Adaptive Riemannian BCI for Enhanced Motor Imagery Training Protocols

Daniel Freer
Hamlyn Centre for Medical Robotics
Imperial College London
London, UK
d.freer15@imperial.ac.uk

Fani Deligianni
Hamlyn Centre for Medical Robotics
Imperial College London
London, UK
f.deligianni@imperial.ac.uk

Guang-Zhong Yang
Hamlyn Centre for Medical Robotics
Imperial College London
London, UK
g.z.yang@imperial.ac.uk

***Abstract*—Traditional methods of training a Brain-Computer Interface (BCI) on motor imagery (MI) data generally involve multiple intensive sessions. The initial sessions produce simple prompts to users, while later sessions additionally provide real-time feedback to users, allowing for human adaptation to take place. However, this protocol only permits the BCI to update between sessions, with little real-time evaluation of how the classifier has improved. To solve this problem, we propose an adaptive BCI training framework which will update the classifier in real time to provide more accurate feedback to the user on 4-class motor imagery data. This framework will require only one session to fully train a BCI to a given subject. Three variations of an adaptive Riemannian BCI were implemented and compared on data from both our own recorded dataset and the commonly used BCI Competition IV Dataset 2a. Results indicate that the fastest and least computationally expensive adaptive BCI was able to correctly classify motor imagery data at a rate 5.8% higher than when using a standard protocol with limited data. In addition it was confirmed that the adaptive BCI automatically improved its performance as more data became available.**

***Index Terms*—Brain-Computer Interface, Electroencephalography, Riemannian Geometry**

I. Introduction

In recent years, Brain-Computer Interfaces (BCI) have been commonly proposed as a method of aiding disabled and healthy individuals with communication, brain state analysis, and device control [1]. Efforts are focused on non-invasive methods of decoding brain signals based on motor imagery and EEG recordings. However, real-time control of assistive robots remains challenging. In order to ensure accurate MI classification, each user must currently complete several intensive training sessions that match their particular EEG signals to a classifying algorithm [2]. Using this type of training framework, the classifier can only be evaluated between sessions, while the user is resting. For this reason, a BCI that can adapt to incoming labelled data in real-time is highly desirable for the field. In addition, having a BCI that can improve its performance in real time is able to provide better feedback to the user more quickly during training sessions. This will lead their recorded data to better reflect their neural response with proper feedback, further enhancing the process.

This research is partially funded by the FAIR-SPACE grant, supplied by the EPSRC

Several attempts at making a robust and adaptive BCI have previously been explored with promising results. In one of the first attempts, it was shown that three to four sessions are required to achieve best performance for a non-adaptive BCI, while the adaptive BCI only required one session to achieve similar accuracy [3]. This adaptive BCI was based off of quadratic discriminant analysis (QDA), updating the inverse covariance matrix for each class using data from a selected interval of each trial [3]. Subsequent attempts focused on adaptive feature selection, mostly frequency bands [4], covariate shift estimation [5], [6] or spatial filters [7], [8], while a more recent attempt used positive and negative feedback to change fuzzy rules for the classification step [9].

Most of these adaptive BCI have utilised the Common Spatial Patterns (CSP) algorithm to essentially generate spatial filters and select features for a standard classifier. However, recent research has suggested that the application of Riemannian geometry to EEG data can achieve better performance than the CSP algorithm alone in most circumstances [10]. Typically, Riemannian based BCI methods estimate the mean covariance matrix of the EEG signal for each class by taking into account the manifold geometry of the multivariate signal. Subsequently, classification is based on the geodesic distance between each new covariance sample and the 'mean'. Here we explore methods to extend this framework into an adaptive BCI for real-time feedback and control. Our approach achieves better performance than traditional methods with a fraction of the training data normally used for successful multi-class BCI.

II. Methods

A. Datasets

1) Data Collection: For this study, we collected data from 3 healthy volunteers, undergoing 4-class BCI motor imagery training sessions with and without feedback. There were approximately 20 trials for each class during each run, each of which consisted of 2-3 seconds of a prompt (red arrows pointing either left, right, up, or down) to imagine either opening and closing their left hand, right hand, both hands, or both feet. For the feedback sessions, the prompt time was shortened to 2 seconds, and 2 seconds of feedback were also provided. These timings are relatively consistent with previous studies, though typically much more data is recorded. These

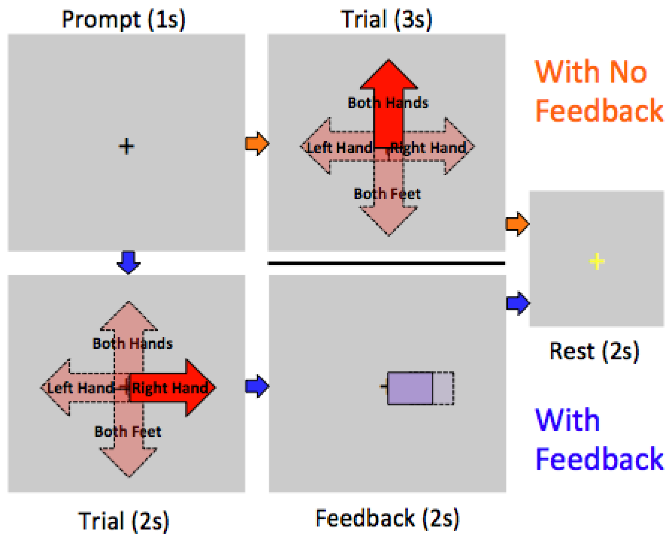

Fig. 1. The timing protocol for a single trial with (bottom, blue arrows) and without (top, orange arrows) feedback. This sequence repeated until 20 trials of each class were completed.

experiments were generated using PsychoPy [11], and the protocol can be seen in Figure 1.

Data was recorded using a 32-channel wet g.Nautilus (g.tec) EEG cap. The data, labels and feedback information were streamed via LabSteamingLayer (LSL) to ensure synchronization between different computers and to allow for parallel processing. The data was band-pass filtered (5-30 Hz) before being streamed to LSL. It was then divided into 50% overlapping 0.5 second long samples before being put through the classifier. Each sample was labelled as one of the 4 classes from 0.5 to 2.5 seconds after a prompt from the PsychoPy experiment to avoid visually evoked potential in the training data. All other data was removed from any training sets.

The feedback for the second stage was generated using the classifier which was initially trained on data recorded without feedback, and manifested itself as a purple bar that changed size based on the certainty that the current data belonged to a given class. This certainty was computed by subtracting the minimum Riemannian distance from each class by the mean of each of these Riemannian distances for each sample, and scaled to improve visualization of the feedback.

$$cert = \frac{\sum_{i=1}^{C} \delta_{Ri}}{C} - min(\delta_{R1...C}) \tag{1}$$

In testing, 20% of each run from this dataset was used for initial training of each classifier, while the remaining 80% was used as the initial test data, though because earlier segments of the test data are continually used to update the classifiers throughout the adaptive session, the amount of test data actually decreased as more updates occurred. To ensure the experiment stopped while enough test data remained, 80 updates were considered for this dataset, with each update using only 2 windowed samples of data.

2) BCI Competition IV Dataset 2a: Datasets from the BCI Competition IV [12] have been commonly used to compare new methods of motor imagery classification, and have come to be regarded as state-of-the-art and benchmark datasets, also used to compare adaptive BCI [4], [9]. In this study, we use dataset 2a, as it is also 4-class motor imagery data. This data was recorded with only 22 EEG channels, and there were more trials and subjects in this dataset. Regardless, all preprocessing was the same in terms of timing, and all channels were used in each dataset. Because more data was recorded for each subject in this dataset, only the first 5% of each subject's data was used for the initial training of all classifiers. This resulted in approximately the same number of training samples as the internally recorded dataset. In addition, 160 updates were considered instead of 80, and each update used 10 samples instead of 2. This made the duration of the adaptive session nearly 10 times longer than our in-house dataset.

B. Adaptive Riemannian Classifier

In this paper we use a Riemannian classifier, which determines the class of a sample based on the Riemannian distance between this sample and the geometric (Riemannian) mean of each of the given classes. Similar methods have been shown to perform at or above the state-of-the-art accuracy on motor imagery data [10], [13], [14].

The method begins by converting each sample of data, consisting of 125 datapoints (0.5 seconds) for each channel, into a Symmetric Positive Definite (SPD) spatial covariance matrix. Training samples within the same class are averaged in Riemannian space using Equation 3, and predictions are made by measuring the Riemannian distance between the current sample and the previously calculated mean covariance matrix. In the below equations, P_1 and P_2 are any two covariance matrices, while P in Equation 3 refers to the predicted Riemannian mean covariance matrix.

$$\delta_R(P_1, P_2) = ||log(P_1^{-1}P_2)||_F = \left[\sum_{i=1}^{n} log^2 \lambda_i\right]^{1/2} \tag{2}$$

$$P_{mean_c}(P_1, ..., P_I) = argmin \sum_{i=1}^{I} \delta_R^2(P, P_i) \tag{3}$$

While the method is relatively simple, being able to adaptively update such a classifier is more difficult, due to the fact that the calculation of a Riemannian mean is not closed form. Three ways of updating the Riemannian classifier in real-time were developed and compared in terms of accuracy and processing speed. The three methods can be briefly described as follows:

1) Continually using incoming data as additional training samples (Retrain Full)
2) Using incoming data as training samples, but removing earlier training samples at the same rate to save on computational complexity (Retrain Window)
3) Approximate the Riemannian class means by using the previously calculated mean as a single sample, but heavily weighting it in future calculations (Mean Estimation)

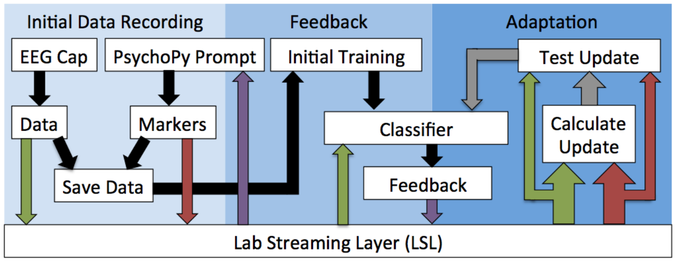

Fig. 2. The proposed framework for data collection and real-time BCI adaptation. Initially, data and markers are simply saved with no feedback. Once the classifier can be trained, feedback is generated by classification of data streamed over LSL. Adaptation is achieved by calculating an update after each trial, using part of the trial for training and the rest to test the update.

For Method 2, a "window size" was determined before processing that was intended to provide enough data for the classifier to be accurate, but would also account for recent environmental changes such as shifts in channel impedance or external noise. The window size was chosen to be the same size as the amount of data that trained the non-adaptive classifier. This was therefore an experiment to see how much recent environmental changes affect classification accuracy, and whether this could be useful for an adaptive BCI.

To compare each method, they were all tested on the all of the data that was remaining in a given session after a segment of new samples had been used to update the classifier. All three methods were also compared against a non-adaptive classifier, using only the training data that was previously defined before the adaptive session began. In addition, it was found that adding geodesic filtering (FGDA) to the data before applying the minimum distance to the Riemannian mean (MDM) classifier had an effect on the classifiers performance, so the results of the non-adaptive classifier with and without FGDA are presented.

The proposed framework for data collection and real-time adaptation is shown in Figure 2. In a single session, some initial data would be recorded without any feedback to provide initial training data for each class. Once each class is expressed, the initial Riemannian means of each class are calculated and feedback starts to be shown to the user. After each trial, the Riemannian means of each class should be updated by recalculating the Riemannian mean of the prompted class, using 75 percent of the new trial data for the update while the remaining 25 percent was added to a changing validation data set. If the accuracy on the validation set does not decrease, then the official classifier will be updated, and subsequent feedback will be shown after passing through the updated classifier.

III. Results and Discussion

The results from the dataset collected in our lab are shown in Figures 3 and 4, while the accuracies of the BCI Competition dataset are shown in Figure 5.

On the in-house dataset, the fully retrained and estimated mean adaptive BCI showed the best overall performance, increasing from about 31% to around 43% throughout a given

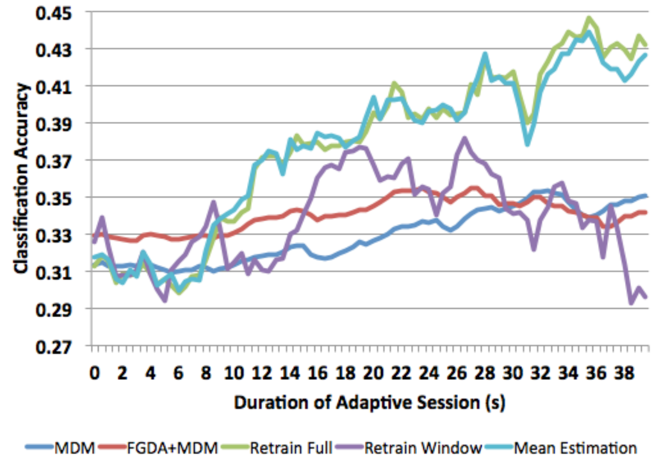

Fig. 3. The average accuracy of each type of Adaptive Riemannian BCI compared against non-adaptive classifiers (MDM and FGDA+MDM) on our in-house dataset. The Retrain Full and Mean Estimation methods improved their accuracy, and showed

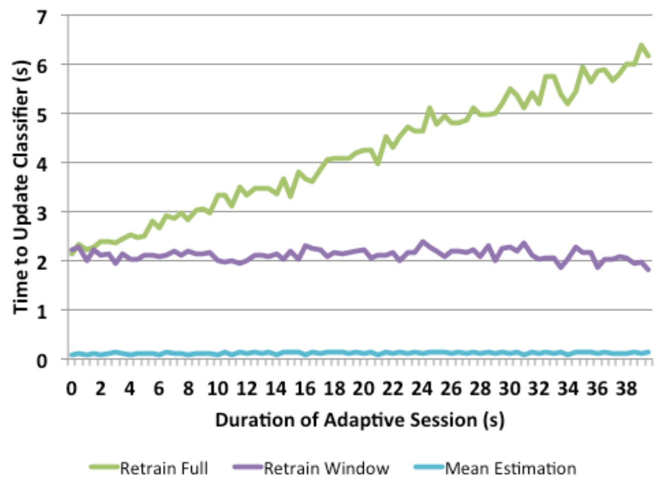

Fig. 4. The average time taken to update each type of Adaptive Riemannian BCI on our in-house dataset. The update time of the Retrain Full adaptation method increased linearly with the amount of data used, while the medium and fast methods stayed consistently lower.

session, on average. These updating classifiers achieved an overall average accuracy 3.7% higher than the non-adaptive classifiers with geodesic filtering, though toward the end of the adaptive session, this difference improved to 8%. Retraining the classifier with the most recent window of samples did not show any major improvement over the non-adaptive classifiers.

As expected, the amount of time to update the fully retrained adaptive classifier increased linearly with the amount of data it was using in its calculations. This led to unacceptable update times of more than 6 seconds on our in-house dataset, and up to more than 18 seconds with the larger BCI Competition dataset. The medium and fast adaptive BCI achieved relatively constant averages of approximately 1.5 ($\pm$0.2) and 0.12 ($\pm$0.01) seconds, respectively.

On the BCI Competition dataset, the slow and fast methods

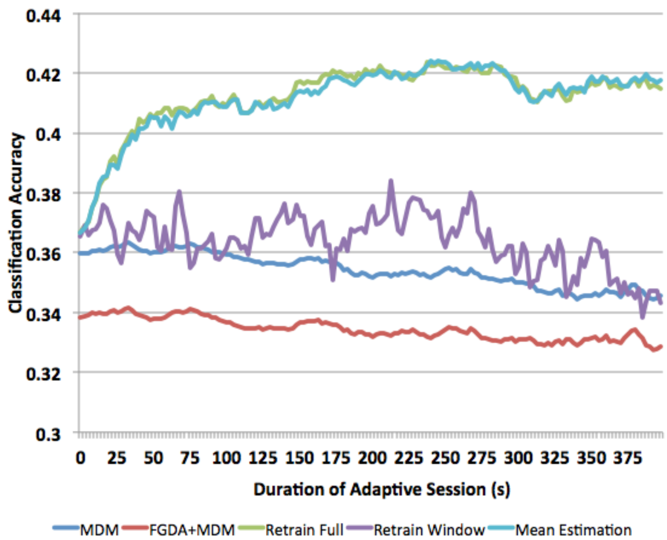

Fig. 5. The average performance of each type of Adaptive Riemannian BCI on BCI Competition IV dataset 2a. The fully retrained and estimated mean adaptive BCI achieved an average of 5.8% better than the best non-adaptive BCI, and improved over time.

of adapting the classifier also showed similar performance in terms of accuracy, achieving an average accuracy approximately 5.8($\pm$2.4)% higher for a given sample than using the MDM algorithm. The medium adaptive classifier was not as successful, with only about 1% improvement on the non-adaptive classifier. In addition, the medium adaptive classifier was volatile and unreliable, indicating that using only recent data to train a classifier does not appreciably improve the classifier's performance. You may notice that the presented accuracy is lower than what has been reported by other research [10], [14], but this is due to the fact that the window sizes we used in this study are smaller, providing less information at a given time for the classifier to make a decision.

On both datasets, the fully retrained and estimated mean adaptive Riemannian BCI clearly improved over time, while the performance of the non-adaptive classifiers and the medium adaptive classifier stayed within a few percentage points of their initial accuracy. In addition, the similarity between the fully retrained and mean estimated adaptations on both datasets validate our method of estimating the mean for real-time adaptation of a BCI that uses Riemannian geometry.

We should also note that on both datasets, the success of the adaptive BCI was most notable in subjects who had better overall performance. This indicates that BCI illiteracy would still be a problem when using adaptive BCI.

IV. Conclusion

In this study, three methods of adapting a Riemannian BCI were developed and evaluated in terms of feasibility for real-time application. The fastest and least computationally expensive method of estimating the Riemannian means was shown to be among the most accurate methods, outperforming the non-adaptive classifier and showing promise for future research into adaptive Riemannian BCI. The other two methods either took too much computation time or did not enhance the classification accuracy as desired.

Future work should include other types of adaptation, such as real-time channel or frequency band selection. In addition, the adaptive Riemannian BCI should be implemented in training protocols with more complex feedback for easier transition to assitive robotic control. Lastly, researchers should investigate the combination of Riemannian geometry with other methods of MI classification, such as CSP or deep learning, which have also shown good results for this application.

Acknowledgments

The authors would like to thank Shamas Khan for help in setting up experiments.

References

[1] N. Birbaumer, "Breaking the silence: Brain-computer interfaces (BCI) for communication and motor control," *Psychophysiology*, vol. 43, no. 6, pp. 517–532, 2006.

[2] J. Meng, S. Zhang, A. Bekyo, J. Olsoe, B. Baxter, and B. He, "Noninvasive Electroencephalogram Based Control of a Robotic Arm for Reach and Grasp Tasks," *Nature Scientific Reports*, vol. 6, no. 1, pp. 1–15, 2016.

[3] C. Vidaurre, A. Schloogl, R. Cabeza, R. Scherer, and G. Pfurtscheller, "A Fully On-Line Adaptive BCI," vol. 53, no. 6, pp. 1049–1050, 2005.

[4] H. Raza, H. Cecotti, and G. Prasad, "Optimising frequency band selection with forward-addition and backward-elimination algorithms in EEG-based brain-computer interfaces," *Proceedings of the International Joint Conference on Neural Networks*, no. September, 2015.

[5] H. Raza, H. Cecotti, Y. Li, and G. Prasad, "Adaptive learning with covariate shift-detection for motor imagery-based brain-computer interface," *Soft Computing*, vol. 20, no. 8, pp. 3085–3096, 2016.

[6] H. Raza, D. Rathee, S. Zhou, H. Cecotti, and G. Prasad, "Covariate Shift Estimation based Adaptive Ensemble Learning for Handling Non-Stationarity in Motor Imagery related EEG-based Brain-Computer Interface," 2018. [Online]. Available: http://arxiv.org/abs/1805.01044

[7] V. Mondini, A. L. Mangia, and A. Cappello, "EEG-Based BCI System Using Adaptive Features Extraction and Classification Procedures," *Computational Intelligence and Neuroscience*, 2016.

[8] C. Sannelli, C. Vidaurre, K. R. Müller, and B. Blankertz, "Ensembles of adaptive spatial filters increase BCI performance: An online evaluation," *Journal of Neural Engineering*, vol. 13, no. 4, 2016.

[9] H.-J. Rong, C. Li, R.-J. Bao, and B. Chen, "Incremental Adaptive EEG Classification of Motor Imagery-based BCI," *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8489283/

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

[11] J. W. Peirce, "Generating stimuli for neuroscience using PsychoPy," *Frontiers in Neuroinformatics*, vol. 2, no. January, pp. 1–8, 2008. [Online]. Available: http://journal.frontiersin.org/article/10.3389/neuro.11.010.2008/abstract

[12] M. Tangermann, K. R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K. J. Miller, G. R. Müller-Putz, G. Nolte, G. Pfurtscheller, H. Preissl, G. Schalk, A. Schlögl, C. Vidaurre, S. Waldert, and B. Blankertz, "Review of the BCI competition IV," *Frontiers in Neuroscience*, vol. 6, pp. 1–31, 2012.

[13] 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.

[14] D. Freer, Y. Ma, and G.-Z. Yang, "Toward Real-time BCI Control of Assistive Robots: A Comparison of State-of-the-Art Methods," in *Hamlyn Symposium on Medical Robotics*, 2018, pp. 55–56.