UNIVERSITY OF WASHINGTON

smile

TECHNICAL DOCUMENTATION — PAGE 1

Software Download

- Go to https://github.com/gregmedlock/SMILE/ and download the folder named “SMILE Programmer”. Place this folder in your MATLAB path and be sure to include the subfolders in doing so.

Folder Contents

- ‘lib’ — Folder containing C++ DLLs and headers used in the program.
- ‘EmotivSMILE.m’ — The class file which is implemented by the main program. Contains machine learning algorithm and biofeedback functionality.
- ‘SMILE_demo.m’ — The script we used to run our demonstration for the industry panel.
- ‘SMILE_main.m’ — The script that a client would use with SMILE.
- ‘classification.xlsx’ — Data we collected through machine learning to construct a decision tree for our demo.
- ‘ClassificationTree_testing.m’ — A script for testing MATLAB’s ClassificationTree with built-in or collected data.
- ‘Electrodes.pdf’ — Electrode color coding and placement information for the Emotiv.

Electrode Placement

- SMILE contains many more electrodes than those we used. These electrodes are color coded (see ‘Electrodes.pdf’), and you can find their corresponding channel number in the field “EE_DataChannels_enum” in the EmotivSMILE class. You can sew the same conductive fabric and foam we used onto the electrodes for more comfortable use. In testing, we used EMG double-sided sticky tape to hold the electrodes onto the test subject. The optimal placement of the grounds is behind the ear, though they are not there for our current prototype.

Data Analysis

1. The program collects 10 seconds of EEG data at a sampling rate of 128 Hz from the left and right hemispheres of the brain with electrodes placed above the eyebrows, as well as from the center of the forehead. The hardware automatically normalizes signals to the grounds prior to transmission to the program.
2. The Fourier transform of each signal is calculated. NOTE: If you change the sampling rate or accumulation time you will experience errors with our program, and will need to re-optimize it.
3. The magnitude spectra from the left and right hemispheres are normalized to that from the center to reduce noise from the forehead region, and the resulting spectrum from the left is divided by that from the right to examine frontal asymmetry in brain signals. Valence and arousal calculations and machine learning use this data.

Machine Learning

- SMILE optimizes use through a machine learning algorithm. The machine learning is implemented by building a classification-based decision tree based on user responses. We collected responses indicating positive, negative, and neutral emotions, though this can be altered to include more or different emotional states.
- An example of what the tree will look like can be seen running by the script 'ClassificationTree_testing.m', which draws on our data and a built-in data set in MATLAB (Fisher's Iris flower data set).
- To discern emotional states, we looked at 1 Hz frequency blocks in α and β waves (8-12 Hz and 12-30 Hz). The class is already functionalized to look at smaller or larger frequency blocks if you wish to modulate this. Look for the field "bandSize" to change this property.
- After each run (i.e. call on EmotivSMILE.runSMILE()) that incorporates machine learning, the new data is added to the initial data and the results are compiled to create a new decision tree. We decided to keep this outside of the execution loop in order to improve efficiency, and thus dynamic learning is not possible with our current class.
- A summary of the running modes which use machine learning are discussed in the User's Manual.