On Transfer Learning for Naive Brain Computer Interface Users

Ruofan Liu[1,*], Satyam Kumar[1,*], Hussein Alawieh[1], Evan Carnahan[1] and José del R. Millán[1,2]

***Abstract*— Motor Imagery (MI) based Brain-Computer Interfaces (BCI) typically require the collection of subject-specific calibration data to build a classifier of motor intent. The BCI users are then trained over multiple online sessions with real-time feedback using the calibrated decoder to acquire MI skills. The subject-specific calibration session is thought to be necessary for accurate MI decoding due to the wide variability in electroencephalogram (EEG) signals across the population. The process of acquiring calibration data is long and tedious and includes training individualized decoding models for each subject. Transfer Learning setups can help circumvent this individualized calibration and decoder training phase by using data acquired from previous subjects. This paper first proposes a geometry-aware deep learning architecture that exploits the spatial similarity of MI neural activity between BCI users. We show the efficacy of the proposed approach by classifying the motor intentions of 18 naïve BCI subjects. In a subject-specific setting, our proposed method significantly outperforms classical decoding algorithms. Next, we train the proposed network and skip the subject-specific calibration data to mimic a transfer learning setting. We show that our model architecture achieves similar performance to subject-specific decoders in the transfer learning setting. This finding opens the door to robust BCIs that are readily transferable across subjects without the need for subject-specific calibration and individualized decoding models.**

***Clinical relevance*— Efficacious Brain-computer Interfaces may help stroke and spinal cord injury patients in rehabilitation and to control assistive devices.**

I. INTRODUCTION

Brain-computer interfaces (BCIs) enable communication between the brain and an external device without any involvement from the peripheral nervous system. BCIs have a range of applications from neurorehabilitation [1] to robotics [2] to interactive and assistive devices [3]. They offer a gateway to users' intentions by decoding their brainwaves from electroencephalography (EEG). Such non-invasive EEG-based BCIs are commonly controlled using Motor Imagery (MI) —the act of imagining the movement of a limb without actually executing the movement [4]. In a typical MI-BCI setup, subjects first perform a calibration session in which their EEG is collected while they perform different MI tasks in an open-loop setting; i.e., without receiving feedback on their brain activity. The calibration data is used to build a decoder that predicts the subject's motor intent in real-time. The decoder is used to provide feedback to subjects in online closed-loop BCI training sessions to help them learn proper control of the BCI [3].

*These authors contributed equally.

[1]Department of Electrical and Computer Engineering, [2]Department of Neurology, The University of Texas at Austin, Austin, TX, USA. E-mail: {rliu17, satyam.kumar}@utexas.edu, jose.millan@austin.utexas.edu

The ability of the subjects to learn to control the BCI is dependent on the consistency of the feedback they receive, and thus on the classification accuracy of the BCI decoder. The latter is affected by the quality of the data collected during the offline calibration period. However, the lack of feedback during calibration presents a challenge since naïve subjects may fail to generate distinctive neural signals for accurate distinction between MI classes in real time. Furthermore, collecting calibration data and training a model for each subject is time-consuming. A potential alternative to subject-specific data collection is to exploit existing data from other subjects to train a generalised decoding model. This technique, called *transfer learning*, has been successful in machine learning applications in which data from apriori known domains is used to build a generalised model for unseen domains [5]. Transfer learning could play a pivotal role in MI-BCIs, as it would allow subjects to immediately use decoders built on previously recorded data This can also eliminate the risk of having a biased subject-specific decoder for naïve BCI users who generate confusing neural patterns during open-loop calibration.

A major challenge in building accurate BCI decoders is the low signal to noise ratio (SNR) of EEG signals. Typically temporal and spatial filtering is performed to increase the SNR. Among many spatial filtering techniques, Common Spatial Patterns (CSP) has emerged as a highly effective method that learns a set of optimal spatial filters [6] over a wide-band. It has also been modified to capture narrow-band frequencies that might correspond to subject-specific modulations: the so-called Filter Bank Common Spatial Pattern (FBCSP) [7]. However, one of the major limitations of this approach is the prior specification of the narrow bands to use. In subsequent efforts, Lawhern et al. [8] proposed EEGNet, a data-driven deep learning method that learns temporal and spatial filters and showed improved classification performances over prior methods.

Other emerging decoding techniques are based on Riemannian geometry, which have shown superior results in several online BCI classification competitions [9]. These methods have been shown to only require small amounts of data, avoid spatial filtering, and be robust against non-stationarities. Barachant et al. [10] proposed the Minimum Distance to Riemannian Mean (MDM) classifier, a simple distance-based classifier that exploits the spatial covariance matrix of EEG signals. MDM-based techniques are shallow learning approaches, while there are deep learning approaches that respect the structure of the Riemannian manifold [11], [12].

In this paper, we first propose a deep network on the Riemannian manifold for MI classification and show its

efficacy as compared to the previously proposed state-of-the-art deep network EEGNet and classical MI decoding algorithms. Next, we experiment with multiple transfer learning schemes and show that transfer learning frameworks have similar performance to that of subject-specific decoding for different models. Thus, we recommend to skip the ubiquitous subject-specific calibration phase in favor of a generic MI classifier.Lastly, we discuss the future of transfer learning in BCIs, and how it may be able to be improved through the use of domain adaptation.

II. Methods

1) Common Spatial Pattern: CSP is one of the most popular algorithms used for spatial filtering in the MI classification pipeline. In a binary classification setting, CSP estimates spatial filters that maximize the ratio of across-class variances for the transformed signals [6]. Let $X \in \mathbb{R}^{N_t \times N_c}$ be a band passed filtered trial of EEG, where N_t is the number of temporal samples in the trial and N_c is the number of EEG channels. The sample wise covariance can be estimated by

$$C = \frac{XX^T}{N_t} \tag{1}$$

Customarily, CSP uses the Euclidean mean of the covariances across trials of the two classes [6].

Let $W^{CSP} \in \mathbb{R}^{m \times N_c}$ (m: number of spatial filters) be the subset of spatial filters, then the transformed signal after CSP filtering is

$$X_i^{CSP}(t) = W^T X_i(t) \tag{2}$$

Next, the features for each trial i can be extracted as

$$\mathbf{f}_i^n = \log\left(\frac{var(\mathbf{X_i^{CSP}}^n)}{\sum_{n=1}^{d} var(\mathbf{X_i^{CSP}}^n)}\right) \tag{3}$$

where $\mathbf{X_i^{CSP}}^n$ is the n^{th} spatially filtered channel and $n \in 1 \cdots m$ (number of selected spatial filters). Once the feature vectors for all samples are estimated, they are used to train a Linear Discriminant Analysis (LDA) classifier. In the evaluation phase, an incoming EEG sample is first transformed to the surrogate space using the selected m spatial filters. Then, its features are extracted using (3) and classified by LDA.

2) EEGNet: EEGNet is a convolutional neural network designed for the classification of EEG signals. It consists of two main blocks:

- Block 1: The first layer includes F_1 2D convolutions with length K_e. Afterwards, D depthwise 2D convolutions are applied to each feature. This structure is reminiscent of that of FBCSP [7], where spatial filters are learned for each filter-band.
- Block 2: The second block features a separable convolution which creates compressed feature maps and combines them optimally for classification.

3) Riemannian MDM: A Minimum Distance to Riemannian Mean (MDM) classifier is a simple distance-based classifier on the Riemannian manifold that embeds symmetric positive definite (SPD) covariance matrices [9]. To build the MDM classifier, we use sample covariances of each trial using (1) to estimate the class prototypes $\bar{C}_k$ corresponding to each MI class i. Note that the class prototypes are estimated using the Riemannian distances, which is defined using an affine invariant Riemannian metric [10]. In the testing phase, for an EEG trial i, we first estimate the sample covariance C_i and then estimate the classification probability to each of the binary classes

$$p_k = \frac{e^{-(\delta_r(C_i, \bar{C}_k))^2}}{e^{-(\delta_r(C_i, \bar{C}_1))^2} + e^{-(\delta_r(C_i, \bar{C}_2))^2}} \quad k \in [1, 2] \tag{4}$$

where p_k is classification probability and $\delta_r(C_i, \bar{C}_k)$ is the affine invariant Riemannian distance between the i^{th} trial and the corresponding k^{th} class prototype.

4) RBNNet: RBNNet is a deep neural network architecture that acts upon the SPD manifold by preserving the SPD structure across the layers. RBNNet is a combination of SPDNet [11] and Riemannian Batch Normalization [12] and consists of four main layers :

- Bilinear Mapping (BiMap): The Bilinear Mapping layer transforms input SPD matrices: $C^o = WC^iW^T$, where W is the transformation matrix, C^o and C^i are the output and input of the BiMap layer respectively . W is a row full-rank matrix to ensure that the output C^i maintains its SPD structure.
- Eigenvalue Rectification (ReEig): The ReEig layer seeks to introduce nonlinearity into the Riemannian Manifold in a fashion similar to ReLu in classical deep nets. Eigenvalues smaller than the threshold are rectified: $C^o = U \max(\epsilon I, \Sigma) U^T$, where U and Σ are the eigenvectors and eigenvalues obtained by eigendecomposition of the C_i matrix respectively, I is the identity matrix and ϵ is the rectification threshold.
- Eigenvalue Logarithm (LogEig): The LogEig layer reduces the manifold to its flat tangent space which allows for the use of Euclidean computations. It does so by taking the logarithm of input SPD matrices using eigendecomposition: $X = U \log(\Sigma) U^T$, where U and Σ are the eigenvectors and eigenvalues respectively of the input matrix C^i to the LogEig layer. X is then flattened to perform euclidean manipulations.
- Riemannian Batch Normalization (RBN): The Riemannian Batch Normalization layer replicates the classical technique of batch normalization on the SPD manifold. However, RBN only normalizes the mean, as opposed to both the mean and the variance in classical Batch Normalization [12].

III. Evaluation

A. Dataset

The experimental data were collected from 18 naïve subjects who imagined moving either their right hand or their left

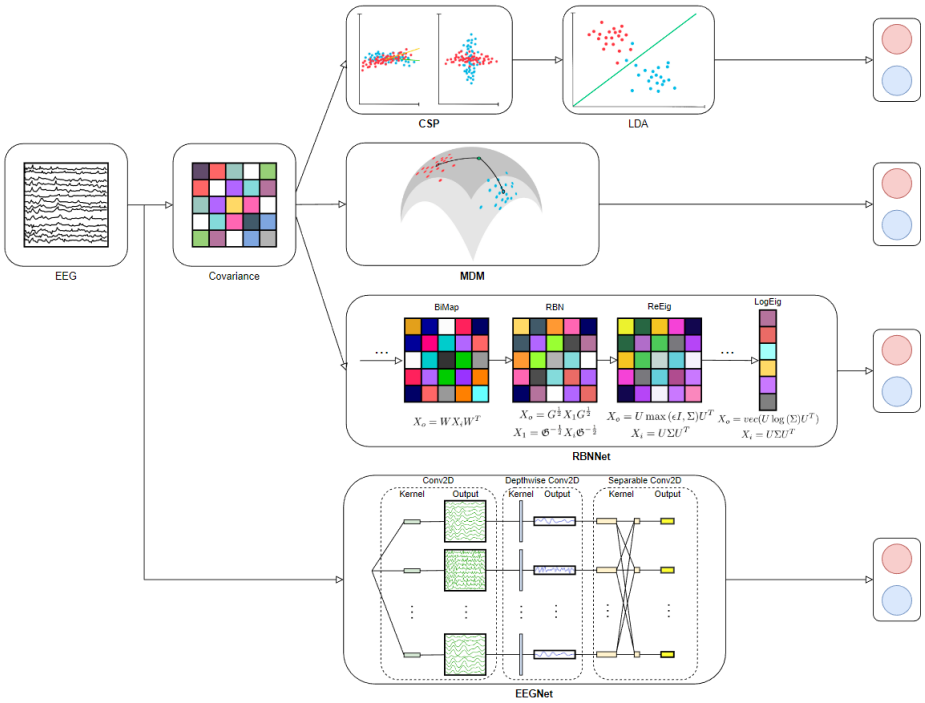

Fig. 1. Pictorial summary of the different methods: Common Spatial Patterns (CSP), Minimum Distance to Mean Classifier (MDM), EEGNet, RBNnet classifiers.

hand. The 22 electroencephalogram (EEG) electrodes used for data acquisition were: F_7, F_3, F_z, F_4, F_8, FC_5, FC_1, FC_2, FC_6, C_3, C_z, C_4, CP_5, CP_1, CP_2, CP_6, P_7, P_3, P_z, P_4, P_8, POz. All subjects performed an offline calibration session followed by an online session on a different day. The sessions consisted of 4 runs. We used a classical bar task [3] for providing feedback to subjects. Each run consisted of 20 trials, 10 for each class. Each trial began with a rest period, followed by a period of fixation on the center of the screen, and the presentation of a cue that prompted the user to perform MI for one of the classes.

B. Preprocessing

For each of the above classifiers we bandpassed the EEG signals to the mu and beta wide band ([8 30] Hz) using a 3rd order Butterworth filter. To mimic online classification, we used one-second sliding windows with 62.5 ms of step size over the MI period. We used shrinkage for estimating the covariance matrices [13] as it has shown to result in better classification performances for BCI [14].

C. Network Parameters

1) EEGNet: We set F_1 and D as 8 and 2, respectively, as the original authors show that this configuration performs well in MI classification. The kernel length K_e is changed to 256 due to the increased sampling rate of our data. Some dimensions for separable convolution were also changed to reflect our sampling rate.

2) RBNNet: The architecture of the proposed RBNNet is { {BiMap-RBN-ReEig} – {BiMap-RBN-ReEig} – {BiMap-RBN} – {LogEig} – Output}. Each BiMap operation operated on a $\mathbb{R}^{22\times22}$ dimensional SPD manifold. We did not choose to reduce the dimensions below 22 as it resulted in lower performances. The eigenvalue rectification threshold ϵ was determined by examining the eigenvalues of the calibration data of nine subjects. In a fashion similar to Principal Component Analysis, we evaluated the eigenvalues for the covariance representation of every trial for nine subjects, and we determined a threshold at which 99.5% of the variance was retained. Lastly, we averaged out the thresholds to determine the ϵ that was used in all the RBNNet models.

TABLE I
INFERENCE TIME PER TRIAL AND NUMBER OF PARAMETERS USED FOR TRAINING THE DECODING MODEL

Method	CSP	MDM	EEGNet	RBNNet
# parameters	-	-	$\sim$ 3500	$\sim$ 3800
Inference time (s)	$\sim$ 0.00005	$\sim$ 0.001	$\sim$ 0.005	$\sim$ 0.007

D. Evaluation Frameworks

We used the following three different frameworks for training the different classifiers discussed above

- Subject-Specific (`SS`): We used the subject-specific calibration session to build the decoders.
- Leave One Subject Out Transfer (`LOSOT`): To exploit the data of all the available subjects, we used the offline

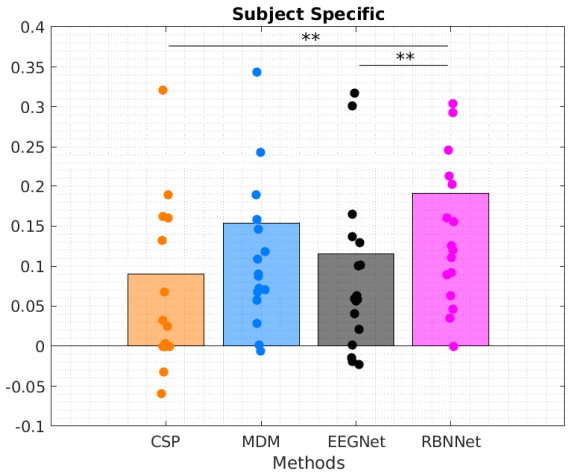

Fig. 2. Bar plot of the average kappa value across subjects for the subject-specific models. Each marker corresponds to the kappa value of a subject. **: $p < 0.01$.

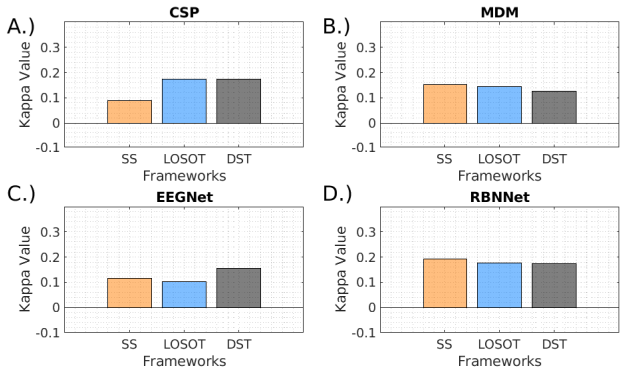

Fig. 3. Bar plot of the average kappa value across subjects for A) CSP, B) MDM, C) EEGNet, and D) RBNnet classifiers across the different evaluation frameworks.

calibration data of all the subjects except the subject for whom we performed the classification performance evaluation. This framework mimics the classical transfer learning scenario.

- Discriminant Subject Transfer (`DST`): The data quality for training deep learning models has shown to be of paramount importance. We hypothesize that many of the naïve subjects might have large overlapping feature spaces between the two MI classes, which could lead to unstable training. In this `DST` framework, we select the nine best subjects out of 18 based on the Riemannian distance between the class prototypes of their offline data. Using (4), we estimated the probability of classification of each prototype, and if $p_k > 0.6$, the subject's data is included in the decoder training pool. The resulting models are evaluated for the subjects whose data was not selected for training. For the subjects whose data was selected for building the decoder, we employed the `LOSOT` strategy, and trained the classifiers using the other eight best subjects to perform evaluation.

All of the classification pipelines benchmarked in this paper consist of a relatively low number of learnable parameters, especially the deep learning (DL) architectures (Table I) compared to classical DL architectures for EEG [8]. These lightweight models allow for fast decoding (inference time $< 0.01s$), which is crucial for providing real-time feedback in closed-loop BCI settings. All the models were evaluated on an AMD Ryzen 7 5700U with Radeon Graphics and 16GB RAM. We predominantly used the pyriemann package[1], Tensorflow implementation of Lawhern et al. [8][2] and PyTorch implementation provided in Brooks et al. [12] to train our models and perform the evaluations.

[1]https://github.com/pyRiemann/pyRiemann
[2]https://github.com/vlawhern/arl-eegmodels

IV. RESULTS AND DISCUSSION

Figure 2 shows the classification kappa value averaged across subjects when different classifiers were built on the subject-specific calibration data. Friedman's analysis of variance (ANOVA) shows a significant main effect of the method on classification performance ($p = 0.0074$). Post hoc pairwise comparison shows that the proposed RBNnet(0.19 ± 0.17) outperforms both CSP (0.09 ± 0.16) and EEGNet (0.11 ± 0.15), and the improvement is statistically significant ($p < 0.01$). However, there was no statistical difference between MDM and RBNnet, although the mean of kappa value across subjects is higher in RBNnet compared to MDM (0.15 ± 0.14).

One-factor repeated measures ANOVA or Friedman's ANOVA (used when the normality criteria is not met) show no statistically significant effects of frameworks (Fig 3). In other words, using (`SS`) does not lead to significant improvements in classification compared to a general classifier trained on some (`DST`) or all subjects (`LOSOT`). Moreover, a repeated measures ANOVA reveals no statistical significance between any of the transfer learning frameworks.

Our proposed model (RBNNet) significantly outperformed CSP and EEGNet in subject specific evaluations (Fig 2), which aligns with previous findings in the literature showing that Riemannian approaches have better performance compared to classical approaches for MI classification especially in data scarce settings [15]. Although it is not significantly better than MDM, the mean performance of the proposed RBNNet was still higher than MDM. This result suggests that introducing nonlinearity in Riemannian classification approaches could increase classification performance of subject-specific BCI decoders.

The results in Fig 3 show statistically insignificant differences between the subject-specific and the transfer learning frameworks. These results corroborate that transfer learning frameworks could be used in BCIs without having compromising accuracy. We observe that using data from other subjects improves CSP performance, suggesting that CSP may

have learned robust spatial filters when the training dataset size increased. For EEGNet, the DST framework shows a trend towards statistically significantly better performance ($p = 0.07$) compared to the LOSOT framework. This can indicate that EEGNet is more sensitive to the quality of the training data. In contrast, for the proposed RBNNet architecture, there is no statistical difference between DST and LOSOT framework, which supports that RBNNet is robust to overlapping feature spaces of naïve subjects that are included in the training pool. Moreover, we notice that the average classification performance across subjects for the proposed RBNNet (DST:0.1737 ± 0.1992 LOSOT:0.1754 ± 0.1946) is better than EEGNet (DST: 0.1553 ± 0.1733 LOSOT: 0.1015 ± 0.1150) in both frameworks. This further highlights the potential advantage of using Riemannian geometry for decoding MI intents.

Finally, the accuracy of the transfer learning frameworks for different classifiers was relatively similar. Although RBNNet consistently performed well compared to the other classifiers in the transfer learning settings, the results are subject and framework dependent. Recent studies point out that long-term subject training is crucial in building robust MI-based BCI systems, including people with severe disabilities [16]–[18]. Thus, we could use any of the proposed transfer learning frameworks with the benchmarked classifiers, specifically RBNnet, to skip the calibration session and to immediately provide appropriate feedback over the course of their closed-loop BCI training to help them gain better control. Moreover, to tackle the non-stationarities between sessions over multiple days, we can use incremental domain adaptation for Riemannian techniques as it has recently showed promising results [19], [20]. In our future work, we plan to incorporate recentering strategies for RBNnet architectures for MI signal classification.

V. CONCLUSION

In this paper, we proposed a deep learning based architecture for MI BCIs. The proposed method outperforms the studied benchmarks on an in-lab recorded MI dataset with 18 naïve BCI subjects. Next, we experimented with different transfer learning strategies and showed that transfer learning could be used in place of subject-specific decoders for MI classification; thus allowing for skipping the tedious subject specific calibration. Finally, we propose the use of transfer learning models in a closed-loop BCI training that fosters subjects' MI skill acquisition.

REFERENCES

[1] A. Biasiucci, R. Leeb, I. Iturrate, S. Perdikis, A. Al-Khodairy, T. Corbet, A. Schnider, T. Schmidlin, H. Zhang, M. Bassolino, D. Viceic, P. Vuadens, A. G. Guggisberg, and J. d. R. Millán. Brain-actuated functional electrical stimulation elicits lasting arm motor recovery after stroke. *Nature Communications*, 9(1):1–13, 2018.

[2] L. Tonin and J. d. R. Millán. Noninvasive brain-machine interfaces for robotic devices. *Annual Review of Control, Robotics, and Autonomous Systems*, 4:191–214, 2021.

[3] R. Leeb, S. Perdikis, L. Tonin, A. Biasiucci, M. Tavella, M. Creatura, A. Molina, A. Al-Khodairy, T. Carlson, and J. d. R. Millán. Transferring brain-computer interfaces beyond the laboratory: Successful application control for motor-disabled users. *Artificial Intelligence in Medicine*, 59(2):121–132, 2013.

[4] J. d. R. Millán, R. Rupp, G. Mueller-Putz, R. Murray-Smith, C. Giugliemma, M. Tangermann, C. Vidaurre, F. Cincotti, A. Kubler, R. Leeb, et al. Combining brain-computer interfaces and assistive technologies: State-of-the-art and challenges. *Frontiers in Neuroscience*, page 161, 2010.

[5] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2009.

[6] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and Klaus-Robert Muller. Optimizing spatial filters for robust EEG single-trial analysis. *IEEE Signal Processing Magazine*, 25(1):41–56, 2007.

[7] K. K. Ang, Z. Y. Chin, H. Zhang, and C. Guan. Filter bank common spatial pattern (FBCSP) in brain-computer interface. In *2008 IEEE International Joint Conference on Neural Networks*, pages 2390–2397, 2008.

[8] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance. EEGNet: A compact convolutional neural network for EEG-based brain-computer interfaces. *Journal of Neural Engineering*, 15(5):056013, 2018.

[9] F. Yger, M. Berar, and F. Lotte. Riemannian approaches in brain-computer interfaces: A review. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1753–1762, 2016.

[10] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten. Multiclass brain-computer interface classification by Riemannian geometry. *IEEE Transactions on Biomedical Engineering*, 59(4):920–928, 2011.

[11] Z. Huang and L. Van Gool. A Riemannian network for SPD matrix learning. In *31st AAAI Conference on Artificial Intelligence*, 2017.

[12] D. Brooks, O. Schwander, F. Barbaresco, J.-Y. Schneider, and M. Cord. Riemannian batch normalization for SPD neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

[13] O. Ledoit and M. Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, 2004.

[14] B. Blankertz, S. Lemm, M. Treder, S. Haufe, and K.-R. Müller. Single-trial analysis and classification of ERP components—A tutorial. *NeuroImage*, 56(2):814–825, 2011.

[15] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, and F. Yger. A review of classification algorithms for EEG-based brain-computer interfaces: A 10 year update. *Journal of Neural Engineering*, 15(3):031005, 2018.

[16] S. Perdikis, L. Tonin, S. Saeedi, C. Schneider, and J. d. R. Millán. The Cybathlon BCI race: Successful longitudinal mutual learning with two tetraplegic users. *PLoS Biology*, 16(5):e2003787, 2018.

[17] S. Perdikis and J. d. R. Millán. Brain-machine interfaces: A tale of two learners. *IEEE Systems, Man, and Cybernetics Magazine*, 6(3):12–19, 2020.

[18] L. Tonin, S. Perdikis, T. D. Kuzu, J. Pardo, B. Orset, K. Lee, M. Aach, T. A. Schildhauer, R. Martínez-Olivera, and J. d. R. Millán. Learning to control a BMI-driven wheelchair for people with severe tetraplegia. *iScience*, https://doi.org/10.1016/j.isci.2022.105418, 2022.

[19] S. Kumar, F. Yger, and F. Lotte. Towards adaptive classification using Riemannian geometry approaches in brain-computer interfaces. In *7th International Winter Conference on Brain-Computer Interface*, pages 1–6, 2019.

[20] C. Benaroch, K. Sadatnejad, A. Roc, A. Appriou, T. Monseigne, S. Pramij, J. Mladenovic, L. Pillette, C. Jeunet, and F. Lotte. Long-term BCI training of a tetraplegic user: Adaptive Riemannian classifiers and user training. *Frontiers in Human Neuroscience*, 15:635653, 2021.