SpotiBCI: A Motor Imagery BCI for Music Player Control Using Deep Learning

Arthur Boschet and Kenji Marshall

Department of Bioengineering, McGill University

BIEN 462: Engineering Principles in Physiological Systems

April 26, 2021

Introduction

Brain computer interfaces (BCIs) are any technology capable of acquiring and analyzing brain signals such as electroencephalography (EEG) in order to effectuate external action [1]**.** This technology has found applications in neuromarketing, gaming, and more, but has been profoundly impactful as a communication and interaction interface for patients suffering from physical disabilities such as quadriplegia [2]. In particular, BCIs restore autonomy, allowing these individuals to communicate with those around them or interact with technology such as computers and phones. For instance, BCIs have been used to control spellers, robotic hands, and motorized wheelchairs [3], [4]. These solutions are extremely valuable and have the potential to impact a large fraction of the population; for instance, in 2013, it was found that 1.7% of Americans (around 5 million individuals) were living with some level of paralysis [5]. In particular, one rare but debilitating form of paralysis is termed locked-in syndrome (LIS), and features complete loss of physical control while maintaining normal intelligence [6]. BCIs are the only viable solution that can allow these patients to engage with the world around them.

Our project aims to expand the existing BCI repertoire by developing a brain-controlled music player called SpotiBCI. This problem has been approached before, however previous implementations have relied on synchronous control and non-intuitive interaction. For instance, one group developed a BCI-controlled music player based on steady-state visually evoked potentials (SSVEP) wherein the user is presented with an array of tiles flashing at different frequencies [7]. By fixating on a particular tile, electrical oscillations at that tile's frequency and its harmonics are amplified at the occipital and parietal electrodes. This phenomena can be leveraged for BCI control. However, this modality relies on signals generated as a result of the interface interaction, not by the user, asynchronously. Moreover, this paradigm is tiring and non-intuitive to use -- mental fatigue and cognitive load limit SSVEP BCIs' capacity for long-term, continuous use [8]. Instead, we aim to develop a more intuitive and entirely asynchronous BCI largely controlled by motor imagery (MI), while also using jaw clenches as a control signal.

MI has been studied extensively, and has been found to produce an acute reduction in the power of alpha and upper beta oscillations in the motor cortex. This is termed an event-related desynchronization (ERD) [9]. As such, modeling of motor imagery typically uses features derived from the signal power spectral density (PSD). For instance, recent papers have applied convolutional neural network (CNN) approaches by reducing MI trials into time-frequency decomposition images [10]–[13]. A similar approach is featured in this project by using the short-time Fourier transform STFT to generate images. Finally, jaw clenches manifest as bursts of oscillatory behavior, and have been characterized by a broadband increase in signal power [14]. As such, this project aims to model jaw clenches using theta, alpha, and beta bandpower

features. Due to the small number of features, a simple classification model is sufficient and thus a logistic regression scheme is employed.

Methods

In order to model jaw clenches, a dataset was recorded using the OpenBCI Cyton biosensing kit on one of the project members. The electrodes were set up over the C3, Cz, and C4 locations in the standard 10-20 system, spanning the left and right motor cortices. 100 jaw clenches were recorded alongside 5-minutes of baseline data. This was segmented using a 2-second window to generate a balanced dataset of 200 examples. In order to extract features from these segments, the Welch method for PSD computation was applied. This involves splitting up a given time series into shorter, possible overlapping segments, and averaging periodogram computations to reduce noise.

As we've seen in class, the PSD is defined as the discrete time Fourier transform (DTFT) of a signal's autocorrelation function. For ergodic random signals, the area under the PSD is directly related to the expected time-averaged power of a realization of the signal. This makes it a useful analysis tool for the spectral distribution of power within a signal. This property can be seen by relating the signal power property of autocorrelation to the inverse DTFT:

$$\phi_{xx}[0] = E\{X^2\} = \frac{1}{2\pi}\int_{-\pi}^{\pi} \Phi_{xx}(e^{jw})dw$$

The PSD can be estimated by taking the square of the signal's discrete Fourier transform (DFT) at each frequency. As a reminder, the DFT is given by:

$$X[k] = \sum_{i=1}^{N-1} x[n]e^{-j\frac{2\pi kn}{N}}$$

Of great importance is N, the sample length, as this defines the frequency resolution of the DFT and thus the PSD. In Hz, this frequency resolution is given by: $\Delta f = \frac{f_s}{N}$. In order to distinguish frequency bands, a frequency resolution of 1 Hz is sufficient. Since the OpenBCI Cyton samples data at 250 Hz, a segment length of 250 samples was used in the Welch PSD. An overlap of 100 samples was used to provide sufficient smoothing, and a Hamming window was used to mitigate finite-sample effects. From the PSD, the average bandpower in the theta, alpha, and beta bands was computed and then averaged across the 3 electrodes, resulting in 3 features per segment. The resulting dataset was modeled using logistic regression -- a technical description of logistic regression is provided at the end of this methods section.

MI modeling was performed using public dataset 2b from the 4th Berlin BCI Competition. This dataset features left- and right-hand motor imagery trials from 9 subjects, using 3 electrodes at C3, Cz, and C4 (as in the jaw clench dataset), with a sampling rate of 250 Hz. 400 labelled trials, equally split between left and right are provided per subject, resulting in a balanced dataset of 3600 examples overall. The trials are 2 seconds in duration. Features were extracted using the STFT, which attempts to localize dynamics in both time and frequency by sliding a window across a signal and stacking local FFT computations to form an image. This process is captured by the following equation, parametrized by both time (controlling window location) and frequency:

$$\mathbf{STFT}\{x(t)\}(\tau,\omega) \equiv X(\tau,\omega) = \int_{-\infty}^{\infty} x(t)w(t-\tau)e^{-i\omega t}\,dt$$

The important STFT parameters are similar to PSD parameters, namely sample length and overlap. These were chosen in accordance with Dai et al wherein, on the same dataset, a length of 64 samples and an overlap of 50 samples was used. Also, a Hamming window was applied to the samples and then they were zero-padded to 512 points before local FFT computation [15]. Although using 512 samples gives a frequency resolution of about 0.49 Hz, this is effectively an interpolation of the resolution from the true sample length of 64, which is about 3.91 Hz. This poorly resolves the alpha band (8-13 Hz), and thus further experimentation with the STFT parameters would be a useful extension on this project and will be discussed later. Using this scheme, 31 frequencies were extracted from the upper theta and alpha band (6 - 13 Hz) as well as the upper beta band (17 - 30 Hz) across 32 times for each electrode. By vertically stacking each electrode's data, this gives a feature image of 93 x 32. An example is shown in Figure 1. Modeling MI was done using a CNN -- a technical introduction to CNNs is provided shortly

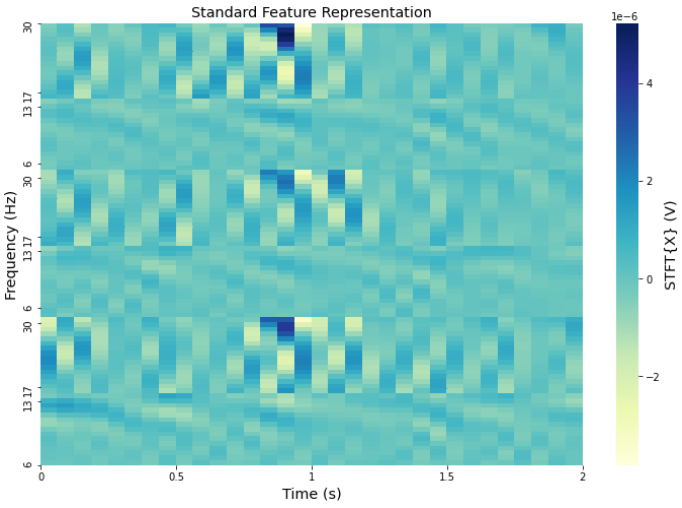

Figure 1: Example feature representation from STFT application

Finally, a Python script was written to integrate the Spotify API with jaw clench and MI models using a real-time data stream from the OpenBCI Cyton (via lab streaming layer). This provides a functional BCI to any user that has access to electrodes.

As previously mentioned, machine learning approaches were used to perform binary classification of jaw clench and left-/right-hand MI. Both logistic regression and CNNs model the probability of a positive outcome using the sigmoid activation function.

$$P(y=1) = \hat{y}_n = \sigma(-w^T x_n) = \frac{1}{1+e^{-w^T x_n}}$$

By using this assumption, the probabilities of both positive and negative outcomes can be expressed using the following simple equation. This formula represents the probability that the classification outputted by the chosen model is correct.

$$P(y_n) = \hat{y}_n{}^{y_n}(1-\hat{y}_n)^{1-y_n}$$

The objective when training a binary classification algorithm is to maximize this probability over all instances contained in the training dataset. It can be assumed that all instances in the dataset are independent from one another and thus the over probability can be expressed as the product of the individual probabilities associated with each instance. Maximum likelihood estimation can be used to find the optimal parameters in our model. The product can be converted to a sum by taking the logarithm of the likelihood function which gives the log-likelihood.

$$argmax_w\, P(y_1, y_2, y_3, \dots, y_n, \dots, y_{N-1}, y_N) = argmax_w \prod_{n=1}^{N} P(y_n)$$

$$\Longrightarrow argmax_w \log\left\{\prod_{n=1}^{N} P(y_n)\right\} = argmax_w \sum_{n=1}^{N} \log\{P(y_n)\}$$

$$\Longrightarrow argmax_w \sum_{n=1}^{N} \log\{\hat{y}_n{}^{y_n}(1-\hat{y}_n)^{y_n}\} = argmax_w \sum_{n=1}^{N} y_n \log\{\hat{y}_n\} + (1-y_n)\log\{1-\hat{y}_n\}$$

$$\Longrightarrow argmax_w \sum_{n=1}^{N} y_n \log\left\{\frac{1}{1+e^{-w^T x_n}}\right\} + (1-y_n)\log\left\{1-\frac{1}{1+e^{-w^T x_n}}\right\}$$

$$\Longrightarrow argmax_w \sum_{n=1}^{N} y_n \log\left\{\frac{1}{1+e^{-w^T x_n}}\right\} + (1-y_n)\log\left\{\frac{e^{-w^T x_n}}{1+e^{-w^T x_n}}\right\}$$

$$\Longrightarrow argmax_w \sum_{n=1}^{N} y_n \log\left\{\frac{1}{1+e^{-w^T x_n}}\right\} + (1-y_n)\log\left\{\frac{1}{1+e^{w^T x_n}}\right\}$$

$$\Longrightarrow argmin_w \sum_{n=1}^{N} y_n \log\{1+e^{-w^T x_n}\} + (1-y_n)\log\{1+e^{w^T x_n}\}$$

This simple analysis gives the cross-entropy cost function that we use for both our logistic regression model and the convolutional neural network.

$$J(w) = \sum_{n=1}^{N} y_n \log\{1 + e^{-w^T x_n}\} + (1 - y_n)\log\{1 + e^{w^T x_n}\}$$

The cross entropy can easily be shown to be a convex function using the theorem that states that a function mapping from an n dimensional vector to the reals is convex if and only if its hessian matrix is positive semi-definite [16].

$$\nabla^2 J(w) = \sum_{n=1}^{N} \left(\frac{1}{1 + e^{-w^T x_n}}\right)\left(1 - \frac{1}{1 + e^{-w^T x_n}}\right) x_n \cdot {x_n}^T$$

Let,

$$M = \left(\frac{1}{1+e^{-w^T x_n}}\right)\left(1 - \frac{1}{1+e^{-w^T x_n}}\right)$$

We can then easily show that the hessian of the cross-entropy loss is indeed positive semi-definite.

$$z^T \nabla^2 J(w) z = z^T \left(\sum_{n=1}^{N} M\, x_n \cdot {x_n}^T\right) z = \sum_{n=1}^{N} M\, z^T (x_n \cdot {x_n}^T) z = \sum_{n=1}^{N} M\, (z^T x_n)^2 \geq 0$$

CNNs were the model of choice for left-/right-hand MI classification because they tend to perform extremely well on image data. The reason for their high performance is their ability to learn spatial features in the image regardless of their location within the image [17]. In essence, CNNs reduce the overall hypothesis space of a fully connected neural network with the hypothesis that local patterns are very important to understand the image structure. This reduction in the overall hypothesis space with this added hypothesis allows CNNs to be less prone to overfitting on image data than conventional neural networks [18]. Our chosen model uses a single convolutional layer with filters of dimensions (93,3) applied to the frequency-time feature representation because we are interested in the presence of frequency patterns regardless of when they occur in time [15]. Although CNNs can contain traditional dense layers and pooling layers which reduce the number of parameters, their main feature are convolutional layers which compute the cross-correlation between a set of filters and the previous layer output as shown in the following equation.

$$\left(F^{[l]}{}_n * A^{[l-1]}\right)[x, y] = \sum_{a=1}^{f_1} \sum_{b=1}^{f_2} \sum_{c=1}^{f_3} F^{[l]}{}_n[a, b, c] \times A^{[l-1]}[sx + a, sy + b, c]$$

As shown in Figure 2, this computation can be performed with a chosen number of filters and then a bias matrix is added to the computation before being passed to a non-linear activation unit. The presence of the non-linear activation in both dense and convolutional layers is crucial since

the model can otherwise act as a linear combination of the inputs with the parameters and reduces to logistic regression, making it unable to to learn non-linear decision boundaries. The specific architecture of the CNN is problem-specific and has to be tuned via cross-validation. For our purposes we used the TensorFlow Python programming framework to create and evaluate different CNN architectures. The strength of this framework is its ability to create very customizable deep-learning architectures and automatically create computational graphs which enable the users to create diverse neural network architectures without explicitly programming the backpropagation algorithm [19].

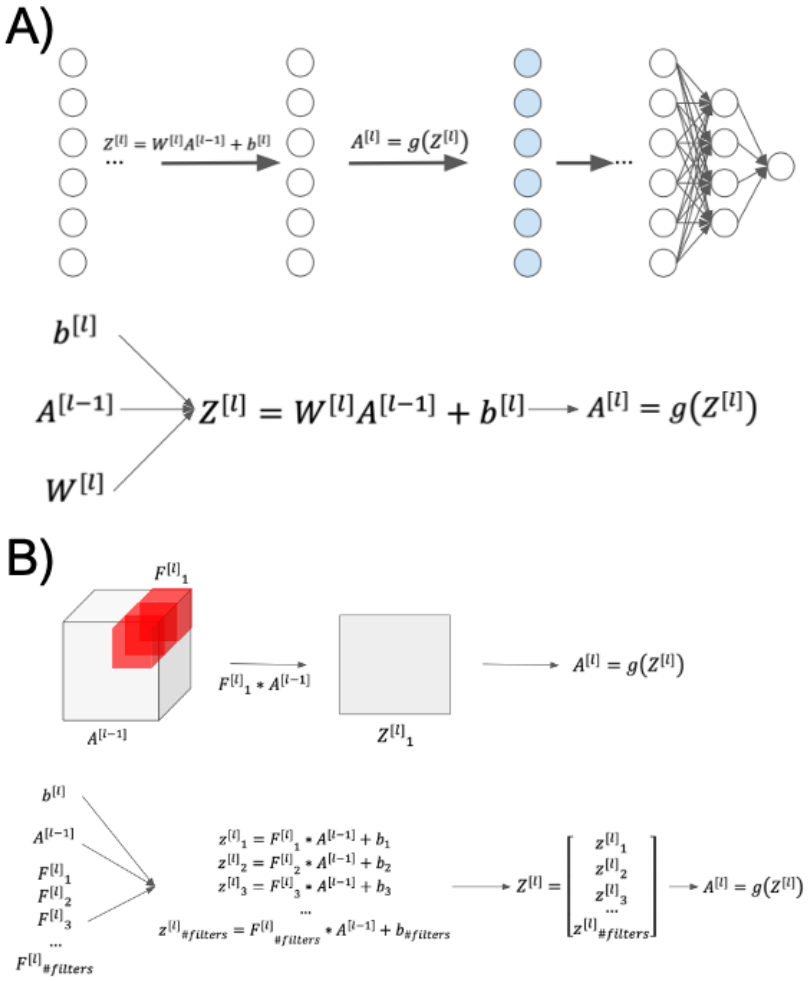

Figure 2: A) Computational graph of dense layers in neural networks. B) computational graph in convolutional layers.

Using a computational graph, TensorFlow uses the multivariate chain rule to compute the gradients of each trainable parameter and backpropagate the gradients throughout the layers of the neural network.

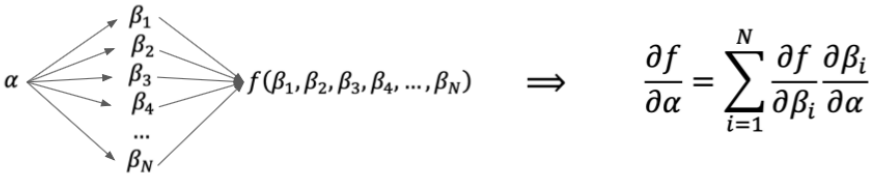

Figure 3: Gradient computation from a computational graph using the chain rule

Results

As expected, jaw clench segments were found to have a greater average theta, alpha, and beta bandpower. In particular, power increased from 0.75 ± 0.40, 0.35 ± 0.23, and 0.13 ± 0.08, to 2.61 ± 2.82, 2.88 ± 1.99, and 4.32 ± 2.63 $\mu V^2/Hz$ for the theta, alpha, and beta bands respectively, averaged across C3, Cz, and C4 (Figure 4a).

Although this was a simple classification problem, the effect of regularization was explored in order to apply course content. As seen in class, regularization modifies a model cost function by adding a penalty for large weights. This penalty can be associated with the L2 norm, which is arises from a maximum a posteriori (MAP) estimation of model parameters with a zero-mean Gaussian prior, or the L1 norm, which can be derived from a MAP estimation with a zero-mean Laplacian prior. In this case, an elastic net regularization scheme was applied to the logistic regression model. Elastic net blends L1 and L2 regularization using a mixing parameter αvia the following scheme:

$$J(\mathbf{w}) = \sum_{i=1}^{N} Cost(\mathbf{w}, \mathbf{x}^{(i)}) + \lambda \left((\frac{1-\alpha}{2}) * ||\mathbf{w}||_2^2 + \alpha * ||\mathbf{w}||_1 \right)$$

In Scikit-Learn, the logistic regression class uses the parameter C, which is the inverse of regularization strength λ.The resulting ten-fold cross-validation accuracies for different values of C and α are shown in Figure 4b.

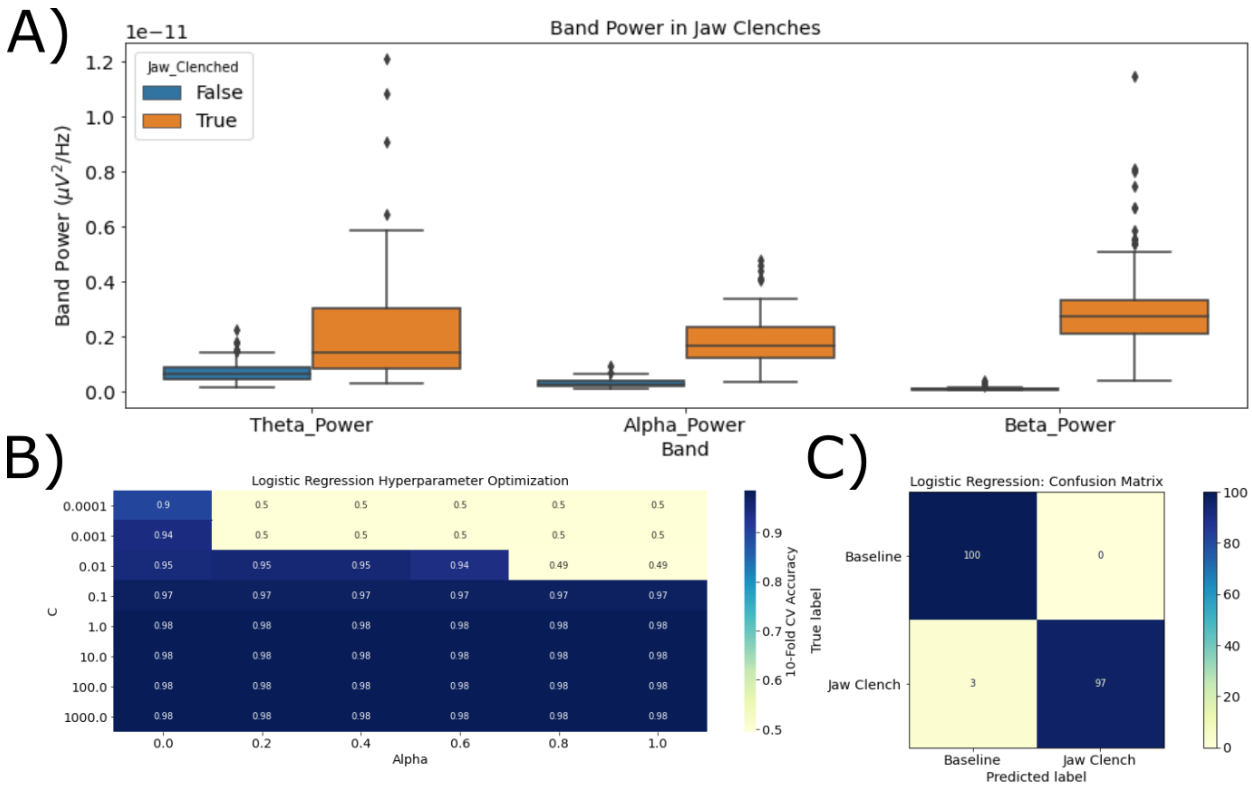

Figure 4: Results of logistic regression jaw clench modeling. (A) Theta, alpha, and beta average bandpowers increase during jaw clench. (B) 10-fold cross-validation accuracy for varying elastic net regularization parameters. (C) Confusion matrix of optimized logistic regression model which has a 0 false positive rate, 98.50 accuracy, and a 98.48 F-score

As expected, extremely large regularization strength constrains model flexibility, increasing model bias and reducing test accuracy. This effect is more strongly observed with higher levels of L1 regularization. This may be explained by the tendency of L1 regularization to induce model sparsity by reducing weights to zero, as very high L1 regularization may reduce most features to 0, resulting in the observed chance-level accuracy. With very weak regularization strength, there is no apparent penalty and validation accuracies remain consistent. This makes sense as this dataset only uses 3 features (and thus model complexity is inherently limited), and is almost completely linearly separable; thus, there is minimal risk of overfitting.

The optimal model was found to use $C = 1.0$; $\alpha = 0.0$, and provided a cross-validation accuracy of 98.50%, with an F-score of 98.48. A confusion matrix is shown in Figure 4c. Importantly, it can be observed that the false positive rate is 0. This is a useful property to ensure robustness of jaw clenches as a control signal.

Before modeling MI, the presence of the ERD effect in the public dataset was first verified. To do so, a similarly parametrized STFT was taken and averaged across all trials for each subject. The results for right-hand MI of an ideal subject is shown in Figure 5a. A strong ERD response

is observed over C3, at the left motor cortex, and less intensely at Cz. This contralateral pattern is typical of MI due to the contralateral organization of motor control [20]. At C4, an event-related synchronization (ERS) is observed; this has been associated with certain MI tasks before as well, and also provides useful discriminative information [21]. Conversely, the subject in figure 5b displays minimal structure across all electrodes during MI. This result is expected; MI patterns are produced to different extents by everyone. In fact, variance in MI proficiency is a major obstacle to developing generalizable MI-BCIs, and this dataset is no exception [22]. This point will be explored in more depth later on.

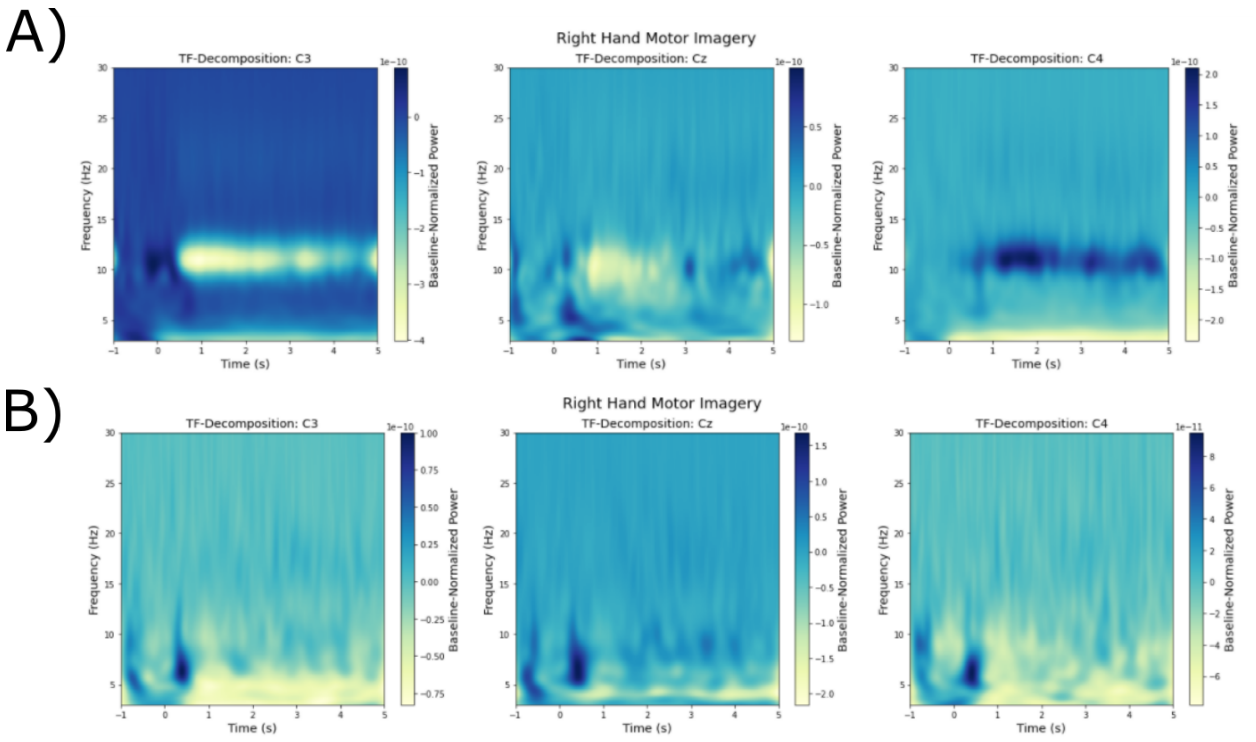

Figure 5: (A) This ideal subject displays distinguishable contralateral ERD patterns in this averaged time frequency. (B) This subject fails to present any identifiable structure; this shows the variance in MI ability between individuals

The performance of the CNN in the MI classification task as a function of the total number of parameters was evaluated as shown in Figure 6. With Figure 6a and 6b, we can notice that regardless of the number of dense layers in our model, the 10-fold cross validation peaked at around 65% with around 4000 parameters before plateauing while the training accuracy continues to increase as the number of parameters increase. Furthermore, we can notice that increasing the number of dense layers had a limited impact on the cross-validation accuracy and in fact seemed to decrease it. On the other hand, the training accuracy significantly increased from 75% to 89% as the number of 10-unit dense layers increased from 0 to 5. This indicates that increasing the number of dense layers led to overfitting and therefore we chose to have no dense

layers. In Figure 6c and 6d, we look more closely at the model with 0 dense layers and plot both the average 10-fold cross-validation and training accuracy along with the standard deviation. We can see that the accuracy peaks 15 filters which corresponds to around 4000 parameters considering our chosen filter dimension of (93,3).

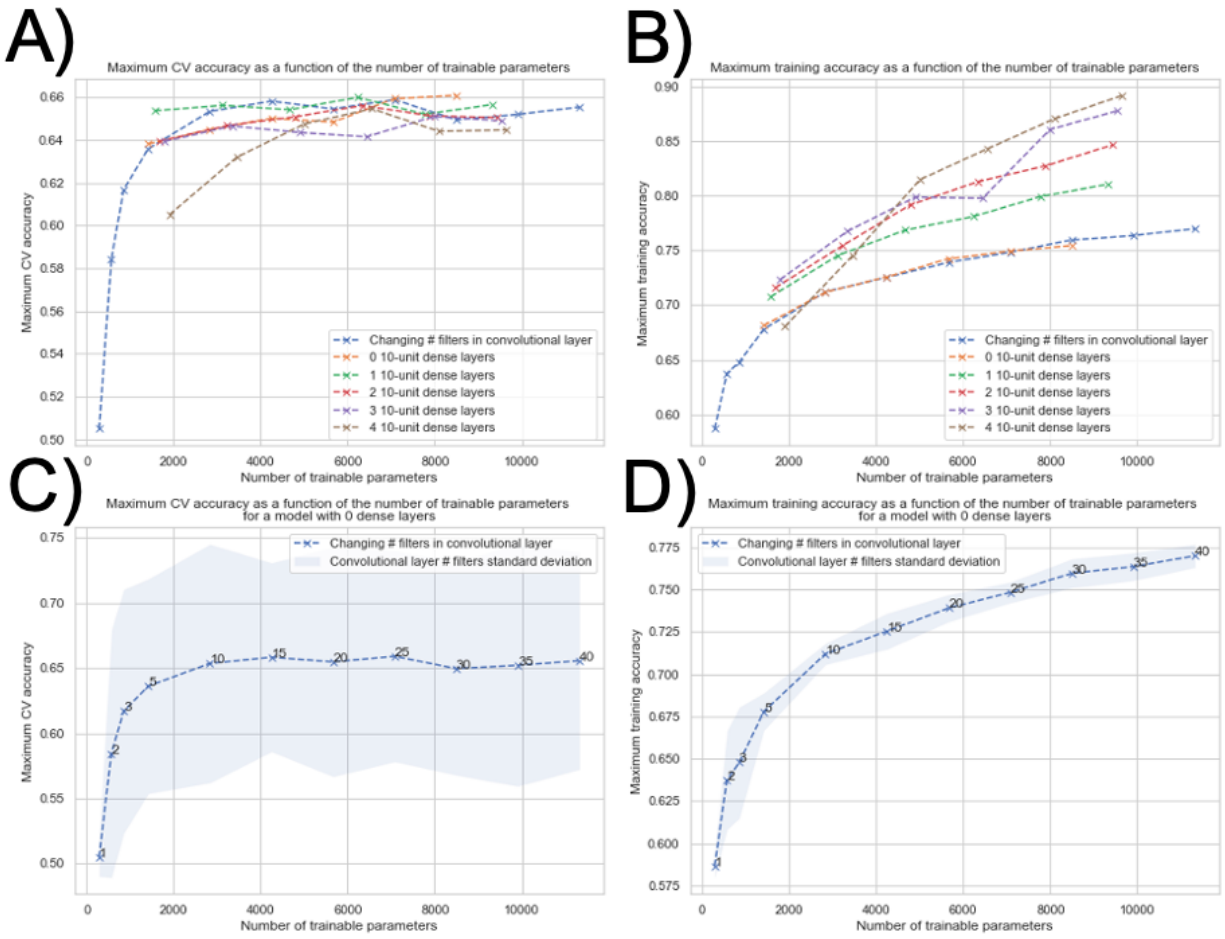

Figure 6: Performance evaluation as a function of model complexity. **A)**Average 10-fold cross-validation accuracy as a function of the number of trainable parameters for different numbers of convolutional filters and 10-unit dense layers. **B)** Average training accuracy as a function of the number of trainable parameters for different numbers of convolutional filters and 10-unit dense layers. **C)**Average 10-fold cross-validation accuracy for CNN with 0 dense layers and a varying number of filters shown by the numbers on the data points. **D)**Average training accuracy for CNN with 0 dense layers and a varying number of filters shown by the numbers on the data points.

From this analysis, we were able to conclude that the ideal CNN architecture was a single convolutional layer with 15 filters, a stride of 1, 0 padding and a filter size of (3,93). To reduce the number of parameters and motivated by Dai et al., we introduced a maximum pooling layer with a stride of 10 in the dimension corresponding to the forward propagated time features in the output of the convolutional layer [15]. This has the effect of only keeping the most significant features for the subsequent sigmoid classification unit. The final tuned model is presented in figure 7a.

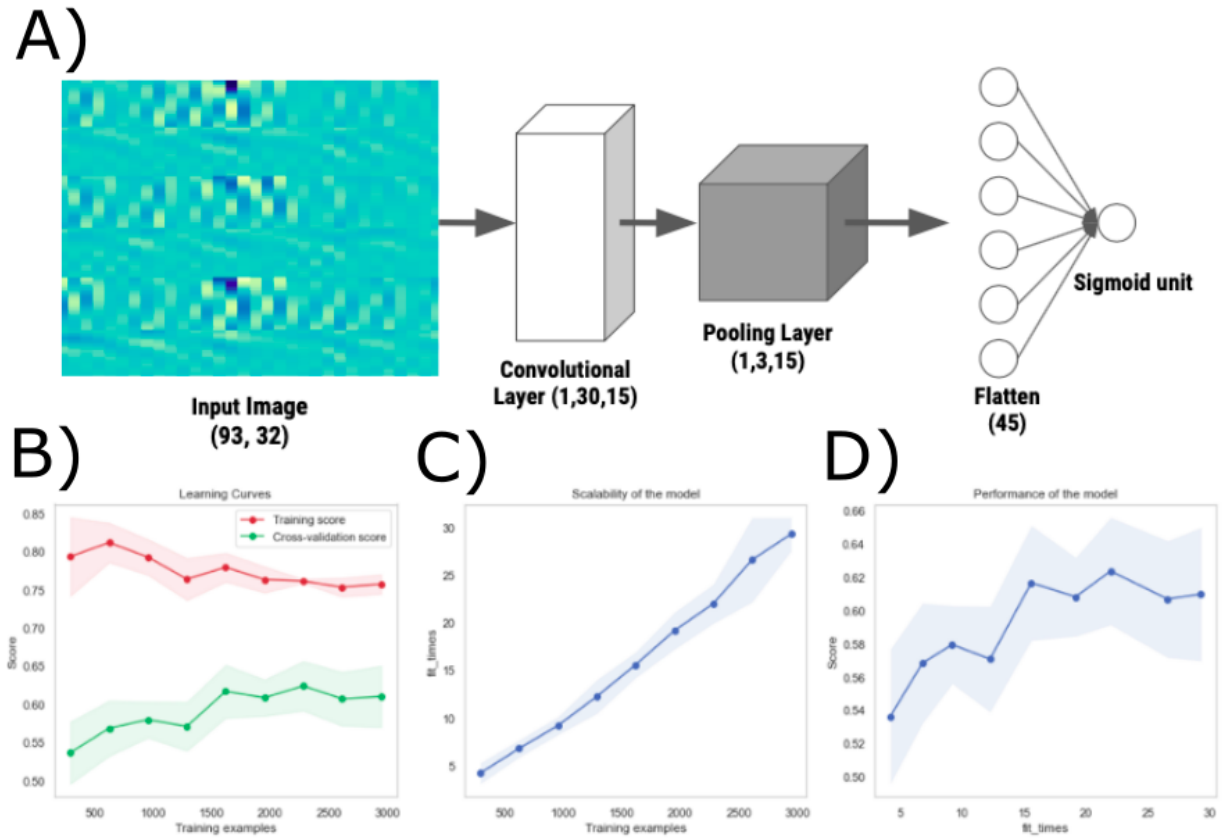

Figure 7: A) This figure shows the best CNN architecture the MI classification task. The first layer is a convolutional layer with 15 filters of size (93,3), the size 93 is in the frequency direction and 3 in the time direction. Next there is a pooling layer with a stride of 10 in the forward propagated time dimension and finally there is a sigmoid activation unit for classification. **B)** Learning curve of the fine tuned model. **C)** Fitting time as a function of the number training exemple. **D)** Cross validation accuracy as a function of fitting time.

The performance of the fine-tuned model is further evaluated by plotting the learning curves as shown in Figure 7b. We can observe in the cross-validation learning curve that the accuracy is levels off as the full size of the training set is used, suggesting that the number of parameters chosen was ideal for this particular task. Indeed, if the accuracy was still increasing, it would indicate that the model would have been more appropriate for a larger dataset. However, we can still notice that there is a 15% gap between the training accuracy and the cross validation accuracy. This indicates that it is difficult to generalize a model trained on other individuals to a new individual since, as has already been mentioned, motor imagery profiles vary greatly between people. In Figure 7c and 7d, we also evaluated the scalability and performance of the model by plotting the training time as a function of the size of the dataset and found that it scaled linearly.

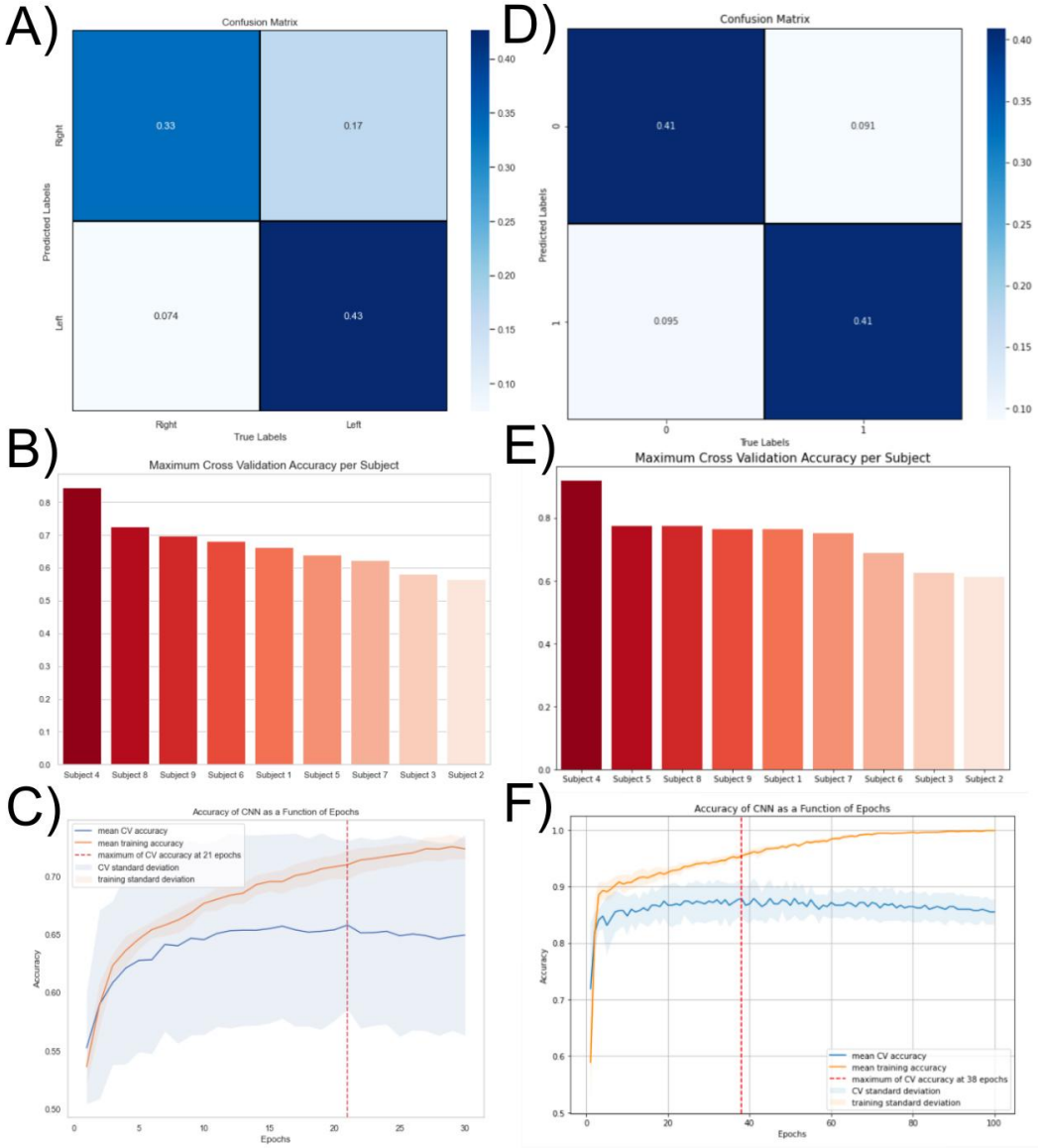

Figure 8: A) Leave-one-subject out average 10-fold cross-validation confusion matrix. **B)** Maximum cross-validation accuracy per subject resulting from early stopping and training via a leave-one subject out scheme. **C)** Training process resulting from the leave-one-subject out training scheme across all individuals. **D)** Subject-specific average 10-fold cross-validation confusion matrix of a responsive subject (subject 4). **E)** Maximum cross-validation accuracy per subject resulting from early stopping and training via a subject-specific scheme. **F)** Training process resulting from the subject-specific training scheme across all individuals.

This analysis served as motivation for the subsequent comparison between a subject-specific training scheme and a leave-one-subject-out training scheme, which gives an estimate of the expected results from training the model on different individuals, and with subject-specific datasets. Considering that obtaining a sufficient amount of quality data is very time consuming and requires a professional setup, this subject specific approach was not feasible for us but would

be ideal given more time and resources. The goal of this analysis is to compare this ideal subject specific training scheme and our training scheme on other individuals to assess the potential for future improvements. In Figure 8a, 8b, and 8c, the results for the leave-one-subject-out training scheme are shown, while the results for the subject-specific training are shown in figure 8d, 8e, and 8f. It was found that the mean accuracy across all subjects increased from 65 ∓12.2% to 74∓8.6 % when using subject-specific datasets. The standard deviations are similar but the means had a difference of almost 10%, which indicates that subject specific training is indeed the preferred method. In figure 8b and 8e, we can also notice that the performance of the algorithm using one training scheme is highly correlated with the performance of the algorithm under the other training scheme on the same patient. This gives evidence to the idea that some individuals are better at MI than others.

These models were integrated into a real-time system, and a video demo is available here.

Discussion and Conclusions

These results demonstrate that jaw clenches are a robust, easily identifiable control signal for BCI interaction. This finding is important for improving interaction paradigms for asynchronous BCIs. Previous research in this area has studied error-potentials (ErrP), where a recognizable electrical pattern arises in EEG when a user commits an error (i.e. unintentionally initiates an action) [23]. This provides a mechanism for knowing when an asynchronous BCI is erroneously activated. However, relying on the response from committing an error is frustrating for the BCI user, and thus the development of other reliable control signals is valuable. Of course, there are limitations to using a jaw clench approach. For instance, jaw clenches cannot be reliably produced by LIS patients, who have complete loss of facial muscular control. For this reason, exploring the modeling of other EEG-derived control signals such as closing the eyes would be a valuable extension of this project, and would improve the usability of this device for locked-in or other severely paralyzd patients. Moreover, the jaw clench dataset analyzed in this report was collected entirely off the head of one group member, and it's yet to be seen whether this model will generalize to other individuals. For instance, it may be the case that resting mean bandpower varies across individuals; this has been found to be the case in patients afflicted with bipolar disorder, addiction, depression, and other psychiatric disorders [24]. To more extensively verify jaw clenches to be an effective control signal, a larger dataset collected across demographic boundaries would be necessary.

Further, for the MI modeling, as was stated earlier, using 2 s segments with 64 sample resolution in the STFT effectively only provides a frequency resolution 3.91 Hz. This is far from adequate for effectively distinguishing and characterizing alpha power. Although the feature representations provided adequate information for subject-specific and, to a lesser extent, subject-independent modeling, the effect of expanding the segment and window lengths would

be useful to explore in the future as this may be a bottleneck on model performance. For example, the segments could be extended by 1 second on either side, and a 128 sample length could be used to bring the frequency resolution below 2 Hz. This would come at the expense of decreased time resolution and a longer BCI interaction paradigm which could be tiring for the user. However, the increase in frequency information may provide model performance boosts that justify this. Moreover, other methods for feature extraction could be explored such as the discrete wavelet transform (DWT). A complete explanation of the DWT is beyond the scope of this report, but essentially many different scales of some mother wavelet are applied to the data in question using a hierarchical algorithm. Wavelets of large scales provide low frequency information and low time resolution, while wavelets of short scale provide high frequency information at high time resolution . This has the effect of producing a time-frequency image, but unlike the STFT, the time and frequency resolution varies at different wavelet scales. In particular, frequency resolution is higher at low frequencies, while time resolution is higher at large frequencies [25]. Using this transformation may give greater frequency resolving power in the alpha band, and would be a useful extension of this project. Another feature extraction method that's been commonly used in MI is the common spatial pattern (CSP) algorithm, which learns filter weights that decompose EEG data into two classes with a maximal difference in variance [26]. These extracted patterns have been applied successfully in tandem with models such as linear discriminant analysis in the past, and exploring this would have been a valuable benchmark for the more advanced deep learning approach [26], [27]

As expected from the model complexity analysis, a relatively simple architecture was sufficient to obtain the maximum leave-one-subject-out cross-validation accuracy as it peaked at 15 filters in the convolutional layer. Furthermore, adding additional 10 unit dense layers only led to increased overfitting. This is expected since the dataset was relatively small with only 3600 instances. With increased model complexity, the hypothesis space of the model increases which enables to learn more complicated decision boundaries but increases the risk of overfitting [28]. Deep-learning algorithms are usually used when large amounts of data are available or when transfer learning can be applied [29]. The reason why convolutional neural networks were used for this classification task was their exceptional performance on image classification data and was motivated by previous results in the scientific literature [10]–[13], [30]. However, it would have provided additional insight if the optimal CNN's performance was compared to other simpler machine learning algorithms such as logistic regression, and especially ensemble classifiers such as random forests considering their proven effectiveness on small image datasets [31]–[33]. Next, we confirmed that a CNN based classifier performed better when trained on the subject specific dataset as opposed to other individuals by about 10%. This was expected as the MI response varies greatly among individuals and it is difficult to generalize from one individual to another [34]. Additionally, individuals who performed better on the leave-one-subject-out cross validation scheme also tended to perform better with subject specific training, giving credence to the idea that some individuals are more gifted at motor imagery; a clear example being subject 4 in our dataset (Figure 8b, 8e). In fact, it's been estimated that 15-30% of the

population is MI-illiterate [35]. As such, for the practical adoption of such a device, a screening stage would be necessary to characterize an individual's MI-literacy. In conclusion, we can say that the performance of CNNs was very satisfactory especially for individuals with strong MI responses. For future improvements, an experimental setup should be designed to easily and rapidly obtain a dataset from the subject of interest for specific training.

Team Work

Kenji performed the jaw clench and MI feature extraction, conducted jaw clench modeling with logistic regression, and implemented the real-time system. Arthur conducted deep learning MI modeling/experimentation and aided in data collection.

References

[1] J. J. Shih, D. J. Krusienski, and J. R. Wolpaw, "Brain-Computer Interfaces in Medicine," *Mayo Clin Proc*, vol. 87, no. 3, pp. 268–279, Mar. 2012, doi: 10.1016/j.mayocp.2011.12.008.

[2] S. N. Abdulkader, A. Atia, and M.-S. M. Mostafa, "Brain computer interfacing: Applications and challenges," *Egyptian Informatics Journal*, vol. 16, no. 2, pp. 213–230, Jul. 2015, doi: 10.1016/j.eij.2015.06.002.

[3] A. Fenton and S. Alpert, "Extending Our View on Using BCIs for Locked-in Syndrome," *Neuroethics*, vol. 1, no. 2, pp. 119–132, Jul. 2008, doi: 10.1007/s12152-008-9014-8.

[4] U. Chaudhary and N. Birbaumer, "Communication in locked-in state after brainstem stroke: a brain-computer-interface approach," *Ann Transl Med*, vol. 3, no. Suppl 1, May 2015, doi: 10.3978/j.issn.2305-5839.2015.02.27.

[5] B. S. Armour, E. A. Courtney-Long, M. H. Fox, H. Fredine, and A. Cahill, "Prevalence and Causes of Paralysis—United States, 2013," *Am J Public Health*, vol. 106, no. 10, pp. 1855–1857, Oct. 2016, doi: 10.2105/AJPH.2016.303270.

[6] E. Smith and M. Delargy, "Locked-in syndrome," *BMJ*, vol. 330, no. 7488, pp. 406–409, Feb. 2005.

[7] R. Zerafa, T. Camilleri, O. Falzon, and K. P. Camilleri, "A Real-Time SSVEP-Based Brain-Computer Interface Music Player Application," in *XIV Mediterranean Conference on Medical and Biological Engineering and Computing 2016*, Cham, 2016, pp. 173–178, doi: 10.1007/978-3-319-32703-7_36.

[8] J. Xie, G. Xu, J. Wang, M. Li, C. Han, and Y. Jia, "Effects of Mental Load and Fatigue on Steady-State Evoked Potential Based Brain Computer Interface Tasks: A Comparison of Periodic Flickering and Motion-Reversal Based Visual Attention," *PLOS ONE*, vol. 11, no. 9, p. e0163426, Sep. 2016, doi: 10.1371/journal.pone.0163426.

[9] K. Nakayashiki, M. Saeki, Y. Takata, Y. Hayashi, and T. Kondo, "Modulation of event-related desynchronization during kinematic and kinetic hand movements," *Journal of NeuroEngineering and Rehabilitation*, vol. 11, no. 1, p. 90, May 2014, doi: 10.1186/1743-0003-11-90.

[10] A. Kar, S. Bera, S. P. K. Karri, S. Ghosh, M. Mahadevappa, and D. Sheet, "A Deep Convolutional Neural Network Based Classification Of Multi-Class Motor Imagery With Improved Generalization," *Annu Int Conf IEEE Eng Med Biol Soc*, vol. 2018, pp. 5085–5088, Jul. 2018, doi: 10.1109/EMBC.2018.8513451.

[11] S. Taheri, M. Ezoji, and S. M. Sakhaei, "Convolutional neural network based features for motor imagery EEG signals classification in brain–computer interface system," *SN Appl. Sci.*, vol. 2, no. 4, p. 555, Mar. 2020, doi: 10.1007/s42452-020-2378-z.

[12] Z. Tang, C. Li, and S. Sun, "Single-trial EEG classification of motor imagery using deep convolutional neural networks," *Optik*, vol. 130, pp. 11–18, Feb. 2017, doi: 10.1016/j.ijleo.2016.10.117.

[13] Y. Rong, X. Wu, and Y. Zhang, "Classification of motor imagery electroencephalography signals using continuous small convolutional neural network," *International Journal of Imaging Systems and Technology*, vol. 30, no. 3, pp. 653–659, 2020, doi: https://doi.org/10.1002/ima.22405.

[14] S. L. Kappel, D. Looney, D. P. Mandic, and P. Kidmose, "Physiological artifacts in scalp EEG and ear-EEG," *BioMedical Engineering OnLine*, vol. 16, no. 1, p. 103, Aug. 2017, doi: 10.1186/s12938-017-0391-2.
[15] M. Dai, D. Zheng, R. Na, S. Wang, and S. Zhang, "EEG Classification of Motor Imagery Using a Novel Deep Learning Framework," *Sensors (Basel)*, vol. 19, no. 3, Jan. 2019, doi: 10.3390/s19030551.
[16] M. Grasmair, "Basic Properties of Convex Functions." Department of Mathematics, Norwegian University of Science and Technology.
[17] H. H. Aghdam and E. J. Heravi, *Guide to Convolutional Neural Networks: A Practical Application to Traffic-Sign Detection and Classification*. Springer International Publishing, 2017.
[18] R. Shanmugamani, A. Ghani. Abdul Rahman, S. Maurice. Moore, and Nishanth. Koganti, *Deep Learning for Computer Vision: Expert techniques to train advanced neural networks using TensorFlow and Keras.*, 1 online resource (304 pages) vols. Birmingham: Packt Publishing, 2018.
[19] M. Abadi *et al.*, "TensorFlow: a system for large-scale machine learning," in *Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation*, USA, Nov. 2016, pp. 265–283, Accessed: Apr. 25, 2021. [Online].
[20] S. H. Johnson, "Cerebral Organization of Motor Imagery: Contralateral Control of Grip Selection in Mentally Represented Prehension," *Psychol Sci*, vol. 9, no. 3, pp. 219–222, May 1998, doi: 10.1111/1467-9280.00042.
[21] Y. Jeon, C. S. Nam, Y.-J. Kim, and M. C. Whang, "Event-related (De)synchronization (ERD/ERS) during motor imagery tasks: Implications for brain–computer interfaces," *International Journal of Industrial Ergonomics*, vol. 41, no. 5, pp. 428–436, Sep. 2011, doi: 10.1016/j.ergon.2011.03.005.
[22] M. Ahn and S. C. Jun, "Performance variation in motor imagery brain-computer interface: a brief review," *J Neurosci Methods*, vol. 243, pp. 103–110, Mar. 2015, doi: 10.1016/j.jneumeth.2015.01.033.
[23] R. Yousefi, A. Rezazadeh Sereshkeh, and T. Chau, "Development of a robust asynchronous brain-switch using ErrP-based error correction," *J Neural Eng*, vol. 16, no. 6, p. 066042, Nov. 2019, doi: 10.1088/1741-2552/ab4943.
[24] J. J. Newson and T. C. Thiagarajan, "EEG Frequency Bands in Psychiatric Disorders: A Review of Resting State Studies," *Front. Hum. Neurosci.*, vol. 12, 2019, doi: 10.3389/fnhum.2018.00521.
[25] M. Weeks and M. Bayoumi, "Discrete Wavelet Transform: Architectures, Design and Performance Issues," *The Journal of VLSI Signal Processing-Systems for Signal, Image, and Video Technology*, vol. 35, no. 2, pp. 155–178, Sep. 2003, doi: 10.1023/A:1023648531542.
[26] I. Xygonakis, A. Athanasiou, N. Pandria, D. Kugiumtzis, and P. D. Bamidis, "Decoding Motor Imagery through Common Spatial Pattern Filters at the EEG Source Space," *Computational Intelligence and Neuroscience*, vol. 2018, p. e7957408, Aug. 2018, doi: 10.1155/2018/7957408.
[27] L. F. Velásquez-Martínez, A. M. Álvarez-Meza, and C. G. Castellanos-Domínguez, "Motor Imagery Classification for BCI Using Common Spatial Patterns and Feature Relevance Analysis," in *Natural and Artificial Computation in Engineering and Medical Applications*, Berlin, Heidelberg, 2013, pp. 365–374, doi: 10.1007/978-3-642-38622-0_38.

[28] Michael. Doumpos and Evangelos. Grigoroudis, *Multicriteria decision aid and artificial intelligence: links, theory and applications*, 1 online resource vols. Chichester, West Sussex, U.K.: John Wiley, 2013.
[29] "Deep learning applications and challenges in big data analytics," *Journal of Big Data*. https://journalofbigdata.springeropen.com/articles/10.1186/s40537-014-0007-7) (accessed Apr. 25, 2021).
[30] U. Michelucci, *Advanced applied deep learning: convolutional neural networks and object detection*, 1 online resource : illustrations (some color) vols. New York: Apress, 2019.
[31] A. Bosch, A. Zisserman, and X. Munoz, "Image Classification using Random Forests and Ferns," in *2007 IEEE 11th International Conference on Computer Vision*, Oct. 2007, pp. 1–8, doi: 10.1109/ICCV.2007.4409066.
[32] C. Yoo, D. Han, J. Im, and B. Bechtel, "Comparison between convolutional neural networks and random forest for local climate zone classification in mega urban areas using Landsat images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 157, pp. 155–170, Nov. 2019, doi: 10.1016/j.isprsjprs.2019.09.009.
[33] R. Arbel and L. Rokach, "Classifier evaluation under limited resources," *Pattern Recognition Letters*, vol. 27, no. 14, pp. 1619–1631, Oct. 2006, doi: 10.1016/j.patrec.2006.03.008.
[34] Th. Mulder, "Motor imagery and action observation: cognitive tools for rehabilitation," *J Neural Transm*, vol. 114, no. 10, pp. 1265–1278, Oct. 2007, doi: 10.1007/s00702-007-0763-z.
[35] T. Dickhaus, C. Sannelli, K.-R. Müller, G. Curio, and B. Blankertz, "Predicting BCI performance to study BCI illiteracy," *BMC Neurosci*, vol. 10, no. 1, Art. no. 1, Sep. 2009, doi: 10.1186/1471-2202-10-S1-P84.