DOI:10.14753/SE.2025.3078

SEMMELWEIS EGYETEM

DOKTORI ISKOLA

Ph.D. értekezések

3078.

RÁCZ MELINDA

Funkcionális Idegtudományok
című program

Programvezető: Dr. Sperlágh Beáta, c. egyetemi tanár

Témavezető: Dr. Márton Gergely, tudományos főmunkatárs

DOI:10.14753/SE.2025.3078

BRAIN–COMPUTER AND HUMAN–MACHINE INTERFACING USING MACHINE LEARNING TECHNIQUES

PhD thesis

Melinda Rácz

Semmelweis University Doctoral School

János Szentágothai Neurosciences Division

Supervisor: Gergely Márton, Ph.D

Official reviewers: Péter Barthó, Ph.D
Otília Menyhart, Ph.D

Head of the Complex Examination Committee: Alán Alpár, D.Sc

Members of the Complex Examination Committee: László Acsády, D.Sc
Márton Szemenyei, Ph.D

Budapest
2025

DOI:10.14753/SE.2025.3078

Table of contents

DOI:10.14753/SE.2025.3078

DOI:10.14753/SE.2025.3078

List of abbreviations

AdaGrad	adaptive gradient method
ALS	amyotrophic lateral sclerosis
ANN	artificial neural network
AP	action potential
A_{pp}	peak-to-peak amplitude
BCI	brain–computer interface
CNN	convolutional neural network
CSP	common spatial pattern
DL	deep learning
EEG	electroencephalography
ECG	electrocardiography
ECoG	electrocorticography
EMG	electromyography
EOG	electrooculography
EP	evoked potential
ERD	event-related desynchronization
ERP	event-related potential
ERS	event-related synchronization
HMI	human–machine interface
ID	isolation distance
LFP	local field potential
LSTM	long short-term memory
MEA	multielectrode array
MI	motor imagery
RBF	radial basis function
ReLU	rectified linear unit
RMSprop	root mean square propagation
RPVR	refractory period violation rate
SD	standard deviation
ssVEP	steady-state visual evoked potential

DOI:10.14753/SE.2025.3078

SUA	single unit activity
SVM	support vector machine

DOI:10.14753/SE.2025.3078

1. Introduction

Transcending traditional computer input devices has been of interest for decades to scientists (1,2) and people outside research (see e.g. (3)) alike. Brain–computer interfaces (BCIs) (4) are a subtype of human–machine interfaces (HMIs) that bypass the common route between the user's mind and software running on a computer. BCIs are of particular importance within the niche of HMIs. This is largely due to the prevalence of diseases related to motor dysfunction, such as tetraplegia (and, to a lesser extent, paraplegia) (5–7) and locked-in syndrome resulting from amyotrophic lateral sclerosis (ALS) (7,8), brainstem stroke or other causes (9). Through BCIs, some aspects of personal freedom could be given back to persons living with the aforementioned impediments. Besides, BCIs open a number of perspectives to improve the life of healthy users (4,10).

1.1. BCIs in general

A general BCI consists of a number of stages, these are: signal acquisition, extraction of the intended action from the given activity (preprocessing, feature extraction, classification and translation) (4,10), output generation and feedback via a suitable application towards the user (11). The schematic representation of a general BCI system is shown in Figure 1.

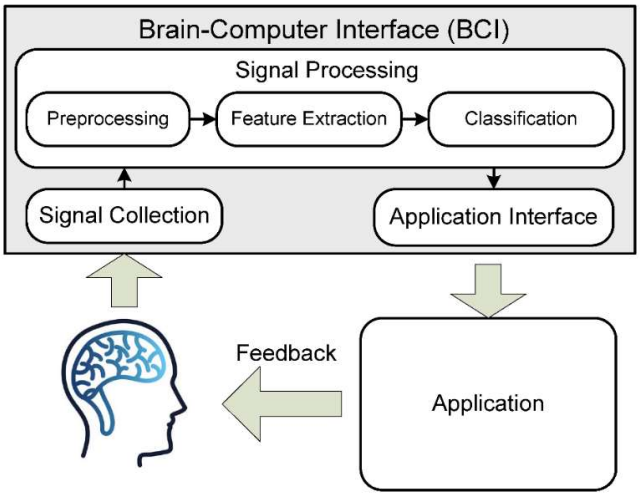

Figure 1. *Schematic block diagram of a general BCI system. Source: (10).*

There is a great diversity regarding the means of acquiring signals from the brain, that has an effect on the subtlety and applicability of the BCI system in question. According to Schwartz et al. (11), the more invasive the recording method, the more detail is

DOI:10.14753/SE.2025.3078

contained in the acquired signal, but also the more risk the setup process involves. Additionally, this limits the availability of data the developmental process may utilize and to some extent, the possible types of signal processing algorithms, according to the 'no free lunch' theorem (12). Electrode placement and spatiotemporal resolution of the most widely applied electrophysiological data acquisition methods are illustrated in Figure 2. Since this work focuses on electrophysiological signal processing, other modalities, such as magnetoencephalography, functional near-infrared spectroscopy and functional magnetic resonance imaging are not discussed further.

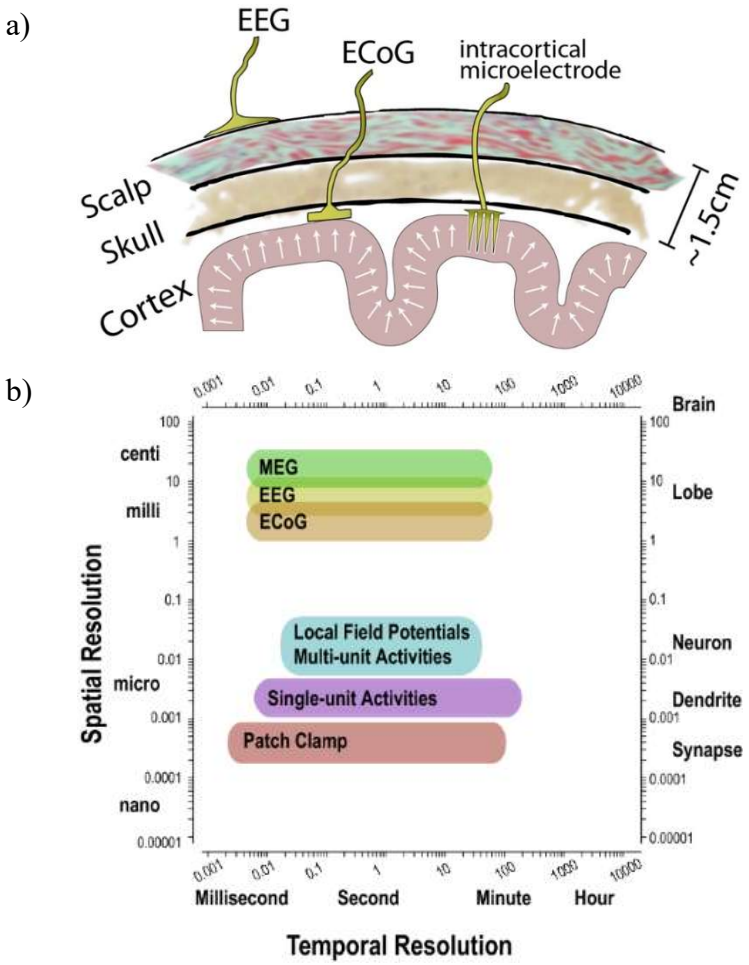

Figure 2. *Comparison of common electrophysiological signal acquisition methods. a) Electrode placement pertaining to recording techniques. Electroencephalography (EEG) captures signals from the scalp, centimetres away from the cortex. Electrocorticography (ECoG) electrodes are placed onto the surface of the brain, above or under the meninges. Intracortical microelectrode arrays are implanted into the parenchyma, between neurons. Source: (7). b) Spatial and temporal resolution of various recording techniques (MEG: magnetoencephalography). Source: (13).*

Robust operation is a fundamental requirement imposed on BCIs. Additionally, due to the individual differences regarding users and the possible changes their neural activity may exhibit, recalibration or automatic adaptation may be necessary, rendering the utilization of hard-coded algorithms (e.g. (14–17)) infeasible. From the 1990s onwards, data-driven machine learning algorithms such as adaptive autoregressive (18) and hidden Markov (19) models, Bayesian classifiers (20,21), linear discriminant analysis (22) and even artificial neural network (ANN)-based solutions (23–30) have been applied as classifiers in BCIs. Support vector machines have been utilized the same way since the early 2000s (31–33).

1.2. Extracellular recordings and spike sorting

Single-unit (SUA) and local field potential (LFP)-based BCIs require extracellular probes implanted into the brain parenchyma. As the method allows the discrimination of activity of different neurons or neuronal ensembles (1), they can implement intricate control patterns for driving e.g. a robotic arm (9). On the other hand, during the insertion process both the skull and the meninges get opened, exposing the user to a risk of tissue damage and infection (1) or other adverse effects (4,34). As a result, the pool of research employing human subjects is rather small (35). For instance, the first closed-loop robotic arm control utilizing neural probes was also implemented using rats (30).

SUA-based BCIs necessitate the discrimination of action potentials (APs, or activities, spikes) based on their putative neuron of origin. This problem is also called spike sorting (36,37). Spike sorting solutions are possible due to the differing waveform of activities originating from different neurons, pertaining to the morphology of the individual cells and their relative position to the neural probe (38). APs are mostly detected from filtered recordings: LFP content below 300–500 Hz should be removed in order to make detection easier (39), using e.g. bandpass filters (40) or wavelet denoising (41) via simple thresholding (37,42). The filtered signal may be transformed using e.g. continuous wavelet transform (43) or some energy operator (44,45). In recent years, deep learning (DL, i.e. ANN architectures consisting of multiple layers) methods also have been applied for spike detection (46). Then various features, such as amplitude and temporal width (36) get extracted from the waveform, enabling clustering. Dimension reduction using e.g. principal component analysis (36,47) or independent

component analysis (48) is also a common practice and wavelet analysis can be applied at this stage, as well (42). Some solutions, such as state-of-the-art clustering algorithm KiloSort, use template matching (49,50). Extracted features get assigned to clusters using a general classifying method such as expectation maximization with mixture of gaussians (51,52) or t-distribution (53), k-means (45,54,55), support vector machines (SVM) (56), or, in the recent years, ANNs (44,57,58) such as convolutional neural networks (CNNs) (46) or variational autoencoders (59). Not uncommonly, ANNs also embed the feature extraction phase (46,58,59).

The already complex problem of spike sorting is perturbed further through a number of phenomena that classification algorithms must withstand. These include sparsely firing neurons that produce so few activities that SUAs may get merged with another cluster, e.g. multi-unit activity, or classified as artifacts (60). AP bursts are series of APs generated by the same neuron but of decreasing amplitude (37) or changing shape (36). Activities of these spike trains may be assigned to different clusters. Overlapping spikes from different neurons frequently result in their waveforms distorted beyond recognition (37) or even being cancellated (36), potentially eluding from proper classification or detection. Electrode drift due to the displacement of the probe (36,37) or other non-stationarities related to e.g. cell damage resulting from probe insertion (61–63) or tissue responses (11) may deteriorate spike sorting performance on the long run.

1.3. Electroencephalography and EEG-based BCI paradigms

Electroencephalography and electrocorticography (ECoG) collect the activity of larger cortical areas, with the former having the smaller spatial and frequency resolution. EEG electrodes are typically centimetres away from each other while micro-ECoG electrodes may have a few millimetres gap (64). Frequency content ranges up to 70 and 200 Hz for EEG and ECoG, respectively (11). ECoG employs a grid of electrodes placed epidurally or subdurally, thus bypassing the filtering and damping effect of the skull and the scalp (1,34). Nevertheless, this technique also requires craniotomy. Data used during the development of BCIs originate mainly from epileptic patients who were implanted with grids serving diagnostic purposes (7). With this said, the resulting signal is detailed enough to drive e.g. an exoskeleton (5). Of the discussed signal acquisition methods, EEG is the least invasive, utilizing electrode caps or even headsets that can be applied and/or

removed at any time, without any risk, at the cost of signal quality (1,34). Therefore, EEG-based BCIs mostly extract large-scale cortical activity patterns. On the other hand, the relatively low cost of measurements, thus the large amount of available data helps the developmental process to a great extent (65).

Contrarily to BCIs that are driven by neuronal activity patterns, EEG-based BCIs may utilize cortical potentials that are more or less the same across users. These methods encompass event-related potentials (ERPs) and evoked potentials (EPs). ERP-based BCIs exploit P300 that is a special waveform elicited 300 milliseconds after the presentation of a looked-for stimulus and may signify the completion of its processing (66). In the classic paradigm reported in Farwell and Donchin (67), rows and columns of letters arranged in a matrix are flashed periodically. If the flashed set of letters contains the letter of interest, a P300 wave is elicited, resulting in the typing of the character. A major drawback of this solution was the low throughput rate (2.3 characters/min) that later solutions successfully attempted to improve (68–70).

EPs elicited by an adequate stimulus are measurable by EEG above the visual (VEP) (71), auditory (AEP) (72) or somatosensory cortical areas (SEP) (73). They have a well-defined waveform and timing to the extent that they may be utilized in clinical practice for diagnosing e.g. demyelination diseases, dementia (73) or hearing loss (72). Stimulus of a frequency exceeding a threshold pertaining to a particular modality produces an EP of the same frequency and its harmonics (74) (steady-state EPs: ssVEP/ssAEP/ssSEP). If multiple stimuli are present, the strongest response will be elicited by the one the subject directs their attention to. This phenomenon is exploited by ssVEP (75) and ssSEP (76) BCIs. The main advantage of these spellers is the fast response (generally, all stimuli are presented at once) and the fact that they do not require training. On the other hand, flickering stimuli may induce fatigue in the user (75).

There are also BCIs that require some form of training from the part of the user, such as MI-based (or, sensorimotor rhythm—SMR—based) BCIs. MI-BCIs make use of the fact that during the planning and the real execution of a specific movement, the same cortical patterns emerge (77,78). In other words, event-related synchronization and desynchronization (ERS/ERD) patterns are formed regardless of the ability to perform the corresponding physical activity. Solutions from the family of MI-BCIs are also used for therapeutic purposes in relation to e.g. stroke, autism spectrum disorder and attention-

deficit hyperactivity disorder (79). The most common imaginary movements involve the two hands (26), feet (80,81) and tongue (82) as the cortical mapping of these body parts separate well (78). A similar early BCI utilizes the voluntary manipulation of the power in the mu (8–12 Hz) EEG band, implementing two-dimensional cursor control (83).

The type of the feature to be extracted depends on the type of the BCI. P300 spellers commonly utilize bandpass (between 0.1–10 Hz (84), 0.1–12 Hz (68), 0.1–30 Hz (69) or 0.2–35 Hz (67)) or lowpass (under 40 Hz (85)) filtered time domain signals. Since the amplitude of the P300 wave is rather small, its detection requires the averaging of multiple occurrences, thus the mental 'typing' of a letter lasts for a series of (even as long as 30) trials (67). P300 wave can be detected via a number of methods, such as score maximization based on features resulting from stepwise discriminant analysis, peak picking, area and covariance estimation (67), linear discriminant analysis on raw data (68), applying covariance matrix shrinkage (85), random forest (70), SVM (70), naïve Bayes classifier operating on spatial filtered data (69) or ANNs (84).

Since ssVEP signals are characterized by their frequency, it seems to be straightforward to utilize data transformed into frequency (or time-frequency) domain. Indeed, many solutions apply this approach either in the form of Fourier (86,87), wavelet (88) or Hilbert-Huang (89,90) transforms. Another family of spellers use spatial filters such as common spatial pattern (CSP) (91), minimum energy combination (92) and canonical correlation analysis (93) along with their variations (75,94). As classifiers, SVM (88), Fisher classifier (89), regularized linear discriminant analysis (91) and DL methods (28) have been utilized beside classical maximum finding approaches (87,90).

ERD and ERS also pertain to certain frequency bands, most commonly to mu and beta (15–30 Hz) bands (95), thus MI signals are usually analysed in frequency domain. For instance, the classic Graz BCI uses frequency components computed for each channel as features, of which the components providing the most accurate classification are selected using sequential floating forward selection (95). Similarly, systems described in (96,97) also use frequency components. With this said, other features, e.g. CSP (78), filter bank CSP (98), Laplacian filters (99) are also can be encountered. Classifiers for MI BCIs include linear discriminant analysis (78,95,97,100), linear classifiers (99), SVM (100,101), k-nearest neighbour (100,101), naïve Bayes (100), logistic

regression, (101) regression trees (100) and, recently, ANN-based (101) and DL methods (102–105).

The structure of the remaining part of this thesis is as follows. Section 2 delineates the objectives of my work. Section 3 describes relevant data acquisition methods in greater depth along with the properties of data and/or recording devices used throughout our investigations. Machine learning techniques are also discussed in this section with the characteristics of the particular solutions implemented. Details on quality assessment metrics conclude this section. Sections 4 and 5 feature and interpret the results in relation to the objectives and the state of the art, respectively. Section 6 concludes this work.

DOI:10.14753/SE.2025.3078

2. Objectives

2.1. Objective 1: development of a CNN-based spike sorter

Spike sorting is a fundamental challenge of neuroscience. In recent years, machine learning methods, most importantly, artificial neural networks emerged as possible solutions for the problem, surpassing hard-coded algorithms. Our objective is to implement a convolutional neural network-based spike sorting algorithm trained on real-life data encoding extracellular potentials, that uses samples from action potentials during their full time course.

2.2. Objective 2: development of a CNN-based spike predictor

In some applications, such as online classifiers, the system benefits from the shortest delay possible. Therefore, a sorter that is able to predict the class of a spike before the neuron actually fires, may have a utility in practice. Our objective is to determine the earliest point in time when spike sorting is already possible (assuming a 50 % accuracy) based on samples from the local field potential preceding action potentials and/or the falling edge of the actual action potentials.

2.3. Objective 3: development of an EEG/EMG-based user interface for robot control utilizing a portable EEG headset

A practical demand imposed on human–machine interfaces is that they may require the least amount of training data for proper functioning. Among machine learning algorithms, support vector machines are praised for their comparatively reduced need for training data. Our objective is to implement a support vector machine-based classifier that functions as an interface between a portable EEG headset and a mobile robot. The complete system is aimed at being applied for demonstrative purposes. Such demonstrations commonly involve subjects inexperienced at using brain–computer interfaces that require training and/or special capabilities (such as motor imagery BCIs). The system to be implemented may exploit the presence of muscular artifacts in the EEG signal.

DOI:10.14753/SE.2025.3078

3. Methods

This work includes the development of two BCI/HMI systems of different paradigms that reside at the opposite ends of the spectrum of HMIs. One of them is invasive and due to the abundance of data, may employ a complex and more accurate classifier of a flatter learning curve. The other one is non-invasive with a data acquisition process of limited time that necessitates the application of a less-sophisticated sorting algorithm that can be trained at a faster pace.

This section aims to provide a description of the data and the classification algorithms as follows. Section 3.1 delineates the data acquisition methods, along with the details of data applied during the development of the invasive BCI and the non-invasive HMI, respectively. Similarly, Section 3.2 attempts to give some insight into the theory of each algorithm featured in this work accompanied by implementational details.

3.1. Electrophysiological data and preprocessing

3.1.1. Extracellular recordings

There are various electrophysiological techniques that enable us to characterize the behaviour of parts or the whole of the central nervous system. EEG provides information about the activity of large cortical areas (11). At the other end of the spectrum, patch clamp is able to record currents flowing through single ion channels of a neuron (106). Extracellular electrophysiology resides halfway between these two extremes, being able to capture the activity of multiple neurons in the vicinity of a measuring microelectrode (37). Early neural probes took the form of stainless steel, tungsten or iridium microwires (107) arranged into a matrix configuration (108). Modern solutions (multielectrode arrays, MEAs) are commonly fabricated from silicon and may include thousands of recording sites on a single shank (109), reaching different cortical layers (61). Image of an action potential recorded using a MEA is shown in Figure 3.

Since neural probes are implanted into extracellular space, they are able to record APs from multiple neurons. These activities can be classified into different clusters based on their waveform given even the simplest recording configuration (61), but MEAs provide a detailed spatial image about their environment. They afford recordings with rich information content (38) enabling triangulation in relation to individual neurons (61). On

the other hand, they produce large-scale data suffering from the curse of dimensionality (110,111). Thus, they both help and complicate the process of spike sorting. Cluster averages from a real-life recording are shown in Figure 4.

Extracellular probes are capable of recording neuronal activity in a radius of a few hundred micrometres, but APs from only their immediate surrounding (distance < ~50 µm) can be clustered into SUA, i.e. activities belonging to a particular neuron. The rest of activity contributes either to multi-unit activity, i.e. detectable spikes without a well definable neuron of origin (~50 µm < distance < ~140 µm), or background noise (distance > ~140 µm) (37,61).

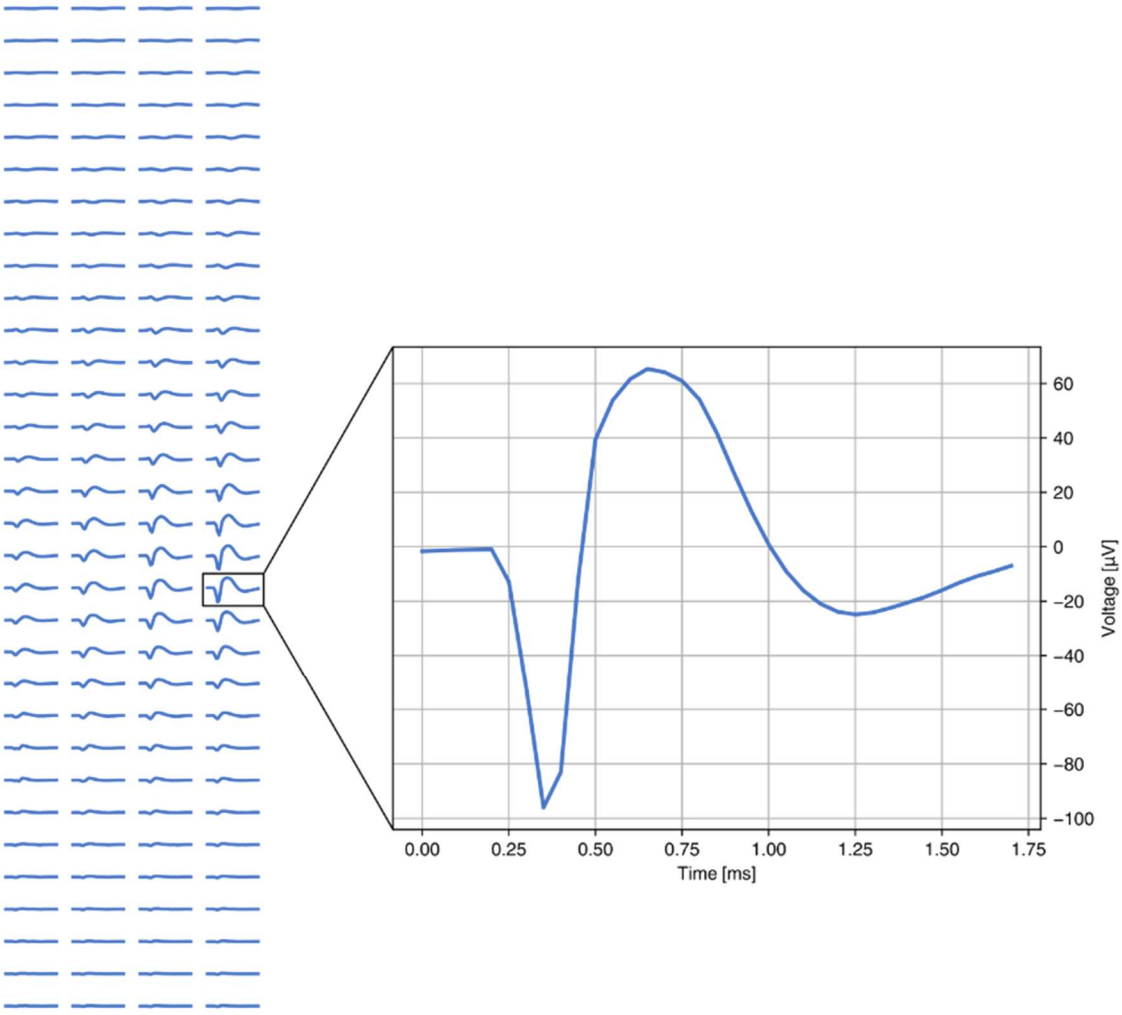

Figure 3. *Typical image of an action potential recorded using a 32 × 4 multielectrode array (that was applied in our research). Inset: voltage and temporal characteristics of the action potential demonstrated on the maximum channel. Source: own figure from (46);*

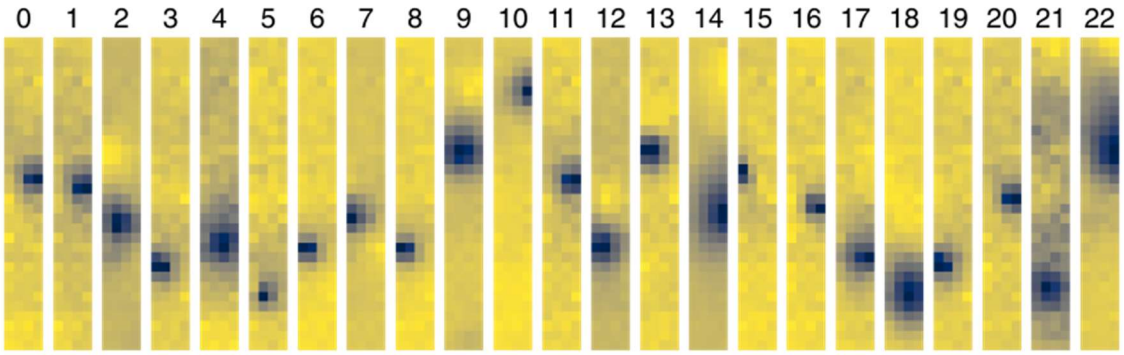

Figure 4. *Average image of spike peaks (minima) for each cluster in an electrophysiological recording (Recording 4, acquired using a MEA also featured in Figure 3). Source: own figure from (46); © IOP Publishing. Reproduced with permission. All rights reserved.*

In our research (46), parts of the database published in (62) were selected for processing. During data acquisition, a high-density complementary metal-oxide semiconductor MEA was applied. The physical dimensions of the probe were 8 mm × 100 µm × 50 µm (length × width × thickness). Its 20 µm × 20 µm recording sites were fabricated of porous titanium nitride, yielding 50 kΩ impedance measured at 1 kHz. The sites were arranged into a 32 × 4 grid with an inter-electrode pitch of 22.5 µm (112). Individual recordings were 45 minutes long, containing spontaneous brain activity acquired from the primary somatosensory cortex of Wistar rats. The subjects were in ketamine-xylazine-induced anaesthesia during the course of the measurements. Probes were implanted to a dorsoventral depth of 1700 µm at an insertion speed of 0.002 mm/s. Recordings were made using an Intan RHD-2000 system employing 20 kHz sampling at 16-bit resolution. Raw recordings got processed using KiloSort (113) then clusters resulting from the analysis were refined by an expert utilizing open-source Python library Phy that was designated for manual curation of large-scale electrophysiological data (114).

Characteristics of the dataset and the recordings within are detailed in Table 1. Peak-to-peak amplitude (A_{pp}) was obtained by averaging the average peak-to-peak amplitude of the units in a particular recording. Cluster quality metrics, i.e. isolation distance (ID) and refractory period violation rate (RPVR) was computed for each cluster using the sortingQuality package (115), then averaged for each recording. Standard deviations (SD) for each metric were also computed akin to averages. ID estimates the distance between APs belonging to a cluster and other spikes recorded on a specific electrode (116). RPVR

corresponds to the percentage of APs within the refractory period, i.e. the time period after a spike when a neuron is not supposed to be active due to membrane hyperpolarization. Activities violating the refractory period probably signify contamination by another unit (117).

Table 1. *Characteristics of the recordings in the database used during our investigations. File: identifier of the recording. N: number of clusters in the recording.* A_{pp}*: peak-to-peak amplitude of the clusters within the recording. ID: isolation distance. RPVR: refractory period violation rate. Average and standard deviation (SD) for each variable was aggregated across the clusters.*

File	N	A_{pp} [µV]		ID		RPVR [%]	
		Average	*SD*	*Average*	*SD*	*Average*	*SD*
1	46	233.923	120.016	53.448	57.102	0.0016	0.0023
2	39	222.113	122.905	43.096	35.965	0.0016	0.0033
3	39	206.657	138.627	44.846	54.814	0.0019	0.0029
4	24	128.441	41.923	50.559	40.727	0.0005	0.0009
5	42	138.467	55.133	25.951	14.219	0.0017	0.0024
6	33	158.464	59.435	41.315	33.122	0.0015	0.0014
7	36	195.474	82.344	53.076	52.332	0.0008	0.0008
8	36	183.173	84.333	46.722	31.809	0.0009	0.0015
9	42	154.401	80.194	40.432	28.106	0.0015	0.0021

Samples used for training, validating and testing the classifiers were taken as follows, with terminology used throughout this work emphasised. We used both 2D and 3D data as samples, i.e. single instances taken at a specific time point and composite data consisting of multiple instances. From this point on, 32 × 4 two-dimensional data will be referred to as *frames* and 3D data composed by a given number of frames will be called *timeslots* of a particular *length*. Frames were taken a given number of sampling periods before (*time offset*) the firing times determined by KiloSort (*marks*) in order to find spike peaks. Marks fall generally 1-2 sampling periods after the actual minimum of APs, depending on the given recording and cluster. Timeslots included a frame from the vicinity of the mark, with a few frames preceding and following it. The distance of the first frame from the mark is referred to as *starting point*. In order to include frames from the entirety of the activity for a better representation while keeping training time reasonably low, data were decimated equidistantly by a *decimation factor*. Terms defined here are demonstrated in Figure 5.

DOI:10.14753/SE.2025.3078

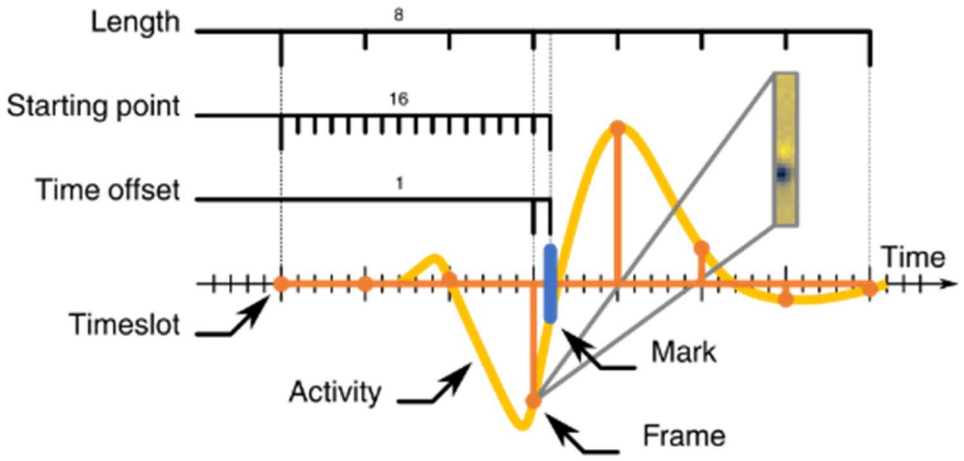

Figure 5. *Demonstration of the terms used throughout the thesis. Activity: the action potential/spike in question. Mark: timing of a particular spike determined by KiloSort. Frame: 32 × 4 sample taken from the recording/activity at a given (single) point of time. Time offset: the number of sampling periods a given frame precedes a mark by. Timeslot: a collection of frames taken equidistantly from an activity, assuming a decimation factor (in this example, this factor equals 5). Length: how many frames a timeslot consists of. Starting point: the number of sampling periods the first frame of a timeslot precedes a mark by. Source: own figure from (46);*

Parameters of frames and timeslots were determined experimentally with results as follows. For frames, the optimal offset was 1, i.e. spike peaks were best represented with frames just preceding marks. The length and optimal starting point for timeslots was 6 and 16, respectively. In other words, we got the best results for timeslots that included the best frames with 3 frames preceding and 2 frames succeeding them, using a decimation factor of 5. Thus, we applied 32 × 4 and 6 × 32 × 4 samples.

Samples for the predictors were of dimensions 32 × 4 and 5 × 32 × 4. Also, we applied a wider range of offsets and starting points. During the exploration stage regarding the hyperparameters of the 2D classifier, classification accuracy for offsets between 10 and −9 were measured (i.e. the first frame was taken 10 sampling periods before, the last one 9 after the marks) using one of the recordings (Recording 4). Thus, we could get some insight into the prediction capabilities of the network. For the 3D predictor, we examined timeslots of a starting point ranging from 10 to 29. In other words, the timeslot was slowly shifted backwards out of the spike. Note that the last frame of the earliest timeslot

corresponds to a frame having an offset of 9 in 2D. The principle of data sampling for the 3D predictor is shown in Figure 6.

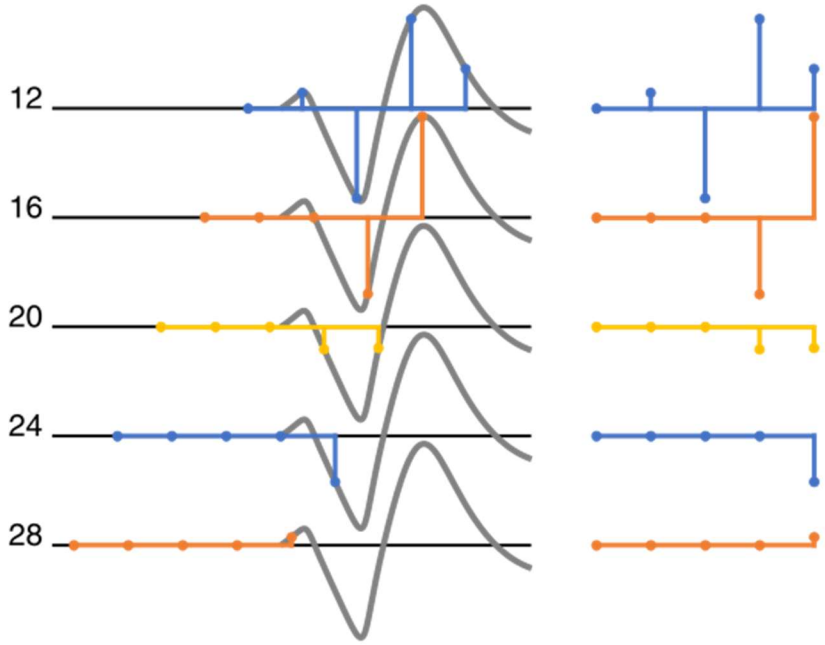

Figure 6. *Demonstration of data sampling for the 3D predictor with the rows corresponding to the starting point. A starting point of 28 assumes the frame to be (almost) entirely shifted out of the actual action potential. Source: own figure from (46);*

Samples were subsequently normed between $[0, \quad 1]$ and $[-1, \quad 1]$ one by one to make the optimization process converge faster. Norming was performed using Equation [1], where x' and x stand for the normed and original samples, x_{max} and x_{min} the maximum and minimum value of the original sample, and x'_{max} and x'_{min} the upper and lower bounds of prescribed range (0 or −1 and 1), respectively.

$$x' = \frac{x'_{max} - x'_{min}}{x_{max} - x_{min}}\left(x - \frac{x_{max} + x_{min}}{2}\right) + \frac{x'_{max} + x'_{min}}{2} \qquad [1]$$

3.1.2. Electroencephalography and electromyographic artefacts

Electroencephalography (EEG) measures the sum of electric dipoles formed by ionic currents flowing through the membrane of the neurons of the brain, called the local field potential (LFP) (118). While extracellular probes are able to capture signals in a radius of a few hundred micrometres (37), one EEG electrode can record neuronal activity above an area of 10 cm^2 (118). In contrast to extracellular measurements, EEG is measured on

the scalp, therefore the method is feasible for being the means of data acquisition in non-invasive brain–computer interfaces (11). On the other hand, LFP gets heavily filtered by the skull and the scalp, thus both amplitude and frequency content of the EEG signal is significantly compromised (with amplitude and frequency reduced to a few hundred microvolts and a few ten hertz, respectively) (11).

In addition to signal originating from neuronal sources, EEG is able to record other signals usually regarded as artifacts. These include the correlates of cardiac activity (ECG), eye blinks, ocular movements (EOG), and myoelectric (EMG) activity pertaining to e.g. frowning, chewing and gulping (119). The aforementioned artifacts are usually of larger amplitude than the EEG signal itself and also overlap with relevant activity in the frequency domain (119). Thus, they are usually removed by means of e.g. artifactual segment rejection (120) or independent component analysis (119). A few of the most common artifacts are illustrated in Figure 7.

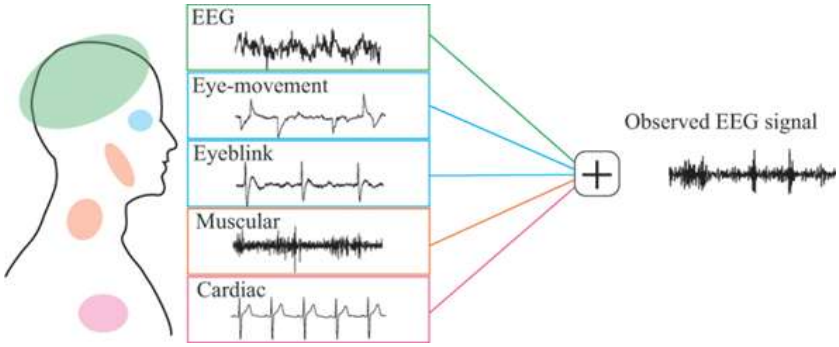

Figure 7. *Commonly encountered artifacts in the EEG signal shown with their characteristic waveforms and places of origin. Source: (121).*

Utilizing EMG artifacts in HMIs can be useful considering a number of practical reasons. Firstly, facial muscles may be intact in traumatic tetraplegia and are rarely affected at the early stage of ALS (122). Particularly, extraocular muscles are among the last that patients lose the control of (123). On the other hand, there is no guarantee that a patient in complete locked-in syndrome (i.e. the stage when all voluntary muscle control have been lost) is able to retain their capabilities for generating cortical patterns (124,125)[1]. Additionally, a great proportion of BCIs have never been tested on actual patients, with only healthy subjects participating in the development process (126). In other words, we

[1] The latter paper was retracted due to scientific misconduct (selective data selection); this is the reason for the citation.

cannot be certain that BCIs are actually useful for people who need it most. Secondly, EMG-based HMIs may not require training. Paradigms implemented in EEG-based brain–computer interfaces often require the user learning to execute a particular task. For instance, in order to operate a motor imagery (MI) BCI, they have to imagine the sense of moving a particular limb, that necessitates practice (127). Moreover, individual differences regarding the structure of the brain (127) and the instantaneous mental state of the user (128) may also have detrimental effects on the efficacy of a given BCI. Other types of BCI, such as event-related potential-based (ERP) and steady-state visual evoked potential-based (ssVEP) applications suffer from similar problems, mainly regarding the amplitude of the potential of interest (127). This phenomenon is mentioned under the name BCI illiteracy in the literature (127,128) and affects approximately 15–30 % of people (129).

In the project involving the implementation of an SVM-based classifier for an EEG/EMG-based human–machine interface (130), the aim was to develop a mobile robot system that could be used during laboratory demonstrations and public science events. Since these situations involve generally (from a BCI point of view) unprepared subjects with a limited timeframe for operating the system, a solution compatible with the broadest pool of users given the shortest setup time possible was desired. Therefore, in order to eliminate BCI illiteracy at the cost of the elegance of interfacing the robot directly with the brain, facial gestures manifesting as EMG artifacts were chosen as control patterns.

The mobile robot was designed and developed at Bejczy Antal Center of Intelligent Robotics of Óbuda University by students. It is controlled by an Intel mini-PC (Intel NUC8i5BEH) running a software based on the Robot Operating System open-source middleware suite over Ubuntu Linux (131). The hardware incorporates a LIDAR (YDLIDAR X4), a depth camera (Intel RealSense D455) and differentially driven wheels equipped with wheel encoders. Data connection and external control can be established via the MQTT protocol (130).

As signal acquisition device, MindRove arc (132) was chosen. This EEG headset features six channels located in positions C5, C3, C1, C2, C4 and C6 according to the 10–20 system. Reference and driven right leg electrodes are placed behind the right and left ear, respectively. Its semi-dry electrodes are fabricated using PtIr threads woven into conductive fabric. The device samples EEG signals at a 500 Hz sample rate and a 24-bit

resolution. UDP data connection to an external computer is established via Wi-Fi. MindRove also provides a software development kit that implements access to raw data from multiple languages including C# (133). The appliance is depicted on Figure 8.

Figure 8. *Promotional graphic of the MindRove arc. Source: own figure from (130).*

In order to make the handling of the system as simple as possible but providing a satisfactory functionality, two control instructions were established, namely, turning to the right direction by 90 ° and steering forward. During an active command, the user is supposed to continuously make the facial gesture of choice at a given rate for a prolonged period of time, e.g. blinking once in every second for 3 seconds. This time period got implemented as a software parameter freely chosen by the user. The only limitation was that it had to be an integer. During preliminary tests, 2-second samples yielded the best results, thus it was set as default value for the algorithm. Additionally, a baseline (resting) state was defined that corresponds to a pause in control and assumes the user staying idle. The paradigm implementing data acquisition, classifier training and then actual robot control was written in C#. For data processing, Emgu CV (version 4.1.1)(134) that is a .NET wrapper for image processing library OpenCV (135), was applied.

During initialization, sample length and the number of training samples per class are to be entered, then training samples are to be provided (i.e. the gestures of choice are to be performed) as prompted by the software. When the sufficient number of samples has been collected, the classifier gets trained. After that, if the user performs a gesture, the sample is classified. The corresponding command is sent via MQTT to the robot that follows the given instruction. The layout and the operation of the paradigm is illustrated in Figure 9. During a given run of the paradigm, samples of $f_s \cdot n_s \times n_{ch} = 500 \cdot n_s \times 6$ size are collected, where f_s stands for the sampling rate, n_s the number of seconds, and n_{ch} the number of channels. Channel values were transformed into frequency domain using

discrete Fourier transform (DFT) using the implementation of OpenCV. Frequency components within the range $(0,\ \ 80]$ Hz were kept for further processing. The upper limit was determined experimentally, i.e. the accuracy of the classifier was optimal using this frequency range. Samples then were whitened using Equation [2](130), where x'_i stands for the whitened sample, x_i the raw sample, and μ_i and σ_i the average and standard deviation of the given sample, respectively.

$$x'_i = \frac{x_i - \mu_i}{\sigma_i} \qquad [2]$$

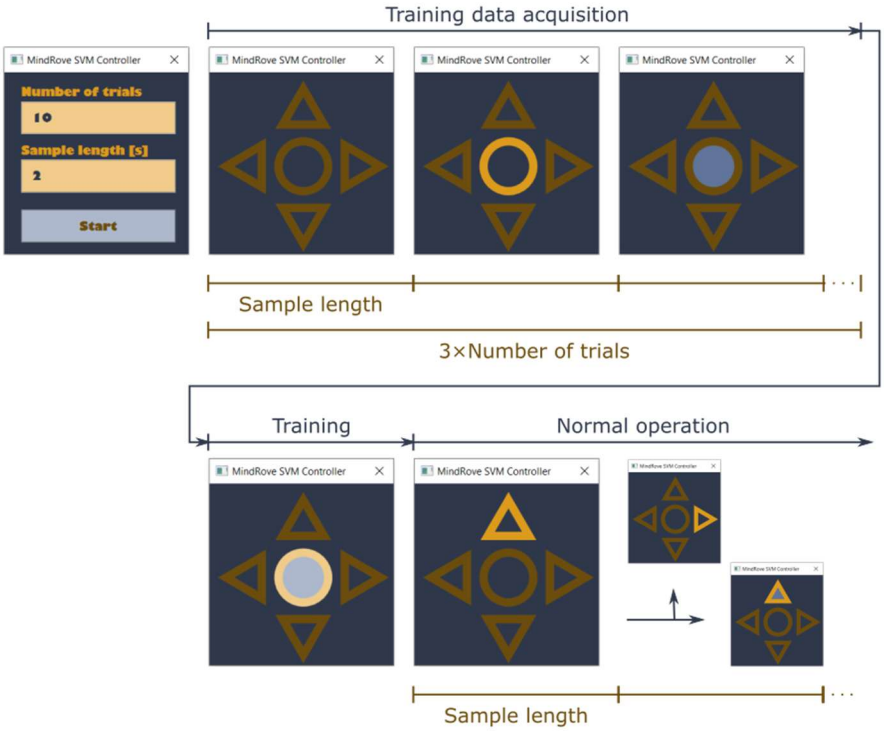

Figure 9. *Data acquisition paradigm of the SVM classifier. Top row, from left to right: initialization window requesting the number of trials per class and the length of the data segments used in seconds. Then the software prompts the user to perform the gestures corresponding to the commands, namely, rest (no objects on the screen are highlighted), right turn (the outline of the central circle is highlighted in yellow), and forward movement (the fill of the circle is highlighted in blue). The actual order of the prompts is randomized. Bottom row, left to right: after sampling collection has ended, classifier training takes place (the outline and fill of the central circle is highlighted in light colours), then the software starts to classify incoming samples, indicating the active direction (with the outline of the corresponding arrow highlighted in yellow) and actual movement (with the fill of the arrow highlighted in blue). Source: own figure from (130).*

DOI:10.14753/SE.2025.3078

3.2. Machine learning methods

3.2.1. Artificial neural networks

Artificial neural networks (ANNs) consist of simple mathematical structures called artificial neurons or perceptrons that are modelled after neurons in the mammalian brain, particularly, after that of the retina (136). A neuron is defined according to Equation [3] (137), where x_i represents the input, w_i the weights of the input (roughly corresponding to synaptic weights), b a bias term (a representation of the firing threshold) and $\sigma(\cdot)$ a suitable activation function that simulates the actual firing of the neuron. This latter is most commonly some nonlinear function such as a rectified linear unit (ReLU) or a sigmoid (138,139). A schematic depiction of an artificial neuron and a few common activation functions are illustrated in Figure 10.

$$y = \sigma\left(\sum_{i=1}^{n}(w_i \cdot x_i) + b\right) \qquad [3]$$

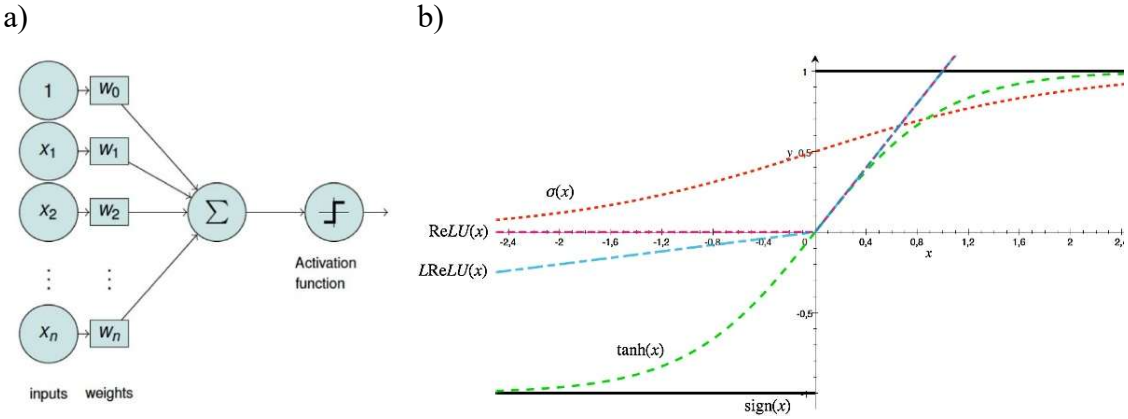

Figure 10. *a) Schematic depiction of an artificial neuron. Note that the bias term is represented by a w_0 weight. b) Plot of few frequently encountered activation functions such as the rectified linear unit (ReLU), its leaky version (LReLU), classical sigmoid (σ), hyperbolic tangent (tanh) and sign functions. Source: (140).*

In a general ANN (multilayer perceptron, multilayer feedforward network, or fully connected network), neurons are organized into layers. A neuron in a given layer gets all neuronal output from the previous layer (sensed through the weights) and transmits its own output to all the neurons in the following layer (137). Layers stacked between the input and the output layers are called hidden layers. According to the universal

approximation theorem, feedforward networks having at least one hidden layer are able to approximate any Borel measurable function in finite dimensional spaces to any desired degree of accuracy (141). This is achieved by tuning the weights and biases of the network to minimize some metric of the error between the output of the network for a given input and the desired output (ground truth) (138,139,142). The metric is called the objective, cost or loss function. Popular examples include the mean squared error, mean absolute error, cosine proximity and categorical crossentropy. Weight values get updated using a suitable optimizer method utilizing the error backpropagation algorithm (138,143). Stochastic gradient descent (144), AdaGrad (145), AdaDelta (146) or RMSprop (147) are optimizers used widely.

The method belongs to the supervised learning paradigms, i.e. assumes that a training database of sufficient size and quality are accessible (139). In other words, appropriate input-output pairs covering all relevant features of the relation to be implemented are essential for good performance in practice. Note that there also are ANN-based methods that adhere to unsupervised or reinforcement learning paradigms but they are out of the scope of this work (138).

Networks containing only fully connected layers are infrequently encountered in practice, due to the exceedingly large number of trainable weights. Contemporary architectures commonly exploit some dependencies in data that introduce some form of sparsity into the network (138). Convolutional layers, specifically, observe only a small fraction of their input at once through filtering kernels that shift over the input. Kernels can extract local features, e.g. edges in a picture (46,139). Concatenated convolutional layers in deep CNNs, however, are able to assemble these simple shapes into more complex ones, similarly to the hierarchical processing in the visual cortex (138,139). Thus, the full network is capable of recognizing instances of specific object classes, e.g. human faces on photographs (139). Down-sampling of data between layers is also a common practice. The most popular implementations of down-sampling are stride and pooling. During slide, the kernel ‘jumps’ between input positions, not evaluating the input in all of them, while pooling layers aggregate segments of the layer output (46,138). The principle of convolution and pooling are illustrated in Figure 11.

Despite of these methods, the great number of tunable parameters (capacity) compared to training data still poses a problem. Networks tend to memorize certain properties of the

training set that leads to insufficient generalization, i.e. unpredicted behaviour when previously unknown data is presented (137,138). This phenomenon is called overfitting and can be avoided by using various regularization methods. Data augmentation is an artificial means to introduce new instances into the training database, increasing its volume and variability. This is achieved by e.g. adding noise or constant offset to existing samples, stretching, rotating or mirroring them. Validation with early stopping involves the benchmarking of the network on a dataset separate from both the training and test sets (i.e. the validation dataset) during optimization (138). Training stops upon reaching e.g. the minimum error or the maximum accuracy on the validation dataset (137). Dropout eliminates random parameters during training, temporarily lowering the complexity of the network (138). Incorporating regularization terms into the objective function in the form of e.g. L1 or L2 penalties is also a common practice (138).

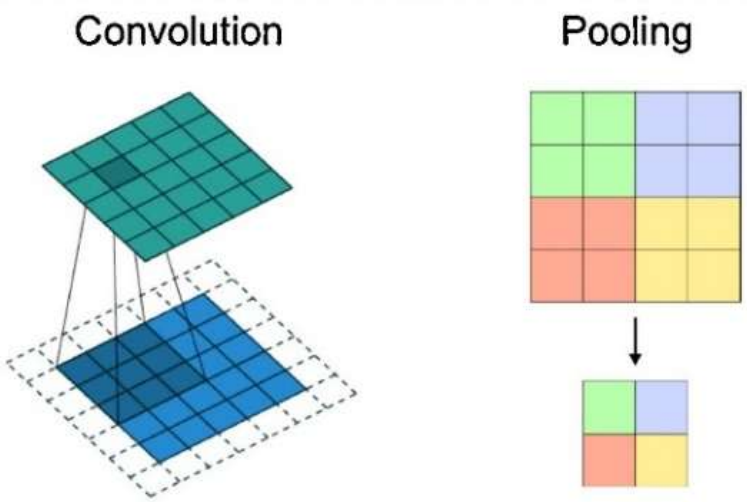

Figure 11. *Convolution and pooling. Left (convolution): a 3 × 3 kernel slides over the input image (blue array) and produces a single pixel in the output (green array). In this example, padding (i.e. filling with zeros) is used to keep the input size constant between the layers. Right (pooling): parts of the layer output are aggregated (areas of the same colour, corresponding to one pixel in the output) to decrease its size. Average and maximum pooling are most common in practice. Source: (140).*

Since CNNs are remarkably suited for processing grid-like input such as images and time-series data (138,139,148,149), most importantly, in medical fields (150–154) (note that these articles were available prior to the beginning of our research), this algorithm was chosen as the base of our spike classifier/predictor solution (46). The models were

implemented in Python (155), using high-level DL library Keras paired with a TensorFlow (156) backend. For array manipulation and confusion matrix computation, Numpy (157) and scikit-learn (158) was applied, respectively.

The overall architecture of the network was the same for all implementations (depicted in Figure 12), with variations regarding the hyperparameters of the classifiers for 2D and 3D data (for details, see Table 2).

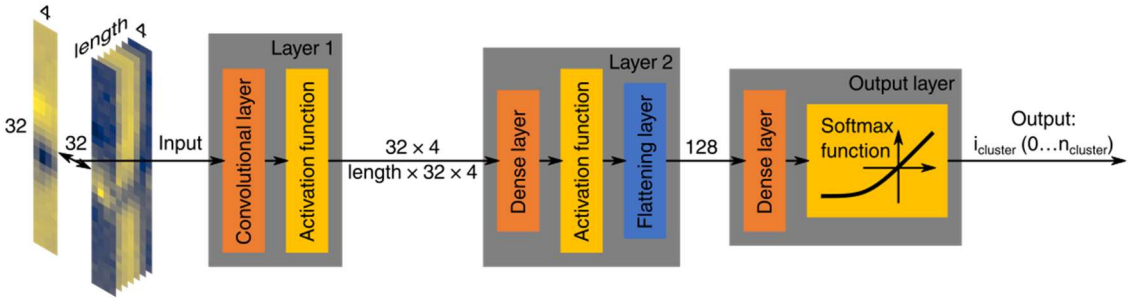

Figure 12. *Structure of the CNN used for classifying frames and timeslots. Out of the two possible input types, the network is able to handle one. The first network layer contains a convolutional Keras layer, the two consecutive network layers dense Keras layers. The activation function of the first two blocks was chosen linear during preliminary tests, the output softmax. Source: own figure from (46); © IOP Publishing. Reproduced with permission. All rights reserved.*

Table 2. *Differing hyperparameters of the two- and three-dimensional classifier networks resulted from preliminary tests. Source: own table from (46); © IOP Publishing. Reproduced with permission. All rights reserved.*

	2D network	***3D network***
Kernel size	16 × 2	Timeslot length × 4 × 2
Optimizer	Adadelta	RMSprop
Loss function	Cosine proximity	Mean squared error

The model consisted of three consecutive layers, with a convolutional layer as the first one and two dense layers at the later stages. Between the layers, as a means of regularization, batch normalization was applied. A network of such a simple structure was chosen considering the presumably simple shape of action potentials in the representation images, and training time. Hyperparameters, such as the type of the activation functions, kernel size, loss and optimizer, were determined experimentally. For hidden layers, linear

activation function was chosen regarding sorters and the 2D predictor, and ReLU regarding the 3D predictor. The activation of the output was softmax in all cases. The padding was chosen ‘same’ to keep input dimensions constant between the layers.

Data were partitioned into training, validation and test data using a 50/25/25 % split for sorters and the 2D predictor, and an 80/10/10 % split for the 3D predictor, respectively. In order to address the problem of electrode drift, we applied two different data configurations during the whole process regarding sorters. One experiment involved data in chronological (sequential or deterministic order), in other words, training data consisted of the earliest samples in the recordings, followed by validation, and finally, test data. Samples were randomized within each data pool to help the optimization converge faster. In the other experiment involving fully randomized data, data were shuffled prior to applying the split. Our expectation was that the performance of the classifiers decreases given sequential data since extracellular conditions may change during even the course of the 45-minute recording, but the amount of this decrease was unknown for us.

3.2.2. Support vector machines

In general, support vector machines (SVMs) are a branch of supervised machine algorithms. The main idea behind SVM is that in an n-dimensional space where a dataset is represented, exists an $n-1$ dimensional hyperplane that is able to separate the dataset into two classes with a maximum margin, thus minimizing the possibility of error (159). Data points closest to this hyperplane are called support vectors (160).

The feature space of data that are not separable using a hyperplane may be transformed into one that allows linear separation using the kernel trick (138). Kernels frequently used include the polynomial and the radial basis function (RBF) (159,160). The principle of SVM and the kernel trick is illustrated in Figure 13 a) and b), respectively.

Although the method was originally meant to separate two classes, it is possible to generalize the solution for n-class problems, as well, using multiple one-versus-all or one-versus-one classifiers (161). Additionally, SVM is reported to have good performance on smaller datasets unlike ANNs (162).

DOI:10.14753/SE.2025.3078

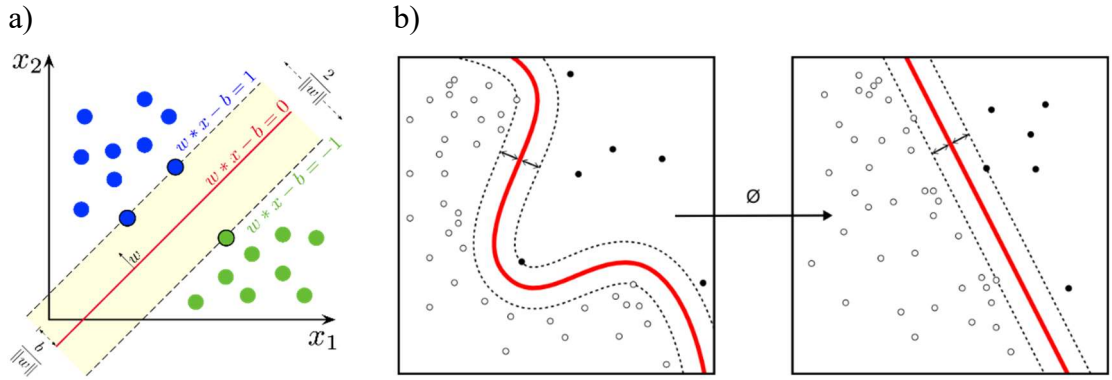

Figure 13. *a) Schematic representation of the support vector machine algorithm in two dimensions. Given two classes (blue and green points), the method is able to compute the hyperplane (red line) that separates the classes with the maximum margin. Support vectors are defined as data points closest to the separating hyperplane (outlined points along the two dashed lines). Source: (163). b) Demonstration of the kernel trick in two dimensions: using a suitable kernel, the parameter space is transformed so that points belonging to two classes (black and white points) become separable using a hyperplane instead of a curve (red line). Source: (164).*

In the project targeting the implementation of an SVM-based classifier for an EEG/EMG-based human–machine interface (130), C-support vector classification (C_SVC, as implemented by OpenCV and interfaced by Emgu CV) was applied. C_SVC allows imperfect classification but penalizes outliers with a multiplier (i.e. C) (165). The kernel was chosen RBF. C and γ were trainable parameters of the system. The training took 1000 iterations to finish. Convergence criterion (threshold) ε was equal to $1.192 \cdot 10^{-7}$.

As both training and test data, samples formatted as described in Section 3.1.2 were applied. Since SVM does not exploit the internal structure of the input, i.e. cannot extract dependencies or patterns across dimensions unlike CNNs, samples were flattened. The length of sample vectors was $n_f \cdot n_{ch}$ where n_f stands for the number of frequency components in the sample and n_{ch} for the number of channels.

The number of training samples per class got implemented as a software parameter freely chosen by the user (the only limitation was that it had to be an integer). During preliminary tests, 10 samples yielded satisfactory results despite of the short course of the training, thus it was set as default value for the algorithm.

3.3. Quality metrics

To assess the performance of the classifiers, accuracy was applied as implemented in Keras and featured in Rácz et al. (46). Accuracy yields the proportion of the correctly classified samples in a given sample pool according to Equation [4], where A stands for accuracy, TP_i the correctly classified instances of cluster i (true positives) of the N classes present in the recording and FN_i the incorrectly classified instances of the same class (false negatives).

$$A = \frac{\sum_{i=1}^{N} TP_i}{\sum_{i=1}^{N} TP_i + FN_i} \quad [4]$$

Accuracy can be given as a dimensionless number between 0 and 1 or as a percentage value.

For a more nuanced view, confusion matrices can be computed, whose rows correspond to the samples actually belonging to a particular class and columns to the classes predicted by a classifier. Ideally, a confusion matrix should contain non-zero elements only in the main diagonal, this assumes a 100 % accuracy. An ideal confusion matrix is shown in Figure 14, along with a matrix that may be encountered in practice (subfigures a) and b)).

To assess the relationship between certain properties of recordings and classification accuracy, we apply the sample Pearson correlation coefficient as implemented in Microsoft Excel [5](166), where r_{xy} stands for the correlation coefficient, x and y the paired data and $\bar{x}$ and $\bar{y}$ their average, respectively.

$$r_{xy} = \frac{\sum_{i=1}^{n}(x - \bar{x})(y - \bar{y})}{\sum_{i=1}^{n}(x - \bar{x})^2 \sum_{i=1}^{n}(y - \bar{y})^2} \quad [5]$$

To qualify the degree of correlation, we use Chan's interpretation (167) (see Table 3).

a)

	C0p	C1p	C2p	C3p	C4p	C5p	C6p	C7p	C8p	C9p
C0r	100	0	0	0	0	0	0	0	0	0
C1r	0	100	0	0	0	0	0	0	0	0
C2r	0	0	100	0	0	0	0	0	0	0
C3r	0	0	0	100	0	0	0	0	0	0
C4r	0	0	0	0	100	0	0	0	0	0
C5r	0	0	0	0	0	100	0	0	0	0
C6r	0	0	0	0	0	0	100	0	0	0
C7r	0	0	0	0	0	0	0	100	0	0
C8r	0	0	0	0	0	0	0	0	100	0
C9r	0	0	0	0	0	0	0	0	0	100

b)

	C0p	C1p	C2p	C3p	C4p	C5p	C6p	C7p	C8p	C9p
C0r	83	1	0	0	7	3	0	0	5	1
C1r	0	99	0	0	0	0	1	0	0	0
C2r	0	3	86	2	0	0	0	5	0	4
C3r	0	0	0	76	0	17	7	0	0	0
C4r	0	0	11	0	89	0	0	0	0	0
C5r	0	0	0	0	0	94	6	0	0	0
C6r	0	4	3	0	0	13	80	0	0	0
C7r	1	0	0	8	0	0	0	91	0	0
C8r	5	0	0	3	0	0	0	0	81	11
C9r	2	1	1	0	7	0	0	0	8	81

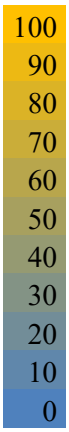

Figure 14. *a) Confusion matrix assuming an ideal classifier given 10 classes. C0r-C9r: real classes, i.e. the class the given sample actually belongs to. C0p-C9p: predicted classes, i.e. the class a given sample belongs to as assumed by the classifier. Values in the cells of the matrix: percentage of the samples of a given class belonging to a particular class as assumed by the classifier (note that the sum of each rows equals 100). Values are colour coded according to the legend on the right showing the shades assigned each percentage value. b) Confusion matrix of a classifier having an accuracy of 86 %.*

DOI:10.14753/SE.2025.3078

Table 3. *Interpretation of the Pearson correlation coefficient. Source: (167).*

r_{xy}	**Interpretation**
±0	None
±0.1	Poor
±0.3	Fair
±0.6	Moderate
±0.8	Very strong
±1	Perfect

DOI:10.14753/SE.2025.3078

4. Results

4.1. CNN-based spike sorters

The performance of 2D and 3D classifiers were evaluated on all datasets. During each run, four instances of the classifiers were trained, employing a combination of a certain norming range ([0, 1] or [−1, 1]) and batch normalization (with/without). The four accuracy values were then averaged. Average accuracies for each recording and sorter are displayed in Figure 15.

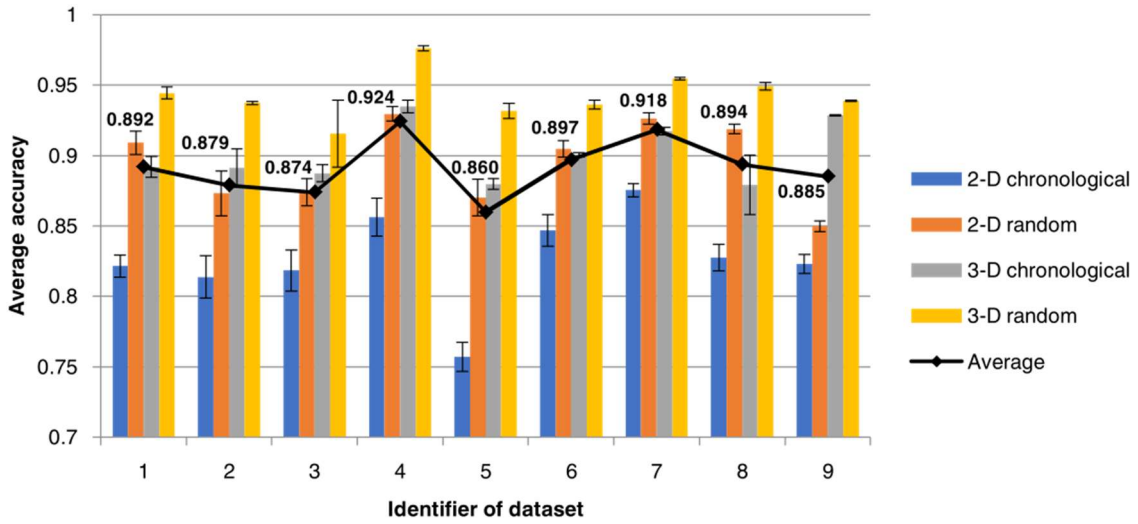

Figure 15. *Effect of the data type (i.e. 2D and 3D) and the order of the training, validation and test samples (i.e. chronological and random) on the classification accuracy with respect to the recordings. Source: own figure from (46); © IOP Publishing. Reproduced with permission. All rights reserved.*

Performance of 3D sorters (92.2 %) was greater than that of their 2D counterparts (82.7 %) by a margin of 6.109 % on average. This is not surprising given that they utilize more information: 3D convolutional kernels are able to extract spatial and temporal features at the same time. In accordance with our expectations, the performance of deterministic sorters (86.4 %) was inferior to the ones trained on fully randomized data (91.9 %) by a 5.493 % margin on average. Interestingly, the performance of 2D random and 3D deterministic sorters were nearly identical. For details, see Figure 16. Confusion matrices illustrating the performance of the 3D sorter with respect to data order for Recording 4 are shown in Figure 17.

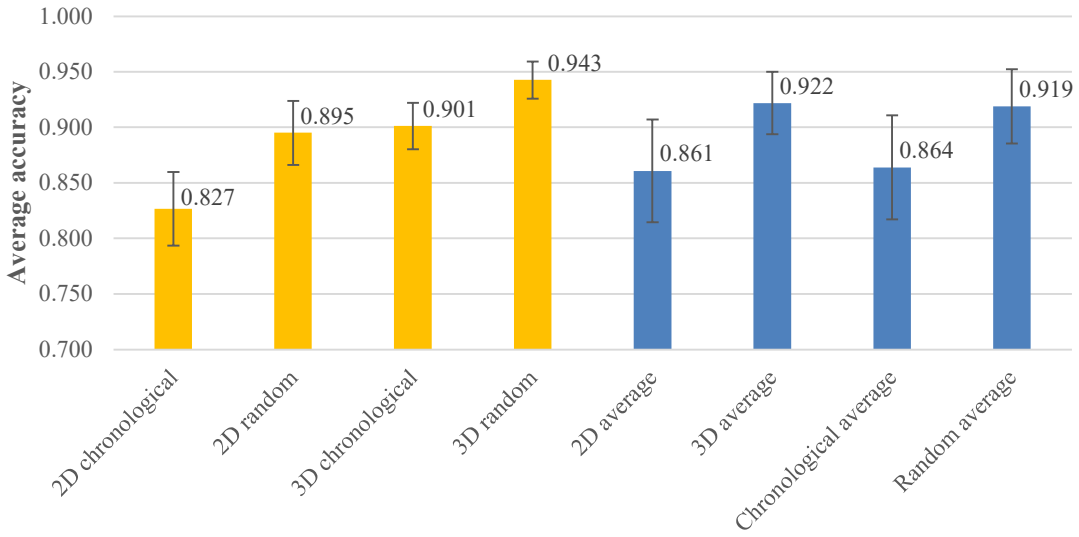

Figure 16. *Yellow bars: average performance of the sorters across recordings. Blue bars: average performances across sorter type (2D or 3D) and training/validation/test database type (deterministic or random). Averages are shown with their respective standard deviations.*

We assessed the relationship between the following recording properties and classification accuracy using the sample Pearson correlation coefficient: the number of clusters (N) within the recording, peak-to-peak amplitude of the clusters (A_{pp}), cluster quality in the form of isolation distance (ID) and the refractory period violation rate (RPVR). The results of this analysis can be found in Table 4.

We could find a fair to very strong positive relationship between average ID and accuracy for all classifiers but most prominently 2D sorters. A fair to very strong negative relationship was also present between both the average and SD of the RPVR and accuracy for all classifiers but most importantly sorters employing fully randomized data. A fair to moderate negative relationship was also shown between N and accuracy for all classifiers. Also, 2D sorters exhibited more sensitivity (fair positive correlation) to the SD of cluster ID than their 3D counterparts (poor positive correlation). The accuracy of 3D classifiers also shows a fair negative correlation with average A_{pp} and also with its SD in contrast with the poor positive, and for SD, weaker negative relationship displayed by 2D sorters. The effect of the refractory period violation rate is particularly interesting because this rate is usually kept considerably low (~ 0.1 %) yet seemingly even these small values

DOI:10.14753/SE.2025.3078

have a detrimental effect on classification accuracy. A scatter plot demonstrating the effect of its average and SD on the average accuracy of all sorters is shown in Figure 18.

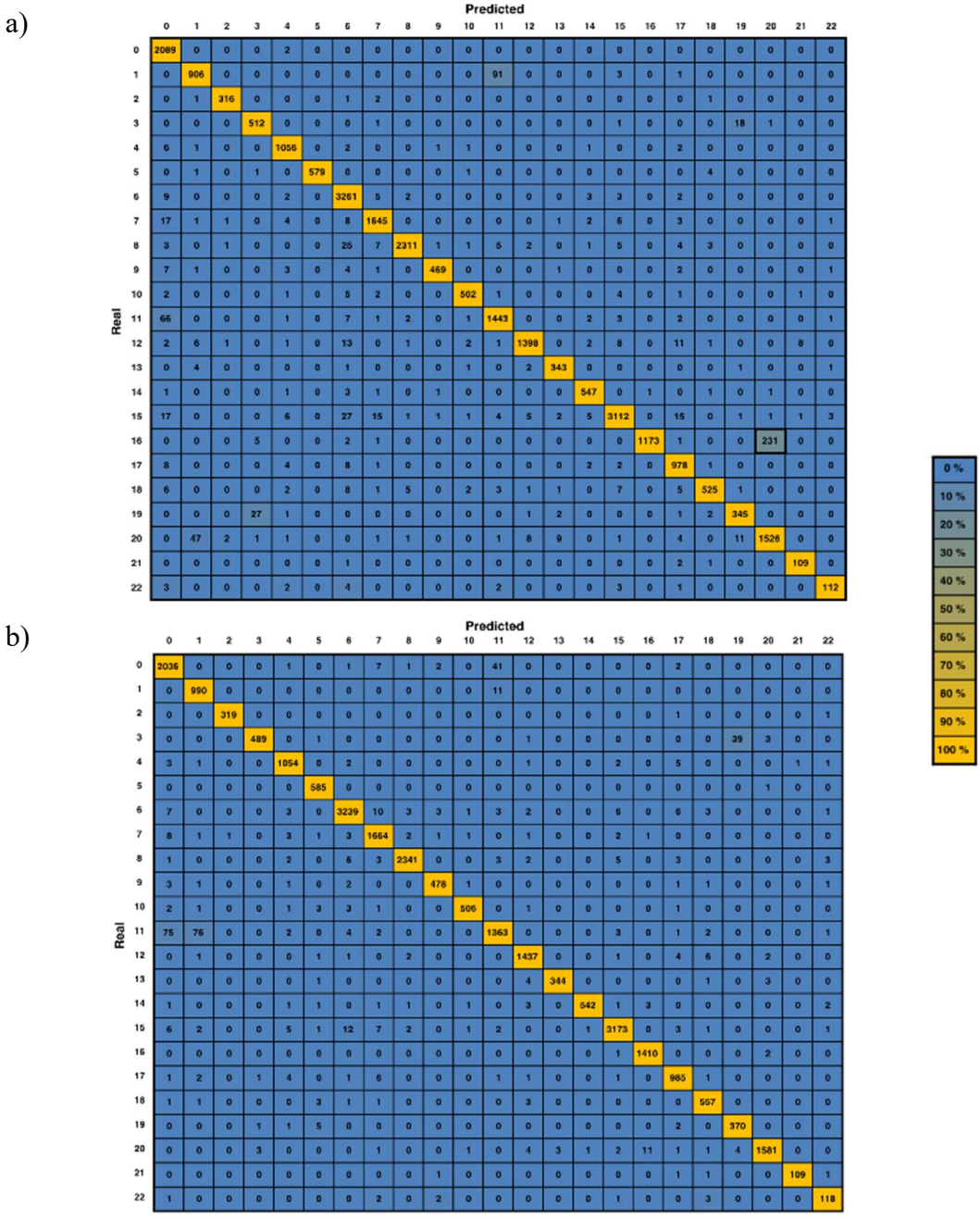

Figure 17. *Confusion matrix of the 3D classifier for one of the recordings (Recording 4). Training, validation and test were performed using a) chronological, b) randomized data. Rows: actual firings of a particular neuron. Columns: neuron clusters predicted by the sorter. The misclassified spikes exceeding the 10% of the total number of APs of a particular neuron are marked by thick borders. Source: own figure from (46); © IOP Publishing. Reproduced with permission. All rights reserved.*

DOI:10.14753/SE.2025.3078

Table 4. *Correlation of the properties of recordings and sorter accuracy with respect to classifier type. Values corresponding to moderate relationship are set in bold. Values denoting a strong relationship are indicated with black border. N: number of clusters, A_{pp}: peak-to-peak amplitude, ID: isolation distance, RPVR: refractory period violation rate.*

	N	App [µV]		ID		RPVR [%]	
		Mean	SD	Mean	SD	Mean	SD
2D deterministic	-0.574	0.075	-0.105	**0.811**	0.578	**-0.618**	**-0.692**
2D random	-0.583	0.003	-0.306	**0.668**	0.415	**-0.764**	**-0.776**
3D deterministic	-0.499	-0.423	-0.430	0.363	0.133	-0.517	-0.540
3D random	**-0.645**	-0.336	-0.571	0.506	0.074	**-0.916**	**-0.746**
2D average	**-0.634**	0.046	-0.217	**0.816**	0.551	**-0.752**	**-0.802**
3D average	**-0.627**	-0.427	-0.547	0.474	0.118	**-0.771**	**-0.702**
Deterministic average	**-0.602**	-0.129	-0.254	**0.704**	0.449	**-0.639**	**-0.699**
Random average	**-0.649**	-0.130	-0.433	**0.652**	0.310	**-0.879**	**-0.820**
Average	**-0.696**	-0.143	-0.375	**0.760**	0.429	**-0.838**	**-0.843**

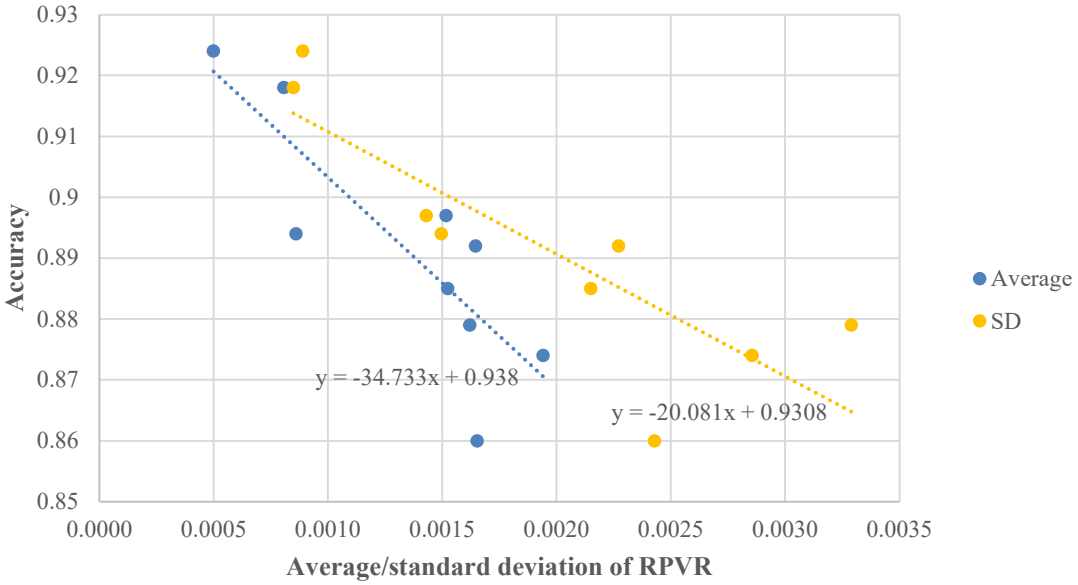

Figure 18. *Effect of the average and standard deviation of the refractory period violation rate on the average accuracy of classifiers. Trend lines (provided with defining expressions) are indicated using dashed lines.*

DOI:10.14753/SE.2025.3078

4.2. CNN-based spike predictors

Performance of the predictors was evaluated using a single recording (Recording 4). During the training/validation/test process, only sequential data were used.

Data fed into the 2D sorter/predictor was formatted similarly to the input for other 2D sorters with respect to norming and batch normalization (described in Section 4.1). Regarding the 3D predictor, only norming between 0 and 1 was applied. Characteristic accuracy curves for the 2D and 3D predictors are shown in Figure 19 and Figure 20, respectively.

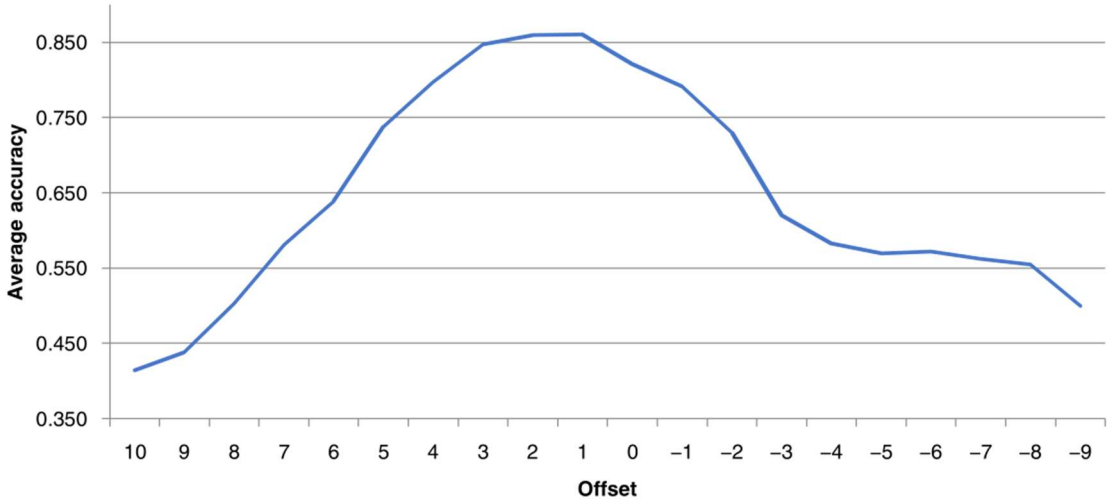

Figure 19. *Accuracy of the 2D sorter/predictor with respect to the offset. Note that offset is positive when the frame in question precedes the mark, thus data points are in chronological order. Source: own figure from (46);*

We encountered the best average 2D predictor performance at an offset of 1 (with an 86.1 % accuracy), closely followed by 2 and 3 (86.0 % and 84.7 %, respectively) as these frames are close to the peak of the activity. As the classifier shifts backwards along the falling edge of the spike, its accuracy monotonically decreases, reaching 50 % at an offset of 8 (50.3 %), that is, 0.35 milliseconds before spike peaks (given a sampling rate of 20 kHz and assuming the peak to be at an offset of 1).

DOI:10.14753/SE.2025.3078

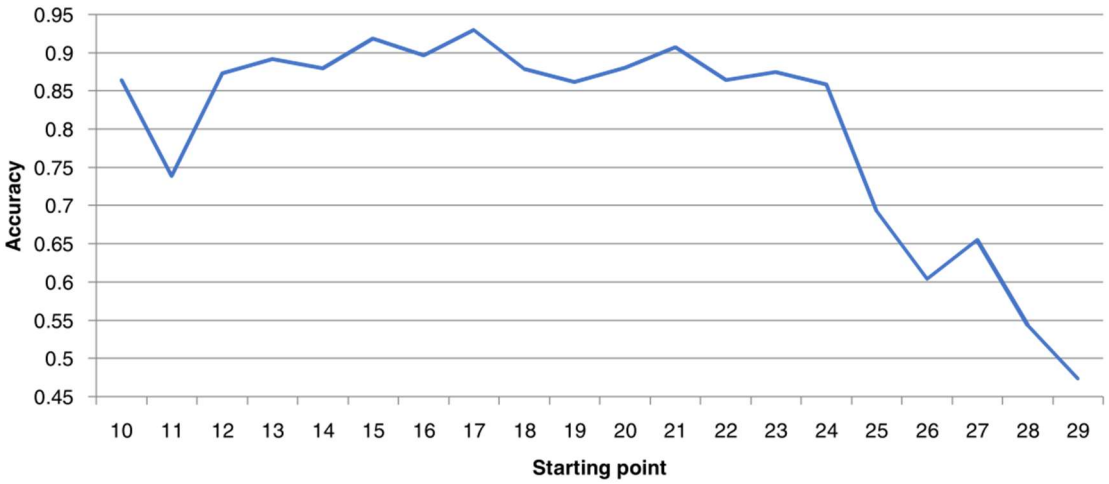

Figure 20. *Accuracy of the 3D sorter/predictor with respect to the starting point. Source: own figure from (46);*

The performance of the 3D predictor oscillates around 90 % until a starting point of 24, where it starts to decline quickly with classification accuracy dropping to 50 % (54.3 %) at 28. Note that the last sample including the spike peak is at a starting point of 21, thus the sorter operates using timeframes containing only the falling edge of the activity from this point onwards. At the crucial point of 28, the last frame of the sample precedes the peak of the AP by 7, thus yielding a 0.35-millisecond margin identically to its 2D counterpart. This possibly signifies that no additional information can be extracted from LFP and/or the activity of the neighbouring neurons that may bear some relevance to the classification of the upcoming AP. Average spike waveforms of Recording 4 are displayed in Figure 21 along with a mark denoting the position of the crucial frame and the number of activities in each cluster.

DOI:10.14753/SE.2025.3078

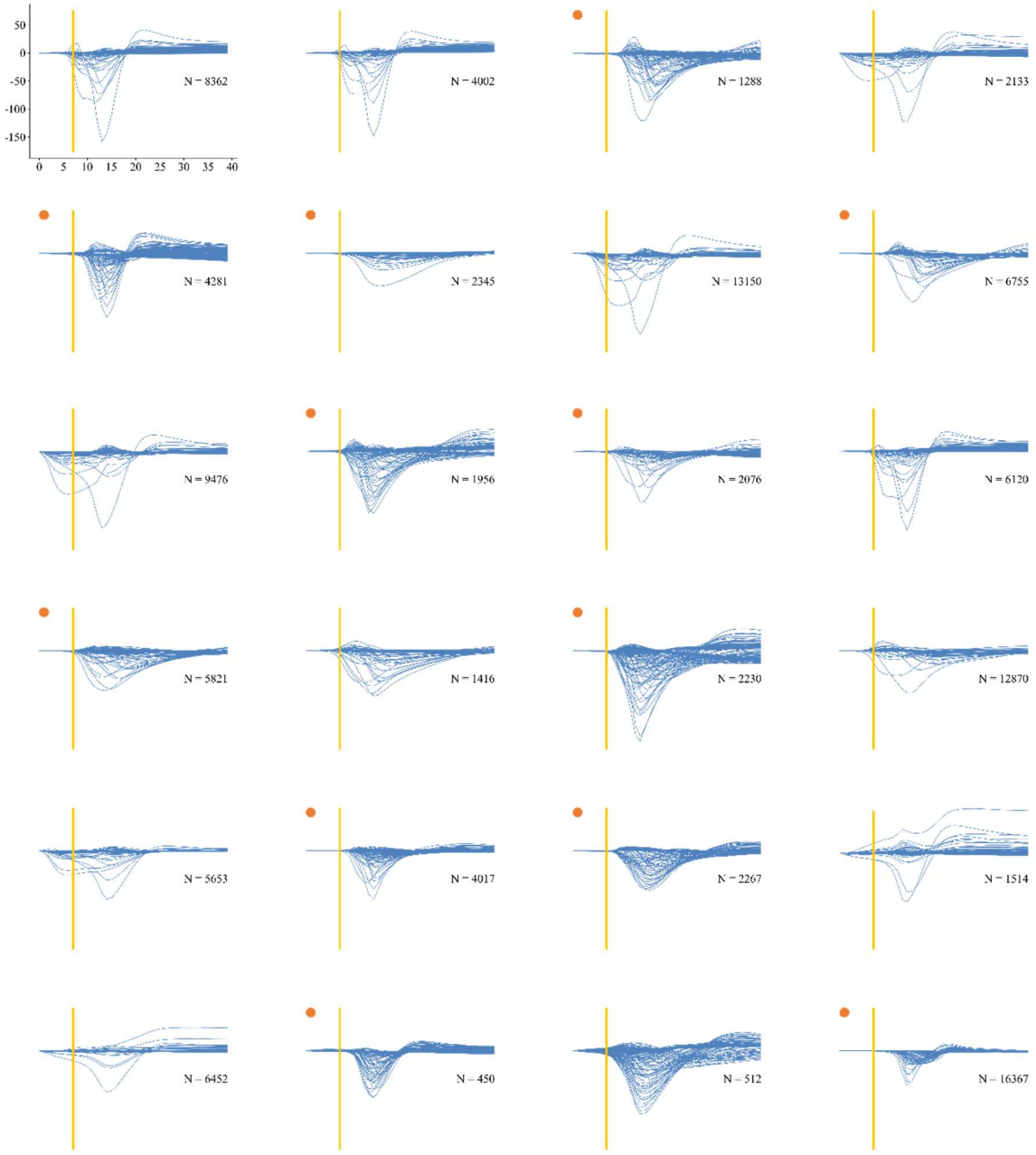

Figure 21. *Single unit averages of Recording 4 with a yellow bar denoting the position of a frame having an offset of 8 (that precedes spike peaks by 0.35 milliseconds corresponding to 50 % classification accuracy) and the number of activities (N). Scales are the same for all subplots, thus they are displayed only in the top left subplot. The vertical scale corresponds to voltage in microvolts, the horizontal to time in samples. Time windows for the subplots start from 15 frames before KiloSort marks and end 25 frames after them, thus they span 40 frames, i.e. 2 milliseconds. Clusters where the crucial frame visibly precede the initiation of spikes are marked with an orange dot.*

DOI:10.14753/SE.2025.3078

4.3. SVM-based EEG/EMG classifier

Tests of the classifier were performed by one of the authors of (130) with the EEG headset placed above the parietal cortex. As control commands, eyebrow rising (switch direction) and chewing (steer forward) were applied. A run consisted of 20 two-seconds long training samples per class. As a result, we encountered an 86.67 % classification accuracy. A corresponding confusion matrix is shown in Figure 22.

Interestingly, the majority of incorrect classification pertains to rest. This is possibly due to the subject making unintentional movements (e.g. blinking) in this situation.

Pattern following tests involving the robot model and the actual hardware in real world also were conducted with an average error of 12.39 %. A detailed description can be found in Rácz et al. (130) (their discussion is out of the scope of this work[2]).

	0	**1**	**2**
0	13	7	0
1	0	19	1
2	0	0	20

Figure 22. *Confusion matrix of the SVM classifier on data recorded by an experienced user. Classes 0, 1 and 2 correspond to rest, right turn and forward steering, respectively. Source: own figure from (130).*

[2] The right to report them in detail belongs to my colleagues at Óbuda University (mainly to Erick Noboa) who implemented the tests.

DOI:10.14753/SE.2025.3078

5. Discussion

So far, this work has reported about the implementation of two BCI/HMI systems, one of which utilizes neuronal signals recorded by extracellular probes, while the other uses EMG artifacts captured by an EEG headset. Although both their signal acquisition and classification methods are remarkably different, the purpose of these interface is the same on a fundamental level, i.e. enabling the manipulation of the environment via computers and/or robotic actuators using signals of biological origin.

Our BCI incorporated an ANN, more specifically, CNN-based AP classifier, whose architecture was also applied for predicting spikes. Particular instances were implemented utilizing data of different formats, i.e. frames (2D data) and timeslots (3D data containing spatiotemporal variations). Although the architecture the implemented model was fairly simple, it exhibited considerably good classification performance even in the presence of LFP activity: except for Recording 5, accuracy was above 80 % for all sorter types. Not unsurprisingly, the performance of sorters using 3D data was 6.109 % better than 2D classifiers. Also, examining the relationship between accuracy and cluster quality metrics, we observed that 2D classifiers are more sensitive to the mean and the SD of ID. Since ID is a metric related to spatial separability of the clusters, it is reasonable to assume that it effects classifiers exploiting only spatial dependencies more. Interestingly, while 2D sorters show a weak positive correlation with mean A_{pp}, 3D sorters exhibit a stronger (fair) negative relationship. On the other hand, they are more sensitive to A_{pp} SD—in other words, in order to be classified more accurately, spikes in 3D samples should be evenly small in each cluster. These effects are even harder to explain due to the samples being normed independently, i.e. scaled one-by-one between well-defined boundaries. Note that there were not significant differences between results for the two ranges, thus we averaged them together. The same applies to the presence of batch normalization. A possible explanation is that in case of greater A_{pp}s, finer details that manifest only across the temporal dimension (such as edge shapes) are scaled down during norming. This putatively makes classification based on these features harder.

Temporal electrode drift was also assessed by applying training, validation and test databases randomized or keeping sequential order. We could indicate the detrimental effect of drift on classification accuracy: the performance of sorters trained on sequential

data was 5.493 % worse, showing that the test database contained samples more dissimilar to training data. Moreover, 2D sorters of this type tended to be more sensitive to the mean and SD of ID than their random counterparts. The rationale behind this might be that the greater variety of random samples may introduce additional robustness in the system, defining cluster boundaries more accurately. On the other hand, random classifiers proved to be more sensitive to the mean and SD of RPVR, contradicting this hypothesis. Additionally, specifically 3D classifiers exhibited greater sensitivity to N. A possible explanation for this may be that during sample shuffling, the proportion of cluster samples also changes randomly, potentially leading to underrepresentation of certain clusters in the training database.

When our results were published, there were only a handful ANN-based spike sorting solutions reported in the literature. Additionally, the stages where ANNs were applied and the utilized learning paradigm makes comparison hard for these early articles: Lee et al. (168) used ANN only for spike detection, Saif-Ur-Rehman et al. (169) for artifact filtering, Yang, Wu and Zeng (170) for feature extraction, embedding principal component analysis into an ANN. Other methods utilized unsupervised ANNs at the classification stage, employing generative adversarial networks (171) and attention-based spiking neural networks (57). Since 2020, multiple models have been implemented utilizing ANNs/CNNs as actual sorters. Li et al. (172) applies a model based on 1D convolutional layers, achieving higher than 99 % accuracy on synthetic and experimental datasets containing at most 5 single units. Yi et al. (173) combines 2D and 1D convolution, yielding 93.1 % accuracy on a neural recording containing 27 single units. The model also got deployed onto a field programmable gate array-based hardware, where the reduced weights allowed classification with 86.1 % accuracy. Liu et al. (174) fuses long short-term memory (LSTM) based layers with convolutional ones, exhibiting approximately 99.9 % accuracy on non-overlapping spike samples and 99 % on overlapping ones. Saif-ur-Rehman et al. (175) incorporates the detector module reported in Saif-ur-Rehman et al. (169) into a model based on a CNN that utilizes k-means with cluster accept or merge algorithm for determining the number of outputs. The sorter achieved 90.2, 87.7 and 91 % accuracy on data containing 1, 2 and 3 single units, respectively. Using data from another recording session, the sorter yielded 90.5 and 92.78 % accuracy for data containing 1 and 2 clusters, respectively. Zhang, Gao and

Wang (176) classifies overlapping spikes using a CNN fed with the Log-Mel spectrogram of activities with 95.74 % accuracy. Similarly to Liu et al. (174), Wang et al. (177) combines CNN and LSTM in a classifier reaching 97.17 % accuracy on a synthetic dataset containing 3 single units. Li et al. (178) applies reinforcement training for a CNN that yields 98.4 and 97.9 % accuracy on synthetic and experimental datasets containing 6 and 2–5 single units, respectively. Meyer, Samann and Schanze (179) takes a step back to the basics by applying a fully connected ANN for classifying, achieving an accuracy of 98.04 % on synthetic data containing 3 clusters and an average accuracy above 96.74 % on experimental data containing 4 single units. For a more extensive review, see Meyer et al. (180). Considering that our experimental data contained significantly more single units than the aforementioned studies report on, while proposing a reasonable accuracy, I personally suppose that our results are still not completely outdated (despite of the four years that passed since the publication of our paper).

Despite of this, our research was not without limitations. Classifiers were fit to data recorded in experimental animals, making the translatability to humans questionable. Secondly, we did not possess a real ground truth, contrarily to e.g. juxtacellular or intracellular recordings. Thus, our results are limited to the quality of prior clustering. We were notified that the database applied in our studies (46,59) has recently been reclustered using KiloSort 2 and was shared at ARP Data Repository (181). It was, however, out of the scope of this work to repeat the measurements.

Our spike prediction solution was unique in the sense that (up to my recent knowledge) no attempt has been made to classify APs before they are fired. In a loosely related work (182), the activity of neurons in the lateral geniculate nucleus is predicted based on visual input. In our research, frames and timeslots were classified using a slightly modified version of our CNN classifier. Data were sequentially shifted back in time so that the predictor could perceive less and less from the APs' initial sections. We reached a point of classification accuracy of 50 % using samples containing information 0.35 milliseconds before activity peaks using both sorters. Note that we did implement sorting, not detection, thus at this point, 50 % of all activities get classified correctly, well above chance level: since the recording contains 24 single units, this would be 4.167 %. We supposed that it meant that information preceding the critical frame does not contain additional exploitable information (e.g. from LFP). Looking at Figure 21, we can see that

this assumption is partially justified, since the start of 12 of the 24 average waveforms are preceded by the marking denoting the frame in question by a visible margin. It is worth noting, however, that the majority of activities belong to the other clusters (i.e. that exhibit characteristic properties this early). Additionally, the applicability of this solution is greatly limited by the variability of the temporal distance between AP initiation and peak, i.e. that activities of the different clusters are at a different stage at a given time.

Our HMI system was based on a mobile robot developed by students and an off-shelf lightweight EEG headband, with a C# application powered by an SVM-based classifier implementing the interface in-between. Although the headset was optimized for being utilized in event-related potential-based BCIs (183), in order to use the system for demonstration purposes, we decided to utilize EMG artifacts for control. The application of eyebrow raising and chewing yielded a satisfactory classification performance (86.67 %) in practice.

Our approach was not entirely unfounded as there are a number of robot control applications that utilize facial gestures instead of—or supporting—pure EEG signal. The system presented in Boubchir et al. (184) applies blinking and chewing for generating error potentials during the operation of an ssVEP BCI. Quadratic time-frequency-domain features generated from these potentials can subsequently be used for fine-tuning the software of the actuator. Using an SVM classifier, error potential features can be detected with accuracies between 86–96 %. The solution from Kuffuor and Samanta (185) combines MI (right and left hand) and facial expression (right and left eye wink and smile) commands for an optimally accurate control. Signals are also recorded using a portable EEG headset (Emotiv EPOC+) and drive a mobile robot. During data processing, CSP and independent components are extracted besides power spectral density patterns. Features are classified using SVM. The combined method yields a 96 % average accuracy. Another system (186) utilizes both ERS/ERD patterns (right and left hand MI) and EOG signals (single and double blink) for driving a robotic hand, with the two types of signal implementing different control commands. CSP resulting from EEG signals were fed to an SVM classifier, providing a 91.1 % average accuracy. A simple system similar to ours (187) also utilizes EMG artifacts (eye blinks and left and right eye winks) for the control of an Arduino-based mobile robot, recorded by an Emotiv headset. Classifying band power features using SVM, the authors could achieve a 100 % accuracy.

DOI:10.14753/SE.2025.3078

Another project reported in Lu et al. (188) uses brow movements and smirks for controlling a robotic arm with 79.5 % accuracy, exploiting CSP with SVM. Taking these results into account, our method is also can be considered a valid solution that may have a place in the literature of the niche.

DOI:10.14753/SE.2025.3078

6. Conclusions

This work described the implementation of a SUA-based BCI software and an HMI for mobile robot control. The former solution applies a CNN-based classifier with spike sorting and activity prediction capabilities while the latter utilizes EMG artifacts corresponding to facial gestures and employs an SVM classifier.

We have proven that these two systems are capable, exhibiting performance that can compete with prior and recent results in the literature. Our 2D and 3D sorters yielded 86.1 and 92.2 % average accuracy, respectively. The robot controller exhibited 86.67 % accuracy during preliminary investigations and the complete system showed 12.39 % error in a pattern following test.

In time, our achievements got also accepted by the scientific community. Table 5 lists the number of citations for the publications related to this thesis written (or co-written) by the author. The robot controller software fit to an Arduino-based tracked Bluetooth mini-robot and used with other MindRove devices (such as the EMG armband or an alternate EEG headset model named bright) was featured many times on public events such as the European Researchers' Night (189) or the Brain Awareness Week (190).

Table 5. *Number of citations according to indexing databases for the articles related to the topic of this thesis written or co-written by the author. MTMT: Hungarian Scientific Bibliography (Magyar Tudományos Művek Tára). WoS: Web of Science. S: Scopus. GS: Google Scholar. Data accessed on 06. 02. 2025.*

Article	Year	MTMT	WoS	S	GS
(46)	2020	48	45	48	64
(59)	2021	5	6	7	11
(131)	2021	9	6	8	9
(130)	2022	16	8	11	16

DOI:10.14753/SE.2025.3078

7. Summary

Introduction: brain–computer interfaces (BCIs) are systems that implement connection between an individual's brain and an actuator. The application of these interfaces may improve the quality of life in patients with motor deficiencies, such as people with tetraplegia or locked-in syndrome resulting from amyotrophic lateral sclerosis or brainstem stroke. There is a variety of signal modalities, paradigms and signal processing methods applied in the field. This thesis describes the implementation of a BCI software that utilizes neuronal activities (spikes) and a human–machine interface that exploits myoelectric artifacts in electroencephalography (EEG) signals for driving a mobile robot.
Methods: our BCI solution operated on data obtained from rat somatosensory cortex. 2D and 3D activity samples were extracted from the recordings. A convolutional neural network was implemented for automatic classification. We also assessed the effect of electrode drift on classification accuracy. In addition to spike classification, we attempted activity prediction via taking samples before spike peaks, shifting backwards in each run. Our HMI system was co-developed with Óbuda University as a tool for laboratory demonstrations and public events. It was based on a mobile robot built by students and an off-shelf EEG headband. The device was meant to be accessible, i.e. not needing user training or special capabilities. Therefore, we decided to utilize electromyographic artifacts resulting from facial gestures, instead of motor imagery. Gestures were assigned to control instructions via a support vector machine-based (SVM) classifier.
Results: 2D sorters yielded an average accuracy of 86.1 %. The performance of 3D classifiers was remarkably better, providing 92.2 %. Sorters utilizing fully randomized data also performed better than their counterparts trained, validated and tested on data blocks containing sequential data, with accuracies of 91.9 and 86.4 %, respectively. Spike predictors reached a classification accuracy of 50 % at 8 frames (0.35 milliseconds) before activity peaks. The SVM-based classifier of the robot control application provided an 86.67 % accuracy and an average error of 12.39 % in a pattern following task.
Conclusions: the solutions proved to be useful, with results comparable to prior and subsequent literature. Publications on spike sorting utilize recordings with significantly less single units with similar results. Our robot HMI was successfully applied during public events such as European Researchers' Night and Brain Awareness Week.

8. References

1. Lebedev MA, Nicolelis MAL. Brain-machine interfaces: From basic science to neuroprostheses and neurorehabilitation. Physiol Rev. 2017;97(2):767–837.
2. Kawala-Sterniuk A, Browarska N, Al-Bakri A, Pelc M, Zygarlicki J, Sidikova M, Martinek R, Gorzelanczyk EJ. Summary of over fifty years with brain-computer interfaces—a review. Brain Sci. 2021;11(1):1–41.
3. Lem S. Summa technologiae. Summa technologiae. Kraków: Wydawnictwo Literackie; 1964.
4. Maiseli B, Abdalla AT, Massawe L V., Mbise M, Mkocha K, Nassor NA, Ismail M, Michael J, Kimambo S. Brain–computer interface: trend, challenges, and threats. Brain Informatics. 2023;10(1).
5. Benabid AL, Costecalde T, Eliseyev A, Charvet G, Verney A, Karakas S, Foerster M, Lambert A, Morinière B, Abroug N, Schaeffer MC, Moly A, Sauter-Starace F, Ratel D, Moro C, Torres-Martinez N, Langar L, Oddoux M, Polosan M, Pezzani S, Auboiroux V, Aksenova T, Mestais C, Chabardes S. An exoskeleton controlled by an epidural wireless brain–machine interface in a tetraplegic patient: a proof-of-concept demonstration. Lancet Neurol. 2019;18(12):1112–22.
6. Collinger JL, Wodlinger B, Downey JE, Wang W, Tyler-Kabara EC, Weber DJ, McMorland AJC, Velliste M, Boninger ML, Schwartz AB. High-performance neuroprosthetic control by an individual with tetraplegia. Lancet. 2013;381(9866):557–64.
7. Miller KJ, Hermes D, Staff NP. The current state of electrocorticography-based brain-computer interfaces. Neurosurg Focus. 2020;49(1).
8. Nijboer F, Birbaumer N, Kübler A. The influence of psychological state and motivation on brain-computer interface performance in patients with amyotrophic lateral sclerosis - a longitudinal study. Front Neurosci. 2010;4(JUL).
9. Hochberg LR, Bacher D, Jarosiewicz B, Masse NY, Simeral JD, Vogel J, Haddadin S, Liu J, Cash SS, Van Der Smagt P, Donoghue JP. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. Nature. 2012;485(7398):372–5.
10. Peksa J, Mamchur D. State-of-the-Art on Brain-Computer Interface Technology.

Sensors. 2023;23(13).

11. Schwartz AB, Cui XT, Weber DJJ, Moran DW. Brain-Controlled Interfaces: Movement Restoration with Neural Prosthetics. Neuron. 2006;52(1):205–20.
12. Wolpert DH, Macready WG. No free lunch theorems for optimization. IEEE Trans Evol Comput. 1997;1(1):67–82.
13. Fabietti M, Mahmud M, Lotfi A, Kaiser MS. ABOT: an open-source online benchmarking tool for machine learning-based artefact detection and removal methods from neuronal signals. Brain Informatics. 2022;9(1).
14. Wolpaw JR, McFarland DJ, Neat GW, Forneris CA. An EEG-based brain-computer interface for cursor control. Electroencephalogr Clin Neurophysiol. 1991;78(3):252–9.
15. Kübler A, Kotchoubey B, Hinterberger T, Ghanayim N, Perelmouter J, Schauer M, Fritsch C, Taub E, Birbaumer N. The thought translation device: A neurophysiological approach to communication in total motor paralysis. Exp Brain Res. 1999;124(2):223–32.
16. Levine SP, Huggins JE, BeMent SL, Kushwaha RK, Schuh LA, Rohde MM, Passaro EA, Ross DA, Elisevich K V., Smith BJ. A direct brain interface based on event-related potentials. IEEE Trans Rehabil Eng. 2000;8(2):180–5.
17. Kennedy PR, Bakay RAE, Moore MM, Adams K, Goldwaithe J. Direct control of a computer from the human central nervous system. IEEE Trans Rehabil Eng. 2000;8(2):198–202.
18. Guger C, Schlögl A, Walterspacher D, Pfurtscheller G. Entwurf eines EEG-basierten brain-computer interfaces (BCI) mit standardkomponenten, das unter windows in echtzeit arbeitet. Biomed Tech. 1999;44(1–2):12–6.
19. Obermaier B, Guger C, Pfurtscheller G. Hidden Markov models used for the offline classification of EEG data. Biomed Tech. 1999;44(6):158–62.
20. Müller-Gerking J, Pfurtscheller G, Flyvbjerg H. Designing optimal spatial filters for single-trial EEG classification in a movement task. Clin Neurophysiol. 1999;110(5):787–98.
21. Roberts SJ, Penny WD. Real-time brain-computer interfacing: A preliminary study using Bayesian learning. Med Biol Eng Comput. 2000;38(1):56–61.
22. Lugger K, Flotzinger D, Schlögl A, Pregenzer M, Pfurtscheller G. Feature

extraction for on-line EEG classification using principal components and linear discriminants. Med Biol Eng Comput. 1998;36(3):309–14.

23. Pfurtscheller G, Flotzinger D, Mohl W, Peltoranta M. Prediction of the side of hand movements from single-trial multi-channel EEG data using neural networks. Electroencephalogr Clin Neurophysiol. 1992;82(4):313–5.
24. Peters BO, Pfurtscheller G, Flyvbjerg H. Mining multi-channel EEG for its information content: An ANN-based method for a brain-computer interface. Neural Networks. 1998;11(7–8):1429–33.
25. Kostov A, Polak M. Parallel man-machine training in development of EEG-based cursor control. IEEE Trans Rehabil Eng. 2000;8(2):203–5.
26. Costa EJX, Cabral EF. EEG-based discrimination between imagination of left and right hand movements using adaptive gaussian representation. Med Eng Phys. 2000;22(5):345–8.
27. Haselsteiner E, Pfurtscheller G. Using time-dependent neural networks for EEG classification. IEEE Trans Rehabil Eng. 2000;8(4):457–63.
28. Albahri AS, Al-Qaysi ZT, Alzubaidi L, Alnoor A, Albahri OS, Alamoodi AH, Bakar AA. A Systematic Review of Using Deep Learning Technology in the Steady-State Visually Evoked Potential-Based Brain-Computer Interface Applications: Current Trends and Future Trust Methodology. Int J Telemed Appl. 2023;2023.
29. Xu D, Tang F, Li Y, Zhang Q, Feng X. An Analysis of Deep Learning Models in SSVEP-Based BCI: A Survey. Brain Sci. 2023;13(3).
30. Chapin JK, Moxon KA, Markowitz RS, Nicolelis MAL. Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. Nat Neurosci. 1999;2(7):664–70.
31. Vallabhaneni A, He B. Motor imagery task classification for brain computer interface applications using spatiotemporal principle component analysis. Neurol Res. 2004;26(3):282–7.
32. Kaper M, Meinicke P, Grossekathoefer U, Lingner T, Ritter H. BCI competition 2003 - Data set IIb: Support vector machines for the P300 speller paradigm. IEEE Trans Biomed Eng. 2004;51(6):1073–6.
33. Zhang A, Zhao Y. Phase synchronization analysis and support vector machine for

recognition of mental tasks. Annu Int Conf IEEE Eng Med Biol - Proc. 2005;7 VOLS:5373–6.

34. Glannon W. Ethical issues with brain-computer interfaces. Front Syst Neurosci. 2014;8(JULY).
35. Yuan H, He B. Brain-computer interfaces using sensorimotor rhythms: Current state and future perspectives. IEEE Trans Biomed Eng. 2014;61(5):1425–35.
36. Lewicki MS. A review of methods for spike sorting: The detection and classification of neural action potentials. Netw Comput Neural Syst. 1998;9(4).
37. Rey HG, Pedreira C, Quian Quiroga R. Past, present and future of spike sorting techniques. Brain Res Bull. 2015;119:106–17.
38. Quiroga RQ. Spike sorting. Curr Biol. 2012;22(2).
39. Bod RB, Rokai J, Meszéna D, Fiáth R, Ulbert I, Márton G. From End to End: Gaining, Sorting, and Employing High-Density Neural Single Unit Recordings. Front Neuroinform. 2022;16.
40. Quian Quiroga R. What is the real shape of extracellular spikes? J Neurosci Methods. 2009;177(1):194–8.
41. Baldazzi G, Solinas G, Valle J Del, Barbaro M, Micera S, Raffo L, Pani D. Systematic analysis of wavelet denoising methods for neural signal processing. J Neural Eng. 2020;17(6).
42. Quiroga RQ, Nadasdy Z, Ben-Shaul Y. Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. Neural Comput. 2004;16(8):1661–87.
43. Nenadic Z, Burdick JW. Spike detection using the continuous wavelet transform. IEEE Trans Biomed Eng. 2005;52(1):74–87.
44. Kim KH, Kim SJ. Neural spike sorting under nearly 0-dB signal-to-noise ratio using nonlinear energy operator and artificial neural-network classifier. IEEE Trans Biomed Eng. 2000;47(10):1406–11.
45. Choi JH, Jung HK, Kim T. A new action potential detector using the MTEO and its effects on spike sorting systems at low signal-to-noise ratios. IEEE Trans Biomed Eng. 2006;53(4):738–46.
46. Rácz M, Liber C, Németh E, Fiáth R, Rokai J, Harmati I, Ulbert I, Márton G. Spike detection and sorting with deep learning. J Neural Eng. 2020;17(1).

47. Abeles M, Goldstein MH. Multispike Train Analysis. Proc IEEE. 1977;65(5):762–73.
48. Leibig C, Wachtler T, Zeck G. Unsupervised neural spike sorting for high-density microelectrode arrays with convolutive independent component analysis. J Neurosci Methods. 2016;271:1–13.
49. Pachitariu M, Steinmetz N, Kadir S, Carandini M, Kenneth D. H. Kilosort: realtime spike-sorting for extracellular electrophysiology with hundreds of channels. bioRxiv. 2016;061481.
50. Gerstein GL, Clark WA. Simultaneous studies of firing patterns in several neurons. Science (80-). 1964;143(3612):1325–7.
51. Wood F, Fellows M, Donoghue JRP, Black MJJ, Wood E, Fellows M, Donoghue JRP, Black MJJ, Wood F, Fellows M, Donoghue JRP, Black MJJ. Automatic spike sorting for neural decoding. Annu Int Conf IEEE Eng Med Biol - Proc. 2004;26 VI:4009–12.
52. Kim KH, Kim SJ. Method for unsupervised classification of multiunit neural signal recording under low signal-to-noise ratio. IEEE Trans Biomed Eng. 2003;50(4):421–31.
53. Shoham S, Fellows MR, Normann RA. Robust, automatic spike sorting using mixtures of multivariate t-distributions. J Neurosci Methods. 2003;127(2):111–22.
54. Hulata E, Segev R, Ben-Jacob E. A method for spike sorting and detection based on wavelet packets and Shannon's mutual information. J Neurosci Methods. 2002;117(1):1–12.
55. Wang GL, Zhou Y, Chen AH, Zhang PM, Liang PJ. A robust method for spike sorting with automatic overlap decomposition. IEEE Trans Biomed Eng. 2006;53(6):1195–8.
56. Vogelstein RJ, Murari K, Thakur PH, Diehl C, Chakrabartty S, Cauwenberghs G. Spike sorting with support vector machines. In: Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings. 2004. p. 546–9.
57. Bernert M, Yvert B. An Attention-Based Spiking Neural Network for Unsupervised Spike-Sorting. Int J Neural Syst. 2019;29(8).
58. Rokai J, Ulbert I, Márton G. Edge computing on TPU for brain implant signal analysis. Neural Networks. 2023;162:212–24.

59. Rokai J, Rácz M, Fiáth RB, Ulbert I, Márton G. ELVISort: encoding latent variables for instant sorting, an artificial intelligence-based end-to-end solution. J Neural Eng. 2021;
60. Shoham S, O'Connor DH, Segev R. How silent is the brain: Is there a "dark matter" problem in neuroscience? J Comp Physiol A Neuroethol Sensory, Neural, Behav Physiol. 2006;192(8):777–84.
61. Buzsáki G. Large-scale recording of neuronal ensembles. Nat Neurosci. 2004;7(5):446–51.
62. Fiáth R, Márton AL, Mátyás F, Pinke D, Márton G, Tóth K, Ulbert I. Slow insertion of silicon probes improves the quality of acute neuronal recordings. Sci Rep [Internet]. 2019;9(1):111. Available from: https://doi.org/10.1038/s41598-018-36816-z
63. Claverol-Tinture E, Nadasdy Z. Intersection of microwire electrodes with proximal CA1 stratum-pyramidale neurons at insertion for multiunit recordings predicted by a 3-D computer model. IEEE Trans Biomed Eng. 2004;51(12):2211–6.
64. Wang W, Degenhart AD, Collinger JL, Vinjamuri R, Sudre GP, Adelson PD, Holder DL, Leuthardt EC, Moran DW, Boninger ML, Schwartz AB, Crammond DJ, Tyler-Kabara EC, Weber DJ. Human motor cortical activity recorded with micro-ECoG electrodes during individual finger movements. Proc 31st Annu Int Conf IEEE Eng Med Biol Soc Eng Futur Biomed EMBC 2009. 2009;586–9.
65. Padfield N, Zabalza J, Zhao H, Masero V, Ren J. EEG-based brain-computer interfaces using motor-imagery: Techniques and challenges. Sensors (Switzerland). 2019;19(6).
66. Picton TW. The P300 wave of the human event-related potential. J Clin Neurophysiol. 1992;9(4):456–79.
67. Farwell LA, Donchin E. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. Electroencephalogr Clin Neurophysiol. 1988;70(6):510–23.
68. Jin J, Horki P, Brunner C, Wang X, Neuper C, Pfurtscheller G. A new P300 stimulus presentation pattern for EEG-based spelling systems. Biomed Tech. 2010;55(4):203–10.
69. Pires G, Nunes U, Castelo-Branco M. Comparison of a row-column speller vs. a

novel lateral single-character speller: Assessment of BCI for severe motor disabled patients. Clin Neurophysiol. 2012;123(6):1168–81.

70. Akram F, Han SM, Kim TS. An efficient word typing P300-BCI system using a modified T9 interface and random forest classifier. Comput Biol Med. 2015;56:30–6.
71. Odom JV, Bach M, Barber C, Brigell M, Marmor MF, Tormene AP, Holder GE, Vaegan A. Visual evoked potentials standard (2004). Doc Ophthalmol. 2004;108(2):115–23.
72. Ignatious E, Azam S, Jonkman M, De Boer F. Frequency and Time Domain Analysis of EEG Based Auditory Evoked Potentials to Detect Binaural Hearing in Noise. J Clin Med. 2023;12(13).
73. Legatt AD. Evoked Potentials. Encycl Neurol Sci. 2014;228–31.
74. Volosyak I. SSVEP-based Bremen-BCI interface - Boosting information transfer rates. J Neural Eng. 2011;8(3).
75. Li M, He D, Li C, Qi S. Brain–computer interface speller based on steady-state visual evoked potential: A review focusing on the stimulus paradigm and performance. Brain Sci. 2021;11(4).
76. Ahn S, Kim K, Jun SC. Steady-state somatosensory evoked potential for brain-computer interface-present and future. Front Hum Neurosci. 2016;9(JAN2016).
77. Pfurtscheller G, Flotzinger D, Kalcher J. Brain-Computer Interface-a new communication device for handicapped persons. J Microcomput Appl. 1993;16(3):293–9.
78. Blankertz B, Dornhege G, Krauledat M, Müller KR, Curio G. The non-invasive Berlin Brain-Computer Interface: Fast acquisition of effective performance in untrained subjects. Neuroimage. 2007;37(2):539–50.
79. Alonso-Valerdi LM, Salido-Ruiz RA, Ramirez-Mendoza RA. Motor imagery based brain-computer interfaces: An emerging technology to rehabilitate motor deficits. Neuropsychologia. 2015;79:354–63.
80. Kardam VS, Taran S, Pandey A. Motor Imagery Tasks Based Electroencephalogram Signals Classification Using Data-Driven Features. Neurosci Informatics. 2023;3(2):100128.
81. Scherer R, Müller GR, Neuper C, Graimann B, Pfurtscheller G. An asynchronously

controlled EEG-based virtual keyboard: Improvement of the spelling rate. IEEE Trans Biomed Eng. 2004;51(6):979–84.

82. Giannopulu I, Mizutani H. Neural Kinesthetic Contribution to Motor Imagery of Body Parts: Tongue, Hands, and Feet. Front Hum Neurosci. 2021;15.
83. Wolpaw JR, McFarland DJ. Multichannel EEG-based brain-computer communication. Electroencephalogr Clin Neurophysiol. 1994;90(6):444–9.
84. Korkmaz OE, Aydemir O, Oral EA, Ozbek IY. An efficient 3D column-only P300 speller paradigm utilizing few numbers of electrodes and flashings for practical BCI implementation. PLoS One. 2022;17(4 April).
85. Acqualagna L, Treder MS, Blankertz B. Chroma Speller: Isotropic visual stimuli for truly gaze-independent spelling. Int IEEE/EMBS Conf Neural Eng NER. 2013;1041–4.
86. Diez PF, Mut VA, Avila Perona EM, Laciar Leber E. Asynchronous BCI control using high-frequency SSVEP. J Neuroeng Rehabil. 2011;8(1).
87. Müller-Putz GR, Eder E, Wriessnegger SC, Pfurtscheller G. Comparison of DFT and lock-in amplifier features and search for optimal electrode positions in SSVEP-based BCI. J Neurosci Methods. 2008;168(1):174–81.
88. Zhang Z, Li X, Deng Z. A CWT-based SSVEP classification method for brain-computer interface system. Proc 2010 Int Conf Intell Control Inf Process ICICIP 2010. 2010;(PART 2):43–8.
89. Zhao L, Yuan P, Xiao L, Meng Q, Hu D, Shen H. Research on SSVEP feature extraction based on HHT. Proc - 2010 7th Int Conf Fuzzy Syst Knowl Discov FSKD 2010. 2010;5:2220–3.
90. Wu CH, Chang HC, Lee PL, Li KS, Sie JJ, Sun CW, Yang CY, Li PH, Deng HT, Shyu KK. Frequency recognition in an SSVEP-based brain computer interface using empirical mode decomposition and refined generalized zero-crossing. J Neurosci Methods. 2011;196(1):170–81.
91. Parini S, Maggi L, Turconi AC, Andreoni G. A robust and self-paced BCI system based on a four class SSVEP paradigm: Algorithms and protocols for a high-transfer-rate direct brain communication. Comput Intell Neurosci. 2009;2009.
92. Friman O, Volosyak I, Gräser A. Multiple channel detection of steady-state visual evoked potentials for brain-computer interfaces. IEEE Trans Biomed Eng.

2007;54(4):742–50.

93. Lin Z, Zhang C, Wu W, Gao X. Frequency recognition based on canonical correlation analysis for SSVEP-based BCIs. IEEE Trans Biomed Eng. 2006;53(12):2610–4.
94. Zerafa R, Camilleri T, Falzon O, Camilleri KP. To train or not to train? A survey on training of feature extraction methods for SSVEP-based BCIs. J Neural Eng. 2018;15(5).
95. Neuper C, Müller-Putz GR, Scherer R, Pfurtscheller G. Chapter 25 Motor imagery and EEG-based control of spelling devices and neuroprostheses. Prog Brain Res. 2006;159:393–409.
96. Baxter BS, Decker A, He B. Noninvasive control of a robotic arm in multiple dimensions using scalp electroencephalogram. Int IEEE/EMBS Conf Neural Eng NER. 2013;45–7.
97. Sarac M, Koyas E, Erdogan A, Cetin M, Patoglu V. Brain Computer Interface based robotic rehabilitation with online modification of task speed. IEEE Int Conf Rehabil Robot. 2013;
98. Ang KK, Guan C. EEG-Based Strategies to Detect Motor Imagery for Control and Rehabilitation. IEEE Trans Neural Syst Rehabil Eng. 2017;25(4):392–401.
99. Murguialday AR, Aggarwal V, Chatterjee A, Cho Y, Rasmussen R, O'Rourke B, Acharya S, Thakor N V. Brain-computer interface for a prosthetic hand using local machine control and haptic feedback. 2007 IEEE 10th Int Conf Rehabil Robot ICORR'07. 2007;609–13.
100. Baig MZ, Aslam N, Shum HPH, Zhang L. Differential evolution algorithm as a tool for optimal feature subset selection in motor imagery EEG. Expert Syst Appl. 2017;90:184–95.
101. Ilyas MZ, Saad P, Ahmad MI, Ghani ARI. Classification of EEG signals for brain-computer interface applications: Performance comparison. Proc 2016 Int Conf Robot Autom Sci ICORAS 2016. 2017;
102. Craik A, He Y, Contreras-Vidal JL. Deep learning for electroencephalogram (EEG) classification tasks: A review. J Neural Eng [Internet]. 2019;16(3):31001. Available from: http://dx.doi.org/10.1088/1741-2552/ab0ab5
103. Li M ai, Han J fu, Yang J fu. Automatic feature extraction and fusion recognition

of motor imagery EEG using multilevel multiscale CNN. Med Biol Eng Comput. 2021;59(10):2037–50.

104. Khademi Z, Ebrahimi F, Kordy HM. A review of critical challenges in MI-BCI: From conventional to deep learning methods. J Neurosci Methods. 2023;383.
105. Wang X, Liesaputra V, Liu Z, Wang Y, Huang Z. An in-depth survey on Deep Learning-based Motor Imagery Electroencephalogram (EEG) classification. Artif Intell Med. 2024;147.
106. Sakmann B, Neher E. Patch clamp techniques for studying ionic channels in excitable membranes. Annu Rev Physiol. 1984;VOL. 46:455–72.
107. Schmidt EM, Bak MJ, McIntosh JS. Long-term chronic recording from cortical neurons. Exp Neurol. 1976;52(3):496–506.
108. Williams JC, Rennaker RL, Kipke DR. Long-term neural recording characteristics of wire microelectrode arrays implanted in cerebral cortex. Brain Res Protoc. 1999;4(3):303–13.
109. Steinmetz NA, Aydin C, Lebedeva A, Okun M, Pachitariu M, Bauza M, Beau M, Bhagat J, Böhm C, Broux M, Chen S, Colonell J, Gardner RJ, Karsh B, Kloosterman F, Kostadinov D, Mora-Lopez C, O'Callaghan J, Park J, Putzeys J, Sauerbrei B, van Daal RJJ, Vollan AZ, Wang S, Welkenhuysen M, Ye Z, Dudman JT, Dutta B, Hantman AW, Harris KD, Lee AK, Moser EI, O'Keefe J, Renart A, Svoboda K, Häusser M, Haesler S, Carandini M, Harris TD. Neuropixels 2.0: A miniaturized high-density probe for stable, long-term brain recordings. Science (80-). 2021;372(6539).
110. Kadir SN, Goodman DFM, Harris KD. High-dimensional cluster analysis with the masked EM algorithm. Neural Comput. 2014;26(11):2379–94.
111. Yang K, Wu H, Zeng Y. Sparse coding approaches for neuron spike sorting. Proc 2016 IEEE Adv Inf Manag Commun Electron Autom Control Conf IMCEC 2016. 2017;600–4.
112. Fiáth R, Raducanu BC, Musa S, Andrei A, Lopez CM, van Hoof C, Ruther P, Aarts A, Horváth D, Ulbert I. A silicon-based neural probe with densely-packed low-impedance titanium nitride microelectrodes for ultrahigh-resolution in vivo recordings. Biosens Bioelectron. 2018;106:86–92.
113. Pachitariu M, Steinmetz N, Kadir S, Carandini M, Harris K. Fast and accurate

spike sorting of high-channel count probes with KiloSort. In: Advances in Neural Information Processing Systems [Internet]. 2016. p. 4455–63. Available from: http://papers.nips.cc/paper/6325-fast-and-accurate-spike-sorting-of-high-channel-count-probes-with-kilosort

114. The Cortical Processing Laboratory at UCL. Phy [Internet]. Available from: https://phy.readthedocs.io/en/latest/
115. The Cortical Processing Laboratory at UCL. sortingQuality [Internet]. Available from: https://github.com/cortex-lab/sortingQuality
116. Schmitzer-Torbert N, Jackson J, Henze D, Harris K, Redish AD. Quantitative measures of cluster quality for use in extracellular recordings. Neuroscience. 2005;131(1):1–11.
117. Hill DN, Mehta SB, Kleinfeld D. Quality metrics to accompany spike sorting of extracellular signals. J Neurosci. 2011;31(24):8699–705.
118. Buzsáki G, Anastassiou CA, Koch C. The origin of extracellular fields and currents-EEG, ECoG, LFP and spikes. Nat Rev Neurosci. 2012;13(6):407–20.
119. Urigüen JA, Garcia-Zapirain B. EEG artifact removal - State-of-the-art and guidelines. J Neural Eng. 2015;12(3).
120. Islam MK, Rastegarnia A, Yang Z. Les méthodes de détection et de rejet d'artefact de l'EEG de scalp : revue de littérature. Neurophysiol Clin. 2016;46(4–5):287–305.
121. Kanoga S, Mitsukura Y. Review of Artifact Rejection Methods for Electroencephalographic Systems. In: Electroencephalography. 2017.
122. DePaul R, Abbs JH, Caligiuri M, Gracco VL, Brooks BR. Hypoglossal, trigeminal, and facial motoneuron involvement in amyotrophic lateral sclerosis. Neurology. 1988;38(2):281–3.
123. Lui AJ, Byl NN. A systematic review of the effect of moderate intensity exercise on function and disease progression in amyotrophic lateral sclerosis. J Neurol Phys Ther. 2009;33(2):68–87.
124. Spüler M. Questioning the evidence for BCI-based communication in the complete locked-in state. PLoS Biol. 2019;17(4).
125. Chaudhary U, Xia B, Silvoni S, Cohen LG, Birbaumer N. Brain–Computer Interface–Based Communication in the Completely Locked-In State. PLoS Biol.

2017;15(1).

126. Rezvani S, Hosseini-Zahraei SH, Tootchi A, Guger C, Chaibakhsh Y, Saberi A, Chaibakhsh A. A review on the performance of brain-computer interface systems used for patients with locked-in and completely locked-in syndrome. Cogn Neurodyn. 2023;
127. Allison BZ, Neuper C. Could Anyone Use a BCI? In 2010. p. 35–54.
128. Thompson MC. Critiquing the Concept of BCI Illiteracy. Sci Eng Ethics. 2019;25(4):1217–33.
129. Becker S, Dhindsa K, Mousapour L, Al Dabagh Y. BCI Illiteracy: It's Us, Not Them. Optimizing BCIs for Individual Brains. Int Winter Conf Brain-Computer Interface, BCI. 2022;2022-Febru.
130. Rácz M, Noboa E, Détár B, Nemes Á, Galambos P, Szűcs L, Márton G, Eigner G, Haidegger T. PlatypOUs—A Mobile Robot Platform and Demonstration Tool Supporting STEM Education. Sensors. 2022;22(6).
131. Noboa E, Rácz M, Szucs L, Galambos P, Márton G, Eigner G. Development of an EMG based SVM supported control solution for the PlatypOUs education mobile robot using MindRove headset. IFAC-PapersOnLine. 2021;54(15):304–9.
132. MindRove Kft. arc - EEG based brain-computer interface [Internet]. Available from: https://mindrove.com/arc/
133. MindRove Kft. MindRove SDK [Internet]. Available from: https://github.com/MindRove/MindRoveSDK
134. Emgu Corporation. Emgu CV Documentation [Internet]. Available from: https://www.emgu.com/wiki/index.php/Documentation
135. Gary Bradski. The OpenCV Library. Dr Dobb's J Softw Tools. 2000;
136. Rosenblatt F. The perceptron: A probabilistic model for information storage and organization in the brain. Psychol Rev. 1958;65(6):386–408.
137. Greener JG, Kandathil SM, Moffat L, Jones DT. A guide to machine learning for biologists. Nat Rev Mol Cell Biol. 2022;23(1):40–55.
138. Ian Goodfellow, Yoshua Bengio AC. Deep Learning Book [Internet]. Vol. 21, Deep Learning. 2015. 111–124 p. Available from: http://www.deeplearningbook.org/
139. Lecun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436–44.

140. Maier A, Syben C, Lasser T, Riess C. A gentle introduction to deep learning in medical image processing. Z Med Phys. 2019;29(2):86–101.
141. Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. Neural Networks. 1989;2(5):359–66.
142. Cunningham P, Cord M, Delany SJ. Supervised learning. In: Cognitive Technologies. 2008. p. 21–49.
143. Werbos PJ. Applications of advances in nonlinear sensitivity analysis. Syst Model Optim. 2005;762–70.
144. Shrestha A, Mahmood A. Review of deep learning algorithms and architectures. IEEE Access. 2019;7:53040–65.
145. Duchi J, Hazan E, Singer Y. Adaptive subgradient methods for online learning and stochastic optimization. J Mach Learn Res. 2011;12:2121–59.
146. Zeiler MD. ADADELTA: An Adaptive Learning Rate Method. 2012; Available from: http://arxiv.org/abs/1212.5701
147. Reddy SVG, Thammi Reddy K, Vallikumari V. Optimization of deep learning using various optimizers, loss functions and dropout. Int J Recent Technol Eng. 2018;7(4):448–55.
148. Xie D, Zhang L, Bai L. Deep Learning in Visual Computing and Signal Processing. Appl Comput Intell Soft Comput. 2017;2017.
149. Vodrahalli K, Bhowmik AK. 3D computer vision based on machine learning with deep neural networks: A review. J Soc Inf Disp. 2017;25(11):676–94.
150. Lee JG, Jun S, Cho YW, Lee H, Kim GB, Seo JB, Kim N. Deep learning in medical imaging: General overview. Korean J Radiol. 2017;18(4):570–84.
151. Dolz J, Desrosiers C, Ben Ayed I. 3D fully convolutional networks for subcortical segmentation in MRI: A large-scale study. Neuroimage. 2018;170:456–70.
152. Alizadeh Savareh B, Emami H, Hajiabadi M, Ghafoori M, Majid Azimi S. Emergence of Convolutional Neural Network in Future Medicine: Why and How. A Review on Brain Tumor Segmentation. Polish J Med Phys Eng. 2018;24(1):43–53.
153. Carneiro G, Zheng Y, Xing F, Yang L. Review of deep learning methods in mammography, cardiovascular, and microscopy image analysis. In: Advances in Computer Vision and Pattern Recognition. 2017. p. 11–32.

154. Summers RM. Deep learning and computer-aided diagnosis for medical image processing: A personal perspective. In: Advances in Computer Vision and Pattern Recognition. 2017. p. 3–10.
155. Rossum G Van, Drake FL. Python Reference Manual. October [Internet]. 2006;22:9117–29. Available from: http://www.python.org/doc/ref/
156. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, Ghemawat S, Goodfellow I, Harp A, Irving G, Isard M, Jia Y, Jozefowicz R, Kaiser L, Kudlur M, Levenberg J, Mane D, Monga R, Moore S, Murray D, Olah C, Schuster M, Shlens J, Steiner B, Sutskever I, Talwar K, Tucker P, Vanhoucke V, Vasudevan V, Viegas F, Vinyals O, Warden P, Wattenberg M, Wicke M, Yu Y, Zheng X. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. 2016; Available from: http://arxiv.org/abs/1603.04467
157. Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, Kern R, Picus M, Hoyer S, van Kerkwijk MH, Brett M, Haldane A, del Río JF, Wiebe M, Peterson P, Gérard-Marchant P, Sheppard K, Reddy T, Weckesser W, Abbasi H, Gohlke C, Oliphant TE. Array programming with NumPy. Nature. 2020;585(7825):357–62.
158. Pedregosa, F; Varoquaux, G; Gramfort, A; Michel, V; Thirion, B; and Grisel, O. and Blondel, . and Prettenhofer P, and Weiss, R. and Dubourg, V. and Vanderplas, J. and Passos A and, Cournapeau, D. and Brucher, M. and Perrot, M. and Duchesnay E, Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay É. Scikit-learn: Machine Learning in Python. J Mach Learn Res [Internet]. 2011;12:2825–30. Available from: http://dl.acm.org/citation.cfm?id=2078195%5Cnhttp://arxiv.org/abs/1201.0490
159. Boser BE, Guyon IM, Vapnik VN. Training algorithm for optimal margin classifiers. Proc Fifth Annu ACM Work Comput Learn Theory. 1992;144–52.
160. Cortes C, Vapnik V. Support-vector networks. Mach Learn. 1995;20(3):273–97.
161. Hsu CW, Lin CJ. A comparison of methods for multiclass support vector machines. IEEE Trans Neural Networks. 2002;13(2):415–25.

162. Althnian A, AlSaeed D, Al-Baity H, Samha A, Dris A Bin, Alzakari N, Abou Elwafa A, Kurdi H. Impact of dataset size on classification performance: An empirical evaluation in the medical domain. Appl Sci. 2021;11(2):1–18.
163. Wikimedia Commons. File:SVM margin.png [Internet]. Available from: https://commons.wikimedia.org/wiki/File:SVM_margin.png
164. Wikimedia Commons. File:Kernel Machine.png [Internet]. Available from: https://commons.wikimedia.org/wiki/File:Kernel_Machine.png
165. OpenCV. SVM Class Reference [Internet]. Available from: https://docs.opencv.org/4.x/d1/d2d/classcv_1_1ml_1_1SVM.html#ab4b93a4c42bbe213ffd9fb3832c6c44fa18157ccaec6a252b901cff6de285d608
166. Microsoft Corporation. CORREL function [Internet]. Available from: https://support.microsoft.com/en-us/office/correl-function-995dcef7-0c0a-4bed-a3fb-239d7b68ca92
167. Akoglu H. User's guide to correlation coefficients. Turkish J Emerg Med. 2018;18(3):91–3.
168. Lee JH, Carlson D, Shokri H, Yao W, Goetz G, Hagen E, Batty E, Chichilnisky EJ, Einevoll G, Paninski L. YASS: Yet another spike sorter. Adv Neural Inf Process Syst. 2017;2017-Decem:4003–13.
169. Saif-Ur-Rehman M, Lienk mper R, Parpaley Y, Wellmer J, Liu C, Lee B, Kellis S, Andersen R, Iossifidis I, Glasmachers T, Klaes C. SpikeDeeptector: A deep-learning based method for detection of neural spiking activity. J Neural Eng. 2019;16(5).
170. Yang K, Wu H, Zeng Y. A Simple Deep Learning Method for Neuronal Spike Sorting. J Phys Conf Ser. 2017;910(1).
171. Wu T, Ratkai A, Schlett K, Grand L, Yang Z. Learning to Sort: Few-shot Spike Sorting with Adversarial Representation Learning. Proc Annu Int Conf IEEE Eng Med Biol Soc EMBS. 2019;713–6.
172. Li Z, Wang Y, Zhang N, Li X. An accurate and robust method for spike sorting based on convolutional neural networks. Brain Sci. 2020;10(11):1–16.
173. Yi J, Xu J, Chen E, Chamanzar M, Chen V. Multichannel Many-Class Real-Time Neural Spike Sorting With Convolutional Neural Networks. IEEE Open J Circuits Syst. 2022;3:168–79.

174. Liu M, Feng J, Wang Y, Li Z. Classification of overlapping spikes using convolutional neural networks and long short term memory. Comput Biol Med. 2022;148.
175. Saif-ur-Rehman M, Ali O, Klaes C, Iossifidis I. Adaptive SpikeDeep-Classifier: Self-organizing and self-supervised machine learning algorithm for online spike sorting. 2023; Available from: http://arxiv.org/abs/2304.01355
176. Zhang L, Gao D, Wang M. Sorting Overlapping Spikes Based on Log-Mel Spectrogram and Convolutional Neural Networks. 2023 6th Int Conf Artif Intell Big Data, ICAIBD 2023. 2023;482–5.
177. Wang M, Zhang L, Yu H, Chen S, Zhang X, Zhang Y, Gao D. A deep learning network based on CNN and sliding window LSTM for spike sorting. Comput Biol Med. 2023;159.
178. Li S, Tang Z, Yang L, Li M, Shang Z. Application of deep reinforcement learning for spike sorting under multi-class imbalance. Comput Biol Med. 2023;164.
179. Meyer LM, Samann F, Schanze T. DualSort: online spike sorting with a running neural network. J Neural Eng. 2023;20(5).
180. Meyer L, Zamani M, Rokai J, Demosthenous A. Deep learning-based spike sorting: a survey. J Neural Eng [Internet]. 2024; Available from: https://iopscience.iop.org/article/10.1088/1741-2552/ad8b6c
181. ARP Adatrepozitórium. Dataset for the study “Slow insertion of silicon probes improves the quality of acute neuronal recordings” [Internet]. Available from: https://repo.researchdata.hu/dataset.xhtml?persistentId=hdl:21.15109/ARP/GBWSNE
182. Mounier E, Abdullah B, Mahdi H, Eldawlatly S. A deep convolutional visual encoding model of neuronal responses in the LGN. Brain Informatics. 2021;8(1).
183. MindRove Kft. Event-related potential measurements. Technical white paper [Internet]. Available from: https://mindrove.com/arc/
184. Boubchir L, Touati Y, Daachi B, Cherif AA. EEG error potentials detection and classification using time-frequency features for robot reinforcement learning. Proc Annu Int Conf IEEE Eng Med Biol Soc EMBS. 2015;2015-Novem:1761–4.
185. Kuffuor J, Samanta B. Brain Computer Interface Using Motor Imagery and Facial Expressions to Control A Mobile Robot. Conf Proc - IEEE SOUTHEASTCON.

2018;2018-April.

186. Yang J, Su X, Bai D, Jiang Y, Yokoi H. Hybrid EEG-EOG system for intelligent prosthesis control based on common spatial pattern algorithm. 2016 IEEE Int Conf Inf Autom IEEE ICIA 2016. 2017;1261–6.
187. Shantala CP, Rashmi CR. Mind Controlled Wireless Robotic Arm Using Brain Computer Interface. 2017 IEEE Int Conf Comput Intell Comput Res ICCIC 2017. 2018;
188. Lu Z, Zhang X, Li H, Li R, Chen J. A Real-Time Brain Control Method Based on Facial Expression for Prosthesis Operation. 2018 IEEE Int Conf Robot Biomimetics, ROBIO 2018. 2018;668–73.
189. Kutatók Éjszakája 2023. Hallgasd meg az agyhullámaid! - Zajkeltés bioelektromos jelekkel [Internet]. Available from: https://app.kutatokejszakaja.hu/esemenyek/hun-ren-termeszettudomanyi-kutatokozpont/hallgasd-meg-az-agyhullamaid-zajkeltes-bioelektromos-jelekkel
190. Agykutatás Hete 2024. Robotok vezérlése gesztusokkal [Internet]. Available from: https://agykutatashete.hu/helyszinek/budapest/

9. Bibliography of the candidate's publications

9.1. Works related to this thesis

1. Rácz M, Liber C, Németh E, Fiáth R, Rokai J, Harmati I, Ulbert I, Márton G. Spike detection and sorting with deep learning. J Neural Eng. 2020;17(1).
2. Rokai J, Rácz M, Fiáth RB, Ulbert I, Márton G. ELVISort: encoding latent variables for instant sorting, an artificial intelligence-based end-to-end solution. J Neural Eng. 2021;
3. Noboa E, Rácz M, Szucs L, Galambos P, Márton G, Eigner G. Development of an EMG based SVM supported control solution for the PlatypOUs education mobile robot using MindRove headset. IFAC-PapersOnLine. 2021;54(15):304–9.
4. Rácz M, Noboa E, Détár B, Nemes Á, Galambos P, Szűcs L, Márton G, Eigner G, Haidegger T. PlatypOUs—A Mobile Robot Platform and Demonstration Tool Supporting STEM Education. Sensors. 2022;22(6).

9.2. Other works

1. Matusz VI, Racz M, Rokai J, Jorgosz NE, Molnar T, Maraki D, Ulbert I, Márton G. Head-mounted, wireless eyetracker for real-Time gaze prediction utilizing machine-learning. In: Proceedings of the 2020 IEEE International Conference on Human-Machine Systems, ICHMS 2020. 2020.
2. Horváth D, Négyesi J, Rácz M, Győri T, Matics Z, Puskin A, Csipor J, Rácz L. Feasibility of a novel neurofeedback system: a parallel randomized single-blinded pilot study. Sci Rep. 2023;13(1):32.

DOI:10.14753/SE.2025.3078

10. Acknowledgements

First of all, I would like to say thanks to my supervisor, Dr. Gergely Márton for making my research possible by inviting me to participate in the fascinating projects I got involved during the years of the PhD program and for considering my progress a top priority, to MindRove for providing the devices and additional resources, and also to the entirety of the Integrative Neuroscience Research Group led by Dr. István Ulbert at the Institute of Cognitive Neuroscience and Psychology at the Research Centre for Natural Sciences, for all the support and the opportunities to learn and practice skills that would have been otherwise unattainable to me, such as surgery and electrode implantation I was introduced to by Dr. Richárd Fiáth and Csaba Horváth. I will miss brain slicing, too, just as much as having our coffee together and the gatherings at Szigliget.

I also owe much to my numerous co-authors for their help (as well as the real ScienceDirect); most importantly to Erik Németh for his generosity to grant me the entirety of the spike prediction part of the work on spike sorting we implemented together.

I would also like to express my gratitude to the Selye János Doctoral College for Advanced Studies for the supportive atmosphere and the eventual home they provided throughout the years; my teachers at the Department of Control Engineering and Informatics at the Budapest University of Technology and Economics (most importantly to Dr. István Harmati, Dr. György Pilászy, Dr. György Rácz and Dr. Márton Szemenyei) and also at my former elementary and high school, the Waldorf School at Gödöllő (above all, to Emőke Buga for the love of English language and not less for the B2 level certificate I could exploit until the very completion of my doctoral studies, as well as Attila Kálózi, Miklós Scharnitzky and Dr. Szilárd Bodó for helping me to get accustomed to some aspects of science).

Despite of the satisfactory progress of my studies and research, there were tougher times I am not sure I could have handle without the help of Dr. Lucia Wittner, Dr. Réka Bod, Péter Szűcs and Fruzsina Horváth who always stood by me and to whom I could vent when needed.

Last but not least, I will never be able to requite all the care, sacrifice and love my parents have given to me. For my father, this opportunity does not exist in theory, either.

I am grateful to you all, this work is just as yours as mine.