



TECHNISCHE UNIVERSITÄT MÜNCHEN

Department of Informatics

Chair of Neuroscientific System Theory

Master's Thesis: Robotics, Cognition, Intelligence



Decoding of 3D Reach and Grasp Movements from  
Non-Invasive EEG Signals using SpiNNaker Neuromorphic  
Hardware

Leonard Rychly





TECHNISCHE UNIVERSITÄT MÜNCHEN

Department of Informatics

Chair of Neuroscientific System Theory

Master's Thesis: Robotics, Cognition, Intelligence

**Decoding of 3D Reach and Grasp Movements from  
Non-Invasive EEG Signals using SpiNNaker Neuromorphic  
Hardware**

**Dekodierung 3D Bewegungen aus nicht invasiven EEG  
Signalen mittels SpiNNaker Neuromorpher Hardware**

Supervisor: Prof. Dr. sc. nat. Jörg Conradt  
Advisors: Dipl.-Ing. Zied Tayeb and Emeç Erçelik, M.Sc.  
Final Submission: 24.07.2017

Leonard Rychly



I confirm that this master's thesis is my own work and I have documented all sources and material used.



## Abstract

Non-invasive, Electroencephalography (EEG) based brain-computer interfaces (BCIs) on motor imagery movements translate the subject's motor intention into control signals through classifying the EEG patterns caused by different imagination tasks, e.g. hand movements. This type of BCI has been widely studied and is already used as an alternate mode of communication and environmental control for the disabled, such as patients suffering from amyotrophic lateral sclerosis, brainstem stroke and spinal cord injury. Together with recent advancements in neuromorphic computing, which allow real-time, low power implementations of large scale spiking models for data processing, BCI applications could profit from this symbiosis.

Inspired from the architecture of the insect's olfactory system, we further advance and implement a spiking neural network model to decode and predict imaginary movements from EEG signals. The network runs on *SpiNNaker*, a neuromorphic hardware platform containing 4 chips with 64 cores. To improve the network's performance a reward based *Spike-Time Dependent Plasticity* (STDP) learning algorithm is implemented and different techniques, as *Homeostasis* and batch-learning for training the network are tested.

Aiming to find EEG signal components that are stable across imagery movements of the same type we further implemented and analyzed multiple feature extraction techniques as calculating sub-band power, logarithmic band power and motor imagery specific characteristics from the signal. Additionally we tested the Discrete Wavelet Transform to decompose the EEG data while preserving information from the frequency as well as the time domain. Of all approaches to extract stable characteristics the sub-band power features proved to yield the best results. Overall the spiking neural network reaches with the STDP learning algorithm a mean accuracy of 73.4% only falls short by an average of 4.12% in classification rate to state of the art machine learning algorithms. This shows SNNs present a valid alternative to classical machine learning algorithms deployed in BCIs.

I would like to thank both of my supervisors, Zied Tayeb and Emeç Erçelik for guiding and inspiring me during working on this thesis. Special thanks go to my dear girlfriend for being a big support throughout every step and to my sister for proofreading my thesis. I also want to thank my family for always having my back during the many years of studies.

# Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
<b>2</b>	<b>Basics</b>	<b>7</b>
2.1	EEG based Brain-Computer Interfaces for Motor Imagery . . . . .	7
2.1.1	Underlying Brain Processes of Motor Imagery . . . . .	8
2.1.2	EEG Data Acquisition of Motor Imagery . . . . .	9
2.2	Neuromorphic Hardware and Algorithms . . . . .	10
2.2.1	Neuromorphic Computing . . . . .	10
2.2.2	Neuromorphic Hardware . . . . .	12
<b>3</b>	<b>Overview on the State of the Art of EEG based Brain-Machine Interfaces</b>	<b>15</b>
3.1	Review: EEG based Brain-Computer Interfaces . . . . .	15
3.2	Decoding Methods of Motor Imagery from EEG Data . . . . .	16
3.3	Spiking Neural Network based on the Insect's Olfactory System . . . . .	19
<b>4</b>	<b>Solution Design</b>	<b>23</b>
4.1	Design . . . . .	23
4.2	Experimental Data . . . . .	23
<b>5</b>	<b>Implementation</b>	<b>25</b>
5.1	Data Preprocessing . . . . .	25
5.2	Feature Extraction Methods . . . . .	25
5.2.1	Event Related Desynchronization and Synchronization . . . . .	26
5.2.2	Discrete Wavelet Transform . . . . .	29
5.2.3	Feature Processing: Cross-Trial Encoding . . . . .	31
5.2.4	Results: Feature Extraction . . . . .	32
5.3	Spiking Neural Network Implementation . . . . .	33
5.3.1	Feature Encoding . . . . .	33
5.3.2	Network Implementation . . . . .	34
5.3.3	Network Evaluation . . . . .	35
5.3.4	Results: Spiking Neural Network . . . . .	41
<b>6</b>	<b>Discussion</b>	<b>43</b>
<b>7</b>	<b>Conclusion</b>	<b>45</b>
<b>List of Figures</b>		<b>47</b>
<b>Bibliography</b>		<b>49</b>



# Chapter 1

## Introduction

The way humans interact with computers has continuously evolved over the last centuries. Interfaces, such as keyboards and mouses are being more and more replaced by modern technologies like touchscreens, voice and gesture control applications. During the next iteration of human-computer interaction voice and gesture controls could be superseded by solely using your thoughts to control the devices. Direct brain-computer interfaces (BCIs) or more generally called brain-machine interfaces (BMIs) are a rapidly expanding branch of Neuroscience and Neurorobotics research that is already being applied for medical treatment and could further enable a new dimension of functionality to traditional human-computer interaction. Today BCIs are already being used as an alternate mode of communication and environmental control for the disabled, such as patients suffering from the locked-in syndrome, amyotrophic lateral sclerosis, brainstem stroke and spinal cord injury. BCIs are also emerging to replace lost sensory functions in the form of cochlea and retinal implants.

In this work we focus on non-invasive, Electroencephalogram (EEG) based BCIs for decoding motor imagery right and left hand movements to control a robotic arm or prosthesis. The core part of any BCI is the decoding of neural signals and determining the underlying intend causing a specific pattern in the signal, as depicted in figure 1.1. For the task of classification we implement and further develop an

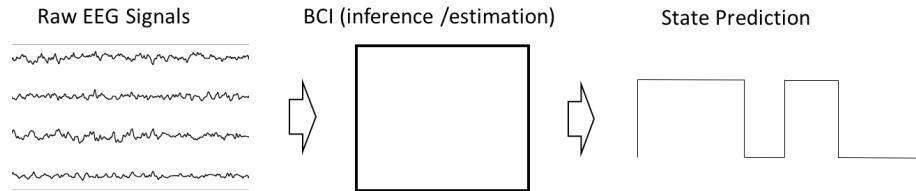


Figure 1.1: BCI schematic

approach presented by Zied et al. [18], who applied a spiking neural network (SNN) based on the insect's olfactory system [19] to decode motor imagery movements from

EEG signals. This network runs on the neuromorphic computing platform *SpiNNaker* [4], which allows the deployment of highly parallel and low power simulations of large SNNs. The corresponding research question of this thesis includes how to enhance the overall classification accuracy of this SNN on motor imagery and how to modify the SNN to optimize its performance. Further this spike based processing approach of the SNN is compared to state of the art machine learning algorithms to evaluate the results. Since the overall classification accuracy also highly depends on the data provided different methods for extracting motor imagery features from EEG data are examined.

After explaining the underlying theory and reviewing recent publications in the corresponding fields, the main focus will be on the implementation and evaluation of the SNN algorithm and the different methods of feature extraction.

# Chapter 2

## Basics

In this chapter, to give a general overview about EEG based motor imagery BCIs, the basic principles of BCIs are explained. First, the different BCI types, the underlying processes of motor imagery and the procedure of recording motor imagery EEG data are introduced. Since the SNN used in this work runs on the *SpiNNaker* hardware a brief introduction to neuromorphic computing, its motivation and the used neuromorphic hardware are presented.

### 2.1 EEG based Brain-Computer Interfaces for Motor Imagery

A BCI is a method of communication based on neural activity generated by the brain that does not depend on the brain's normal output channels of peripheral nerves and muscles. The neural activity used in BCIs can be recorded using invasive or noninvasive techniques. For the BCI described in this work noninvasive EEG measurements were used to analyze the brain activity. The goal of BCIs generally is to either determine a person's intent or mental state by monitoring the user's brain activity, or to provide a new channel of output for the brain that requires voluntary adaptive control by the user. For the specific case of using a BCI for decoding motor imagery, subjects perform volitional mental tasks, that evoke changes in their EEG rhythms. The EEG bands employed in such measurements are slow cortical potentials, in the range of alpha (8 – 12 Hz), beta (12 – 25 Hz) and gamma (25 – 70 Hz) waves.

**BCI Types:** There are three major types of brain-computer interfaces that each serve a different purpose in the BCI field. The system described in this work is an *active* BCI, that derives its output from brain activity which is directly consciously controlled by the user and independently from external stimuli, for controlling an application. In contrast to active BCI systems, *reactive* BCI systems derive their output from brain activity arising in reaction to external stimulation, which is in-

directly modulated by the user for controlling an application. The third BCI type are *passive* BCI systems, which derive their output from arbitrary brain activity without the purpose of voluntary control.

### 2.1.1 Underlying Brain Processes of Motor Imagery

Since all BCIs have to operate on observable effects of the brain activity, only large scale neural dynamics in the order of 50.000 and more near-synchronous firing neurons can be detected using the common 10-20 system [12], described in section 2.1.2. Such measurable events are often cascades of related neural processes either triggered by external events (e.g. in perception), by internal events such as a sudden internal "aha!" or if neural populations enter a synchronized steady-state firing pattern (e.g. idle oscillations). While recording EEG signals from consciously controlled brain activity, it is possible to often not directly detect the root cause because electromagnetic fields can cancel each other out or populations could be widely scattered and the resulting electromagnetic field is then too weak to be detected. Therefore often spatially compact populations are observed because they are more likely to have coordinated firing times and therefore it is not certain that the observable signals directly correspond to the consciously controlled brain activity. To increase the likelihood of detecting the right patterns corresponding to the executed imaginary movement the spatial and temporal characteristics are used to determine the correct pattern.

**Spatial Characteristics:** Spatial characteristics are important to determine where to place the electrodes. Since motor imagery mainly occurs in the sensorimotor cortex, electrodes placed there have the best chance to detect mental patterns corresponding to the motor imagery, e.g left and right hand movements. Figure 2.1 shows the different areas of the cortex and their functional mapping.

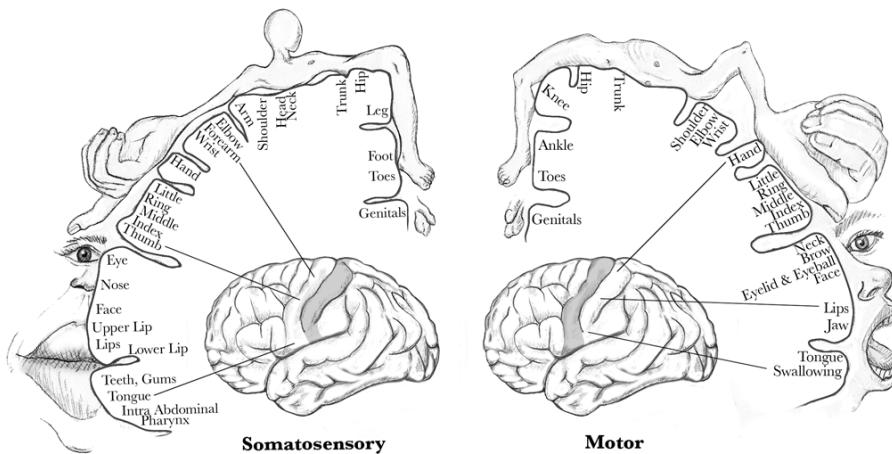


Figure 2.1: The functional mapping of the somatosensory and motor cortex [13]

**Temporal Characteristics:** Temporal characteristics of an EEG signal can be further subdivided into event-related potentials and oscillatory processes, which both aim at creating recognizable and stable patterns from the noisy raw recordings. EEG signals are permeated by oscillatory processes of different frequency bands. For detecting patterns of motor imagery the alpha (8 – 12 Hz) and beta (12 – 25 Hz) bands provide a proven range in which the near-synchronous firing patterns of the neurons in motor cortex can be detected. Because the brain activity can vary between different subjects and recording times the specific frequencies of oscillatory processes can also differ between the different subjects. To create subject and recording invariant representations of mental processes event-related potentials are calculated by averaging the EEG activity related to individual events. Hence event related potentials should ideally be transferable between different sessions and subjects.

### 2.1.2 EEG Data Acquisition of Motor Imagery

Motor imagery or the mental simulation of an action generates patterns of mental activation spanning over both hemispheres of the brain. With electrodes placed on the positions provided by the 10-20 system shown in figure 2.2 the electrical activity of the brain is monitored. For recording motor imagery signals of left and right hand movements only  $C_3$ ,  $C_4$  and  $C_Z$  electrodes in figure 2.2-B are sufficient since they cover most parts of the motor cortex [1].

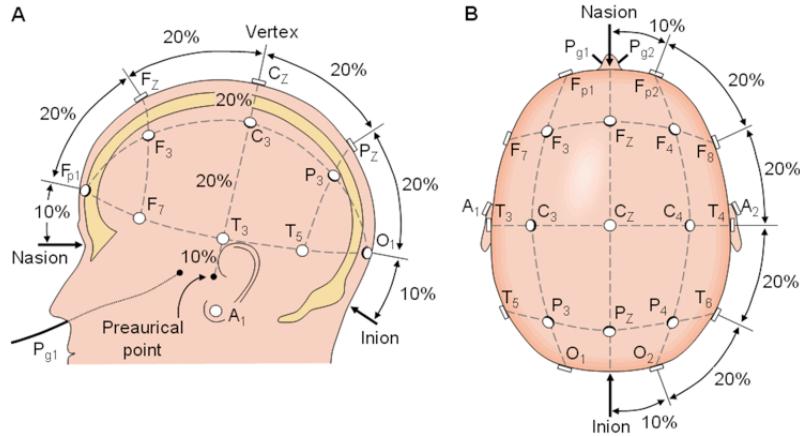


Figure 2.2: The electrode positions provided by the 10-20 System [2]

For the purpose of further developing a BCI that detects left and right hand movements we used various data sets provided by the University of Graz [3] and further covered in section 4.2. The data consists of several recording sessions where one session contains typically 120 trials like the one shown in figure 2.3. Each trial starts with a rest period where no motor imagery takes place. After a few seconds of rest a random selected cue is presented to the subject. Shortly after the cue the subject

has to imagine to either move its left or right hand depending on the cue. After executing the motor imagery, for example for four seconds, a rest period and the preparation for the next trial follow. Since motor imagery is very exhausting no more than 120 trials can be conducted without loss of quality.

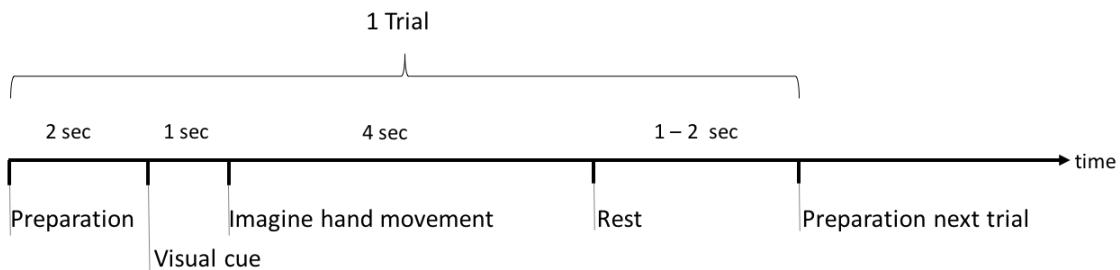


Figure 2.3: One trial of a typical recording session

## 2.2 Neuromorphic Hardware and Algorithms

### 2.2.1 Neuromorphic Computing

Neuromorphic computing covers a large range of approaches of neural information processing. It refers to a variety of brain-inspired hardware systems and models. This biologically inspired approach has created highly parallel architectures containing heavily connected artificial neurons and synapses that can be used to model neuroscience theories as well as to solve challenging machine learning problems. The aim of the neuromorphic computing field is to create a brain-like ability to learn and adapt but until this goal is reached many technical challenges have still to be solved. Some major issues today are that accurate neuroscience models of how the brain works, materials and architectures that could support these models do not exist yet [25].

**Motivation:** The idea of using custom hardware to implement neurally inspired systems already dates back to the 1950's when two of the most influential pioneers in computer science, John von Neumann [26] and Alan Turing [27] started discussing concepts of brain-inspired machines. And, as figure 2.4 shows, more people than ever are recently working on neuromorphic hardware or model solutions.

Early developers mainly concentrated their work on increasing the computing speed [32, 33, 34, 35] and the high degrees of parallelism [28, 29, 30, 31] which hence resulted in vastly improved real-time performance. These devices tended to be able of running neural network computations faster than implementations on von Neumann computers for applications such as real-time control [36], real-time digital image reconstruction [37], and autonomous robot control [38]. They further started

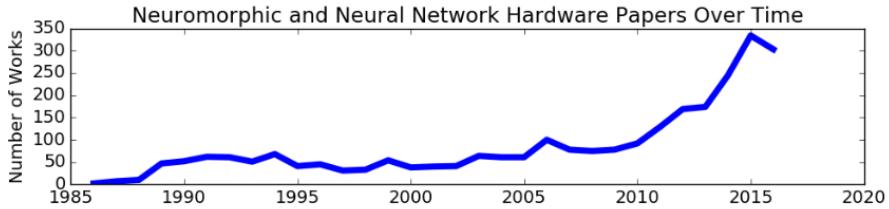


Figure 2.4: Publications of neuromorphic and neural network hardware over time [25]

to recognize that neural networks could be a natural model for hardware implementations because of their fault tolerance, both in the massively parallel representation and in potential adaptation or self-healing capabilities [28, 39]. These were and continue to be relevant characteristics for developing new hardware implementations, where device and process variation can lead to imperfect fabricated devices.

Today the most popular motivation presented in the literature of neuromorphic systems is the emphasis on their potential for very low power consumption. This desire arises when modern hardware is compared to the human brain, which only requires approximately 20 watts of power while performing extremely complex computations. Similarly to the focus on power efficiency, the possibility of creating devices capable of executing neural network computations in a small form factor also became a major motivation in recent years of neuromorphic research.

**Learning in Spiking Neural Networks:** The dynamics implemented by a neural network are defined by its connectivity and the strength of its synaptic connections. Learning implies the adaption of the latter ones through repeated optimization. Which learning algorithm is best suited for a specific task depends on three major characteristics of the model. Certain algorithms are specific to certain network topologies, neuron models, or other network model characteristics and hence the selected models limit the choice for an algorithm. Beyond that, a second issue is whether training or learning for a system should be implemented on chip or if networks should be trained off-chip and then transferred to the neuromorphic system. This also determines the system’s online learning capabilities, since only chips that provide on-chip learning also enable online learning. A third discrimination that has to be made is whether the network is trained in an unsupervised or supervised fashion.

For the SNN in our work we implement a variation of the *Spike-Timing Dependent Plasticity* (STDP) rule, one of the most prominent and intensively studied forms of local neural learning rules. STDP is originally an unsupervised learning algorithm as it models changes of synaptic connection strength, also known as weights, based on a causal relation between spikes occurring within a defined time window. This process is shown in figure 2.5. If a presynaptic spike is followed by a postsynaptic one, the relationship is regarded as causal and the synapse is strengthened. Anti-

causal spike sequences with a postsynaptic spike preceding a presynaptic one lead to synaptic depression. These two processes are commonly referred to as Long-Term Potentiation and Long-Term Depression [8].

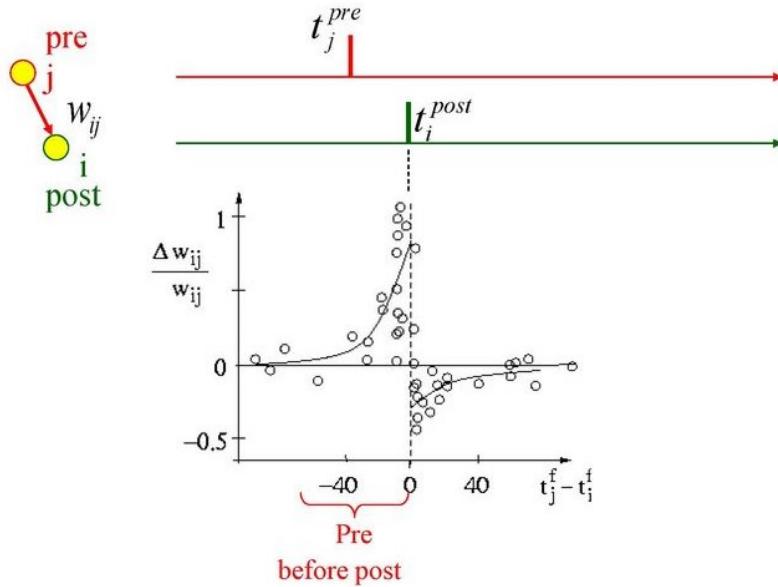


Figure 2.5: *Spike-Timing Dependent Plasticity* schematic: The STDP-function shows the change of synaptic connections as a function of the relative timing of pre- and postsynaptic spikes after 60 spike pairings [9]

The general STDP update rule is formulated in equations 2.1 and 2.2 below.

$$\Delta w^+ = F_+(w) \exp\left(\frac{-|\Delta t|}{\tau_+}\right) \text{ if } \Delta t < 0 \quad (2.1)$$

$$\Delta w^- = -F_-(w) \exp\left(\frac{-|\Delta t|}{\tau_-}\right) \text{ if } \Delta t \geq 0 \quad (2.2)$$

Where  $\Delta t$  defines the time difference between the presynaptic and the postsynaptic spike. The functions  $F_+(\cdot)$  and  $F_-(\cdot)$  model how the weight update depends on the current synaptic weight. Parameters  $\tau_+$  and  $\tau_-$  are time constants which describe the width of the STDP window.

## 2.2.2 Neuromorphic Hardware

Neuromorphic hardware implementations can be divided into three major categories: digital, analog and mixed digital/analog systems. Since the SNN implemented in this work was simulated using the *Spinnaker* hardware [4], only this digital platform is covered.

**SpiNNaker Neuromorphic Hardware:** Full custom or application specific integrated circuit (ASIC) chips, like the *SpiNNaker* system are popular neuromorphic implementations [25]. *SpiNNaker* uses an highly parallel architecture to simulate large spiking neural networks efficiently. This hardware system does not use silicon neurons but relies on regular microprocessors to maintain the flexibility and scalability of ordinary computers and at the same time avoid the Von Neumann Bottleneck when simulating large numbers of neurons. For the size of our networks the 4-node *SpiNNaker* board depicted in figure 2.6 which consists of 64 processors and is able to simulate up to 10,000 neurons was sufficient.

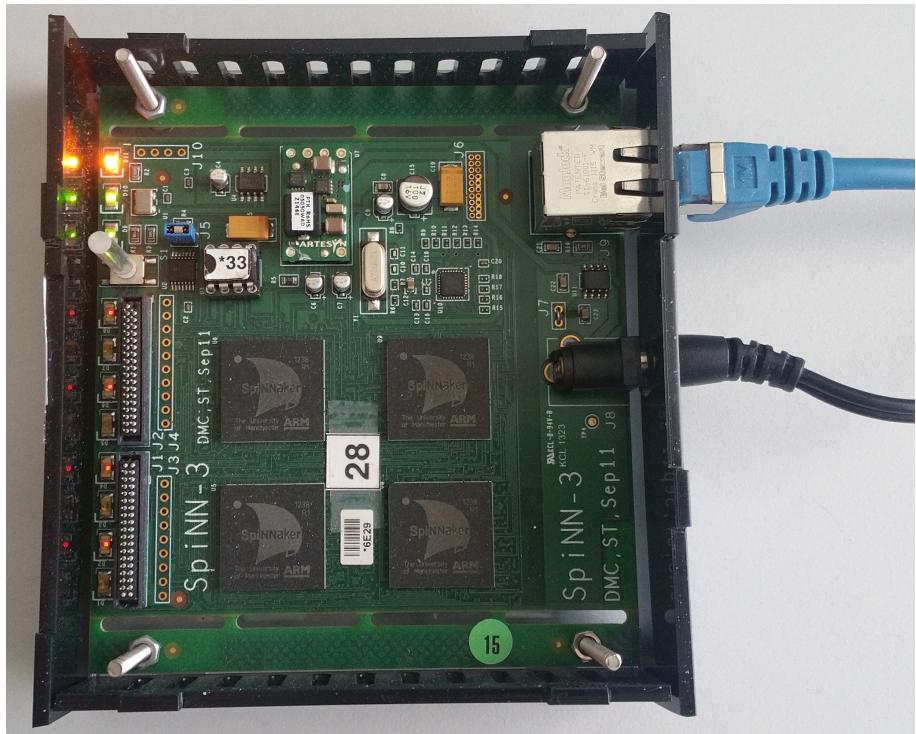


Figure 2.6: The 4-node *SpiNNaker* chip



# Chapter 3

## Overview on the State of the Art of EEG based Brain-Machine Interfaces

In order to provide a glance at this vast and quickly growing field, it can be useful to mention some relevant work of previous groups. In this chapter recent work on EEG based BCIs using statistical and neuromorphic classifiers is covered first. The second part of this chapter presents different decoding and feature extraction methods for motor imagery movements from raw EEG data. In the last part the architecture of a SNN, that is based on the olfactory system of insects is described.

### 3.1 Review: EEG based Brain-Computer Interfaces

Impressive work has been done during the last twenty years in BCI-based motor imagery and various approaches for decoding the EEG signals have been introduced and widely investigated. The majority of publications however are based on BCIs using statistical machine learning algorithms like Support Vector Machines (SVM) [11, 15] and Linear Discriminant Analysis (LDA) [14, 15, 16] for decoding EEG data. For example Xinyi Yong and Carlo Menon demonstrated the classification of different imaginary movements from EEG within the same Limb [11]. In their work they classified EEG signals corresponding to the three classes: rest, imaginary grasp-, and imaginary elbow movements. They achieved the best results using the SVM algorithm and training it in an offline manner. For the binary classification of imaginary grasp and elbow movements, the average accuracy they achieved was 66.9%. For the 3-class problem of discriminating rest against imaginary grasp and elbow movements, the average classification accuracy achieved was 60.7%.

However, to the best of our knowledge very few papers investigated the use of SNNs for BCIs [7, 6]. Zied Tayeb and Emeç Erçelik [18] proved the concept of classifying motor imagery movements using a SNN, which is based on the architecture of the biological olfactory system of insects described in section 3.3. Their network was simulated on the neuromorphic computing platform *SpiNNaker*, described in chapter 2.2. To process the EEG signals the sub bands of the alpha and beta band were extracted. This process is further described in section 3.2. The resulting numeric values where then injected into the network as rate-based spikes trains. The learning process i.e. the tuning of weights within the SNN was manually done since no learning algorithm was implemented. Using this technique they achieved an offline classification accuracy for left and right hand movements with the SNN of 70% and 75% with SVM.

A similar approach was made by Schmuker and his colleagues [20] in 2016, who used the same network architecture based on the insect’s olfactory system to predict two opposing directions during goal-oriented movements. In contrast to the work described above by Zied Tayeb and Emeç Erçelik this network was trained and tested on the *Spikey* chip [21], a neuromorphic hardware platform that uses analog elements to simulate a neuron’s behavior. This network did not require an intermediate real value-based representation because the recorded data were spike based, all computations were carried out with spiking neurons and it therefore operated completely in spike-based fashion. With this approach, they reached an classification accuracy of 89.32% on unseen data.

## 3.2 Decoding Methods of Motor Imagery from EEG Data

**Band Power Features:** For extracting features in [18], they computed the band power from the original raw EEG data. For each channel, band power features in 72 frequency bands, within the alpha and beta bands were calculated using different overlapping narrow windows between 8 and 25 Hz within the alpha and beta frequency bands.

Besides the approach of extracting the band power of the sub-bands in alpha and beta, as described above different methods have been proposed in recent time. A.Sivakami et al. [22] presented two further methods for extracting features for motor imagery from raw EEG data described below.

**ERD Quantification:** One method for calculating features is the quantification of Event Related Desynchronization (ERD) and the Event Related Synchronization (ERS) during a motor imagery movement. This technique is based on the assumption that if an assembly of neurons engaged in the same process work desynchronized

in frequency, the power at this frequency band decreases. Similarly after an assembly of neurons was engaged in the same process they synchronize for a short period until they return to their idle firing rhythm. Movement or preparation for movement is related to a decrease in neural activity in the alpha band. This characteristic is called ERD. It's opposite, rhythm increase or ERS occurs in the post-movement period and with relaxation in the beta frequency band. ERD and ERS are quantified as percentage EEG power decrease or increase within specific frequency bands compared to the idle period before the imagery movement [22].

$$ERD\% = \frac{P_A - P_R}{P_R} \quad (3.1)$$

Where  $P_A$  is the power within the frequency band of interest i.e. alpha band and  $P_R$  is the power of reference during the idle state. In this paper [22] the ERD and ERS features were calculated as follows:

1. Bandpass filtering of all event-related trials.
2. Squaring of the amplitude samples to obtain power samples.
3. Averaging of power samples across all trials.
4. Averaging over time samples to smooth the data and reduce variability.

Using these ERD and ERS features [22] achieved a mean classification accuracy of 70.5% for right and left hand movements.

**Discrete Wavelet Transform:** The second approach presented, the Discrete Wavelet Transform (DWT), gives the ability to analyze a signal which is localized in time and frequency domain. Since EEG signals are non-stationary and transient, it is difficult to extract their characteristics by only considering the frequency domain through ordinary spectrum analysis methods. That is why classification could greatly benefit from applying wavelet analysis which decomposes a signal into a set of functions obtained by shifting and dilating one single function called the mother wavelet. This means that it can extract its components in different frequency bands while keeping the temporal characteristics of the signals. The DWT of a signal  $f(t)$  is given as follows:

$$C_{j,k}(f, \phi_{j,k}) = 2^{-j/2} \sum_{n=-\infty}^{\infty} f(t) \bar{\phi}(2^{-j}t - k), \quad \text{for } j, k \in Z \quad (3.2)$$

Where  $\phi_{j,k}(t) = 2^{-j/2}\phi(2^{-j}t - k)$  is a wavelet sequence,  $\phi(t)$  is the basic wavelet,  $j$  and  $k$  are the frequency resolution and time of the transform respectively. This yields the low-frequency, high-scale component of the signal as the approximation  $A$  and the high-frequency, low-scale component as the detail  $D$ . The extended scheme

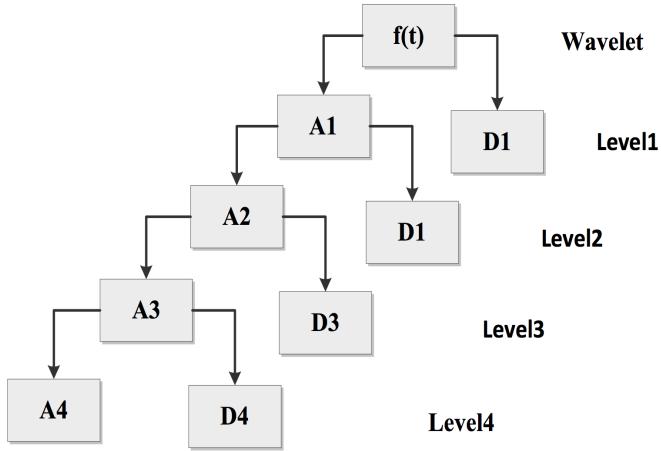


Figure 3.1: Decomposition of signal  $f(t)$  with a DWT of level 4 [23]

presenting the components of a multi-level analysis is shown below in figure 3.1. For decomposing raw EEG signals they chose Daubechies6, with 6 band orthogonal wavelets. The level of decomposition is based on the frequency of interest and the sampling frequency of the raw EEG signal, which was 256 Hz in their case. As a result, level 5 decomposition was chosen and the EEG signal split into detail coefficients D1-D5 and approximation coefficient A5. The D5 coefficients correspond to the alpha band and the D4 coefficients correspond to the central beta band. The extracted features considered are the mean of the absolute value of wavelet coefficients and the standard deviation of the relevant sub-band coefficients D4 and D5. These features represent the frequency distribution and the amount of changes in the frequency distribution. Thus, for each channel, 4 features namely mean and standard deviations corresponding to alpha and beta sub-bands are extracted. Using these characteristics [22] achieved a mean classification accuracy of 72% for right and left hand movements.

A similar approach using the DWT to extract features of left and right hand movements from raw EEG signals was presented by Mingai Li and his colleagues [23]. They demonstrated a modified approach to the one described above. The number of coefficients results from the level of the DWT:  $A_L, D_L, D_{L-1}, \dots, 1$  where  $L$  is the level of decomposition.

$$\left[0, \frac{f_s}{2^{L+1}}\right], \left[\frac{f_s}{2^{L+1}}, \frac{f_s}{2^L}\right], \dots, \left[\frac{f_s}{2^2}, \frac{f_s}{2}\right] \quad (3.3)$$

Their motor imagery EEG data is sampled at 128 Hz and was decomposed into 6 levels by a Daubechies 4-tap wavelet. The resulting frequency bands are:  $0Hz \sim 1Hz$ ,  $1Hz \sim 2Hz$ ,  $2Hz \sim 4Hz$ ,  $4Hz \sim 8Hz$ ,  $8Hz \sim 16Hz$ ,  $16Hz \sim 32Hz$ ,  $32Hz \sim 64Hz$ . The decomposed signal is depicted in figure 3.2.

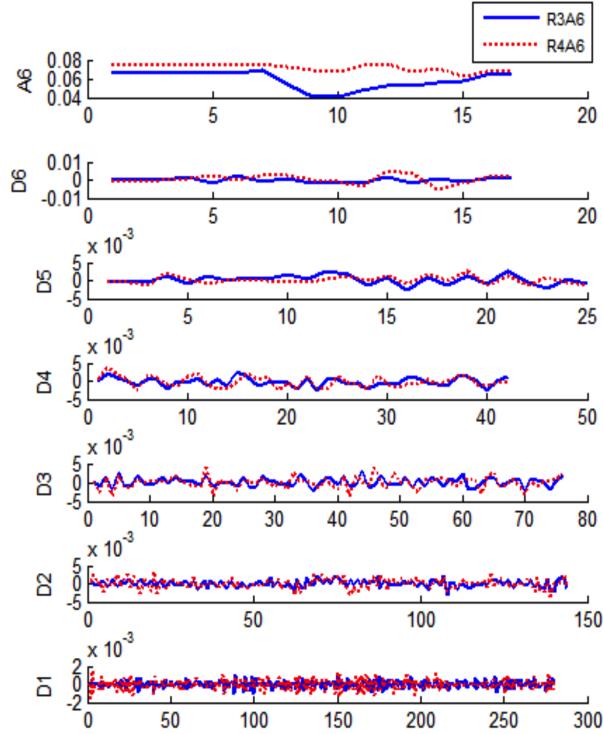


Figure 3.2: Decomposition coefficients of a 6 level DWT [23]

From these bands the motor imagery EEG features are calculated as:

$$F = C_3 A_6 + C_4 A_6 + C_Z A_6 \quad (3.4)$$

Where  $F$  is the feature vector to distinguish between left- and right-hand motor imagery.  $C_i A_j$  represents  $A_j$  of the wavelet packet coefficients on channel  $C_i$ . With these features 93.3% accuracy was achieved using a recurrent neural network and 89% with the Naive Bayes classification algorithm.

### 3.3 Spiking Neural Network based on the Insect's Olfactory System

The architecture used in this work and in the above mentioned BCI publications [20, 18] is based on the insect's olfactory system published in [19] by Michael Schmuker and his colleagues. They introduced a neuromorphic spiking network for supervised classification of multivariate data and showed how the classifier supports nonlinear separation.

**Motivation:** The classifier network was designed to approximate the basic principle of the insect's olfactory system. The chemical sense of insects has evolved to

precisely encode and classify odorants over time. Thus, the neural circuits in their olfactory system are likely to implement an efficient method for coding, processing, and classifying information. In the olfactory system, chemical information is translated into neuronal signals, which undergo processing as they are relayed to higher brain centers. To examine whether the processing principles in the olfactory systems can generally be applied to information processing, they designed a simplified computational model of the three basic processing stages in the olfactory system.

**Architecture:** Its three-stage architecture consists of an input layer, a decorrelation layer, and an association layer shown in figure 3.3-A.

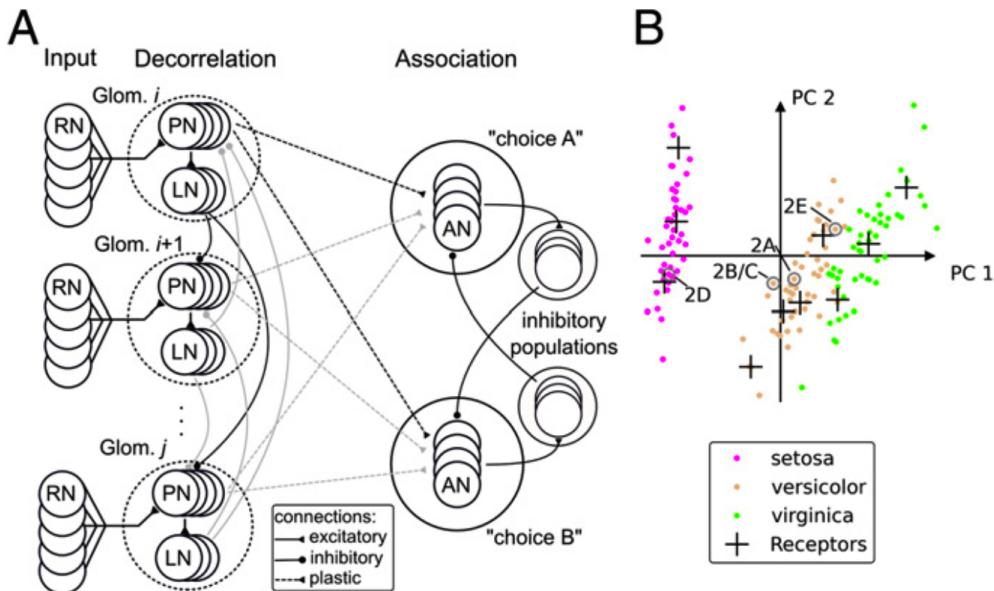


Figure 3.3: (A) The spiking neural network's 3-layer architecture, (B) The placement of the virtual receptors on the *Iris* data set [19]

The function of the 3 layers are described as following: The input layer, real-valued multidimensional data is transformed into bounded and positive firing rates. The data enters the network via ensembles of receptor neurons (RNs). RNs fire spikes at specified rates which are computed from the real-valued input data using virtual receptors (VRs). A VR corresponds to the center of a linear radial basis function in feature space. The magnitude of its response to a data point depends on the distance between the VR and the stimulus. Hence, the VR response is large for small distances between stimulus and receptor, and vice versa. VRs are placed in data space in a self-organized manner, using the neural gas algorithm [24], as shown in figure 3.3-B for the *iris* data set. RN ensembles project onto projection neurons (PNs) in the decorrelation layer, which are grouped in ensembles that represent the so-called glomeruli in the insect antennal lobe. Each RN ensemble targets one

glomerulus, which thus receives excitatory input that represents the activation of one VR. The PNs project to local inhibitory neurons (LNs), which laterally inhibit other glomeruli. The output of the decorrelation layer is projected to the association layer, in which supervised learning for data classification is realized. Association neurons (ANs) are grouped in as many populations as there are classes in the dataset. Each population in the association layer is assigned one label from the dataset. The AN populations project onto associated populations of inhibitory neurons. The strong inhibition between AN populations induces a soft winner-take-all behavior in the association layer. The synaptic weights from PNs to ANs are initialized randomly. An activity pattern presented to the network will thus by chance deliver more input to one of the "choice" populations than to the others, resulting in higher firing rates of that population. If the label of the winner population matches the one of the stimulus, the network performed a correct classification.

The network is trained offline in a supervised fashion by presenting stimuli with known class labels. If classification was correct, active synapses from PNs to the winner population are potentiated. If classification was incorrect, active synapses are depressed.

**Classification Results:** Different data sets were used to test the classification performance of this architecture. First, the network demonstrated its ability to classify nonlinear data on a 2D "Ring" data set consisting of two classes, where one class is located in a cluster centered at the origin and a second class surrounding it. The classifier network running on the neuromorphic *Spikey* chip [21] achieved an average classification rate of 96% on this "Ring" data set just falling short to the Naive Bayes classifier, which reached 98%.

To further explore the capabilities of this architecture a subset of the MNIST data set consisting of the digits "5" and "7", represented in  $28 \times 28$  pixels was used. On this data the spiking neural network reached an accuracy of 94% and outperformed the Naive Bayes classifier which reached 82%.



# Chapter 4

## Solution Design

This chapter first presents an overview over the solutions we aimed to achieve in this work. In the second part the experimental data sets used to test and validate our implementations are described.

### 4.1 Design

The goal of this work is to further improve and extend the approach published in [18] to classify binary motor imagery movements in EEG signals using a SNN. The network is based on the insect's olfactory system presented in 3.3 and is employed on the neuromorphic computing platform *SpiNNaker* described in 2.2.1.

To improve this approach we replaced the hand tuning of the weights between the last two layers of the SNN with a reward based STDP learning algorithm. Further we optimized the network's parameters and tested *Homeostasis* and batch learning on cross-trial encoded features. Additionally to enhancing the network's performance and since the overall classification accuracy also strongly depends on the quality of the features trained on we therefore tested different methods for feature extraction. The techniques tested are: sub-band power, logarithmic band power, ERD/ERS quantification and using the DWT to calculate features. The resulting characteristics are then compared using statistical classifier as well as the SNN on their capability of capturing meaningful information of the motor imagery movements in EEG data.

### 4.2 Experimental Data

If possible we used the same data, as referenced in the literature to better compare our outcomes with published results. All EEG based motor imagery data sets feature a similar chronology of actions during recording. This sequence is described in section 2.1.2. The different recordings only vary in the subject's tasks and in the time frames, depending on the group's individual objective for a recording session.

One experiment set used in this work is data set 3 of the "BCI Competition 2003" contest database recorded from Graz University [40]. It contains 140 trials for both training and testing a classifier. The process of their experimental data acquisition is shown in figure 4.1 Acquisitions during the recording session were obtained every 9 seconds and intervals of two minutes between experiments. The first 2 seconds were quite, at  $t = 2s$ , an acoustic stimulus indicated the beginning of the trial, and a cross "+" was displayed for 1s. At  $t = 3s$ , an arrow (left or cross "+") was displayed for 1 seconds. At  $t = 3s$ , an arrow (left or right) was displayed as a cue. At the same time the subject was asked to move a bar into the direction of the cue by thinking of an corresponding hand movement. The recording was made using a *G.tec* amplifier and some Ag/AgCl electrodes. Three bipolar motor imagery EEG channels were measured over C3, Cz and C4. The data was sampled with 128 Hz. Data from these recording session will from now on be referenced as "Graz 3".

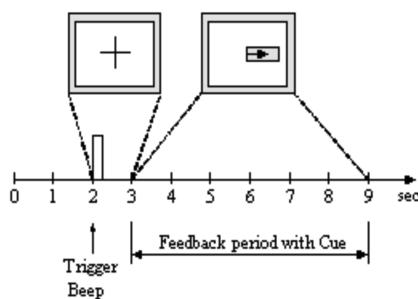


Figure 4.1: BCI Competition 2003, Graz data set 3 [40]

More recent published EEG data also used in this work is the Graz data set B from the BCI Competition 2008 [41]. There, i.a. three bipolar recordings (C3, Cz, and C4) were recorded with a sampling frequency of 250 Hz. The cue-based screening paradigm shown in figure 4.2 consisted of two classes, namely the motor imagery of left and right hand. Each trial started with a fixation cross and an additional short acoustic warning tone at  $t = 2s$ . At  $t = 3s$  a visual cue was presented for 1.25 seconds. Afterwards the subjects had to imagine the corresponding hand movement over a period of 4 seconds. Each trial was followed by a short break of 1.5 seconds. A randomized time of up to 1 second was added to the break to avoid adaptation. Overall 9 data sets, each containing 120 trials, resulted from these recording sessions. These data sets will from now on be referenced as "Graz B1" to "Graz B9".

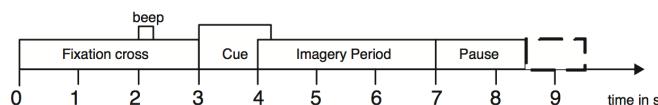


Figure 4.2: BCI Competition 2008, the Graz data set B [41]

# Chapter 5

## Implementation

### 5.1 Data Preprocessing

Before extracting features the raw signals have to be preprocessed in order to achieve similar results with different data sets. All preprocessing steps are implemented using *Python*'s *NumPy* and *SciPy.Signal* libraries.

**Band Pass Filter:** First the EEG sginals are filtered using a digital Butterworth band pass filter of order 5 in the range 0.5-30 Hz because for motor imagery mostly the lower frequency bands (alpha and beta) are of interest.

**Band Stop Filter:** Because the effects of filtering are weighted and by band pass filtering not all 50 Hz interference are eliminated a notch filter is additionally applied. A notch filter is a band-stop filter with a narrow stop band and a high quality factor  $Q = 30$  to attenuate the frequencies around the 50 Hz range.

**Normalize:** After filtering, the raw data are normalized. For each run and electrode  $i$  the mean  $\bar{x}_i$  of the signal is subtracted from every time sample  $x_i(t)$  and the result is divided by the standard deviation  $SD(x)_i$  as shown in equation 5.1 below.

$$x_i^*(t) = \frac{x_i(t) - \bar{x}_i}{SD(x)_i} \quad (5.1)$$

### 5.2 Feature Extraction Methods

In order to classify spatial and temporal patterns in EEG signals, meaningful characteristics have to be extracted before applying classification algorithms. In the following paragraphs different approaches to feature extraction are presented. First, the calculation of band power features is described. The second part covers ERD

and ERS features and the last part shows how we implemented the DWT to extract features. All steps of the feature extraction methods are implemented using *Python's NumPy* and *SciPy.Signal* libraries.

### 5.2.1 Event Related Desynchronization and Synchronization

Motor imagery induce a power increase or decrease of EEG amplitudes, as described in section 3.2. This change mainly occurs in the alpha and beta frequency bands relative to a reference period during rest.

**Sub-Band Power Features:** To calculate band power features for each of the three channels (C3, Cz, C4), the idle and motor imagery sections of the signal are cropped corresponding to the schematic depicted in figure 5.1. These sections are then band pass filtered in 72 frequency bands using different overlapping narrow bands between 8 and 30 Hz. The features are calculated by subtracting the 216 power values of the idle segments from the power values during motor imagery.

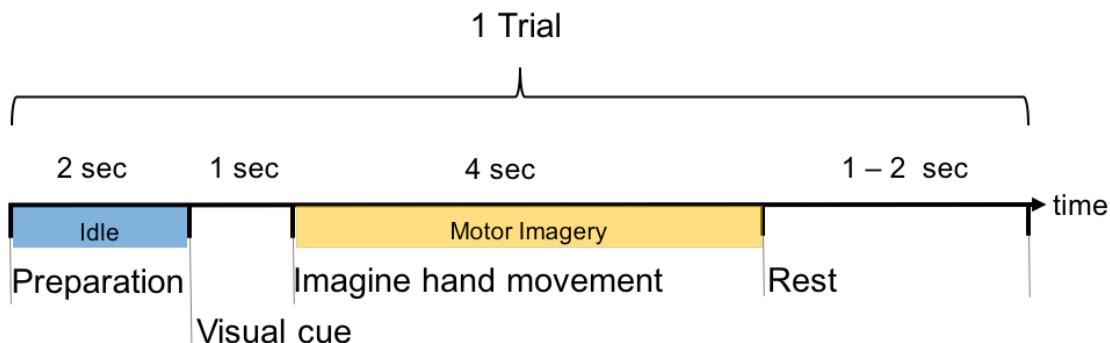


Figure 5.1: Schematic of the band power feature extraction

In order to evaluate the classification results and obtain a reliable and stable model, 10-fold cross validation is employed to calculate the classification accuracy on the Graz B1 data set.

SVM: 75.8% ( $\pm 7.4\%$ )

Random Forest: 78.0% ( $\pm 3.0\%$ )

Logistic Regression: 79.1% ( $\pm 5.8\%$ )

**ERD/ERS Quantification:** ERD/ERS features are computed similar to the description in section 3.2. First, each channel (C3, Cz and C4) of the signal is cropped to isolate the idle, ERD and ERS periods. These time-windows are shown in figure 5.2. The idle window is then band pass filtered using a digital Butterworth filter

of order 5 in the range 8-12 Hz and 13-25 Hz since the power during idle serves as a reference in both frequency bands. The ERD and ERS sections are also filtered using the same Butterworth filter in the range 8-12 Hz and 13-30 Hz, respectively. Next, power samples of all cropped and band pass filtered sections are created by

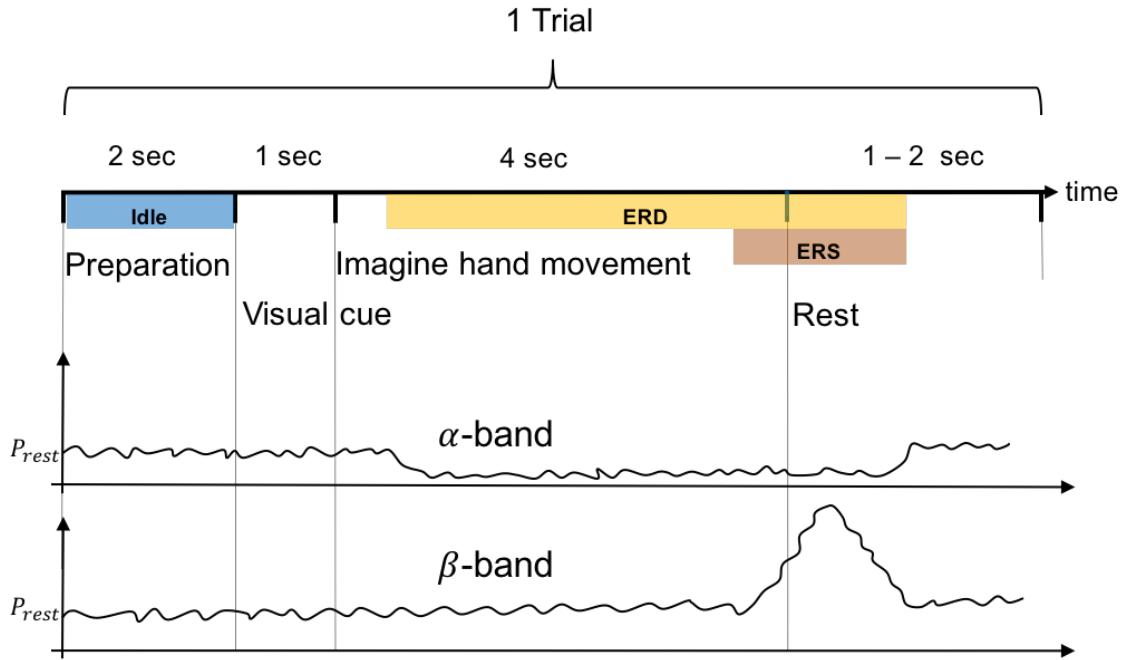


Figure 5.2: Schematic of the ERS / ERD in the alpha and beta band

squaring the amplitude samples. To calculate the features the average power over a trial during the ERD windows is subtracted from the alpha-filtered idle sections and the average power samples during the ERS windows is subtracted from the beta-filtered idle sections. When applying this feature extraction approach to the 120 trials of the Graz B1 data set, 6 features (2 per channel and trial) are created per trial. The following overview shows the achieved mean classification accuracies over 10-fold cross validation.

SVM:	68.1% ( $\pm 9.7\%$ )
Random Forest:	65.9% ( $\pm 12.2\%$ )
Logistic Regression:	67.0% ( $\pm 10.9\%$ )

**Logarithmic Band Power Features:** An alternative approach is estimating ERD / ERS patterns with the logarithm of the variance of band pass filtered EEG during specific time intervals. The logarithm is applied to make the distribution of features approximately normal and better classifiable for linear algorithms [48].

The variance of EEG segments in the time domain for each trial  $i$  and each channel  $c$  is computed as follows:

$$v_c(i) = \frac{1}{\tau - 1} \sum_{j=t_n}^{k_n+\tau-1} (x_j - \bar{x}_i)^2 \quad (5.2)$$

Where  $X_{ij}$  is the  $j$ th sample in the time interval  $\tau_n = [t_n, t_n + T - 1]$  of the  $i$ th trial of the  $w_m$  band pass filtered EEG data. The mean value  $\bar{v}_i$  over all samples of filtered EEG data in the time interval  $\tau_n$  of the  $i$ th trial. For each channel the logarithmic band power feature is defined as:

$$BP_c(i) = \log(v_e(i)) \quad (5.3)$$

According to this definition, the overall band power  $\widetilde{BP}_c^X$ , for each class and each channel is defined by taking the logarithm of the median of the mean of data variances over trials. There we use the median value because its more robust to outliers. The overall band power is written as:

$$\widetilde{BP}_c^X = \log(\tilde{v}_e) \quad (5.4)$$

Where  $\tilde{v}_e$  is the median of the data variances  $v_e(i)$  over all trials of one class.

From this the pattern difference ( $PD_c$ ) for channel  $C3$  and  $C4$  between the two classes ( $L, R$ ) can be defined as:

$$PD_{C3} = \widetilde{BP}_{C3}^L - \widetilde{BP}_{C3}^R \quad (5.5)$$

$$PD_{C4} = \widetilde{BP}_{C4}^L - \widetilde{BP}_{C4}^R \quad (5.6)$$

When the  $PD$  is computed on small alpha sub-bands of a data set the results show in which frequency bands the band power decrease caused by the motor imagery is the most visible. Since this band differs between subjects and recording sessions the results also differ within the range of the alpha band. The  $PDs$  and  $\Delta PD = PD_{C3} - PD_{C4}$  for the first three subjects of the Graz B data set show the following results.

With this information the log band power features are computed from the frequency bands that show the highest  $\Delta PD$ . For example, the frequency range between 9 – 11Hz is chosen for Subject 1 from the Graz B data set listed in table 5.1. This scores the following mean accuracies over 10-fold cross validation.

SVM:	66.4% ( $\pm 4.5\%$ )
Random Forest:	67.7% ( $\pm 6.8\%$ )
Logistic Regression:	62.0% ( $\pm 7.3\%$ )

Table 5.1: Pattern Differences for the first three subjects of the Graz B data

Frequency Range	$PD_{C3}$	$PD_{C4}$	$\Delta PD$
<b>Graz B, Subject 1</b>			
8 – 10Hz	-0.0141	-0.0204	0.0063
9 – 11Hz	-0.0019	-0.0328	0.0308
10 – 12Hz	-0.0063	-0.0057	-0.0006
<b>Graz B, Subject 2</b>			
8 – 10Hz	-0.0986	-0.070	-0.0278
9 – 11Hz	-0.1331	-0.1073	-0.0257
10 – 12Hz	-0.2580	-0.3692	0.1112
<b>Graz B, Subject 3</b>			
8 – 10Hz	0.0004	0.0010	-0.0006
9 – 11Hz	0.0025	-0.0006	0.0031
10 – 12Hz	0.0070	0.0065	0.0005

Table 5.2: Level 6 DWT coefficients of a signal with a sampling frequency of 128 Hz

DWT Coefficients	Frequency Range
$A6:$	0 Hz ~ 1 Hz
$D6:$	1 Hz ~ 2 Hz
$D5:$	2 Hz ~ 4 Hz
$D4:$	4 Hz ~ 8 Hz
$D3:$	8 Hz ~ 16 Hz
$D2:$	16 Hz ~ 32 Hz
$D1:$	32 Hz ~ 64 Hz

### 5.2.2 Discrete Wavelet Transform

Opposed to the previous methods no band pass filtering of the alpha and beta bands is required when applying the DWT to calculate features. For the Graz 3 data set, which is sampled at  $F_s = 128$  Hz, a 6 level DWT with a Daubechies 4-tap wavelet decomposes the EEG signal into detail coefficients D1-D6 and approximation coefficient A6. As shown in the table below the D3 coefficients correspond to alpha and D2 coefficients correspond to central beta band. For every level of decomposition the number of samples is also divided by 2, which can be seen in figure 5.3.

**EEG Feature Selection:** The wavelet decomposed sub-bands are illustrated in Figure 5.3. It shows the decomposition of both C3 and C4 signals for left hand motor imagery.

The statistical features are extracted from the sub-bands corresponding to alpha and central beta waves. The features considered are the mean of the absolute value of wavelet coefficients and standard deviation of the relevant sub-band coefficients D2 and D3. These features represent the frequency distribution and the amount of changes in the frequency distribution. Thus, for each channel, 4 statistical features namely mean and standard deviations corresponding to alpha and beta sub-bands are extracted as features.

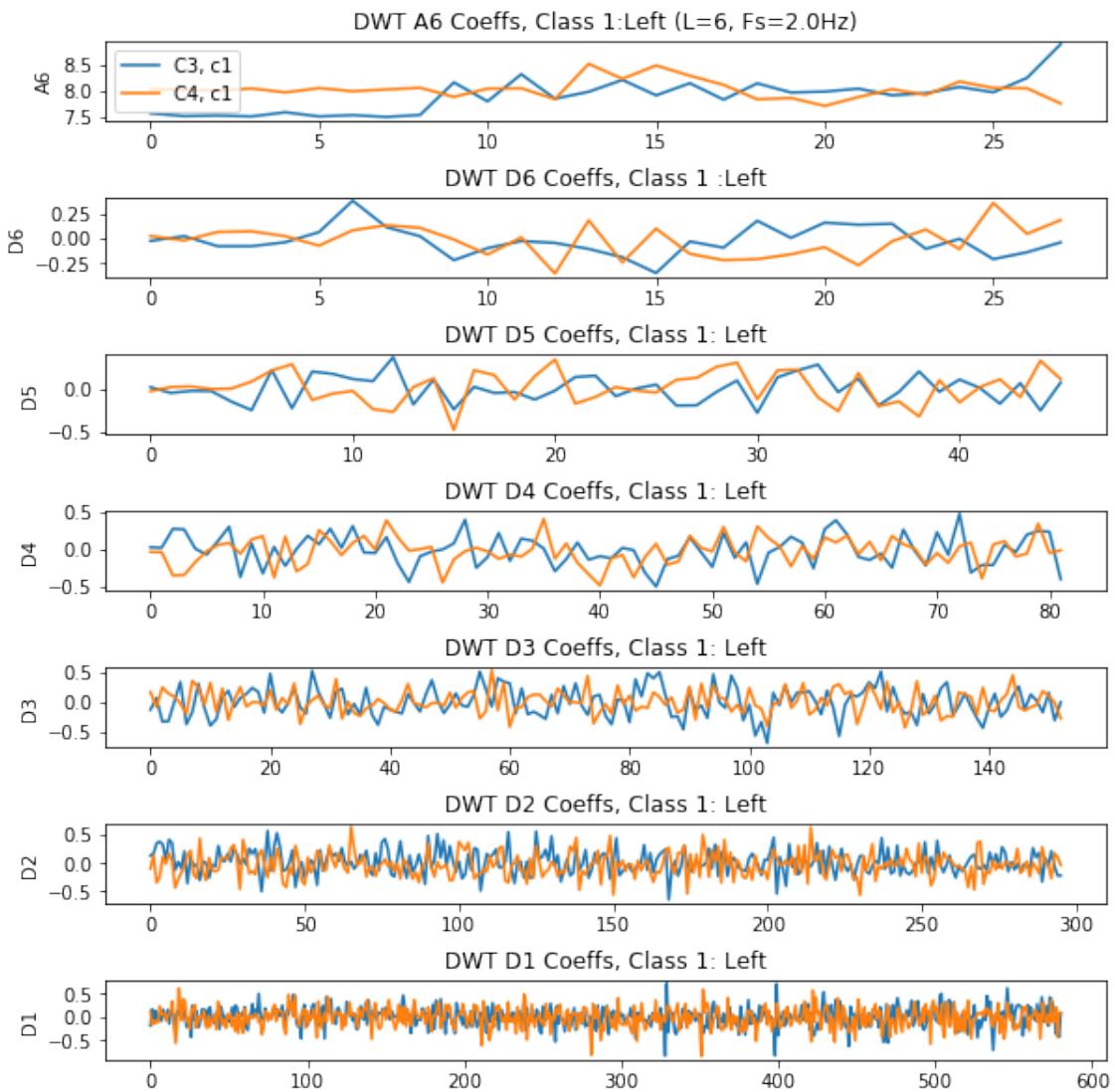


Figure 5.3: Wavelet decomposition of C3 and C4 for right hand motor imagery

Further the power of the approximation coefficients A6 for C3 and C4 is used since it indicates the activity of the two hemispheres. Altogether 10 features are calculated for every trial. Mean classification accuracies for 10-fold cross validation are shown below:

SVM:	62.4% ( $\pm 8.2\%$ )
Random Forest:	63.0% ( $\pm 7.5\%$ )
Logistic Regression:	58.9% ( $\pm 10.1\%$ )

### 5.2.3 Feature Processing: Cross-Trial Encoding

Classifiers often generalize and thus perform poorly on EEG data because the signals show different characteristics for different people and recording sessions. Additionally, the acquisition of big, high-quality data sets for single subjects proves difficult since motor imagery is very exhausting over long periods. Aiming to find signal components that are stable across trials of the same class, we changed the training scheme to cross-trial encoding [44], where we generate new data to achieve greater generalization capabilities. For this task the mean values of every combination of two trials of the original data set  $\mathbf{X}_{original}$  is computed and used as a new trial  $\mathbf{x}_{i,k}$  for further training.

$$\mathbf{x}_{cte,ik} = \frac{\mathbf{x}_i + \mathbf{x}_k}{2} \quad (5.7)$$

With  $\mathbf{x}_i$  and  $\mathbf{x}_k \in \mathbf{X}_{original}$ .

The total number of trials  $n_{cte,total}$  from the new generated data set  $\mathbf{X}_{cte}$  is calculated as:

$$n_{total} = \sum_{i=1}^C \frac{n_i(n_i + 1)}{2} \quad (5.8)$$

Given trial  $n_i$  and class  $c_i$  for the classes in  $C$  of the data set  $\mathbf{X}_{original}$ .

Applying cross-trial encoding to the 102 artifact free, sub-band power trials of the Graz B1 data set we generate a total of 2652 samples for both classes combined. We then tested the generalization performance using the same statistical classification algorithms as in the previous sections. Using 10-fold cross validation where the total data set is randomly split into 10 batches the following results were achieved:

SVM:	94.0% ( $\pm 4.0\%$ )
Random Forest:	92.5% ( $\pm 2.0\%$ )
Logistic Regression:	88.2% ( $\pm 4.1\%$ )

However, if only the 2550 new generated features are used for training and the original 102 samples for testing the classification accuracies declined to:

SVM:	78.4%
Random Forest:	79.2%
Logistic Regression:	62.7%

### 5.2.4 Results: Feature Extraction

The reached classification accuracies on the sub-band power, log band power, ERD quantification and the DWT features extracted from the Graz B1 data set are summarized in figure 5.4. Overall the sub-band power features yielded the best results. The accuracy for these features confirm the results by [18], who achieved 75% classification accuracy with the SVM.

The DWT did, contrary to statements in the literature [22, 23], not outperform the other methods which leads to the assumption that the feature extraction using the DWT was not conducted optimally. The calculations specified by both, [22] and [23] were implemented exactly as described but their results of 72% and 89.29% respectively, differ by a great margin to the ones achieved in this work. Since all calculations were double checked, this leads to the assumption that some preprocessing steps or data selection processes not specified in their work was performed prior to the calculations. As the DWT was the last method implemented in this thesis the limited time did not allow any further investigations and optimization of the results.

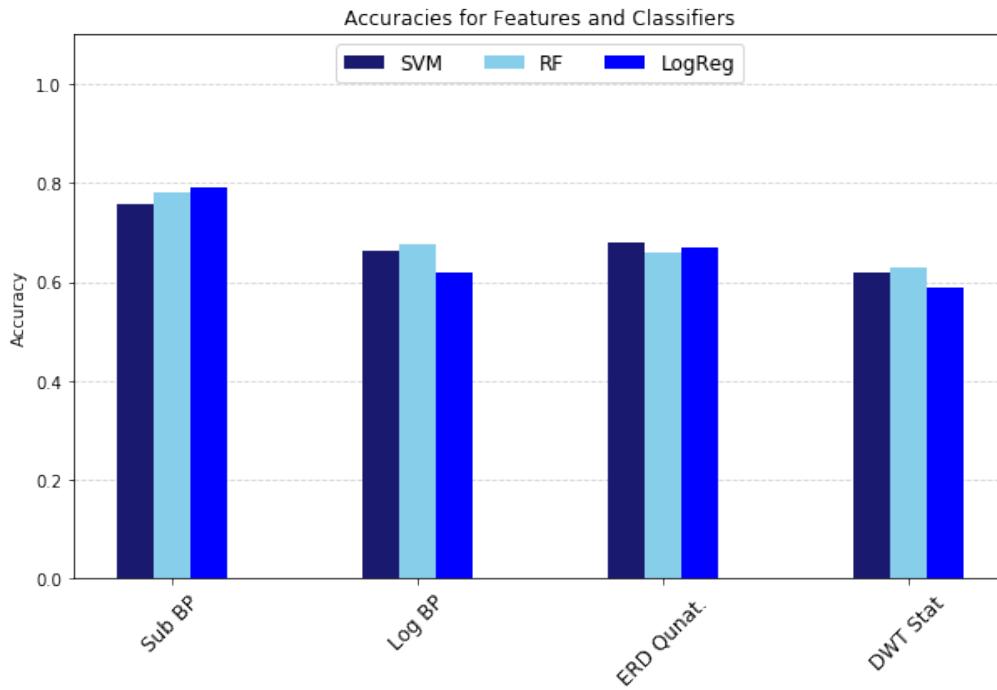


Figure 5.4: Classification results of the different feature extraction methods on statistical classifiers

## 5.3 Spiking Neural Network Implementation

This section covers the implementation of the SNN classifier based on the insect's olfactory system, which is described in more detail in section 3.3. The network's aim is to decode movement intentions from recorded EEG signals. The SNN is implemented to run on the *Spinnaker* hardware platform (section 2.2) using the *Python* programming library *PyNN* [47].

### 5.3.1 Feature Encoding

Encoding is the process of generating spiking patterns from the real valued input stimuli. For this task of converting the extracted features into a spike-based representations two different encoding approaches were tested.

**Population Coding:** Each of the features are encoded using 20 population-coding neurons. The activation of the different population-coded neurons is determined by a Gaussian function and implemented on *Spinnaker* as a Spike Source Array. The feature values are bounded between 0 and 2 and as shown in figure 5.5, 20 Gaussian distributions with variance  $\sigma = 0.025$  are spread evenly between the two bounds. Each Gaussian corresponds to one population-coding neuron  $i$  and determines their firing rate  $R_{j,i}$  as:

$$R_{j,i} = \mathcal{N}(x_j | \mu_i, \sigma) \quad (5.9)$$

with  $x_j$  being the  $j$ -th feature of the data set and  $\mu_i$  the  $i$ -th Gaussian mean.

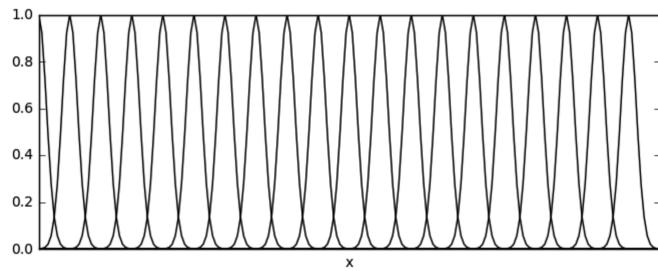


Figure 5.5: Gaussian Distributions that each determine the firing rate of a corresponding population coding neuron

Using this population based encoding scheme an average accuracy on the 2-class *Iris* data set [45] of 96.0% was reached.

**Population and Time Coding:** Furthermore, the effect of adding time information to the previously described population coding method was examined. For rate

$R_i$  of neuron  $i$  a delay was added that is reverse proportional to the feature value. The new rate  $Rt_i$  is calculated as follows:

$$Rt_{j,i} = \mathcal{N}(x_j | \mu_i, \sigma) + \frac{R_{j,max}}{R_{j,i}} \quad (5.10)$$

Where  $R_{j,max}$  is the maximum rate of all neurons for feature  $j$ . The average classification accuracy on the Iris data set did not change noticeably compared to the results of using population coding.

### 5.3.2 Network Implementation

**Architecture:** The network's architecture is implemented similar to the description in section 3.3. Each spike source array from feature encoding is connected to a population of neurons in the encoding (input) layer. The input layer consists of 200 populations of 10 neurons each, which connect one-to-one to the second (decorrelation) layer. The decorrelation layer is organized in functional groups of neurons where each consists of the same number (here  $200 * 10$ ) of populations and neurons as the previous one. Each population in the decorrelation layer has inhibitory connections with other populations with a probability of 40% and excitatory connections to the neurons in the third (association) layer with a probability of 50%. The association layer consists of two inhibitory and two excitatory populations. Each inhibitory population has inhibitory connections to the excitatory ones with a 70% probability. The whole schematic is depicted in figure 5.6.

**Supervised STDP:** The classification decision of the network corresponds to two labeled (class 1: left vs. class 2: right) populations in the association layer. To optimize the weights between the association and decorrelation layer a model of the STDP algorithm was implemented as described in section 2.2.1 with the modification of an added supervisory signal. When passing a trial through the network a reward signal in the form of a post-synaptic spike is injected into the population corresponding to the trial's class in the association layer. This spike triggers the STDP algorithm and increases the weights connecting the decorrelation layer with the association layer. The STDP parameters, that yielded the best results are discussed in the sections below.

**Parameter:** The delay of projections is set to 1 ms. Implemented neurons are leaky integrate and fire type neurons, and the synapse models are exponential current-based synapses. Furthermore, we used 60 Poisson Generators that excite the input layer with 10 Hz firing rate. An overview of the network's connection weights and STDP parameters are listed in table 5.3 and the network neuron parameters are listed in table 5.4.

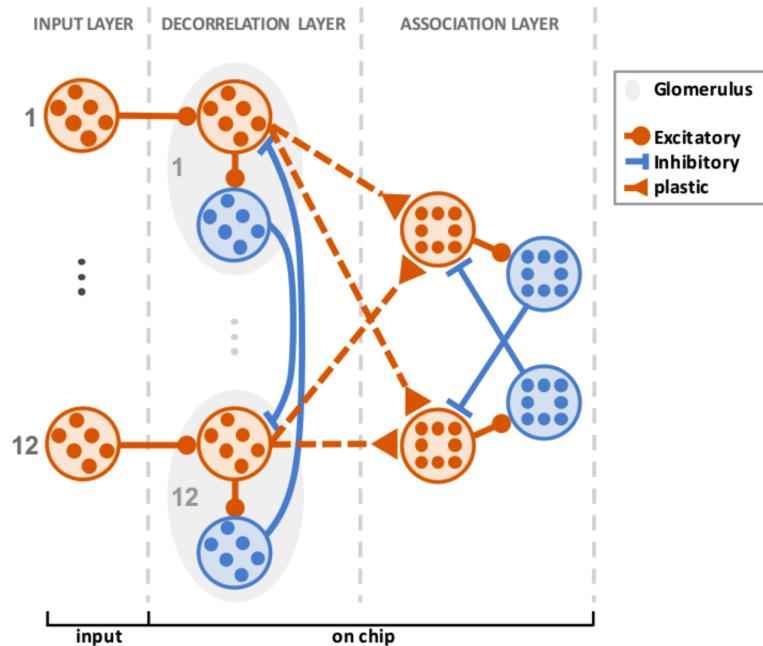


Figure 5.6: Spiking Neural Network schematic [18]

### 5.3.3 Network Evaluation

A major part of this work was to find the optimal network parameters since they greatly define if and how well the network learns to classify the motor imagery from the EEG data. The first implementation of the supervised STDP algorithm showed some variation in classification accuracies. To verify if the learning algorithm was correctly implemented we tested the network on the *Iris* data set. After performing hyper parameter optimization in form of a grid search over the STDP and network connection parameters and the number of epochs in which the data are presented to the network, a mean accuracy of 0.96 ( $\pm 6.0\%$ ) in 10-fold cross-validation was achieved using the parameters in table 5.3. The weights after training, depicted in figure 5.7 show a clear separation for the two classes and confirm the successful learning process.

When further presenting sub band power EEG features from the Graz B1 data set to the SNN it yielded very unstable results between 40% and 80% accuracy for all connection and STDP parameters we tested. To attain consistent classification accuracies we checked the effects of eliminating all random factors from the network. First, the connection probabilities of inhibitory connections within the filtering layer, the connection probabilities of the STDP connections and the connection probabilities of the inhibitory connections within the output layer were replaced with predefined connections. Further the Poisson noise generator was disconnected from the network and fixed values to calculate the spike times were used instead.

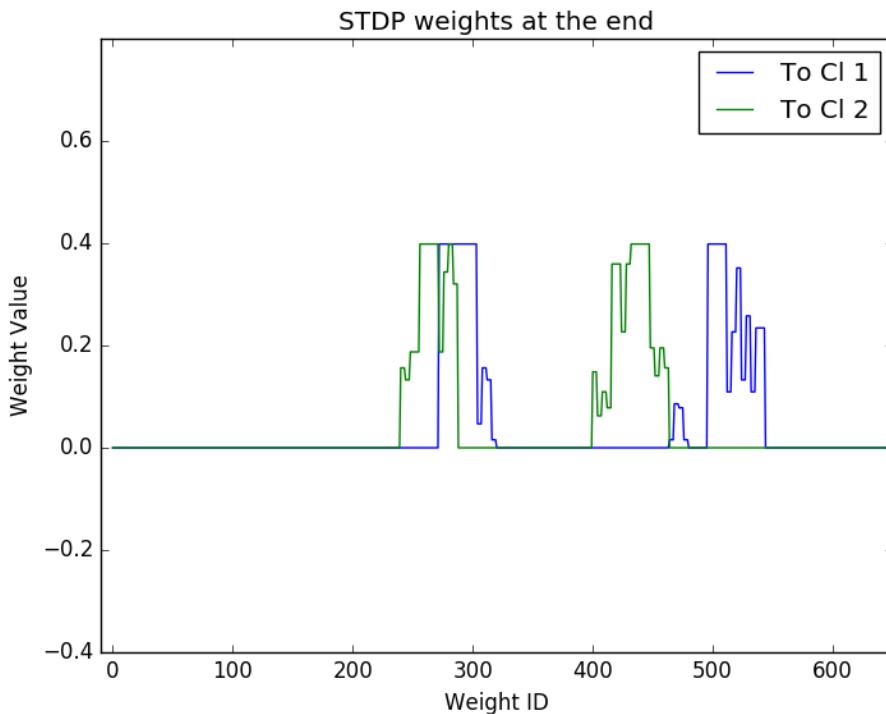


Figure 5.7: Weights after training on the *Iris* data set

Table 5.3: Optimal SNN Parameters

Network Parameter	Value
Number of epochs:	2
Number of trials presented during an epoch:	90
STDP window width $\tau_{plus}$	10.0
STDP Max weight $w_{max}$	0.4 nA
STDP Min weights $w_{min}$	0.0 nA
STDP Weight update parameter $A_{plus}$	0.02
Weights: Spike Source Arrays to encoding layer	0.2 nA
Weights: Encoding layer to decorrelation layer	0.6 nA
Weights: Inh. connections inside the decorrelation layer	0.03 nA
Weights: Decorrelation layer to association layer	0.9 nA
Weights: Inhibitory connection within association layer	0.1 nA
Weights: Source array to Association Layer (Reward)	10 nA
Weights: Poisson Noise connection to Input layer	0.2 nA
Delay of projections:	1ms

Table 5.4: Leaky integrate and fire neuron parameters

Neuron Parameters	Value
Capacity of the membrane $c_m$	0.25 nF
Offset current $i_{offset}$ :	0 nA
Membrane time constant $\tau_m$	20 ms
Duration of refractory period $\tau_{refrac}$	2 ms
Rise time excitatory synaptic alpha function $\tau_{synE}$	5 ms
Rise time inhibitory synaptic alpha function $\tau_{synI}$	5 ms
Reset potential after a spike $v_{reset}$	-70 mV
Resting membrane potential $v_{rest}$	-65 mV
Spike threshold $v_{thresh}$	-50 mV

Unfortunately these changes did not result in any improvement and some even prevented the network to learn at all. Removing the randomness from the connections did not enhance nor degrade the stability of the network and the deviations stayed unchanged. This indicates that a connection between specific populations prior to training does not effect the end result. However, when further removing the Poisson noise from the network the SNN stopped learning and the classification accuracy fell to a mean value of  $\sim 50\%$  accuracy and the weights did not show any sign of separation. This indicates, that the network needs additional noise to learn recognizing different variations of features.

The high instability of the network was overcome by slightly altering the neuron parameters. Using a leaky integrate and fire model with fixed threshold and decaying-exponential post-synaptic current with the parameters listed in table 5.4 stabilized the network. A mean classification accuracy of 68.6% ( $\pm 11.2\%$ ) was achieved on the sub-band power features extracted from the Graz B1 data set.

Further testing indicated that reducing both, the number of training epochs and the dimensionality of the data improved the classification accuracy. The best results were achieved training the SNN over only 2 or 3 epochs. The dimensionality of the feature matrix was reduced from 7 dimensions using Principle Component Analysis (PCA) [46] to 4 while persevering 90.2% of the data's variance. Despite the loss of information during dimensionality reduction the SNN reached higher accuracies. Additionally, training the SNN for only a few epochs prevents overfitting and further improves the networks performance. With these adjustments the mean classification accuracy could be enhanced to 73.4% ( $\pm 9.1\%$ ). The weights after training are plotted in figure 5.8 and show, similar to figure 5.7, a clear separation of weights for the two classes with little overlap.

**Homeostasis:** To further optimize the networks performance *Homeostasis*, a method for self regularization of the weights during training was tested. For this purpose the

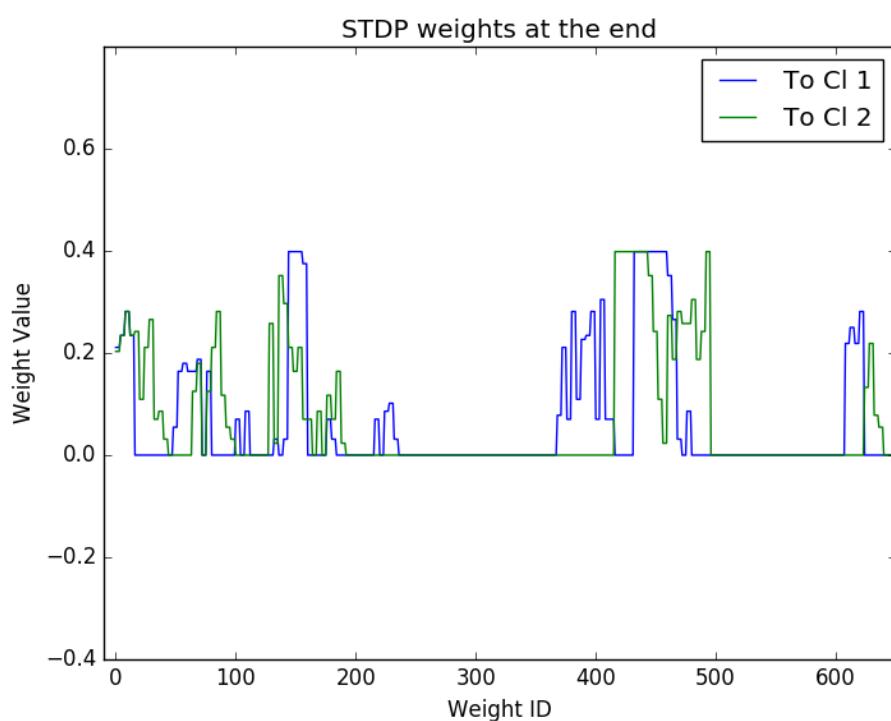


Figure 5.8: Weights after training on sub-band power EEG features yielding a classification accuracy of 73.4%

Graz B1 data set, which contains 102 artifact free samples was split into 9 training batches of sizes 10 and one test set containing the remaining 12 samples. After training the network on batch  $i$  with  $i \in [1, 9]$ , the weights  $\mathbf{w}_i$  are multiplied with a constant factor  $\gamma$  with  $\gamma \in [0, 1]$ , where 0 represents no learning and 1 the same learning process as without *Homeostasis*. The network is then further trained using the weights  $\mathbf{w}_{i,new}$  on the remaining batches:

$$\mathbf{w}_{i,new} = \gamma * \mathbf{w}_i \quad (5.11)$$

This method intends to reduce the effects individual samples that invoke high changes on the weights have and should therefore lead to a better generalization performance of the network. In multiple tests multiplication values between 0.7 and 1 proved to work best for the SNN. If the multiplication value is set too small the information learned by the weights is degraded too much and the network's accuracy drops significantly. It is also necessary to adapt the STDP weight-update parameter  $A_{plus}$  and STDP window width  $\tau_{plus}$  when changing the  $\gamma$  value to overcome the decrease in overall weight value. The lower  $\gamma$  is chosen, the higher  $A_{plus}$  and  $\tau_{plus}$  have to be. The best results were achieved with  $\gamma = 0.9$ ,  $\tau_{plus} = 10$  and  $A_{plus} = 10$ . However the classification accuracy did not improve and remained at 68.0% ( $\pm 12.5\%$ ), 5.8% worse than without *Homeostasis*. The weights after training, depicted in figure 5.9 show some separation for the two classes but it is not as clear as in figure 5.7 trained in the *iris* data and 5.8 trained without *Homeostasis*. This difference shows that altering STDP weights between training runs does not improve the learning process. Although the results without *Homeostasis* are superior, batch learning proves to work and could be a valid option for larger data sets, that don't fit into the *SpiNNaker*'s memory.

**Batch Learning with Cross-Trial Encoded Features:** Since most learning algorithms profit from larger training sets the cross-trial encoding method to create additional samples, described in section 5.2.3 was also tested on the SNN. Cross-trail encoding applied to the sub-band power features of the Graz B1 data set resulted in a total of 2652 samples including the 102 original trials. Since the whole data set does not fit into the memory of the 4-Chip *SpiNNaker* board we divided it in 10 batches. To account for the high number of training samples the STDP parameters had to be adjusted accordingly for the STDP-weights to stay in the range of 0.0 nA and 0.4 nA. The applied parameters for batch learning are listed in table 5.5. When using the 10 batches for 10-fold cross validation a mean accuracy of 74.3% ( $\pm 10.9\%$ ) was reached. This difference to the previously best result of 73.8% is not considered as an improvement since the results are too close and within their standard deviation. When training the SNN on only the 2550 new created features and testing it on the 102 original trials the accuracy declined to 54.0% ( $\pm 21.4\%$ ). A similar decrease could be observed when using statistical classifiers on cross-trial encoded data sets in section 5.2.3.

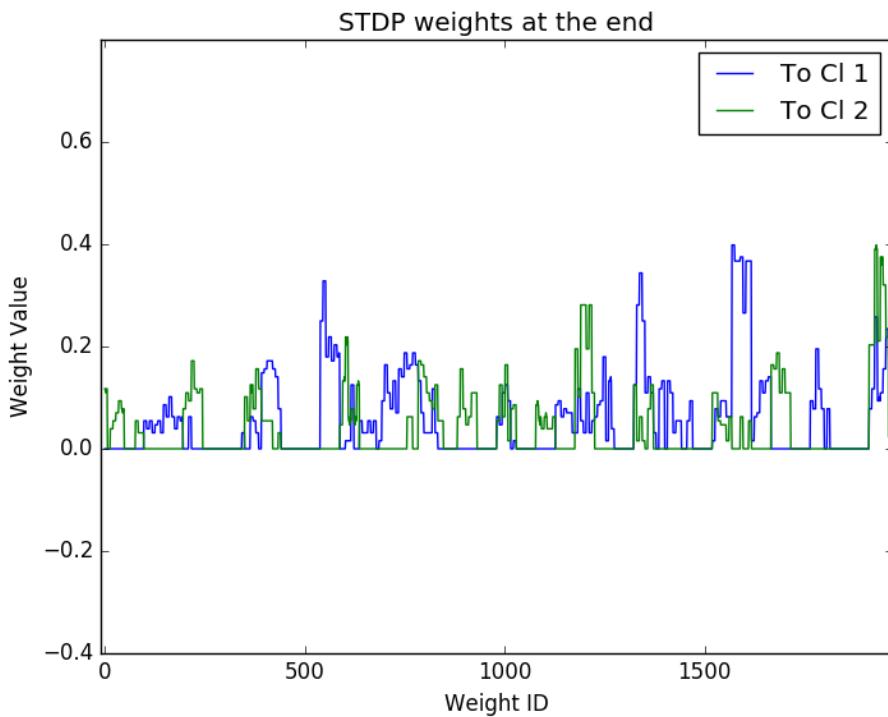


Figure 5.9: Weights after training on EEG data using *Homeostasis*

Table 5.5: SNN parameters for batch learning

<b>Network Parameter</b>	<b>Value</b>
Number of epochs:	1
Number of trials presented during an epoch:	265
STDP window width $\tau_{plus}$	10
STDP Max weight $w_{max}$	0.4 nA
STDP Min weights $w_{min}$	0.0 nA
STDP Weight update parameter $A_{plus}$	0.02

### 5.3.4 Results: Spiking Neural Network

The SNN scored the best results when training the network for 2 or 3 epochs on a low dimensional mapping of the original feature matrix. Together with the network parameters listed in 5.3 and the neuron parameters in table 5.4 the maximal mean classification accuracy over 10-fold cross validation of  $73.8\%(\pm 9.1\%)$  was reached on the sub-band power features extracted of the Graz B1 data set.

None of the other approaches to enhance the SNN's performance proved viable. Adding time information to the the feature encoding for the network did not change the results. Neither did training the network with *Homeostasis*, which even reduced the overall rate by 5.8%. Generating new trials from the extracted features using cross trial encoding only shows to be suitable when also testing on the newly generated features. Training the classifier on the new samples to encode the original trials, fails and reaches 54%, near the classification rate of randomly sampling the two classes.



# Chapter 6

## Discussion

With the implementation of the reward based STDP learning algorithm we could slightly improve the classification accuracy of the SNN compared to [18], with the same network and sub-band power features of Graz B1 data set by 3.8% to 73.8%. Table 6.1 shows an overview of our results compared to similar approaches of classifying binary motor imagery movements.

Overall the SNN did not surpass the statistical classifiers in our tests but further reduced the difference in accuracy. SVM, Random Forest and Logistic Regression outperformed the SNN with an average of 4.12% on all data sets. The results of the SNN compared with state of the art statistical classifiers on the various feature sets is plotted in figure 6.1. The small difference between the statistical classifiers and the SNN indicates the limiting factor for achieving higher classification rates might not be the network but the extracted features.

Of the different signal characteristics we extracted the sub-band power features yielded the best results. Motor imagery induced characteristics often differ for different subjects and the overall EEG signals are not stationary. Therefore, analyzing small overlapping sub-bands might capture additional information compared to seeking more general signal characteristics in wider frequency bands as in the ERD quantification or the logarithmic band power extraction. In our tests the sub-band

Table 6.1: Comparison of classification results on binary motor imagery

<b>Author</b>	<b>Features</b>	<b>Classifier</b>	<b>Accuracy</b>
This work	Sub-band power	SNN	73.8%
This work	Sub-band power	Log.Reg.	79.1%
[18]	Sub-band power	SNN	70.0%
[11]	ERD/ERS	SVM	66.9%
[22]	ERS/ERD	Backprop. ANN	70.5%
[18]	DWT	SNN	72.0%
[23]	DWT	Naive Bayes	89.2%

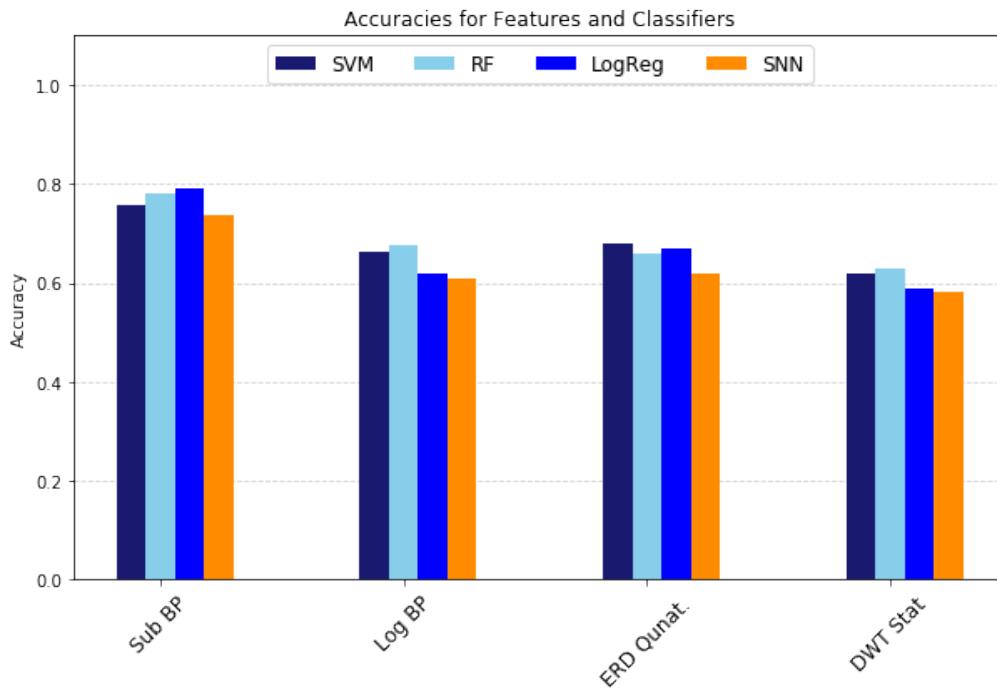


Figure 6.1: Accuracy of the SNN compared with statistical classifiers on different feature sets

power features also surpass the features calculated using the DWT in accuracy. This result was not expected since the DWT presents the most sophisticated signal processing tool tested and [23] reached an accuracy of 89.2% using this technique. Instead of only analyzing the frequency domain during a certain signal interval, as the other feature extraction techniques do, the DWT decomposes the signal while maintaining its frequency and time information. These results lead to the conclusion that although we conducted the feature extraction exactly as presented in various literature reports some preprocessing step might have been performed, that is not mentioned there. It can also not be completely ruled out that our implementation contains an error or bug. Since the DWT was the last method tested during this thesis, the limited time did not allow us to further examine this problem.

# Chapter 7

## Conclusion

This thesis further approaches the realization of a EEG based BCI running on neuromorphic hardware. We showed, the olfactory based architecture is capable of successfully classifying motor imagery EEG signals using a reward based STDP learning algorithm. The reached accuracy surpasses the previously achieved rates of the same architecture but still falls short to the outcome of classical machine learning algorithms.

Two tasks originally intended for the thesis could not be realized and therefore the objective had to be changed correspondingly. First, it was planned to further scale up the network to classify additional reach and grasp movements. This, however would reduce the accuracies, since more classes lead to lower random classification rates and in our case would probably result in rates below 60%. Similar, the optional task of this thesis of implementing the SNN in a real-time BCI application to control a robotic arm could also not be realized. Online classification often drops in accuracy, since the data, the classifier is trained on differs from the live data stream the BCI later classifies. Also subjects are often more nervous in an online BCI environment compared to offline recordings, which would further reduce the BCI's performance to accuracies not suitable for demonstration purposes. As we did not achieve the needed classification accuracies for both additional tasks with neither the SNN nor the statistical classifiers we decided to further examine different feature extraction techniques to improve the classification performance. The results of testing various methods indicate that calculating meaningful features from EEG signals is very difficult because of their non stationary and subject individual characteristics.

To realize a functional BCI running on neuromorphic hardware, most importantly, consistent features have to be found. The DWT, despite of us not achieving meaningful results with, has the potential of creating the features needed to increase the classification rates on all classifiers.

An alternative to extracting features is presented by [20] and briefly described in section 3.1. Using the same SNN they achieve an accuracy of 89.32% while operating on the actual spikes that have been emitted by motor cortical neurons, without the need to construct intermediate non-spiking representations. This possibly takes the

most advantage of spiking information processing and neuromorphic hardware since no information is lost while decoding real-valued features to spiking representations. The downside of this approach is clearly the requirement to record from individual motor cortical neurons, which is very complex and difficult.

# List of Figures

1.1	BCI schematic . . . . .	5
2.1	The functional mapping of the somatosensory and motor cortex [13] .	8
2.2	The electrode positions provided by the 10-20 System [2] . . . . .	9
2.3	One trial of a typical recording session . . . . .	10
2.4	Publications of neuromorphic and neural network hardware over time [25] . . . . .	11
2.5	<i>Spike-Timing Dependent Plasticity</i> schematic: The STDP-function shows the change of synaptic connections as a function of the relative timing of pre- and postsynaptic spikes after 60 spike pairings [9] . .	12
2.6	The 4-node <i>SpiNNaker</i> chip . . . . .	13
3.1	Decomposition of signal $f(t)$ with a DWT of level 4 [23] . . . . .	18
3.2	Decomposition coefficients of a 6 level DWT [23] . . . . .	19
3.3	(A) The spiking neural network's 3-layer architecture, (B) The placement of the virtual receptors on the <i>Iris</i> data set [19] . . . . .	20
4.1	BCI Competition 2003, Graz data set 3 [40] . . . . .	24
4.2	BCI Competition 2008, the Graz data set B [41] . . . . .	24
5.1	Schematic of the band power feature extraction . . . . .	26
5.2	Schematic of the ERS / ERD in the alpha and beta band . . . . .	27
5.3	Wavelet decomposition of C3 and C4 for right hand motor imagery .	30
5.4	Classification results of the different feature extraction methods on statistical classifiers . . . . .	32
5.5	Gaussian Distributions that each determine the firing rate of a corresponding population coding neuron . . . . .	33
5.6	Spiking Neural Network schematic [18] . . . . .	35
5.7	Weights after training on the <i>Iris</i> data set . . . . .	36
5.8	Weights after training on sub-band power EEG features yielding a classification accuracy of 73.4% . . . . .	38
5.9	Weights after training on EEG data using <i>Homeostasis</i> . . . . .	40

6.1 Accuracy of the SNN compared with statistical classifiers on different feature sets . . . . .	44
---	----

# Bibliography

- [1] Hauk, Olaf, Ingrid Johnsrude, and Friedemann Pulvermüller. "Somatotopic representation of action words in human motor and premotor cortex." *Neuron* 41.2 (2004): 301-307.
- [2] Foldvary-Schaefer, Nancy, and Madeleine M. Grigg-Damberger. "Identifying Interictal and Ictal Epileptic Activity in Polysomnograms." *Sleep Medicine Clinics* 7.1 (2012): 39-58.
- [3] Leeb, R., et al. "BCI Competition 2008 - Graz data set B." *Graz University of Technology, Austria* (2008).
- [4] Furber, Steve B., et al. "Overview of the spinnaker system architecture." *IEEE Transactions on Computers* 62.12 (2013): 2454-2467.
- [5] Backus, John. "Can programming be liberated from the von Neumann style?: a functional style and its algebra of programs." *Communications of the ACM* 21.8 (1978): 613-641.
- [6] Kasabov, Nikola K. "NeuCube: A spiking neural network architecture for mapping, learning and understanding of spatio-temporal brain data." *Neural Networks* 52 (2014): 62-76.
- [7] Chen, Yixiong, et al. "NeuCubeRehab: A pilot study for EEG classification in rehabilitation practice based on spiking neural networks." *International Conference on Neural Information Processing*. Springer, Berlin, Heidelberg, 2013.
- [8] Walter, Florian, Florian Röhrbein, and Alois Knoll. "Neuromorphic implementations of neurobiological learning algorithms for spiking neural networks." *Neural Networks* 72 (2015): 152-167.
- [9] Song, Sen, Kenneth D. Miller, and Larry F. Abbott. "Competitive Hebbian learning through spike-timing-dependent synaptic plasticity." *Nature neuroscience* 3.9 (2000): 919-926.
- [10] Meng, Jianjun, et al. "Noninvasive electroencephalogram based control of a robotic arm for reach and grasp tasks." *Scientific reports* 6 (2016).

- [11] Yong, Xinyi, and Carlo Menon. "EEG classification of different imaginary movements within the same limb." *PloS one* 10.4 (2015): e0121896.
- [12] Christian A. Kothe, Introduction to Modern Brain-Computer Interface Design, Lecture 2: EEG Basics, [ftp://sccn.ucsd.edu/pub/bcilab/lectures/02\\_EEG\\_Basics.pdf](ftp://sccn.ucsd.edu/pub/bcilab/lectures/02_EEG_Basics.pdf)
- [13] <https://www.ebmconsult.com/articles/homunculus-sensory-motor-cortex>, [accessed: 22.05.2017]
- [14] Wang, Yijun, et al. "Implementation of a brain-computer interface based on three states of motor imagery." Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE. IEEE, 2007.
- [15] Subasi, Abdulhamit, and M. Ismail Gursoy. "EEG signal classification using PCA, ICA, LDA and support vector machines." *Expert Systems with Applications* 37.12 (2010): 8659-8666.
- [16] Garrett, Deon, et al. "Comparison of linear, nonlinear, and feature selection methods for EEG signal classification." *IEEE Transactions on neural systems and rehabilitation engineering* 11.2 (2003): 141-144.
- [17] Guler, Inan, and Elif Derya Ubeyli. "Multiclass support vector machines for EEG-signals classification." *IEEE Transactions on Information Technology in Biomedicine* 11.2 (2007): 117-126.
- [18] Zied Tayeb, Emeç Erçelik, and Jörg Conradt, (2017), Decoding of Motor Imagery Movements from EEG Signals using SpiNNaker Neuromorphic Hardware
- [19] Schmuker, Michael, Thomas Pfeil, and Martin Paul Nawrot. "A neuromorphic network for generic multivariate data classification." *Proceedings of the National Academy of Sciences* 111.6 (2014): 2081-2086.
- [20] Lungu, I-A., et al. "Predicting voluntary movements from motor cortical activity with neuromorphic hardware." *IBM Journal of Research and Development* 61.2/3 (2017): 5-1.
- [21] Pfeil, Thomas et al. "Six networks on a universal neuromorphic computing substrate." *Frontiers in Neuroscience*, vol.7, (2013)
- [22] Sivakami, A., and S. Shenbaga Devi. "Analysis of EEG for motor imagery based classification of hand activities."
- [23] Li, Mingai, et al. "Combined long short-term memory based network employing wavelet coefficients for MI-EEG recognition." *Mechatronics and Automation (ICMA), 2016 IEEE International Conference on*. IEEE, 2016.

- [24] Martinetz, Thomas, and Klaus Schulten. "A" neural-gas" network learns topologies." (1991): 397-402.
- [25] Schuman, Catherine D., et al. "A Survey of Neuromorphic Computing and Neural Networks in Hardware." arXiv preprint arXiv:1705.06963 (2017).
- [26] Von Neumann, John. The computer and the brain. Yale University Press, 2012.
- [27] Turing, Alan M. "Computing machinery and intelligence." Mind 59.236 (1950): 433-460.
- [28] Murray, Alan F., and Anthony VW Smith. "Asynchronous VLSI neural networks using pulse-stream arithmetic." IEEE Journal of Solid-State Circuits 23.3 (1988): 688-697.
- [29] Blayo, FranÃ§ois, and Philippe Hurat. "8.2 A VLSI SYSTOLIC ARRAY DEDICATED TO HOPFIELD NEURAL NETWORK." VLSI for Artificial Intelligence 68 (1989): 255.
- [30] Salam, Fathi MA. "A model of neural circuits for programmable VLSI implementation." Circuits and Systems, 1989., IEEE International Symposium on. IEEE, 1989.
- [31] Bibyk, Steven, et al. "Current-mode neural network building blocks for analog MOS VLSI." Circuits and Systems, 1990., IEEE International Symposium on. IEEE, 1990.
- [32] Burr, James B. "Digital neural network implementations." Neural networks, concepts, applications, and implementations 3 (1991): 237-285.
- [33] Chiang, M. L., T. C. Lu, and J. B. Kuo. "Analogue adaptive neural network circuit." IEE Proceedings G (Circuits, Devices and Systems) 138.6 (1991): 717-723.
- [34] Madani, K. U. R. O. S. H., et al. "Two analog counters for neural network implementation." IEEE journal of solid-state circuits 26.7 (1991): 966-974.
- [35] Murray, Alan F., Dante Del Corso, and Lionel Tarassenko. "Pulse-stream VLSI neural networks mixing analog and digital techniques." IEEE Transactions on Neural Networks 2.2 (1991): 193-204.
- [36] Hasler, P., and L. A. Akers. "VLSI neural systems and circuits." Computers and Communications, 1990. Conference Proceedings., Ninth Annual International Phoenix Conference on. IEEE, 1990.
- [37] Lee, J-C., and Bing J. Sheu. "Parallel digital image restoration using adaptive VLSI neural chips." Computer Design: VLSI in Computers and Processors, 1990. ICCD'90. Proceedings, 1990 IEEE International Conference on. IEEE, 1990.

- [38] Tarassenko, Lionel, et al. "Real-time autonomous robot navigation using VLSI neural networks." *Advances in neural information processing systems*. 1991.
- [39] Akers, Lex, et al. "A limited-interconnect, highly layered synthetic neural architecture." *VLSI for artificial intelligence*. Springer US, 1989. 218-226.
- [40] Blankertz, Benjamin, et al. "The BCI competition 2003: progress and perspectives in detection and discrimination of EEG single trials." *IEEE transactions on biomedical engineering* 51.6 (2004): 1044-1051.
- [41] Leeb, R., et al. "BCI Competition 2008 - Graz data set B." *Graz University of Technology, Austria* (2008).
- [42] Ubeda, Andres, et al. "Classification of upper limb center-out reaching tasks by means of EEG-based continuous decoding techniques." *Journal of neuroengineering and rehabilitation* 14.1 (2017): 9.
- [43] Müller-Putz, G. R., et al. "From classic motor imagery to complex movement intention decoding: The noninvasive Graz-BCI approach." *Progress in brain research* 228 (2016): 39-70.
- [44] Stober, Sebastian, et al. "Deep feature learning for EEG recordings." *arXiv preprint arXiv:1511.04306* (2015).
- [45] Fisher, Ronald A. "The use of multiple measurements in taxonomic problems." *Annals of human genetics* 7.2 (1936): 179-188.
- [46] Wold, Svante, Kim Esbensen, and Paul Geladi. "Principal component analysis." *Chemometrics and intelligent laboratory systems* 2.1-3 (1987): 37-52.
- [47] Davison AP, Brüderle D, Eppler JM, Kremkow J, Muller E, Pecevski DA, Perrenet L and Yger P (2008) PyNN: a common interface for neuronal network simulators. *Front. Neuroinform.* 2:11 doi:10.3389/neuro.11.011.2008
- [48] Yang, Yuan, et al. "Time-frequency optimization for discrimination between imagination of right and left hand movements based on two bipolar electroencephalography channels." *EURASIP Journal on Advances in Signal Processing* 2014.1 (2014): 38.

# License

This work is licensed under the Creative Commons Attribution 3.0 Germany License. To view a copy of this license, visit <http://creativecommons.org> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.

**MASTER THESIS**

For Leonard Rychly

Mat.-N.03668046, Field of Study: Robotics, Cognition, Intelligence

**Decoding of 3D Reach and Grasp Movements from Non-invasive EEG Signals using SpiNNaker Neuromorphic Hardware****Problem description:**

Brain-machine interface (BMI) technologies aim to provide a bridge between the human brain and external devices. Prior research using non-invasive BMI to control virtual objects, such as computer cursors and virtual helicopters, and real-world objects, such as wheelchairs and quadcopters, has demonstrated the promise of BMI technologies [1]. Moreover, such previous efforts have primarily constrained the BMI control system to be discrete in one dimension or a plane without exploring the full possibility of controls in three-dimensional space [2]. However, controlling a robotic arm to complete reach- and-grasp tasks efficiently in a 3D environment using non-invasive BMI has yet to be shown [3]. Thus, several challenges and opportunities exist for extending the field of BMI from virtual object control to a real-time neuroprosthesis control.

The main objective of this thesis is to decode reach-and-grasp movements in a three-dimensional (3D) space using non-invasive BMI. Ultimately, a real-time control of high number of degrees-of-freedom (DoF) of dexterous prosthetic limb in a 3D environment can be shown. By using the novel massively-parallel computer architecture SpiNNaker, we are aiming to achieve faster computation and enhance the classification accuracy with reduced training time and smaller energy and power consumption.

**Tasks:**

The student shall use our developed algorithm inspired from the olfactory system for insects and extend the previous approach to classify 3D grasp and reach task movements.

This master thesis project requires the student to:

- Implement STDP method on SpiNNaker to train the network
- Scale-up the developed spiking model to decode reach and grasp movements
- 

**Optional task:**

- Test and validate the developed algorithm in a real-time scenario when the user imagines a reach to a target, this imagination would be decoded with a BMI and the robotic arm executes the reach and grasp movement.

**Bibliography:**

- [1] Jianjun Meng, Shuying Zhang, Angeliki Bekyo, Jaro Olsoe, Bryan Baxter, Bin He," Noninvasive Electroencephalogram Based Control of a Robotic Arm for Reach and Grasp Tasks", vol 6, Scientific Reports, 2016
- [2] Yong X, Menon C. "EEG Classification of Different Imaginary Movements within the Same Limb". PLoS ONE, 2015
- [3] Gernot Müller-Putz and Andreas Schwarz and Joana Pereira and Patrick Ofner", from classic motor imagery to complex movement intention decoding: "The noninvasive Graz-BCI approach", vol 228, 2016, Pages 39–70, progress in Brain Research. Elsevier BV, 2016.

**Supervisor:** Dipl.-Ing. Zied Tayeb  
**Co-supervisor:** M. Eng. Emeç Erçelik  
**Start:** 23.01.2016  
**Intermediate Report:** 01.05.2016  
**Delivery:** 01.08.2016



(Jörg Conradt)  
Professor