DETAILED REPORT

The project works in modules. The major blocks are:

1. Extraction of raw EEG signals
2. Processing of the EEG signals
3. Machine learning
4. Application of data
5. Interfacing with the drone

Extraction of EEG signals

The most important and one of many challenges of this project is the acquiring of the raw EEG data.

The acquiring instrument used is EMOTIV headset which has been developed by the EMOTIV Company which has opened up completely new methods for research. This headset contains 16 sensors which read from the brain in a specific spatial arrangement. The signals can be processed collectively (all 16 channels at once) or independently (Independent analysis). So let’s start with the signal acquisition. Check the EMOTIV kit, it should contain contains

1. The EMOTIV headset
2. The sensors (gold plated sensors , 16 in number)
3. A Bluetooth module
4. A charging cable for the EMOTIV headset

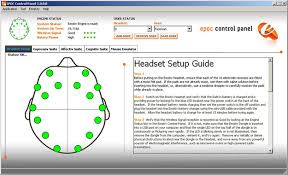
To start you must have the following soft wares.

1. Emotiv Simulink server
2. The EMOTIV control panel
3. MATLAB 2016
4. BCILAB 1.1
5. EEGLAB
6. Open-vibe
7. Lab streaming layer

* Setting up the headset:

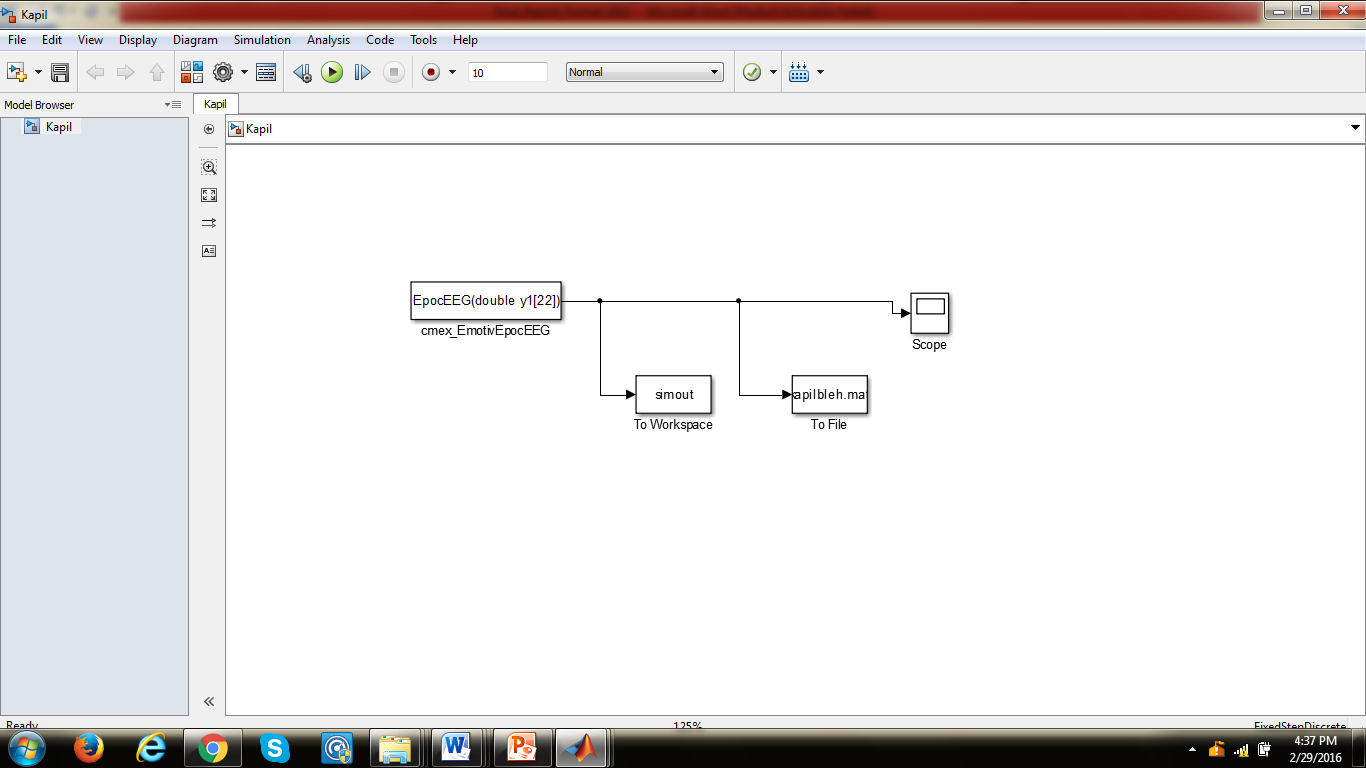
The felt pad of the sensors has to be soaked with saline and set carefully in the headset. Once this is done set the headset carefully and the status has to be checked. All the sensors must go green. This may take some time because setting the headset at right spot takes little time. Check the manual for the panel if having difficulty.

* Acquiring the signals



Start the Simulink server: All three statuses should go green





The MATLAB has to be started to run the Simulink circuit for acquiring the signal. You can find this in the EPOC Simulink server in examples or you can read the readme PDF.

Run this simulink circuit and go to matlab command window to save the time series object to a MATLABfile.

//run in MATLAB command window

C(any variable) = simout.Data;

Save(‘anyfilename.mat’,’c’);

* Data processing using EEGLAB

The noise and the artifacts need to be removed from the acquired data. To do so we need EEGLAB

Load EEGLAB and import the .mat file that u created. Here in this case its anyfilename.mat.

After loading the data we need to filter the noise from the data. To do so pass a band pass filter [1 to 50 hz].

We need to put event markers for preparing the model. [ if the event field is showing null , add a boundary marker by removing some data from the dataset.]

The event markers must be of two different kinds eg; left and right or type1, type2.

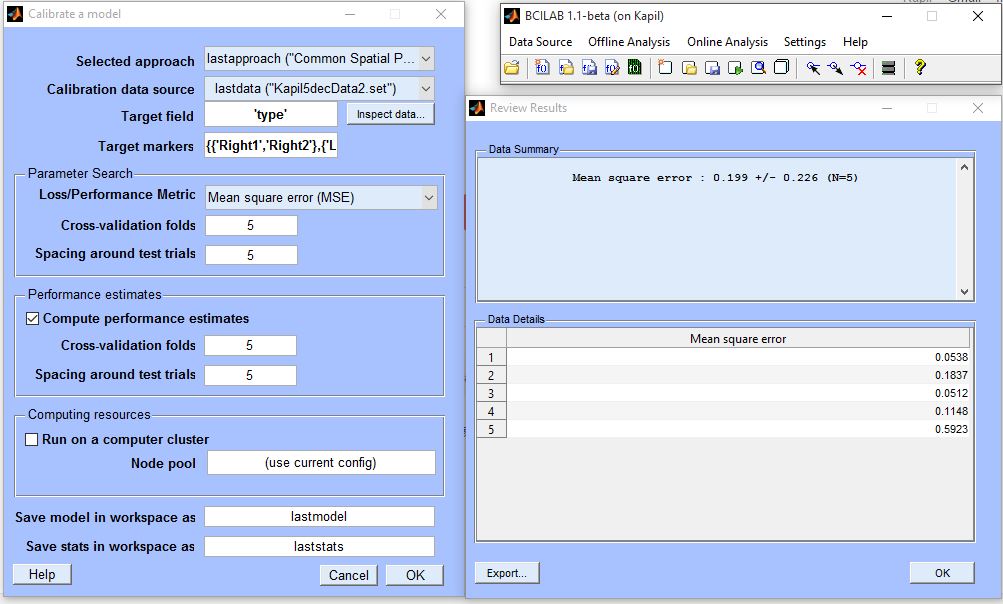
Save this as a .SET file.

* Training the model

Load the saved .set data to BCILAB . The approach for offline analysis must be defined. for eg: In this case CSP(common spatial pattern) was used because subject was instructed to imagine movements . CSP is best suited for oscillatory processes.

The fir filter values in name value pair is :'FIRFilter',[7 8 26 28] since the target is to tap the alpha and beta waves. The machine algorithm chosen is LDA analysis.

{ {‘Righ1’,Right2’} , {Left1’,’Left2’} } [\\two](file:///\\two) classes in this case



Once the modelling is done, we need to apply a twin data to the prepared model to check the prediction response. The entire process needs to be scripted later for automatic loading and calculation in BCILAB.(traintesst.mat)

* Interfacing with the Arduino

Download the Arduino toolbox on MATLAB and refer to the given code.

Servotry.mat

Satya3am.mat

Servotry2.mat

If faced with error related to string and char expected data types, change the strjoin.m file in the MATLAB.

Which all -strjoin.m // to find where the strjoin file is saved.

Online analysis

* Install python and pyside to install lab streaming layer and openvibe in the system.(acquisition software)
* Connect to the emotiv driver in openvibe.

Common error: first call to edk.dll failed. Fix it by either setting the right path. Give the directory of EMOTIV premium library. Put \applicationx86 as path. If it doesn’t work(it did work for us !) then u need to change the cmake file and also change the system PATH variable in the system.

ENABLE: LSL output

NFT CONNCTION SUCCESFUL! Is the message.

* Go to MATLAB and run the BCILAB. In BCILAB go to online analysis and >online analysis>read from LSL.

Change the ‘type’ to ‘name’ and ‘value’ to ‘openvibeSignal’ as displayed in the openvibe GUI,

* The real time data is being read to MATLAB and can be written to LSL. Onl\_predict()will predict the output. Refer to the provided codes.(Script1 and Script2)
* The same prepared model will be used for this prediction as well.

Interfacing with the drone.

Refer to drone code (ARDrone.mat)

Make a drone object in MATLAB command window

B = ARDrone

B.moveUp() will take it upward.(play with it!)

Refer to drone test.m