra" is thus a set of one or more CWTCps (which are signals) per input. The BD of the subsystem stICAra is shown in Fig. 7. As indicated above, "Apply stICAra" is performed once using all CWTCps, i.e. for all inputs. Each output of "Apply stICAra" is a cleaned version of the corresponding input. The method then replaces, on a per input basis, the critical WTCps in the full set of WTCps by their cleaned versions. The output signal corresponding to each input is then obtained by adding the new WTCps. The output is thus a set of cleaned PSG signals.

Method 7 differs from Methods 6 and 6′ by the order in which it applies WT and ICA. Methods 6 and 6′ start with ICA, whereas Method 7 starts with WT. The existing Method 7.b is the method of [31], called "Automatic wavelet independent analysis (AWICA)". In AWICA, the ICA operations that appear here in "stICAra" (Figs. 10 and 7) are implemented using E-INFOMAX, under the NG hypothesis. The three new Methods 7.a, 7.c, and 7.d are obtained by replacing E-INFOMAX by FastICA, INFOMAX, and EFICA, respectively. For memory, under the NG hypothesis, FastICA, INFOMAX, and E-INFOMAX are the most popular ICA tools, and EFICA the most powerful [16]. The epochs used in [31], and here, have durations of 7.5 s.

The logic behind AWICA is as follows. The main drawback of ICA is that the number of recorded signals must ideally be larger than the number of sources for correctly separating the different types of artifacts. Therefore, ICA has difficulty in separating the OA sources from the true EEG sources. To improve the performance of source separation, AWICA increases the number of signals (to which ICA is applied) by applying ICA to the set of some WTCps (computed using DWT) of the input signals rather than to the input signals themselves. The reason for using the WT is that the PDF of the WT coefficients representing a signal is highly peaked. As said earlier, the ICA tools based on the NG model enhance the non-Gaussianity of the PDFs of the sources. Therefore, by applying ICA to some WTCps rather than to the input signals, the sources will be described by more highly peaked distributions. As a consequence, the performance of ICA and, thus, the separation of artifacts should be improved.

For generating the four WTCps per PSG signal, AWICA first applies the DWT, with the Daubechies 4 mother wavelet, and with $L=3$ decomposition levels. Then, it constructs the WTCps corresponding to the detail part at levels 1, 2, and 3 (d_1, d_2, d_3), and to the approximation part at level 3 (a_3). In [31], the choice of the mother wavelet and the number of decomposition levels for computing the WTCps are justified as follows. The Daubechies 4 wavelet is selected because it generates a family of wavelets that look like the spikes in EEG signals. The WTCps that are generated correspond to the four classical subbands of the EEG spectrum: delta ([0–4] Hz), theta ([4–8] Hz), alpha ([8–12] Hz), and beta ([12–20] Hz). These subbands are respectively close to the frequency ranges of a_3, d_3, d_2, and d_1. The artifact reduction is improved since the frequency range of OAs extends over the four subbands ([0–13/16] Hz). Indeed, let's assume that one epoch of one particular channel of the input PSG signal contains an OA signal. If the spectrum of this OA signal is not strictly confined to one particular subband of interest (delta, theta, alpha, or beta), several WTCps will contain artifactual information, and ICA will benefit from this redundancy. On the contrary, if the spectrum is confined to a single subband of interest, then just one WTCp will be artifactual and ICA will benefit from focusing on a specific frequency range rather than processing the full set of WTCps. Moreover, when ICA does not perform optimally (i.e. does not separate well the artifacts from the neural activity), the artifactual source related to the OA might contain a component coming from neural activity, but focusing on specific frequency ranges reduces the information loss.

3.2.4.2.1. Subsystem "Identify critical WT components". Fig. 11 shows the BD of the subsystem "Identify critical WT components" used by Method 7 (Fig. 10). The input is the set of WTCps, and the output is the set of critical WTCps (CWTCps). The critical WTCps are identified as follows. For each input source, one computes ten statistical measures. For each of the WTCps, one computes two statistical measures quantifying the peakiness and the randomness of the WTCps, i.e. the Renyi's entropy (*RenE*) and the kurtosis (*Kt*). Then, one normalizes these two measures to zero mean and unit variance with respect to the entire set of WTCps. If the amplitude of one of these normalized measures exceeds a fixed threshold (set here to 1.5), the WTCp is declared to be a critical WT component (CWTCp).

3.2.4.2.2. Subsystem "Identify artifactual sources (2)". Method 7 (Fig. 10) uses subsystem stICAra (Fig. 7), which itself uses subsystem "Identify artifactual sources (*i*)". This subsystem is also used by Methods 6 and 6′. However, the implementation of this subsystem is different depending upon whether it is used in Methods 6, 6′, or here in Method 7. Fig. 12 shows the BD of this subsystem, as used in Method 7. The input is the set of sources (produced by applying some ICA tool to the set of the CWTCps), and the output is the set of artifactual sources.

The artifactual sources are identified as follows. One partitions the epoch into 15 non-overlapping sub-epochs of 0.5 s. One then computes the *RenE* and the *Kt* measures for each source on each sub-epoch. One then normalizes them to zero mean and unit variance with respect to the entire set of sources. If the amplitude of one of these measures exceeds a fixed threshold (set to 1.5) in more than 20% of all sub-epochs, the source is declared to be artifactual.

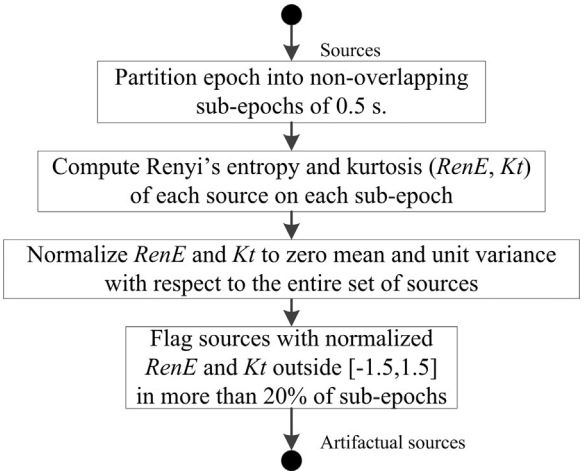


Fig. 12. Block diagram of subsystem "Identify artifactual sources (2)".

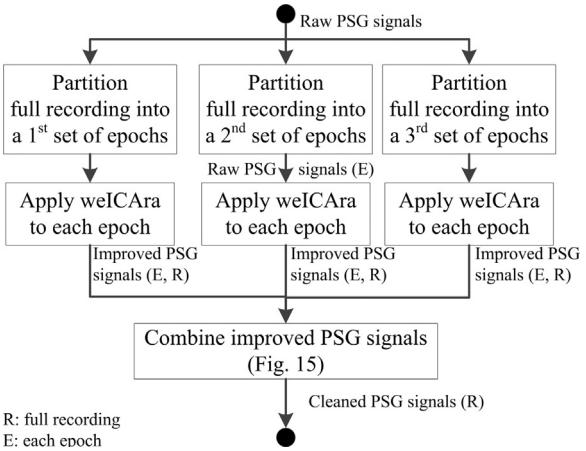


Fig. 13. Block diagram of Method 8.

3.2.4.2.3. Remark about a possible Method 7′. Methods 6 and 6′ differ only by their use of either stICAra or weICAra. By replacing stICAra by weICAra in Method 7, we could create a new Method 7′, also with four new Methods 7′.x. We did not investigate these possible new methods for the following reason. We found (see Section 5) that none of the four variants of Method 7 performed well in detecting OA zones. Since the difference between stICAra and weICAra lies in the way of dealing with artifactual sources after they have been identified, the replacement of stICAra by weICAra only provides an improvement of the cleaned PSG signals in the OA zones. (This will be seen in the illustrative results for Methods 6 and 6′ in Fig. 18.)

3.2.4.3. Methods 8 (based on DWT and ICA). Among the Methods 8.x, the only existing method is Method 8.e, which is the method of [30] (Table 2), further discussed below. All others are new. Fig. 13 shows the BD of main Method 8. The inputs and outputs are identical to those of Methods 6, 6′, and 7, i.e. the sets of raw PSG signals and cleaned PSG signals, respectively.

In contrast with all other main methods, Method 8 is applied to a full recording (instead of a single epoch). The method consists of three steps. First, the method partitions the full recording into contiguous epochs, this in three different ways. Each of the three sets of epochs is referred to as a covering. Second, it applies weICAra (Fig. 8) to each epoch of each covering. This result into one set of improved PSG signals for each epoch, and thus for each covering. Finally, the three sets of improved PSG signals for the full recording are combined into a final set of cleaned PSG signals for this recording. In other words, the signals (of the full recording) are cleaned in two successive ways, with the first performed over three distinct partitionings of the recording.

The existing Method 8.e is the method of [30], and is called "Robust Artifact Removal (RAR)". It was initially developed for long-term neonatal EEG recordings but its principles can be applied to PSG recordings. In RAR, the ICA operations that appear here in weICAra (Fig. 8) are implemented using BGSEP, under the NS hypothesis. Unlike the existing Methods 6.c and 7.b that are entirely based on the NG model, the general scheme of Method 8.e does not require a specific model for the sources produced by ICA. Therefore, the six variants of Methods 8.e are obtained by replacing BGSEP by the four ICA tools under the NG hypothesis (i.e. INFOMAX, E-INFOMAX, FastICA, and EFICA) and the two ICA tools under the SD hypothesis (SOBI and WASOBI). For memory, together with BGSEP, EFICA and WASOBI are the best non-hybrid ICA tools [16]. FastICA, INFOMAX, INFOMAX, and E-INFOMAX are the most popular ICA tools under the NG hypothesis, and SOBI is the most popular ICA tool under the SD hypothesis. A significant reason for including SOBI in our evaluation is that it is the only tool based on the SD model and not restricted to stationary signals, which EEG signals are not! Our innovation here consists in the use of the six above-mentioned ICA tools.

The expressions below give the intervals $[n_1, n_2]$ defining the epochs for the three different coverings,

$$[1+(k_1-1)N_e, k_1N_e], [1+N_e/3+k_2N_e, N_e/3+(k_2+1)N_e],$$
$$[1+2N_e/3+(k_3-1)N_e, 2N_e/3+k_3N_e] k_1=1,\ldots,M, k_2, k_3=1,\ldots,M-1,$$

where N is the length (number of samples) of the recording (and no longer of an epoch), N_e the length of an epoch, and M the number of epochs in the first partitioning ($M=\lfloor N/N_e \rfloor$).

Methods 6′ and 8 both use the subsystem weICAra (Fig. 8), which itself uses the subsystem "Identify artifactual sources (i)". However, the strategy of this subsystem is different for the two methods. Below, we describe the strategy used for Method 8.

3.2.4.3.1. Subsystem "Identify artifactual sources (3)". Fig. 14 shows the BD of this subsystem. The input is the set of sources (produced by applying some ICA tool to one covering), and the output is the set of artifactual sources.

The strategy for identifying which the sources $\underline{s}_k$'s are artifactual is as follows. Consider one specific source $\underline{s}_k$, denote its components by $\underline{s}_{k,n}$, and their magnitudes by $|\underline{s}_{k,}$

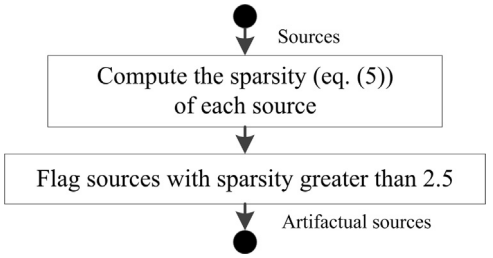


Fig. 14. Block diagram of subsystem "Identify artifactual sources (3)".

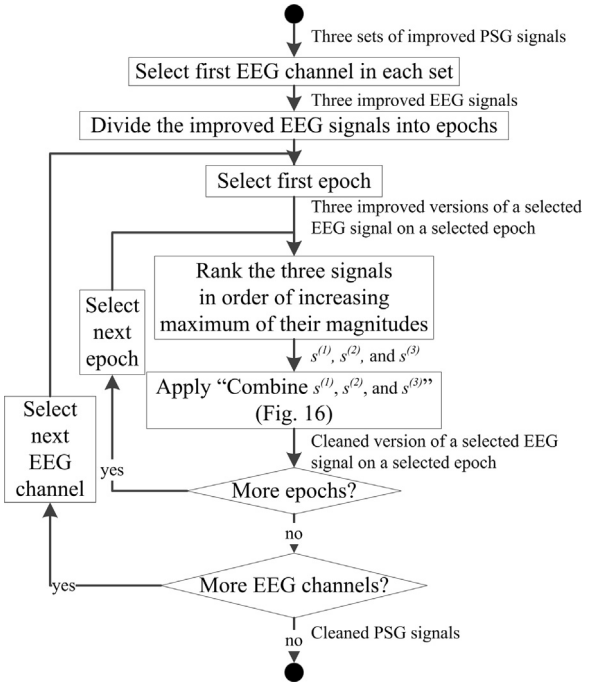


Fig. 15. Block diagram of subsystem "Combine improved PSG signals".

$_n|$, and denote the maximum, standard deviation, and median of the $|\underline{s}_{k,n}|$'s by max_k, std_k, and med_k, respectively. Reference [30] defines the sparsity $sp(\underline{s}_k)$ of $\underline{s}_k$ as

$$sp(\underline{s}_k) = \frac{max_k}{std_k} \log \frac{std_k}{med_k}. \tag{5}$$

A source with sparsity greater than some threshold (set here to 2.5) is declared to be artifactual.

The interpretation of the sparsity measure is as follows. The magnitudes of the components $\underline{s}_{k,n}$ of a sparse source $\underline{s}_k$ have a large maximum max_k relative to the standard deviation std_k, and, simultaneously, a median med_k that is close to zero relative to std_k. In the present problem, the artifactual sources are considered to be sparse because one assumes that artifacts look like spikes having high magnitude and short time duration [30].

Methods 6′ and 8 both use the subsystem weICAra (Fig. 8), which itself uses the subsystem stWD. However, the tools and the parameters used in this subsystem differ for the two methods. For Method 8, we use, as in [30], the

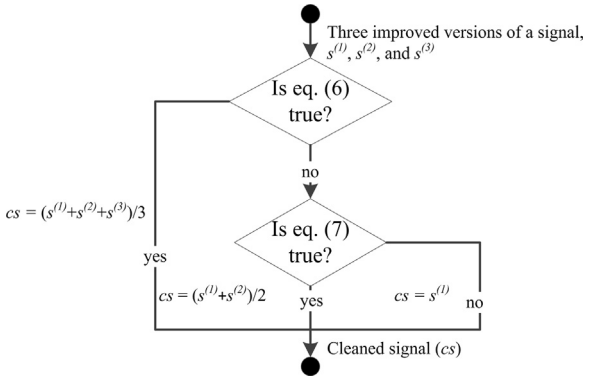


Fig. 16. Block diagram of subsystem "Combine $s^{(1)}$, $s^{(2)}$, and $s^{(3)}$".

DWT, the Daubechies 6 mother wavelet, and $L=7$ decomposition levels.

3.2.4.3.2. Subsystem "Combine improved PSG signals". Fig. 15 shows this subsystem. The input is the three sets of improved PSG signals, and the output is a set of cleaned PSG signals. The subsystem first divides the improved PSG signals into epochs of length T_e, generally with $T_e < N_e$,

$$[1+(k-1)T_e, kT_e], k = 1, ..., \lfloor N/T_e \rfloor.$$

It then proceeds sequentially, channel by channel, and epoch by epoch within each channel.

Fig. 16 shows the subsystem "Combine $s^{(1)}$, $s^{(2)}$, and $s^{(3)}$". It is generic since it deals with arbitrary signals. The input is the set of three improved versions of a signal, ranked in order of increasing maximum of their magnitudes ($s^{(1)}$, $s^{(2)}$, and $s^{(3)}$), and the output is the cleaned signal (cs). The specific combination used by RAR is called "adaptive folding". It allows one to reduce the possible remaining artifacts by averaging the signals containing the fewest artifacts. One assumes that at least one reconstruction, necessarily $s^{(1)}$, is artifact free. Let s denote the signal obtained by selecting, in the input PSG signals of the main Method 8, the EEG channel concerned here. Let $\delta^{(l,m)} = ||s^{(l)} - s^{(m)}||^2$ be the squared norm of the difference between two improved signals $s^{(l)}$ and $s^{(m)}$, and let δ be the average squared norm of signals of length T_e, selected randomly or systematically from s. The final reconstruction (cs) is obtained as the average of either (1) the three reconstructions $s^{(k)}$ if at least one of the following conditions is valid,

$$\max(\delta^{(1,2)}, \delta^{(1,3)}, \delta^{(2,3)}) < 2\delta, \quad \max(\delta^{(1,2)}, \delta^{(1,3)}, \delta^{(2,3)}) < 2\min(\delta^{(1,2)}, \delta^{(1,3)}, \delta^{(2,3)}), \tag{6}$$

or (2) the first two reconstructions if the following condition is valid,

$$\delta^{(1,2)} < \delta^{(2,3)}. \tag{7}$$

Otherwise, the final reconstruction is the first reconstruction. Eq. (6) means that there is no artifact in each of the three signals $s^{(1)}$, $s^{(2)}$, and $s^{(3)}$, whereas Eq. (7) means that only the two first signals ($s^{(1)}$ and $s^{(2)}$) are artifact-free.

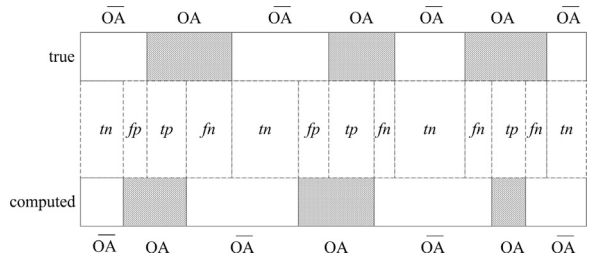


Fig. 17. Quantitative evaluation: segmentations into true (top) and computed (bottom) OA zones and non−OA($\overline{OA}$) zones.

For the implementation of the RAR method, we use an existing Matlab code[12].

4. Method for evaluating and comparing performance

For memory, Method 1 is a detection method, and all others are correction methods. We evaluated the performance of all methods both visually and quantitatively. Since we do not have any simple way of knowing the true, uncorrupted EEG signal (i.e. without OAs), it is essentially impossible to quantify the correction performance of the correction methods. Therefore, for the quantitative evaluation, we turned all correction methods into detection methods by, once again, subtracting the corrected EEG signal from the raw EEG signal and thresholding the result. The resulting binary signal thus segments the time axis of each epoch into zones with OAs and zones without OAs, i.e. into computed OA zones and computed non-OA zones.

To quantify the detection performance of all above methods, we defined our ground truth by manually segmenting many 2 s epochs of 1000 samples each into true OA zones and true non-OA zones. For this, we used a tool included in the Matlab toolbox FieldTrip [33].

The top part of Fig. 17 illustrates the "true" segmentation of, say, one epoch performed manually by an observer into OA zones and non−OA($\overline{OA}$) zones. The bottom part illustrates the corresponding "computed" segmentation performed automatically by some method. The boundaries of the true and computed zones define intervals that can each be labeled as true positive (*tp*), true negative (*tn*), false positive (*fp*), and false negative (*fn*). We transform this labeling into the customary *tp*, *tn*, *fp*, and *fn* numbers by simply adding the lengths of the intervals that have the same label.

These four numbers define a confusion matrix. However, the two fundamental measures of performance that we use to compare the methods are the sensitivity and specificity, defined as follows,

$$\text{sensitivity} = tp \text{ rate} = \frac{tp}{tp+fn},$$

$$\text{specificity} = 1 - fp \text{ rate}, \; fp \text{ rate} = \frac{fp}{tn+fp}.$$

[12] http://si.utia.cas.cz/CODE/cleareeg.m.

We also use the common receiver operating characteristic (ROC) curve, i.e. the graph of the *tp* rate vs. the *fp* rate.

5. Results

We successively performed a visual evaluation of the results obtained for one specific epoch, and a quantitative evaluation of the results obtained for several epochs. We used a total of five PSG recordings containing a total of about 900 epochs of 2 s each.

5.1. Visual evaluation

The plots of Fig. 18 illustrate, for a common 2 s epoch of 1000 samples (from one PSG recording), containing a single OA, the results obtained with each of the 27 methods, grouped according to the eight main methods. Each plot shows the raw, OA-corrupted EEG signal for this epoch (thin line) and the resulting cleaned EEG signal (thick line). (The corresponding EOG signal used by some methods is not shown). Recall that Method 1 is a detection method, and that all others are correction methods. For visualization purpose, we turned Method 1 into a correction method by cancelling, inside the computed OA zone, the values of the input raw EEG signal.

Method 1 (DWT) detects correctly the single OA present. Method 2 (SWT) is not capable of correcting the EEG signal. This observation is in contradiction with the results presented in [6]. However, one should not expect this method to work since it is a denoising method, thus applicable only to white noise, which OAs and their WT coefficients are not! Methods 3.x (LMS) and 4.x (RLS) provide good and very similar results: the spike due to the OA is weakened. Methods 5.x (H$^\infty$–TV) reduce the OA spike, but also perturb useful data. All Methods 6.x and 6'.x (SWT+ICA) remove the spike due to the OA and seem to keep the rest of the signal unchanged. None of the Methods 7.x (DWT+ICA) are able to correct the EEG signal: the cleaned EEG signal has the same shape as the original one. Methods 8.x (DWT+ICA) do not clear EEG signals correctly either: the spike due to the OA appears to be reversed, and useful data is also perturbed.

5.2. Quantitative evaluation

Fig. 19 presents five plots, each corresponding to a given PSG recording (i.e. for a given subject), with each plot showing the corresponding ROC curves for the best method within each main method (i.e. the method with the best quantitative performance in sensitivity and specificity). Below, we often need to describe the relative positions of ROC curves: (1) when two ROC curves do not cross, one can talk about one curve being above or below the other; (2) in the ideal case, the ROC curve would reduce to the point in the upper-left corner, and in near-ideal cases, the curve is close to the left and top edges of the plot frame (and one can also talk about a large area under the curve).

The plot for Subject B suggests that Method 5.2 (SWT+H$^\infty$–TV) produces better results for correcting

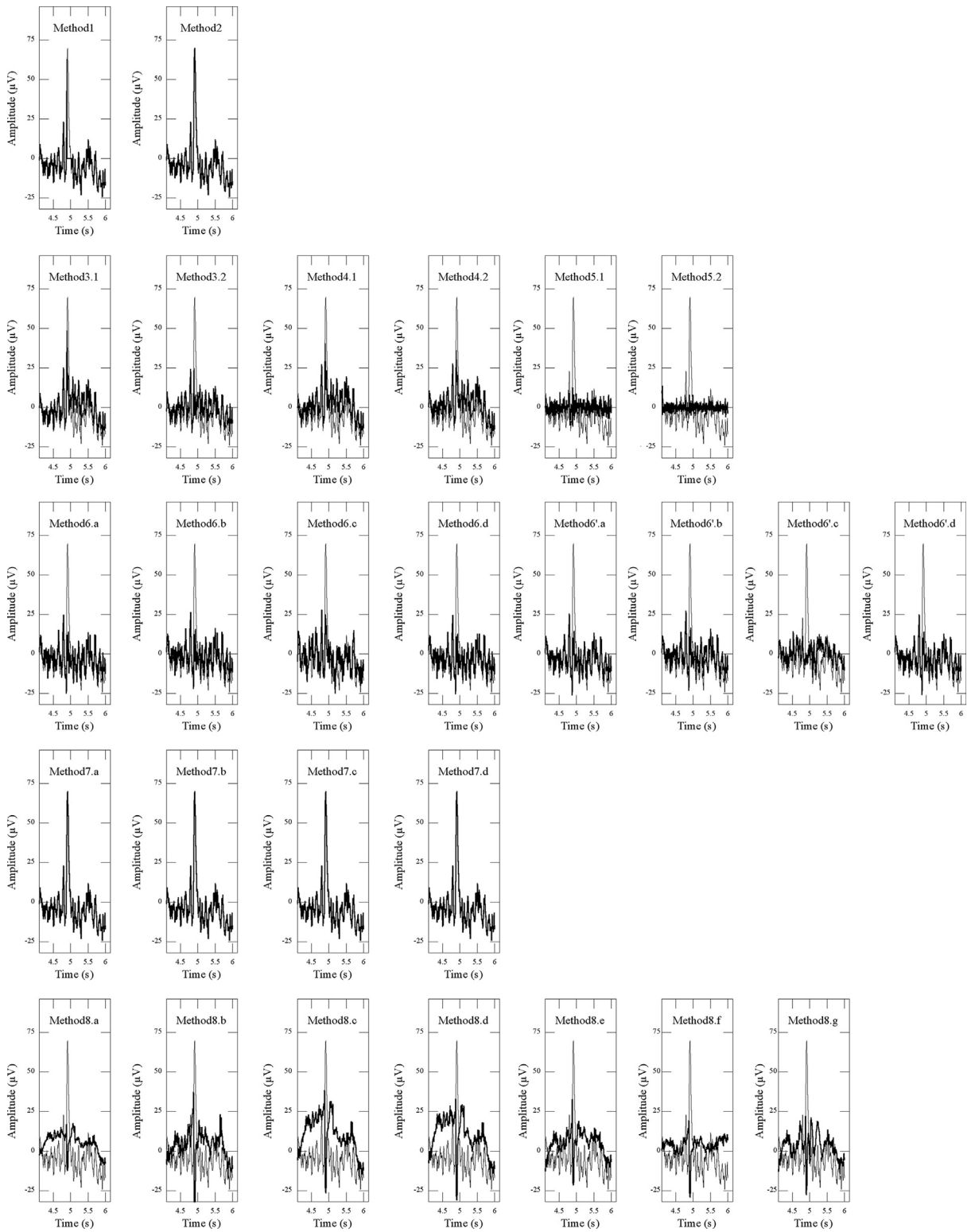


Fig. 18. Illustrative results produced by the 27 methods, grouped according to the eight main methods, on one epoch of 1000 samples from one of the five PSG recordings available. The thin (thick) lines show the raw (cleaned) EEG signals.

OAs than the other methods. Indeed, the corresponding ROC curve is located closer to the left and top edges of the plot frame than the other curves are. Method 5.2 produces similar results for Subjects D and M. The ROC curve for Methods 6′.x (SWT+wICAra) intersects the ROC curve for Method 5.2 and, by visual inspection, one can see that the

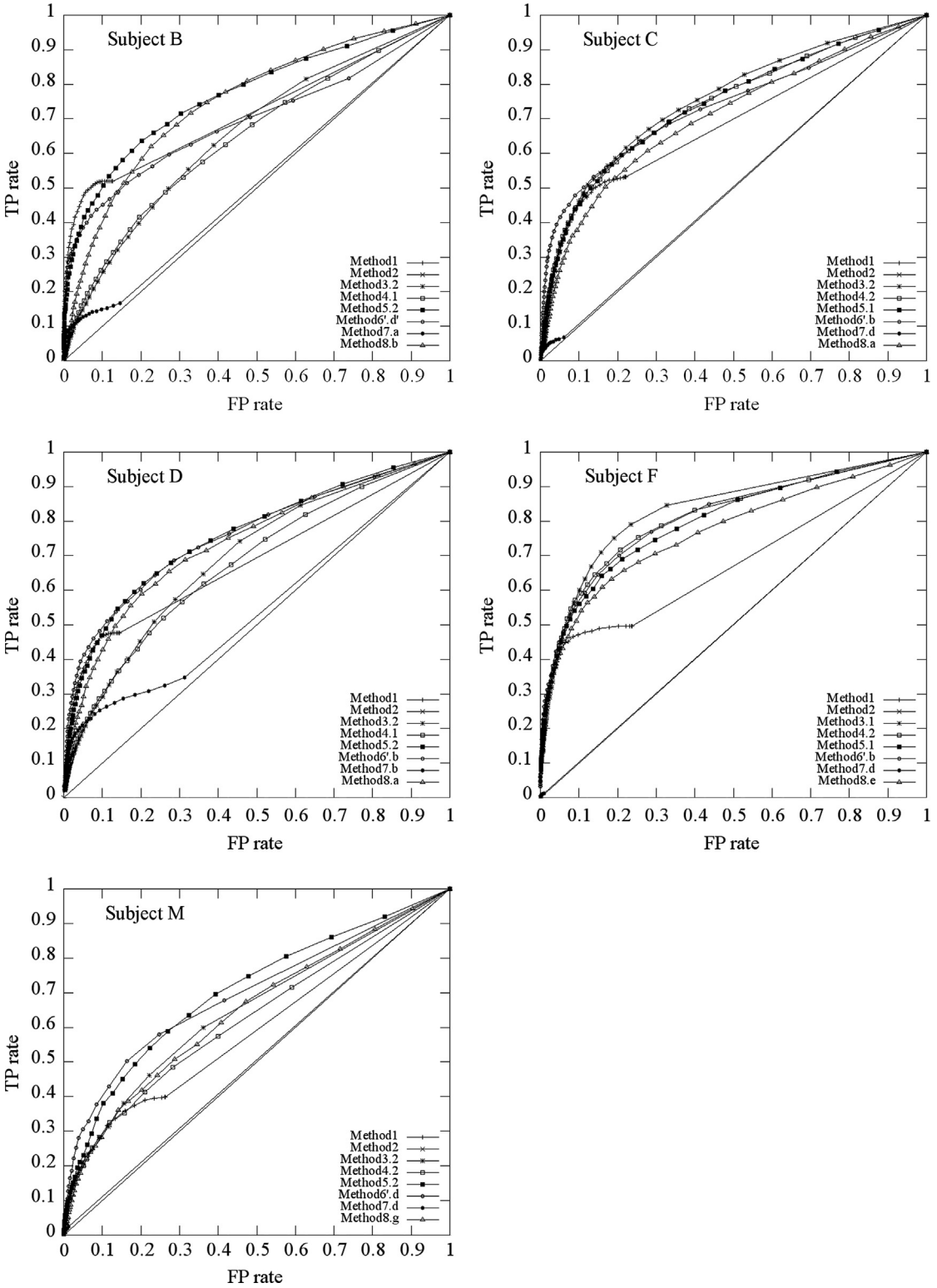


Fig. 19. ROC curves for the best method within each main method for the five PSG recordings.

areas under both curves are large and almost equal. Therefore, for these two subjects, Methods 6′.x performs almost as well as Method 5.2. The plot for Subject C shows that the curve for Method 3.2 (SWT+LMS) is slightly above the others. Similarly, the plot for Subject F shows that the curve for Method 3.1 (LMS) is also slightly above

Table 4
Sensitivity, specificity, and their sum for the 27 methods.

Methods	Sensitivity	Specificity	Sens. + Spec.
Method 1	0.445 ± 0.075	0.904 ± 0.034	1.349 ± 0.086
Method 2	1.000 ± 0.000	0.000 ± 0.000	1.000 ± 0.000
Method 3.1	0.555 ± 0.165	0.753 ± 0.088	1.308 ± 0.159
Method 3.2	0.615 ± 0.095	0.724 ± 0.092	1.339 ± 0.128
Method 4.1	0.533 ± 0.107	0.778 ± 0.043	1.311 ± 0.125
Method 4.2	0.523 ± 0.148	0.792 ± 0.052	1.315 ± 0.128
Method 5.1	0.590 ± 0.033	0.797 ± 0.041	1.387 ± 0.071
Method 5.2	0.615 ± 0.031	0.789 ± 0.033	1.404 ± 0.057
Method 6.a	0.534 ± 0.081	0.850 ± 0.028	1.384 ± 0.063
Method 6.b	0.560 ± 0.098	0.831 ± 0.044	1.391 ± 0.066
Method 6.c	0.520 ± 0.071	0.828 ± 0.062	1.348 ± 0.055
Method 6.d	0.562 ± 0.077	0.830 ± 0.065	1.392 ± 0.056
Method 6.a	0.536 ± 0.079	0.851 ± 0.027	1.387 ± 0.062
Method 6′.b	0.573 ± 0.099	0.820 ± 0.046	1.393 ± 0.068
Method 6′.c	0.491 ± 0.092	0.841 ± 0.039	1.332 ± 0.063
Method 6′.d	0.568 ± 0.109	0.824 ± 0.056	1.392 ± 0.061
Method 7.a	0.277 ± 0.418	0.778 ± 0.435	1.055 ± 0.067
Method 7.b	0.277 ± 0.414	0.774 ± 0.434	1.051 ± 0.066
Method 7.c	0.272 ± 0.415	0.777 ± 0.435	1.049 ± 0.060
Method 7.d	0.076 ± 0.081	0.979 ± 0.018	1.055 ± 0.064
Method 8.a	0.571 ± 0.046	0.813 ± 0.025	1.384 ± 0.066
Method 8.b	0.573 ± 0.063	0.754 ± 0.073	1.327 ± 0.078
Method 8.c	0.547 ± 0.089	0.768 ± 0.044	1.315 ± 0.068
Method 8.d	0.573 ± 0.055	0.781 ± 0.038	1.354 ± 0.068
Method 8.e	0.585 ± 0.062	0.764 ± 0.047	1.349 ± 0.074
Method 8.f	0.955 ± 0.020	0.115 ± 0.024	1.070 ± 0.015
Method 8.g	0.561 ± 0.084	0.743 ± 0.067	1.304 ± 0.098

the others. This means that Methods 3.x (LMS) better correct the EEG signal for Subjects C and F than the other methods do. For all subjects, the curves for Method 2 (SWT) and Methods 7.x (DWT+ICA) are below the curves for all other methods. Methods 2 and 7.x are thus the worst for all subjects.

Theoretically, the sensitivity and the specificity exhibit opposite behaviors. Therefore, another way of comparing the performances of the methods is to consider the sum of these measures. Then, the larger the sum is, the better the performance is. Table 4 lists the average and standard deviation (on the five PSG recordings available) of the sensitivity, the specificity, and their sum. The rows of the four best methods are shown in gray, with the performance increasing from light gray to dark gray. The best values of the sum of the sensitivity and the specificity correspond to Methods 5.2, 6′.b, 6.d, and 6′.d, and the worst correspond to Methods 2 and 7.x.

5.3. Discussion

Method 1 (DWT), which is a detection method based on a thresholding of wavelet approximation coefficients, does not seem to correctly identify all OAs in the considered PSG recordings (in comparison to the "true", manual labelings). Indeed, in Fig. 19, one can observe that the area under the ROC curve corresponding to Method 1 is quite small compared to the areas corresponding to the other methods, especially for Subjects F and M. Moreover, the sum of the sensitivity and the specificity is only in the middle of the ranking in Table 4.

Fig. 18 shows that most of the other methods – which are correction methods – are able to eliminate a substantial part of the OA signal. However, some methods are not capable of correcting the EEG recording. Method 2 (SWT) provides a cleaned EEG signal that is exactly the same as the original one. This can also be observed for all Methods 7.x (DWT+ICA). Methods 5.x (H^{∞}–TV) reduce the spike due to the OA but modify the rest of the data. This is also the case for all Methods 8.x (DWT+ICA). Nevertheless, one should note that the graphs in Fig. 18 are from one epoch from one EEG recording. These results are thus largely anecdotal. In addition, as indicated earlier, it is difficult to evaluate the performance of the correction methods because we cannot measure the activities of the brain and of the eyes separately. We will thus discuss the performance of these methods of correction in terms of their ability to correctly detect the OAs in the EEG recordings.

From Table 4, one can notice that the best results are obtained for Method 5.2 (SWT+H^{∞}–TV) and for some of the Methods 6.x and 6′.x (SWT+ICA). The sum of the values of sensitivity and specificity is larger for these methods than for the others. The better performance is confirmed in Fig. 19, where one can see that the ROC curves for Methods 5.x (H^{∞}–TV) and 6′.x (SWT+wICArа) are located close to the left and top edges of the plots for most subjects. The worst performing methods are Method 2 (SWT) and all Methods 7.x (DWT+ICA), since, in comparison to all other methods, they have ROC curves with smaller areas (Fig. 19) and small values for both the sensitivity and the specificity (Table 4). However, the results are not always as clear for all other methods. For instance, one can observe in Fig. 19 that the ROC curves for Method 3.x (LMS) are slightly above those of the other methods for Subjects C and F, but below for the other subjects. Table 4 shows that the sum of sensitivity and specificity is quite small for Methods 3.x compared to the other methods. This is probably due to the fact that the values in Table 4 are averages over the five subjects. We can therefore state that Method 3.x does not perform equally well for all subjects. This is also confirmed by the large values of the standard deviations in Table 4.

From Table 4 and Fig. 19, one can also conclude that combining any of the adaptive filters with the SWT slightly improves the results as compared to using an adaptive filter alone. (The Methods 3.2, 4.2, and 5.2 give the best performances in most cases.) Moreover, the sum of the values of sensitivity and specificity for the Methods 6.x and 6′.x in Table 4 is even larger for our new methods than for Method 6.c, which is the only existing method under Method 6.

6. Conclusions

Detecting and/or correcting ocular artifacts (OAs) in polysomnographic (PSG) recordings is critical for the meaningful analysis of these signals. The OAs mask the true, underlying EEG signal due to the sole neuronal/electrical activity of the brain. As a result, the OAs make the analysis of EEG recordings more difficult and, more importantly, they can lead to incorrect analysis and wrong

conclusions. To avoid losing valuable data, it is critical to develop robust methods for cleaning out EEG signals from OAs.

We identified 11 existing methods for detecting and/or correcting OAs in PSG signals. We found out that these methods use one or two "tools" among the following tool families: wavelet transform (WT), adaptive filtering (AF), and independent component analysis (ICA). By combining these tools in different ways, and by using different implementations of the tools (for ICA), we created 16 novel methods for detecting and/or correcting OAs in PSG signals.

We performed a comparative, study of the performance of these 27 methods (existing and new) using a common set of PSG recordings and a common set of performance measures. The data consists in five PSG recordings. They were recorded during a driving task of about 30 minutes each, in a driving simulator. The performance measures were developed for emphazing the ability of the correction and the detection methods to correctly detect OA zones in EEG recordings, as compared to a ground truth established visually. The performance measures are derived from the usual metrics of *tp* rate, *fp* rate, sensitivity, and specificity.

The highest performing methods are Methods 5.2, 6′.b, 6′.b, 6.d, and 6′.d which comprise two existing methods and three new ones. The sum of the sensitivity and the specificity computed for these methods is 1.404 ± 0.057, 1.393 ± 0.068, 1.392 ± 0.056, 1.392 ± 0.061, respectively. The best method is the existing Method 5.2, which uses a combination of a WT (SWT) and an AF (H^{∞}–TV). The other four methods all fall equally under Methods 6 and 6′, which use a combination of WT (SWT) and ICA (E-INFOMAX and EFICA).

The lowest performing methods are Methods 2 (SWT), 7.a, 7.b, 7.c, and 7.d (DWT + ICA) which comprise two existing methods and three new ones. The sum of the sensitivity and the specificity computed for these methods is 1.000 ± 0.000, 1.055 ± 0.067, 1.051 ± 0.067, 1.049 ± 0.060, and 1.055 ± 0.064, respectively. The worst method is the existing Method 2.

Our comprehensive, comparative study can be used as a benchmark for evaluating the performance of any new method that might be proposed in the future for detecting and/or correcting OAs in PSG signals.

Acknowledgments

We thank IFSTTAR for making available to us one of their professional driving simulators, and the "Centre d'Etudes des Troubles de l'Eveil et du Sommeil" (CETES) for making available their facilities and equipments.

References

[1] R.J. Croft, R.J. Barry, Removal of ocular artifact from the EEG: a review, Clin. Physiol. 30 (1) (2000) 5–19.
[2] A. Kandaswamy, V. Krishnaveni, S. Jayaraman, N. Malmurugan, K. Ramadoss, Removal of ocular artifacts from EEG: a survey, IETE J. Res. 51 (2) (2005) 121–130.
[3] M.T. Akhtar, W. Mitsuhashi, C.J. James, Employing spatially constrained ICA and wavelet denoising, for automatic removal of artifacts from multichannel EEG data, Signal Process 92 (2) (2012) 401–416.
[4] S. Mallat., A Wavelet Tour of Signal Processing, second ed. Academic Press, 1999.
[5] G.P. Nason, B.W. Silverman., The stationary wavelet transform and some statistical applications, Lecture Notes in Statistics 103 (1995) 281–300.
[6] V. Krishnaveni, S. Jayaraman, S. Aravind, V. Hariharasudhan, K. Ramadoss, Automatic identification and removal of ocular artifacts from EEG using wavelet transform, Meas. Sci. Rev 6 (2) (2006) 45–57. (no. 4).
[7] V. Krishnaveni, S. Jayaraman, L. Anitha, K. Ramadoss, Removal of ocular artifacts from EEG using adaptive thresholding of wavelet coefficients, J. Neural Eng 3 (4) (2006) 338–346.
[8] D.L. Donoho, I.M. Johnstone, Adapting to unknown smoothness via wavelet shrinkage, J. Am. Statist 90 (432) (1995) 1200–1224Assoc 90 (432) (1995) 1200–1224.
[9] Z. Xiao-Ping, M.D. Desai, Adaptive denoising based on SURE risk, IEEE Signal Process. Lett 5 (10) (1998) 265–267.
[10] C.M. Stein, Estimation of the mean of a multivariate normal distribution, Ann. Stat. 9 (6) (1981) 1135–1151.
[11] A.V. Oppenheim, R.W. Schafer, et al., Discrete-time Signal Processing, second ed. Prentice-Hall, 1999.
[12] S. Puthusserypady, T. Ratnarajah, Robust adaptive techniques for minimization of EOG artifacts from EEG signals, Signal Process 86 (9) (2006) 2351–2363.
[13] M.A. Klados, C. Papadelis, C. Lythari, P.D. Bamidis. The removal of ocular artifacts from EEG signals: a comparison of performances for different methods, in: Fourth European Conf. of the International Federation for Medical and Biological Engineering. J. Sloten, P. Verdonck, M. Nyssen and J. Haueisen, Springer Berlin Heidelberg, 22 (2008) 1259–1263.
[14] M.A.G. Correa, E.L. Leber. Noise removal from EEG signals in polysomnographic records applying adaptive filters in cascade. Adaptive Filtering Applications, L.G. (Ed.), 2011.
[15] P.S. Kumar, R. Arumuganathan, K. Sivakumar, C. Vimal. Removal of artifacts from EEG signals using adaptive filter through wavelet transform signal processing. In: ninth IEEE International Conference on Signal Process., 2008.
[16] P. Tichavský, Z. Koldovský., Fast and accurate methods of independent component analysis: a survey, Kybernetika 47 (2011) 426–438.
[17] T.W. Lee, Independent Component Analysis: Theory and Applications, Kluwer Academic Publishers, 1998.
[18] A. Hyvärinen, E. Oja, Independent component analysis: algorithms and applications, Neural Networks 13 (2000) 411–430.
[19] A. Bell, T. Sejnowski., An information-maximization approach to blind separation and blind deconvolution, Neural Comput. 7 (6) (1995) 1129–1159.
[20] T.-W. Lee, M. Girolami, T.-J. Sejnowski, Independent component analysis using an extended infomax algorithm for mixed sub-Gaussian and super-Gaussian sources, Neural Comput. 11 (1999) 417–441.
[21] Z. Koldovský, P. Tichavský, E. Oja, Efficient variant of algorithm FastICA for independent component analysis attaining the Cramér–Rao lower bound, IEEE Trans. Neural Networks 17 (2006) 1265–1277.
[22] J.-F. Cardoso, High-order contrasts for independent component analysis, Neural Computat 11 (1) (1999) 157–192.
[23] D.-T. Pham, J.F. Cardoso, Blind separation of instantaneous mixtures of non stationary sources, IEEE Trans. Signal Process 49 (2001) 1837–1848.
[24] D.T. Pham, Joint approximate diagonalization of positive definite Hermitian matrices, SIAM J. Matrix Anal. Appl 22 (2001) 1136–1152.
[25] P. Tichavský, A. Yeredor, Fast approximate joint diagonalization incorporating weight matrices, IEEE Trans. Signal Process 57 (2009) 878–891.
[26] A. Belouchrani, K. Abed-Meraim, J.F. Cardoso., A blind source separation technique using second-order statistics, IEEE Signal Process 45 (2) (1997) 434–444.
[27] D.T. Pham, Blind separation of instantaneous mixture of sources via the Gaussian mutual information criterion, Signal Process 81 (2001) 855–870.
[28] S. Dégerine, A. Zaïdi, Separation of an instantaneous mixture of Gaussian autoregressive sources by the exact maximum likelihood approach, IEEE Trans. Signal Process 52 (2004) 1492–1512.
[29] H. Ghandeharion, A. Erfanian., A fully automatic ocular artifact suppression from EEG data using higher order statistics: improved

performance by wavelet analysis, Med. Eng. Phys 32 (7) (2010) 720–729.
[30] M. Zima, P. Tichavský, K. Paul, V. Krajča, Robust removal of short-duration artifacts in long neonatal EEG recordings using wavelet-enhanced ICA and adaptive combining of tentative reconstructions, Phys. Meas. 33 (8) (2012) 39–49.
[31] N. Mammone, F. La Foresta, F.C. Morabito, Automatic artifact rejection from multichannel scalp EEG by wavelet ICA, IEEE Sens. J. 12 (3) (2012) 533–542.
[32] N.P. Castellanos, V.A. Makarov, Recovering EEG brain signals: artifact suppression with wavelet enhanced independent component analysis, J. Neurosci. Methods 158 (2006) 300–312.
[33] R. Oostenveld, P. Fries, E. Maris, J.M. Schoffelen. FieldTrip: open source software for advanced analysis of MEG, EEG, and invasive electrophysiological data. Comput. Intell. Neurosci. (2011).