Facial Expressions Recognition Using the Emotiv EPOC Headset

Alexandru Cohal, Vitor Ribeiro Roriz,
March, 2018

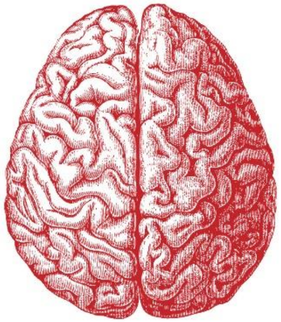
Prof. Dr.-Ing. Andreas König

Overview

1. Introduction
2. Objective
3. Data Acquisition
4. Data Preprocessing
5. Feature Extraction
6. Classifiers
7. Real-Time Recognition
8. Conclusions

Introduction

- Multiple **Human - Computer Interaction (HCI)** methods have been developed recently make technology available to more users with different needs and goals
- A **Brain – Computer Interface (BCI)** a solution for HCI which uses signals captured from the brain for controlling an external activity
- BCI used in the rehabilitation of persons with limited physical control (due to a brain injury for example)
- **Electroencephalography (EEG)** a method for monitoring the electrical neural activity of the brain

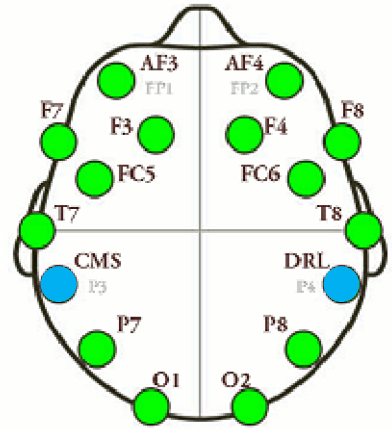


http://wiki.cas.mcmaster.ca/index.php/Motivations_for_the_Studies_of_HCI

Investigating the Emotiv EPOC for cognitive control in limited training time – M. Lang, T. Mitrovic, *University of Canterbury* (2012)

Introduction

- **Emotiv EPOC** ➡ a compact, wireless EEG device, easy to setup and use
- Has *16* sensors (electrodes): *14* for data (μV) and *2* for reference



- Data is acquired with *128 Hz* frequency
- A saline solution has to be used for increasing conductivity

Investigating the Emotiv EPOC for cognitive control in limited training time – M. Lang, T. Mitrovic, *University of Canterbury* (2012)
https://www.emotiv.com/epoc/

Introduction

- **Emotiv EPOC** ➡ used in many researches

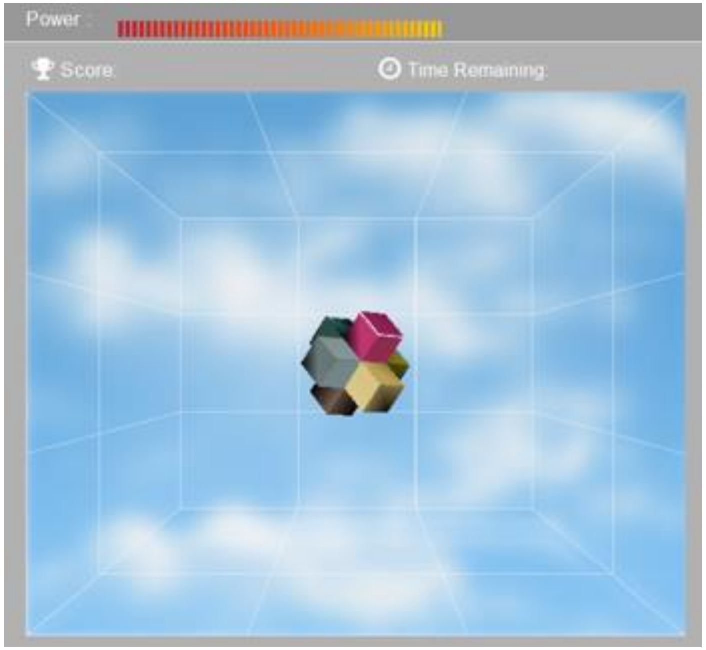
- The general trend ➡ Emotiv EPOC is not suited for recording mental commands

  - *“[…] only suitable for a beginner level brain signal measurement and research“*
  - *“[…] the Emotiv EPOC neuroheadset is not medically functional or reliable”*
  - *“With noninvasive technology, the skull hinders the signals being detected by EEG”*

Consumer-grade EEG devices: are they usable for control tasks? – R. Maskeliunas et al., *PeerJ* (2016)
Using the Emotiv EPOC Neuroheadset as an EEGControlled BrainComputer Interface – A. Hawkes et al., *Appalachian State University* (2015)

Introduction

- Tests done by us ➡ the Emotiv own software (**Emotiv Control Panel**)

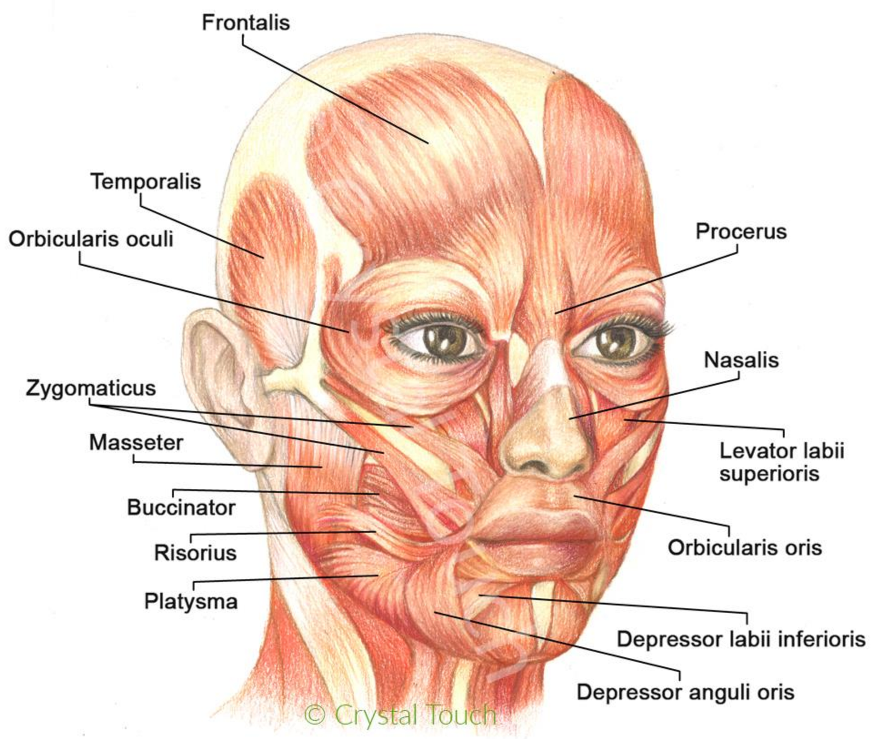


  - *3* users
  - *20 – 25* minutes of training each
  - *3* classes (“*Neutral*” and *2* actions for moving the cube)
  - Very poor classification results

https://www.emotiv.com/brain-controlled-technology/

Introduction

- Emotiv EPOC can be used for **Electromyography (EMG)** → monitoring the electrical activity of the muscle tissue → facial expressions can be detected
  - Blink
  - Left wink
  - Right wink
  - Raise brow
  - Smile
  - Laugh



https://www.emotiv.com/comparison/
https://crystal-touch.nl/muslces-of-facial-expressions-and-how-they-work/

Objective

- Using the *14* channels data provided by the *Emotiv EPOC* headset, recognize in real-time the *Neutral* state and *5* facial expressions:
    - Left wink / blink (BL)
    - Right wink / blink (BR)
    - Strong blink (BB)
    - Open mouth (OM)
    - Full mouth (FM)

https://www.emotiv.com/comparison/

Data Acquisition

- For the training step:
  - Each recording file is composed of *100 raw data samples* for each of the *14 channels* ($\frac{100\ samples}{128\ Hz} = 0.78\ sec$)
  - The length was experimentally observed to be enough to capture only the behavior of the facial expression and to give enough time to the user to react
  - *20 files* were recorded for each class
  - Implementation: `data-acquisition.py`

```
vitorroriz@vitorroriz-XPS-15-9560:~$ python data-acquisition.py v_bl_0
Prepare to act in :
3
2
1
ACTION!
Recording done.
v_bl_0.npy file saved
```

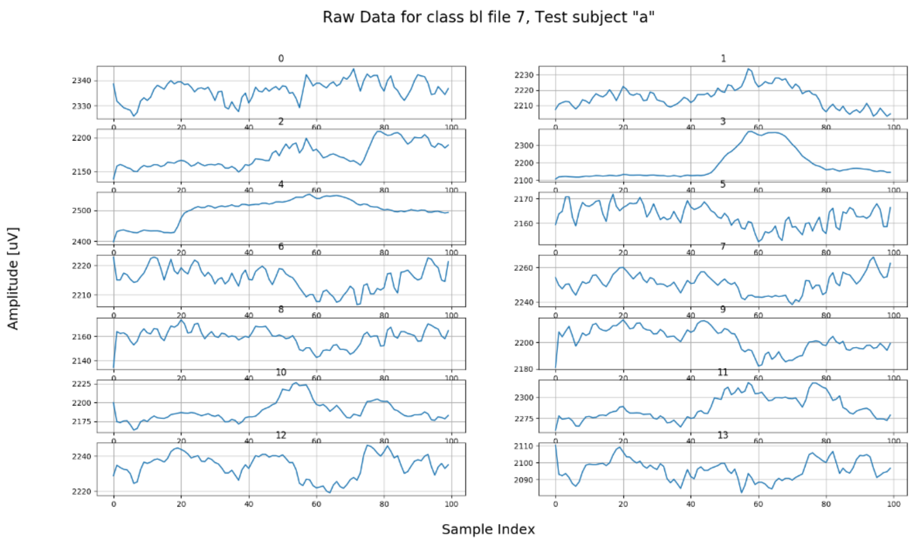
Data Acquisition

- We observed that:
  - The same facial expression is performed differently by distinct users ➡ results in different signal patterns
  - Signal patterns ➡ influenced by external factors such as:
    - positions of the headset and of the sensors
    - amount of the saline solution used, etc.

- For better results in the training / classification phase:
  - Not mixing data from different users
  - Not mixing data from different acquisition sessions

- For each utilization, new training dataset has to be build to capture the current environment conditions ➡ Large datasets are inconvenient to acquire for the users ➡ Classifiers relying on large datasets are not feasible

Data Acquisition



Distinct users having different signal patterns for the “Blink Left” expression

Data Preprocessing

- **DC value extraction**
  - We observed in the frequency spectrum that all the signals have a DC component (*2000* µV)
  - We extracted *1250* µV → for some facial expressions, some signals drop below *2000* µV (e.g. “Open Mouth”)

- **Filtering**
  - We used a *Low-Pass Filter* (*Digital Butterworth Filter*) with the cut-off frequency *50 Hz* and order *15*

- **Normalization**
  - Done for each channel individually, based on the average of *200* “Neutral” state *samples*

Data Preprocessing

The resulted signals of the "Blink Left" expression after filtering and DC component removal* for all the channels

Feature Extraction

- **Ratios for each channel**
  - Data was split into windows (25 samples) and overlapped (15 samples)
  - $Per\ Channel\ Ratio = \frac{Average(Window\ preprocessed\ data)}{Average(Neutral\ state)}$

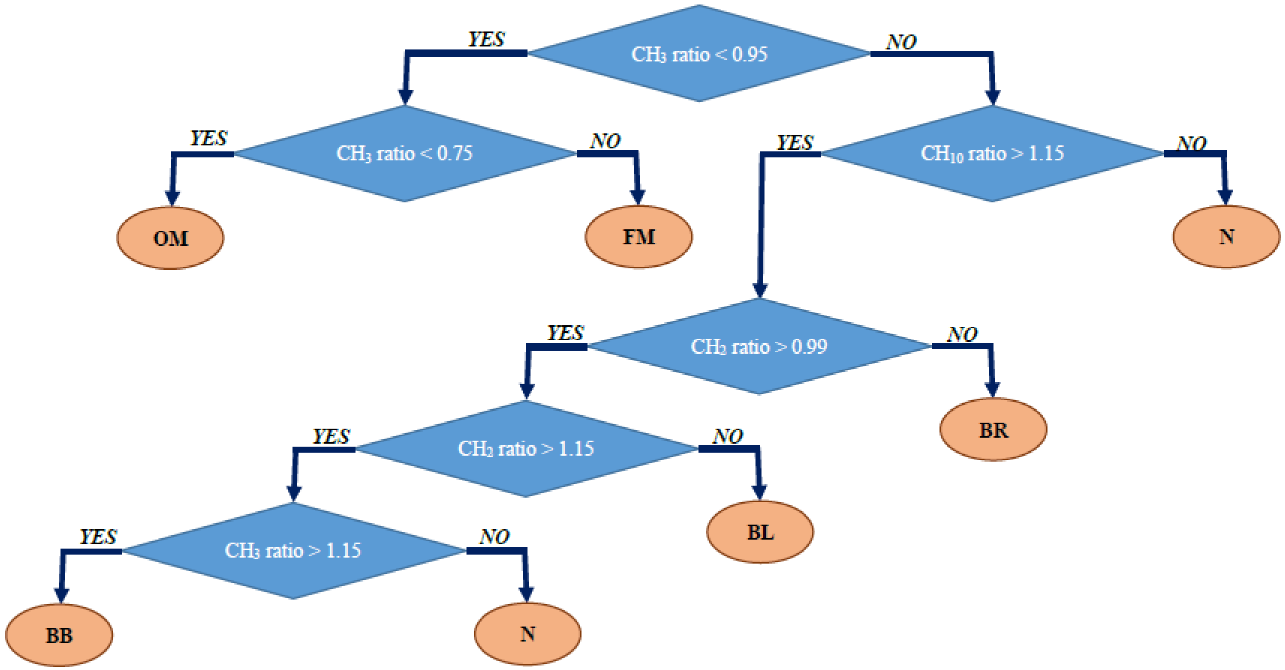
- **Hjorth parameters**
  - Mobility: $\sqrt{\frac{var(\frac{dy(t)}{dt})}{var(y(t))}}$ Complexity: $\frac{Mobility(\frac{dy(t)}{dt})}{Mobility(y(t))}$

- **Ratio between the average power of frequency spectrum bands**
  - *[0, 10] Hz* and *[25, 45] Hz*

A Novel EEG Feature Extraction Method Using Hjorth Parameter – S.-H. Oh et al., *International Journal of Electrical Engineering* (2014)
Online Recognition of Facial Actions for Natural EEG-based BCI Applications, D. Heger et al., *Affective Computing and Intelligent Interaction: Fourth International Conference* (2011)

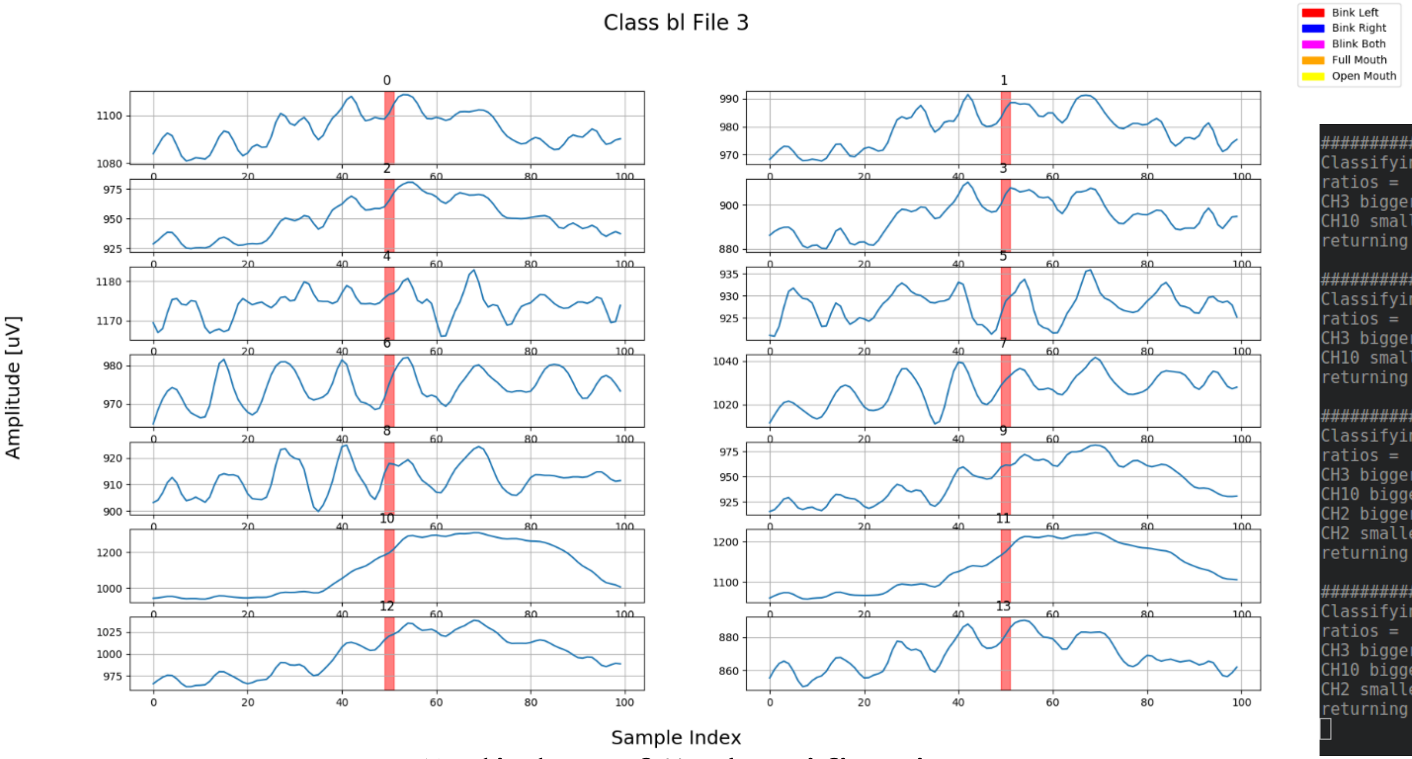
Classifiers

- **Decision Tree classifier based on thresholds** *(own implementation)*
  - All preprocessing steps (DC value extraction, filtering, normalization)
  - Features used: Ratios for each channels
  - “Filtering” the results of the classification



Classifiers

- **Decision Tree classifier based on thresholds** *(own implementation)*
  - Overall accuracy: *73.33%* (for all classes)




```
################################################################
Classifying window  0
ratios =  0.969982665519 1.05020700212 1.05159074405
CH3 bigger than 0.95
CH10 smaller than 1.16
returning neutral

################################################################
Classifying window  25
ratios =  0.994605245566 1.06739435379 1.15199260266
CH3 bigger than 0.95
CH10 smaller than 1.16
returning neutral

################################################################
Classifying window  50
ratios =  1.01111983509 1.07198208417 1.42345145753
CH3 bigger than 0.95
CH10 bigger than 1.16
CH2 bigger than 1.01
CH2 smaller than 1.15
returning bl

################################################################
Classifying window  75
ratios =  0.985876238844 1.0607620802 1.301138085
CH3 bigger than 0.95
CH10 bigger than 1.16
CH2 smaller than 1.01
returning br
```

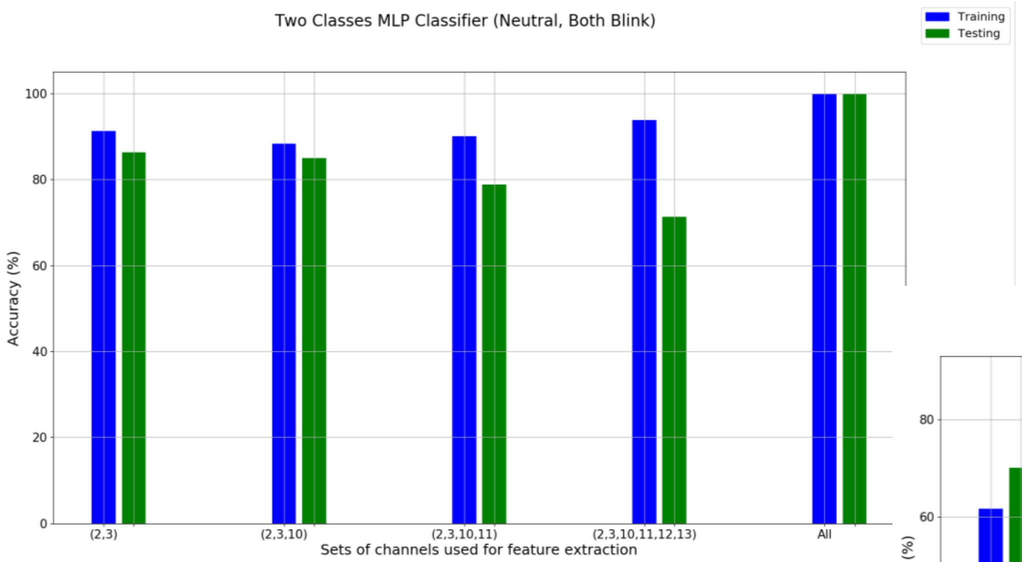
"Blink Left" classification

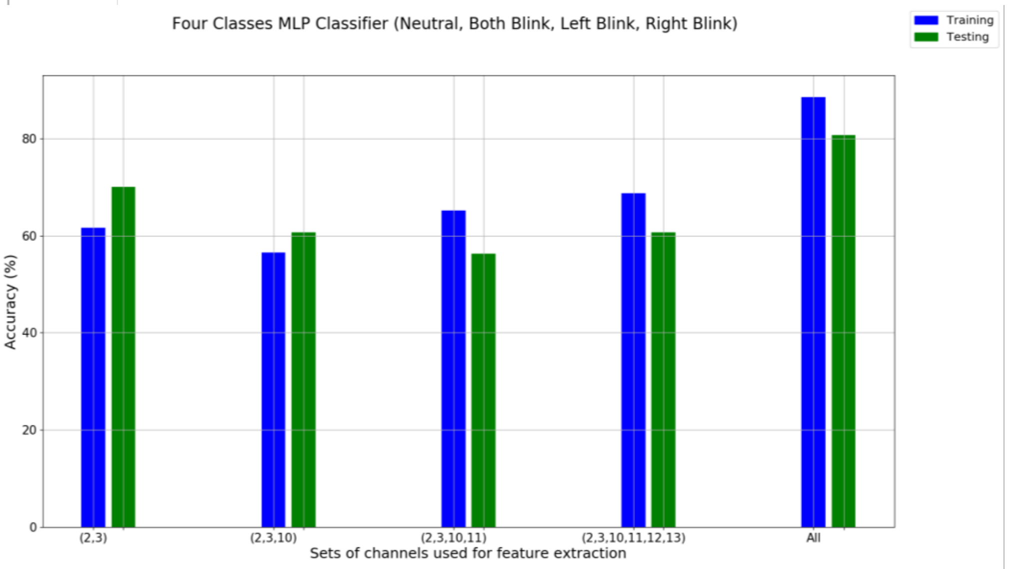
Classifiers

- **Multi-Layer Perceptron**
  - Preprocessing steps: DC value extraction, filtering
  - Features used: Ratios for each channels
  - Classification with different sets of classes based on different sets of channels
  - One hidden layer with more neurons when using more channels (number of neurons between *5* and *17*)
  - Logistic activation function used

Classifiers

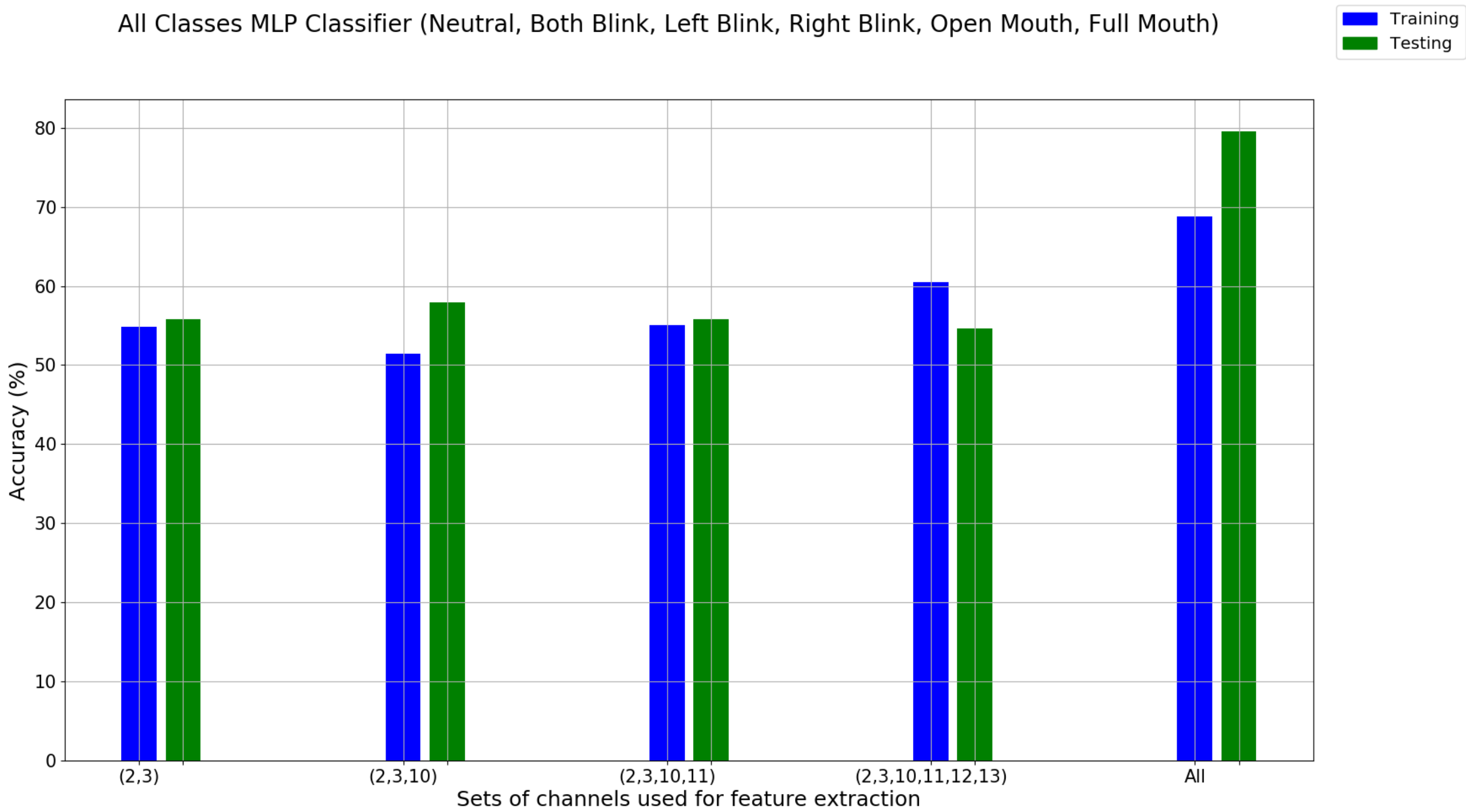
- **Multi-Layer Perceptron**





Classifiers

- **Multi-Layer Perceptron**

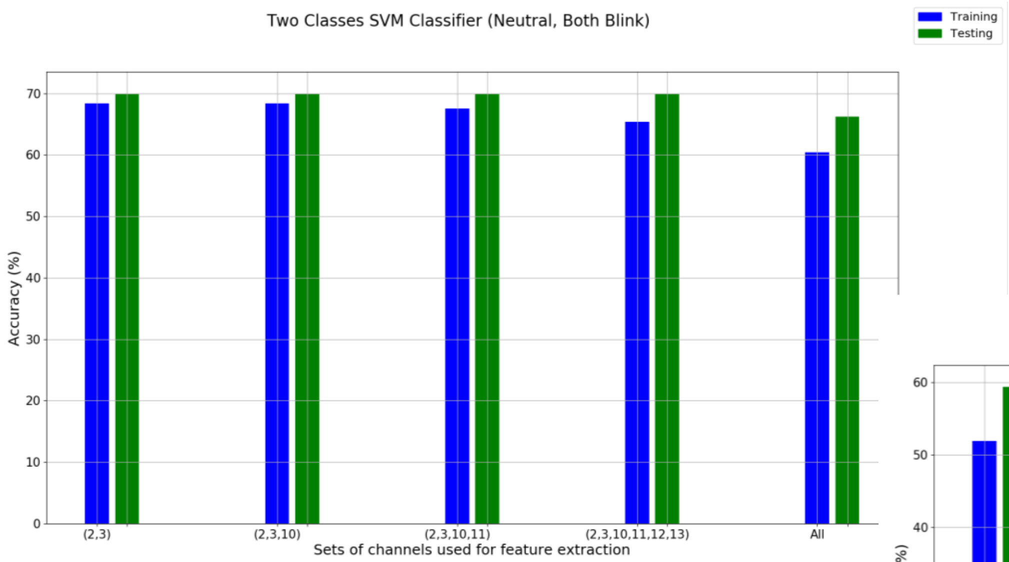


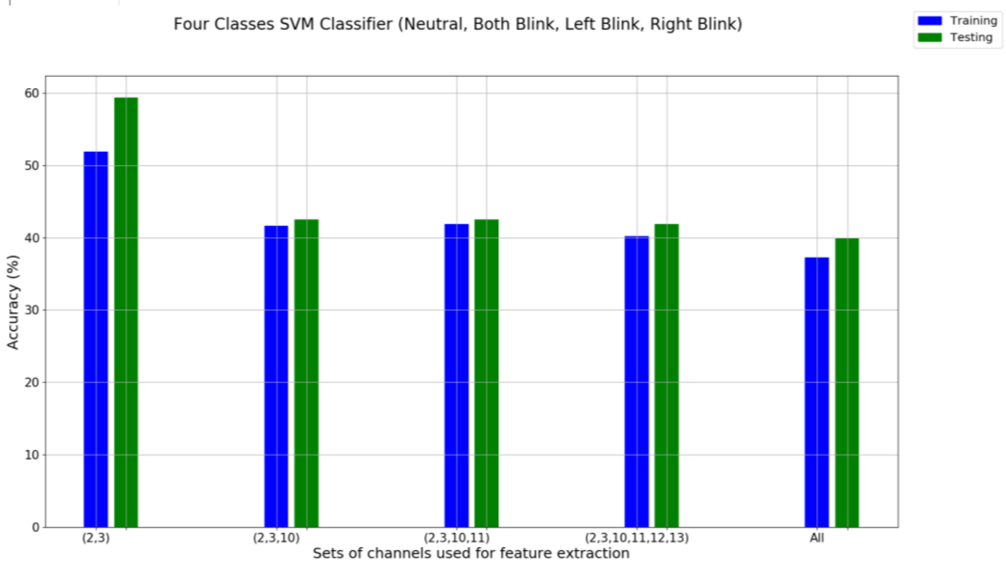
Classifiers

- **Support Vector Machine**
  - Preprocessing steps: DC value extraction, filtering
  - Features used: Ratios for each channels
  - Classification with different sets of classes based on different sets of channels

Classifiers

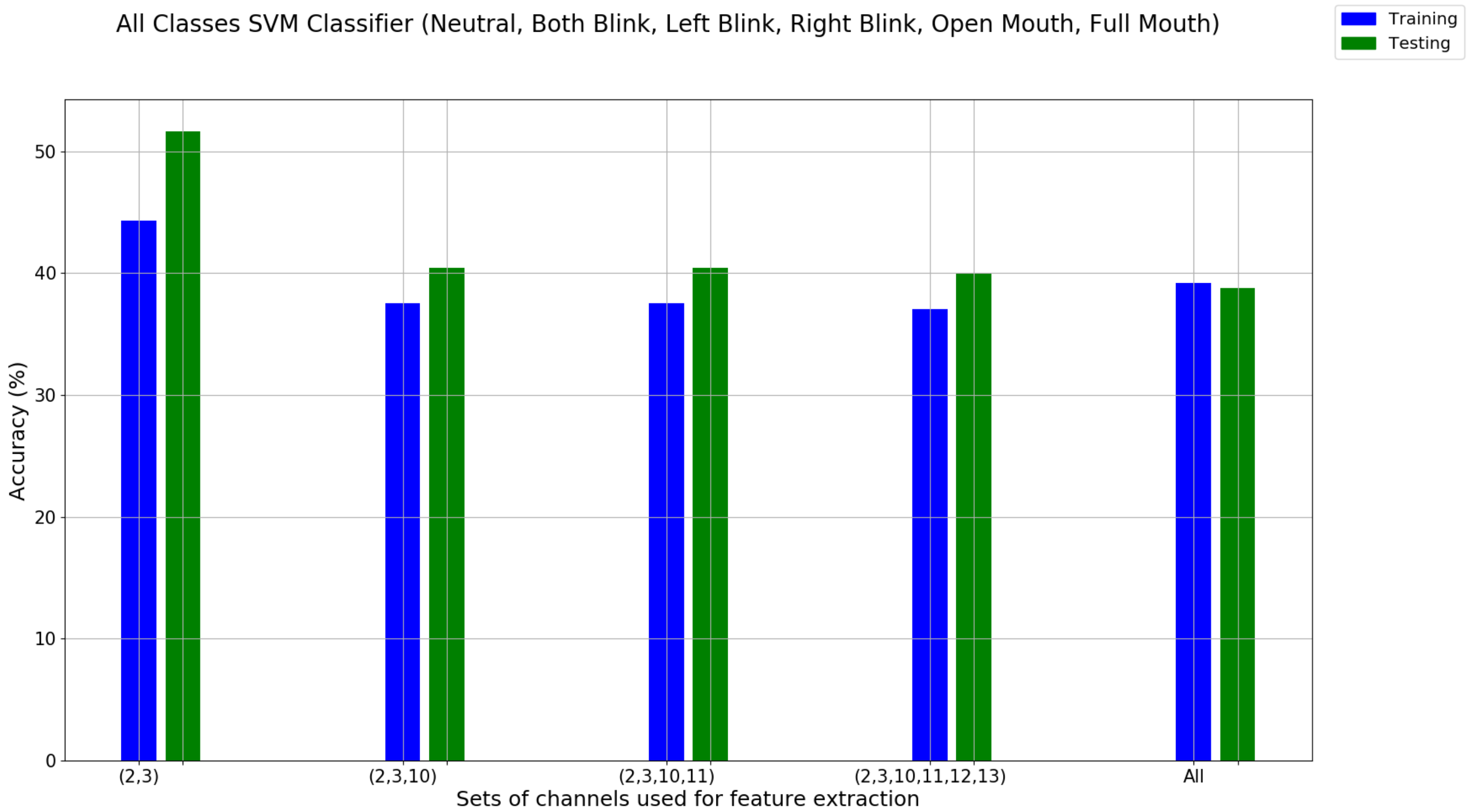
- **Support Vector Machine**





Classifiers

- **Support Vector Machine**



Classifiers

- **K-Nearest Neighbours**
  - Preprocessing steps: DC value extraction, filtering
  - Features used: Ratios for each channels, Hjorth parameters*, Average power of frequency bands*

    * Poor results
  - Classification with different sets of classes based on different sets of channels

Classifiers

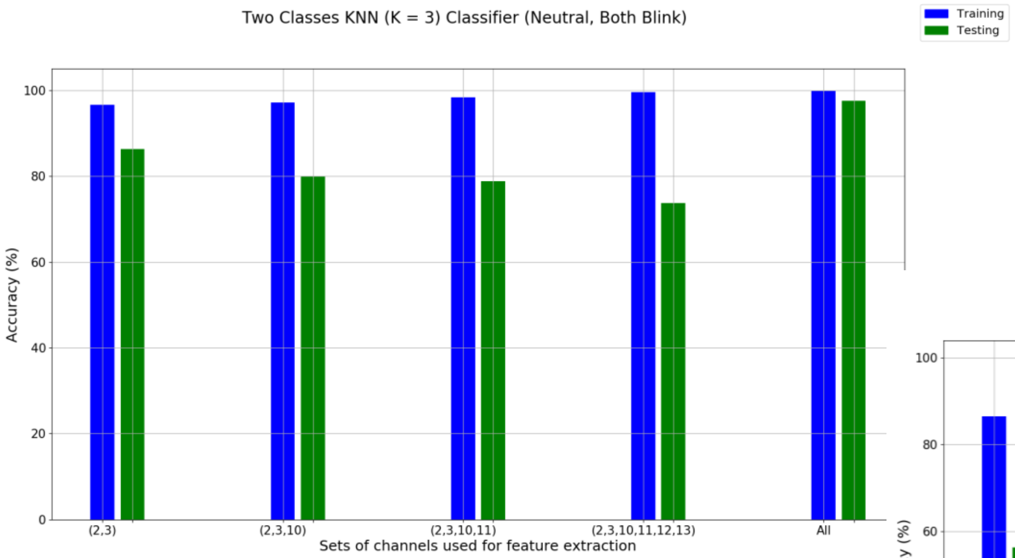
- **K-Nearest Neighbours**

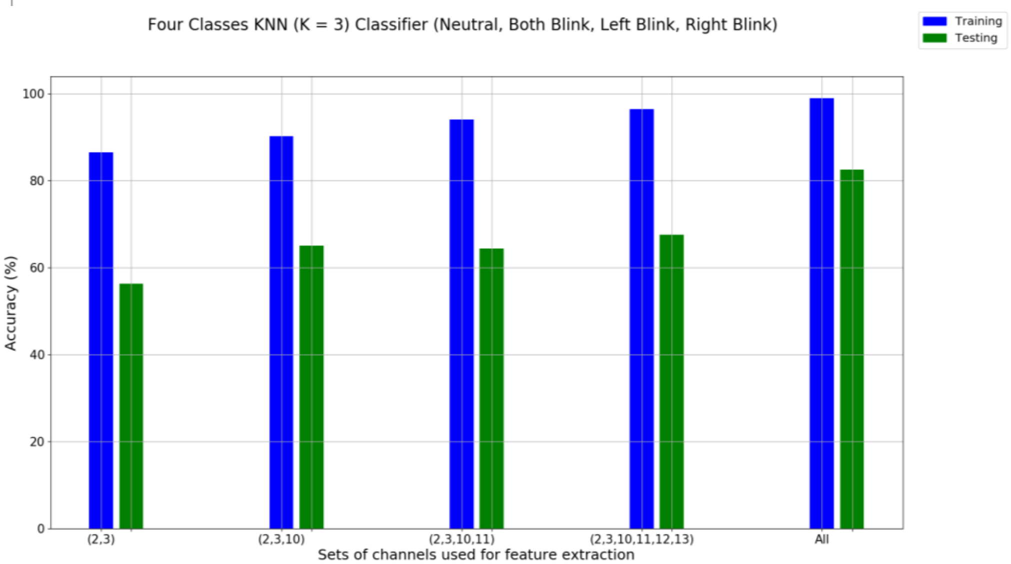
    - Using only Hjorth parameters, the overall accuracy for all classes recognition using all channels is *46.67%*

    - Using Hjorth parameters and Average power of frequency bands, the overall accuracy for all classes recognition using all channels is *27.08%*

Classifiers

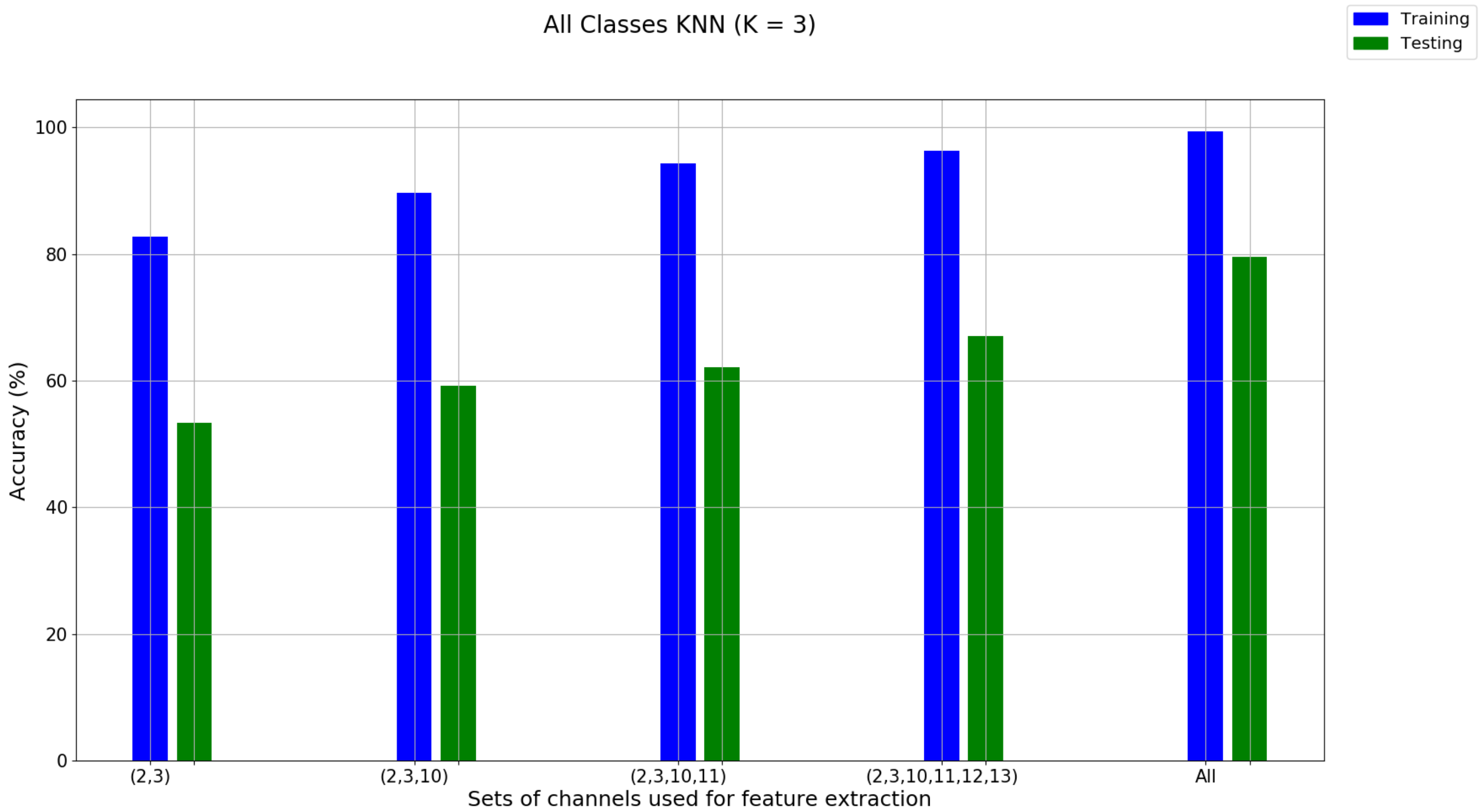
- **K-Nearest Neighbours**





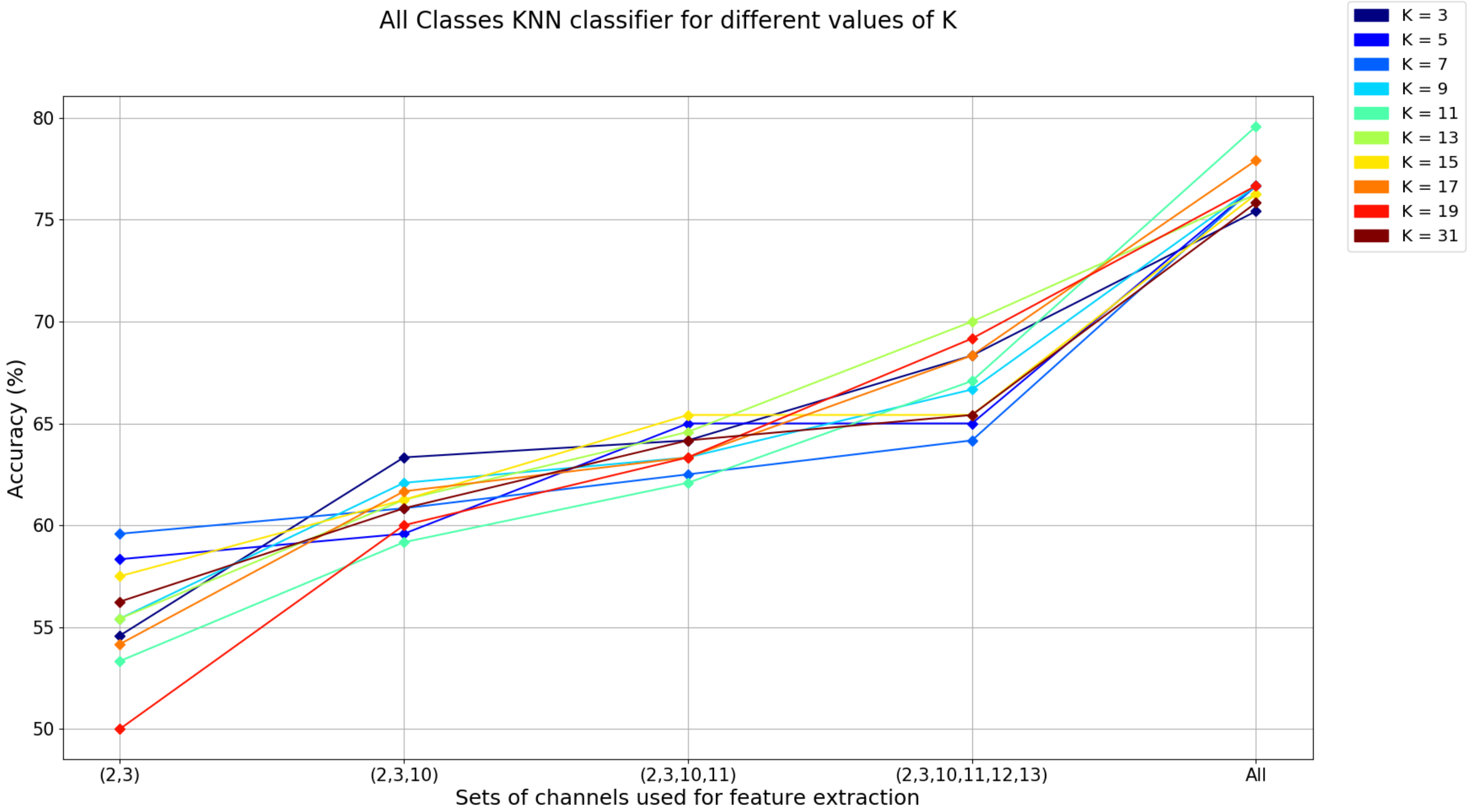
Classifiers

- **K-Nearest Neighbours**



Classifiers

- **K-Nearest Neighbours**



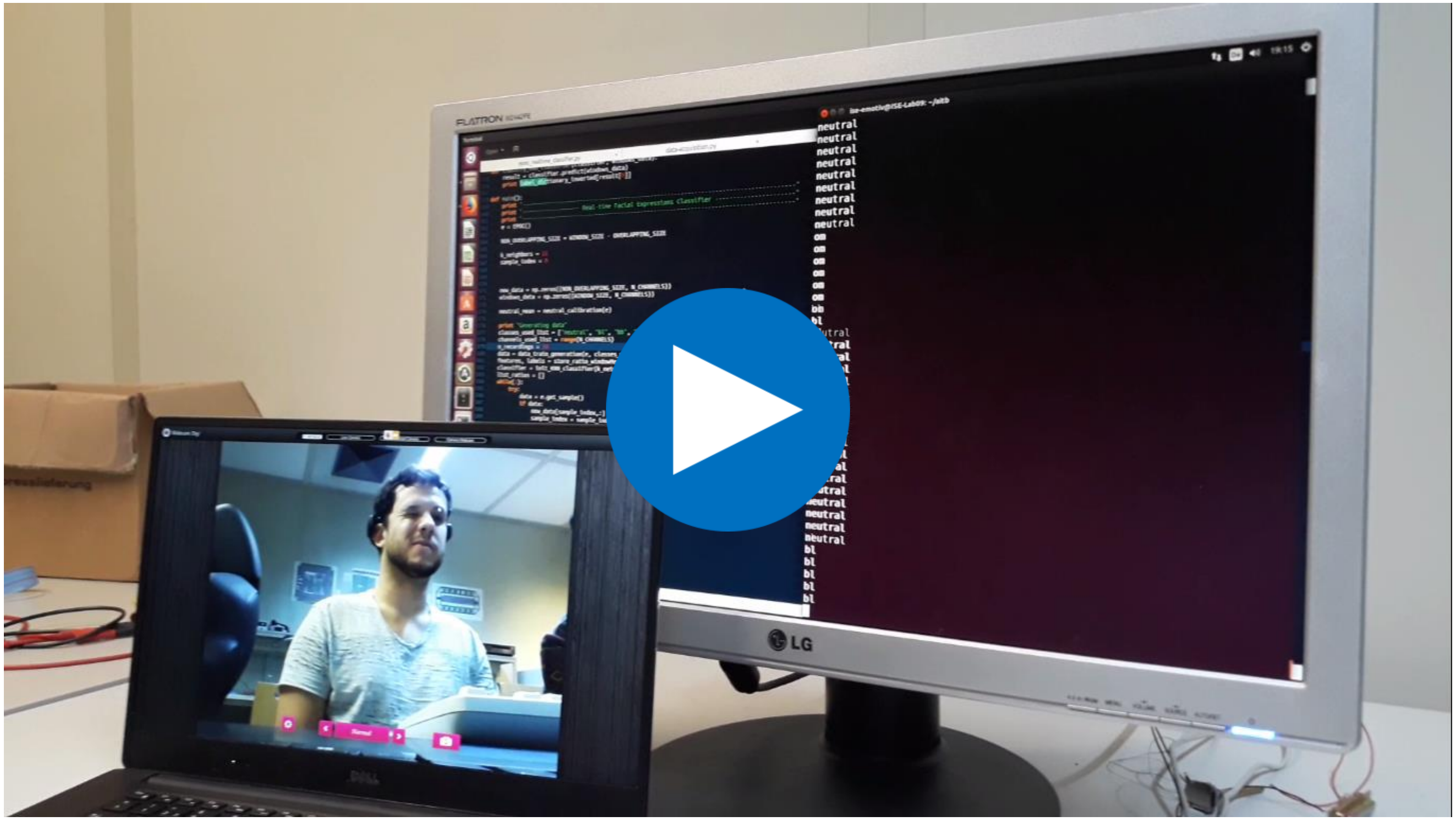
Real - Time

- **Setup**
  - Calibration of the Neutral state → calculation of the mean value of each channel for 10 windows of 25 samples each
  - Capturing the data for each class → 10 recordings for each class, each recording having 10 windows of 25 samples each
  - Training* of the KNN classifier

- **Loop**
  - Real-time acquisition of data split into windows of 25 samples with an overlapping of 15 samples
  - Classification for each window

Real - Time

[Link to the Video of the entire Real-Time process (Setup + Loop)]



Conclusions

- We successfully classified in Real-Time the "Neutral" state and 5 facial expressions:
  - Left wink / blink
  - Right wink / blink
  - Strong blink
  - Open mouth
  - Full mouth

- The best classifier was KNN (K = 11)

- Due to the window size of *25 samples* and the overlapping of *15 samples*, a new classification result can be delivered every *0.07 seconds*

Thank you!

Danke!

Obrigado!

Mulțumim!