Building A Brain Computer Interface To Play Space Invaders Using Python

Tim de Boer

Hi!
My name's Tim

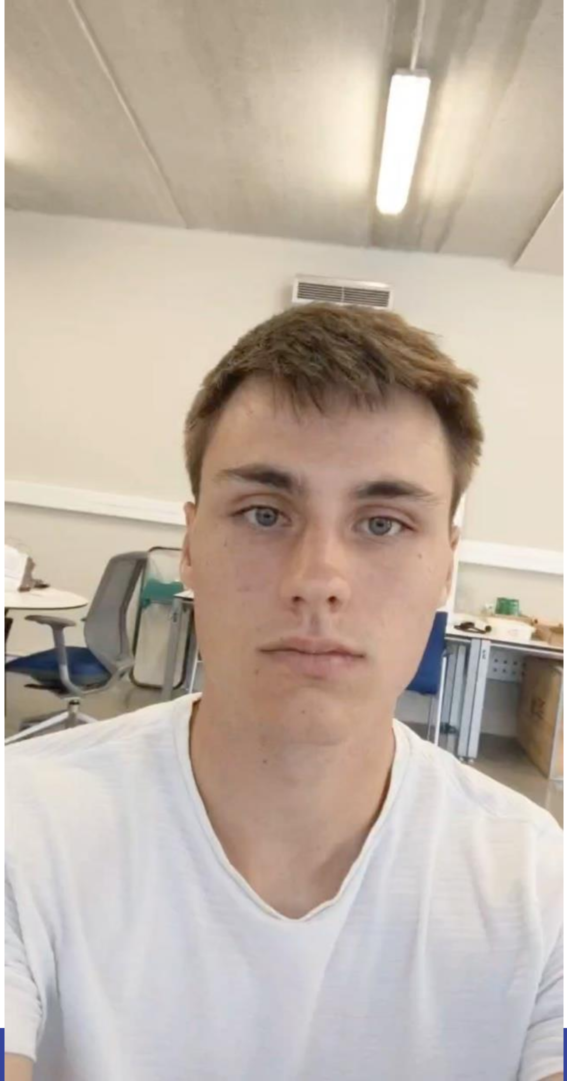
- Based in **Amsterdam, Netherlands**
- **AI Engineer** at **Delphyr**, building **medical AI**
- Passionate about **AI** & the **human body**
- **MSc Artificial Intelligence**
- **BSc Human Movement Sciences**
- Running | Fitness | Dancing | Languages

In **2022,** I spent 6 months building **a BCI** to play **Space Invaders** using **Python** & **g.tec Unicorn Hybrid Black**

Sunny Elche!

BMIlab

Brain-Machine Interface Systems Lab



What will I talk about today?

- Quick **introduction** to **Motor Imagery BCIs** and … **important**
- The **6 building blocks** of the BCI syst…
  - The user
  - Data collection
  - Pre-process…
  - Feature extra…
  - Prediction
  - Output
- **Real-time Experiments**

Alongside code examples!

What is an MI-BCI?

- **Motor Imagery (MI)**: Thinking about a certain movement without execution
- Record the brain signals
- **Machine Learning** interprets these signals to control **external devices** like computers to play games
- But also, to control robotic arms and exoskeletons!
- A system like this is a type of **Brain-Computer Interface**

Why?

Spinal cord injury: as many as 500 000 people suffer each year

2 December 2013 | News release | GENEVA |Reading time: 3 min (741 words)

Survival rates worse in low-income and middle-income countries

News release

2 December 2013 | *GENEVA* - As many as 500 000 people suffer a spinal cord injury each year. People with spinal cord injuries are 2 to 5 times more likely to die prematurely, with worse survival rates in low- and middle-income countries. The new WHO report, "International perspectives on spinal cord injury", summarizes the best available evidence on the causes, prevention, care and lived experience of people with spinal cord injury.

Males are most at risk of spinal cord injury between the ages of 20-29 years and 70 years and older, while females are most at risk between the ages of 15-19 years and 60 years and older. Studies report male to female ratios of at least 2:1 among adults.

BMIlab

Brain-Machine Interface Systems Lab

How did we build it?

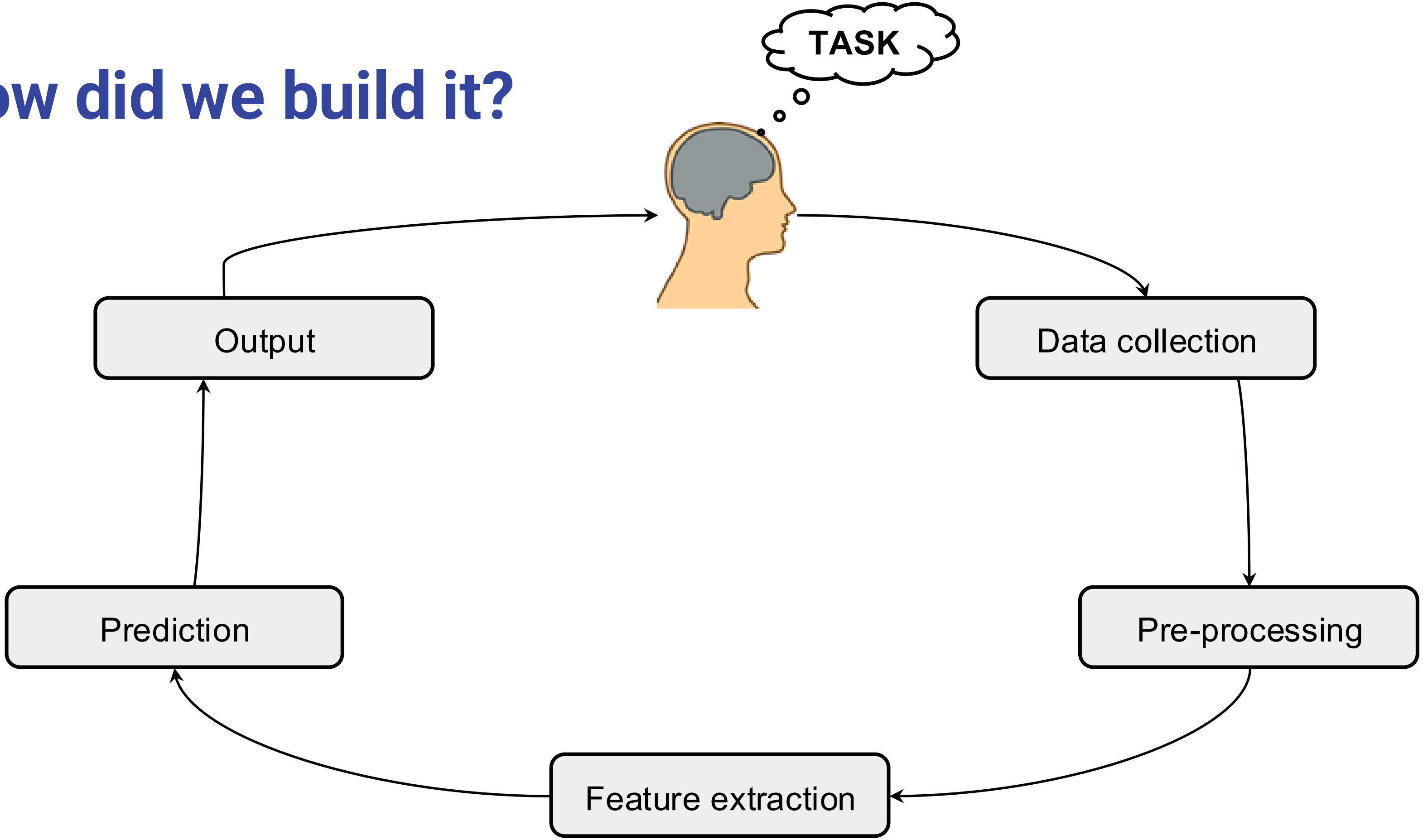


Figure inspired from: Marcel Van Gerven, Jason Farquhar, Rebecca Schaefer, Rutger Vlek, Jeroen Geuze, Anton Nijholt, Nick Ramsey, Pim Haselager, Louis Vuurpijl, Stan Gielen, et al. The brain–computer interface cycle. *Journal of neural engineering*, 6(4):041001, 2009.

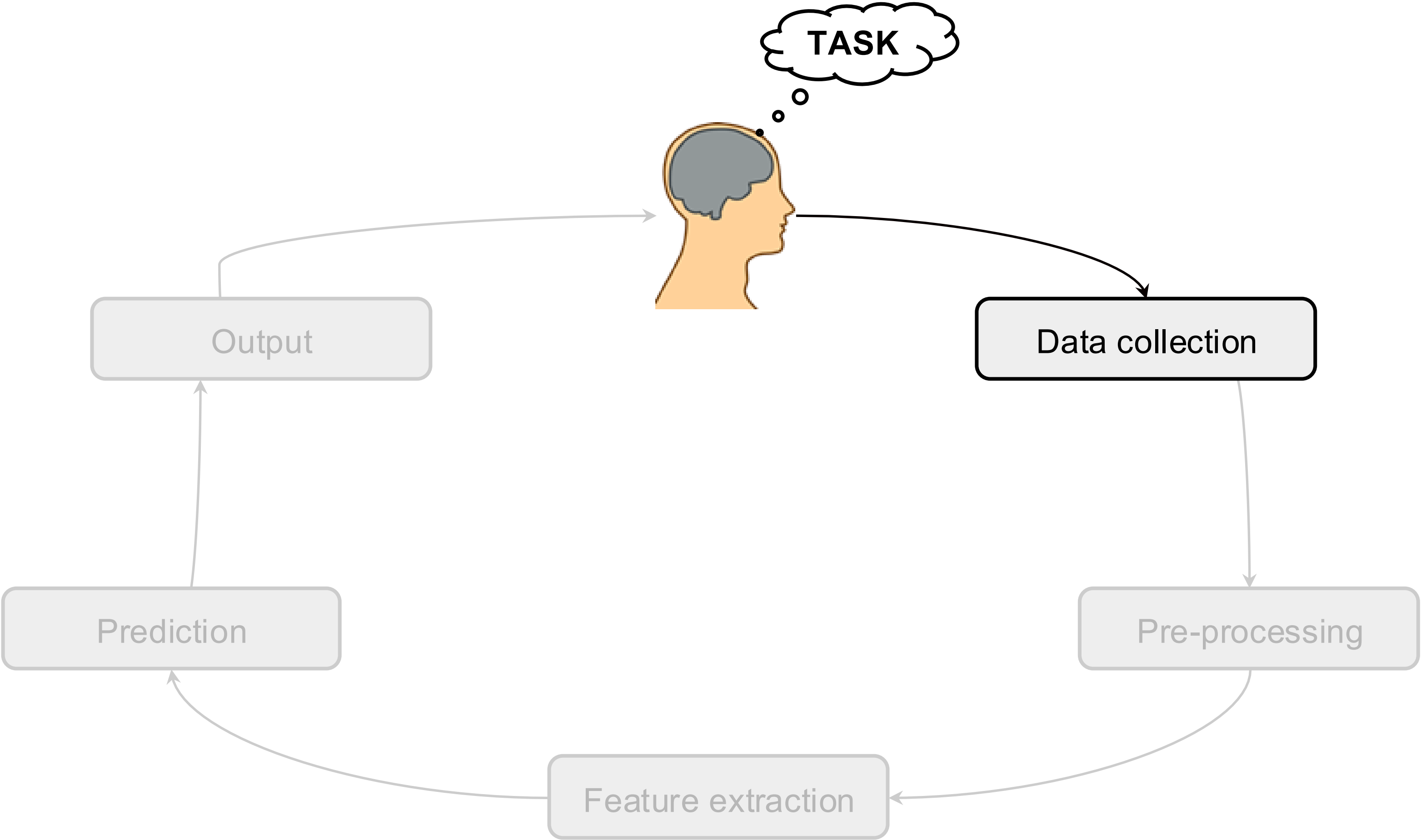


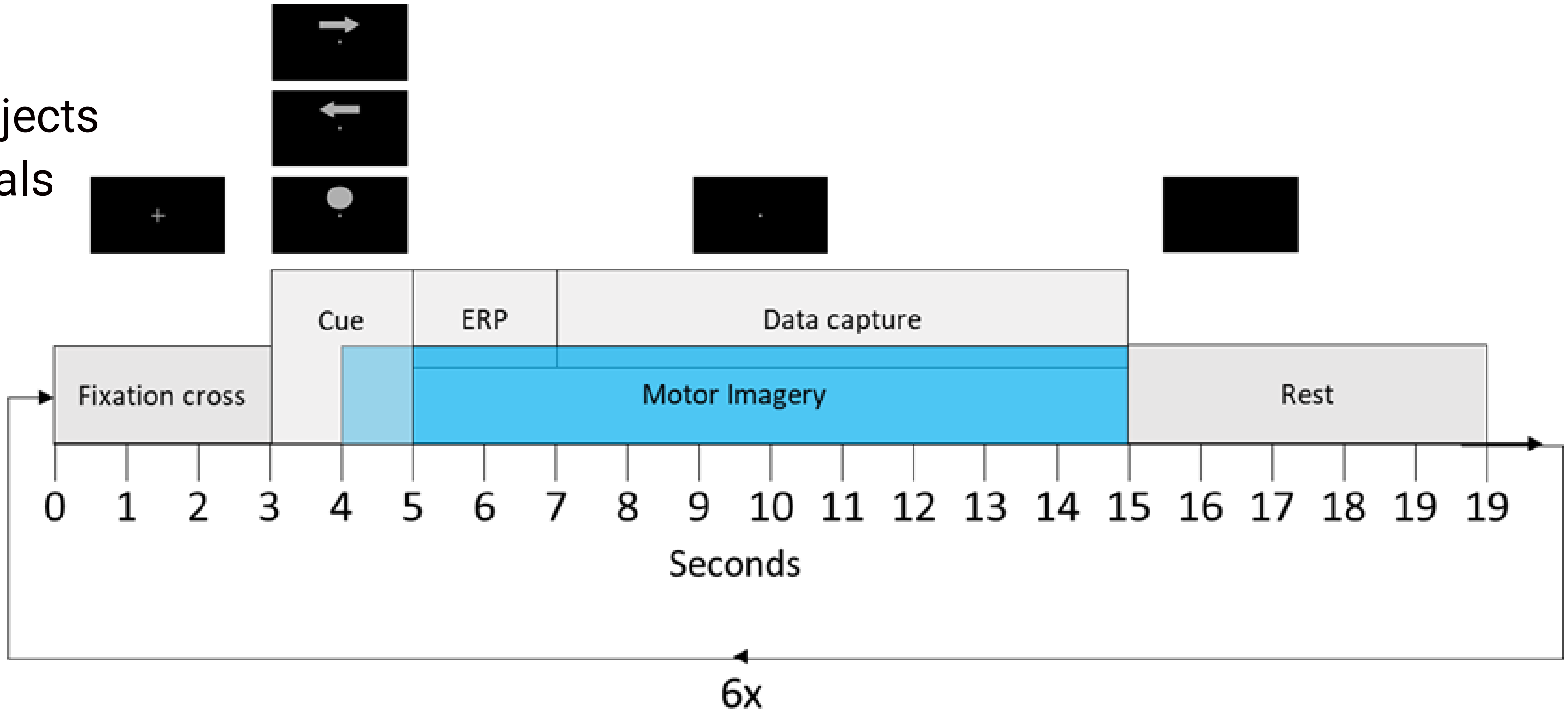
Figure inspired from: Marcel Van Gerven, Jason Farquhar, Rebecca Schaefer, Rutger Vlek, Jeroen Geuze, Anton Nijholt, Nick Ramsey, Pim Haselager, Louis Vuurpijl, Stan Gielen, et al. The brain–computer interface cycle. *Journal of neural engineering*, 6(4):041001, 2009.

The Users

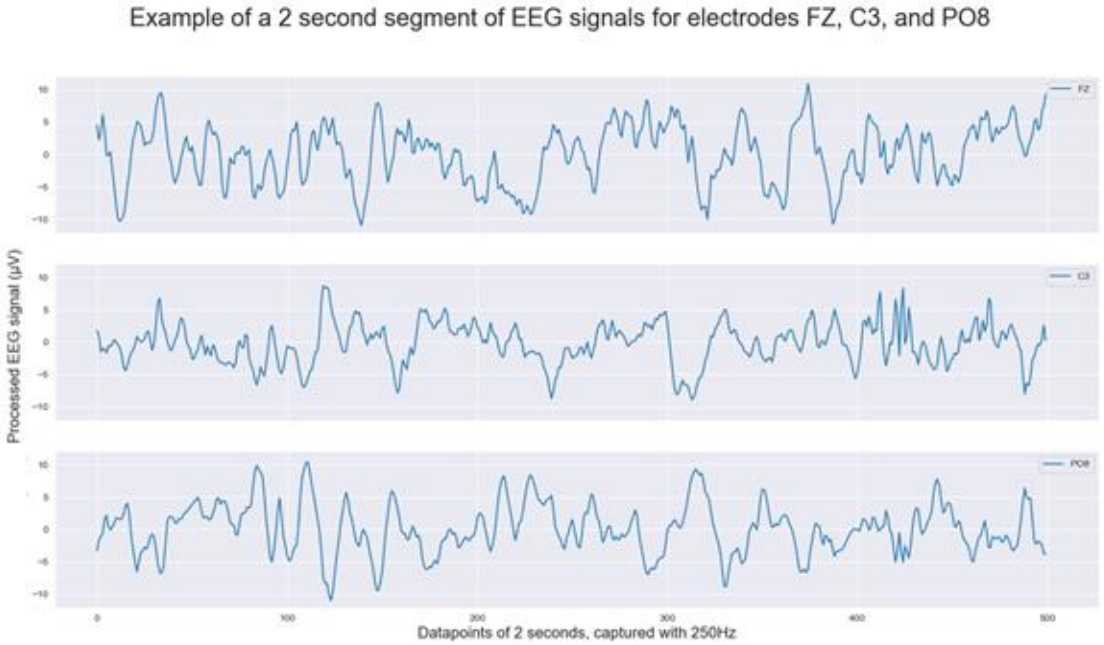
- 10 healthy subjects
- They were tasked to think about moving their left hand, right hand, or just relax
- Without actually moving anything!

Data experiments

- 9 subjects
- 10 trials



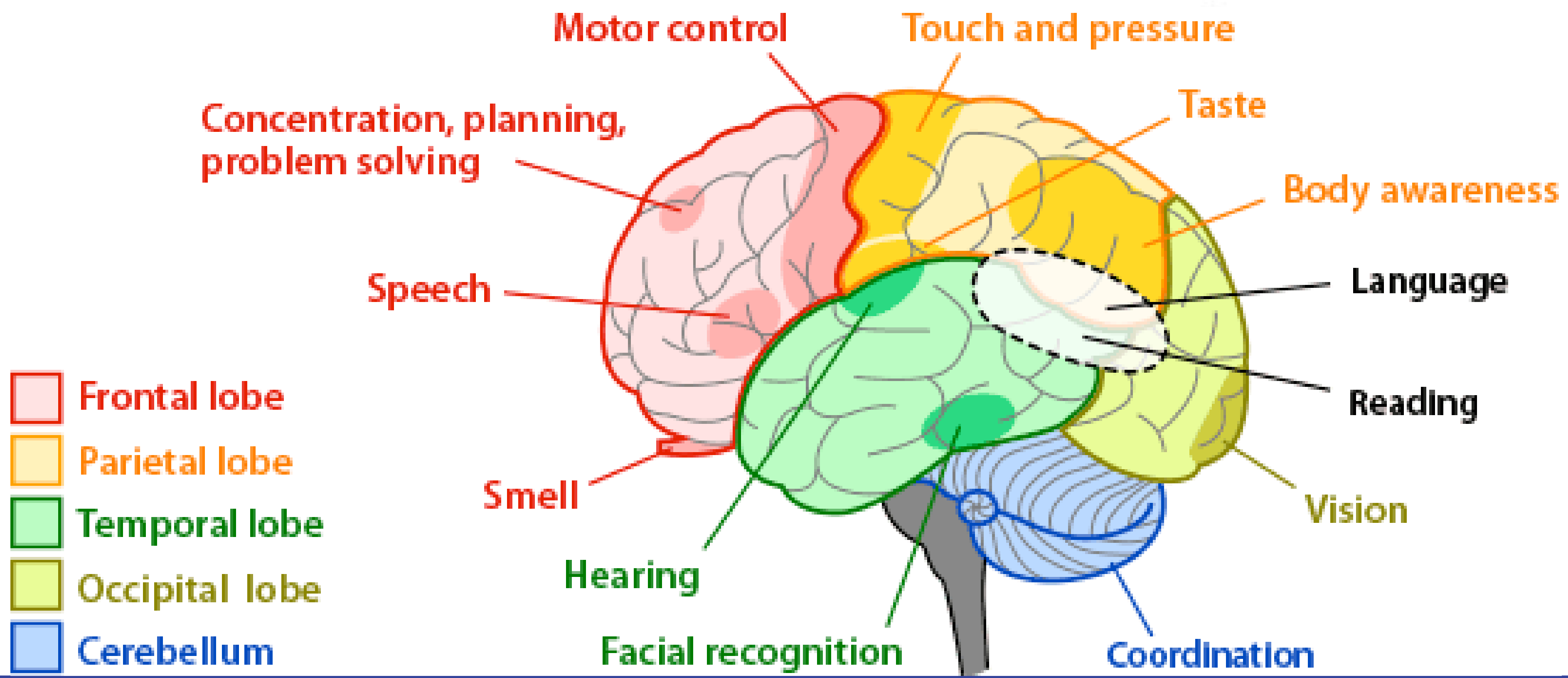
Data collection



Right figure adapted from: Piotr Szczuko. Real and imaginary motion classification based on rough set analysis of eeg signals for multimedia applications. *Multimedia Tools and Applications*, 76(24):25697–25711, 2017.

```
from pylsl import StreamInlet, resolve_stream
import pandas as pd

# initialize the streaming layer
finished = False
streams = resolve_stream()
inlet = StreamInlet(streams[0])

# initialize the colomns of your data and your dictionary to capture the data.
columns=['Time','FZ', 'C3', 'CZ', 'C4', 'PZ', 'PO7', 'OZ', 'PO8','AccX','AccY','AccZ',
'Gyro1','Gyro2','Gyro3', 'Battery','Counter','Validation']
data_dict = dict((k, []) for k in columns)

while not finished:
   # get the streamed data. Columns of sample are equal to the columns variable, only th
   # concatenate timestamp and data in 1 list
   data, timestamp = inlet.pull_sample()
   all_data = [timestamp] + data

   # updating data dictionary with newly transmitted samples
   i = 0
   for key in list(data.keys()):
      data_dict[key].append(all_data[i])
      i = i + 1

   # data is collected at 250 Hz. Let's stop data collection after 60 seconds. Meaning w
   if len(data_dict['Time']) >= 250*60:
      finished = True
```

Why did we use the g.tec Unicorn Hybrid Black?



The BCI cycle

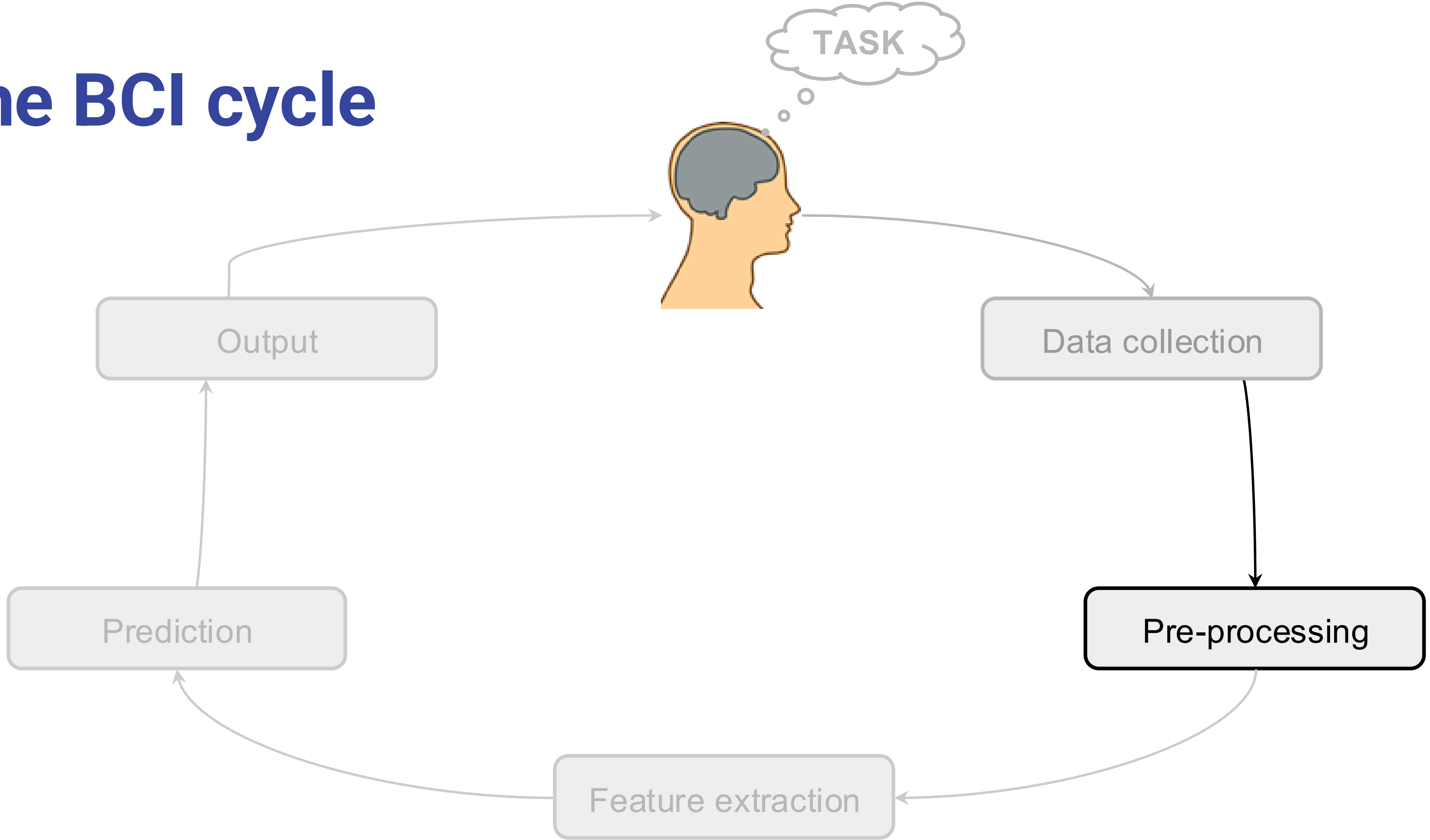


Figure inspired from: Marcel Van Gerven, Jason Farquhar, Rebecca Schaefer, Rutger Vlek, Jeroen Geuze, Anton Nijholt, Nick Ramsey, Pim Haselager, Louis Vuurpijl, Stan Gielen, et al. The brain–computer interface cycle. *Journal of neural engineering*, 6(4):041001, 2009.

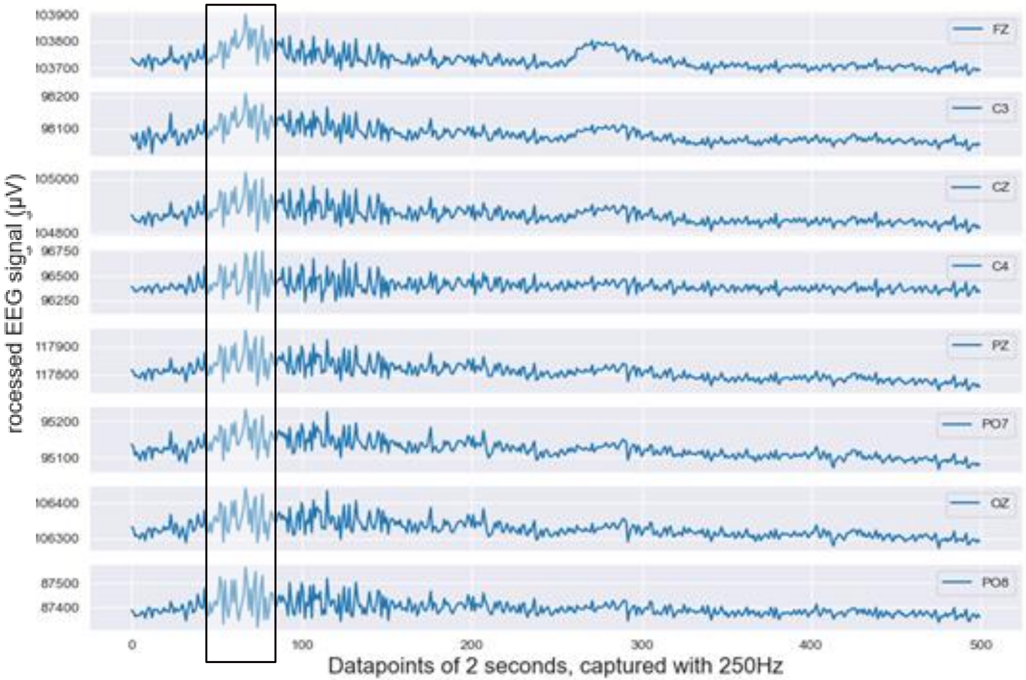
Pre-processing

- Notch filter
  - Removing power line noise at 50Hz
  - Built into the g.tec software

```
b_notch, a_notch = signal.iirnotch(50, 30, sampling_frequency)
for column in curr_segment.columns:
    curr_segment.loc[:, column] = signal.filtfilt(
        b_notch, a_notch, curr_segment.loc[:, column]
    )
```

Pre-processing

- Outlier rejection
  - Artifact due to movement
    - Based on amplitude




```
for i, j in curr_segment.iterrows():
    if (abs(j - np.mean(j)) > 125).any():
        outlier += 1
```

Pre-processing

- Common Average Referencing
  - Reducing noise present over all channels
  - For each timepoint, subtract mean of all channels

```
curr_segment -= curr_segment.mean()
```

The BCI cycle

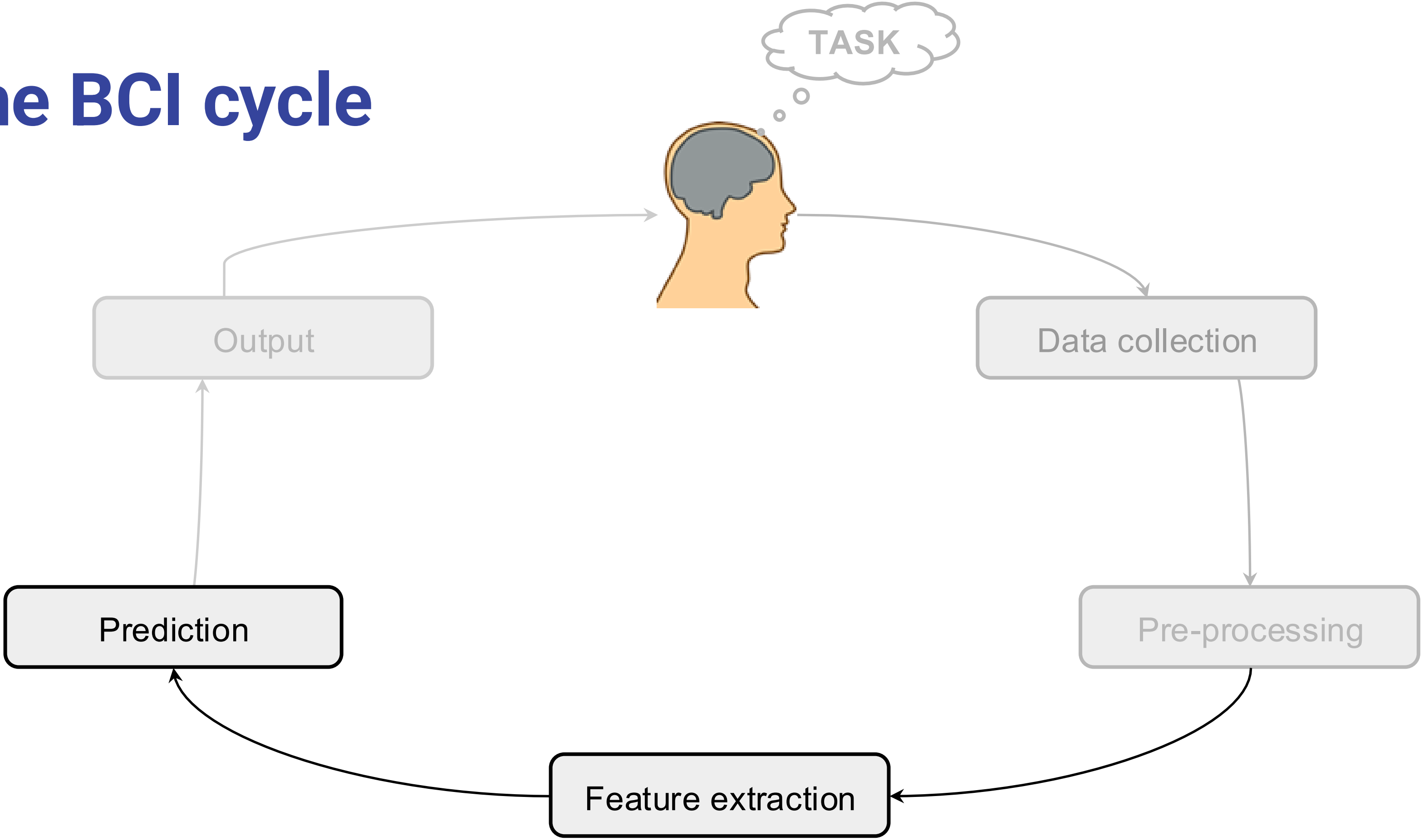
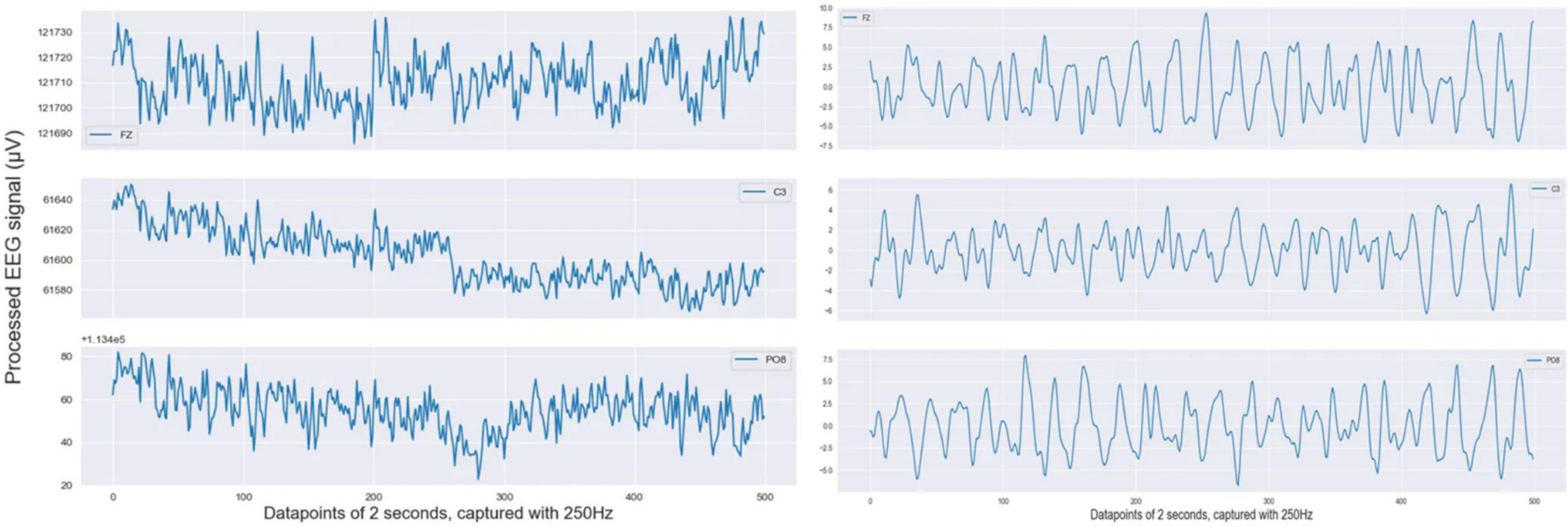
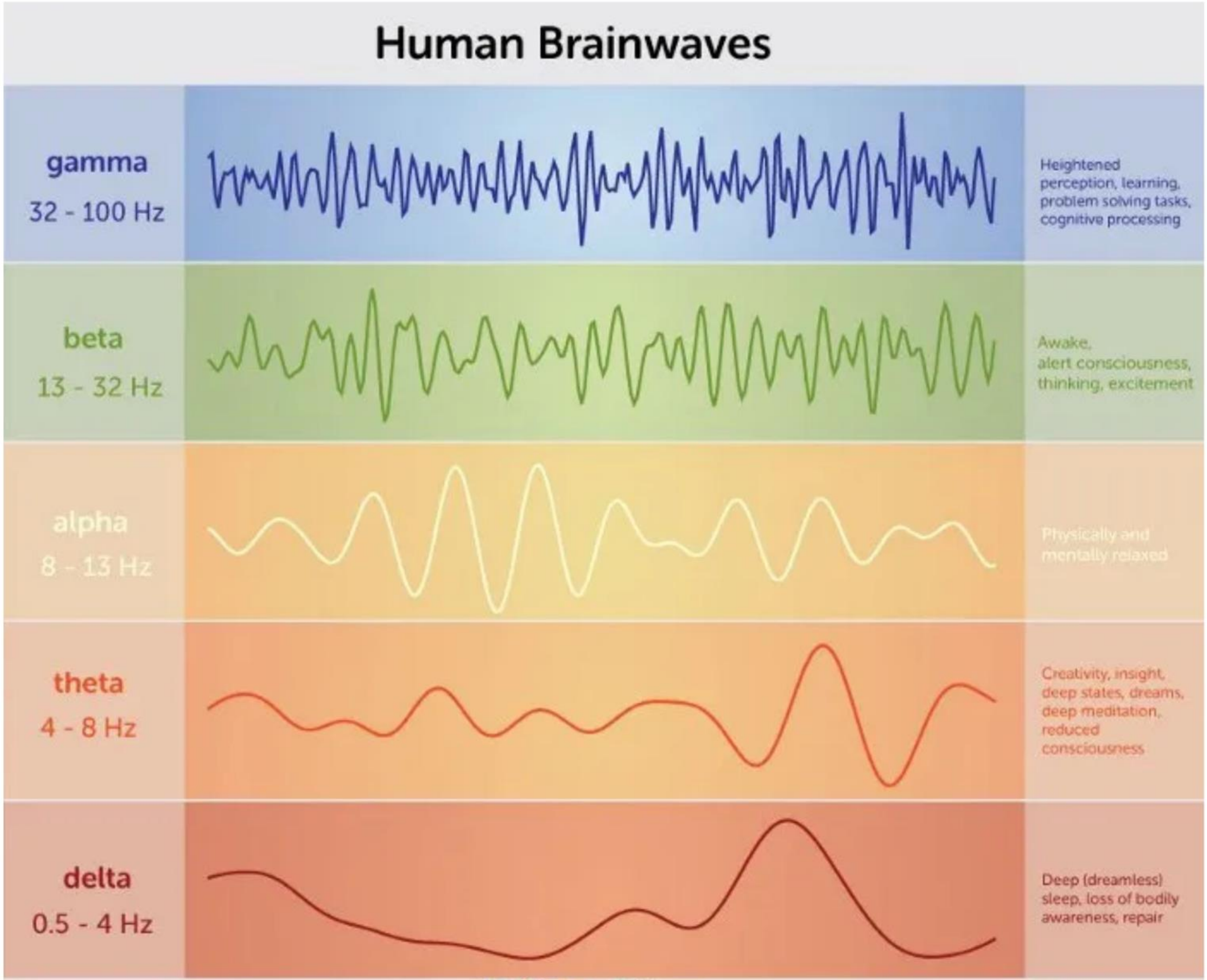


Figure inspired from: Marcel Van Gerven, Jason Farquhar, Rebecca Schaefer, Rutger Vlek, Jeroen Geuze, Anton Nijholt, Nick Ramsey, Pim Haselager, Louis Vuurpijl, Stan Gielen, et al. The brain–computer interface cycle. *Journal of neural engineering*, 6(4):041001, 2009.

Feature extraction

- Spectral filtering
  - Removing low and high frequencies from signal
  - Brain activity during MI mostly revolves around changes in activity between 8 to 30Hz





Human Brainwaves
gamma
32 - 100 Hz
Heightened perception, learning, problem solving tasks, cognitive processing
beta
13 - 32 Hz
Awake, alert consciousness, thinking, excitement
alpha
8 - 13 Hz
Physically and mentally relaxed
theta
4 - 8 Hz
Creativity, insight, deep states, dreams, deep meditation, reduced consciousness
delta
0.5 - 4 Hz
Deep (dreamless) sleep, loss of bodily awareness, repair

Feature extraction

- Spatial filtering
  - EEG signal has poor spatial resolution, and relevant signal can be spread around electrodes
  - Combining information of EEG signals from several channels to isolate relevant signal
    - Gold standard in BCI: **Filter Bank Common Spatial Pattern** [1]
    - More recent popular method: **Riemannian Geometry** [2]

1. Kai Keng Ang, Zheng Yang Chin, Chuanchu Wang, Cuntai Guan, and Haihong Zhang. Filter bank common spatial pattern algorithm on bci competition iv datasets 2a and 2b. *Frontiers in neuroscience*, 6:39, 2012.
2. Alexandre Barachant, St´ephane Bonnet, Marco Congedo, and Christian Jutten. Multiclass brain–computer interface classification by riemannian geometry. *IEEE Transactions on Biomedical Engineering*, 59(4):920–928, 2012.

Prediction

- Using traditional machine learning approaches
  - Support vector machine
  - Linear discriminant analysis
  - Random forest

```python
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis as LDA
from sklearn.ensemble import RandomForestClassifier as RFC
from sklearn.pipeline import Pipeline
from mne.decoding import CSP
from pyriemann.estimation import Covariances
from pyriemann.tangentspace import TangentSpace

csp = Pipeline(steps=[('csp', CSP()),('lda', LDA())])
csp.fit(X_train, y_train)
preds = csp.predict(X_val)

rg = Pipeline(steps=[('cov', Covariances("oas")),('tg', TangentSpace(metric="riemann")),
rg.fit(X_train, y_train)
preds = rg.predict(X_val)
```

Deep transfer learning

- EEG data is highly variable [1]
  - Between subject, but also between days of same subject
  - Model calibration needed each day again
- Deep Transfer Learning (DTL) has faster model calibration (± 10 min) than traditional ML (± 45 min) [2]

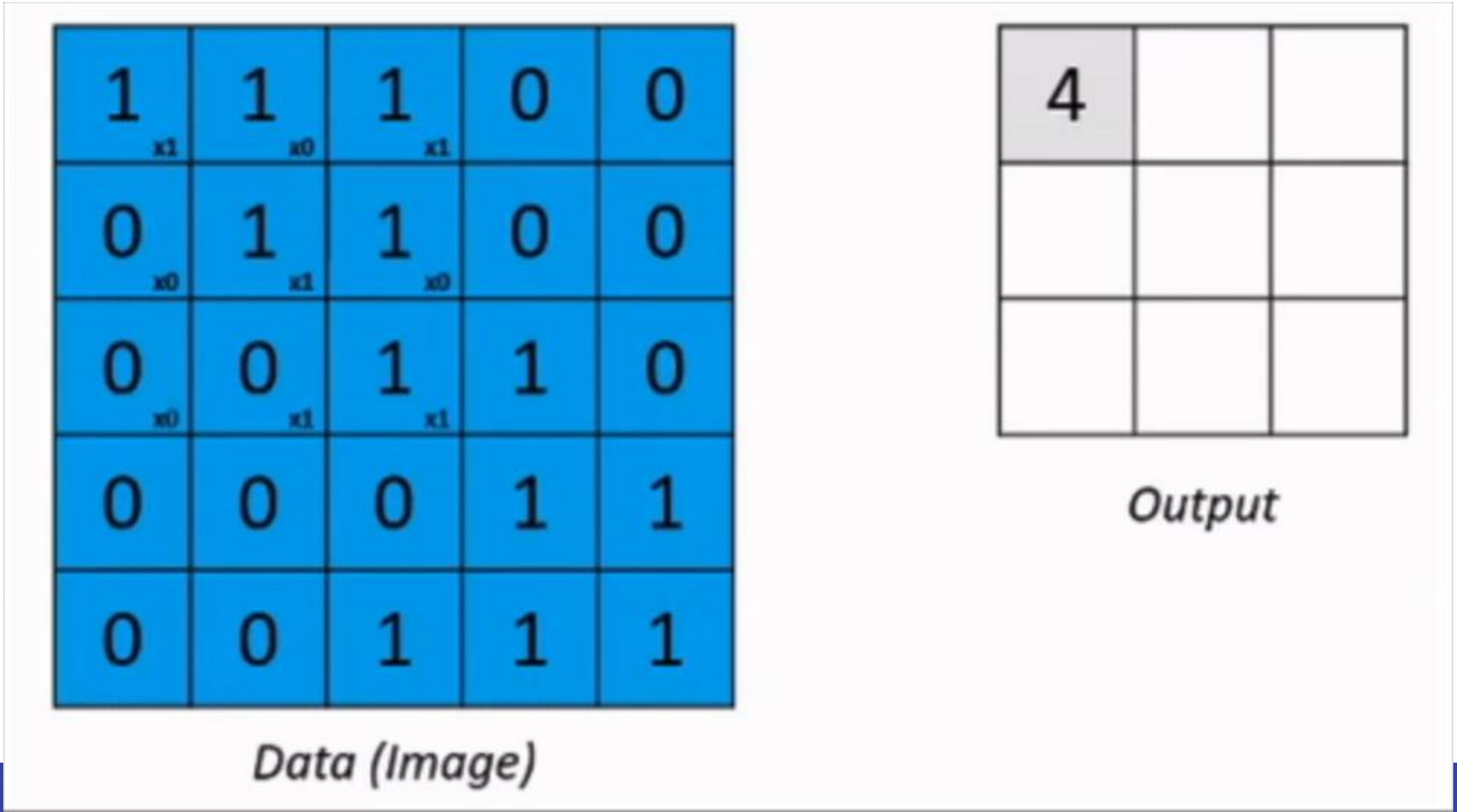
1. Yalda Shahriari, Theresa M Vaughan, LM McCane, Brendan Z Allison, Jonathan R Wolpaw, and Dean J Krusienski. An exploration of bci performance variations in people with amyotrophic lateral sclerosis using longitudinal eeg data. *Journal of neural engineering*, 16(5):056031, 2019
2. Laura Ferrero, Vicente Quiles, Mario Ortiz, Eduardo Iáñez, and José M Azorín. A bmi based on motor imagery and attention for commanding a lower-limb robotic exoskeleton: A case study. *Applied Sciences*, 11(9):4106, 2021.

Deep transfer learning

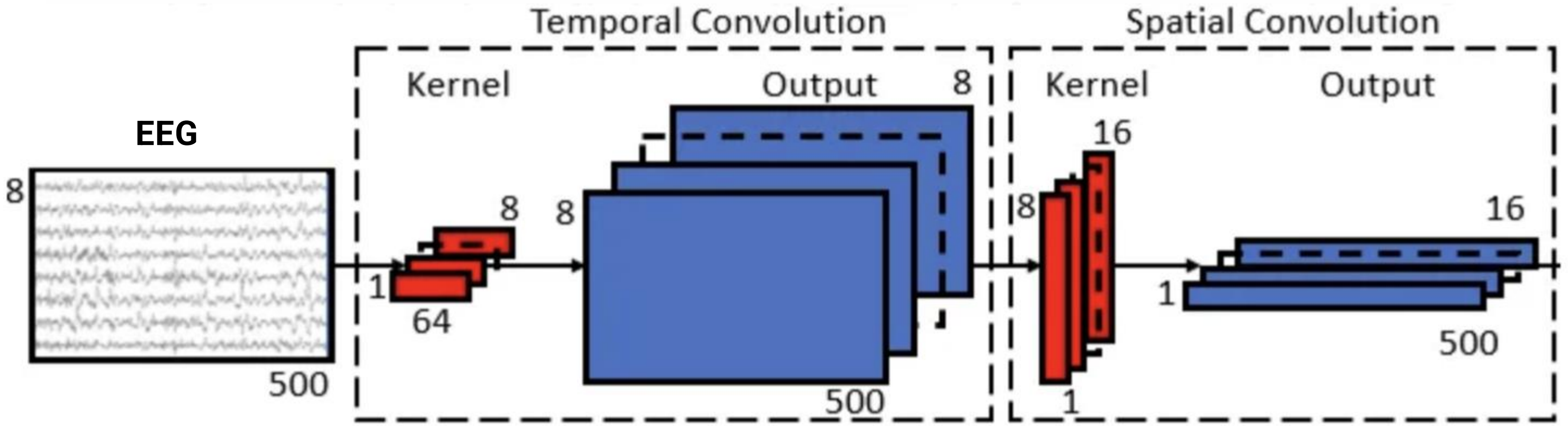
- **DTL:** Decrease calibration time with still good performance
  - Collect data
  - Cross-subject pre-training
  - Fine-tuning on specific subject, starting from pre-trained weights

1. Yalda Shahriari, Theresa M Vaughan, LM McCane, Brendan Z Allison, Jonathan R Wolpaw, and Dean J Krusienski. An exploration of bci performance variations in people with amyotrophic lateral sclerosis using longitudinal eeg data. *Journal of neural engineering*, 16(5):056031, 2019

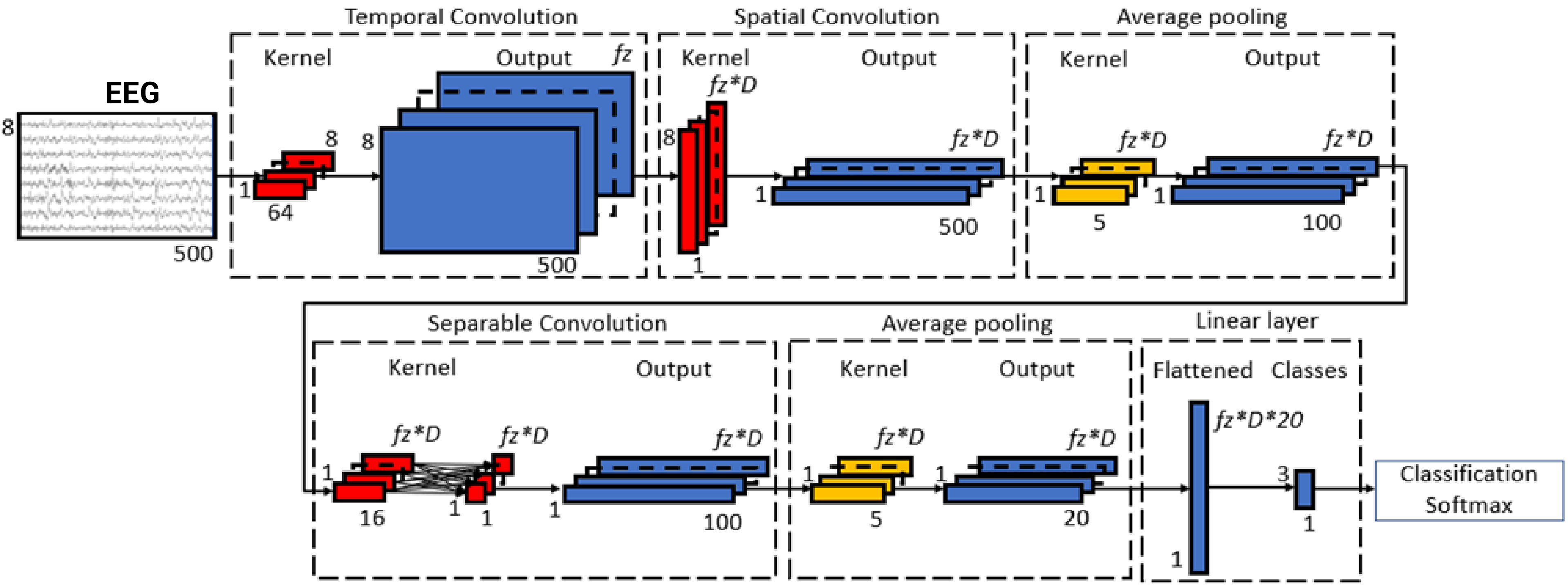
Convolutional Neural Networks



Spatial and Temporal Convolution



End-to-end Deep Learning: EEGNET



Implemented network adapted from: Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces. *Journal of neural engineering*, 15(5):056013, 2018

```python
class EEGNET(nn.Module):
    def __init__(self, receptive_field=64, mean_pool=5, filter_sizing, dropout, D):
        super(EEGNET,self).__init__()
        channel_amount = 8
        num_classes = 3
        self.temporal=nn.Sequential(
            nn.Conv2d(1,filter_sizing,kernel_size=[1,receptive_field],stride=1, bias=Fal
                padding='same'),
            nn.BatchNorm2d(filter_sizing),
        )
        self.spatial=nn.Sequential(
            nn.Conv2d(filter_sizing,filter_sizing*D,kernel_size=[channel_amount,1],bias=
                groups=filter_sizing),
            nn.BatchNorm2d(filter_sizing*D),
            nn.ELU(True),
        )

        self.seperable=nn.Sequential(
            nn.Conv2d(filter_sizing*D,filter_sizing*D,kernel_size=[1,16],\
                padding='same',groups=filter_sizing*D, bias=False),
            nn.Conv2d(filter_sizing*D,filter_sizing*D,kernel_size=[1,1], padding='same',
            nn.BatchNorm2d(filter_sizing*D),
            nn.ELU(True),
        )
        self.avgpool1 = nn.AvgPool2d([1, 5], stride=[1, 5], padding=0)
        self.avgpool2 = nn.AvgPool2d([1, 5], stride=[1, 5], padding=0)
        self.dropout = nn.Dropout(dropout)
        self.view = nn.Sequential(Flatten())

        endsize = 320
        self.fc2 = nn.Linear(endsize, num_classes)
```

```python
    def forward(self,x):
        out = self.temporal(x)
        out = self.spatial(out)
        out = self.avgpool1(out)
        out = self.dropout(out)
        out = self.seperable(out)
        out = self.avgpool2(out)
        out = self.dropout(out)
        out = out.view(out.size(0), -1)
        prediction = self.fc2(out)
        return prediction
```

Offline analysis

- Train FBCSP and RG from scratch
- Fine-tune pre-trained DL model
- **DTL highest performance**

Table 2: Average classification accuracy ± σ of 5 runs on test set, using 2 trials a training set, 1 trial as validation set, and 5 trials as test set. Highest average accurac ± σ for each subject is highlighted in bold.

Subject	Pipeline	Accuracy ± σ
X01	RG	44.7 ± 0.5
	CSP	**45.3 ± 0.8**
	DTL	39.9 ± 1.2
X02	RG	70.6 ± 1.0
	CSP	68.6 ± 1.2
	DTL	**78.8 ± 2.6**
X03	RG	39.9 ± 1.1
	CSP	41.5 ± 1.6
	DTL	**50.9 ± 2.4**
X04	RG	64.9 ± 1.7
	CSP	69.9 ± 1.9
	DTL	**79.7 ± 3.3**
X05	RG	79.6 ± 0.6
	CSP	77.0 ± 1.1
	DTL	**86.9 ± 1.0**
X06	RG	53.9 ± 0.3
	CSP	55.0 ± 0.6
	DTL	**57.9 ± 6.4**
X07	**RG**	**38.1± 0.6**
	CSP	35.8 ± 0.6
	DTL	36.0 ± 2.3
X08	RG	49.0 ± 0.3
	CSP	**51.2 ± 0.9**
	DTL	50.9 ± 3.0
X09	RG	41.4 ± 0.5
	CSP	39.1 ± 0.7
	DTL	**41.7 ± 2.9**
Average	RG	53.3 ± 14.3
	CSP	54.0 ± 13.8
	DTL	**58.1 ± 18.0**

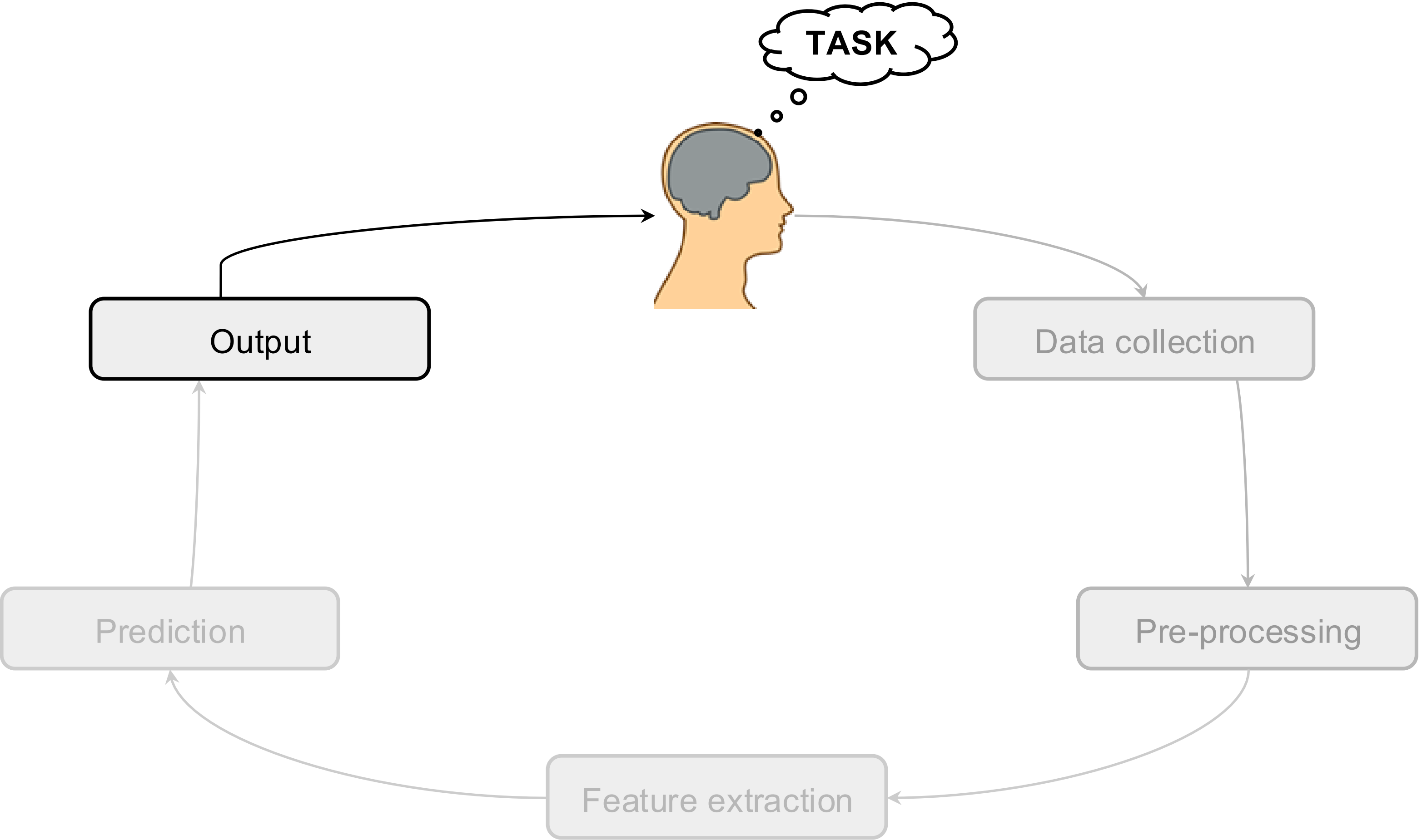


Figure inspired from: Marcel Van Gerven, Jason Farquhar, Rebecca Schaefer, Rutger Vlek, Jeroen Geuze, Anton Nijholt, Nick Ramsey, Pim Haselager, Louis Vuurpijl, Stan Gielen, et al. The brain–computer interface cycle. *Journal of neural engineering*, 6(4):041001, 2009.

Output

- Control a game, exoskeleton
- Finishing the circle:
  - Output influences next thoughts and goals of user!

Now we made the full pipeline...

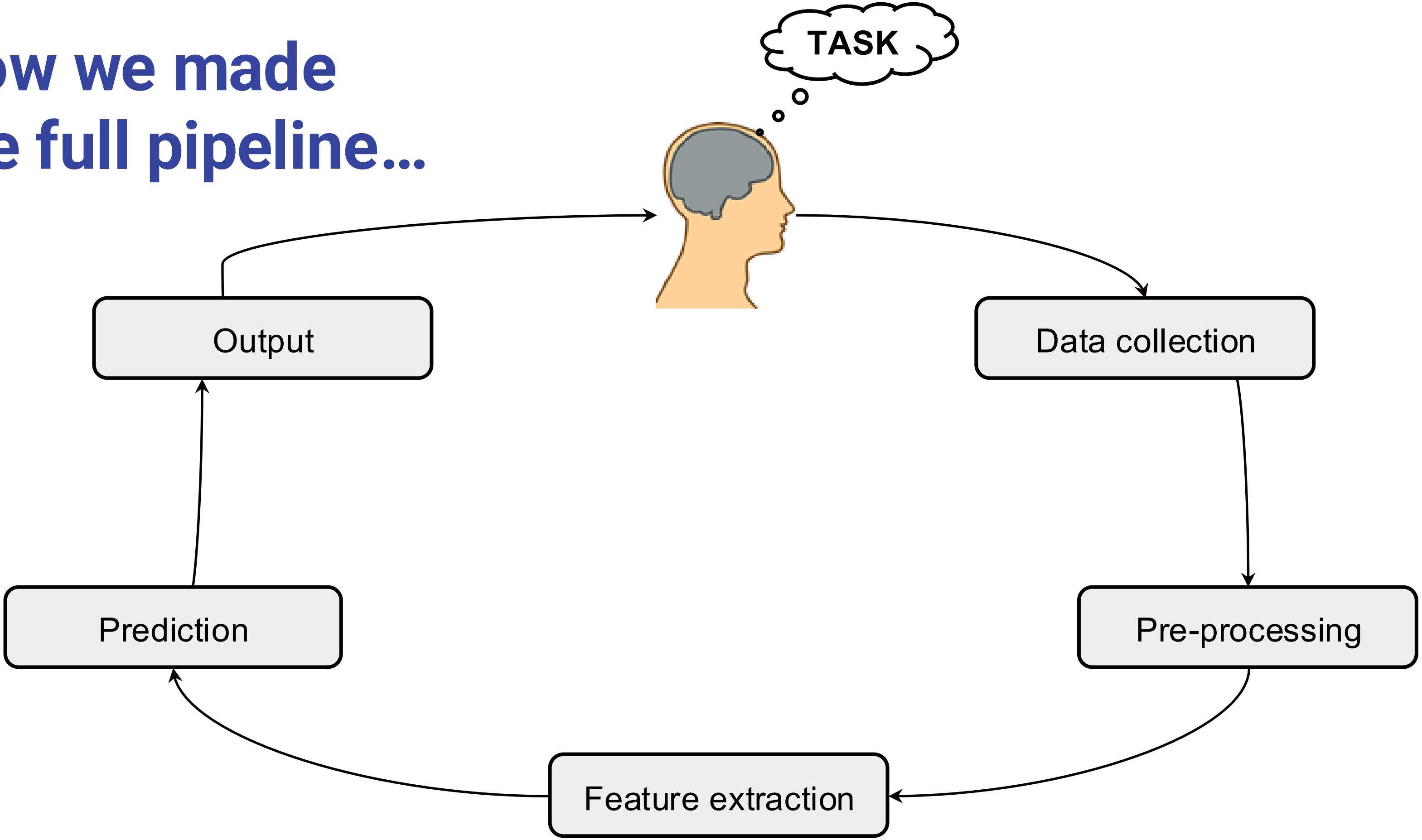


Figure inspired from: Marcel Van Gerven, Jason Farquhar, Rebecca Schaefer, Rutger Vlek, Jeroen Geuze, Anton Nijholt, Nick Ramsey, Pim Haselager, Louis Vuurpijl, Stan Gielen, et al. The brain–computer interface cycle. *Journal of neural engineering*, 6(4):041001, 2009.

Time for real-time experiments!

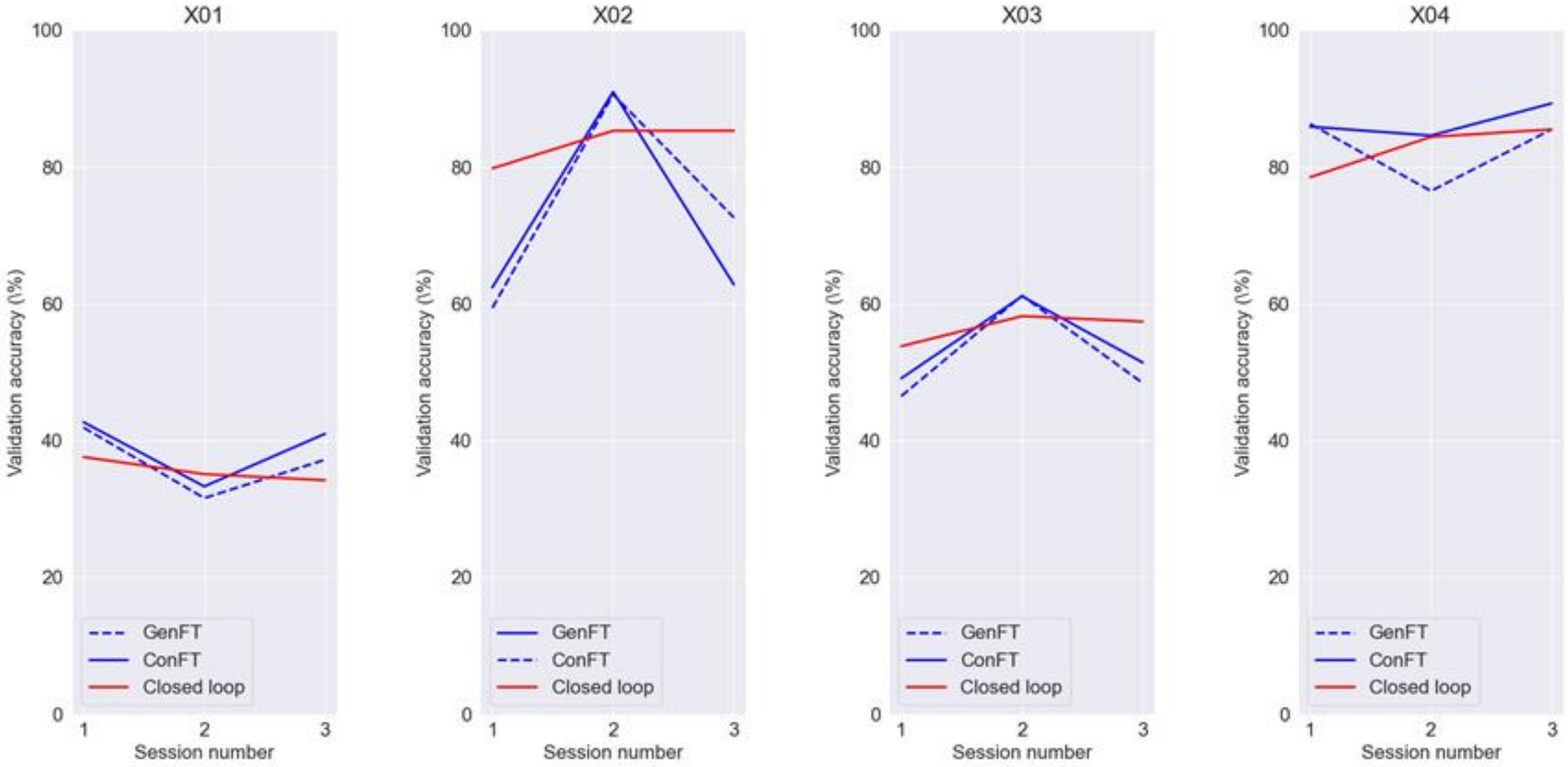
3 days of:

- 3 data-collection trials for fine-tuning
- 5 real-time experiments
  - Dot movement from model predictions
- 3 game trials
  - Game control from model predictions

Ready?

Results real-time

Validation accuracy for open-loop fine-tuning with GenFT or ConFT, together with average closed-loop accuracy



Playing a game

- Possible for subjects X02, X03, and X04
- Due to low performance, not effectively for X01


Score : 6
SAMSUNG

A step towards cheaper and more practical BCIs

- Good performance, but not for all subjects
- Low costs
- Low preparation time
- Low calibration time

Implementation with exoskeletons!

RESEARCH **Open Access**

Check for updates

Brain–machine interface based on deep learning to control asynchronously a lower-limb robotic exoskeleton: a case-of-study

Laura Ferrero[1,2,3,4,6*], Paula Soriano-Segura[1,2,3], Jacobo Navarro[4,5,6], Oscar Jones[4,6], Mario Ortiz[1,2,3], Eduardo Iáñez[1,2,3], José M. Azorín[1,2,3,7†] and José L. Contreras-Vidal[4,6†]

BMIlab
Brain-Machine Interface Systems Lab

Thank you!

I Built A Brain Computer Interface To Play Space Invaders Using Thoughts

And I will explain you how, including code examples in Python

Tim de Boer 15 min read · Sep 13, 2022

Connect with me on LinkedIn: