

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, user-friendly, 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

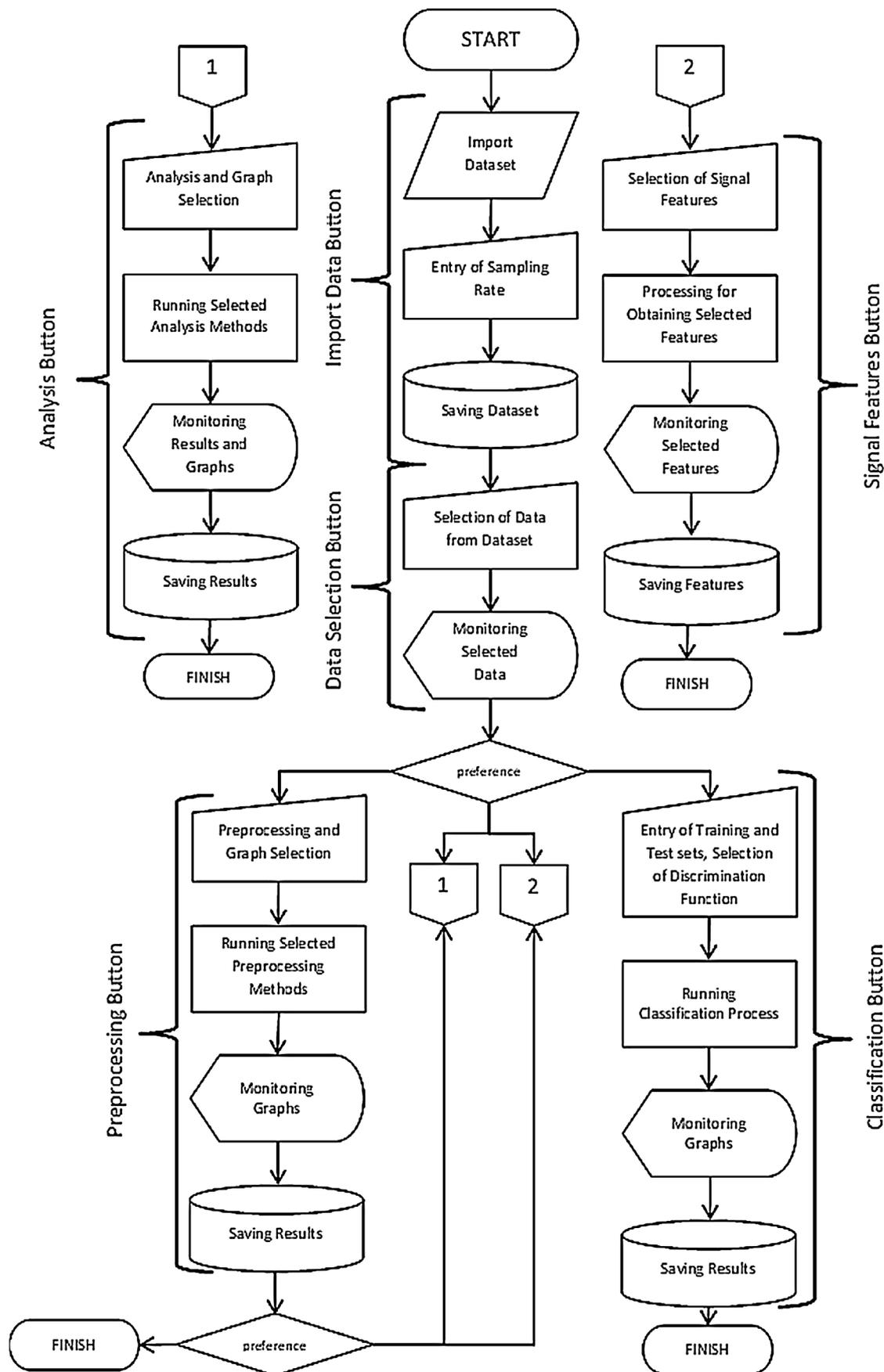


Fig. 3 – The flowchart of the designed tool.

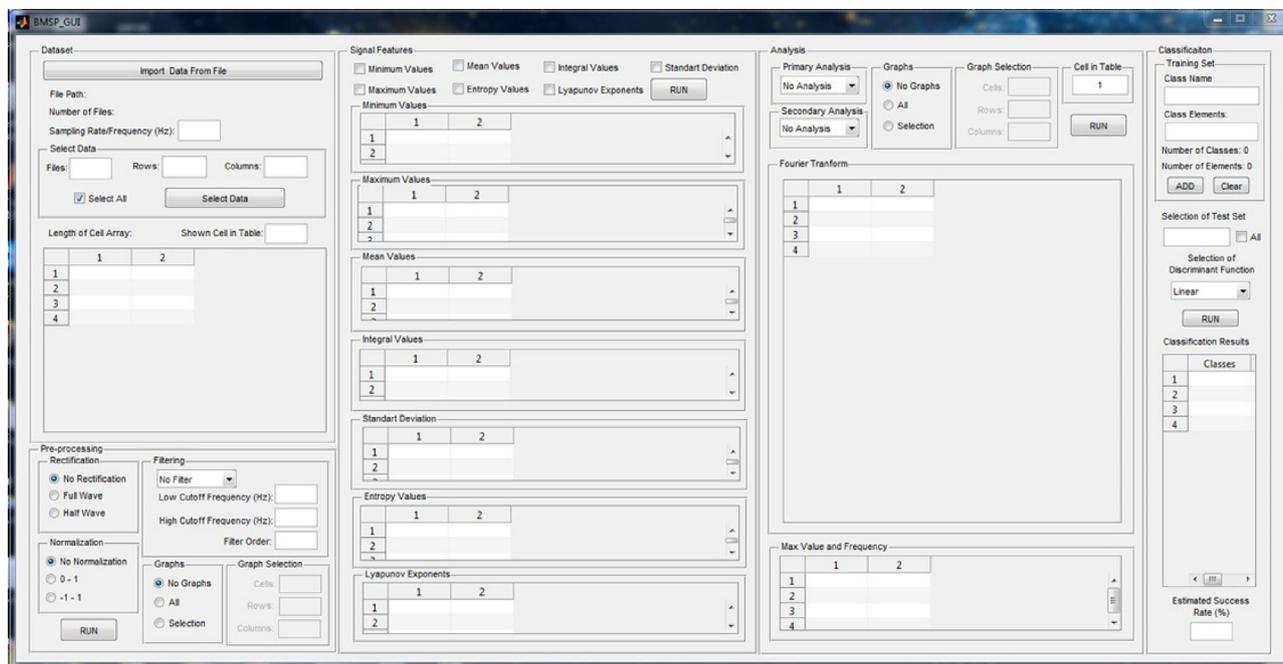


Fig. 4 – The main GUI of the designed biomedical signal analysis tool which provides the big picture of the functionality of the tool. The zoomed-in versions of different parts are shown in the subsequent figures.

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.

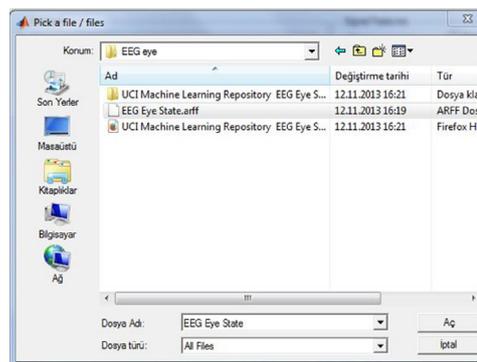


Fig. 5 – Selection of a data file.

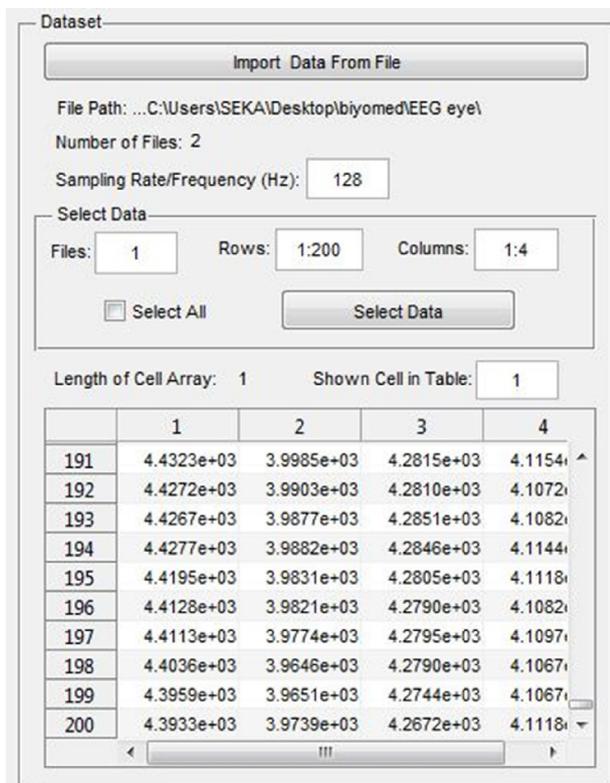


Fig. 6 – Importing a dataset and data selection in “Dataset” part.

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

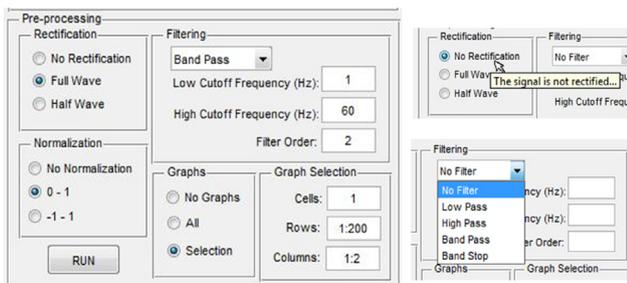


Fig. 7 – “Preprocessing” section and selection of the processes.

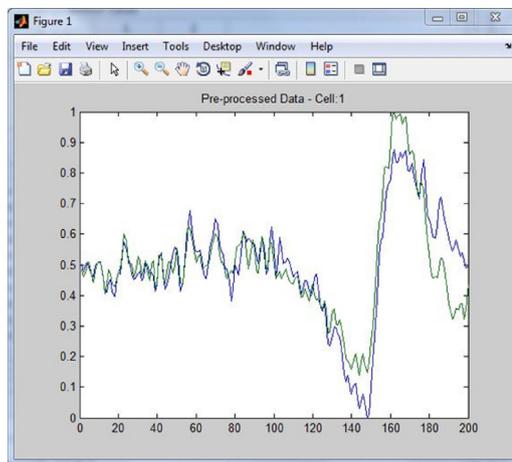


Fig. 8 – The graph of preprocessed data.

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.

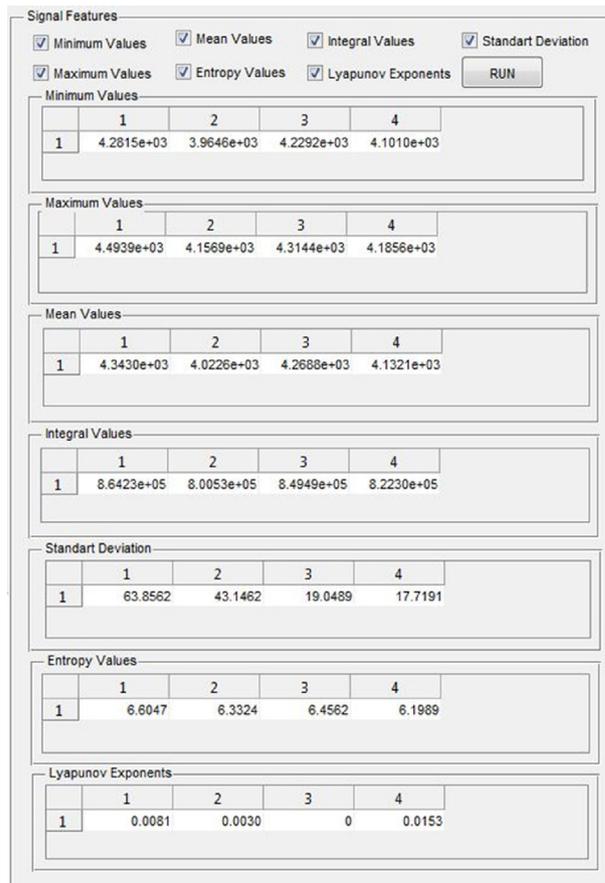


Fig. 9 – Calculated signal features.

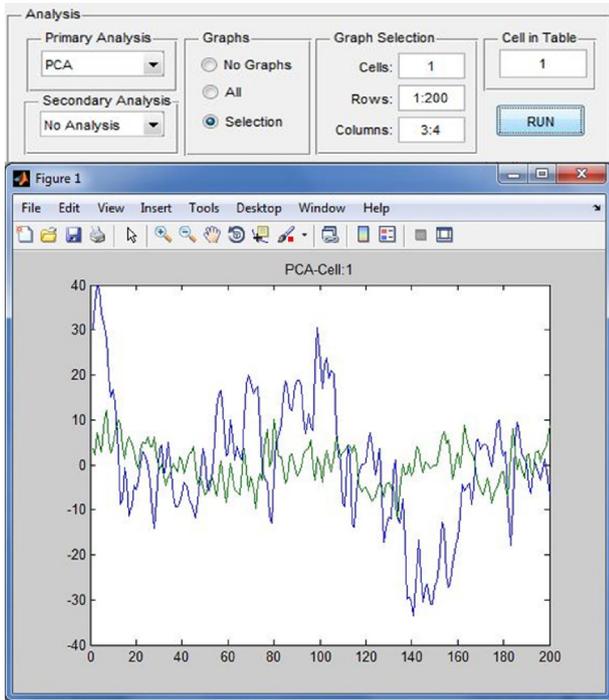


Fig. 13 – Graph selection and comparing of two principle components.

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

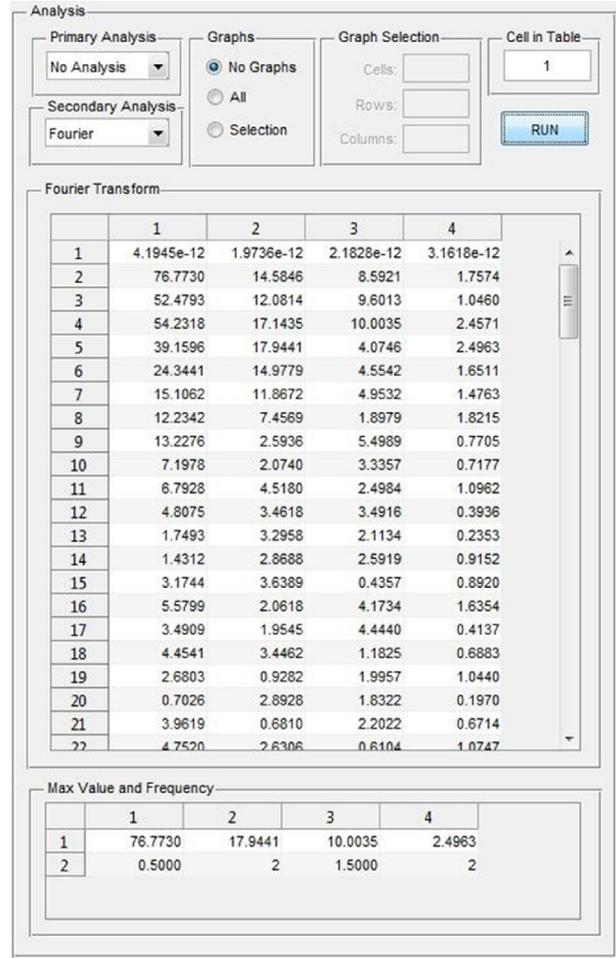


Fig. 14 – Fourier transform results.

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

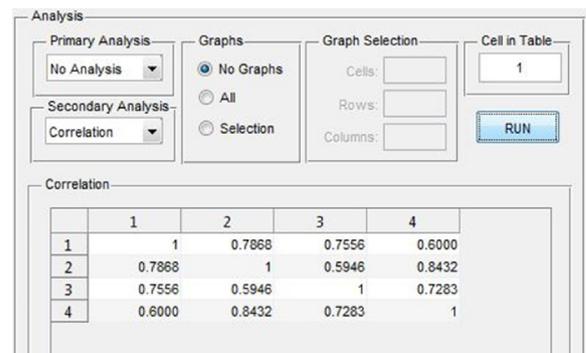


Fig. 15 – Correlation results.

- 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, BCLAB, 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. Zygierevicz, *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).