CARL VON OSSIETZKY universität OLDENBURG

Hearing4all

DEPARTMENT OF PSYCHOLOGY
CARL VON OSSIETZKY UNIVERSITY OLDENBURG

Brain-Computer Interfaces on Android

Sarah Blum*, Stefan Debener*, Reiner Emkes*, Nils Volkening+, Sebastian Fudickar+ & Martin Bleichner*

*Neuropsychology Lab, Oldenburg, + Systems in Medical Engineering Lab, Oldenburg

Introduction

Electroencephalography (**EEG**) is a well-established approach enabling the non-invasive recording of human brain-electrical activity. It is widely used in the field of brain-computer interfaces **(BCI)** and monitoring. We present here a modular signal processing and classification application for mobile BCIs using EEG on Android devices. The software application **SCALA** (**S**ignal Pro**C**essing and C**L**assification on **A**ndroid) supports the Labstreaming Layer **(LSL)** to exchange information with external software and hardware.

We integrate

- data acquisition
- stimulus presentation
- data pre-processing
- classification
- delivery of feedback

on one off-the-shelf Android smartphone.

Methods

We validate SCALA using a simple auditory selective attention paradigm. In this paradigm, users are asked to shift their attention to the left or the right side while listening to a complex auditory stimulus.

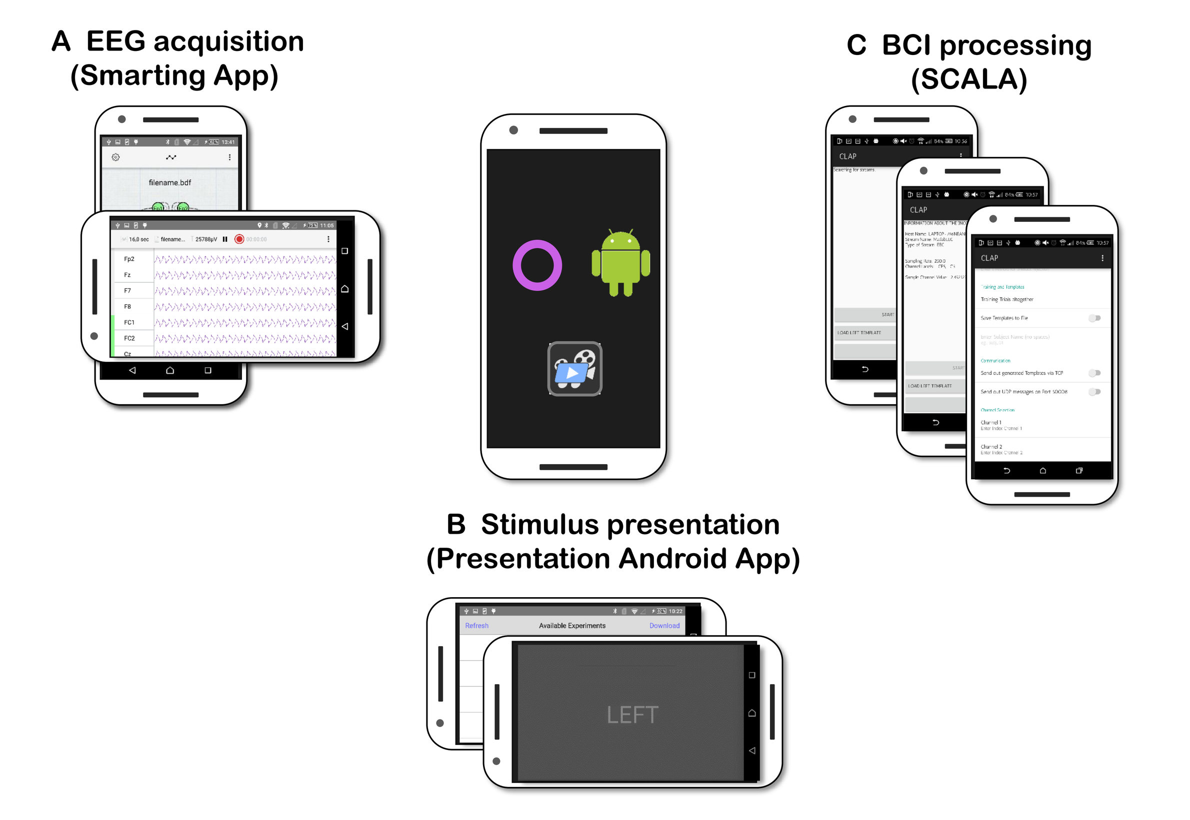

Figure 1| Our Multi-App BCI. An EEG data acquisition application **(A)** and a stimulus presentation application **(B)** communicate with our signal processing software SCALA **(C)**. All three applications run on the same smartphone and exchange data using the LSL framework.

(A) We used the Smarting Android application for EEG acquisition and storage (mbraintrain.com). It receives EEG data via Bluetooth from a small, mobile 24-channel amplifier. The Smarting application opens an LSL stream containing the raw EEG data and simultaneously records the EEG and marker data into a file on the phone for offline analyses.
(B) The stimulus presentation application from Neurobehavioral Systems (neurobs.com) was used to present experimental stimuli with high temporal precision. It uses LSL to send out and receive stimulus event marker. Incoming data is being visualized as a feedback to the user after each trial.
(C) Our signal processing application receives the event marker and the corresponding raw data. It pre-processes the data and classifies it. The classification result is send back to the Presentation application which visualizes the result. SCALA supports communication via LSL and UDP, as well as file-based communication.

Results

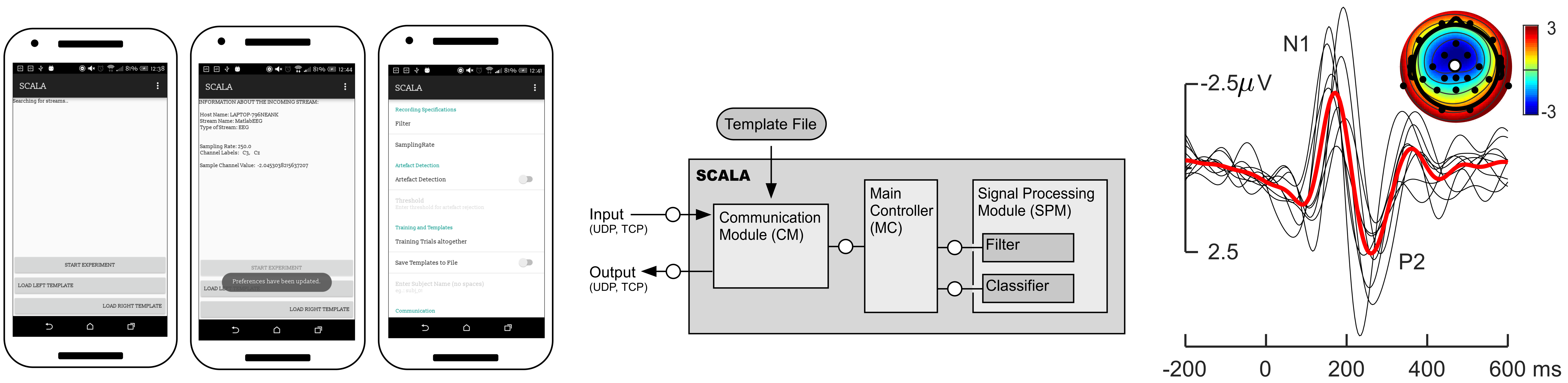

Figure 2| Results. (Left) Three screens of SCALA. The pictures show from left to right: The home screen, which is visible when the application starts. SCALA immediately starts to scan for suitable LSL streams in the network. The center picture shows meta information, which is displayed as soon as a stream has been found. The right picture shows the menu, which is accessible from the home screen. **(Middle)** An FMC Diagram showing the different functional modules of SCALA. A line depicts a bidirectional communication channel. The diagram shows the modular structure of SCALA which allows the adaptation and extension of modules. **(Right)** Auditory evoked potentials recorded with our multi-app approach. Single subject (black traces) and group mean AEPs (red line) at channel Cz. The topographic map shows the group mean N100 topography.

System Properties. We have developed a signal processing application for Android devices which processes any kind of time-series data on a trial-by-trial basis. It uses parallel processing to receive and process incoming data. The current processing pipeline contains a baseline correction, rereferencing and a Butterworth bandpass filter. The classifier uses a simple template matching procedure for the decision.
Signal Quality. We found auditory evoked potentials showing the typical P1-N1-P2 morphology and amplitude at the expected timepoint. Our signal analysis results confirm the good quality of the recorded signal.

System Performance. We analysed the data online, in a simulated online analysis (post-hoc online) and offline. We reach a classification accuracy above the empirical chance level for nearly all subjects in both the offline and the post-hoc online analysis. The mean decoding accuracy was $65,51\%$ in the offline analysis. The best subject reached an accuracy of $71,43\%$.
Stimulus Presentation. We measured the constant deviation between an event marker and the sound onst (delay) and jitter of the auditory playback using the Presentation application. We find a within-session jitter of < 3 ms which is a very good result for the sound playback accuracy on standard Android devices.

Future Work

We successfully implemented a modular BCI software solution which can be used in mobile BCI setups. Its modular architecture in combination with the support of LSL facilitates multimodal measurements including data from many senors at once. While this proof-of-concept shows that the above-chance classification of EEG data is feasible on a smartphone, some extensions would improve the existing code base substantially.

- artefact correction: for example with **subspace artefact correction method** (C.Kothe)
- advanced machine learning methods for the classification
- data cleaning, for example bad channel interpolation
- feature extraction for the usage with more advanced classifiers (such as SVM)

References

Swartz Center for Computational Neuroscience & Kothe, C. (2015). Labstreaming Layer (LSL). github.com/sccn/labstreaminglayer

Choi, I., Rajaram, S., Varghese, L.A. & Shinn-Cunningham, B.G. (2013). Quantifying attentional modulation of auditory-evoked cortical responses from single-trial electroencephalography. *Frontiers in Human Neuroscience*, 7(April), 115.

Bleichner, M.G., Mirkovich, B. & Debener, S. (2016). Identifying auditory attention with ear-EEG: cEEGrid versus high-density cap-EEG comparison. *Journal of Neural Engineering, 13*(6), 66004.

Contact

Sarah Blum, M.Sc.
Department of Psychology
University of Oldenburg

sarah.blum@uol.de
github.com/s4rify/

Source Code