HAL open science

An OpenViBE Python-based framework for the efficient handling of MI BCI protocols

Arthur Desbois, Tristan Venot, Marie-Constance Corsi, Fabrizio de Vico Fallani

To cite this version:

Arthur Desbois, Tristan Venot, Marie-Constance Corsi, Fabrizio de Vico Fallani. An OpenViBE Python-based framework for the efficient handling of MI BCI protocols. Journées CORTICO 2022, Mar 2022, Autrans en Vercors, France. hal-03740275

HAL Id: hal-03740275

https://inria.hal.science/hal-03740275v1

Submitted on 29 Jul 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An OpenViBE Python-based framework for the efficient handling of MI BCI protocols

Authors & contributors

A. Desbois, T. Venot, M.-C. Corsi, F. De Vico Fallani

Sorbonne Université, Institut du Cerveau - Paris Brain Institute - ICM, CNRS, Inria, Inserm, AP-HP, Hôpital de la Pitié Salpêtrière, F-75013, Paris, France

Abstract

A typical Motor Imagery (MI) experimental pipeline is composed of an EEG data acquisition phase, followed by an analysis phase to train a classification algorithm (such as LDA). The training uses features extracted from the acquired data such as spectral power for a subset of sensors and frequencies, in which desynchronization can be observed between the MI tasks. In the final phase of the experiment, the trained classifier is used in an online setup.

The feature selection phase is crucial for the optimal functioning of a BCI system, but the large number of parameters can make it a long step of trial-and-error, not acceptable in clinical settings. Training a classification algorithm can be challenging and time consuming as it may include multiple manipulations and datatype conversions with external softwares.

Here, we propose a new Python-based framework to manage the whole experimental pipeline smoothly, integrating seamlessly with OpenViBE. We focused our work on feature analysis, selection, and classifier training. An easy-to-use GUI allows to keep track of the multiple acquired EEG signal files, and to process them for analysis and training. Convenient tools allow to compute spectral features and visualize them in the form of statistical R^2 maps, PSD, scalp topography and ERD/EDS time-frequency maps, for selected sensors or frequency bands, combining trials across multiple runs. Finally, a set of runs can be chosen for training the classification algorithm with only a few clicks and seconds of processing. All signal processing operations use OpenViBE in the background, transparent to the user.

This framework has been successfully validated on real EEG data obtained with a Graz MI protocol. It allows the experimenter to identify the underlying brain processes during MI and choose the best combination of features for the subsequent classification. Work is ongoing to add further functionalities, notably functional connectivity.