FYRP Report: Towards a real-time Driving BCI

Tim Stadlbauer s5640245
Supervisors: Andreea Sburlea, Ivo de Jong

Abstract

Previous research in motor imagery-based brain computer interfaces showed that training a classifier on simple calibration paradigm and transferring it to a more complex task is a feasible approach. Still this was only done using isolated epochs with a consistent ground truth. For a online BCI system to function we need to make predictions at every timestamp. In this paper we set out to investigate the effects transfer learning has when applied to a sliding window approach, instead of isolated trials. Furthermore different techniques such as a filter-bank CSP and individual frequency CSP were implemented to try and improve the performance of the classifiers. From the investigation of the results it can be seen that none of the more novel approaches perform better than the baseline. This can be explained by the differences in the calibration and the more complex driving paradigm. This shows that transferring from a simple calibration task to an online-BCI setting is difficult endeavor that needs further investigation. Specifically designing or changing the control paradigms in a way to reflect the setting of the more complex task.

Introduction

This research project primarily builds upon the work conducted in (de Jong, van den Wittenboer, Valdenegro-Toro, & Sburlea, 2024), in which the main focus points were creating a driving paradigm and collect data to create EMG and EEG-based algorithms. In their work they collected data from 20 participants 19 of which were included in the further investigation. Participants in this study were first tasked to participate in the Graz-BCI Motor Imagery paradigm to collect calibration data, this data was then used to create a EMG classifier for each participant and allowed them to steer a car left and right in a virtual setting using wrist flexors. During all of this EEG data was also collected to create an offline classifier. Epochs where created in this paper based on continuous predictions of the EMG classifier, meaning that whenever it predicted the same label for 3.75 seconds it was considered as one class. The classification accuracies in this study show that there is a huge difference in performance between individual participants, f1-scores are ranging from 0.79 to 0.2.

The main research question this project tried to answer is how to transition from extracted epochs towards continuous predictions. Furthermore the main goal was to find a way to improve upon a simple baseline to get to a starting point where this classifier could be put to use. The ultimate goal at the start of the project was to then find a way to interleave EMG and EEG predictions to have participants learn on the fly how to use the EEG system. This was a bit too ambitious as the other aspects already proved to be more difficult/ showed less improvement than initially expected.

Methods

After getting acquainted with some literature, the provided code from (de Jong et al., 2024) and MNE itself, the first step in the project was to establish a baseline for continuous prediction. This was accomplished similarly to what was already done in (de Jong et al., 2024), which means the recorded calibration data was loaded, a band-pass filter between 5 and 35 Hz was applied and ICA channels were rejected based on the manual inspection done by Ivo de Jong. This data was then split into epochs based on markers set when recording the data, after which the most informative six channels where selected using common spatial patterns. Furthermore these six feature channels were then used as input for the logistic regression algorithm to predict the labels for each epoch. The same pre-processing steps were applied to the driving data, using the ICA fitted for the calibration data, with the crucial difference that the epochs are not based on continuous prediction of the EMG classifier but rather a sliding window approach was taken. The epochs have a length of 2 seconds with an overlap of 1.8 seconds, resulting in predictions every 200ms, in line with the EMG classifier. After making these epochs they were down-sampled to 1024 Hz to improve computational time and resources needed. This did not impact the performance much (I don't have the actual values anymore but I think it was a percent at most). For this the EMG predictions were used as the ground truth. The classifiers were trained both in the "normal" case with left, right and rest predictions as well as in the binary left and right case. For this all the rest predictions from the EMG classifier were removed from the driving data. The results of the sliding window approach are compared to the results achieved by (de Jong et al., 2024) in figure 3 and 4 for subject 66 and 812 respectively.

Filter Bank CSP

The first attempt to improve the accuracy of the baseline method was to incorporate a filter bank to find more patterns that are important for the prediction. This was done because it seems to be a promising method when looking at literature dealing with similar problems. In the BCI IV (Tangermann et al., 2012) challenge for example the winning entry used a filter bank CSP as well as some entries ranked 3-5.

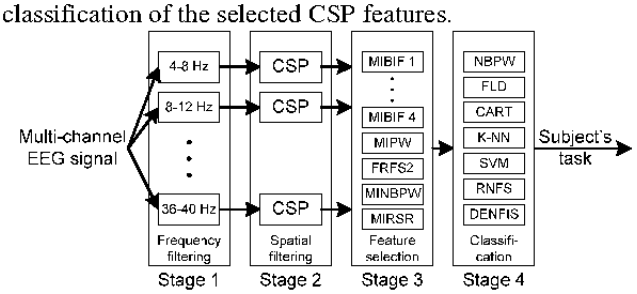


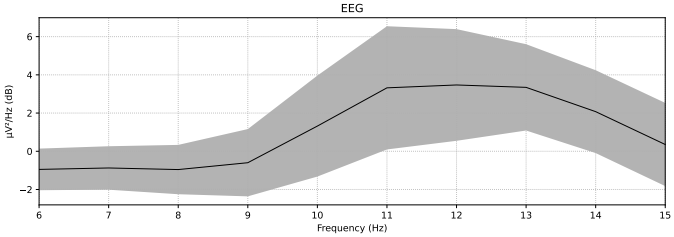
Fig. 1. Architecture of the proposed Filter Bank Common Spatial

Figure 1: Schematic figure of a filter bank CSP approach from (Ang et al., 2008).

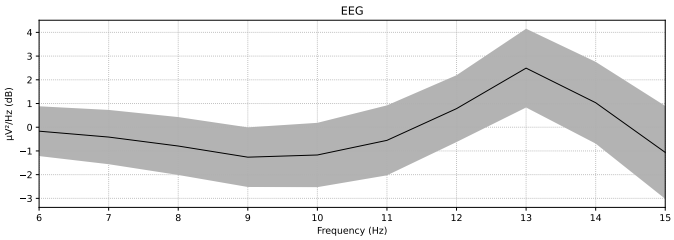
The main idea of filter bank CSP is that the incoming data stream gets filtered at different frequencies that can be arbitrarily selected. An intuitive filter bank would have filters for each frequency band, but it is not limited to that approach. The winners of the BCI-IV challenge adopted a filter bank with 8 filters and constant intervals in the logarithmic domain resulting in narrower frequency centers at lower frequencies and more spread apart filters in the higher frequencies. (Zhang, Guan, Ang, & Chin, 2012). After that some feature selection has to be done to find the most informative features that are then used as input for the classifier. In my work I started with the more simple approach adding additional filters in the alpha band to take multiple peak alpha frequencies into account. The pre-processing steps here are the same as mentioned above, but after the epochs are made, the filter bank CSP is applied and six CSP channels are extracted for each of the filter ranges. The 10 most informative channels were then extracted from all frequency ranges using mutual information as the criterion, and then used as input for the logistic regression algorithm. After this has been trained on the calibration data the same CSP channels were extracted from the driving data. A major difference the filter bank approach has to the baseline is that due to the implementation ram usage was too high to use the same two second windows with 1.8 second overlap. A first experiment was done using only 200ms windows with no overlap yielding poor results. Future work could re-try using larger windows with a dynamic generation of the windows to reduce the demand on the ram. This will have negative effects on runtime which already took up to an hour per participant. The detailed results of this section can be found in the GitHub repository.

Individual Alpha Frequency

Using individual alpha frequencies was the second approach I looked at when trying to improve from the baseline predictions. This approach again is widely used in the field and promising as can be seen by the BCI-IV challenge. The second place entry used IAF as well as the ones that did not use frequency banks in spots 3-5 (Tangermann et al., 2012). For this approach the power spectral densities of participants were plotted and then a peak alpha frequency was visually extracted. A small window around this peak was then used to filter the data and extract the CSP from it.



(a) Subject 66: Calibration Power spectral density, the black line showing the average over all channels while the grey part is the standard deviation.



(b) Subject 66: Driving Power spectral density, the black line showing the average over all channels while the grey part is the standard deviation.

Figure 2: (a) The power spectral density plot from the calibration data of the participant with ID 066. A slight peak can be seen at around 11 Hz, but it plateaus and stays at a similar value until 13 Hz. (b) Shows the PSD for the driving data of subject 66. A clear peak can be seen at 13 Hz.

After performing visual inspection, the individual peak frequencies were also computed automatically. This calculation was quite simple and would need some refinement to be more precise but it does find a fairly good filter for most participants. It calculates the average power for each frequency over all channels and then takes the range from one Hz above and below the peak. The main drawback of this is when the PSD shows a plateau like in Figure 2a it might take an interval that is sub-optimal. For the above-mentioned subject 66, the automatic detection would suggest 10-11 as the peak interval but we can see that 11-13 would be a better choice. Results for the IAF of participant 066 and 812 are shown in Figure 5 and further discussed in the results section.

Training on Driving Data

After investigating methods to improve upon the baseline with little avail, the next step was to look into switching

training and testing sets around to see if the algorithm would transfer better from driving to calibration data. This was done in order to see if a predictor trained on more complex data is able to transfer to easier tasks. This would make the training procedure more fun for participants as driving a virtual car is more interesting than seeing arrows on the screen. Additionally participants in a calibration task are usually more focused and perform the actions more deliberately resulting in stronger signals that influence the thresholds that the predictor learns.

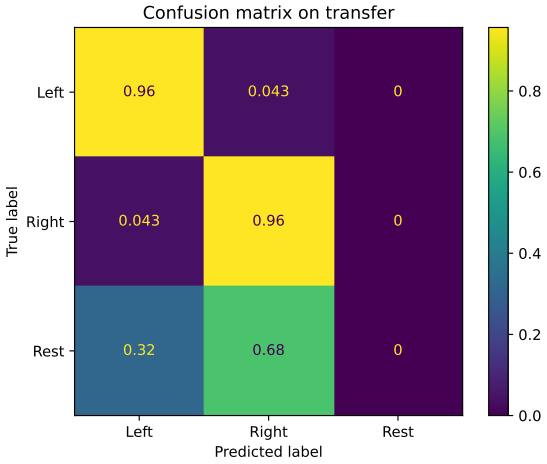
This was done by training the CSP and Logistic regression on the driving epochs of 2 seconds again with the 1.8 second overlap, and then the calibration data was used for testing.

Another thing I briefly looked into was training with part of the driving data and then testing on the rest of it. For this the driving data was split into train and test sets (80/20). The only difference in pre-processing was that this time the driving ICA was used instead of the calibration.

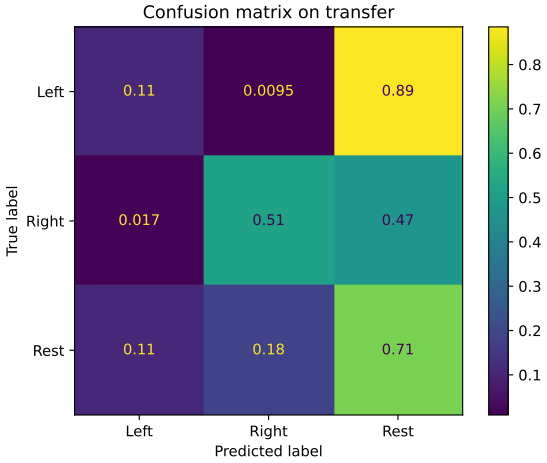
Results

In this section I will present an overview of the results obtained using the various approaches mentioned in the Methods section. Most of the results will show two of the 19 usable subjects namely subject 066 and subject 812. Those were chosen as they are representative of the group's average performance.

Making predictions every 200ms is a more difficult task than making predictions for pre-defined epochs based on EMG predictions like it was done in (de Jong et al., 2024). Prediction are not only made with less data but also the ground truth is more complex. Small corrections are not included in the epoched setting but have to be predicted in the sliding window approach. Following that it was expected that the predictions are less accurate.



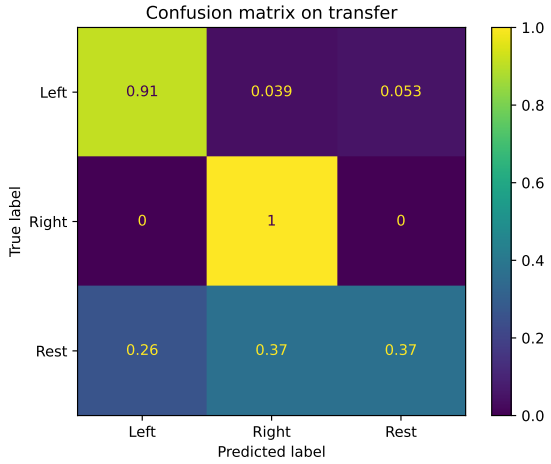
(a) Subject 66: Confusion Matrix showing performance on transfer using epochs based on continuous EMG predictions.



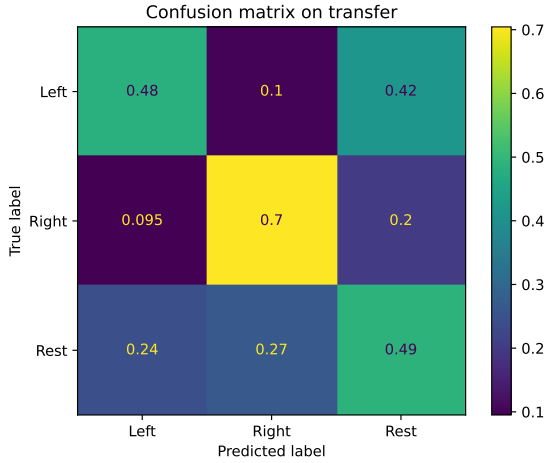
(b) Subject 66: Confusion Matrix using sliding windows and prediction every 200ms with the baseline predictor. (Note that the header of this plot is wrong, it is transfer and sliding window confusion matrix)

Figure 3: The confusion matrices show the achieved performance in the previous study (de Jong et al., 2024) and compares it to the baseline achieved in this project.

It can be seen in Figure 3 that the performance decreases, especially prediction left vs. right seems to be a harder task in a continuous setting. A noticeable change is that the new baseline predicts the rest cases well, and it gets half of the right cases correct. The major problem here is that this baseline for subject 066 predicts rest in a majority of the cases.



(a) Subject 812: Confusion Matrix showing performance on transfer using epochs based on continuous EMG predictions.



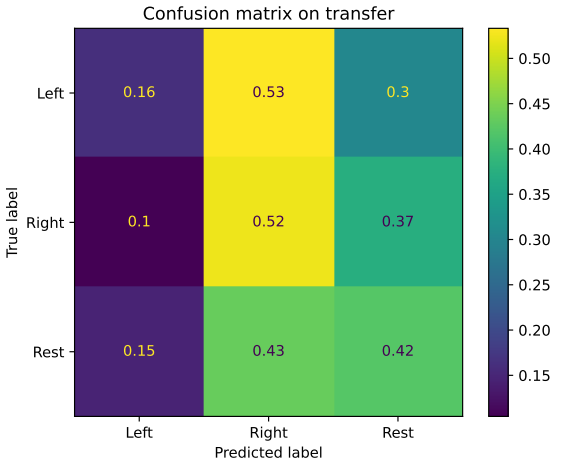
(b) Subject 812: Confusion Matrix using sliding windows and prediction every 200ms with the baseline predictor.

Figure 4: The confusion matrices show the achieved performance in the previous study by (de Jong et al., 2024) and compares it to the baseline achieved in this project.

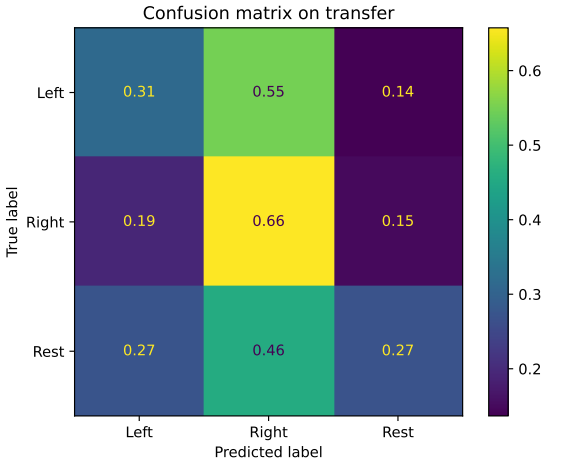
Subject 812 shows similar results but we can see that the overall performance is better in the baseline case than what we see for subject 066. This already starts to resemble what was found in the previous study (de Jong et al., 2024) that the results are severely subject dependent. The subjects that performed well in their study like subject 061, 381 and 812 typically also performed better in this work, but a detailed comparison has not been made about the overall correlation of the participant performances.

This was one reason why as a first approach to improve performance a filter bank CSP was applied, as this in theory will capture differences in frequency ranges from subject to subject. The implementation of this was too inefficient to be run on the large data using bigger windows with overlap, thus resulting in overall very poor performance across the board. The results of this are included in the GitHub repository at the end of the results section.

Using the individual alpha frequency to extract subject-specific features has been shown to be an effective method (Pfurtscheller, Brunner, Schlögl, & Lopes da Silva, 2006).



(a) Subject 066: Confusion Matrix using the individual peak frequency.

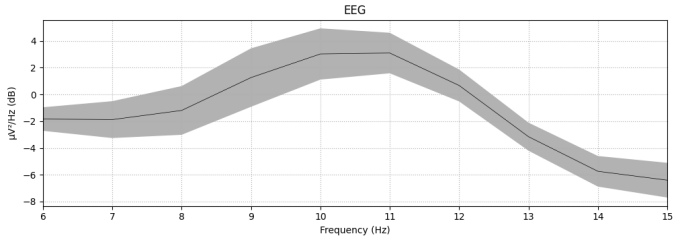


(b) Subject 812: Confusion Matrix using the individual peak frequency.

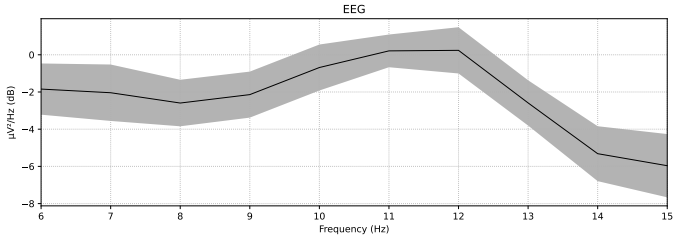
Figure 5: Above, the comparison between two subjects performance using the individual alpha frequency can be seen.

With this approach we run into the problem that was already hinted at when showing the PSD plots of subject 066 in Figure 2, the individual peak frequency might differ between calibration and driving paradigm. This also explains why performance decreases for participant 066. Participant 812 also shows this degrading performance, in this case the PSD do overlap better than it does for participant 066 but with the automated peak detection a different peak was found.

Inspecting it visually, the selected peak is the same selecting it at 11 Hz as this captures most of the elongated peak that can be seen.



(a) Subject 812: Power spectral density plot of the calibration data of participant 812.



(b) Subject 812: Power spectral density plot of the driving data of participant 812.

Figure 6: It can be seen that in (a) the peak is around 10 or 11 Hz, while in (b) the peak is shifted a bit backwards at 12 Hz. If we select the peak at 11 Hz in (a), which would be reasonable as the interval from 10-12 Hz captures the full extend of the elongated peak. This range also captures the full extend of the peak in (b).

Still a decrease in performance can be seen when using IAF over the baseline classifier. This means we are losing some crucial information that appears not in the alpha frequency band but in a different one. (de Jong et al., 2024) discusses the appearance of a short dip in the beta band that is aligned with movement onset, this might explain the decrease in performance.

The performance of a binary classifier was also investigated looking only at the conditions that are predicting left and right. The results of the binary classification can be seen in Figure 7, 8, 9, 10. Also in this setting we can see that the performance from baseline degrades when using the individual alpha frequency. This loss in performance is even more apparent in subject 812, as the prediction accuracy for left goes down by 40%. Overall the baseline results are quite promising. One way this could be implemented into a final online BCI setting is that whenever we have low certainty for either left or right we predict it as rest. The exact threshold would need investigation itself.

The final investigation went towards seeing how performance changes when we use different training data. Two different variations of this were looked at. First driving data was split into training and testing to see what performance we reach when the algorithm is trained on parts of the driving data. This was only checked for a single participant so far, as the focus was then put towards finding a way to improve the transfer accuracies. The results of this for participant 061 can be seen in table 1.

	Left	Right	Rest
Training f1-scores	0.59	0.6	0.59
Testing f1-scores	0.54	0.6	0.58
Filter-Bank f1-scores	0.058	0.0	0.6

Table 1: Table showing the f1-scores for training and testing on the driving data for participant 061 and compares it to the result of the filter bank approach that was trained on the calibration data and uses the same 200ms windows.

These results show already promising f1-scores for all three conditions for participant 061 while only using 200ms windows of data. These scores are better than what the filter-bank approach using transfer learning achieves even though the same window length is used. Showing once more that going from calibration to driving data is a difficult task.

The third variation was training on the driving data and then testing on the easier calibration data. The main purpose behind this variation was that it might be interesting to see if it is possible to train with more complex data and then solve easier tasks with the trained classifier. The results of this also vary from subject to subject.

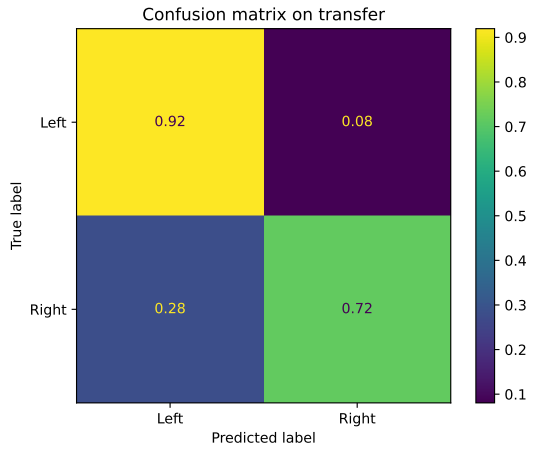


Figure 7: Subject 066: Baseline binary classification accuracies

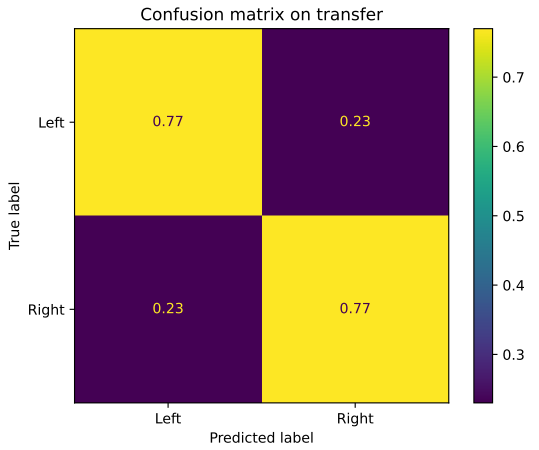


Figure 8: Subject 066: Classification accuracies using the individual alpha frequencies.

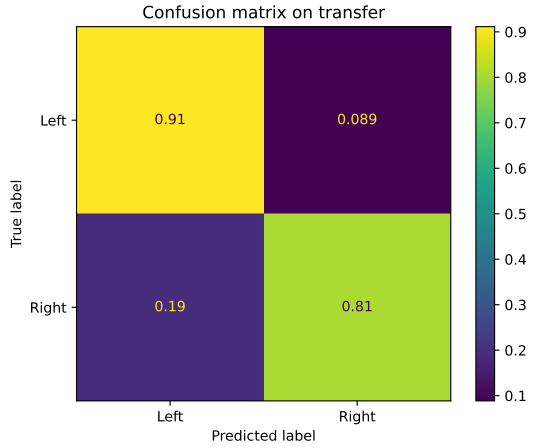


Figure 9: Subject 812: Baseline binary classification accuracies

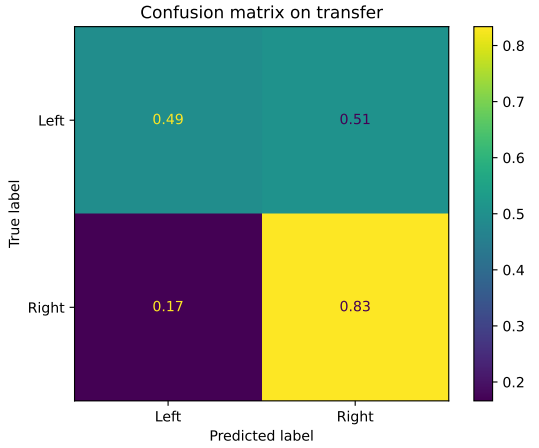
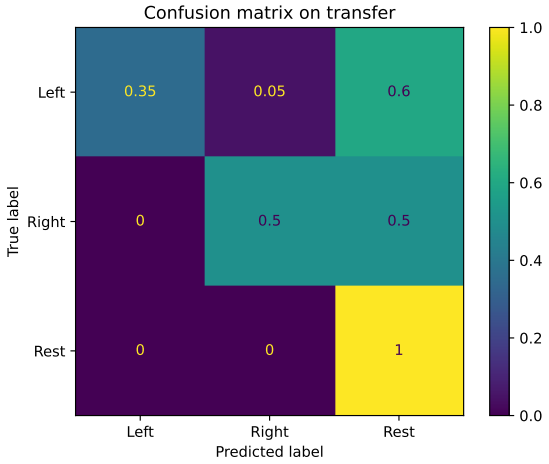
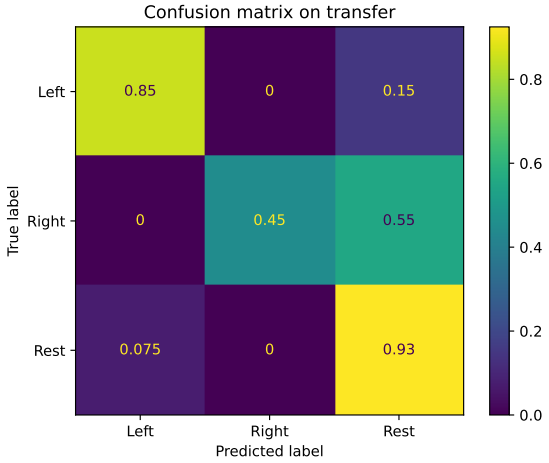


Figure 10: Subject 812: Classification accuracies using the individual alpha frequencies.



(a) Subject 061: Confusion Matrix showing the results when training on the driving data and testing on calibration.



(b) Subject 812: Confusion Matrix showing the results when training on the driving data and testing on calibration.

Figure 11: Results showing the difference between subjects in the reversed training and testing case.

It can be seen that subject 812 shows promising results especially for predicting left and rest. The right case is only predicted correctly 45% of the time. But comparing these results to subject 061, we can immediately see that this subject is preforming way worse in predicting the left case correctly.

The code for this project as well as results and the plots can be found in the Github Repository: `https://github.com/Stadopower/FYRP/tree/main`

Conclusion

From the previous sections it can be seen that all of the attempts to improve upon the baseline classifier failed to do so. The sliding window approach which was later used as a baseline showed that for some participants the performance is good while for others it is lacking. Furthermore comparing its results to the results from (de Jong et al., 2024) it can be seen that predicting continuously is harder than using isolated trials. The baseline results when only predicting left vs right shows good results, raising the question if a binary classifier could work. Rest would then be predicted if the certainty for either direction is low. Comparing the results of the baseline to the classifier using individual alpha frequencies we can see an overall decrease in performance for the binary and three class case. This can be attributed on the one hand to the differences in alpha peaks between calibration and driving, as can be seen in figure 2, and on the other hand to the fact that we are losing information from the beta band (de Jong et al., 2024) when we only extract the alpha band.

The filter bank approach did not yield good results, because the implementation is very RAM dependent and only small windows could be used. Future work might want to look into improving the efficiency of the algorithm and run it with bigger windows to conclusively see how this method compares to the others.

When investigating how well an algorithm functions when training it on the more complex driving data and testing it on the simpler calibration data it can be seen (Figure 11) that it is very subject dependent. Subject 812 showed quite promising results, but Subject 061 did not. (de Jong et al., 2024) already showed that there is a big difference between subjects, this was again seen in this work not only for the reversed learning approach but in general. The reason for this might be that some participants are better at controlling their EEG signals than others.

Lastly for Subject 061 it was investigated whether training the algorithm on driving data itself would be a feasible approach. This was done using short 200ms windows and already showed promising results. This needs further investigation in order to give proper insight on the problem. If this works well, it might be worth exploring how we can change the calibration paradigm to match the driving more closely in future research. Having a first shorter driving session using only EMG might be a suitable calibration task.

One other explanation that could result in the bad transfer-ability is that participants might be very focused in the calibration task resulting in a better and stronger EEG signal, which in turn leads to higher thresholds for the classification algorithm. The same signal strength might not be reached in the more noisy driving task, where the participants are confronted with more visual information and noise.

Another question that needs to be answered in the future is what accuracies are acceptable for EEG based BCI control. This could then be used as a starting point to investigate one of the initial research questions, how would a BCI work where we interleave EMG and EEG predictions to help participants learn relying more on EEG signals on the fly. Good starting accuracies are needed in order for the BCI system to not be frustrating.

References

Ang, K. K., Chin, Z. Y., Zhang, H., & Guan, C. (2008). Filter bank common spatial pattern (fbcsp) in brain-computer interface. *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 2390-2397. Retrieved from https://api.semanticscholar.org/CorpusID:14656324

de Jong, I. P., van den Wittenboer, L. L., Valdenegro-Toro, M., & Sburlea, A. I. (2024). *Transferring bci models from calibration to control: Observing shifts in eeg features.* Retrieved from https://arxiv.org/abs/2403.15431

Pfurtscheller, G., Brunner, C., Schlögl, A., & Lopes da Silva, F. (2006). Mu rhythm (de)synchronization and eeg single-trial classification of different motor imagery tasks. *NeuroImage*, *31*(1), 153-159. Retrieved from https://www.sciencedirect.com/science/article/pii/S1053811905025140 doi: https://doi.org/10.1016/j.neuroimage.2005.12.003

Tangermann, M., Müller, K.-R., Aertsen, A., Birbaumer, N., Braun, C., Brunner, C., ... Blankertz, B. (2012). Review of the bci competition iv. *Frontiers in Neuroscience*, *6*. Retrieved from https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2012.00055 doi: 10.3389/fnins.2012.00055

Zhang, H., Guan, C., Ang, K. K., & Chin, Z. Y. (2012). Bci competition iv – data set i: Learning discriminative patterns for self-paced eeg-based motor imagery detection. *Frontiers in Neuroscience*, *6*. Retrieved from https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2012.00007 doi: 10.3389/fnins.2012.00007