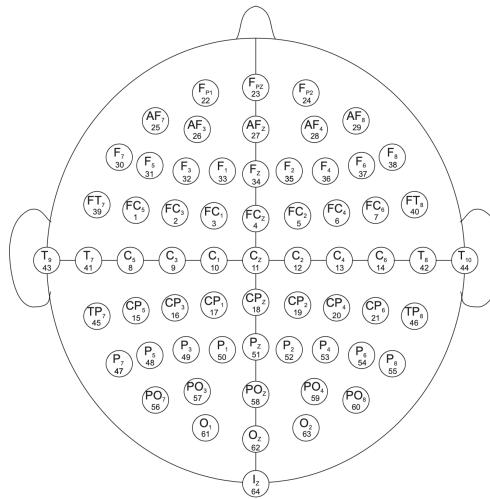


DEPARTMENT OF
INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING
Spring Semester 2020

Channel Selection for Low Power Brain-Computer Interfaces

Semester Project



Tianhong Gan
tiagan@student.ethz.ch

29/05/2020

Advisors: Michael Hersche, herschmi@iis.ee.ethz.ch
Xiaying Wang, xiaywang@iis.ee.ethz.ch
Dr. Michele Magno, michele.magno@iis.ee.ethz.ch

Professor: Prof. Dr. Luca Benini, lbenini@iis.ee.ethz.ch

Abstract

Achieving a high classification accuracy using electroencephalogram (EEG) based Brain-Computer Interfaces (BCIs) is a big challenge. Maximising the classification accuracy requires sophisticated models, which rely on many channels. This brings limitations when targeting a practical embedded system, due to the high power and memory requirements for both data acquisition and processing. It is therefore beneficial to reduce the number of channels as well as the model size. This project investigates channel selection for low power Motor-Imagery Brain-Computer Interfaces (MI-BCIs). It develops a systematic method of channel selection by targeting areas with high Motor-Imagery (MI) neural activity in the brain. To determine channels with the most significant neural activity, an investigation is conducted using Common Spatial Pattern (CSP) filter weights and EEGNet filter weights. Using EEGNet filter weights, the 4-class global model with the optimum configuration (with $N_{ch} = 16$, $ds = 3$ and $T = 3$ s) that satisfies the SRAM requirements of the Cortex-M4 processor obtains a validation accuracy of 63.45%, surpassing the state-of-the-art [1] by 0.94%. Moreover, Subject-Specific Transfer Learning (SS-TL) increases the validation accuracy by 5.28% on average. On this optimal model, SS-TL increases the validation accuracy by 4.44% to 67.89%. To satisfy the Flash memory requirements of Subject-Specific (SS) models, layer freezing is also investigated, which can drastically increase the number of SS models stored on a device with limited memory.

Acknowledgments

I would like thank my advisors, Xiaying Wang and Michael Hersche, for their advice and guidance during our weekly meetings and discussions. I would also like to thank my advisor, Dr. Michele Magno, for his support. They no doubt gave me the inspiration and motivation needed to remain on track and complete this project despite the ongoing global pandemic. Thank you also to Prof. Dr. Luca Benini, Prof. Frank Gürkaynak, and all the staff and colleagues at the Integrated Systems Laboratory (IIS) for providing me with essential resources and a warm, welcoming atmosphere during my stay.

Contents

1. Introduction	1
2. Background	3
2.1. EEGNet [2]	3
2.2. Physionet EEG Motor Movement/Imagery Dataset [3]	5
3. Implementation	7
3.1. Test Approaches	7
3.1.1. Inter-Subject Classification	7
3.1.2. Intra-Subject Classification	8
3.2. Channel Selection using CSP Weights	9
3.2.1. Multiscale CSP	9
3.2.2. Single Time Window and Frequency Band CSP	10
3.2.3. CSP-Rank [13]	11
3.3. Channel Selection using EEGNet Weights	11
3.4. Satisfying Memory Requirements	11
3.4.1. SRAM	11
3.4.2. Flash Memory and Layer Freezing	12
4. Results	13
4.1. Channel Selection	13
4.1.1. CSP Weights (Inter-Subject Classification)	13
4.1.2. EEGNet Weights (Inter-Subject Classification)	16
4.1.3. EEGNet Weights (Intra-Subject Classification)	22
4.2. Validation Accuracy vs Memory	22
4.2.1. Validation Accuracy vs SRAM	22
4.2.2. Validation Accuracy vs Flash Memory	24
5. Conclusion and Future Work	28
A. SS-TL Methods for Channel Selection	29

Contents

B. Channel Selection using EEGNet Weights Results	30
List of Acronyms	37
List of Figures	38
Bibliography	39

Chapter 1

Introduction

BCIs provide a means of direct communication between the human brain and an external system. An EEG uses non-invasive electrodes attached to the scalp to record neural activity from the brain. This electrical activity can then be processed and classified into commands, which manipulate an external system or device. MI is the mental execution of a movement without physically performing it [4]. MI-BCIs aim to discover patterns in the data recorded from EEGs in order to accurately classify MI and match it to the correct motor movement. This has a range of beneficial applications, including control for severely paralysed users [5], movement in virtual reality [6], and hands-free control [7].

Traditionally, MI-BCIs have been tackled using methods featuring complex pre-processing (i.e. feature extraction using Frequency Band Common Spatial Patterns (FBCSPs) [8] or Riemannian covariances [9]) and post-processing (i.e. classification using Linear Discriminant Analysis (LDA) or Support Vector Machines (SVMs)) steps. Convolutional Neural Networks (CNNs) have been introduced as a popular alternative approach, providing complete and concise models that process and classify raw input data in a simple manner (for example, EEGNet [2]). However, CNNs models are often computationally complex and resource heavy, and thus requires data to be transferred from the sensor node to an external computer to enable classification. This is unfeasible for use in real BCI applications due to the high power consumption, short battery life, latency and lack of portability. Where processing is done over a remote host, privacy concerns are also raised.

This work builds onto the investigation in [1], which makes use of temporal down-sampling, channel selection, and narrowing of the classification window in order to scale down the EEGNet model, to meet the memory requirements of conventional microcontrollers with negligible accuracy degradation. This brings forth the possibility of edge computing, reducing power consumption, latency, and negating privacy issues since processing is done locally on an embedded device.

1. Introduction

However, establishing a reliable and systematic method of channel selection can be tricky. In the original investigation [1], channel selection is performed using new or standard configurations with the aim of covering the whole 64-channel region of the brain equally. However, as MI should be associated with specific parts of the brain, a better approach would be to systematically select channels specifically from regions with high MI activity. This could further increase classification accuracies after the channel selection is performed, by selecting useful channels and eliminating noise from those least used.

This work explores the usage of CSP filter weights and EEGNet filter weights to ensure channels with the most neural activity are selected in the channel selection process. This work also explores two training methods, inter- and intra-subject classification, and compares the accuracies obtained both globally and for SS-TL. In terms of SS-TL, layer-freezing is introduced, which reduces the amount of Flash memory required to store a large number of SS models by freezing and reusing the weights for some layers. Finally, the result of this work is evaluated against the original work, both in terms of memory usage and validation accuracy.

Chapter 2

Background

2.1. EEGNet [2]

EEGNet is a compact CNN that makes use of depthwise and separable convolutions to accurately classify EEG signals for use in BCIs.

The model is made up of four main layers, as shown in Fig. 2.1:

- ϕ_1 is a 2D temporal convolution, which creates a kernel, N_f that is convolved with the layer input and produces a tensor of outputs. The number of kernels used in this layer is 8.
- ϕ_2 is a 2D depthwise convolution. The depth multiplier argument, which controls how many output channels are generated per input channel, is set to 2.
- ϕ_3 is a 2D depthwise separable convolution, which consists in first performing a depthwise spatial convolution followed by a pointwise convolution. The output width is divided by 8, which is the stride.

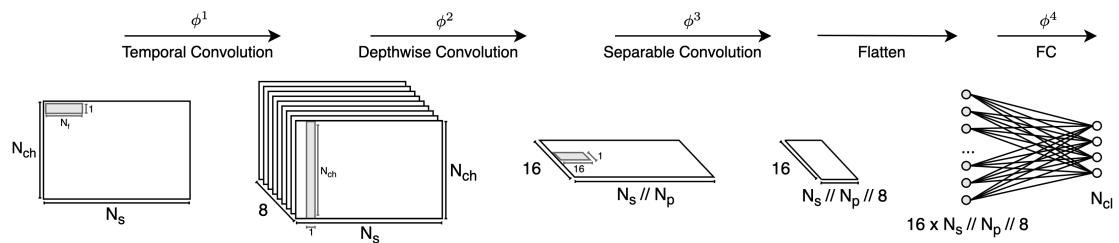


Figure 2.1.: EEGNet Architecture [1]: N_{ch} is the number of channels, N_s is the number of samples, N_p is the pooling length, N_{cl} is the number of classes and N_f is the filter size of the first temporal filter.

2. Background

Layer	Type	n	p	k	s	Parameters	Output Shape
ϕ^1	Conv2d	8	same	$N_f \times 1$	1×1	$8N_f$	$N_s \times N_{ch} \times 8$
	BatchNorm2d			-		32	
ϕ^2	DepthConv2d	16	valid	$1 \times N_{ch}$	1×0	$N_{ch} \cdot 16$	$1 \times N_s // N_p \times 16$
	BatchNorm2d			-		64	
	EluAct			-		-	
ϕ^3	AvgPool2d	-	valid	$N_p \times 1$	$N_p \times 1$	-	$1 \times N_s // N_p // 8 \times 16$
	SepConv2d	16	same	16×1	1×1	512	
	BatchNorm2d			-		64	
ϕ^4	EluAct			-		-	N_{cl}
	AvgPool2d	-	valid	8×1	1×8	-	
ϕ^4	FC	4		-		$(N_s // N_p // 8 \cdot 16 + 1)N_{cl}$	$N_s(9N_{ch} + 18/N_p) + N_{cl}$
	SoftMaxAct			-		-	
Total (inkl. Input Feature Map)				$672 + 16N_{ch} + 8N_f$		$+(2N_s/N_p + 1)N_{cl}$	$N_s(9N_{ch} + 18/N_p) + N_{cl}$

Figure 2.2.: Detailed description of EEGNet in MI classification copied from [1]. For visualisation, refer to Fig. 2.1. For each map, n is the number of filters, p is the padding strategy, k is the kernel size and s is the stride. Parameters refer to filter weights and output shape gives the feature map size.

- ϕ_4 is a fully connected layer that classifies the input into one of the N_{cl} classes (in the figure shown, $N_{cl} = 4$).

Fig. 2.2 shows the architecture of EEGNet in more depth, including the configuration used for each layer.

The standard configuration for 4-class MI on the Physionet dataset (Section 2.2), where no memory reduction methods are applied, are as follows:

- $N_s = 480$
- $N_{ch} = 64$
- $N_p = 8$
- $N_f = 128$

However, the process of channel selection, downsampling, and narrowing of the classification window may affect these parameters such that:

- $N_s = 160$ Hz x time window
- $N_{ch} =$ number of channels selected
- $N_p = \text{ceil}(8/\text{downsampling factor})$
- $N_f = \text{ceil}(128/\text{downsampling factor})$

2. Background

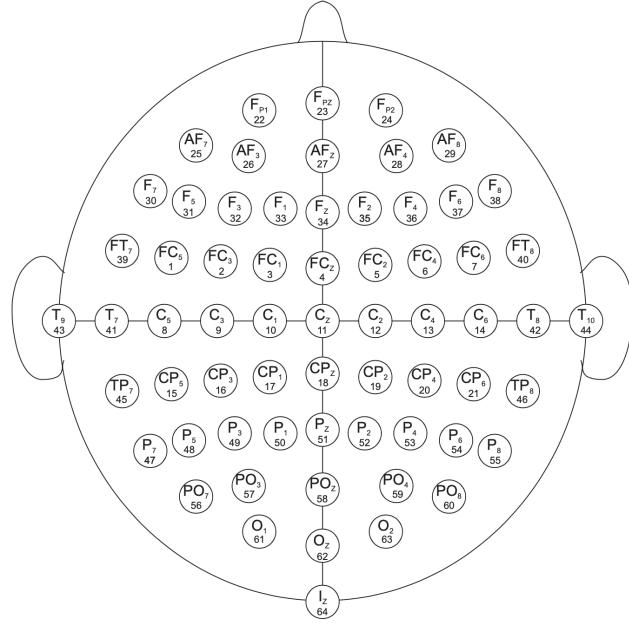


Figure 2.3.: 64-Electrode Configuration [3]

2.2. Physionet EEG Motor Movement/Imagery Dataset [3]

To train, validate and test the model, the Physionet EEG Motor Movement/Imagery Dataset will be used. The data was recorded using 64 EEG electrodes (labelled from 0 to 63) as per the international 10-10 system (excluding electrodes Nz, F9, F10, FT9, FT10, A1, A2, TP9, TP10, P9, and P10), as shown in Fig. 2.3.

These 64-channel EEG recordings consist of a range of motor movement/imagery tasks sampled at 160 Hz using the BCI2000 system [10], taken from 109 subjects. This project discards 4 subjects due to variability in the number of trials, leaving 105 subjects. Each subject performs 14 experimental runs, including two one-minute baseline runs.

This project uses the baseline runs (runs 1, 2), as well as the motor imagery runs, which consists of three two-minute runs of:

- Imagining opening and closing either the right or the left fist (runs 4, 8, 12),
- Imagining opening and closing either both fists or both feet (runs 6, 10, 14),

corresponding to a visual signal on the screen in front of the subject. The motor movement runs were not used.

2. Background

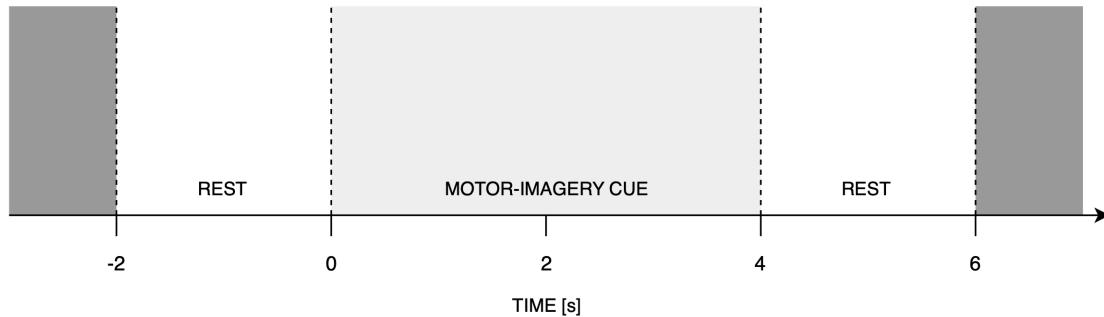


Figure 2.4.: Trial Paradigm of Physionet EEG Motor Movement/Imagery Dataset

For the baseline runs, subjects provide resting data, where they are required to do nothing and no cues are shown. For the motor imagery runs, subjects imagine the motor movement corresponding to a given cue. Each trial starts at -2s. A cue appears at 0s, lasting for 4s, where the subject should imagine the action specified by the cue. This is followed by a 2s rest before the next trial begins, as shown in Fig. 2.4.

For the purposes of this project, classes are defined as follows:

- Imagining opening and closing the left fist (L),
- Imagining opening and closing the right fist (R),
- Resting (no motor imagery) (0),
- Imagining clenching and unclenching both feet (F),

where the first two, three and four items on the list define the two-, three- and four-classes, respectively.

The data is provided in EDF+ format.

Further information regarding the dataset may be found on the Physionet website.

Chapter 3

Implementation

The investigation into how channel selection affects the validation accuracy will be investigated for 8-, 16- and 24-channels. The results for 19- and 38-channels will also be investigated and evaluated as a comparison to the accuracies obtained in the original work [1].

The models are based on EEGNet, and are trained and validated using Tensorflow and Keras, using the same base code as the original work [1]. The Physionet EEG motor movement/imagery dataset (Section 2.2) is used.

This work focuses on 4-class classification.

3.1. Test Approaches

Two test approaches, inter- and intra-subject classification, are evaluated in this work. The approaches are evaluated using the Physionet dataset on the EEGNet model, both globally and for SS-TL.

3.1.1. Inter-Subject Classification

Inter-subject classification is where the classifier is trained and tested using data from different subjects.

For the global case, 5-fold cross validation is performed using 105 subjects, as shown in Fig. 3.1. As can be seen, data from 80% of the subjects are used for training and 20% used for testing and validation. The training parameters used are taken from the previous investigation [11], featuring a dropout rate of 0.2, and a learning rate of 0.01, 0.001 and 0.0001 at the 1st, 20th and 50th epoch, respectively. Training is done over a total of 100 epochs per fold using the Adam optimizer.

3. Implementation

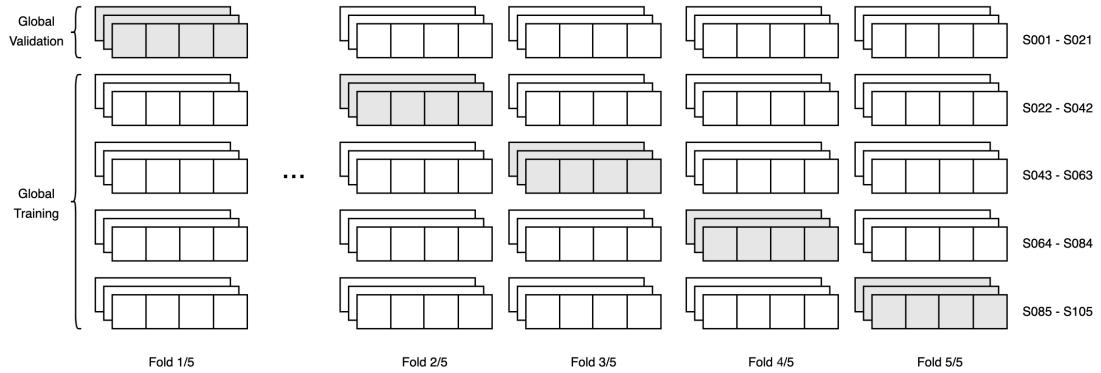


Figure 3.1.: Inter-Subject Global Classifier Paradigm

For the SS case, 4-fold cross validation is performed per subject, as shown in Fig. 3.2. For every subject, the SS-TL is performed using the global model for which the subject was not part of the global training, such that the SS-TL for each subject is trained and validated on new and unseen data. The training is completed with a learning rate of 0.001 and a dropout rate of 0.2 for 10 epochs.

For the case of channel selection, several SS-TL methods were explored and tested (details can be found in Appendix A). The highest accuracy was obtained using the following method:

1. Acquire trained 64-channel global model
2. Select N channels based on global EEGNet weights
3. Train from scratch N-channel global model
4. Perform SS-TL on N-channel global model

For the inter-subject case, the classifier is not tested using data from any subjects that it is trained on, for both the global and SS-TL models. Thus, the validation accuracies obtained should be an accurate representation for all new data.

3.1.2. Intra-Subject Classification

Intra-subject classification is where the classifier is trained (and tested) on part of the data from all subjects.

For the global case, 3-fold cross validation is performed across all subjects, as shown in Fig. 3.3. Global training is performed on two thirds of the data from each subject, and validation using the remaining third. For the Physionet case, runs (4, 6), (8, 10) and

3. Implementation

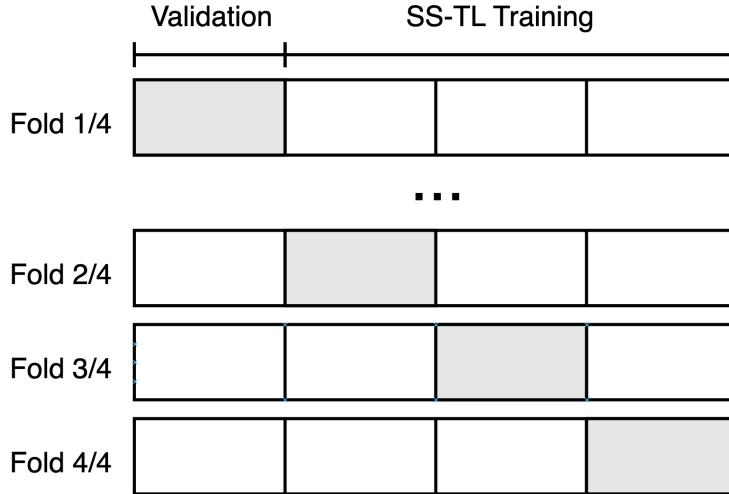


Figure 3.2.: Inter-Subject SS-TL Classifier Paradigm, for 1 Subject

(12, 14) are used interchangeably. The training parameters were again taken from the investigation in [11], with a learning rate of 0.0001 and a dropout rate of 0.1 trained over 100 epochs per fold.

For the SS case, SS-TL training is applied to the same two-thirds of the data for each subject as during global training, and validated on the remaining unseen one third of the data. The same method is used in the case of channel selection as for inter-subject classification.

3.2. Channel Selection using CSP Weights

This section uses the work from the feature extraction aspect of [12] to obtain a set of spatial filters that maximises the average variance between two classes. For the 4-class case, this results in a set of 12 spatial filters, two for each pair of classes in $\{(L,R), (L,0), (L,F), (R,0), (R,F), (0,F)\}$, each containing weights for all 64 channels. The filter weights for each channel signify the importance of that channel in the filter. This means that channels with a larger weight have a bigger impact on the resulting filtered signal and those with smaller weights are less significant.

3.2.1. Multiscale CSP

Multiscale CSP obtains a set of spatial filters for every combination of time window and frequency band. Each set of spatial filters have dimension [64, 12]. In order to perform

3. Implementation

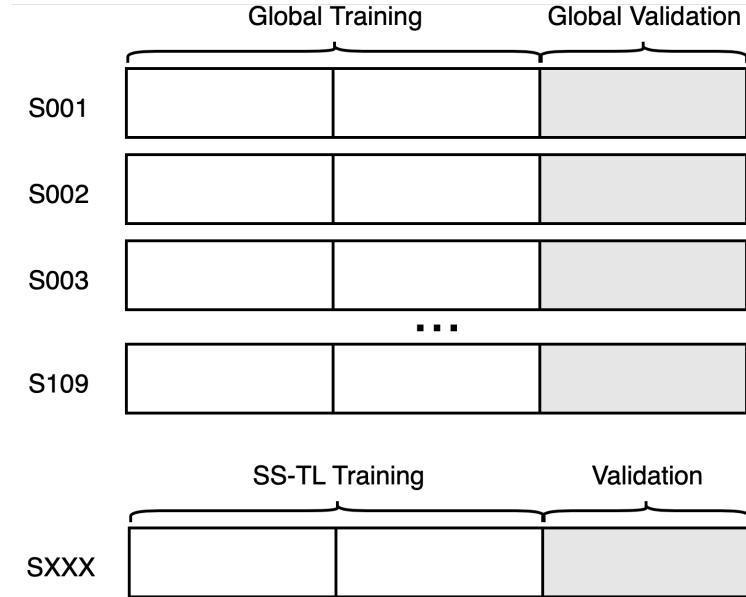


Figure 3.3.: Intra-Subject Global and SS-TL Classifier Paradigm

channel selection, the absolute weights for all channels across each combination of time windows and frequency bands, and for every filter set, are calculated and summed per channel and sorted in descending order. The channels with the highest value are then returned as the selected channels. The significance of each channel is calculated using the sum of the absolute values of all the weights for each channel as follows:

$$e^{(c)} = \sum_{i,k \in \{L,R,0,F\}} |w_{i,k}^{(c)}|, \quad (3.1)$$

where $e^{(c)}$ is the energy of channel c and $w_{i,k}^{(c)}$ is the value of the weight associated with channel c in filter (i,k) .

3.2.2. Single Time Window and Frequency Band CSP

MI activities generally cause brain oscillations in the μ (8–14 Hz) and β (14–30 Hz) bands. Additionally, most of the neural activity from the Physionet dataset occurs within 3 s from when the MI cue first appears. Instead of applying multiscale CSP to obtain several sets of spatial filters for different frequency bands and time windows, a single frequency band (4–30 Hz) and time window (0–3 s) is selected for this application, which should increase the classification accuracy. This produces a single set of spatial filters of dimension [64, 12]. The significance of each channel is again sorted according to the absolute value, as calculated in Eq. (3.1), and the top channels selected.

3. Implementation

3.2.3. CSP-Rank [13]

CSP-rank sorts the significance of the channels by taking the channel with the highest absolute weight from each filter respectively. For example, selecting the highest weight from filter one, followed by filter two, all the way until filter twelve, before alternating back to filter one. If the channel with the highest weight in a specific filter has already been selected, pick the channel with the next highest weight in that filter. The channels are sorted in this order, with the highest ranked channels selected. The filter weights used are those obtained from single time window and frequency band CSP.

3.3. Channel Selection using EEGNet Weights

An alternative approach is to perform channel selection directly using the EEGNet filter weights. Using the global models trained with 64 channels, access the filter weights for the ‘depthwise convolution 2D’ layer. Channel selection is then performed directly by sorting the sum of the absolute values in the filters for each of the 64 channels, and selecting the channels with the highest ranked sums.

3.4. Satisfying Memory Requirements

This project uses the B-L475-IOT01A2 microcontroller with an ARM Cortex-M4F processor, and the STM32F756 Nucleo-144 microcontroller with an ARM Cortex-M7 processor. The Cortex-M4 and Cortex-M7 processors have a Flash requirement of 1MB and a SRAM requirement of 128kB and 320kB, respectively. To ensure that the models with the highest validation accuracies chosen also satisfy the memory criteria of the microcontroller, the approximate Flash memory and SRAM required for models with different configurations must be calculated. The calculations assume weights are stored as 32-bit floating-point numbers, and are calculated using Eq. (3.2).

$$N_{weights} \times 4, \quad (3.2)$$

where $N_{weights}$ is the number of weights and 4 represents the number of bytes per weight.

Details regarding number of parameters and feature map size for each layer of the EEGNet architecture can be found in Section 2.1

3.4.1. SRAM

The amount of SRAM required depends on feature map size. Assuming at least two consecutive feature maps must be stored at any time, the maximum number of stored features can be calculated as the sum of the input and first layer, as shown in equation Eq. (3.3).

3. Implementation

$$N_s \times N_{ch} + N_s \times N_{ch} \times 8, \quad (3.3)$$

where N_s is the number of samples and N_{ch} is the number of channels.

As the SRAM needed depends on the feature map size, it is the same for both global and subject specific models.

3.4.2. Flash Memory and Layer Freezing

The Flash memory required depends on the total number of parameters required for the model. For the global model, this is just the sum of the number of parameters needed for each layer, as stated in Eq. (3.4). For the 4-class case, this is only approximately 13kB and way below the Flash requirement.

$$N_G = N_{\phi 1} + N_{\phi 2} + N_{\phi 3} + N_{\phi 4}, \quad (3.4)$$

where N_G is the number of parameters needed for the global model, and $N_{\phi x}$ is the number of parameters in layer x .

However, for the SS case, the total number of parameters depend on the number of subject specific models that are stored, and how much of the parameters differ between subjects. If a large number of subject specific models are stored, each with unique parameters, the amount of Flash memory needed would quickly increase linearly corresponding to the number of subjects, according to Eq. (3.5).

$$N_{parameters} = N_{subjects} \times N_G, \quad (3.5)$$

where $N_{parameters}$ is the number of parameters and $N_{subjects}$ is the number of subject specific models stored.

To prevent the large amount of Flash memory associated with storing several unique SS models, layer freezing may be applied, such that parameters for some layers are reused for all subjects. For example, in the case of the first layer being frozen, SS-TL is only applied to weights in the last three layers, and the first layer of weights is shared between all subject specific models. Eq. (3.6), Eq. (3.7) and Eq. (3.8) show the number of parameters needed in the SS case for different numbers of subjects and for one, two and three layers frozen respectively.

$$N_{parameters} = N_{\phi 1} + N_{subjects} \cdot (N_{\phi 2} + N_{\phi 3} + N_{\phi 4}) \quad (3.6)$$

$$N_{parameters} = N_{\phi 1} + N_{\phi 2} + N_{subjects} \cdot (N_{\phi 3} + N_{\phi 4}) \quad (3.7)$$

$$N_{parameters} = N_{\phi 1} + N_{\phi 2} + N_{\phi 3} + N_{subjects} \cdot N_{\phi 4} \quad (3.8)$$

Chapter 4

Results

4.1. Channel Selection

The following section discusses the results of channel selection performed using methods specified in Chapter 3. The validation accuracies of each method are analysed and compared to the accuracies obtained in the original work [1] as well as with each other.

4.1.1. CSP Weights (Inter-Subject Classification)

The classification accuracy of the channel selection obtained using CSP weights are shown in Fig. 4.1. Multiscale CSP and CSP-rank obtain similar results, with the lowest validation accuracy. As predicted, by limiting the time window and frequency band to the relevant sections, the validation accuracy obtained for single time window and frequency band CSP is significantly higher in comparison to multiscale CSP, by 2.31% on average across all channels. However, as can be observed, none of the channel selection methods managed to achieve a higher validation accuracy than the original channel selection, which is always superior by at least 0.67%.

The low accuracy obtained for CSP-rank was suspected to be due to the fact that the function iterates through 12 filters for 4-class classification (rather than just two filters in [13]). As a result, the large 'number of filters' to 'channels selected' ratio could mean the selected channels are not a fair representation of the filter values. To test this theory, the CSP-rank method was applied to 2-class classification and compared with results for the original selection, as shown in Fig. 4.2. As can be seen from the results, 2-class CSP-rank still obtains a lower accuracy than the original hand-picked positions.

To evaluate the failure of the CSP channel selection methods, a heatmap of the weights determining the selected channels for single time window and frequency band CSP (which obtained the highest accuracy of all the CSP channel selection methods) has been plotted. The heatmap in Fig. 4.3 shows the plot taken using the average of all 12 CSP filter

4. Results

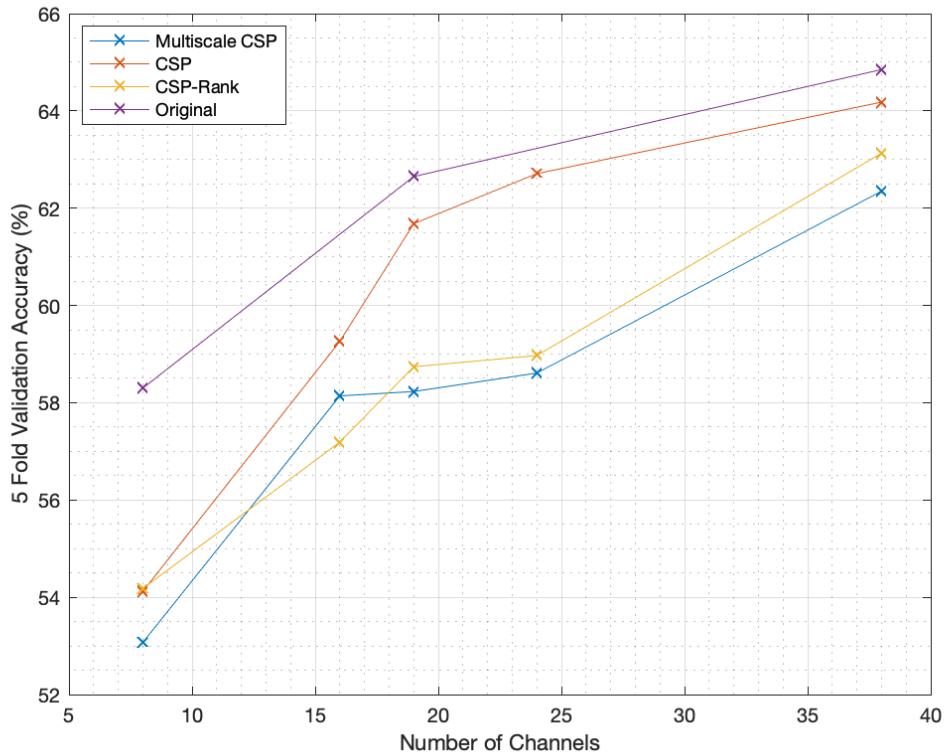


Figure 4.1.: Comparison of inter-subject validation accuracies obtained using channel selection with multiscale CSP, single time window and frequency band CSP, CSP-rank, and the original channel selection (classification with EEGNet) for 4-class classification ($ds=1$, $T=3$ s).

4. Results

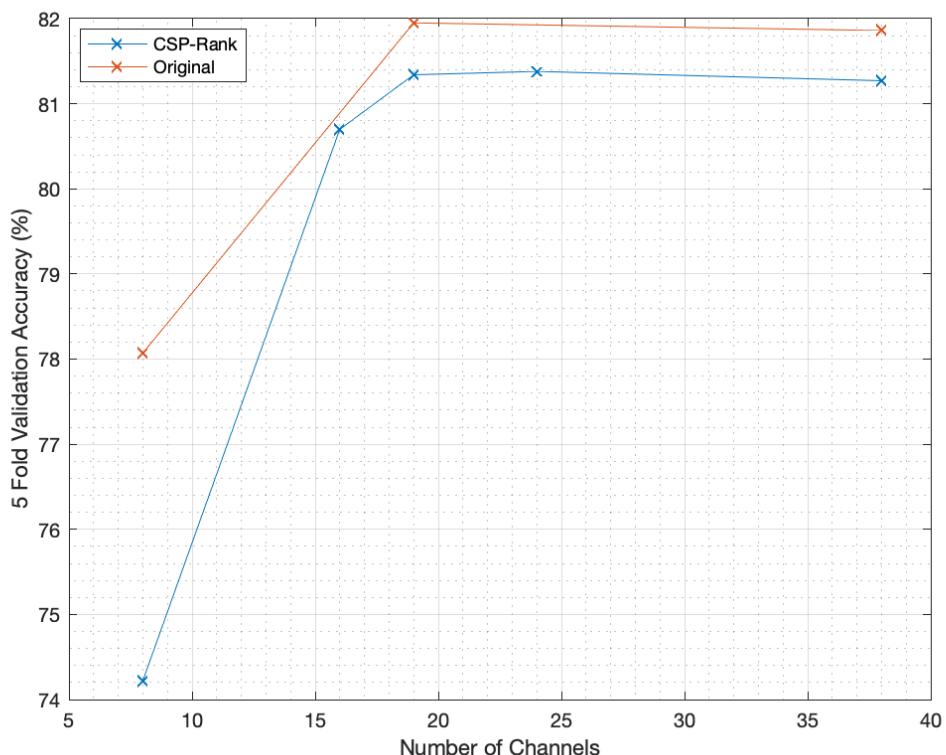


Figure 4.2.: Comparison of inter-subject validation accuracies obtained with CSP-rank and the original channel selection for 2-class classification ($ds=1$, $T=3$ s)

4. Results

weights used for 4-class CSP channel selection.

The channels with the largest weights are spaced around the central region of the 10-10 international system, however, appear to be missing the centre-most channels (C1, C2, C3, C4 and Cz). Neural activity for MI is generally associated with the C channels, especially C3 and C4, thus the unselected C channels in the central region may be the cause for the lower accuracies obtained from CSP channel selection.

A closer look into the filter weights can be seen in Fig. 4.4, where heatmaps are taken for each individual filter of the 12 CSP filters used in 4-class classification. As the filters are associated with maximising the average variance between $\{(L,R) (L,0) (L,F) (R,0) (R,F) (0,F)\}$ classes, it can be observed that to maximise the variance between two different classes, neural activity is higher in different parts of the brain. As such, channel selection using the highest weights on average between all the filters may not have been the best use of the CSP filter weights. Instead, a future exploration could be to use specific filters for channel selection for classification against different classes.

4.1.2. EEGNet Weights (Inter-Subject Classification)

The results of the inter-subject channel selection using EEGNet weights are shown in Fig. 4.5. As can be observed, channel selection using EEGNet weights allow accuracies to be obtained that surpass the original channel selection accuracy for both global models (by 0.77% on average) and SS models (by 2.23% on average).

SS-TL on the global models where channels are selected using EEGNet weights obtain a gain of 5.28% on average with respect to the corresponding global model on which the SS-TL was performed.

As a comparison to the heatmap visualising CSP weights, a similar heatmap has been plot for channel selection using EEGNet weights, as shown in Fig. 4.6. Here, we see that channels are selected mainly in the central region, including most of the C channels associated with MI neural activity, which may be the reason for the higher validation accuracy obtained by channel selection using EEGNet weights.

A closer look into how weights are distributed in specific filters is shown in Fig. 4.7. As can be observed, the weight distribution is a lot more random across different EEGNet filters, compared to CSP. This may be the reason taking the channels with the highest average weight across all the spatial filters obtained a more successful result for EEGNet in comparison with CSP.

4. Results

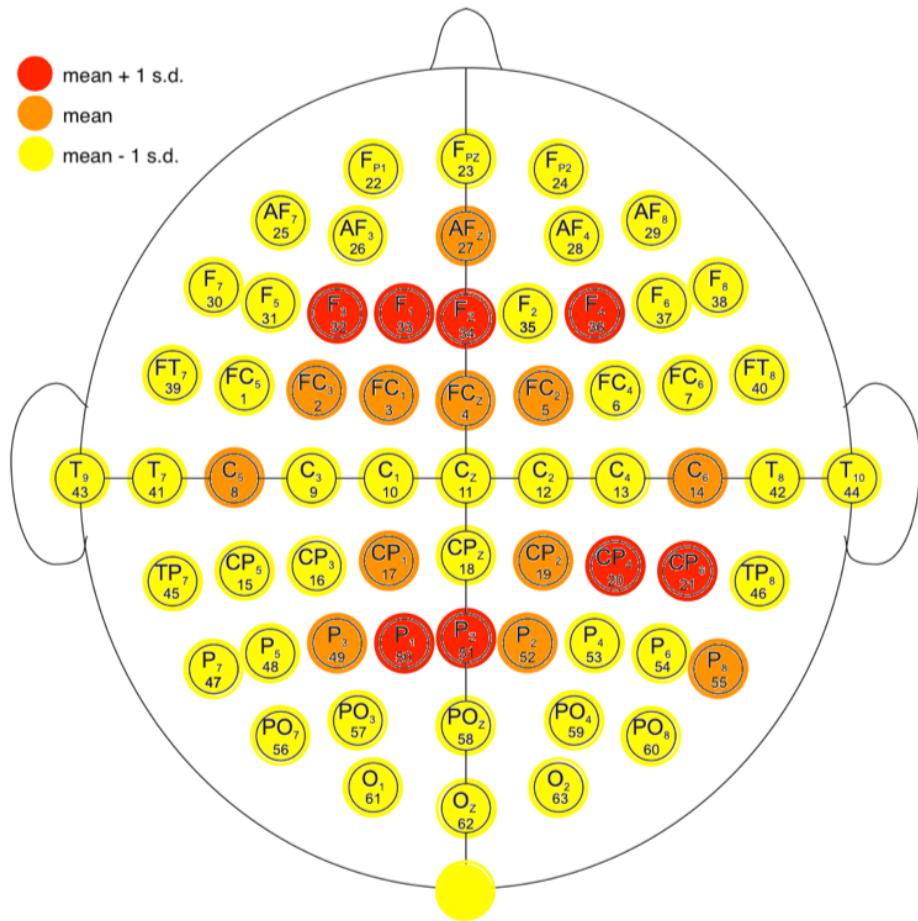


Figure 4.3.: Heatmap to visualise weights determining the selected channels for single time window and frequency band CSP, taken using the average of all 12 CSP filter weights used for 4-class CSP channel selection. Red values are above the mean plus one standard deviation, orange values are above the mean, and yellow values are above the mean minus one standard deviation.

4. Results

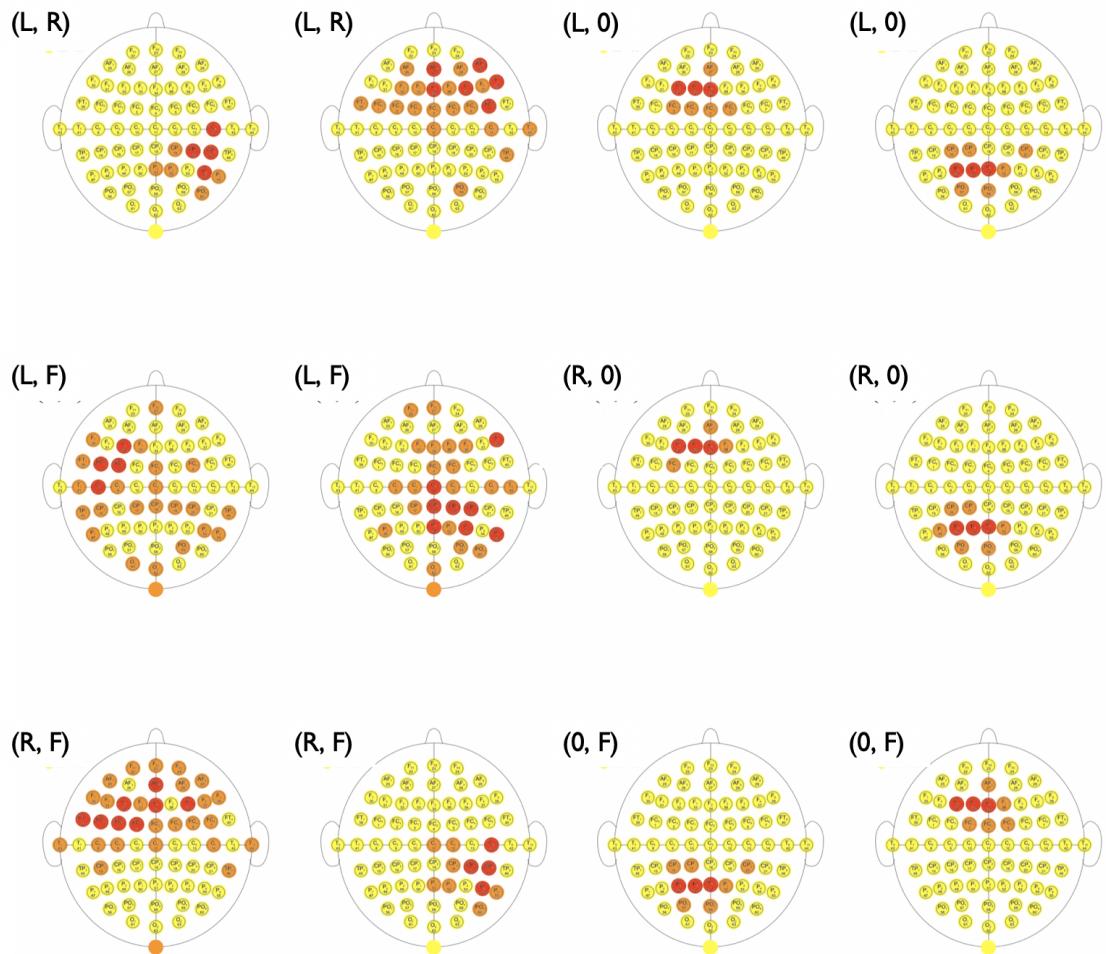


Figure 4.4.: Heatmaps for each individual filter in the 12 CSP filters used for 4-class classification. From left to right, top to bottom, the filters are associated with maximising the average variance between $\{(L,R) (L,0) (L,F) (R,0) (R,F) (0,F)\}$ classes.

4. Results

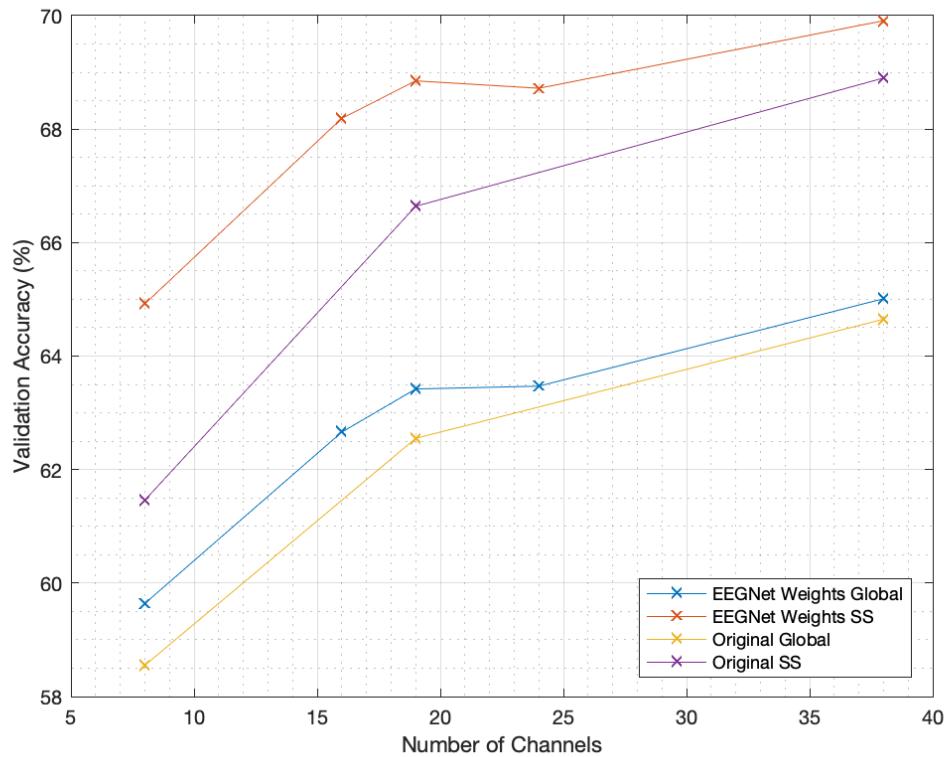


Figure 4.5.: Validation accuracy for inter-subject EEGNet channel selection (4-class, ds=1, T=3 s)

4. Results

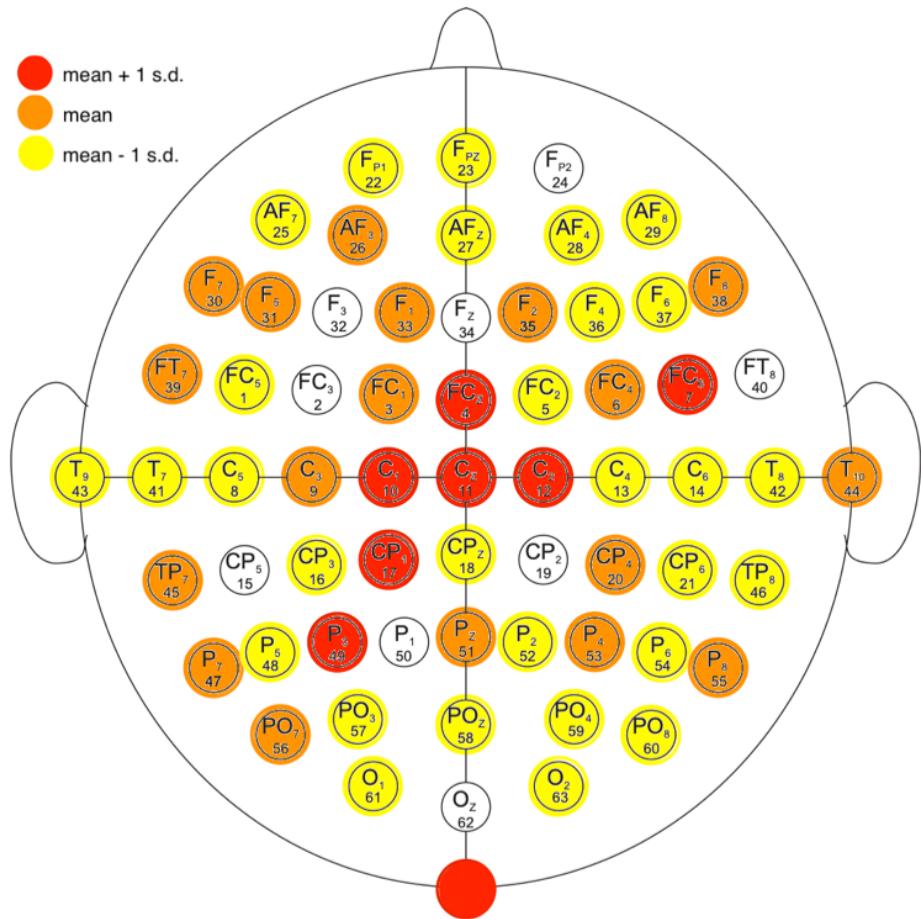


Figure 4.6.: Heatmap to visualise weights determining the selected channels for EEGNet channel selection, taken using the average of all 16 EEGNet filters used for inter-subject channel selection in model 0 (fold 1) of the 64-channel (ds=1, T=3s) global model. Red values are above the mean plus one standard deviation, orange values are above the mean, yellow values are above the mean minus one standard deviation, and white values are anything below.

4. Results

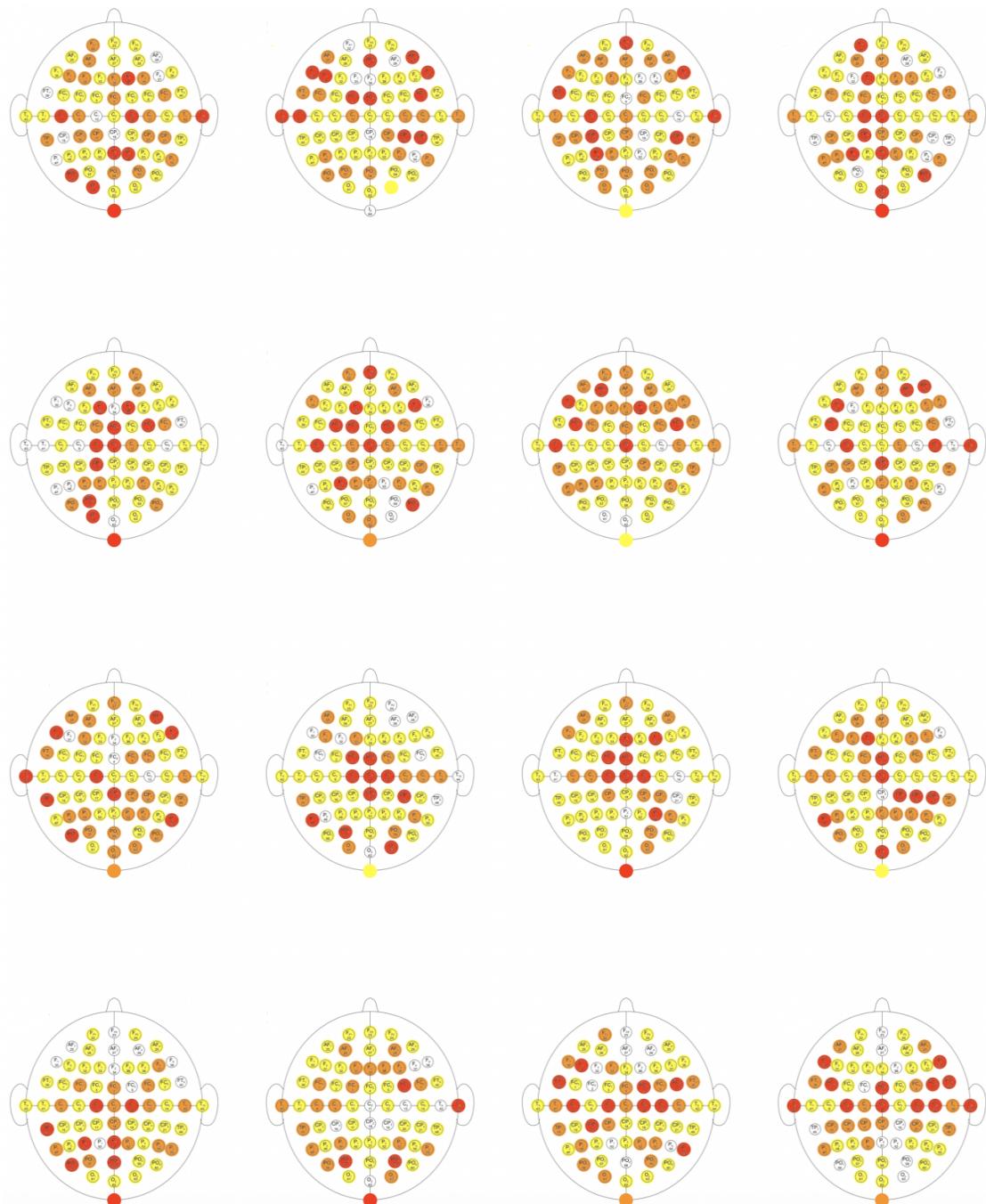


Figure 4.7.: Heatmaps taken for each individual filter of the 16 EEGNet filters.

4. Results

4.1.3. EEGNet Weights (Intra-Subject Classification)

The results of intra-subject channel selection using EEGNet weights are shown in Fig. 4.8, and compared to the the results for inter-subject channel selection using EEGNet weights.

In the global case, intra-subject channel selection achieves a higher validation accuracy, by 1.18% on average, in comparison to inter-subject channel selection. This makes sense, as in intra-subject channel selection, training is done on part of the data from all subjects, whereas inter-subject channel selection is only tested on data from subjects it is not trained on. In this way, the results for inter-subject channel selection is a good placeholder for how new data would perform, whereas intra-subject channel selection can be used to maximise the performance on existing data.

In the SS case, inter-subject channel selection obtains a better validation accuracy, by 1.19% on average, in comparison to intra-subject channel selection. The increase in accuracy obtained by SS-TL for the intra-subject case may not be as high as the increase for inter-subject as the validation accuracy for intra-subject channel selection is already fairly higher compared to inter-subject.

For more detailed results for both inter and intra-subject channel selection, refer to Appendix B.

4.2. Validation Accuracy vs Memory

One of the largest limiting factors of low-power BCIs is their memory requirement, which is not only limited by the SRAM, but also by the Flash memory for SS models. This work maximises the validation accuracies obtained while satisfying both SRAM and Flash requirements.

4.2.1. Validation Accuracy vs SRAM

A plot of the validation accuracy against SRAM for all possible configurations of channel selection, down-sampling and time windows can be found in Fig. 4.9. As can be observed, for the Cortex-M4 SRAM limit (shown by the first dotted black line), the new channel selection method produces several models that achieve a higher accuracy than the accuracy achieved by the model with the optimal configuration in the original channel selection (shown by the first black cross). The highest accuracy achieved using the new channel selection on the Cortex-M4 has $N_{ch} = 16$, $ds = 3$ and $T = 3\text{ s}$, achieving an accuracy of 63.45%, which is 0.94% higher in comparison to the original channel selection.

On the contrary, for the Cortex-M7 SRAM limit, the original channel selection remains to hold the highest validation accuracy. Comparing the original model (second black cross) to the corresponding configuration in the new channel selection (shown by the largest red square), the original model achieves 64.76%, which is 0.83% higher than the

4. Results

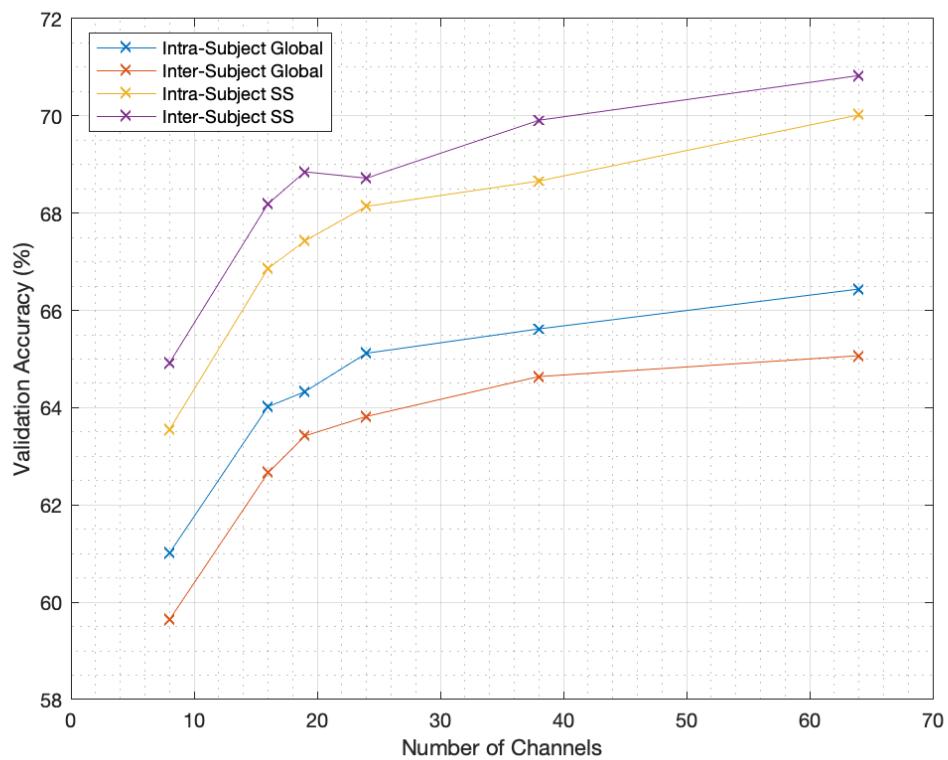


Figure 4.8.: Validation accuracy for intra-subject EEGNet channel selection (4-class, ds=1, T=3s)

4. Results

new configuration. The original model also achieves a 0.55% higher validation accuracy compared to the model with the highest validation accuracy for the new channel selection (excluding the model that lies on the edge of the memory limit) within the Cortex-M7 limit (with configuration $N_{ch} = 19$, $ds = 3$, $T = 3$ s).

Studying the work where the original channel selection was used [1], it is stated that for the Cortex-M7 core, which uses $ds = 3$, $N_{ch} = 38$ and $T = 2$ s, an accuracy of 64.76% is achieved with the original channel selection. However, it is also stated that using $ds = 1$, $N_{ch} = 38$ and $T = 3$ s, an accuracy of only 64.65% is achieved, which is lower. These results are correct and verified, so a further investigation into how downsampling and narrowing of the classification window affects the validation accuracy during channel selection may be beneficial to achieving a higher validation accuracy, especially for the Cortex-M7 case.

4.2.2. Validation Accuracy vs Flash Memory

The effect of layer freezing on the SS-TL validation accuracy for $N_{ch} = 64$, $N_{cl} = 4$ can be observed in Fig. 4.10. As expected, the highest validation accuracy is achieved when each subject has a fully unique set of parameters, where no layers are frozen. The second highest validation is achieved when only one layer is frozen, followed by three layers, and finally two layers. The increase in validation accuracy achieved by SS-TL (in comparison to the global model) are 5.75%, 3.08%, 1.38% and 2.06% for zero, one, two and three layers frozen, respectively.

The validations accuracies corresponding to layer freezing for different numbers of channels, and the amount of Flash memory SS-TL on the full set of 105 subjects (from the Physionet dataset) would occupy (for $ds = 1$ and $T = 3$ s) is found in Fig. 4.11. As shown, for all channel configurations, freezing one layer is sufficient to allow the full set of 105 subjects to fit on the Cortex-M4.

SS-TL was applied to the global model (with configuration $N_{ch} = 16$, $ds = 3$ and $T = 3$ s) that achieved the highest validation accuracy within the Cortex-M4 SRAM limit, achieving an accuracy of 67.89%, which is a 4.44% increase compared to the global model. For any $N_{ch} = 16$ configuration, the Flash memory is able to satisfy the entire 105 subject dataset without any layer freezing, as can be observed in Fig. 4.11. Therefore, there is no need for layer freezing for our optimum configuration. For larger datasets, however, layer freezing is a beneficial way to maximise the number of SS models stored while maintaining a high SS validation accuracy.

4. Results

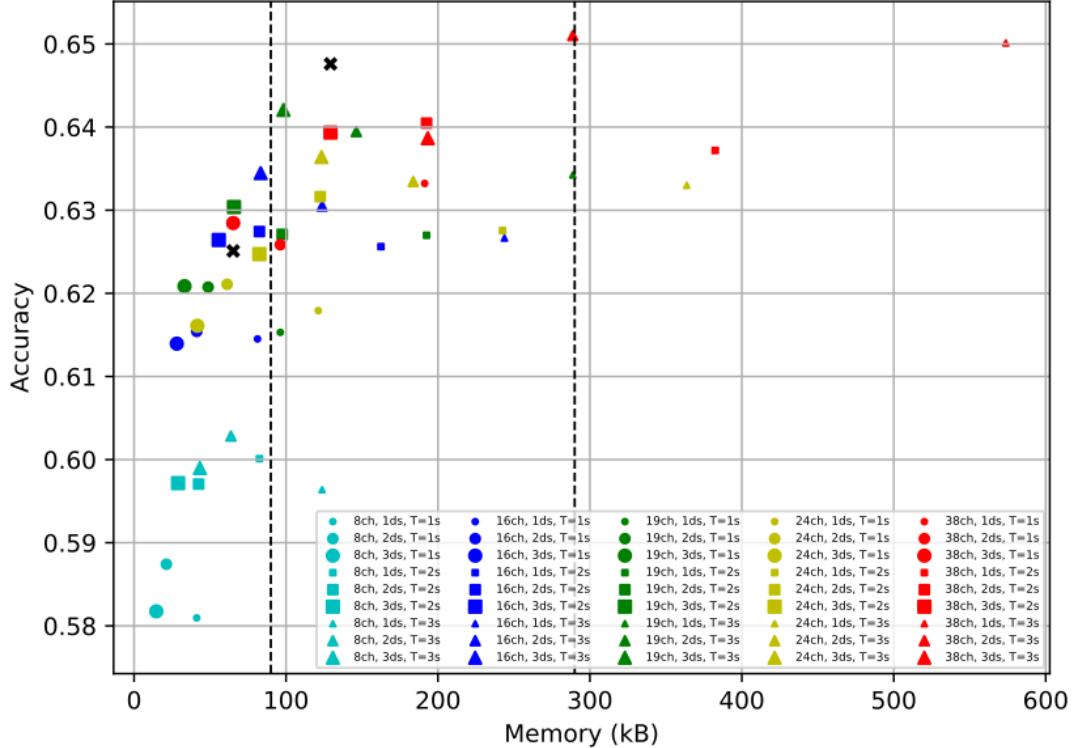


Figure 4.9.: Inter-subject validation accuracy vs SRAM required (4 class) - the two black crosses represent the models selected for the M4 and M7 processors in the original channel selection ($N_{ch} = 38$, $ds = 3$ and $T = 1$ s for M4 and $T = 2$ s for M7), and the two black dotted lines represent the RAM requirements for M4 and M7 from the original plot in [1]

4. Results

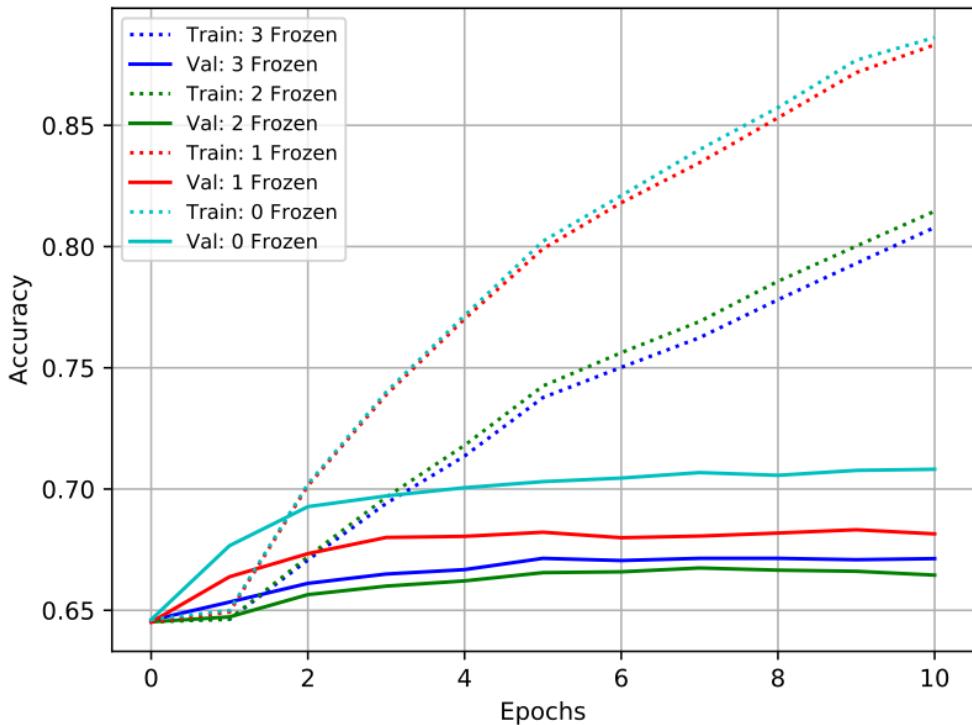


Figure 4.10.: Validation accuracy of SS-TL with layer freezing (inter-subject, $N_{ch} = 64$, $N_{cl} = 4$). Zero layers frozen means each subject has a set of completely unique parameters, one layer frozen corresponds to only $N_{\phi 1}$ being frozen, two layers frozen corresponds to both $N_{\phi 1}$ and $N_{\phi 2}$ being frozen and three layers frozen corresponds to $N_{\phi 1}$, $N_{\phi 2}$ and $N_{\phi 3}$ being frozen.

4. Results

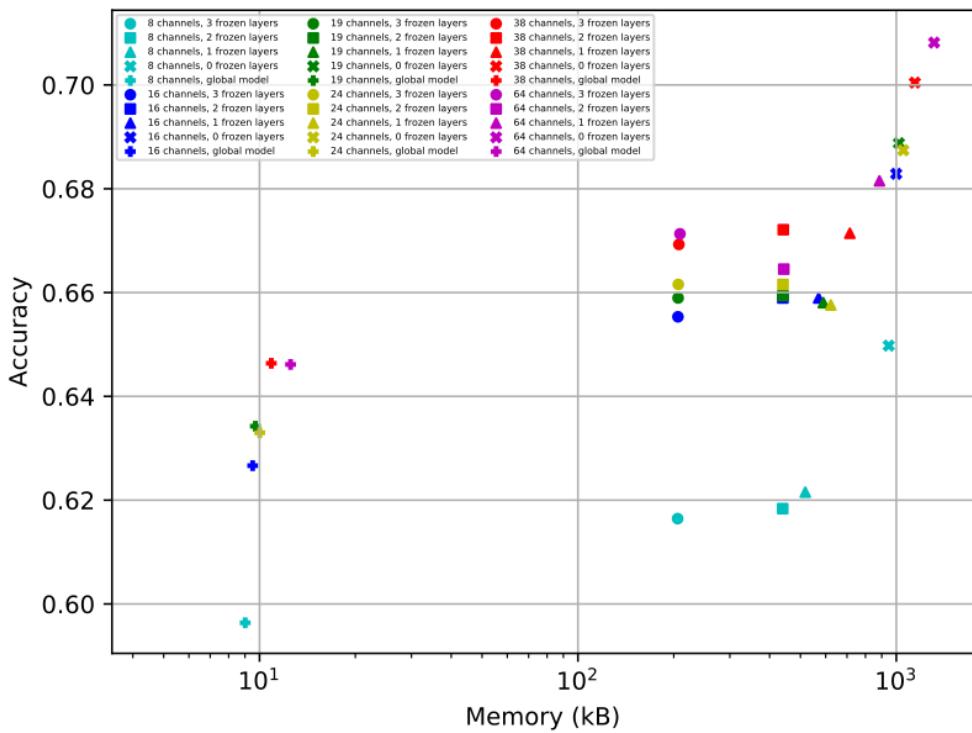


Figure 4.11.: Inter-subject Flash memory vs validation accuracy ($N_{subjects} = 105$, $N_{cl} = 4$, $ds = 1$, $T = 3$ s)

Chapter 5

Conclusion and Future Work

This work provides a reliable, systematic method of channel selection, which highlights channels that have the most significant MI neural activity automatically. Channel selection using EEGNet weights is able to improve validation accuracy by approximately 0.77% for global models and 2.23% for SS models on average in comparison to the original selection used. Using the new method, the global model with the optimum configuration (with $N_{ch} = 16$, $ds = 3$ and $T = 3$ s) that satisfies the memory requirements of the Cortex-M4 processor obtains a validation accuracy of 63.45%, surpassing the state-of-the-art [1] by 0.94%.

For channel selection using EEGNet weights, it would be beneficial to perform a more detailed study on how downsampling and narrowing of the classification time window can affect the validation accuracy during channel selection, as this may be the root of the issues presented in the Cortex-M7 configuration. Further work can also be done to minimise fluctuations and variations in the validation accuracy achieved for each memory reduction configuration.

Based on the results of this work, SS-TL can improve the validation accuracy by 5.28% on average for inter-subject models. For applications with a large number of subjects, the Flash memory is insufficient to contain fully unique models for each subject. This work introduces layer-freezing, that reduces the amount of Flash memory required to store a large number of SS models by freezing and reusing weights for some layers. This allows more SS models to be stored, while still allowing an increase in validation accuracy compared to using a global model by up to 3.08% when one layer is frozen. This could be investigated further to minimise the reduction in accuracy.

More investigations can also be conducted for CSP channel selection, by using specific filter weights for channel selection between specific classes, which may allow CSP channel selection validation accuracies to also surpass the original channel selection.

Appendix A

SS-TL Methods for Channel Selection

The SS-TL methods that were tested in order to obtain the highest accuracy for applications in channel selection are as follows:

1. 64-channels global model trained, select N channels based on EEGNet weights, train from scratch N channels global model, final epochs SS using N channel global model.
2. 64 channels global model trained, 64 channels final epoch SS, then select N channels based on SS EEGNet weights, train last epochs N channels SS by selecting and creating a new model with the N channels from the 64 channel global model.
3. 64 channels global model trained, 64 channels final epoch SS, then select N channels based on SS EEGNet weights, train from scratch N channels global model, then last epochs SS.
4. 64 channels global model trained, 64 channels final epoch SS, then select N channels based on SS EEGNet weights, train from scratch N channels for that subject.
5. 64 channels global model trained, 64 channels final epoch SS, then select N channels based on SS EEGNet weights, train final epochs N channels SS using the 64-ch SS model instead of global model.
6. 64 channels global model trained, 64 channels final epoch SS, then select N channels based on SS EEGNet weights, train final epochs N channels SS by setting unused channels in 64-ch global model to zero.

Appendix **B**

Channel Selection using EEGNet Weights Results

B. Channel Selection using EEGNet Weights Results

No. Channels	Original (from paper)	4-Class Accuracy (%) [n_ds = 1]		
		EEGNet Weights (T = 3s)	EEGNet Weights (T = 2s)	EEGNet Weights (T = 1s)
8	58.55	59.64	60.01	58.10
16	-	62.66	62.56	61.45
19	62.55	63.42	62.70	61.53
24	-	63.47	62.76	61.79
38	64.65	65.01	63.72	63.32

No. Channels	Original (from paper)	4-Class Accuracy (%) [n_ds = 2]		
		EEGNet Weights (T = 3s)	EEGNet Weights (T = 2s)	EEGNet Weights (T = 1s)
8	58.55	60.28	59.71	58.74
16	-	63.05	62.74	61.54
19	62.55	63.95	62.71	62.07
24	-	63.34	63.16	62.11
38	64.65	65.10	64.05	62.59

No. Channels	Original (from paper)	4-Class Accuracy (%) [n_ds = 3]		
		EEGNet Weights (T = 3s)	EEGNet Weights (T = 2s)	EEGNet Weights (T = 1s)
8	58.55	59.90	59.72	58.17
16	-	63.45	62.64	61.39
19	62.55	64.21	63.04	62.09
24	-	63.64	62.47	61.61
38	64.65	63.87	63.93	62.85

Figure B.1.: Inter-subject global validation accuracy for all memory reduction configurations (4 class).

No. Channels	4-Class Accuracy (%) [n_ds = 1, T = 3s]			
	Original - Global	Original - SS (v1)	EEGNet Weights - Global	EEGNet Weights – SS (v1)
8	58.12	61.46	59.64	64.92
16	-	-	62.66	68.19
19	62.64	66.64	63.42	68.85
24	-	-	63.82	68.72
38	64.42	68.90	64.64	69.91

Figure B.2.: Inter-subject subject-specific validation accuracies.

B. Channel Selection using EEGNet Weights Results

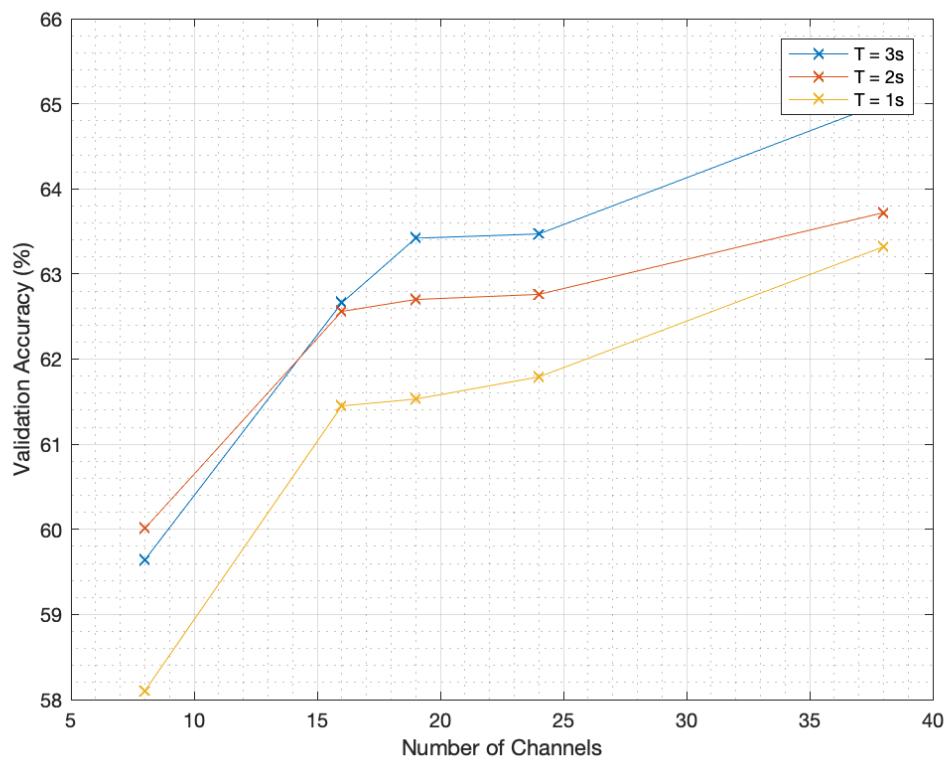


Figure B.3.: Inter-subject validation accuracy difference for different time windows ($N_{cl} = 4$, $ds = 1$)

B. Channel Selection using EEGNet Weights Results

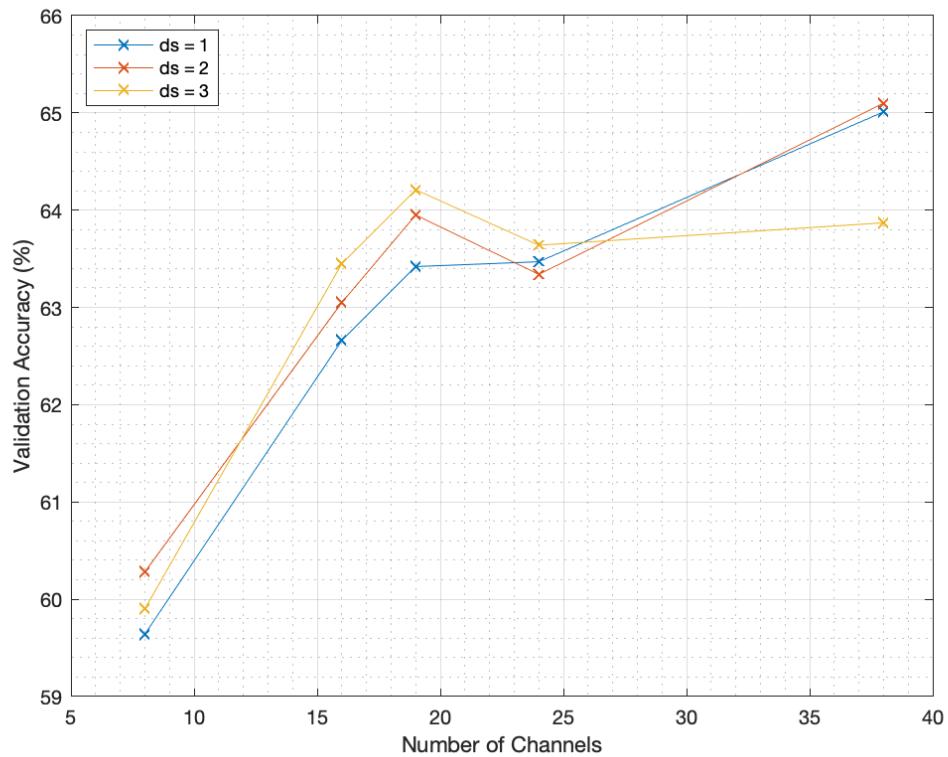


Figure B.4.: Inter-subject validation accuracy difference for different downsampling factors ($N_{cl} = 4$, $T = 3\text{ s}$)

B. Channel Selection using EEGNet Weights Results

No. Channels	4-Class Accuracy (%) [n_ds = 1]		
	EEGNet Weights (T = 3s)	EEGNet Weights (T = 2s)	EEGNet Weights (T = 1s)
8	61.01	60.37	58.81
16	64.02	62.49	62.96
19	64.33	63.08	62.91
24	65.12	62.89	62.62
38	65.62	64.75	63.84
64	66.44	64.82	64.31

No. Channels	4-Class Accuracy (%) [n_ds = 2]		
	EEGNet Weights (T = 3s)	EEGNet Weights (T = 2s)	EEGNet Weights (T = 1s)
8	60.34	59.92	59.48
16	63.55	63.40	62.05
19	63.97	63.80	62.76
24	64.61	64.50	63.23
38	65.76	64.91	63.88
64	65.82	65.43	64.23

No. Channels	4-Class Accuracy (%) [n_ds = 3]		
	EEGNet Weights (T = 3s)	EEGNet Weights (T = 2s)	EEGNet Weights (T = 1s)
8	60.10	59.60	59.71
16	63.22	63.05	61.92
19	63.32	64.04	62.17
24	64.35	63.83	62.71
38	65.26	63.99	63.56
64	65.43	65.83	64.23

Figure B.5.: Intra-subject global validation accuracy for all memory reduction configurations (4 class).

B. Channel Selection using EEGNet Weights Results

No. Channels	4-Class Accuracy (%) [n_ds = 1, T = 3s]	
	EEGNet Weights - Global	EEGNet Weights – SS (v1)
8	61.01	63.55
16	64.02	66.86
19	64.33	67.43
24	65.12	68.14
38	65.62	68.66
64	66.44	70.02

Figure B.6.: Intra-subject subject-specific validation accuracies.

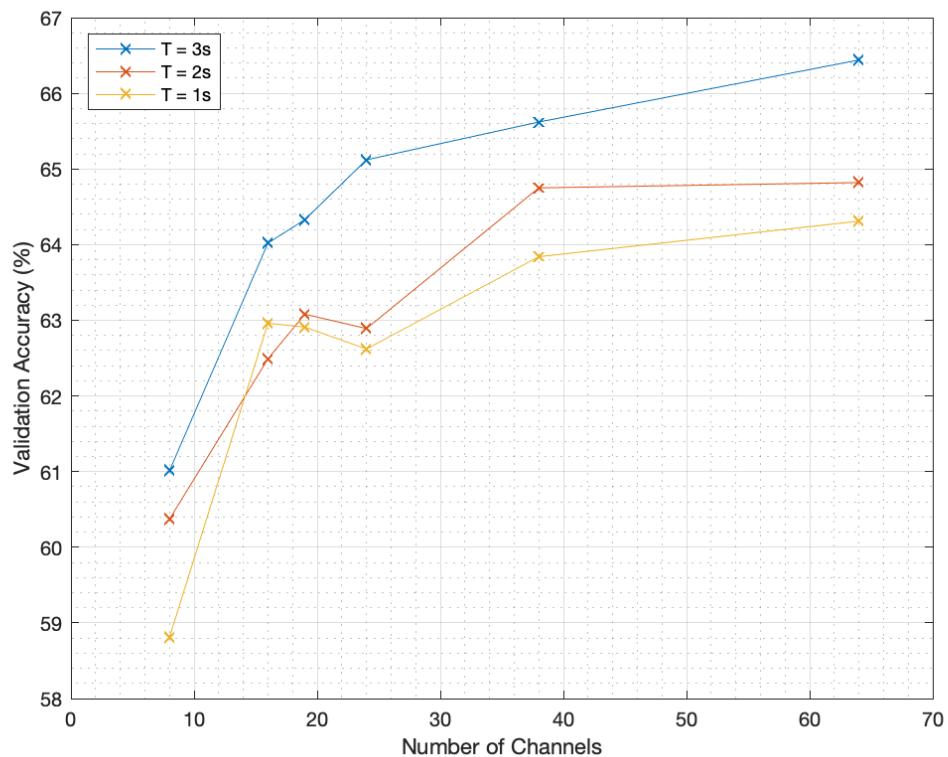


Figure B.7.: Intra-subject validation accuracy difference for different time windows ($N_{cl} = 4$, ds = 1)

B. Channel Selection using EEGNet Weights Results

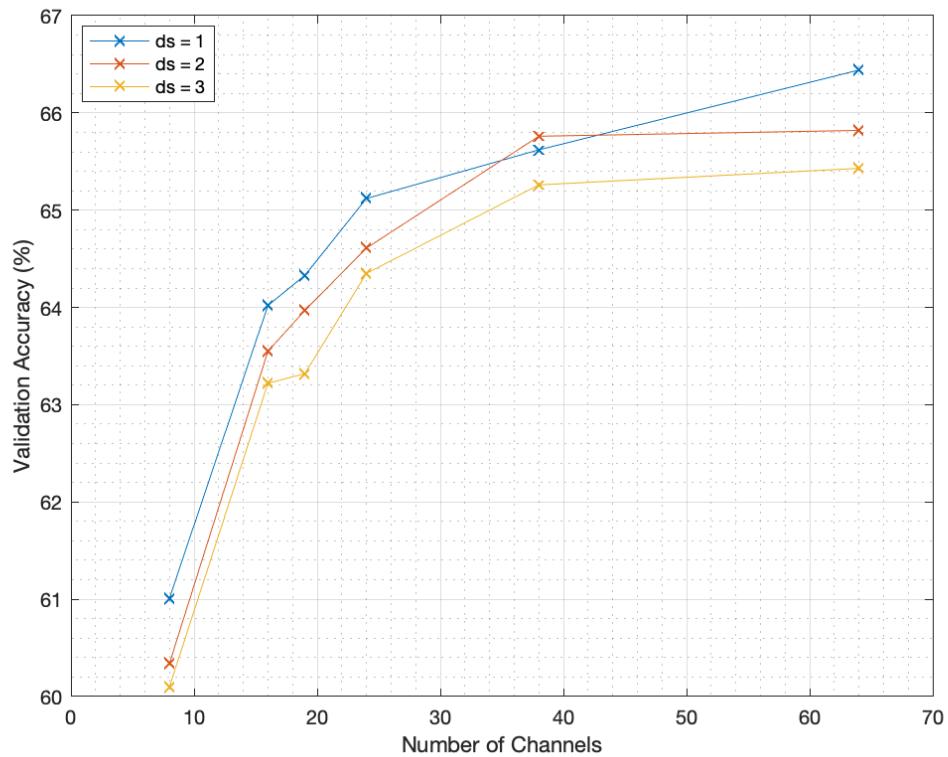


Figure B.8.: Intra-subject validation accuracy difference for different downsampling factors ($N_{cl} = 4$, $T = 3\text{ s}$)

List of Acronyms

BCI	Brain-Computer Interface
CNN	Convolutional Neural Network
CSP	Common Spatial Pattern
EEG	electroencephalogram
FBCSP	Frequency Band Common Spatial Pattern
IIS	Integrated Systems Laboratory
LDA	Linear Discriminant Analysis
MI	Motor-Imagery
MI-BCI	Motor-Imagery Brain-Computer Interface
SS	Subject-Specific
SS-TL	Subject-Specific Transfer Learning
SVM	Support Vector Machine

List of Figures

2.1.	EEGNet Architecture [1]: N_{ch} is the number of channels, N_s is the number of samples, N_p is the pooling length, N_{cl} is the number of classes and N_f is the filter size of the first temporal filter.	3
2.2.	Detailed description of EEGNet in MI classification copied from [1]. For visualisation, refer to Fig. 2.1. For each map, n is the number of filters , p is the padding strategy, k is the kernel size and s is the stride. Parameters refer to filter weights and output shape gives the feature map size.	4
2.3.	64-Electrode Configuration [3]	5
2.4.	Trial Paradigm of Physionet EEG Motor Movement/Imagery Dataset	6
3.1.	Inter-Subject Global Classifier Paradigm	8
3.2.	Inter-Subject SS-TL Classifier Paradigm, for 1 Subject	9
3.3.	Intra-Subject Global and SS-TL Classifier Paradigm	10
4.1.	Comparison of inter-subject validation accuracies obtained using channel selection with multiscale CSP, single time window and frequency band CSP, CSP-rank, and the original channel selection (classification with EEGNet) for 4-class classification (ds=1, T=3 s).	14
4.2.	Comparison of inter-subject validation accuracies obtained with CSP-rank and the original channel selection for 2-class classification (ds=1, T=3 s) .	15
4.3.	Heatmap to visualise weights determining the selected channels for single time window and frequency band CSP, taken using the average of all 12 CSP filter weights used for 4-class CSP channel selection. Red values are above the mean plus one standard deviation, orange values are above the mean, and yellow values are above the mean minus one standard deviation.	17
4.4.	Heatmaps for each individual filter in the 12 CSP filters used for 4-class classification. From left to right, top to bottom, the filters are associated with maximising the average variance between {(L,R) (L,0) (L,F) (R,0) (R,F) (0,F)} classes.	18
4.5.	Validation accuracy for inter-subject EEGNet channel selection (4-class, ds=1, T=3 s)	19

List of Figures

4.6.	Heatmap to visualise weights determining the selected channels for EEG-Net channel selection, taken using the average of all 16 EEGNet filters used for inter-subject channel selection in model 0 (fold 1) of the 64-channel ($ds=1$, $T=3s$) global model. Red values are above the mean plus one standard deviation, orange values are above the mean, yellow values are above the mean minus one standard deviation, and white values are anything below.	20
4.7.	Heatmaps taken for each individual filter of the 16 EEGNet filters.	21
4.8.	Validation accuracy for intra-subject EEGNet channel selection (4-class, $ds=1$, $T=3s$)	23
4.9.	Inter-subject validation accuracy vs SRAM required (4 class) - the two black crosses represent the models selected for the M4 and M7 processors in the original channel selection ($N_{ch} = 38$, $ds = 3$ and $T = 1s$ for M4 and $T = 2s$ for M7), and the two black dotted lines represent the RAM requirements for M4 and M7 from the original plot in [1]	25
4.10.	Validation accuracy of SS-TL with layer freezing (inter-subject, $N_{ch} = 64$, $N_{cl} = 4$). Zero layers frozen means each subject has a set of completely unique parameters, one layer frozen corresponds to only $N_{\phi 1}$ being frozen, two layers frozen corresponds to both $N_{\phi 1}$ and $N_{\phi 2}$ being frozen and three layers frozen corresponds to $N_{\phi 1}$, $N_{\phi 2}$ and $N_{\phi 3}$ being frozen.	26
4.11.	Inter-subject Flash memory vs validation accuracy ($N_{subjects} = 105$, $N_{cl} = 4$, $ds = 1$, $T = 3s$)	27
B.		
B.1.	Inter-subject global validation accuracy for all memory reduction configurations (4 class).	31
B.2.	Inter-subject subject-specific validation accuracies.	31
B.3.	Inter-subject validation accuracy difference for different time windows ($N_{cl} = 4$, $ds = 1$)	32
B.4.	Inter-subject validation accuracy difference for different downsampling factors ($N_{cl} = 4$, $T = 3s$)	33
B.5.	Intra-subject global validation accuracy for all memory reduction configurations (4 class).	34
B.6.	Intra-subject subject-specific validation accuracies.	35
B.7.	Intra-subject validation accuracy difference for different time windows ($N_{cl} = 4$, $ds = 1$)	35
B.8.	Intra-subject validation accuracy difference for different downsampling factors ($N_{cl} = 4$, $T = 3s$)	36

Bibliography

- [1] X. Wang, M. Hersche, B. Tömekce, B. Kaya, M. Magno, and L. Benini, “An accurate eegnet-based motor-imagery brain-computer interface for low-power edge computing,” 2020.
- [2] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, “Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces,” *Journal of Neural Engineering*, vol. 15, no. 5, p. 056013, Jul 2018. [Online]. Available: <http://dx.doi.org/10.1088/1741-2552/aace8c>
- [3] A. Goldberger, L. Amaral, L. Glass, J. Hausdorff, P. Ivanov, R. Mark, J. Mietus, G. Moody, C. Peng, and H. Stanley, “Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals. circulation [online].” *Circulation*, vol. 101, no. 23, p. 215–220, 2000.
- [4] T. Mulder, “Motor imagery and action observation: cognitive tools for rehabilitation,” *Journal of Neural Engineering*, vol. 114, no. 10, pp. 1265–1278, 2007.
- [5] Y. Yu, Z. Zhou, Y. Liu, J. Jiang, E. Yin, N. Zhang, Z. Wang, Y. Liu, X. Wu, and D. Hu, “Self-paced operation of a wheelchair based on a hybrid brain-computer interface combining motor imagery and p300 potential,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 12, pp. 2516–2526, 2017.
- [6] R. Abbasi-Asl, M. Keshavarzi, and D. Y. Chan, “Brain-computer interface in virtual reality,” in *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*, 2019, pp. 1220–1224.
- [7] A. Chiuzbaian, J. Jakobsen, and S. Puthusserypady, “Mind controlled drone: An innovative multiclass ssvep based brain computer interface,” in *2019 7th International Winter Conference on Brain-Computer Interface (BCI)*, 2019, pp. 1–5.
- [8] Kai Keng Ang, Zheng Yang Chin, Haihong Zhang, and Cuntai Guan, “Filter bank common spatial pattern (fbcsp) in brain-computer interface,” in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 2008, pp. 2390–2397.

Bibliography

- [9] F. Yger, M. Berar, and F. Lotte, "Riemannian approaches in brain-computer interfaces: A review," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 10, pp. 1753–1762, 2017.
- [10] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, "Bci2000: a general-purpose brain-computer interface (bci) system," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 1034–1043, 2004.
- [11] B. Tömekçe and B. Alp Kaya, "Embedded machine learning on eeg data analysis," 2020.
- [12] M. Hersche, T. Rellstab, P. D. Schiavone, L. Cavigelli, L. Benini, and A. Rahimi, "Fast and accurate multiclass inference for mi-bcbs using large multiscale temporal and spectral features," *2018 26th European Signal Processing Conference (EUSIPCO)*, Sep 2018. [Online]. Available: <http://dx.doi.org/10.23919/EUSIPCO.2018.8553378>
- [13] W. Tam, Z. Ke, and K. Tong, "Performance of common spatial pattern under a smaller set of eeg electrodes in brain-computer interface on chronic stroke patients: A multi-session dataset study," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2011, pp. 6344–6347.