COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE XXX (2015) XXX-XXX





دائلو دکنده مقالات علم FREE reepaper.me

journal homepage: www.intl.elsevierhealth.com/journals/cmpb

Design of a novel biomedical signal processing and analysis tool for functional neuroimaging

Sezgin Kaçar^{a,*}, Ünal Sakoğlu^{b,1}

^a Department of Electrical and Electronics Engineering, Technology Faculty, Sakarya University, Turkey
 ^b Department of Computer Science, Texas A&M University – Commerce, Commerce, TX, USA

ARTICLE INFO

Article history: Received 21 July 2015 Received in revised form 8 October 2015 Accepted 13 November 2015

Keywords: Biomedical signal processing and analysis EEG ECG Graphical user interface Non-linear time-series analysis Multivariate analysis

ABSTRACT

In this paper, a MATLAB-based graphical user interface (GUI) software tool for general biomedical signal processing and analysis of functional neuroimaging data is introduced. Specifically, electroencephalography (EEG) and electrocardiography (ECG) signals can be processed and analyzed by the developed tool, which incorporates commonly used temporal and frequency analysis methods. In addition to common methods, the tool also provides non-linear chaos analysis with Lyapunov exponents and entropies; multivariate analysis with principal and independent component analyses; and pattern classification with discriminant analysis. This tool can also be utilized for training in biomedical engineering education. This easy-to-use and easy-to-learn, intuitive tool is described in detail in this paper.

© 2015 Elsevier Ireland Ltd. All rights reserved.

1. Introduction

Biomedical signals are used by engineers for designing bioelectrical and bio-mechanical systems, and by physicians and healthcare specialists for diagnosing health problems. Biomedical signals must be processed before using them for any analysis, design or diagnosis. Majority of the engineers/scientists working on biomedical applications can easily process the biomedical signals according to their purposes using existing signal processing tools/programs or by developing in-house programs. Even though many tools for processing and analyses exist, they are usually sophisticated, need significant amount of time to learn, and there is still the need for programming to use those tools. On the other hand, physicians and healthcare specialists do not have as good programming background as the engineers and programmers do. For this reason, a comprehensive, easy-to-learn/use, intuitive graphical user interface (GUI) tool is needed for biomedical signal processing and analysis. Such a GUI tool can be used without any programming knowledge. Additionally, it can be used as an educational and demonstration tool for biomedical engineering students.

There are many GUI software/tools for processing/analysis and educational purposes in biomedical signal processing field. In 2003, a C++-based GUI software, Biosignal Laboratory, was designed for biomedical signal processing and analysis by a group of scientists [1]. This software allows to process data in different file formats and it includes features such as basic functions, preprocessing, transforms, segmentation, feature

* Corresponding author. Tel.: +90 264 295 64 55; fax: +90 264 295 64 24.

E-mail addresses: skacar@sakarya.edu.tr (S. Kaçar), unal.sakoglu@tamuc.edu (Ü. Sakoğlu).

¹ Tel.: +1 903 886 5242.

http://dx.doi.org/10.1016/j.cmpb.2015.11.011

0169-2607/© 2015 Elsevier Ireland Ltd. All rights reserved.

ARTICLE IN PRESS

selection and extraction, pattern matching and clustering. Another visual tool named BioLab has been developed as an educational tool for training of biomedical signal processing [2]. This MATLAB-based GUI tool provides signal processing functions such as preprocessing, independent component analysis (ICA), signal averaging and spectral estimation functions. In addition, LabVIEW has been also used for creating biomedical signal acquisition and processing tool [3]. This LabVIEW-based tool is very effective for signal acquisition and processing, but the users of the tool need LabVIEW experience. Another educational and analysis tool in biomedical signal processing field is called UW Digiscope 3.0 [4]. This tool, which dates back to early 1990s and MS-DOS operating system, has been updated using MATLAB functions and it now features a GUI. The GUI enables digital filtering and visualization of the time and frequency results of the biomedical signals.

There are also detailed and complicated GUI tools for processing and analysis of specific biomedical signals such as electrocardiogram (ECG), electromyogram (EMG), electroencephalogram (EEG). One of the graphical tools designed for ECG signal analysis has been designed by Potse et al. in 2002 [5]. The tool is a MATLAB based GUI for multichannel ECG signals analysis. An online system based on MATLAB Web Server has also been developed for remote clinical diagnosis support from ECG signals [6]. By using this system, ECG data files can be analyzed and the results can be received via internet with only a web browser. Niskanen et al. [7], have developed software for advanced heart rate variability (HRV) analysis from ECG signals. Statistical, geometric, nonlinear and spectral measures have been obtained by this MATLAB/C-based software with time-domain, frequency-domain and non-linear methods. There are varieties of MATLAB-based GUI tools which range from simple/basic to detailed/complex for ECG signal analysis [8,9]. In the biomedical signal processing literature, there are also studies about processing and analysis of EMG signals. For example, Elcano et al. [10], have designed an interactive visual tool for EMG signal analysis. This C++-based tool enables basic computation, editing and manual correction of the EMG signals. EEG signals processing is also a popular part of biomedical signal processing and various tools have been developed for advanced EEG signals processing [11]. One of the most widely used EEG signal analysis tools, EEGLAB, has been developed as a MATLAB-based GUI tool with an advanced technical background [11,12]. EEGLAB has been improved and it now encompasses many different signal processing and analysis methods. It can be said that EEGLAB is currently the most sophisticated open source biomedical signal processing tool. Another advanced GUI tool, ELAN, can analyze EEG, magnetoencephalography (MEG), intracranial EEG (iEEG) and local field potentials (LFPs) [13]. With this C-based GUI tool, one can visualize topographical activation maps and time-frequency analysis based on the signals.

In this paper, a comprehensive, general-purpose, userfriendly, GUI biomedical signal processing tool, which is easy to learn and use has been developed based on advanced signal processing and analysis methods some of which have not been applied in any other biomedical signal processing tool before. The features provided by the developed tool are: (a) preprocessing: rectification, normalization, filtering; (b) statistical analysis: minimum, maximum, mean, and standard deviation; (c) time-domain analyses: integral, correlation analysis; (d) frequency-domain analysis: Fourier transform; (e) non-linear analysis methods: Lyapunov exponents analysis, entropy/information analysis; (f) multivariate analysis: principal and independent component analysis; (g) pattern classification: discriminant analysis. An additional important feature of this MATLAB-based GUI tool is that it provides users an easy way to import the biomedical datasets, select desired portions of the imported datasets, select desired processing and analysis methods, and save all the results by clicking a few buttons. It lets the user visualize the desired results using plots.

The paper is organized as follows: in Section 2, the technical background regarding the methods implemented in the tool has been described; in Section 3, the details of the designed GUI have been presented; discussions and conclusions are in Section 4.

2. Technical background of the designed tool

2.1. Computer-aided diagnosis and therapy based on biomedical signal processing and analysis

Biomedical signals are continuous-time, highly variable, complex and generally noisy quantities measured from biological systems, especially from humans for obtaining information [14]. These quantities can be physical (such as temperature, pressure), electrical (such as potential, current) and biochemical (such as hormones, neurotransmitters). The most important examples of the biomedical signals are counted as action potential, electroneurogram (ENG), electromyogram (EMG), electrocardiogram (ECG), electroencephalogram (EEG), event-related potentials (ERPs), electrogastrogram (EGG), phonocardiogram (PCG), carotid pulse (CP), signals from catheter-tip sensors, speech signal, vibromyogram (VMG), vibroarthrogram VAG, oto-acoustic emission signal [15]. There can be many different type biomedical signals which can be measured by specific sensors. Functional magnetic resonance imaging (fMRI) data also provide functional data with high spatial resolution and relatively low temporal resolution, and fMRI data have been mostly used to study functioning of healthy and diseased brain, under various task conditions and under rest [16-24]. It can be said that all of the functional biomedical signals are time series and they are processed by similar signal processing and analysis techniques. Certain types of analysis methods for certain types of imaging techniques including EEG, MEG and ECG have been also used for diagnosis, which has been incorporated into therapy [15]. The process of obtaining useful information and deducing a diagnosis from various biomedical signals using computers and using it for therapy is called computer-aided diagnosis (CAD) and therapy [15]. Fig. 1 shows the computer-aided diagnosis and therapy based on biomedical signal analysis.

As seen from Fig. 1, CAD process consists of three stages. The first stage of the CAD process, signal data acquisition, is not the subject of this paper. The second stage, which is the signal processing, has two main steps: (i) filtering to remove artifacts and (ii) detection of events and components. These

COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE XXX (2015) XXX-XXX



Fig. 1 – Computer-aided diagnosis (CAD) and therapy based on biomedical signal analysis. Modified and adopted from [15].

two steps can be named as biomedical signal processing. The third stage, biomedical signal analysis, has also two steps such as (i) feature extraction and (ii) pattern classification. The visual tool presented in this paper includes all biomedical signal processing and analysis steps in the CAD process, as seen in Fig. 1.

2.2. Methods used in the designed tool

There are many techniques which are used for biomedical signal processing and analysis. They are mentioned and classified differently in different sources [14,15,25–30]. For example, it can be said that preprocessing methods, basic statistical analysis, time-domain methods, frequency-domain methods, spectral analysis methods, multivariate analysis methods, nonlinear or chaos and complexity analysis methods, classification methods are the main classes of biomedical signal processing and analysis. There are various specific methods under these categories. Including all of the techniques under every class in a biomedical signal processing and analysis tool makes the tool very complicated and hard to use. The tool we designed and described here proposes a user-friendly graphical interface. Commonly used methods are implemented in this tool, with the addition of largest Lyapunov exponents as a novelty for nonlinear analysis. Fig. 2 summarizes the methods included in our tool with a simple diagram.

Preprocessing involves "cleaning up" the data before the subsequent processing and analysis technique. The designed tool has three preprocessing methods: filtering, normalization, rectification. The most commonly used preprocessing technique is filtering which is used to remove artifacts and noise in biomedical signals [15]. The Butterworth filter type is used in the developed tool for filtering. Low-pass, high-pass, band-pass and band-stop filtering can be chosen as different options. Desired frequency bands to remove from the signal can be specified. Normalization module of the tool can transform the dynamic range of the signal between 0 and 1 or -1and 1. The signals can be formed as single polarity (positive in general) half-wave or full-wave signals by using the rectification module. Normalization and rectification steps may be necessary for some biomedical signal processing and analysis techniques.

دائلو دکنده مقالات علم FRE مانلو دکنده مقالات علم

3



computer methods and programs in biomedicine xxx (2015) xxx-xxx



Fig. 2 - The methods included by the designed tool with a basic classification diagram.

With the statistics module of the developed tool, the basic statistical parameters such as minimum, maximum, mean values and standard deviation can be calculated. These basic statistics provide useful summary information about biomedical signals [14]. Simple subsequent analyses can be performed by using these parameters in the designed tool.

The time-domain methods module of the tool can be used for time-domain analysis of the signals, which are already converted to discrete-time, but can still have continuous or discrete values [25,28]. Integration and correlation are two fundamental time-domain methods which are implemented in the developed tool. Integration can be used, for example, in calculating the power of the biomedical signal after fullwave rectification, or, to convert the signal to another unit (e.g. current to voltage). Correlation is used to characterize the relationship between the two signals and it can convey the information that whether two signals are similar, or dissimilar, and it can provide a clue about whether the signals are dependent on one another [29,30].

Frequency-domain signals are defined as a specific set of frequencies and these type signals are processed by the frequency-domain methods [26,28]. The developed tool implements the Fourier transform since it is the most commonly used frequency-domain analysis method. The Fourier transform module of the tool shows the amplitudes and frequencies of sinusoidal components or basis functions which make up the analyzed signal [29].

Multivariate analysis describes the analysis techniques that are used to analyze multiple signals measured concurrently from different parts of a system. Multivariate analysis techniques are used to transform a multivariate dataset to a smaller and more understandable form [29]. Principle component analysis (PCA) and independent component analysis (ICA) are the multivariate analysis techniques implemented in the multivariate analysis module of the developed tool. The signal can be decomposed linearly and without loss by PCA into components by using the first- and second-order statistics [29,30]. The principle components which constitute the majority of the information in the signal can be selected by using the PCA functionality in the module. Therefore, the "noisy" components, i.e. the components which do not contribute to the information in the signal much can be thrown out and thus the data size can be reduced. The purpose of ICA, on the other hand, is to convert the signals into number of components which are statistically independent, i.e. mutual information between different components is minimal. ICA is more useful if one thinks that the signals acquired are a linear mix of different independently acting sources, and it can identify these sources. ICA can also find artifact sources, and these components can be subtracted from the signal. ICA does not reduce the dimensions of the data set as PCA, but it can help identifying and throwing the artifact sources [29,30].

Many analyses methods have been implemented and incorporated in analysis tools for different functional neuroimaging data obtained such as the ones obtained by functional magnetic resonance imaging (fMRI) techniques (i.e. AFNI) [31] and near infrared spectroscopy (NIRS) techniques [32]. Some other analysis tools have also been developed to handle many different techniques of functional neuroimaging, including fMRI, PET, EEG and MEG; such as the Statistical Parametric Mapping toolbox [33]. All of these tools implement important analysis algorithms and methods which are useful to analyze different kinds of properties of neuroimaging data in an efficient manner. However, these tools lack nonlinear chaos analyses and pattern recognition/classification (machine learning) analyses, which are very useful analyses that are desired by many researchers in the functional neuroimaging field. There are codes and snippets which can do these kinds of analyses, but not specifically for neuroimaging data; therefore it takes an extra effort by researchers to modify their datasets into a suitable format for analyses by these tools, or to modify the programs/codes extensively in order to be able to analyze their datasets.

COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE XXX (2015) XXX-XXX

دائلو دکننده مقالات علمی freepaper.me paper





|--|

COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE XXX (2015) XXX-XXX

Number of Face Iterature	Import Data From File	Signal Features Analysis Graphs Selection Cell in Ta	Classification Training Set
ref refs: Sample fail/requency (rig: Naminatation Sample fail/requency (rig:	Photo Description	Maximum Values Entropy Values Lyacunov Exponents RUN No Analysis	Class Name
Sampla Rathfraguncy (R):	File Path: Number of Files:	Minimum Values OAI Rows	Class Elements:
Select Jaha Press Columns Maximum Values 1 1 2 1 2 <tr< td=""><td>Sampling Rate/Frequency (Hz):</td><td>1 Selection Columns: KUN</td><td></td></tr<>	Sampling Rate/Frequency (Hz):	1 Selection Columns: KUN	
rank Control I a 1 a 1 a 1 a 1 a 1 b <td>Select Data</td> <td>2</td> <td>Number of Classes: 0</td>	Select Data	2	Number of Classes: 0
Seet Al Seet Data Length of Cell Array: Shown Cell in Table 1 2 1 2 3 - 4 - 1 2 3 - 4 - - - - - - - 1 2 3 - -<	Files: Columns: Columns:	Maximum Values	Number of Elements.
Length of Cell Army: Shown Cell in Table Length of Cell Army: Shown Cell in Table Length of Cell Army: Shown Cell in Table Length of Cell Army:	Select All Select Data		ADD Clear
Lengh of Cel Array: Shown Cel In Table 1 2 1 2 3 - 4 - 1 2 3 - 4 - 1 2 2 - 4 - 1 2 2 - 4 - 1 2 2 - 1 2 2 - 1 2 2 - 1 2 3 - 1 2 3 - 1 2 2 - 58rdert Deviation - 1 2 2 - 1 2 2 - 1 2 - - - - - - - - - - - </td <td></td> <td></td> <td>Selection of Test Set</td>			Selection of Test Set
1 2 2 1 2 3 1 2 4 1 2 - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -	Length of Cell Array: Shown Cell in Table:		
1 2 1 2 1 1 2 1 1 2 1	1 2	Mean Values 4	Selection of
1 2	1		Discriminant Function
4	3		Linear
Image: Standard Deviation Image: Standard Deviation Image: Standard Deviation Image: Standard Deviation Standard Deviation Image: Standard Deviation Image: Stan	4		RUN
1 2		Integral Values	Classification Result
No Rectification Image: Constraint of the co			Classes
Standard Deviation			1
Image: Standard Dervision I			2
re-processing Rectification No Rectification No Rectification		Standart Deviation	4
Rectination Terring B No Rectination In The Rectination Full Wave Information High Cutoff Frequency (Hz): Information Normalization Filer Order: 0 -1 Information -1-1 An Borris			
No Rectification Full Wave Haft Wave High Cutoff Frequency (ritz) 1 2 2 1 2 1 2 2 1 2 2 1 2 2 1 2 2 1 2 2 1 2 2 1 2 2 1 2 1 2 2 1 2 1 2 1 2 2 1 2 2 1 2 1 2 2 1 2 2 1 2 2 1 2 2 1 2 2 1 2 2 1 2 2 1 2 2 1 2 2 1 2 2 2 2 1 2	Rectification Filtering		
Of Wave Low Cutoff Frequency (Vt2) Image: Construction of the constru	No Rectification No Filter		
High Cutoff Frequency (Hz); 1 2 Normalization File: Order; 1 2 0 - 1 0 1 2 1 - 1 0 All 1 2	C Full Wave Low Cutoff Frequency (Hz):	Entropy Values	
Normalization Filer Order: 1 1 9 No Instruction Graphs Crain 1 2 -1-1 A Brows I 2 I	Half Wave High Cutoff Frequency (Hz):		
No Normalization O -1 O -1 O AB Roys I I Cels I I Cels Cels I Cels Cel	Normalization Filter Order:	A A A A A A A A A A A A A A A A A A A	
O -1 O Ho Graphs Cefs O -1-1 O A8 Rows I Cefs Cefs I Cefs Cef	No Normalization Graphs Graph Selection		
	0 - 1 No Graphs Cells		· · · · · · · · · · · · · · · · · · ·
		1 4	Estimated Success

Fig. 4 – The main GUI of the designed biomedical signal analysis tool which provides the big picture of the functionality of the tool. The zoomed-in versions of different parts are shown in the subsequent figures.

Nonlinear analysis of biomedical signals provides understanding of nonlinear and dynamic behaviors of the system that the signals are obtained from, e.g. whether the signals are chaotic or not [14]. In the nonlinear chaos and complexity analysis module of the developed tool, entropy and largest Lyapunov exponents (LLE) methods are implemented in order to determine the nonlinear behavior of the signals [33–36]. LLE is similar to entropy, which is a measure used to determine the irregularity of the time-series [37]. The greater the entropy value of a time series, the greater the irregularity of the signal. Likewise, the LLE shows that the analyzed signal is chaotic or not. The more the LLE value is greater than zero, the more chaotic the signal or the time-series; if the LLE is zero, the signal is not chaotic [33,38].

There exist some programs which implement LLE calculation algorithms. One of them is the Chaos Data Analyzer program, developed by JC Sprott [38,39]. This program is a tool for research and teaching which contains LLE calculation, among other tests for detecting hidden determinism and the underlying behavior in time-series data. In this work, we have utilized the LLE calculation program developed by Mohammadi [40] based on the LLE calculation algorithm proposed by Rosenstein et al. [41]. This implementation allows both manual and automatic initialization of embedding lag and embedding dimension parameters. We have used automatic selection of these parameters in our GUI. For the automatic initialization of embedding lag, autocorrelation and minimum mutual information is used. For the automatic initialization of the embedding dimension, false nearest neighbors and symplectic geometry methods are used.

Pattern recognition and classification involves identifying to which of a set of categories a new observation belongs to, on the basis of a training set of data. Determining or identifying categories are usually done by using the extracted data features from the signals, and feature creation and selection is thus crucial. For this reason, preprocessing and extraction of features which are performed before classification are very important steps for a good classification [15]. There are many techniques for classification. In the developed tool, mathematical functions which are called as linear discrimination functions are implemented in the classification module for separating the biomedical signals into different categories. This basic classification method is defined as discrimination analysis. The mean, std, min, max, integral value and a combination of these can be selected as features into the classification module of the tool, classification into multiple categories can be performed by the module.

rick a file / fi	les		Spin'renorm	23
Konum	EEG eye	•	← 🗈 💣 💷 •	
(Fig	Ad	^	Değiştirme tarihi	Tür
Son Yeder	UCI Mach	nine Learning Repository EEG Eye S	12.11.2013 16:21	Dosya kla
son tener	EEG Eye S	tate.arff	12.11.2013 16:19	ARFF Dos
	UCI Mach	ine Learning Repository EEG Eye S	12.11.2013 16:21	Firefox H
Masaüstü				
Kitapliklar				
Bigisayar				
Bilgisayar Q	٠ [m		,
Bilgisəyər Qə Ağ	∢ Dosya Adı:	III EEG Eye State		Aç

Fig. 5 - Selection of a data file.

COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE XXX (2015) XXX-XXX

	Im	port Data From	File	
File Pati Number	h:C:\Users\SEK of Files: 2	A\Desktop\biyo	med\EEG eye\	
Sampin	g Rate/Frequency	/ (HZ): 120		
Select	Data			
iles:	1 Row	/s: 1:200	Columns:	1:4
_				_
E	Select All	Se	elect Data	
			F	
Length	of Cell Array: 1	Shown	Cell in Table:	1
				- E
	1	2	3	4
191	1 4.4323e+03	2 3.9985e+03	3 4.2815e+03	4 4.1154
191 192	1 4.4323e+03 4.4272e+03	2 3.9985e+03 3.9903e+03	3 4.2815e+03 4.2810e+03	4 4.1154 4.1072
191 192 193	1 4.4323e+03 4.4272e+03 4.4267e+03	2 3.9985e+03 3.9903e+03 3.9877e+03	3 4.2815e+03 4.2810e+03 4.2851e+03	4 4.1154 4.1072 4.1082
191 192 193 194	1 4.4323e+03 4.4272e+03 4.4267e+03 4.4277e+03	2 3.9985e+03 3.9903e+03 3.9877e+03 3.9882e+03	3 4.2815e+03 4.2810e+03 4.2851e+03 4.2846e+03	4 4.1154 4.1072 4.1082 4.1144
191 192 193 194 195	1 4.4323e+03 4.4272e+03 4.4267e+03 4.4277e+03 4.4195e+03	2 3.9985e+03 3.9903e+03 3.9877e+03 3.9882e+03 3.9882e+03 3.9831e+03	3 4.2815e+03 4.2810e+03 4.2851e+03 4.2846e+03 4.2805e+03	4 4.1154 4.1072 4.1082 4.1144 4.1118
191 192 193 194 195 196	1 4.4323e+03 4.4272e+03 4.4267e+03 4.4277e+03 4.4195e+03 4.4128e+03	2 3.9985e+03 3.9903e+03 3.9877e+03 3.9882e+03 3.9831e+03 3.9821e+03	3 4.2815e+03 4.2810e+03 4.2851e+03 4.2846e+03 4.2805e+03 4.2790e+03	4 4.1154 4.1072 4.1082 4.1144 4.1118 4.1082
191 192 193 194 195 196 197	1 4.4323e+03 4.4272e+03 4.4267e+03 4.4277e+03 4.4195e+03 4.4128e+03 4.4113e+03	2 3.9985e+03 3.9903e+03 3.9877e+03 3.9882e+03 3.9831e+03 3.9821e+03 3.9774e+03	3 4.2815e+03 4.2810e+03 4.2851e+03 4.2846e+03 4.2805e+03 4.2790e+03 4.2795e+03	4 4.1154 4.1072 4.1082 4.1144 4.1118 4.1082 4.1097
191 192 193 194 195 196 197 198	1 4.4323e+03 4.4272e+03 4.4267e+03 4.4277e+03 4.4195e+03 4.4195e+03 4.4128e+03 4.4113e+03 4.4036e+03	2 3.9985e+03 3.9903e+03 3.9877e+03 3.9882e+03 3.9831e+03 3.9821e+03 3.9774e+03 3.9774e+03 3.9646e+03	3 4.2815e+03 4.2810e+03 4.2851e+03 4.2846e+03 4.2805e+03 4.2790e+03 4.2795e+03 4.2795e+03 4.2790e+03	4 4.1154 4.1072 4.1082 4.1144 4.1118 4.1082 4.1097 4.1067
191 192 193 194 195 196 197 198 199	1 4.4323e+03 4.4272e+03 4.4267e+03 4.4277e+03 4.4195e+03 4.4195e+03 4.4128e+03 4.4113e+03 4.4036e+03 4.3959e+03	2 3.9985e+03 3.9903e+03 3.9877e+03 3.9882e+03 3.9831e+03 3.9821e+03 3.9774e+03 3.9646e+03 3.9651e+03	3 4.2815e+03 4.2810e+03 4.2851e+03 4.285e+03 4.2805e+03 4.2790e+03 4.2795e+03 4.2790e+03 4.2790e+03 4.2744e+03	4 4.1154 4.1072 4.1082 4.1144 4.1118 4.1082 4.1097 4.1067 4.1067

Fig. 6 – Importing a dataset and data selection in "Dataset" part.

3. Presentation of the designed tool

The purpose of this section is to present the user-friendly and simple GUI tool which was designed based on MATLAB for analysis of biomedical signals [42]. Here, the modules, their functionality and usage are described. The flow chart in Fig. 3 simply summarizes the usage of the designed GUI tool.

The designed tool consists of four main parts and it has been controlled with five buttons (Fig. 4). The first part is "Dataset" part and the data import and selection process is performed in this part of the tool.

At the start of application, the first step is to import the dataset from a data file. When the user clicks the "Import Data From File" button, the browsing window in Fig. 5 appears. The user can browse the files and select single or multiple



Fig. 7 – "Preprocessing" section and selection of the processes.



Fig. 8 - The graph of preprocessed data.

dataset files (such as .txt, .mat, etc.) using this window. The datasets must be in two dimensional row-and-column format where different columns should correspond to different signal source or attribute (such as an EEG channel or contact) and different rows should correspond to different instances/samples or time-points. The first line of the file can include header information and the program can recognize whether there is a header line at the beginning of the file.

Mini	mum Values	Mean Value	s 🔽 Integ	ral Values	Standart Deviatio
/ Max	imum Values	Entropy Val	ies 🔽 I van	unov Exponents	RUN
- Minim	um Values		in clab	unor exponento	
	1	2	3	4	
1	4.2815e+03	3.9646e+03	4.2292e+03	4.1010e+03	
Maxin	num Values				
[]	1	2	3	4	
1	4.4939e+03	4.1569e+03	4.3144e+03	4.1856e+03	
- Mean	Values				
	1	2	3	4	
1	4.3430e+03	4.0226e+03	4.2688e+03	4.1321e+03	
Integr	al Values				
	1	2	3	4	
1	8.6423e+05	8.0053e+05	8.4949e+05	8.2230e+05	
Stand	lart Deviation —				
Stand	lart Deviation —	2	3	4	
- Stand	lart Deviation — 1 63.8562	2 43.1462	3 19.0489	4 17.7191	
- Stand	lart Deviation 1 63.8562 py Values	2 43.1462	3 19.0489	4 17.7191	
- Stand	lart Deviation	2 43.1462	3 19.0489	4 17.7191	
- Stand	lart Deviation 1 63.8562 py Values 1 6.6047	2 43.1462 2 6.3324	3 19.0489 3 6.4562	4 17.7191 4 6.1989	
- Stand	tart Deviation1 63.8562 py Values1 6.6047 unov Exponents	2 43.1462 2 6.3324	3 19.0489 3 6.4562	4 17.7191 4 6.1989	
- Stand	Iart Deviation 1 63.8562 py Values 1 6.6047 unov Exponents 1	2 43.1462 2 6.3324 2	3 19.0489 3 6.4562 3	4 17.7191 4 6.1989 4	

Fig. 9 - Calculated signal features.

-

7

ANTICLE IN PRESS

COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE XXX (2015) XXX-XXX

Primary Analysis	- Primary An	alysis	Graphs	- Graph Select	tion	Cell in Tabl
lo Analysis 🛛 🗲	ICA	-	No Graphs	Cells:		1
No Analysis			O AI			
CA P	Secondary	Analysis-		Rows.		0.111
PCA	No Analysis	s 🔻	Selection	Columns;		RUN
Secondary Analysis						
lo Analysis	ICA					
No Analysis		1	2	3	4	
Fourier	1	51.5986	-70.7045	-299.6744	26.3316	-
Correlation	2	51.6005	-70.5633	-299.9417	26.7494	ſ
	3	51.5956	-70.6302	-300.6073	25.8074	
	4	51.7500	-70.7062	-300.3779	26.2881	1
	5	51.7892	-70.7068	-299.8877	26.5345	
	6	51.6174	-70.6359	-300.2345	24.8994	
	7	51.5098	-70.5946	-300.1799	24.4996	
	8	51.5268	-70.7129	-299.1486	25.6614	
	9	51.6383	-70.7849	-298.6320	26.1583	
	10	51.6580	-70.7972	-298.8565	25.8703	
	11	51.6083	-70.8275	-298.9205	25.1459	
	12	51.4763	-70.7230	-298.3706	24.3092	
	13	51.3102	-70.5795	-297.5003	24.2119	
	14	51.5170	-70.6171	-297.1613	25.3661	
	15	51.7196	-70.7126	-297.4660	26.0363	
	16	51.5698	-70.7193	-297.5113	25.2419	
	17	51.5049	-70.6372	-297.1175	24.7837	
	18	51.5507	-70.5725	-297.1443	25.1462	
	19	51.4912	-70.6406	-297.3113	25.6227	
	20	51.5303	-70.7033	-297.0615	26.1448	
	21	51.6560	-70.7102	-297.0977	26.4351	
	22	51.6492	-70.8936	-297.7122	25.6622	
	23	51.8736	-71.1157	-298.0521	25.3460	
	24	51.8428	-71.0282	-297.9306	25.3794	
	25	51.5334	-70.8185	-297.8613	25.0600	
	26	51.6246	-70.7987	-297.4804	25.4075	
	27	51.7078	-70.7947	-297.0621	25.2385	
	28	51.5045	-70.6973	-296.9250	24.7179	
	29	51.6466	-70.6639	-297.0349	25,7151	

Fig. 10 - "Analysis" section of the designed tool and ICA results.

After file selection process, the dataset is imported and the file path (or paths if multiple files are selected), and the row numbers are shown as in Fig. 6. The tool saves automatically the imported dataset to "dataset.mat" file in cell array form in the MATLAB working directory. Every selected file is kept as a cell array in memory. The sampling rate or frequency has to be entered to the appropriate edit boxes for processing and analysis.

Users may want to study with whole dataset or a part of the dataset. So, there is a part for data selection from the dataset as seen in Fig. 6. Users can select the whole dataset by selecting the "Select All" check box or they can enter the desired file numbers, row and column number ranges in the "Files", "Rows" and "Columns" boxes, respectively. "Select Data" button should be clicked for finishing the selection process. The selected rows and columns of the first cell (file) of the selected data are updated and shown in the table in "Dataset" part. The cell array corresponding to the selected file is shown at the top of the table in the "Shown Cell in Table" box. If the user wants to change the cell corresponding to any desired file, the desired file number should be entered in the edit box "Shown Cell in Table". In Fig. 6, one cell (file) has been selected from two cells, first 200 rows and 4 columns of the cell have been selected, and rows 191 through 200 and columns 1 through 4 are shown.

The second part in the designed tool is "Preprocessing" module which consists of rectification, normalization, filtering and graph selection functionalities (Fig. 7). All functionalities are shown in Fig. 7. The snapshot at bottom right of Fig. 7 shows several options of filtering functionality. Every functionality has an option for "No" as in Fig. 7. If users do not want to preprocess data with a particular functionality, they can select the "No" option (the default option).

In the developed tool, a very convenient feature for helping the users also exists. If the user waits the cursor over a task/module/functionality of the tool for more than a second, a brief pop-up explanation appears about that task/module/functionality as seen in the snapshot at top right of Fig. 7.

For the selected data in Fig. 6, a preprocessing has been performed in Fig. 7. The data has been rectified to a full wave (i.e. absolute value is taken), normalized between 0 and 1, and filtered with a second-order band-pass Butterworth filter between 1 Hz and 60 Hz frequencies. Also, graph selection is entered for plotting rows 1–200 and columns 1–2 of the first cell of the preprocessed data. If "All" option is selected or multiple cells are entered to graph selection, the data in every separated cell is shown in a separated figure and graph. The cell numbers are written in the title at top of the graphs to distinguish between data files conveniently.

COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE XXX (2015) XXX-XXX



Fig. 11 – Fourier transform results for PCA components; top: the array, bottom: the plot.

After all necessary entries, the "RUN" button needs to be clicked and all selected preprocessing steps will be performed. When the preprocessing is finished, the selected graphs are created and the preprocessing results are saved automatically to "dataprepro.mat" file in the working directory. The plot of the preprocessing in Fig. 7 can be seen in Fig. 8.

The third part of the tool is "Signal Features" part, which include the statistics, the time-domain, and the nonlinear methods described in the previous section. Basic statistical parameters such as minimum, maximum, mean and standard deviation values; integral values, entropy values and largest Lyapunov exponents are included in this part of the tool. After

Primary Anal	ysis (Graphs-	- Graph Select	tion	- Cell in Tabl	
PCA Secondary Analysis		No Graphs			1	
		© All	Rows:		ſ	
No Analysis		Selection	Columns:		RUN	
CA-	[1			
	1	2	3	4		
1	-11.3805	1.2446	29.9571	3.5697	-	
2	-16.5307	-0.0400	35.8012	2.2351		
3	-11.3568	2.6059	39.9129	7.1776	-	
4	-7.7986	5.6857	39.0950	4.7732		
5	-11.4926	5.6858	33.8129	2.9806		
6	-20.5780	4.3766	30.5889	10.3682		
7	-24.6235	2.1923	27.9868	11.9842		
8	-18.5888	0.2111	19.7245	5.5740		
9	-17.0973	1.9151	15.1815	2.6769		
10	-16.3024	3.2030	16.7398	4.2114		
11	-17.1776	3.5419	14.0085	7.3403		
12	-32.7357	0.6409	2.8634	10.0154		
13	-48.7891	-4.9506	-8.8749	9.1371		
14	-45.7440	-2.1518	-7.7932	3.7432		
15	-35.4886	2.4234	-0.6564	1.3552		
16	-36.8326	0.3229	-3.9572	4.7835		
17	-48.1596	-1.3377	-11.2319	6.1015		
18	-51.1643	-1.2490	-8.9793	4.6146		
19	-40.5347	-2.9210	-4.6249	2.9197		
20	-35.7264	-2.7774	-5.5223	0.3557		
21	-35.3793	-0.1999	-3.6154	-0.8431		
22	-21.1165	2.7677	0.4004	3.3267		
23	-8.0075	10.4179	2.9836	5.0901		
24	-14.7122	8.9566	1.7076	4.7708		
25	-26.8482	0.7464	-0.4209	6.0976		
26	-31.3263	1.8014	-3.7101	4.0321		
27	-38.1317	3.6334	-10.0259	4.0582		
28	-46.0040	-1.0590	-14.2191	6.0829		
29	-43 7221	0.4938	-7 8338	2 0519		

Fig. 12 - PCA results.

preprocessing of the data, the desired features can be selected by the user and the selected features are computed and saved automatically to "features.mat" file by clicking the "RUN" button. For example, all features has been selected and computed in Fig. 9 for the example dataset being analyzed.

As seen in Fig. 10, the last main part is "Analysis" part with which users can perform ICA, PCA, Fourier Transform and correlation analysis. ICA and PCA are called the "Primary analysis"; Fourier transform and correlation are called "Secondary analysis" in this part of the tool. This separation allows users to carry out two analyses successively for a dataset. For example, Fourier transform can be done for PCA components of a dataset as in Fig. 11. Additionally, the graph of Fourier transform of third component is shown in Fig. 11.

The analysis options can be seen at top left of Fig. 10. After appropriate selections, users can click on "RUN" button to perform the selected analysis. In Fig. 10, independent components of the preprocessed data in Fig. 7 have also been shown. The results which are shown in table in the figure belong to the cell in the edit text box named "Cell in Table" (Fig. 10). User can write the number of the cells to the edit box that they want to see in the table.

Fig. 12 shows the PCA results of the preprocessed data in Fig. 7 by the module. Fig. 13 illustrates the graph selection for

COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE XXX (2015) XXX-XXX



Fig. 13 – Graph selection and comparing of two principle components.

comparing two components of PCA results. The graph selection in "Analysis" part is achieved by the same way of the graph selection in "Pre-processing" part.

Figs. 14 and 15 show Fourier transform and correlation results for the preprocessed data in Fig. 7. Also, in Fig. 14, maximum valued components in Fourier transform can be seen after performing the Fourier transform. After performing the analyses, the results are saved automatically to "analysis.mat" file into the working directory.

The last main part, "Classification", is shown in Fig. 16. Users can perform basic classification based on discriminant analysis for a selected data set in "Dataset" part. "Classification" part can classify the data which is selected and shown in the table in "Dataset" part as in Fig. 6. This means that users firstly import and select the data than they classify the data by using "Classification" part. Linear, diagonal linear, quadratic, diagonal quadratic and Mahalanobis discriminant functions of MATLAB are implemented for classification.

The classification process in the GUI needs four parameters for discriminant analysis: a training set, training category/class labels, a test set and a discriminant function. As in Fig. 16, category/class names of training set and training classes can be entered in the edit boxes at the top. Data belonging to a particular category can be selected with their indexes, which can be seen in the table in "Selected Data" part. For every class, "ADD" button is clicked after the category name and data entries are selected. Once the training set and classes are created, the number of entered classes/categories and corresponding data entries can be seen under the edit boxes. If there is a problem with the entries, "Clear" button can be used for a new training set entry. Another requirement is selection of a test set for classification. Test set is created by using the

la Anak		No Cranha			1
			Cells:		
Seconda	ry Analysis		Rows:	6	
ourier	•	Selection	Columns:		RUN
ourier T	ransform		_ I		
	1	2	3	4	
1	4.1945e-12	1.9736e-12	2.1828e-12	3.1618e-12	
2	76.7730	14.5846	8.5921	1.7574	
3	52.4793	12.0814	9.6013	1.0460	E
4	54.2318	17.1435	10.0035	2.4571	
5	39.1596	17.9441	4.0746	2.4963	
6	24.3441	14.9779	4.5542	1.6511	
7	15.1062	11.8672	4.9532	1.4763	
8	12.2342	7.4569	1.8979	1.8215	
9	13.2276	2.5936	5.4989	0.7705	
10	7.1978	2.0740	3.3357	0.7177	
11	6.7928	4.5180	2.4984	1.0962	
12	4.8075	3.4618	3.4916	0.3936	
13	1.7493	3.2958	2.1134	0.2353	
14	1.4312	2.8688	2.5919	0.9152	
15	3.1744	3.6389	0.4357	0.8920	
16	5.5799	2.0618	4.1734	1.6354	
17	3.4909	1.9545	4.4440	0.4137	
18	4.4541	3.4462	1.1825	0.6883	
19	2.6803	0.9282	1.9957	1.0440	
20	0.7026	2.8928	1.8322	0.1970	
21	3.9619	0.6810	2.2022	0.6714	
22	4 7520	2 6306	0.6104	1.0747	Ŧ
lax Valu	e and Frequency	(
	1	2	3	4	
1	76.7730	17.9441	10.0035	2.4963	
2	0.5000	2	1.5000	2	

Fig. 14 - Fourier transform results.

edit box which is labeled as "Selection of test set" as seen in Fig. 16. If the check box is selected, all data set except training elements will be used as test set. The last parameter is the discriminant function which is selected by using the drop down list in Fig. 16. There are five functions for discrimination of the classes in the list as seen on the left side of Fig. 16.

After all parameters are entered, the classification process is performed by clicking the "RUN" button. As seen in Fig. 16, the classes of the test set elements are shown in the table and the estimated success rate of the classification is shown in the







Fig. 16 - Classification section and results.

edit box at the bottom. All results and input parameters are saved to "classification.mat" file. Fig. 17 shows all files of the saved dataset and results after all processes in the designed tool.



Fig. 17 - All files of the saved dataset and results.

4. Conclusions

In this work, a MATLAB-based graphical user interface has been designed as a biomedical signal processing and analysis tool. The purpose was to develop a tool that the users can perform the most commonly used processing and analysis easily by using this uncomplicated, user-friendly, intuitive and flexible tool. By using the tool, the users can easily perform the following: importing and forming dataset in single file or multiple files, selecting methods, visualization dataset and results numerically and graphically, comparing the signals in dataset. Also, the tool saves the imported datasets, parameters and the results automatically. Additionally, the tool provides basically all the required steps (except data acquisition) for biomedical signal processing and analysis by implementing most commonly used methods, and the largest Lyapunov exponents method for measuring the nonlinearity and chaos in the signal. One scientist who is familiar with MATLAB and the processing/analysis methods was trained on the developed tool and it took him just 45 min to fully understand it and run the processing/analysis on then dataset shown above.

The tool can also be utilized as an educational tool in biomedical engineering field. All of the properties mentioned above shows that this, easy-to-use, easy-to-learn, intuitive and flexible yet comprehensive tool makes an important contribution to the literature of biomedical field, when compared with the other similar tools.

Addition of different methods, features and functionalities to all parts/modules is planned for the future version of the tool while keeping the user-friendly and uncomplicated form. Also we are currently planning on implementing our graphical user interface software on other scientific computing software platforms such as Octave and R, as part of our future work. The tool will be tested on various datasets. The tool will then be shared in a website prepared by the authors for greater use and evaluation by the biomedical research community.

REFERENCES

- D. Cuesta, P.M. Aboy, D. Novak, R. Brezny, L. Samblas, D. Pastor, S. Sancho, Biosignal Laboratory: a software tool for biomedical signal processing and analysis, in: Proceedings of the 25th Annual International Conference of the IEEE EMBS, Cancun, Mexico, 2003, pp. 3544–3547.
- [2] J.F. Guerrero, M. Bataller, E. Soria, R. Magdalena, BioLab: an educational tool for signal processing training in biomedical engineering, IEEE Trans. Educ. 50 (1) (2007) 34–40.
- [3] M. Lascu, D. Lascu, Graphical programming based biomedical signal acquisition and processing, Int. J. Circuits Syst. Signal Process. 1 (4) (2007) 317–326.
- W.J. Tompkins, J.A. Wilson, Software for biomedical engineering signal processing laboratory experiments, in: 31st Annual International Conference of the IEEE EMBS: 2008–2010, Minneapolis, MN, USA, 2009.
- [5] M. Potse, A.C. Linnenbank, C.A. Grimbergen, Software design for analysis of multichannel intracardial and body surface electrocardiograms, Comput. Methods Programs Biomed. 69 (2002) 225–236.
- [6] J. García, I. Martínez, L. Sörnmo, S. Olmos, A. Mur, P. Laguna, Remote processing server for ECG-based clinical diagnosis

aper.me paper

11

ARTICLE IN PRESS

support, IEEE Trans. Inf. Technol. Biomed. 6 (4) (2002) 277–284.

- [7] J.P. Niskanen, M.P. Tarvainen, P.O. Ranta-aho, P.A. Karjalainen, Software for advanced HRV analysis, Comput. Methods Programs Biomed. 76 (2004) 73–81.
- [8] R. Gupta, J.N. Bera, M. Mitra, Development of an embedded system and MATLAB-based GUI for online acquisition and analysis of ECG signal, Measurement 43 (2010) 1119–1126.
- [9] P. Perakakis, M. Joffily, M. Taylor, P. Guerra, J. Vila, KARDIA: a Matlab software for the analysis of cardiac interbeat intervals, Comput. Methods Programs Biomed. 98 (2010) 83–89.
- [10] E. Elcano, O. Miranda, I. Rodriguez, J. Navallas, L. Piquer, A. Malanda, L. Gila, Interactive software tool for needle EMG signal analysis, in: IEEE 46th Midwest Symposium on Circuits and Systems, vol. 1, 2003, pp. 361–363.
- [11] A. Delorme, T. Mullen, C. Kothe, Z.A. Acar, N. Bigdely-Shamlo, A. Vankov, S. Makeig, EEGLAB, SIFT, NFT, BCILAB, and ERICA: new tools for advanced EEG processing, Comput. Intell. Neurosci. (2011), http://dx.doi.org/10.1155/2011/130714.
- [12] A. Delorme, S. Makeig, EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis, J. Neurosci. Methods 134 (2004) 9–21.
- [13] P.E. Aguera, K. Jerbi, A. Caclin, O. Bertrand, ELAN: a software package for analysis and visualization of MEG, EEG, and LFP signals, Comput. Intell. Neurosci. (2011), http://dx.doi.org/10.1155/2011/158970.
- [14] S. Cerutti, C. Marchesi, Advanced Methods of Biomedical Signal Processing, John Wiley & Sons, Inc., Hoboken, NJ, 2011.
- [15] R.M. Rangayyan, Biomedical Signal Analysis: A Case Study Approach, Wiley-IEEE Press, New York, 2001.
- [16] Ü. Sakoğlu, V.D. Calhoun, Temporal dynamics of functional network connectivity at rest: a comparison of schizophrenia patients and healthy controls, in: Proceedings of the 15th Annual Meeting of the Organization for Human Brain Mapping, June 2009, San Francisco, CA, 2009.
- [17] Ü. Sakoğlu, A.M. Michael, V.D. Calhoun, Classification of schizophrenia patients vs. healthy controls with dynamic functional network connectivity, in: Proceedings of the 15th Annual Meeting of the Organization for Human Brain Mapping, June 2009, San Francisco, CA, 2009.
- [18] Ü. Sakoğlu, V.D. Calhoun, Functional network connectivity with temporal derivatives of sICA time-courses in schizophrenia patients vs healthy controls, in: Proceedings of the 17th Annual Meeting of the International Society for Magnetic Resonance in Medicine, April 2009, Honolulu, HI, 2009.
- [19] Ü. Sakoğlu, V.D. Calhoun, Dynamic windowing reveals task-modulation of functional connectivity in schizophrenia patients vs healthy controls, in: Proceedings of the 17th Annual Meeting of the International Society for Magnetic Resonance in Medicine, April 2009, Honolulu, HI, 2009.
- [20] U. Sakoglu, G.D. Pearlson, K.A. Kiehl, Y.M. Wang, A.M. Michael, V.D. Calhoun, A method for evaluating dynamic functional network connectivity and task-modulation: application to schizophrenia, MAGMA 23 (2010) 351–366.
- [21] A.M. Michael, M.D. King, S. Ehrlich, G.D. Pearlson, T. White, D.J. Holt, N. Andreasen, Ü. Sakoğlu, B.-C. Ho, S.C. Schulz, V.D. Calhoun, A data-driven investigation of gray matter-function correlations in schizophrenia during a working memory task, Front. Hum. Neurosci. 5 (71) (2011) 1–13.
- [22] Ü. Sakoğlu, K. Bohra, Effect of task/condition-related signal variance on functional connectivity: an fMRI simulation

study, in: Proceedings of the 21st Annual Meeting of the International Society for Magnetic Resonance in Medicine (ISMRM), April 2013, Salt Lake City, UT, 2013.

- [23] D. Akgun, U. Sakoglu, M. Mete, J. Esquivel, B. Adinoff, GPU-accelerated dynamic functional connectivity analysis for functional MRI data using OpenCL, in: Proceedings of the IEEE International Conference on Electro/Information Technology, 5–7 June 2014, Milwaukee, WI, 2014, pp. 479–484.
- [24] D. Akgun, U. Sakoglu, J. Esquivel, B. Adinoff, M. Mete, GPU accelerated dynamic functional connectivity analysis for functional MRI data, Comput. Med. Imaging Graph. 43 (2015) 53–63.
- [25] R.E. Challis, R.I. Kitney, Biomedical signal processing (in four parts). Part 1: time-domain methods, Med. Biol. Eng. Comput. 28 (1990) 509–524.
- [26] R.E. Challis, R.I. Kitney, Biomedical signal processing (in four parts). Part 2: the frequency transforms and their inter-relationships, Med. Biol. Eng. Comput. 29 (1990) 1–17.
- [27] R.E. Challis, R.I. Kitney, Biomedical signal processing (in four parts). Part 3: the power spectrum and coherence function, Med. Biol. Eng. Comput. 29 (1990) 225–241.
- [28] K.J. Blinowska, J. Zygierewicz, Practical Biomedical Signal Analysis Using MATLAB, CRC Press, Taylor & Francis Group, LLC, Florida, 2012.
- [29] J.L. Semmlow, Biosignal and Biomedical Image Processing MATLAB-Based Applications, Marcel Dekker, Inc., New York, 2004.
- [30] F.J. Theis, A. Meyer-Base, Biomedical Signal Analysis: Contemporary Methods and Applications, The MIT Press, Massachusetts, 2010.
- [31] R.W. Cox, AFNI: software for analysis and visualization of functional magnetic resonance neuroimages, Comput. Biomed. Res. 29 (3) (1996) 162–173.
- [32] T.J. Huppert, S.G. Diamond, M.A. Franceschini, D.A. Boas, HomER: a review of time-series analysis methods for near-infrared spectroscopy of the brain, Appl. Opt. 48 (10) (2009) D280–D298.
- [33] Statistical Parametric Mapping Toolbox. http://www.fil.ion.ucl.ac.uk/spm (accessed 12.09.15).
- [34] G. Benettin, L. Galgani, A. Giorgilli, J.M. Strelcyn, Lyapunov characteristic exponents for smooth dynamical systems and for Hamiltonian systems; a method for computing all of them. Part 1: theory, Meccanica 15 (1) (1980) 9–20.
- [35] G. Benettin, L. Galgani, A. Giorgilli, J.M. Strelcyn, Lyapunov characteristic exponents for smooth dynamical systems and for Hamiltonian systems; a method for computing all of them. Part 2: numerical application, Meccanica 15 (1) (1980) 21–30.
- [36] P. Bryant, R. Brown, H. Abarbanel, Lyapunov exponents from observed time series, Phys. Rev. Lett. 65 (13) (1990) 1523–1526.
- [37] T. Cover, J.A. Thomas, Elements of Information Theory, 2nd edition, Wiley, Hoboken, NJ, 2006.
- [38] J.C. Sprott, Chaos Data Analyzer. http://sprott.physics.wisc.edu/cda.htm (accessed 12.09.15).
- [39] J.C. Sprott, Chaos and Time-Series Analysis, Oxford University Press, Oxford, UK, 2003.
- [40] S. Mohammadi, Lyaprosen: MATLAB function to calculate Lyapunov exponent, University of Tehran, 2009.
- [41] M.T. Rosenstein, J.J. Collins, C.J. De Luca, A practical method for calculating largest Lyapunov exponents from small data sets, Physica D: Nonlinear Phenom. 65 (1993) 117–134.
- [42] MATLAB Creating Graphical User Interfaces, The MathWorks, Inc., Massachusetts, 2013, http://www. mathworks.com/help/pdf.doc/matlab/buildgui.pdf (accessed 9.12.13).