EEGNet: A Compact Convolutional Neural Network for EEG-based Brain-Computer Interfaces

Vernon J. Lawhern[1,*], Amelia J. Solon[1,2], Nicholas R. Waytowich[1,3], Stephen M. Gordon[1,2], Chou P. Hung[1,4], and Brent J. Lance[1]

[1]Human Research and Engineering Directorate, U.S. Army Research Laboratory, Aberdeen Proving Ground, MD
[2]DCS Corporation, Alexandria, VA
[3]Department of Biomedical Engineering, Columbia University, New York, NY
[4]Department of Neuroscience, Georgetown University, Washington, DC
[*]Corresponding Author

April 30, 2018

Abstract

Objective: Brain computer interfaces (BCI) enable direct communication with a computer, using neural activity as the control signal. This signal is generally chosen from a variety of well-studied electroencephalogram (EEG) signals. For a given BCI paradigm, feature extractors and classifiers are tailored to the distinct characteristics of its expected EEG control signal, limiting its application to that specific signal. Convolutional Neural Networks (CNNs), which have been used in computer vision and speech recognition to perform automatic feature extraction and classification, have successfully been applied to EEG-based BCIs; however, they have mainly been applied to single BCI paradigms and thus it remains unclear how these architectures generalize to other paradigms. Here, we ask if we can design a single CNN architecture to accurately classify EEG signals from different BCI paradigms, while simultaneously being as compact as possible (defined as the number of parameters in the model). *Approach*: In this work we introduce EEGNet, a compact convolutional neural network for EEG-based BCIs. We introduce the use of depthwise and separable convolutions to more efficiently extract relevant features for EEG-based BCIs. We compare EEGNet, both for within-subject and cross-subject classification, to current state-of-the-art approaches across four BCI paradigms: P300 visual-evoked potentials, error-related negativity responses (ERN), movement-related cortical potentials (MRCP), and sensory motor rhythms (SMR). *Results*: We show that EEGNet generalizes across paradigms better than, and achieves comparably high performance to, the reference algorithms, while simultaneously fitting up to two orders of magnitude fewer parameters. We also demonstrate three different approaches to visualize the contents of a trained EEGNet model to enable interpretation of the learned features. *Significance*: Our results suggest that EEGNet is robust enough to learn a wide variety of interpretable features over a range of BCI tasks, suggesting that the observed performances were not due to artifact or noise sources in the data.

Keywords: Brain-Computer Interface, EEG, Deep Learning, Convolutional Neural Network, P300, Error-Related Negativity, Sensory Motor Rhythm

1 Introduction

A Brain-Computer Interface (BCI) is a mechanism for communicating with a machine via brain signals, bypassing normal neuromuscular outputs by using neural activity [1]. Traditionally, BCIs, which have leveraged both invasive and noninvasive recording methods, have been used for medical applications such as neural control of prosthetic artificial limbs [2]. However, recent research has opened up the possibility for novel BCIs focused on enhancing performance of healthy users, often with noninvasive approaches based on electroencephalography (EEG). Generally speaking, a BCI consists of five main processing stages [3]: a data collection stage, where neural data is recorded; a signal processing stage, where the recorded data is preprocessed and cleaned; a feature extraction stage, where meaningful information is extracted from the neural data; a classification stage, where a decision is interpreted from the data; and a feedback stage, where the result of that decision is provided to the user. While these stages are largely the same across BCI paradigms, each paradigm relies on manually specified methods for signal processing [4], feature extraction [5] and classification [6], a process which often requires significant subject-matter expertise and/or *a priori* knowledge about the expected EEG signal. It is also possible that, because the EEG signal preprocessing steps are often very specific to the EEG feature of interest (for example, band-pass filtering to a specific frequency range), that other potentially relevant EEG features are being excluded from analysis (for example, features outside of the band-pass frequency range). The need for robust feature extraction techniques will only continue to increase as BCI technologies evolve into new application domains [7–12].

Deep Learning has largely alleviated the need for manual feature extraction, achieving state-of-the-art performance in fields such as computer vision and speech recognition [13,14]. The use of deep convolutional neural networks (CNNs) in particular has increased significantly, due in part to their success in many challenging image classification problems [15–19], surpassing methods that have relied on the use of hand-crafted features (see [14] and [20] for recent reviews). The majority of BCI systems, however, still rely on the use of handcrafted features. This raises the following question: Can Deep Learning approaches be used to design more robust features suitable for classification of EEG signals? There have been many previous works on applying Deep Learning approaches to EEG signals. For example, CNNs have been used for epilepsy prediction and monitoring [21–25], for auditory music retrieval [26, 27], for detection of visual-evoked responses [28–31] and for motor imagery classification [32], while Deep Belief Networks (DBNs) have been used for sleep stage detection [33], anomaly detection [34] and in motion-onset visual-evoked potential classification [35]. Convolutional networks on time-frequency transforms of the EEG were used for mental workload classification [36] and for motor imagery classification [37–39]). Restricted Boltzman Machines (RBMs) have been used for motor imagery [40]. An adaptive method based on stacked denoising autoencoders has been proposed for mental workload classification [41]). These studies focused primarily on classification in a single BCI task, often times using task-specific knowledge in designing the network architecture. In addition, the amount of data used to train these networks varied significantly across studies, in part due to the difficulty in collecting data under different experimental paradigms. Thus, it remains unclear how these previous approaches would generalize both to other BCI tasks as well as to variable training data sizes.

In this work we introduce *EEGNet*, a compact convolutional neural network for classification and interpretation of EEG-based BCIs. We introduce the use of *Depthwise* and *Separable* convolutions, previously used in computer vision [42], to EEG signal classification and show that these they can be used to significantly reduce model size while simultaneously improving model performance. We test the generalizability of EEGNet by systematically evaluating our method against EEG datasets collected from four different BCI paradigms: P300 visual-evoked potential (P300), error-related negativity (ERN), movement-related cortical potential (MRCP) and the sensory motor rhythm (SMR), representing a spectrum of paradigms based on classification of Event-Related Potentials (P300, ERN, MRCP) as well as classification of oscillatory components (SMR). In addition, each of these data collections contained varying amounts of data, allowing us to determine the efficacy of EEGNet across a variety of training data sizes. Our results are three-fold: We show that EEGNet achieves improved classification performance over an existing paradigm-agnostic EEG CNN model across nearly all tested paradigms when limited training data is available. We also show that EEGNet performs just as well as a more paradigm-specific EEG CNN model, but with two orders of magnitude fewer parameters to fit, representing a more efficient use of model parameters (an aspect that has been explored in previous deep learning literature, see [42,43]). Finally, through the use of feature visualization and model ablation analysis, we show that neurophysiologically interpretable features can be extracted from the EEGNet model. This is particularly important as CNNs, despite their ability for robust and automatic feature extraction, often produce hard to interpret features. We validate our architecture's ability to extract neurophysiologically interpretable signals on several well-studied BCI paradigms, providing further validation and evidence that the network performance is not being driven by noise or artifact signals in the data.

The remainder of this manuscript is structured as follows. Section 2.1 gives a brief description of the four datasets used in our CNN model validation procedure. Section 2.2 describe our EEGNet model as well as other BCI models (both CNN and non-CNN based models) used in our model comparison. Section 3 gives the results of both within-subject and cross-subject classification performance, as well as results of our feature ablation study in order to interpret feature significance on overall performance. We discuss our findings in more detail in the Discussion.

2 Materials and Methods

2.1 Data Description

BCIs are generally categorized into two types, depending on the EEG feature of interest [44]: event-related and oscillatory. *Event-Related Potential* (ERP) BCIs are designed to detect an EEG response to a known, time-locked external stimulus. They are generally robust across subjects and contain well-stereotyped waveforms, enabling the exact time course of the ERP to be modeled through machine learning efficiently [45]. In contrast to ERP-based BCIs, which rely mainly on the detection of the ERP waveform from some external event or stimulus, *Oscillatory* BCIs use the signal power of specific EEG frequency bands for external control and are generally not time-locked to an external stimulus [46]. When oscillatory signals are time-locked to an external stimulus, they can be represented through event-related spectral perturbation (ERSP) analyses [47]. Oscillatory BCIs are more difficult to train, generally due to the lower SNR as well as greater variation across

Paradigm	Feature Type	Bandpass Filter	# of Subjects	Trials per Subject	# of Classes	Class Imbalance?
P300	ERP	1-40Hz	15	$\sim$ 2000	2	Yes, 4:1
ERN	ERP	1-40Hz	26	340	2	Yes, $\sim$ 3.4:1
MRCP	ERP/Oscillatory	0.1-40Hz	13	$\sim$ 1100	2	No
SMR	Oscillatory	4-40Hz	9	288	4	No

Table 1: Summary of the data collections used in this study. Class imbalance, if present, is given as odds; i.e.: an odds of 2:1 means the class imbalance is 2/3 of the data for class 1 to 1/3 of the data for class 2. For the ERN dataset, the class imbalance is subject-dependent; therefore, the odds is given as the average class imbalance over all subjects.

subjects [46]. A summary of the data used in this manuscript can be found in Table 1

2.1.1 Dataset 1: P300 Event-Related Potential (P300)

The P300 event-related potential is a stereotyped neural response to novel visual stimuli [48]. It is most commonly elicited with the visual oddball paradigm, where participants are shown repetitive "non-target" visual stimuli that are interspersed with infrequent "target" stimuli at a fixed presentation rate (for example, 1 Hz). Observed over the parietal cortex, the P300 waveform is a large positive deflection of electrical activity observed approximately 300 ms post stimulus onset, the strength of the observed deflection being inversely proportional to the frequency of the target stimuli. The P300 ERP is one of the strongest neural signatures observable by EEG, especially when targets are presented infrequently [48]. When the image presentation rate increases to 2 Hz or more, it is commonly referred to as rapid serial visual presentation (RSVP), which has been used to develop BCIs for large image database triage [49–51].

The EEG data used here have been previously described in [50]; a brief description is given below. 18 participants volunteered for an RSVP BCI study. Participants were shown images of natural scenery at 2 Hz rate, with images either containing a vehicle or person (target), or with no vehicle or person present (non-target). Participants were instructed to press a button with their dominant hand when a target image was shown. The target/non-target ratio was 20%/80%. Data from 3 participants were excluded from the analysis due to excessive artifacts and/or noise within the EEG data. Data from the remaining 15 participants (9 male and 14 right-handed) who ranged in age from 18 to 57 years (mean age 39.5 years) were further analyzed. EEG recordings were digitally sampled at 512 Hz from 64 scalp electrodes arranged in a 10-10 montage using a BioSemi Active Two system (Amsterdam, The Netherlands). Continuous EEG data were referenced offline to the average of the left and right earlobes, digitally bandpass filtered, using an FIR filter implemented in EEGLAB [52], to 1-40 Hz and downsampled to 128 Hz. EEG trials of target and non-target conditions were extracted at $[0, 1]s$ post stimulus onset, and used for a two-class classification.

2.1.2 Dataset 2: Feedback Error-Related Negativity (ERN)

Error-Related Negativity potentials are perturbations of the EEG following an erroneous or unusual event in the subject's environment or task. They can be observed in a variety of tasks, including time interval production paradigms [53] and in forced-choice paradigms [54, 55]. Here we focus on the feedback error-related negativity (ERN), which is an amplitude perturbation of the EEG following the perception of an erroneous feedback produced by a BCI. The feedback ERN is characterized as a large negative deflection approximately 300ms after feedback, followed by a positive component 500ms to 1s after feedback (see Figure 7 of [56] for an illustration). The detection of the feedback ERN provides a mechanism to infer, and to possibly correct in real-time, the incorrect output of a BCI. This two-stage system has been proposed as a hybrid BCI in [57, 58] and has been shown to improve the performance of a P300 speller in online applications [59].

The EEG data used here comes from [56] and was used in the "BCI Challenge" hosted by Kaggle (https://www.kaggle.com/c/inria-bci-challenge); a brief description is given below. 26 healthy participants (16 for training, 10 for testing) participated in a P300 speller task, a system which uses a random sequence of flashing letters, arranged in a 6×6 grid, to elicit the P300 response [60]. The goal of the challenge was to determine whether the feedback of the P300 speller was correct or incorrect. The EEG data were originally recorded at 600Hz using 56 passive Ag/AgCl EEG sensors (VSM-CTF compatible system) following the extended 10-20 system for electrode placement. Prior to our analysis, the EEG data were subsequently band-pass filtered, using an FIR filter implemented in EEGLAB [52], to 1-40 Hz and down-sampled to 128Hz. EEG trials of correct and incorrect feedback were extracted at $[0, 1.25]s$ post feedback presentation and used as features for a two-class classification.

2.1.3 Dataset 3: Movement-Related Cortical Potential (MRCP)

Some neural activities contain both an ERP component as well as an oscillatory component. One particular example of this is the movement-related cortical potential (MRCP), which can be elicited by voluntary movements of the hands and feet and is observable through EEG along the central and midline electrodes, contralateral to the hand or foot movement [61]. The MRCP can be seen both before movement onset (an early desynchronization in the 10-12Hz frequency band) as well as after movement onset (a late synchronization of 20-30Hz activity approximately 1s after movement execution). The MRCP has been used previously to develop motor control BCIs for both healthy and physically disabled patients [62, 63]

The EEG data used here have been previously described in [64]; a brief description is given below. In this study, 13 subjects performed self-paced finger movements using the left index, left middle, right index, or right middle fingers. This produced the well-known alpha and beta synchronizations (i.e. increases in power) and desynchronizations (i.e. decreases in power), most clearly observed over the contralateral motor cortex [65–67]. The data was originally recorded using a 256 channel BioSemi Active II system at 1024 Hz. Due to extensive signal noise present in the data, the EEG data were first processed with the PREP pipeline [68]. The data were referenced to linked mastoids, bandpass filtered, using an FIR filter implemented in EEGLAB [52], between 0.1 Hz and 40 Hz, and then downsampled to 128 Hz. We further downsampled the channel space to the standard 64 channel BioSemi montage. The index and middle finger blocks for each hand were combined for binary classification of movements originating from the left or right hand. EEG trials of left and right hand finger movements were extracted at $[-.5, 1]s$ around finger movement onset and used for a two-class classification.

2.1.4 Dataset 4: Sensory Motor Rhythm (SMR)

A common control signal for oscillatory-based BCI is the sensorimotor rhythm (SMR), wherein mu (8-12Hz) and beta (18-26Hz) bands desynchronize over the sensorimotor cortex contralateral to an actual or imagined movement. The SMR is very similar to the oscillatory component of the MRCP. While SMR-based BCIs can facilitate nuanced, endogenous BCI control, they are not without their practical challenges. As signals, SMRs tend to be weak and highly variable across and within subjects, conventionally demanding user-training (neurofeedback) and long calibration times (20 minutes) in order to achieve reasonable performance [44].

The EEG data used here comes from BCI Competition IV Dataset 2A [69] (called the SMR dataset for the remainder of the manuscript). The data consists of four classes of imagined movements of left and right hands, feet and tongue recorded from 9 subjects. The EEG data were originally recorded using 22 Ag/AgCl electrodes, sampled at 250 Hz and bandpass filtered between 0.5 and 100Hz. We follow the same EEG pre-processing procedure as described in [32], using software that was provided by the authors. For both the training and test sets we epoched the data at [0.5, 2.5] seconds post cue onset (the same window which was used in [39, 44]). Note that we make predictions for only this time range on the test set. We perform a four-class classification using accuracy as the summary measure.

2.2 Classification Methods

2.2.1 EEGNet: Compact CNN Architecture

Here we introduce EEGNet, a compact CNN architecture for EEG-based BCIs that (1) can be applied across several different BCI paradigms, (2) can be trained with very limited data and (3) can produce neurophysiologically interpretable features. A visualization and full description of the EEGNet model can be found in Figure 1 and Table 2, respectively, for EEG trials, collected at 128Hz sampling rate, having C channels, T time samples and F filters. We fit the model using Adam, using default parameter as described in [70], minimizing the categorical cross-entropy loss function. We run 500 training iterations (epochs) and perform validation stopping, saving the model weights which produced the lowest validation set loss. All models were trained on an NVIDIA Quadro M6000 GPU, with CUDA 9 and cuDNN v7, in Tensorflow [71], using the Keras API [72]. We omit the use of bias units in all convolutional layers. Note that, while all convolutions are one-dimensional, we use two-dimensional convolution functions for ease of software implementation. Our software implementation can be found at https://github.com/vlawhern/EEGModels. technically not out yet but will be when we resubmit

- In Block 1, we perform two convolutional steps in sequence. First, we fit F 2D convolutional filters of size $(1, 64)$, with the filter length chosen to be half the sampling rate of the data (here, 128Hz), outputting F filter maps containing the EEG signal at different band-pass frequencies. Setting the length of the temporal kernel at half the sampling rate allows for capturing frequency information at 2Hz and above. We then use a *Depthwise Convolution* [42]

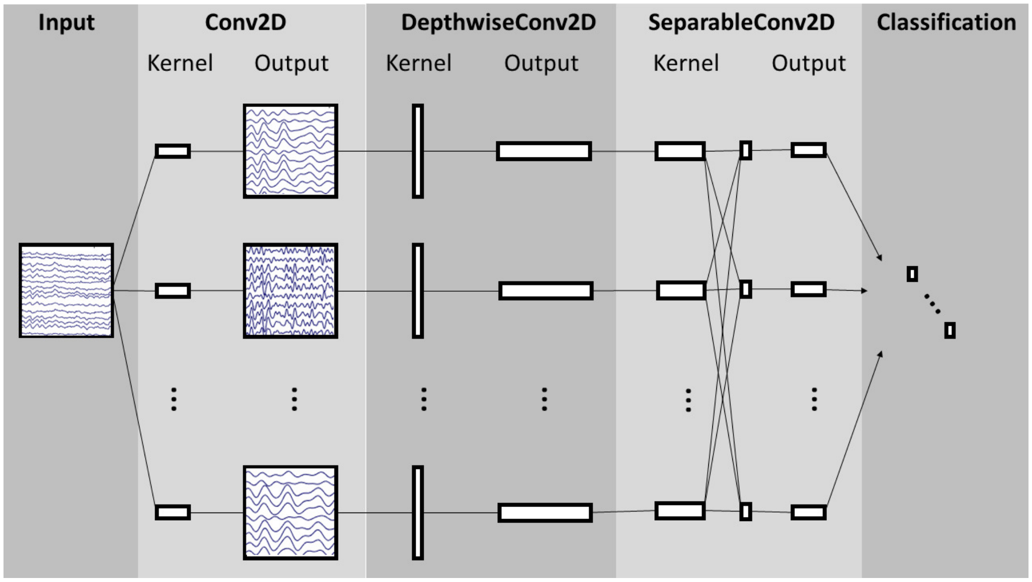

Figure 1: Overall visualization of the EEGNet architecture. Lines denote the convolutional kernel connectivity between inputs and outputs (called *filter maps*) . Here we see that the depthwise convolution (middle column) works along each filter map individually, while the separable convolution (fourth column) is a combination of a depthwise convolution, followed by a pointwise convolution, to optimally mix the filter maps. Full details about the network architecture can be found in Table 2.

of size $(C, 1)$. In CNN applications for computer vision the main benefit of a depthwise convolution is reducing the number of trainable parameters to fit, as these convolutions are not fully-connected to all previous filter maps (see Figure 1). Importantly, when used in EEG-specific applications, this operation provides a direct way to learn spatial filters within each filter map, thus enabling the efficient extraction of frequency-specific spatial filters (see the middle column of Figure 1). A depth parameter D controls the number of spatial filters to learn for each filter map ($D = 1$ is shown in Figure 1 for illustration purposes). This two-step convolutional sequence is inspired in part by the Filter-Bank Common Spatial Pattern (FBCSP) algorithm [73] and is similar in nature to another decomposition technique, Bilinear Discriminant Component Analysis [74]. We keep both convolutions linear as we found no significant gains in performance when using nonlinear activations. We apply Batch Normalization [75] along the filter map dimension before applying the exponential linear unit (ELU) nonlinearity [76]. To help regularize or model, we use the Dropout technique [77]. We set the dropout probability to 0.5 for within-subject classification to help prevent over-fitting when training on small sample sizes, whereas we set the dropout probability to 0.25 in cross-subject classification, as the training set sizes are much larger (see Section 2.3 for more details on our within- and cross-subject analyses). We apply an average pooling layer fo size (1, 4) to reduce the sampling rate of the signal to 32Hz. We also regularize each spatial filter by using a maximum norm constraint of 1 on its weights; $\|w\|^2 < 1$.

- In Block 2, we use a *Separable Convolution*, which is a Depthwise Convolution (here, of size $(1, 16)$, representing 500ms of EEG activity at 32Hz, the output sampling rate from

Block	Layer	# filters	size	# params	Output	Activation	Options
1	Input				(C, T)		
	Reshape				(1, C, T)		
	Conv2D	F	(1, 64)	$64 * F$	(F, C, T)	Linear	mode = same
	BatchNorm			$2 * F$	(F, C, T)		
	DepthwiseConv2D	D * F	(C, 1)	$C * D * F$	(D * F, 1, T)	Linear	mode = valid, depth = D, max norm = 1
	BatchNorm			$2 * F$	(D * F, 1, T)		
	Activation				(D * F, 1, T)	ELU	
	AveragePool2D		(1, 4)		(D * F, 1, T // 4)		
	Dropout*				(D * F, 1, T // 4)		$p = 0.25$ or $p = 0.5$
2	SeparableConv2D	D * F	(1, 16)	$16 * D * F + (D * F)^2$	(D * F, 1, T // 4)	Linear	mode = same
	BatchNorm			$D * F$	(D * F, 1, T // 4)		
	Activation				(D * F, 1, T // 4)	ELU	
	AveragePool2D		(1, 8)		(D * F, 1, T // 32)		
	Dropout*				(D * F, 1, T // 32)		$p = 0.25$ or $p = 0.5$
	Flatten				(D * F * (T // 32))		
Classifier	Dense	N * (D * F * T // 32)			N	Softmax	max norm = 0.25

Table 2: EEGNet architecture, where C = number of channels, T = number of time points, F = number of filters, D = depth multiplier (number of spatial filters) and N = number of classes, respectively. For the Dropout layer, we use $p = 0.5$ for within-subject classification and $p = 0.25$ for cross-subject classification (see Section 2.1.1 for more details)

Block 1) followed by a $(1, 1)$ Pointwise Convolution [42]. The main benefits of separable convolutions are (1) reducing the number of parameters to fit and (2) explicitly decoupling the relationship within and across filters by first learning a kernel within each filter map, then optimally merging the outputs afterwards. When used for EEG-specific applications this operation separates learning how to summarize individual filter maps in time (the depthwise convolution) with how to optimally combine the filters maps (the pointwise convolution). This operation is also particularly useful for EEG signals as different spatially filtered data may represent features at different time-scales of information. In our case we first learn a 500ms "summary" of each filter map, then combine the outputs. We fit $D * F$ filters using this approach. An Average Pooling layer of size $(1, 8)$ is used for dimension reduction.

- In the classification block, the features are passed directly to a softmax classification with N units, N being the number of classes in the data. We omit the use of a dense layer for feature aggregation prior to the softmax classification layer to reduce the number of free parameters in the model, inspired by the work in [78].

We investigate several different configurations of the EEGNet architecture by varying the number of filters, F, and the number of spatial filters, D to learn. This was done to examine the relationship between model size (the number of free parameters to estimate) and classification performance. We use the notation EEGNet-F,D to denote the number of temporal and spatial filters to learn; i.e.: EEGNet-4,2 denotes learning 4 temporal filters and 2 spatial filters per temporal filter.

	Trial Length (sec)	DeepConvNet	ShallowConvNet	EEGNet-4,2	EEGNet-8,2
P300	1	174,127	104,002	**1,066**	2,258
ERN	1.25	169,927	91,602	**1,082**	2,290
MRCP	1.5	175,727	104,722	**1,098**	2,322
SMR*	2	152,219	40,644	**796**	1,716

Table 3: Number of trainable parameters per model and per dataset for all CNN-based models. We see that the EEGNet models are up to two orders of magnitude smaller than both DeepConvNet and ShallowConvNet across all datasets. Note that we use a temporal kernel length of 32 samples for the SMR dataset as the data were high-passed at 4Hz.

2.2.2 Comparison with existing CNN Approaches

We compare the performance of EEGNet against the DeepConvNet and ShallowConvNet models proposed by [32]; full table descriptions of both models can be found in the Appendix. We implemented these models in Tensorflow and Keras, following the descriptions found in the paper. As their architectures were originally designed for 250Hz EEG signals (as opposed to 128Hz signals used here) we divided the lengths of temporal kernels and pooling layers in their architectures by 2 to correspond approximately to the sampling rate used in our models. We train these models in the same way we train the EEGNet model (see Section 2.2.1).

The DeepConvNet architecture consists of five convolutional layers with a softmax layer for classification (see Figure 1 of [32]). The ShallowConvNet architecture consists of two convolutional layers (temporal, then spatial), a squaring nonlinearity ($f(x) = x^2$), an average pooling layer and a log nonlinearity ($f(x) = \log(x)$). We would like to emphasize that the ShallowConvNet architecture was designed specifically for oscillatory signal classification (by extracting log band-power features); thus, it may not work well on ERP-based classification tasks. However, the DeepConvNet architecture was designed to be a general-purpose architecture that is not restricted to specific feature types [32], and thus it serves as a more valid comparison to EEGNet. Table 3 shows the number of trainable parameters per model across all CNN models.

2.2.3 Comparison with Traditional Approaches

We also compare the performance of EEGNet to that of the best performing traditional approach for each individual paradigm. For all ERP-based data analyses (P300, ERN, MRCP) the traditional approach is the approach which won the Kaggle BCI Competition (code and documentation at http://github.com/alexandrebarachant/bci-challenge-ner-2015), which uses a combination of xDAWN Spatial Filtering [79], Riemannian Geometry [80, 81], channel subset selection and L_1 feature regularization (referred to as xDAWN + RG for the remainder of the manuscript). Here we provide a summary of the approach, which is done in five steps:

1. Train two set of 5 xDAWN spatial filters, one set for each class of a binary classification task, using the ERP template concatenation method as described in [81, 82].

2. Perform EEG electrode selection through backward elimination [83] to keep only the most relevant 35 channels.

3. Project the covariance matrices onto the tangent space using the log-euclidean metric [80,84].

4. Perform feature normalization of the covariance matrices using an L_1 ratio of 0.5, signifying an equal weight for L_1 and L_2 penalties. An L_1 penalty encourages the sum of the absolute values of the parameters to be small, whereas an L_2 penalty encourages the sum of the squares of the parameters to be small (a theoretical overview of these penalties can be found in [85]).

5. Perform classification using an Elastic Net regression.

We use the same xDAWN+RG model parameters across all comparisons (P300, ERN, MRCP) with the exception of the number of EEG channels to use, which was set to 56 for ERN and 64 for P300 and MRCP. While the original solution used an ensemble of classifiers using bagging, for this analysis we only compared a single model with this approach to a single EEGNet model on identical training and test sets, as we expect any gains from ensemble learning to benefit both approaches equally. The original solution also used a set of "meta features" that were specific to that data collection. As the goal of this work is to investigate a general-purpose CNN model for EEG-based BCIs, we omitted the use of these features as they are specific to that particular data collection.

For oscillatory-based classification of SMR, the traditional approach is our own implementation of the One-Versus-Rest (OVR) filter-bank common spatial pattern (FBCSP) algorithm as described in [73]. Here we provide a brief summary of our approach:

1. Bandpass filter the EEG signal into 9 non-overlapping filter banks in 4Hz steps, starting at 4Hz: 4-8Hz, 8-12Hz, ..., 36-40Hz.

2. As the classification problem is multi-class, we use OVR classification, which requires that we train a classifier for all pairs of OVR combinations, which there are 4 here (class 1 vs all others, class 2 vs all others, etc). We train 2 CSP filter pairs (4 filters total) for each filter bank on the training data using the auto-covariance shrinkage method by [86]. This will give a total of 36 features (9 filter banks $\times$ 4 CSP filters) for each trial.

3. Train an elastic-net logistic regression classifier [87] for each OVR combination. We set the elastic net penalty $\alpha = 0.95$.

4. Find the optimal λ value for the elastic-net logistic regression that maximizes the validation set accuracy by evaluating the trained classifiers on a held-out validation set. The multi-class label for each trial is the classifier that produces the highest probability among the 4 OVR classifiers.

5. Apply the trained classifiers to the test set, using the λ values obtained in Step 4.

Note that this approach differs slightly from the original technique as proposed in [73], where they use a Naive Bayes Parzen Window classifier. We opted to use an elastic net regression for ease of implementation, and the fact that it has been used in existing software implementations of FBCSP (for example, in BCILAB [88]).

2.3 Data Analysis

Classification results are reported for two sets of analyses: within-subject and cross-subject. Within-subject classification uses a portion of the subjects data to train a model specifically for that subject, while cross-subject classification uses the data from other subjects to train a subject-agnostic model. While within-subject models tend to perform better than cross-subject models on a variety of tasks, there is ongoing research investigating techniques to minimize (or possibly eliminate) the need for subject-specific information to train robust systems [44, 51].

For within-subject, we use four-fold blockwise cross-validation, where two of the four blocks are chosen to be the training set, one block as the validation set, and the final block as testing. We perform statistical testing using a repeated-measures Analysis of Variance (ANOVA), modeling classification results (AUC for P300/MRCP/ERN and Classification Accuracy for SMR) as the response variable with subject number and classifier type as factors. For cross-subject analysis in P300 and MRCP we choose, at random, 4 subjects for the validation set, one subject for the test set, and all remaining subjects for the training set (see Table 1 for number of subjects per dataset). This process was repeated 30 times, producing 30 different folds. We follow the same procedure for the ERN dataset, except we use the 10 test subjects from the original Kaggle Competition as the test set for each fold. We perform statistical testing using a one-way Analysis of Variance, using classifier type as the factor. For the SMR dataset, we partitioned the data as follows: For each subject, select the training data from 5 other subjects at random to be the training set and the training data from the remaining 3 subjects to be the validation set. The test set remains the same as the original test set for the competition. Note that this enforces a fully cross-subject classification analysis. This process is repeated 10 times for each subject, creating 90 different folds. The mean and standard error of classification performance were calculated over the 90 folds. We perform statistical testing for this analysis using the same testing procedure as the within-subject case.

When training both the within-subject and cross-subject models, we apply a class-weight to the loss function whenever the data is imbalanced (unequal number of trials for each class). The class-weight we apply is the inverse of the proportion in the training data, with the majority class set to 1. For example, in the P300 dataset, there is an approximate 80%/20% split between non-target and target trials (corresponding to a 4:1 odds, see Table 1) . In this case the class-weight for non-targets was set to 1, while the class-weight for targets was set to 5. This procedure was applied to the P300 and ERN datasets only, as these were the only datasets where significant class imbalance was present.

Note that for the SMR analysis, we set the temporal kernel length to be 32 samples long (as opposed to 64 samples long as given in Table 2) since the data was high-passed at 4Hz.

2.4 EEGNet Feature Explainability

need better lead in.. The development of methods for enabling feature explainability from deep neural networks has become an active research area over the past few years, and has been proposed as an essential component of a robust model validation procedure, to ensure that the classification performance is being driven by relevant features as opposed to noise or artifacts in the data [16, 89–95]. We present three different approaches for understanding the features derived by EEGNet:

1. **Summarizing averaged outputs of hidden unit activations:** This approach focuses on summarizing the activations of hidden units at layers specified by the user. In this work we choose to summarize the hidden unit activations representing the data after the depthwise convolution (the spatial filter operation in EEGNet). Because the spatial filters are tied directly to a particular temporal filter, they provide additional insights into the spatial localization of narrow-band frequency activity. Here we summarize the spatially-filtered data by calculating averaged time-frequency representations, using Morlet wavelets [96].

2. **Visualizing the convolutional kernel weights:** This approach focuses on directly visualizing and interpreting the convolutional kernel weights from the model. Generally speaking, interpreting the convolutional kernel weights is very difficult due to the cross-filter-map connectivity between any two layers. However, because we limit the connectivity of the convolutional layers (using depthwise and separable convolutions), it is possible to interpret the convolutional kernels derived from EEGNet.

3. **Calculating single-trial feature *relevance* on the classification decision:** This approach focuses on calculating, on a single-trial basis, the *relevance* of individual features on the resulting classification decision. Positive values of relevance denote evidence supporting the outcome, while negative values of relevance denote evidence against the outcome. In our analysis we used DeepLIFT with the Rescale rule [93], as implemented in [94], to calculate single-trial EEG feature relevance. DeepLIFT is a gradient-based relevance attribution method, similar to Layerwise Relevance Propagation (LRP) used previously for EEG analysis [97], which calculates relevance values per feature relative to a "reference" input (here, an input of zeros, as is suggested in [93]). Perturbation-based methods, on the other hand, directly perturb the input and measure the resulting change on the output (a perturbation-based method was used previously for EEG relevance analysis in [32]). A summary of gradient-based and perturbation-based relevance attribution methods can be found in [94]. This analysis can be used to elucidate feature relevance from high-confidence versus low-confidence predictions, and can be used to confirm that the features learned are plausible, as opposed to noise or artifact features.

3 Results

3.1 Within-Subject Classification

We compare the performance of both the CNN-based reference algorithms (DeepConvNet and ShallowConvNet) and the traditional approach (xDAWN+RG for P300/MRCP/ERN and FBCSP for SMR) with EEGNet-4,2 and EEGNet-8,2. Within-subject four-fold cross-validation results across all algorithms for P300, MRCP and ERN datasets are shown in Figure 2. We observed,

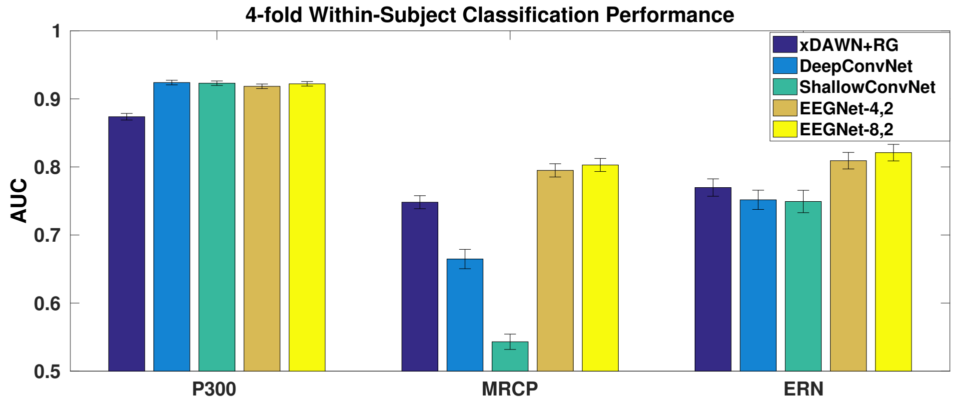

Figure 2: 4-fold within-subject classification performance for the P300, ERN and MRCP datasets for each model, averaged over all folds and all subjects. Error bars denote 2 standard errors of the mean. We see that, while there is minimal difference between all the CNN models for the P300 dataset, there are significant differences in the MRCP dataset, with both EEGNet models outperforming all other models. For the ERN dataset we also see both EEGNet models performing better than all others ($p < 0.05$).

across all paradigms, that there was no statistically significant difference between EEGNet-4,2 and EEGNet-8,2 ($p > 0.05$), indicating that the increase in model complexity did not statistically improve classification performance. For the P300 dataset, all CNN-based models significantly outperform xDAWN+RG ($p < 0.05$) while not performing significantly different amongst themselves. For the ERN dataset, EEGNet-8,2 outperforms DeepConvNet, ShallowConvNet and xDAWN+RG ($p < 0.05$), while EEGNet-4,2 outperforms DeepConvNet and ShallowConvNet ($p < 0.05$). The biggest difference observed among all the approaches is in the MRCP dataset, where both EEGNet models statistically outperform all others by a significant margin (DeepConvNet, ShallowConvNet

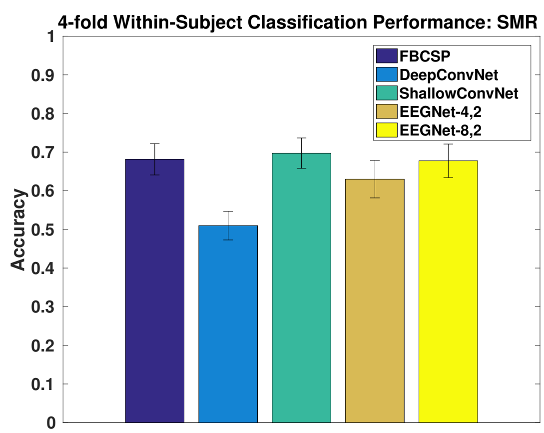

Figure 3: 4-fold within-subject classification performance for the SMR dataset for each model, averaged over all folds and all subjects. Error bars denote 2 standard errors of the mean. Here we see DeepConvNet statistically performed worse than all other models ($p < 0.05$). ShallowConvNet and EEGNet-8,2 performed similarly to that of FBCSP.

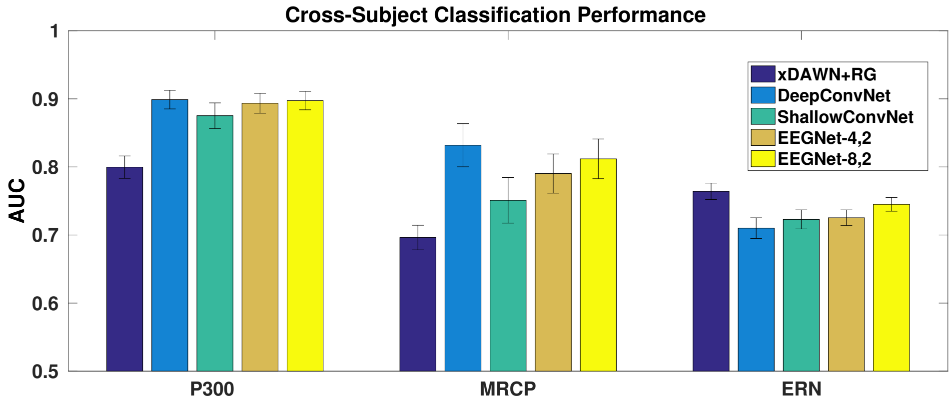

Figure 4: Cross-Subject classification performance for the P300, ERN and MRCP datasets for each model, averaged for 30 folds. Error bars denote 2 standard errors of the mean. For the P300 and MRCP datasets there is minimal difference between the DeepConvNet and the EEGNet models, with both models outperform the ShallowConvNet. For the ERN dataset the reference algorithm (xDAWN + RG) significantly outperforms all other models.

and xDAWN+RG, $p < 0.05$ for each comparison). This is interesting to note, as the effective model size of the EEGNet models is up to two orders of magnitude smaller than DeepConvNet and ShallowConvNet (see Table 3), representing improved model efficiency.

Four-fold cross-validation results for the SMR dataset are shown in Figure 3. Here we see the performances of ShallowConvNet and FBCSP are very similar, replicating previous results as reported in [32], while DeepConvNet performance is significantly lower. We also see that EEGNet-8,2 performance is similar to FBCSP as well.

3.2 Cross-Subject Classification

Cross-subject classification results across all algorithms for P300, MRCP and ERN datasets are shown in Figure 4. Similar to the within-subject analysis, we observed no statistical difference between EEGNet-4,2 and EEGNet-8,2 across all datasets ($p > 0.05$). For the P300 dataset, all CNN-based models significantly outperform xDAWN+RG ($p < 0.05$) while not performing significantly different amongst themselves. For the MRCP dataset EEGNet-8,2 and DeepConvNet significantly outperform ShallowConvNet ($p < 0.05$). We also see that both DeepConvNet and ShallowConvNet performance is significantly better when compared to its within-subject performance for the MRCP dataset. For the ERN dataset, xDAWN + RG outperforms all CNN models ($p < 0.05$). Cross-subject classification results for the SMR dataset are shown in Figure 5, where we found no significant difference in performance across all CNN-based algorithms ($p > 0.05$).

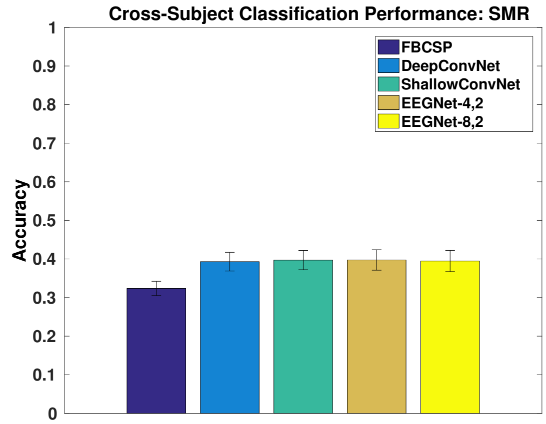

Figure 5: Cross-Subject classification performance for the SMR for each model, averaged over all folds and all subjects. Error bars denote 2 standard errors of the mean. We see that all CNN-based models perform similarly, while slightly outperforming FBCSP.

3.3 EEGNet Feature Characterization

We illustrate three different approaches to characterize the features learned by EEGNet on three different datasets: (1) Summarizing averaged outputs of hidden unit activations, (2) visualizing convolutional kernel weights, and (3) calculating single-trial feature relevances on classification decision. We illustrate Approach 1 on the P300 dataset for an EEGNet-4,1 model. We chose to analyze the filters from the P300 dataset due to the fact that multiple neurophysiological events occur simultaneously: participants were told to press a button with their dominant hand whenever a target image appeared on the screen. Because of this, target trials contain both the P300 event-related potential as well as the alpha/beta desynchronizations in contralateral motor cortex due to button presses. Here we were interested in whether the EEGNet architecture was capable of separating out these confounding events. We were also interested in quantifying the classification performance of the architecture whenever specific filters were removed from the model.

Figure 6 shows the spatial topographies of the four filters along with an average wavelet time-frequency difference, calculated using Morlet wavelets [96], between all target trials and all non-target trials. Here we see four distinct filters appear. The time-frequency analysis of Filter 1 shows an increase in low-frequency power approximately 500ms after image presentation, followed by desynchronizations in alpha frequency, suggesting that this filter is extracting the button press response. As nearly all subjects in the P300 dataset are right-handed, we also see significant activity along the left motor cortex. Time-frequency analysis of Filter 2 appears to show a significant theta-beta relationship; while increases in theta activity have been previously noted in the P300 literature in response to targets [98], a relationship between theta and beta has not previously been noted. The time-frequency difference for Filter 4 appears to correspond with the P300, with an increase low-frequency power approximately 350ms after image presentation.

We also conducted a feature ablation study, where we iteratively removed a set of filters (by replacing the filters with zeros) and re-applied the model to predict trials in the test set. We do this for all combinations of the four filters. Classification results for this ablation study are shown in

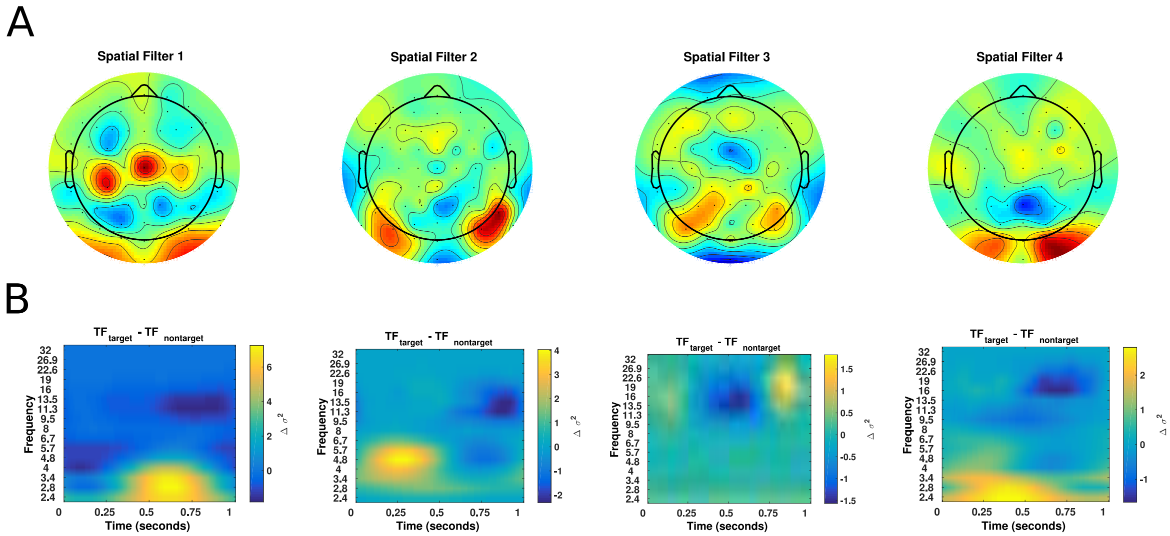

Figure 6: Visualization of the features derived from an EEGNet-4,1 model configuration for one particular cross-subject fold in the P300 dataset. (A) Spatial topoplots for each spatial filter. (B) The mean wavelet time-frequency difference between target and non-target trials for each individual filter.

Table 4. We see that test set performance is minimally impacted by the removal of any single filter, with the largest decrease occurring when removing Filter 4. As expected, when removing pairs of filters the decrease in performance is more pronounced, with the largest decrease observed when removing Filters 3 and 4. Removing Filters 2 and 3 results in practically no change in classification performance when compared to the full model, suggesting that the most important features in this task are being captured by Filters 1 and 4. This finding is further reinforced when looking

Filters Removed	Test Set AUC
(1)	0.8866
(2)	**0.9076**
(3)	0.8910
(4)	0.8747
(1, 2)	0.8875
(1, 3)	0.8593
(1, 4)	0.8325
(2, 3)	**0.8923**
(2, 4)	0.8721
(3, 4)	0.8206
(1, 2, 3)	**0.8637**
(1, 2, 4)	0.8202
(1, 3, 4)	0.7108
(2, 3, 4)	0.7970
None	**0.9054**

Table 4: Performance of the EEGNet-4,1 model when removing certain filters from the model, then using the model to predict the test set for one randomly chosen fold of the cross-subject P300 analysis. AUC values in bold denote the best performing model when removing 1, 2 or 3 filters at a time. As the number of filters removed increases, we see decreases in classification performance, although the magnitude of the decrease depends on which filters are removed.

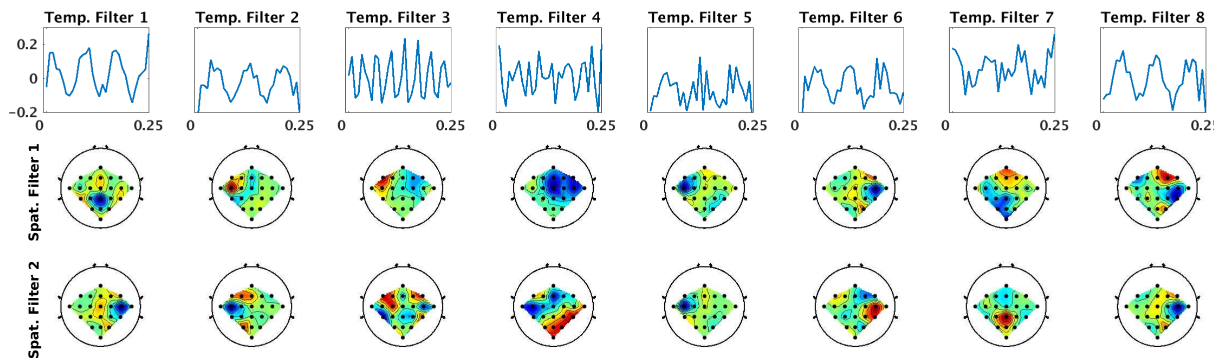

Figure 7: Visualization of the features derived from EEGNet-8,2 for Subject 3 of the SMR dataset. Each of the 8 columns shows the learned temporal kernel for a 0.25 second window (top) with its two associated spatial filters (bottom two). We see that, while many of the temporal filters are isolating slower-wave activity, the network identifies a higher-frequency filter at approximately 32Hz (Temp. Filter 3, which shows 8 cycles in a 0.25s window).

at classification performance when three filters are removed; a model that contains only Filter 4 (0.8637 AUC) performs fairly well when compared to models that contain only Filter 2 (0.7108 AUC) or Filter 1 (0.7970 AUC).

We now illustrate Approach 2 (visualizing the convolutional kernel weights) using the SMR dataset. Figure 7 shows the filters learned for the EEGNet-8,2 model for a within-subject classification of Subject 3 for the SMR dataset. Note that we are learning temporal filters of length 32, which correspond to 0.25s in time; hence, we estimate the frequency for each temporal filter as four times the number of observed cycles. Here we see that EEGNet-8,2 learns both slow-frequency activity at approximately 12Hz (Filters 1, 2 and 8, which show three cycles in a 0.25s window) and high-frequency activity at approximately 32Hz (Filter 3, which show 8 cycles).

Approach 3 is illustrated in Figure 8, which shows the single-trial feature relevances for EEGNet-8,2, calculated using DeepLIFT, for three three different test trials for one cross-subject fold of the MRCP dataset. Here we see that the high-confidence predictions (Figure 8A and Figure 8B, for left and right finger movement, respectively) both correctly show the contralateral motor cortex relevance as expected, whereas for a low-confidence prediction (Figure 8C), the feature relevance is more broadly distributed, both in time and in space on the scalp.

4 Discussion

In this work we proposed *EEGNet*, a compact convolutional network for EEG-based BCIs that can generalize across different BCI paradigms in the presence of limited data and can produce interpretable features. We evaluated EEGNet against the state-of-the-art approach for both ERP and Oscillatory-based BCIs across four EEG datasets: P300 visual-evoked potentials, Error-Related Negativity (ERN), Movement-Related Cortical Potentials (MRCP) and Sensory Motor Rhythms (SMR). To the best of our knowledge, this represents the first work that has validated the use of a

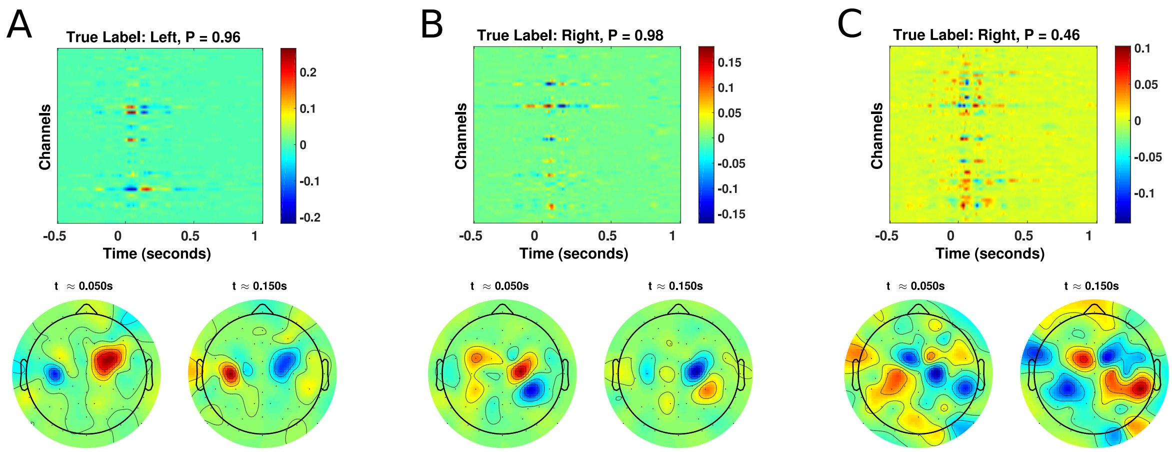

Figure 8: (Top row) Single-trial EEG feature relevance for EEGNet-8,2, using DeepLIFT, for three different test trials for one cross-subject fold of the MRCP dataset: (A) a high-confidence, correct prediction of left finger movement, (B) a high-confidence, correct prediction of right finger movement and (C) a low-confidence, incorrect prediction of left finger movement. Titles include the true class label and its predicted probability. (Bottom row) Spatial topoplots of the relevances at two time points: approximately 50 ms and 150 ms after button press. As expected, the high-confidence trials show the correct relevances corresponding to contralateral motor cortex for left (A) and right (B) button presses, respectively. For the low-confidence trial we see the relevances are more mixed and broadly distributed, without a clear spatial localization to motor cortices.

single network architecture across multiple BCI datasets, each with their own feature characteristics and data set sizes. Our work introduced the use of Depthwise and Separable Convolutions [42] to EEG signal classification, and showed that they can be used to significantly reduce model complexity (the number of free parameters to fit) while simultaneously improving model performance. Finally, through the use of feature visualization and ablation analysis, we show that neurophysiologically interpretable features can be extracted from the EEGNet model, providing further validation and evidence that the network performance is not being driven by noise or artifact signals in the data. This last finding is particularly important, as it is a critical component to understanding the validity and robustness of CNN model architectures not just for EEG [97], but for CNN architectures in general [95].

Generally speaking, the classification performance of DeepConvNet and EEGNet were similar across all cross-subject analyses, whereas DeepConvNet performance was lower across nearly all within-subject analyses (with the exception of P300). One possible explanation for this discrepancy is the amount of training data used to train the model; in cross-subject analyses the training set sizes are about 10-15 times larger than that of within-subject analyses. This suggests that DeepConvNet is more data-intensive compared to EEGNet, an unsurprising result given that the model size of DeepConvNet is two orders of magnitude larger than EEGNet (see Table 3). We believe this intuition is consistent with the findings originally reported by the developers of DeepConvNet [32], where they state that a training data augmentation strategy by cropping was needed to obtain good classification performance. While the cropping strategy they propose performs well for oscillatory signal classification, it is unclear how to apply such a strategy to an ERP signal (or more generally, a time-locked signal). In contrast to their work, we show that EEGNet performed well across all tested datasets, regardless of the size of the training data.

In general we found that, both in within- and cross-subject analyses, that ShallowConvNet tended to perform worse on the ERP BCI datasets than on the oscillatory BCI dataset (SMR), while the opposite behavior was observed with DeepConvNet. We believe this is due to the fact that the ShallowConvNet architecture was designed specifically to extract log bandpower features; in situations where the dominant feature is signal amplitude (as is the case in many ERP BCIs), ShallowConvNet performance tended to suffer. The opposite situation occurred with DeepConvNet; as its architecture wasn't designed to extract frequency features, its performance was a bit lower in situations where frequency power is the dominant feature. In contrast, we found that EEGNet performed just as well as ShallowConvNet in SMR classification and just as well as DeepConvNet in ERP classification (and outperforming in the case of within-subject MRCP and ERN classifications), suggesting that EEGNet is robust enough to learn a wide variety of features over a range of BCI tasks.

The severe underperformance of ShallowConvNet on within-subject MRCP classification was unexpected, given the similarity in neural responses between the MRCP and SMR, and the fact that ShallowConvNet performed well on SMR. This discrepancy in performance is not due to the amount of training data used, as the within-subject MRCP classification has approximately 1000 training trials, evenly split among left and right finger movements, whereas the SMR dataset has only 192 training trials, evenly split among four classes. In addition, we did not observe large deviations in ShallowConvNet performance on the other datasets (P300 and ERN). It is currently unclear of the source of this phenomena. VL: anything else to add here?

The learning capacity of CNNs comes in part from their ability to automatically extract intricate feature representations from raw data. However, since the features are not hand-designed by human engineers, understanding the meaning of those features poses a significant challenge in producing interpretable models [91]. This is especially true when CNNs are used for the analysis of EEG data where features from neural signals are often non-stationary and corrupted by noise artifacts [99,100]. In this study, we illustrated three different approaches for visualizing the features learned by EEGNet: (1) analyzing spatial filter outputs, averaged over trials, on the P300 dataset, (2) visualizing the convolutional kernel weights on the SMR dataset, and (3) performing single-trial relevance analysis on the MRCP dataset. In addition, we conducted a feature ablation study to understand the impact of a classification decision on the presence or absence of a particular feature. In each of these analyses, we showed that EEGNet was capable of extracting interpretable components that generally corresponded to known neurophysiological phenomena, a critical aspect that should be incorporated into future CNN model development pipelines for EEG [32,97]. These results confirm that the classification performances we observed were not due to artifact or noise sources in the data.

Deep Learning models for EEG generally employ one of three input styles, depending on their targeted application: (1) the EEG signal of all available channels, (2) a transformed EEG signal (generally a time-frequency decomposition) of all available channels [36] or (3) a transformed EEG signal of a subset of channels [37]. Models that fall in (2) generally see a significant increase in data dimensionality, thus requiring either more data or more model regularization (or both) to learn an effective feature representation. This introduces more hyperparameters that must be learned, increasing the potential variability in model performance due to hyperparameter misspecification. Models that fall in (3) generally require *a priori* knowledge about the channels to select. For example, the model proposed in [37] uses the time-frequency decomposition of channels Cz, C3 and C4 as the inputs for a motor imagery classification task. This channel selection is intentional, given the fact that neural responses to motor actions (the sensory motor rhythm) are observed strongest at those channels and are easily observed through a time-frequency analysis. Also, by only working with three channels, the authors reduce the significant increase in dimensionality of the data. While this approach works well if the feature of interest is known beforehand, this approach is not guaranteed to work well in other applications where the features are not observed at those channels, limiting the overall utility of this approach. We believe models that fall in (1), such as EEGNet and others [28, 30, 31], offer the best tradeoff between input dimensionality and the flexibility to discover relevant features by providing all available channels. This is especially important as BCI technologies evolve into novel application spaces, as the features needed for these future BCIs may not be known beforehand [7–12].

Acknowledgments

This project was sponsored by the U.S. Army Research Laboratory under CAST 076910227001, ARL-H70-HR52, ARL-74A-HRCYB and through Cooperative Agreement Number W911NF-10-2-0022. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

Conflict of Interest Statement

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

5 Appendix

5.1 DeepConvNet and ShallowConvNet architectures

The DeepConvNet and ShallowConvNet architectures are given in Tables 5 and 6, respectively. The DeepConvNet was designed to be a general-purpose architecture that is not restricted to specific feature types, whereas ShallowConvNet is designed specifically for oscillatory signal classification.

Layer	**# filters**	**size**	**# params**	**Activation**	**Options**
Input		(C, T)			
Reshape		(1, C, T)			
Conv2D	25	(1, 5)	150	Linear	mode = valid, max norm = 2
Conv2D	25	(C, 1)	25 * 25 * C + 25	Linear	mode = valid, max norm = 2
BatchNorm			2 * 25		epsilon = 1e-05, momentum = 0.1
Activation				ELU	
MaxPool2D		(1, 2)			
Dropout					p = 0.5
Conv2D	50	(1, 5)	25 * 50 * C + 50	Linear	mode = valid, max norm = 2
BatchNorm			2 * 50		epsilon = 1e-05, momentum = 0.1
Activation				ELU	
MaxPool2D		(1, 2)			
Dropout					p = 0.5
Conv2D	100	(1, 5)	50 * 100 * C + 100	Linear	mode = valid, max norm = 2
BatchNorm			2 * 100		epsilon = 1e-05, momentum = 0.1
Activation				ELU	
MaxPool2D		(1, 2)			
Dropout					p = 0.5
Conv2D	200	(1, 5)	100 * 200 * C + 200	Linear	mode = valid, max norm = 2
BatchNorm			2 * 200		epsilon = 1e-05, momentum = 0.1
Activation				ELU	
MaxPool2D		(1, 2)			
Dropout					p = 0.5
Activation				log	
Flatten					
Dense				softmax	max norm = 0.5

Table 5: DeepConvNet architecture, where C = number of channels, T = number of time points and N = number of classes, respectively.

Layer	**# filters**	**size**	**# params**	**Activation**	**Options**
Input		(C, T)			
Reshape		(1, C, T)			
Conv2D	40	(1, 13)	560	Linear	mode = same, max norm = 2
Conv2D	40	(C, 1)	40 * 40 * C	Linear	mode = valid, max norm = 2
BatchNorm			2 * 40		epsilon = 1e-05, momentum = 0.1
Activation				square	
AveragePool2D		(1, 75), stride (1, 15)			
Activation				log	
Flatten					
Dropout					p = 0.5
Dense				softmax	max norm = 0.5

Table 6: ShallowConvNet architecture, where C = number of channels, T = number of time points and N = number of classes, respectively. Here, the 'square' and 'log' activation functions are given as $f(x) = x^2$ and $f(x) = \log(x)$, respectively. Note that we clip the log function such that the minimum input value is a very small number ($\epsilon = 10e^{-7}$) for numerical stability.

References

[1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, “Brain-computer interfaces for communication and control.” *Clinical neurophysiology : official journal of the International Federation of Clinical Neurophysiology*, vol. 113, no. 6, pp. 767–91, jun 2002. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/12048038

[2] A. B. Schwartz, X. T. Cui, D. Weber, and D. W. Moran, “Brain-controlled interfaces: Movement restoration with neural prosthetics,” *Neuron*, vol. 52, no. 1, pp. 205 – 220, 2006.

[3] L. F. Nicolas-Alonso and J. Gomez-Gil, “Brain computer interfaces, a review,” *Sensors*, vol. 12, no. 2, p. 1211, 2012.

[4] A. Bashashati, M. Fatourechi, R. K. Ward, and G. E. Birch, “A survey of signal processing algorithms in brain–computer interfaces based on electrical brain signals,” *Journal of Neural Engineering*, vol. 4, no. 2, p. R32, 2007. [Online]. Available: http://stacks.iop.org/1741-2552/4/i=2/a=R03

[5] D. J. McFarland, C. W. Anderson, K. R. Muller, A. Schlogl, and D. J. Krusienski, “Bci meeting 2005-workshop on bci signal processing: feature extraction and translation,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 2, pp. 135–138, June 2006.

[6] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, “A review of classification algorithms for eeg-based brain–computer interfaces,” *Journal of Neural Engineering*, vol. 4, no. 2, p. R1, 2007. [Online]. Available: http://stacks.iop.org/1741-2552/4/i=2/a=R01

[7] T. O. Zander and C. Kothe, “Towards passive brain–computer interfaces: applying brain–computer interface technology to human–machine systems in general,” *Journal of Neural Engineering*, vol. 8, no. 2, p. 025005, 2011. [Online]. Available: http://stacks.iop.org/1741-2552/8/i=2/a=025005

[8] B. J. Lance, S. E. Kerick, A. J. Ries, K. S. Oie, and K. McDowell, “Brain-computer interface technologies in the coming decades,” *Proceedings of the IEEE*, vol. 100, no. Special Centennial Issue, pp. 1585–1599, May 2012.

[9] S. Saproo, J. Faller, V. Shih, P. Sajda, N. R. Waytowich, A. Bohannon, V. J. Lawhern, B. J. Lance, and D. Jangraw, “Cortically coupled computing: A new paradigm for synergistic human-machine interaction,” *Computer*, vol. 49, no. 9, pp. 60–68, Sept 2016.

[10] J. van Erp, F. Lotte, and M. Tangermann, “Brain-Computer Interfaces: Beyond Medical Applications,” *Computer*, vol. 45, no. 4, pp. 26–34, Apr. 2012.

[11] B. Blankertz, M. Tangermann, C. Vidaurre, S. Fazli, C. Sannelli, S. Haufe, C. Maeder, L. E. Ramsey, I. Sturm, G. Curio, and K. R. Müeller, “The berlin brain-computer interface: Non-medical uses of bci technology,” *Frontiers in Neuroscience*, vol. 4, no. 198, 2010.

[12] S. M. Gordon, M. Jaswa, A. J. Solon, and V. J. Lawhern, “Real world bci: Cross-domain learning and practical applications,” in *Proceedings of the 2017 ACM Workshop on An Application-oriented Approach to BCI out of the Laboratory*, ser. BCIforReal ’17. New York, NY, USA: ACM, 2017, pp. 25–28. [Online]. Available: http://doi.acm.org/10.1145/3038439.3038444

[13] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups,” *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, Nov 2012.

[14] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, pp. 436–444, 2015.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[16] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: http://arxiv.org/abs/1409.1556

[17] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” *CoRR*, vol. abs/1409.4842, 2014. [Online]. Available: http://arxiv.org/abs/1409.4842

[18] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[19] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, "Densely connected convolutional networks," *CoRR*, vol. abs/1608.06993, 2016. [Online]. Available: http://arxiv.org/abs/1608.06993

[20] J. Schmidhuber, "Deep learning in neural networks: An overview," *arXiv*, vol. abs/1404.7828, 2014. [Online]. Available: http://arxiv.org/abs/1404.7828

[21] A. Antoniades, L. Spyrou, C. C. Took, and S. Sanei, "Deep learning for epileptic intracranial eeg data," in *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, Sept 2016, pp. 1–6.

[22] J. Liang, R. Lu, C. Zhang, and F. Wang, "Predicting seizures from electroencephalography recordings: A knowledge transfer strategy," in *2016 IEEE International Conference on Healthcare Informatics (ICHI)*, Oct 2016, pp. 184–191.

[23] A. Page, C. Shea, and T. Mohsenin, "Wearable seizure detection using convolutional neural networks with transfer learning," in *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2016, pp. 1086–1089.

[24] P. Mirowski, D. Madhavan, Y. LeCun, and R. Kuzniecky, "Classification of patterns of {EEG} synchronization for seizure prediction," *Clinical Neurophysiology*, vol. 120, no. 11, pp. 1927 – 1940, 2009.

[25] P. Thodoroff, J. Pineau, and A. Lim, "Learning robust features using deep learning for automatic seizure detection," *CoRR*, vol. abs/1608.00220, 2016. [Online]. Available: http://arxiv.org/abs/1608.00220

[26] S. Stober, D. J. Cameron, and J. A. Grahn, "Using convolutional neural networks to recognize rhythm stimuli from electroencephalography recordings," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 1449–1457.

[27] S. Stober, A. Sternin, A. M. Owen, and J. A. Grahn, "Deep feature learning for EEG recordings," *CoRR*, vol. abs/1511.04306, 2015. [Online]. Available: http://arxiv.org/abs/1511.04306

[28] H. Cecotti and A. Graser, "Convolutional neural networks for p300 detection with application to brain-computer interfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 433–445, March 2011.

[29] R. Manor and A. Geva, "Convolutional neural network for multi-category rapid serial visual presentation bci," *Frontiers in Computational Neuroscience*, vol. 9, no. 146, 2015.

[30] J. Shamwell, H. Lee, H. Kwon, A. R. Marathe, V. Lawhern, and W. Nothwang, "Single-trial eeg rsvp classification using convolutional neural networks," pp. 983 622–983 622–10, 2016. [Online]. Available: http://dx.doi.org/10.1117/12.2224172

[31] H. Cecotti, M. P. Eckstein, and B. Giesbrecht, "Single-trial classification of event-related potentials in rapid serial visual presentation tasks using supervised spatial filtering," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 11, pp. 2030–2042, Nov 2014.

[32] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball, "Deep learning with convolutional neural networks for eeg decoding and visualization," *Human Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, 2017. [Online]. Available: http://dx.doi.org/10.1002/hbm.23730

[33] M. Längkvist, L. Karlsson, and A. Loutfi, "Sleep stage classification using unsupervised feature learning," *Adv. Artif. Neu. Sys.*, vol. 2012, pp. 5:5–5:5, Jan. 2012. [Online]. Available: http://dx.doi.org/10.1155/2012/107046

[34] D. F. Wulsin, J. R. Gupta, R. Mani, J. A. Blanco, and B. Litt, "Modeling electroencephalography waveforms with semi-supervised deep belief nets: fast classification and anomaly measurement," *Journal of Neural Engineering*, vol. 8, no. 3, p. 036015, 2011.

[35] T. Ma, H. Li, H. Yang, X. Lv, P. Li, T. Liu, D. Yao, and P. Xu, "The extraction of motion-onset vep bci features based on deep learning and compressed sensing," *Journal of Neuroscience Methods*, vol. 275, pp. 80 – 92, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165027016302680

[36] P. Bashivan, I. Rish, M. Yeasin, and N. Codella, "Learning representations from EEG with deep recurrent-convolutional neural networks," *CoRR*, vol. abs/1511.06448, 2015. [Online]. Available: http://arxiv.org/abs/1511.06448

[37] Y. R. Tabar and U. Halici, "A novel deep learning approach for classification of eeg motor imagery signals," *Journal of Neural Engineering*, vol. 14, no. 1, p. 016003, 2017. [Online]. Available: http://stacks.iop.org/1741-2552/14/i=1/a=016003

[38] X. An, D. Kuang, X. Guo, Y. Zhao, and L. He, *A Deep Learning Method for Classification of EEG Data Based on Motor Imagery.* Cham: Springer International Publishing, 2014, pp. 203–210. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-09330-7_25

[39] S. Sakhavi, C. Guan, and S. Yan, "Parallel convolutional-linear neural network for motor imagery classification," in *2015 23rd European Signal Processing Conference (EUSIPCO)*, Aug 2015, pp. 2736–2740.

[40] N. Lu, T. Li, X. Ren, and H. Miao, "A deep learning scheme for motor imagery classification based on restricted boltzmann machines," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 6, pp. 566–576, June 2017.

[41] Z. Yin and J. Zhang, "Cross-session classification of mental workload levels using eeg and an adaptive deep learning model," *Biomedical Signal Processing and Control*, vol. 33, pp. 30 – 47, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1746809416302014

[42] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *CoRR*, vol. abs/1610.02357, 2016. [Online]. Available: http://arxiv.org/abs/1610.02357

[43] Z. Yang, M. Moczulski, M. Denil, N. d. Freitas, A. Smola, L. Song, and Z. Wang, "Deep fried convnets," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 1476–1483.

[44] F. Lotte, "Signal Processing Approaches to Minimize or Suppress Calibration Time in Oscillatory Activity-Based Brain-Computer Interfaces," *Proceedings of the IEEE*, vol. 103, no. 6, pp. 871–890, Jun. 2015.

[45] R. Fazel-Rezai, B. Z. Allison, C. Guger, E. W. Sellers, S. C. Kleih, and A. Kübler, "P300 brain computer interface: current challenges and emerging trends," *Frontiers in Neuroengineering*, vol. 5, no. 14, 2012.

[46] G. Pfurtscheller and C. Neuper, "Motor imagery and direct brain-computer communication," *Proceedings of the IEEE*, vol. 89, no. 7, pp. 1123–1134, Jul 2001.

[47] S. Makeig, "Auditory event-related dynamics of the {EEG} spectrum and effects of exposure to tones," *Electroencephalography and Clinical Neurophysiology*, vol. 86, no. 4, pp. 283 – 293, 1993.

[48] J. Polich, "Updating p300: An integrative theory of {P3a} and {P3b}," *Clinical Neurophysiology*, vol. 118, no. 10, pp. 2128 – 2148, 2007.

[49] P. Sajda, E. Pohlmeyer, J. Wang, L. C. Parra, C. Christoforou, J. Dmochowski, B. Hanna, C. Bahlmann, M. K. Singh, and S. F. Chang, "In a blink of an eye and a switch of a transistor: Cortically coupled computer vision," *Proceedings of the IEEE*, vol. 98, no. 3, pp. 462–478, March 2010.

[50] A. R. Marathe, V. J. Lawhern, D. Wu, D. Slayback, and B. J. Lance, "Improved neural signal classification in a rapid serial visual presentation task using active learning," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 3, pp. 333–343, March 2016.

[51] N. Waytowich, V. Lawhern, A. Bohannon, K. Ball, and B. Lance, "Spectral transfer learning using information geometry for a user-independent brain-computer interface," *Frontiers in Neuroscience*, vol. 10, p. 430, 2016. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnins.2016.00430

[52] A. Delorme and S. Makeig, "Eeglab: an open source toolbox for analysis of single-trial {EEG} dynamics including independent component analysis," *Journal of Neuroscience Methods*, vol. 134, no. 1, pp. 9 – 21, 2004.

[53] W. H. R. Miltner, C. H. Braun, and M. G. H. Coles, "Event-related brain potentials following incorrect feedback in a time-estimation task: Evidence for a "generic" neural system for error detection," *Journal of Cognitive Neuroscience*, vol. 9, no. 6, pp. 788–798, 1997.

[54] W. J. Gehring, B. Goss, M. G. H. Coles, D. E. Meyer, and E. Donchin, "A neural system for error detection and compensation," *Psychological Science*, vol. 4, no. 6, pp. 385–390, 1993. [Online]. Available: http://pss.sagepub.com/content/4/6/385.abstract

[55] M. Falkenstein, J. Hohnsbein, J. Hoormann, and L. Blanke, "Effects of crossmodal divided attention on late {ERP} components. ii. error processing in choice reaction tasks," *Electroencephalography and Clinical Neurophysiology*, vol. 78, no. 6, pp. 447 – 455, 1991.

[56] P. Margaux, M. Emmanuel, D. Sébastien, B. Olivier, and M. Jérémie, "Objective and subjective evaluation of online error correction during p300-based spelling," *Advances in Human-Computer Interaction*, vol. 2012, p. 4, 2012.

[57] T. O. Zander, C. Kothe, S. Welke, and M. Roetting, *Utilizing Secondary Input from Passive Brain-Computer Interfaces for Enhancing Human-Machine Interaction.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 759–771.

[58] J. d. R. Millán, R. Rupp, G. Mueller-Putz, R. Murray-Smith, C. Giugliemma, M. Tangermann, C. Vidaurre, F. Cincotti, A. Kubler, R. Leeb, C. Neuper, K. R. Müeller, and D. Mattia, "Combining brain-computer interfaces and assistive technologies: State-of-the-art and challenges," *Frontiers in Neuroscience*, vol. 4, no. 161, 2010.

[59] M. Spüler, M. Bensch, S. Kleih, W. Rosenstiel, M. Bogdan, and A. Kübler, "Online use of error-related potentials in healthy users and people with severe motor impairment increases performance of a p300-bci," *Clinical Neurophysiology*, vol. 123, no. 7, pp. 1328 – 1337, 2012.

[60] D. J. Krusienski, E. W. Sellers, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw, "Toward enhanced p300 speller performance," *Journal of neuroscience methods*, vol. 167, no. 1, pp. 15–21, 2008.

[61] C. Toro, G. Deuschl, R. Thatcher, S. Sato, C. Kufta, and M. Hallett, "Event-related desynchronization and movement-related cortical potentials on the {ECoG} and {EEG}," *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section*, vol. 93, no. 5, pp. 380 – 389, 1994.

[62] E. C. Leuthardt, G. Schalk, D. Moran, and J. G. Ojemann, "The emerging world of motor neuroprosthetics: a neurosurgical perspective," *Neurosurgery*, vol. 59, no. 1, pp. 1–14, 2006.

[63] E. Yom-Tov and G. F. Inbar, "Detection of movement-related potentials from the electro-encephalogram for possible use in a brain-computer interface," *Medical and Biological Engineering and Computing*, vol. 41, no. 1, pp. 85–93, Jan 2003. [Online]. Available: https://doi.org/10.1007/BF02343543

[64] S. Gordon, V. Lawhern, A. Passaro, and K. McDowell, "Informed decomposition of electroencephalographic data," *Journal of Neuroscience Methods*, vol. 256, pp. 41 – 55, 2015.

[65] G. Pfurtscheller and A. Aranibar, "Event-related cortical desynchronization detected by power measurements of scalp {EEG}," *Electroencephalography and Clinical Neurophysiology*, vol. 42, no. 6, pp. 817 – 826, 1977.

[66] G. Pfurtscheller and F. L. da Silva, "Event-related eeg/meg synchronization and desynchronization: basic principles," *Clinical Neurophysiology*, vol. 110, no. 11, pp. 1842 – 1857, 1999.

[67] K. Liao, R. Xiao, J. Gonzalez, and L. Ding, "Decoding individual finger movements from one hand using human eeg signals," *PLoS ONE*, vol. 9, no. 1, pp. 1–12, 01 2014.

[68] N. Bigdely-Shamlo, T. Mullen, C. Kothe, K. M. Su, and K. A. Robbins, "The prep pipeline: Standardized preprocessing for large-scale eeg analysis," *Frontiers in Neuroinformatics*, vol. 9, no. 16, 2015.

[69] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K. Miller, G. Mueller-Putz, G. Nolte, G. Pfurtscheller, H. Preissl, G. Schalk, A. Schlögl, C. Vidaurre, S. Waldert, and B. Blankertz, "Review of the bci competition iv," *Frontiers in Neuroscience*, vol. 6, p. 55, 2012. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnins.2012.00055

[70] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiV*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[71] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," in *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, ser. OSDI'16. Berkeley, CA, USA: USENIX Association, 2016, pp. 265–283. [Online]. Available: http://dl.acm.org/citation.cfm?id=3026877.3026899

[72] F. Chollet, "Keras," https://github.com/fchollet/keras, 2015.

[73] K. K. Ang, Z. Y. Chin, C. Wang, C. Guan, and H. Zhang, "Filter bank common spatial pattern algorithm on bci competition iv datasets 2a and 2b," *Frontiers in Neuroscience*, vol. 6, p. 39, 2012. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnins.2012.00039

[74] M. Dyrholm, C. Christoforou, and L. C. Parra, "Bilinear discriminant component analysis," *Journal of Machine Learning Research*, vol. 8, no. May, pp. 1097–1111, 2007.

[75] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv*, vol. abs/1502.03167, 2015. [Online]. Available: http://arxiv.org/abs/1502.03167

[76] D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *CoRR*, vol. abs/1511.07289, 2015. [Online]. Available: http://arxiv.org/abs/1511.07289

[77] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014. [Online]. Available: http://jmlr.org/papers/v15/srivastava14a.html

[78] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. A. Riedmiller, “Striving for simplicity: The all convolutional net,” *arXiV*, vol. abs/1412.6806, 2014. [Online]. Available: http://arxiv.org/abs/1412.6806

[79] B. Rivet, A. Souloumiac, V. Attina, and G. Gibert, “xDAWN algorithm to enhance evoked potentials: Application to brain-computer interface,” *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 8, pp. 2035–2043, Aug 2009.

[80] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, “Multiclass Brain-Computer Interface Classification by Riemannian Geometry,” *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 4, pp. 920–928, Apr. 2012.

[81] A. Barachant and M. Congedo, “A Plug&Play P300 BCI Using Information Geometry,” *arXiv:1409.0107 [cs, stat]*, Aug. 2014, arXiv: 1409.0107. [Online]. Available: http://arxiv.org/abs/1409.0107

[82] M. Congedo, A. Barachant, and A. Andreev, “A new generation of brain-computer interface based on riemannian geometry,” *CoRR*, vol. abs/1310.8115, 2013. [Online]. Available: http://arxiv.org/abs/1310.8115

[83] A. Barachant and S. Bonnet, “Channel selection procedure using riemannian distance for bci applications,” in *2011 5th International IEEE/EMBS Conference on Neural Engineering*, April 2011, pp. 348–351.

[84] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, “Classification of covariance matrices using a Riemannian-based kernel for BCI applications,” *Neurocomputing*, vol. 112, pp. 172–178, Jul. 2013.

[85] A. Y. Ng, “Feature selection, l1 vs. l2 regularization, and rotational invariance,” in *Proceedings of the Twenty-first International Conference on Machine Learning*, ser. ICML ’04. New York, NY, USA: ACM, 2004, pp. 78–. [Online]. Available: http://doi.acm.org/10.1145/1015330.1015435

[86] O. Ledoit and M. Wolf, “A well-conditioned estimator for large-dimensional covariance matrices,” *Journal of Multivariate Analysis*, vol. 88, no. 2, pp. 365 – 411, 2004.

[87] H. Zou and T. Hastie, “Regularization and variable selection via the elastic net,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 2, pp. 301–320, 2005. [Online]. Available: http://dx.doi.org/10.1111/j.1467-9868.2005.00503.x

[88] C. A. Kothe and S. Makeig, “Bcilab: a platform for brain–computer interface development,” *Journal of Neural Engineering*, vol. 10, no. 5, p. 056014, 2013. [Online]. Available: http://stacks.iop.org/1741-2552/10/i=5/a=056014

[89] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K.-R. MÃžller, “How to explain individual classification decisions,” *Journal of Machine Learning Research*, vol. 11, no. Jun, pp. 1803–1831, 2010.

[90] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 818–833.

[91] A. M. Nguyen, J. Yosinski, and J. Clune, “Deep neural networks are easily fooled: High confidence predictions for unrecognizable images,” *CoRR*, vol. abs/1412.1897, 2014. [Online]. Available: http://arxiv.org/abs/1412.1897

[92] M. T. Ribeiro, S. Singh, and C. Guestrin, “”why should i trust you?”: Explaining the predictions of any classifier,” in *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD ’16. New York, NY, USA: ACM, 2016, pp. 1135–1144. [Online]. Available: http://doi.acm.org/10.1145/2939672.2939778

[93] A. Shrikumar, P. Greenside, and A. Kundaje, “Learning important features through propagating activation differences,” *CoRR*, vol. abs/1704.02685, 2017. [Online]. Available: http://arxiv.org/abs/1704.02685

[94] M. Ancona, E. Ceolini, C. Öztireli, and M. Gross, “Towards better understanding of gradient-based attribution methods for deep neural networks,” in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=Sy21R9JAW

[95] G. Montavon, W. Samek, and K.-R. Müller, “Methods for interpreting and understanding deep neural networks,” *Digital Signal Processing*, vol. 73, pp. 1 – 15, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1051200417302385

[96] C. Torrence and G. P. Compo, "A practical guide to wavelet analysis," *Bulletin of the American Meteorological society*, vol. 79, no. 1, pp. 61–78, 1998.

[97] I. Sturm, S. Lapuschkin, W. Samek, and K.-R. Müller, "Interpretable deep neural networks for single-trial eeg classification," *Journal of Neuroscience Methods*, vol. 274, pp. 141 – 145, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165027016302333

[98] A. Mazaheri and T. W. Picton, "Eeg spectral dynamics during discrimination of auditory and visual targets," *Cognitive Brain Research*, vol. 24, no. 1, pp. 81 – 96, 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0926641004003519

[99] G. Johnson, N. Waytowich, and D. J. Krusienski, "The challenges of using scalp-eeg input signals for continuous device control," in *Foundations of Augmented Cognition. Directing the Future of Adaptive Systems*, D. D. Schmorrow and C. M. Fidopiastis, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 525–527.

[100] V. Lawhern, W. D. Hairston, K. McDowell, M. Westerfield, and K. Robbins, "Detection and classification of subject-generated artifacts in {EEG} signals using autoregressive models," *Journal of Neuroscience Methods*, vol. 208, no. 2, pp. 181 – 189, 2012.