Neural Networks 136 (2021) 1–10

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Adaptive transfer learning for EEG motor imagery classification with deep Convolutional Neural Network

Kaishuo Zhang [a], Neethu Robinson [a], Seong-Whan Lee [b], Cuntai Guan [a,*]

[a] School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798, Singapore
[b] Department of Artificial Intelligence, Korea University, Seoul 02841, South Korea

ARTICLE INFO

Article history:
Received 5 June 2020
Received in revised form 3 November 2020
Accepted 13 December 2020
Available online 23 December 2020

Keywords:
Transfer learning
Brain–computer interface (BCI)
Electroencephalography (EEG)
Convolutional Neural Network (CNN)

ABSTRACT

In recent years, deep learning has emerged as a powerful tool for developing Brain–Computer Interface (BCI) systems. However, for deep learning models trained entirely on the data from a specific individual, the performance increase has only been marginal owing to the limited availability of subject-specific data. To overcome this, many transfer-based approaches have been proposed, in which deep networks are trained using pre-existing data from other subjects and evaluated on new target subjects. This mode of transfer learning however faces the challenge of substantial inter-subject variability in brain data. Addressing this, in this paper, we propose 5 schemes for adaptation of a deep convolutional neural network (CNN) based electroencephalography (EEG)-BCI system for decoding hand motor imagery (MI). Each scheme fine-tunes an extensively trained, pre-trained model and adapt it to enhance the evaluation performance on a target subject. We report the highest subject-independent performance with an average ($N = 54$) accuracy of 84.19% ($\pm$9.98%) for two-class motor imagery, while the best accuracy on this dataset is 74.15% ($\pm$15.83%) in the literature. Further, we obtain a statistically significant improvement ($p = 0.005$) in classification using the proposed adaptation schemes compared to the baseline subject-independent model.

1. Introduction

Deep learning has emerged as a prevalent methodology in machine learning in recent years, leading to significant breakthroughs in computer vision and speech recognition (LeCun, Bengio, & Hinton, 2015). The learning capacity of deep neural networks stems in part from its ability to discover intricate feature representations from raw data. This has inspired a growing interest among neuro-engineering researchers to apply deep learning to the development of Brain–Computer Interface (BCI) systems because it largely alleviates the need for manual feature extraction as seen in conventional BCI, which requires domain-specific expertise in the signal (Zhang et al., 2019).

Electroencephalography (EEG) is a noninvasive brain data acquisition modality widely used in BCI research. Numerous studies have shown correlations between EEG signals and actual or imagined movements and between EEG signals and mental tasks (Wolpaw, Birbaumer, McFarland, Pfurtscheller, & Vaughan, 2002). Motor Imagery (MI) is one of these paradigms in which a mental rehearsal of movement is performed. This will typically elicit a decrease in mu and beta rhythms (event-related desynchronization, ERD) contralateral to the movement (Pfurtscheller & Da Silva, 1999). The termination of the movement imagination is usually followed by an increase in beta rhythm (event-related synchronization, ERS) over the ipsilateral side of the brain Pfurtscheller, Stancak, and Edlinger (1997).

The conventional BCI classification uses discriminative features that represent ERD/ERS to classify MI. The state-of-the-art BCI classification algorithm, filter-bank common spatial patterns (FBCSP) (Ang, Chin, Zhang, & Guan, 2008), finds a set of linear projection (i.e. spatial filtering) that maximizes the differences in the variance of the multiple classes of EEG measurements using temporally filtered signals with different frequency bands. This is followed by feature selection and classification.

In conventional BCI classification, the models are trained and evaluated on the same subject's data. However, the distribution of the features may shift due to the covariate shift of data distribution in the training phase and evaluation phase of a subject (Blankertz, Tomioka, Lemm, Kawanabe, & Muller, 2007). To address this, in Arvaneh, Guan, Ang, and Quek (2013), a supervised and unsupervised EEG data space adaptation algorithm is proposed using Kullback–Leibler (KL) divergence criterion. In Liyanage et al. (2013), a dynamically weighted ensemble

* Corresponding author.
E-mail addresses: kzhang015@e.ntu.edu.sg (K. Zhang), nrobinson@ntu.edu.sg (N. Robinson), sw.lee@korea.ac.kr (S.-W. Lee), ctguan@ntu.edu.sg (C. Guan).
URL: https://personal.ntu.edu.sg/ctguan/ (C. Guan).

https://doi.org/10.1016/j.neunet.2020.12.013
0893-6080/

classification (DWEC) framework is presented based on the distance in the clustered features across sessions. In Raza, Cecotti, Li, and Prasad (2016), a transductive and adaptive learning method with Covariate Shift-Detection (CSD) is proposed to detect the covariate shifts in the data in real-time and initiate adaptive corrective action.

In recent years, several novel deep learning approaches have been proposed for EEG-based BCI (Craik, He, & Contreras-Vidal, 2019). Many of which are based on convolutional neural networks (CNN). In Schirrmeister et al. (2017), a deep CNN architecture is reported, which uses a combination of temporal and spatial convolution filters for their first convolution-pooling block and introduced three more convolution-pooling blocks to further reduce the dimensionalities before feeding the output to a fully-connected layer. The shallow architecture EEGNet, proposed in Lawhern et al. (2018), uses separable convolution in place of convolution-pooling blocks to reduce the number of parameters and decouple the relationship within and across feature maps. In Robinson, Lee, and Guan (2019), a multi-band, multi-channel EEG input representation to the deep CNN is used and further increased the accuracy.

However, deep neural networks have a massive amount of parameters to train compared to the aforementioned classical model. For example, the deep CNN in Schirrmeister et al. (2017) has as many as 305,077 trainable parameters for the binary classification of MI. This would require a large amount of data and time for training. While there are many publicly available BCI datasets (Goldberger et al., 2000; Lee et al., 2019; Tangermann et al., 2012), the amount of data available for a single subject is usually small. Moreover, collecting an extensive amount of data for a new subject is time-consuming, which may in turn mentally exhaust the subject during a prolonged recording session and affect the quality of the data. To overcome this lack of subject-specific data, there have been some transfer-based approaches proposed, using pre-existing data from other subjects. Nevertheless, it is challenging to transfer the knowledge learned from other subjects due to substantial inter-subject variabilities (Wronkiewicz, Larson, & Lee, 2015). Therefore, adaptation is needed to fine-tune the model for the target subject

In Fahimi et al. (2019) various EEG representations, including raw, band-pass filtered, and multi-band signals, are studied for inter-subject transfer learning. In Sakhavi and Guan (2017), Sakhavi, Guan, and Yan (2018), the author explored FBCSP (Ang et al., 2008) based representation of the EEG data and, and utilized knowledge distillation techniques with a combination of hard labels and soft predictions to fine-tune a deep CNN model. A subject-independent framework based on deep CNN is reported in Kwon, Lee, Guan, and Lee (2019) using spectral–spatial input generation, which significantly outperforms the conventional subject-dependent approaches. An online pre-alignment strategy based on Riemannian Procrustes Analysis (RPA) (Rodrigues, Jutten, & Congedo, 2018) is proposed in Xu et al. (2020) for aligning the EEG distributions of different subjects before training and inference processes. It turned out that the idea of transfer learning benefits not only deep neural networks but also other machine learning methodologies for BCI classification. In Zhang and Wu (2020), the author proposed Manifold Embedded Knowledge Transfer (MEKT) with a combination of alignment, feature extraction, and domain adaptation techniques to produce projection matrices that minimize the joint probability distribution shift between the source and the target domains. The projected features are then used to train classifiers like Support Vector Machine (SVM) and Linear Discriminant Analysis (LDA). A regularized covariance matrix estimation framework for common spatial pattern (CSP) (Ramoser, Muller-Gerking, & Pfurtscheller, 2000) based on dynamic time warping (DTW) and transfer learning is proposed in Azab, Ahmadi, Mihaylova, and Arvaneh (2020).

However, these transfer learning methodologies focus on extracting features and adapt them from the source subject(s) to the target subject. The projected feature vectors are then used to train a classifier (deep CNN, SVM, etc.). Deep neural networks, although being treated as a classifier, can also learn and extract features and offer an end-to-end decoding. Instead of manually crafting the feature projections for transfer learning, we aim to manipulate the neural network itself for adaptation, while incorporating only little data pre-processing.

In this paper, we study 5 schemes for adaptation of a deep CNN with limited EEG data. We utilize the aforementioned network architecture in Schirrmeister et al. (2017) as a baseline to exploit the full learning capacity of a deep CNN with minimal human intervention. The goal is to leverage the features extracted from the convolution filters in the model and adapt the classifier to a subject it has never encountered. Literature reports the classification accuracy for motor imagery EEG using state-of-the-art approaches ranging from 60% to 80% (Kwon et al., 2019; Lotte, Congedo, Lécuyer, Lamarche, & Arnaldi, 2007; Zhang et al., 2019). In this study, we report an enhanced subject-independent motor imagery classification with an average ($N = 54$) accuracy of 84.19% ($\pm$9.98%), while the best accuracy on this dataset is 74.15% ($\pm$15.83%) in the literature (Kwon et al., 2019). Using the proposed adaptation methodology, we are able to further improve this, resulting in a statistically significant ($p = 0.005$) 3.21% increase in average accuracy. We further study the effect of different learning rates and percentages of adaptation data, and also demonstrate the variability in performance of individual subjects. To the best of our knowledge, a similar high accuracy for subject-independent MI classification and similar schemes for subject adaptation with a deep CNN for MI classification have not yet been reported in the literature. The source code of our implementation is available on GitHub.[1]

The rest of the paper is organized as follows: Section 2 explains the methodology, Section 3 describes the dataset and evaluation criteria, Section 4 reports the results, discusses their significance. We finally conclude the paper in Section 6.

2. Methodology

In this section, we first describe the definitions and notations used in the paper and introduce the network architecture and optimization techniques. Then, we establish 2 baselines to compare the performance of the proposed subject adaptation: subject-specific and subject-independent classification. Finally, we present different schemes for subject-adaptive classification.

2.1. Definitions and notations

Assuming that the input to the model is on a per-trial basis; i.e.: the continuous EEG signals are segmented into labeled trials, we denote (X^i, y^i) as a single trial i. The input matrix $X^i \in \mathbb{R}^{N_c \times N_t}$ is the pre-processed signal, where N_c is the number of EEG channels and N_t is the time samples for a single trial. Its corresponding label is denoted as $y^i \in L = \{0 : \text{“Right"}, 1 : \text{“Left"}\}$ for the hand MI paradigm used in our experiment.

The CNN can be represented as a classifier $f : \mathbb{R}^{N_c \times N_t} \rightarrow L$, defined as:

$$f(X^i; \theta) = g(\phi(X^i; \theta_{\phi_1}, \ldots, \theta_{\phi_{N_\phi}}); \theta_g) \tag{1}$$

In this equation, $\phi : \mathbb{R}^{N_c \times N_t} \rightarrow \mathbb{R}^{N_g}$ is all the convolution layers with θ_{ϕ_j} denoting the parameters for convolution block j and N_ϕ

[1] `eeg-adapt` codebase: https://github.com/zhangks98/eeg-adapt.

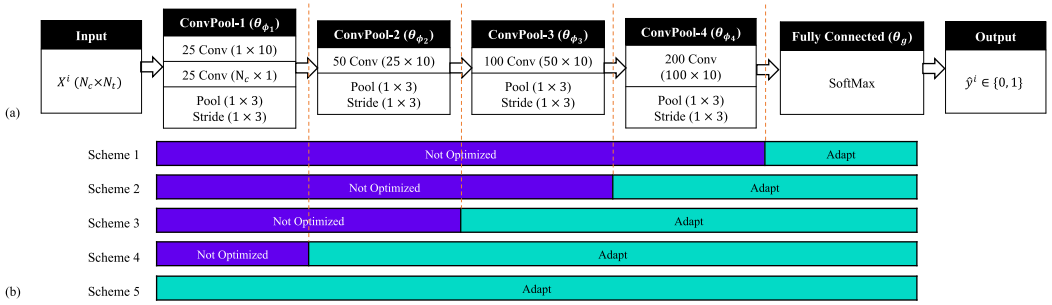


Fig. 1. Illustrations of (a) Network architecture and (b) Adaptation schemes: optimizing a subset of model parameters θ.

denoting the number of convolution blocks. N_g is dimension of the flattened output of the final convolution block. This serves as the input for the fully-connected layer $g : \mathbb{R}^{N_g} \rightarrow L$ with parameters θ_g.

2.2. Network architecture

In this study, we follow the deep CNN architecture described in Schirrmeister et al. (2017), which has been extensively analyzed for the choice of learning parameters and optimization strategies, and produces state-of-the-art results. The deep CNN consists of a temporal and spatial filter with max-pooling, 3 convolution-max-pooling blocks, and a fully-connected *softmax* classification layer (see Fig. 1a). Hence, Eq. (1) for the deep CNN will have $N_\phi = 4$ convolution blocks.

In this study, first we investigate the performance of deep CNN in the conventional scenario in which the network is trained and evaluated on the same subject's data. Next, we study subject-transfer in which the model trained on a set of subjects is evaluated on a new target subject. Further, to enhance the performance of the transferred model by taking into account the inter-subject variability of data, we propose different adaptation schemes as indicated in Fig. 1b. The following sub-sections will explain the methods in more detail.

2.3. Training and optimization

We used AdamW (Loshchilov & Hutter, 2017) to optimize the negative log-likelihood loss. We also applied cosine annealing to accelerate the training (Loshchilov & Hutter, 2016), and performed Batch Normalization (Ioffe & Szegedy, 2015) and Dropout (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014) for each convolution-max-pooling block. We trained the network for a maximum of 200 epochs, and select the epoch with the lowest validation loss. The training techniques are used consistently for subject-specific, subject-independent and subject-adaptive (Sections 2.4, 2.5) classification.

2.4. Subject-specific and subject-independent classification

Subject-specific classification is the first baseline for our experiment, where we train and validate a model for each subject using only the same subject's data.

Subject-independent classification (Kwon et al., 2019) is the second baseline for our study, where we implement the leave-one-subject-out (LOSO) paradigm for evaluation. The training and validation data for a subject consists of data from all available subjects excluding target subject.

Table 1
An overview of adaptation schemes.

Scheme	Params for Adaptation	# Trainable Params
1	θ_g	2,802
2	$\theta_g, \theta_{\phi_4}$	203,202
3	$\theta_g, \theta_{\phi_4}, \theta_{\phi_3}$	253,402
4	$\theta_g, \theta_{\phi_4}, \theta_{\phi_3}, \theta_{\phi_2}$	266,002
5	$\theta_g, \theta_{\phi_4}, \theta_{\phi_3}, \theta_{\phi_2}, \theta_{\phi_1}$	305,077

2.5. Subject-adaptive classification

In the subject-independent classification case, the model never observes any data from the target subject during the training process. However, as mentioned in Section 1, this is prone to inter-subject variations. Therefore, in subject-adaptive classification, we fine-tune and adapt a pre-trained model using a small amount of data from the target subject. For fairness of comparison, the data for evaluation is never observed during the adaptation process, and is consistent with the baseline methods in Section 2.4. Details for the division of data will be explained in Section 3.2. For each target subject, the model trained in subject-independent classification (Section 2.4) serves as a pre-trained model. We studied the performance gains when the amount of adaptation data increases from 10% to 100% in steps of 10%.

We proposed different strategies of adaptation as illustrated in Fig. 1b. Because the adaptation data is limited compared to trainable parameters in network, we chose to adapt the fully-connected layer in our first scheme, i.e. optimizing the classifier parameter θ_g, leaving the parameters for the feature extractor $\theta_{\phi_i}, i = 1...4$ unchanged. This is also based on the hypothesis that the convolutional layers can act as a feature extractor, and have already extracted useful representations of the EEG data. However, we do not want to omit the possibility that the convolutional layers can also be adapted, so we included other schemes which also adapted the parameters for the convolution layers, which are shown in Fig. 1b. As indicated, in each scheme, the first k layers ($\theta_{\phi_{1...k}}$) are pre-trained and kept unchanged, whereas the rest of the layers ($\theta_{\phi_{k+1...4}}$ and θ_g) are re-trained using the adaptation data, thus updating the trainable parameters to match the target subject. All the adaptation schemes are outlined in Table 1, where we highlighted the number of trainable parameters for each scheme. Note that scheme 5 has the same amount of training parameters as the models in Section 2.4.

In these schemes, since the adaptation data is small comparing to the data to train the subject-independent model, we need to tune down the learning rate to avoid clobbering the initialization (Girshick, Donahue, Darrell, & Malik, 2015). Hence, we investigate the choice of optimal learning rate for better adaptation. Let the original learning rate in the base model be η (in our configuration, $\eta = 0.01$ for the subject-independent

model), and let α be the coefficient that scales down the learning rate, then:

$$\theta^{[i+1]} = \theta^{[i]} - \alpha\eta\frac{\partial L}{\partial \theta^{[i]}} = (1-\alpha)\theta^{[i]} + \alpha(\theta^{[i]} - \eta\frac{\partial L}{\partial \theta^{[i]}}) \quad (2)$$

where $\theta^{[i]}$ is the trainable parameters at the ith iteration and L is the loss function. The equation above showed that scaling down the learning rate (i.e. $\alpha < 1$) can be viewed as accepting only α portion of the new parameters. Thus, lowering the learning rate can be interpreted as weighted adaptation. We experimented with $\alpha = 1, 0.1, 0.05, 0.01$ and observed the result.

In the following sections, an *adaptation scheme* is defined as optimizing a subset of θ for f. The *adaptation rate* is defined as the percentage of available adaptation data used in each scheme, which ranges from 10% to 100% in steps of 10%. An *adaptation configuration* is a combination of an adaptation scheme and an adaptation rate.

3. Evaluation

In this section, we will introduce the dataset and the pre-processing steps, as well as how the data will be used to train, validate, and evaluate different models.

3.1. Dataset

The EEG dataset used in our research is collected by the Department of Brain and Cognitive Engineering, Korea University. In their experiments, 54 healthy subjects (ages 24–35) performed binary class MI tasks, and their EEG signals were recorded using BrainAmp (Brain Products; Munich, Germany) with 62 Ag/AgCl electrodes at a sampling rate of 1000 Hz. The design of the experiments follows the well-established protocol in Pfurtscheller and Neuper (2001). Each trial begins with a fixation mark at the center of the screen for the subject to prepare for the trial. Then a left or right arrow will appear as a visual clue for 4 s, during which the subject performed the imagery task of grasping with the appropriate hand. After each task, the screen remained blank for 6 s (±1.5 s). More details on the data and the experiment protocol can be found in Lee et al. (2019).

Each subject participated in two data recording sessions with a total of 400 trials. Each session consists of an offline training phase to record data and to construct the classifier, and an online test phase that provided visual feedback to the subject by decoding data using the classifier. Each phase has 100 trials.

For our experiment, all 62 EEG channels were used. Each 4-second MI task was first segmented from the continuous data for each trial. The signals were further down-sampled by a factor of 4 with an order-8 Chebyshev type-I filter for anti-aliasing. Hence, in our experiment, $N_c = 62$ and $N_t = 1000$.

3.2. Division of data

All the methodologies described in Section 2 are evaluated with session 2, phase 2 data, which consists of 100 trials. As stated in Section 2.5, this part of the data is never used to train and validate any model in our study.

In subject-specific classification, we use phase 1 and phase 2 of session 1 data for training and validate the model using phase 1 of session 2 data.

In subject-independent classification, the entire data from all but the target subject is used for training. For each target subject, we performed a 6-fold cross validation on the data from remaining subjects for model selection. In each fold, the data from remaining 53 subjects are randomly split into training (85%) and validation (15%) sets and the network is trained as indicated in Section 2.3. The network with minimum validation loss across all cross-validation folds is used to evaluate the target subject.

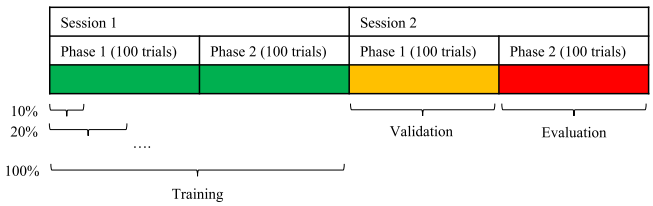


Fig. 2. Division of data for subject-adaptive classification for target subject.

Table 2
Average classification accuracy (%) and standard deviation for baseline models.

	Subject Specific	Subject Independent
Mean (SD)	63.54 (14.25)	84.19 (9.98)

In subject-adaptive classification, the model with the least validation loss across cross-validation folds in the subject-independent classification is used as the base model. We use only portions of phase 1 and 2 of session 1 data to fine-tune the network, ranging from 10% to 100% in steps of 10%. Each adaptation configuration is validated with phase 1 of session 2 data. This is illustrated in Fig. 2.

4. Results

The results of the two baseline models and different adaptation configurations are reported in this section, including the average classification accuracy, computation time, a comparison with other BCI classification methodologies, and inter-subject variations in performance. This is followed by a discussion on the significance of the results.

For each configuration, we train the model using one NVIDIA Tesla V100 SXM2 16 GB GPU from DGX-1 nodes. Each node is powered by Dual 20-Core Intel Xeon E5-2698 v4 2.2 GHz.

4.1. Average classification accuracy

The following sub-sections report the average classification accuracy across all 54 subjects for various training and adaptation configurations. We analyze the significance of the average improvement using paired-sample t-test across all subjects.

4.1.1. Baseline accuracy

The average classification accuracy for subject-specific classification and subject-independent classification is outlined in Table 2. The average classification accuracy for subject-adaptive classification when the learning rate is unchanged ($\alpha = 1$) is listed in Table 3.

The subject-specific scheme has an accuracy of 63.54%, whereas the subject-independent scheme has an accuracy of 84.19% (32.50% increase, $p = 6.97 \times 10^{-17}$). This shows that more data has significantly increased the performance of the deep CNN model, which has 305,077 trainable parameters.

When $\alpha = 1$, the subject-adaptive models have the same learning rate as the base model. As shown in Table 3, all the best accuracy for each adaptation rate is achieved in scheme 1. The overall best accuracy is obtained with an adaptation rate of 60%, with a 2.13% increase in accuracy ($p = 0.03$). While the increase is significant, we will see in the following section that tuning down the learning rate will further decrease the p-value by an

Table 3
Average classification accuracy (%) for each adaptation configuration when $\alpha = 1$. The highest accuracy (in bold) indicates the best adaptation scheme for a specific adaptation rate.

Adaptation Rate	Adaptation Scheme				
(%)	1	2	3	4	5
10	**84.54**	83.44	83.76	84.17	83.43
20	**84.78**	84.09	83.96	84.31	83.35
30	**84.94**	84.44	83.94	83.83	83.33
40	**85.65**	84.22	83.70	83.69	83.81
50	**85.83**	84.39	84.50	83.50	82.52
60	**85.98**	85.28	83.96	83.96	82.96
70	**85.69**	84.52	83.54	84.31	82.24
80	**85.24**	84.56	84.31	84.06	82.70
90	**85.76**	85.37	84.39	84.41	81.09
100	**85.81**	84.98	84.41	83.91	82.80

Table 4
Average classification accuracy (%) for each adaptation configuration with $\alpha = 0.1$. The highest accuracy (in bold) indicates the best adaptation scheme for this adaptation rate.

Adaptation Rate	Adaptation Scheme				
(%)	1	2	3	4	5
10	84.83	84.87	**84.94**	84.43	83.98
20	84.80	85.19	84.94	85.17	**85.26**
30	85.07	84.93	85.07	**85.35**	84.80
40	**85.87**	85.37	85.50	85.46	83.94
50	85.69	85.35	85.76	**85.98**	85.44
60	85.93	85.91	85.76	**86.11**	84.61
70	86.06	86.19	86.00	**86.48**	85.02
80	86.17	85.98	85.94	**86.33**	84.76
90	85.96	86.07	86.35	**86.46**	85.37
100	86.04	85.96	86.24	**86.69**	85.98

Table 5
Average classification accuracy (%) for each adaptation configuration with $\alpha = 0.05$. The highest accuracy (in bold) indicates the best adaptation scheme for this adaptation rate. Significant ($p < 0.01$) improvements between the result from scheme 4, adaptation rate 80% and other configurations are marked with *.

Adaptation Rate	Adaptation Scheme				
(%)	1	2	3	4	5
10	84.78*	84.80*	**84.87***	84.35*	84.85*
20	85.09	84.96*	84.94*	**85.50**	85.15
30	85.22	85.04*	85.07	**85.41**	84.69*
40	**85.96**	85.74	85.80	85.85	85.09*
50	85.93	85.67	85.81	**86.07**	85.28*
60	85.76	**86.28**	86.22	86.06	85.07*
70	85.98	86.33	86.37	**86.41**	85.17*
80	85.93	86.19	85.89*	**86.89**	85.24*
90	85.81	86.31	86.31	**86.43**	85.37*
100	86.11	86.35	86.41	**86.80**	85.85

Table 6
Average classification accuracy (%) for each adaptation configuration with $\alpha = 0.01$. The highest accuracy (in bold) indicates the best adaptation scheme for this adaptation rate.

Adaptation Rate	Adaptation Scheme				
(%)	1	2	3	4	5
10	84.91	**84.93**	84.76	84.61	84.70
20	85.11	**85.30**	85.04	85.24	85.13
30	85.13	**85.56**	85.48	85.48	85.15
40	85.35	85.78	**86.00**	85.80	85.00
50	85.56	85.93	85.85	**86.04**	85.69
60	85.28	86.24	**86.37**	85.91	85.07
70	85.41	86.17	86.17	**86.54**	85.56
80	85.69	**86.19**	85.93	86.17	85.80
90	85.67	**86.22**	86.07	86.20	85.52
100	85.57	86.44	86.50	**86.80**	85.93

order of magnitude while increasing average accuracy. Also notice that all the classification accuracy in scheme 5 is worse than the subject-independent model, this shows that at a large learning rate, the adaptation of all convolutional layers with numerous training parameters can produce sub-optimal results.

4.1.2. Scaling down the learning rate

As mentioned in Section 2.5, different learning rates can have an impact on the final classification accuracy. Here, we report the results of 3 different scaling factors (α in Eq. (2)) that tunes down the learning rate for each adaptation configuration. The results are shown in Tables 4, 5 and 6 respectively, in decreasing order of α. In each table, the highlighted accuracy indicates the best adaptation scheme for a particular adaptation rate.

In contrast to $\alpha = 1$, when $\alpha = 0.1$ (10% of the learning rate in the base model), the best performance for each adaptation rate comes from different schemes. This shows that the convolutional layers can indeed be further adapted when we decrease the learning rate. When there is more adaptation data, the adaptation schemes tend to perform better. Also notice that a majority of high classification accuracy comes from scheme 4, especially in higher adaptation rates. The best performance is obtained with scheme 4, 100% adaptation rate, which improves the subject-independent result by 2.97% ($p = 0.003$).

Similar to $\alpha = 0.1$, when $\alpha = 0.05$ (5% of the learning rate in the base model), the best performance for each adaptation rate is seen from different schemes. One difference is that scheme 5 never achieves a better performance in all adaptation rates. Scheme 4 with 80% adaptation rate gives the best result of 86.89% (+3.21%, $p = 0.005$ comparing to the subject-independent model).

When $\alpha = 0.01$ (10% of the learning rate in the base model), more scheme 2 configurations exhibit higher average classification accuracy. Scheme 4 with 100% adaptation rate gives the best result of 86.80% (+3.10%, $p = 0.005$ comparing to the subject-independent model).

To further illustrate the effect of tuning down the learning rate, we turned off the early-stopping mechanism for each subject and trained the model for 200 epochs. We compared $\alpha = 1, 0.1, 0.01, 0.05$ for scheme 4 with 100% of adaptation data, and plotted the average training and validation accuracy across 54 subjects, as shown in Fig. 3. When $\alpha = 1$, there is a downward overshoot in average validation accuracy, while the training accuracy increases rapidly. Also, the average validation accuracy never goes above that of epoch 0, which is the accuracy of the subject-independent model. This is because in the subject-adaptive scenario, the data available for fine-tuning is only a small fraction of the total amount of data on which the original network was trained, and keeping the same learning rate leads to overfitting to the training data during adaptation. When $\alpha < 1$, an increase in average validation accuracy can be seen, which shows the model is improving when we tune down the learning rate. This is consistent with our earlier results. However, it is worth noting that this is only indicative since here we are using the validation data, not the evaluation data, and during the actual run, the adaptation of each subject stops at a different epoch according to its own validation loss.

So far, $\alpha = 0.05$ yields the best average result with significant ($p = 0.005$) increase. The following discussions, unless otherwise stated, will be based on the result obtained using this scaling factor for the learning rate.

We further performed paired-sample t-test to obtain the p-values between the best result (scheme 4, adaptation rate 80%) and other adaptation configurations. The significant ($p < 0.01$) improvements are marked with * in Table 5. Overall, this configuration performs significantly better than almost all scheme 5

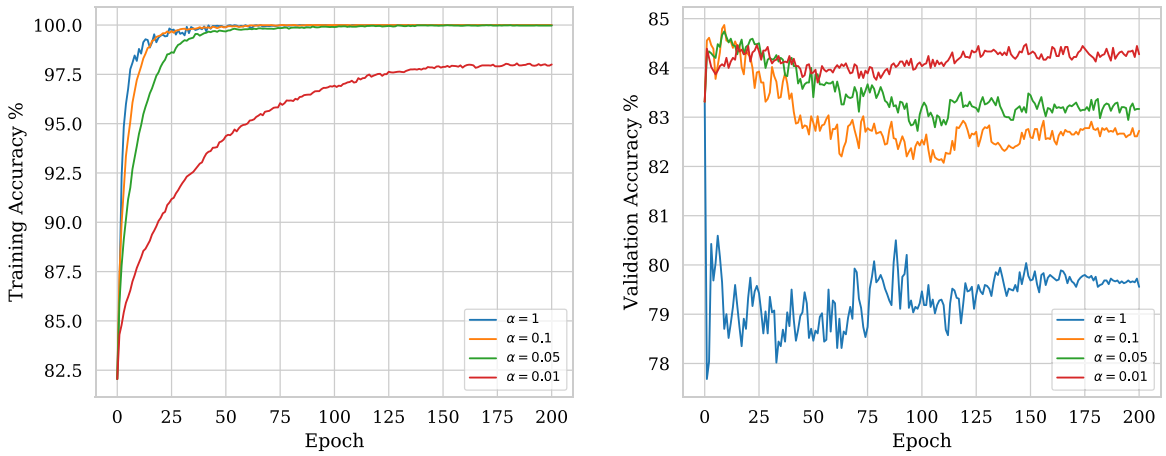


Fig. 3. Demonstrating the impact of learning rate in training and validation accuracy (%) for $\alpha = 1, 0.1, 0.05, 0.01$ when training scheme 4 model with 100% adaptation rate.

configurations, which fine-tunes the first convolutional block for discovering elementary spatial and temporal features, along with all the layers in scheme 4. This shows that the first convolutional block in our subject-independent model has been adequately trained and further adapting it will lead to overfitting. On the other hand, when the adaptation rate is large ($>$ 80%), the improvement from our best result comparing with other schemes are not as significant.

4.1.3. Alternative partition of data

To truly evaluate the capacity of the network, we re-run the classification experiment using session 2 as training data, and phase 2 of session 1 data for evaluation (i.e. flipping the roles of session 1 and 2 in Fig. 2). We obtained a 25.14% ($p = 6.83 \times 10^{-9}$) improvement in average accuracy from subject-specific to subject-independent classification. As for subject-adaptive classification, the same $\alpha = 0.05$ is used, and the configuration that yields the most improvement in average accuracy is scheme 4 with 100% adaptation data, which is 4.22% ($p = 0.0004$) comparing to subject-independent classification. The result is similar, if not better than the original partition of data. Nevertheless, for the fair of comparison with the result reported in Kwon et al. (2019) later in Section 4.3, which is only evaluated on session 2, phase 2 data, we report the results with the original partition of data in Fig. 2 for the rest of the analysis.

4.2. Computation time

The average computation time over all 54 subjects for each methodology is reported in Table 7. Overall, the training time for subject-specific and subject-adaptive models is on the same order of magnitude (within 20 s), whereas subject-independent models take more than 9 h to train on average. When the amount of adaptation data increases, it takes longer to train the network. Scheme 1 tends to have longer training time than other schemes, this is because we employed early-stopping in the training. Other schemes have more trainable parameters, so they are more prone to overfitting, and their validation losses rise quickly after the first few epochs. Therefore, the training will stop earlier. It is worth noting that this result is highly machine-dependent, and is only served as a reference.

Table 7
Summary of computation time for each methodology (in seconds). Models are trained on NVIDIA Tesla V100 SXM2 16 GB GPU.

	Subject Specific				Subject Independent
Mean	7.25				32961.01
Adaptation Rate	Adaptation Scheme				
(%)	1	2	3	4	5
10	7.71	6.17	4.23	3.44	1.54
20	8.58	6.03	5.52	5.64	3.38
30	11.18	5.71	5.45	7.08	4.81
40	10.30	5.63	5.99	7.51	4.96
50	11.66	5.82	6.63	8.81	5.61
60	11.56	7.70	8.52	8.97	6.15
70	11.26	7.38	8.85	9.90	7.02
80	12.35	6.96	8.96	11.46	8.20
90	17.30	10.42	10.79	13.17	8.29
100	17.60	12.31	12.38	15.31	9.51

4.3. Comparison with other methodology

First, we compare our best average classification result (scheme 4, using 100% adaptation data) with other commonly used machine learning techniques in EEG such as common spatial pattern (CSP) (Ramoser et al., 2000), common spatio-spectral pattern (CSSP) (Lemm, Blankertz, Curio, & Muller, 2005), filter bank common spatial pattern (FBCSP) (Ang et al., 2008), and Bayesian spatio-spectral filter optimization (BSSFO) (Suk & Lee, 2012). These results were reported in Lee et al. (2019) using the same dataset. For each methodology, a subject-specific classifier is constructed and evaluated. We selected the results that are evaluated on the last 100 trials (the online testing phase) of session 2, which is consistent with our test set. However, Lee et al. (2019) did extra pre-processing steps on the raw EEG data: their test data only involves 20 electrodes in the motor cortex region (FC-5/3/1/2/4/6, C-5/3/1/z/2/4/6, and CP-5/3/1/z/2/4/6), and is band-pass filtered between 8 and 30 Hz with a 5th order Butterworth digital filter.

Next, we compare our result with the best accuracy on this dataset reported in Kwon et al. (2019). They examined both subject-specific and subject-independent classification using CNN. They also employed extra pre-processing steps to the signal, including the selection of the aforementioned 20 electrodes, and uses a spectral–spatial feature representation. The results are shown in Table 8.

It can be seen that subject-independent and subject-adaptive models gives lower variations in accuracy among all methodologies. In addition, all the subjects achieves classification accuracy of more than 50%.

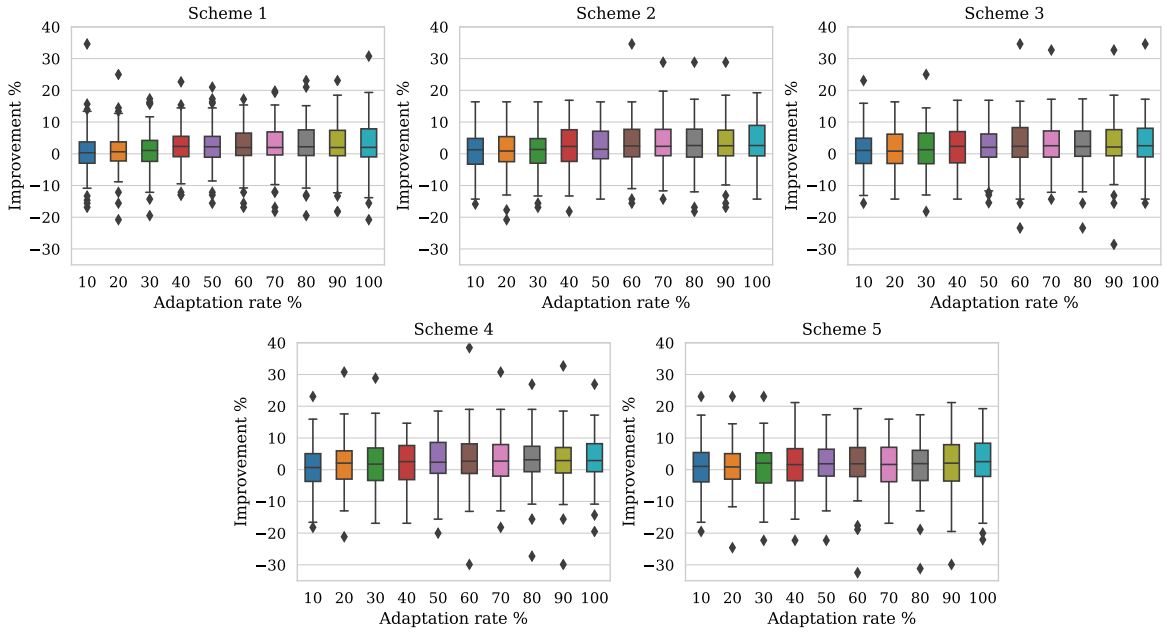


Fig. 4. Box plots of the percentage improvement in every single subject, from subject-independent classification to each adaptation configuration.

Table 8
Comparison of average classification accuracy (%) for different methodologies.

Methodology	Mean (SD)	Median	Range (Max–Min)
Subject-Specific			
CSP (Ramoser et al., 2000)	68.57 (17.57)	64.50	58.00 (100.00-42.00)
CSSP (Lemm et al., 2005)	69.69 (18.53)	63.00	58.00 (100.00-42.00)
FBCSP (Ang et al., 2008)	70.59 (18.56)	64.00	55.00 (100.00-45.00)
BSSFO (Suk & Lee, 2012)	71.02 (18.83)	63.50	52.00 (100.00-48.00)
CNN (Kwon et al., 2019)	71.32 (15.88)	66.45	53.10 (99.00-45.90)
Deep CNN	63.54 (14.25)	60.50	57.00 (100.00-43.00)
Subject-Independent			
CNN (Kwon et al., 2019)	74.15 (15.83)	75.00	60.00 (100.00-40.00)
Deep CNN	**84.19 (9.98)**	**84.50**	**47.50 (99.50-52.00)**
Subject-Adaptive			
Deep CNN	**86.89 (11.41)**	**88.50**	**44.00 (100.00-56.00)**

4.4. Inter-subject variations in performance

The results in previous sections report the performance of proposed methods as average over 54 subjects. The range of performance across subjects indicate that there exists a variation in how each method benefits an individual subject. This variability is demonstrated in Fig. 4, in which we use box plots to show the distribution of percentage improvement in the accuracy of all subjects of each adaptation configuration with varying amounts of adaptation data. On each box, the central mark indicates the median, and the bottom and top edges of the box indicate the 25th and 75th percentiles. The whiskers extend to the most extreme data points excluding the outliers, and the outliers are plotted individually using the ♦ symbol. It can be seen that each adaptation configuration has a few outliers, i.e., the classification accuracy of some subjects increases by more than 20%. This motivates us to further analyze the result on a subject level.

We illustrate the subject-level performance in Fig. 5, for three selected adaptation configurations. In these figures, the subjects are sorted in decreasing order of their difference in performance with and without adaptation. For each subject, the orange and blue markers indicate performance with and without adaptation respectively. The green bar indicates an increase in accuracy for the subject, and the red bar indicates a decrease in accuracy.

The configuration with largest increase in accuracy for a single subject is identified as scheme 4 with 60% adaptation rate. The results for this configuration are illustrated in Fig. 5(a). It can be noted that, subject 50 achieves the highest increase in accuracy of 38.46% (from 52% to 72%) by adapting the subject-independent model. Overall, 19 out of 54 (35.18%) subjects in this configuration achieves accuracy improvement of greater than 5%.

The largest percentage of subjects with more than 5% increase is 40.74% (22 out of 54), which is seen in scheme 4 with 100% adaptation rate. In this configuration, subject 50 only sees a 26.92% increase, but it is still the largest improvement in this configuration. The detailed subject-level comparison for this configuration is illustrated in Fig. 5(b).

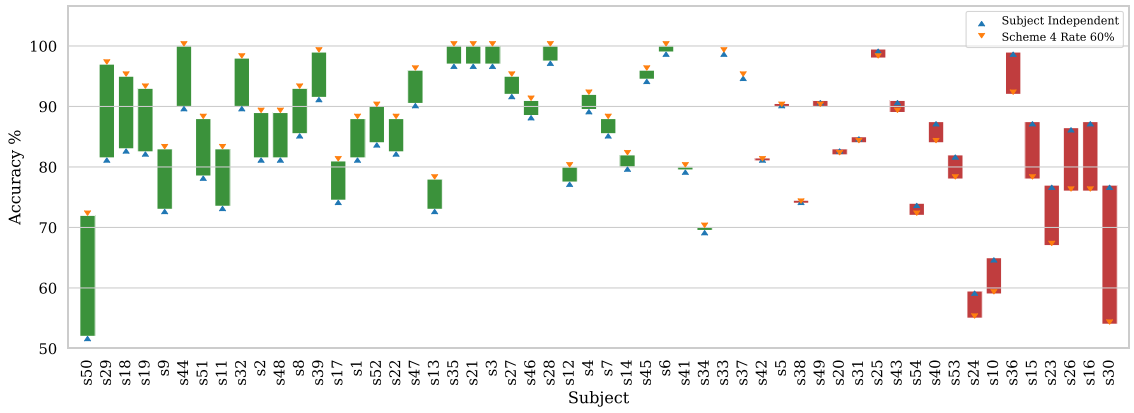
Fig. 5(c) gives a subject-level comparison between the subject-independent model and *best* average adaptation configuration: scheme 4 with 80% adaptation rate.

5. Discussion

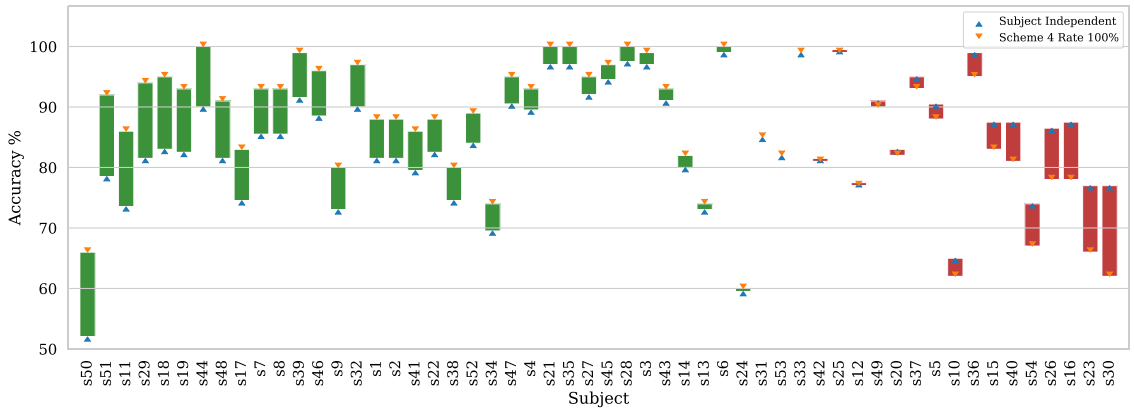
In this study, we explored the feasibility of adapting a pre-trained subject-independent deep CNN model to perform high performance classification of MI data from a new unseen target subject. The results reported in Section 4 demonstrate the superiority of our network model, training and adaptation strategy.

In Section 1, we have seen that inter-subject variability is one of the major concerns for MI classification in EEG-BCI systems (Wronkiewicz et al., 2015). The desired scenario is to train subject-specific models, using data entirely from a subject. The traditional approach in BCI is to develop feature extraction and classification algorithms on limited amount of subject-specific data. However, to train a complex deep CNN model with large number of trainable parameters, subject-specific data will be inadequate. We demonstrate this in Section 4.1.1 and the results are indicated in Table 8. The subject-independent model extensively trained using large number of samples offers a significant ($p < 10^{-16}$) 32.50% increase in accuracy over subject-specific model.

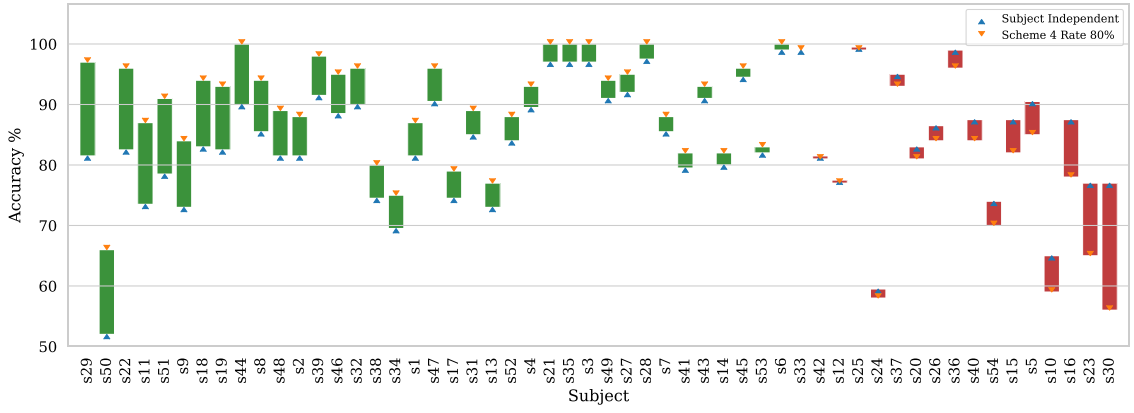
Nonetheless, the subject-independent models still have the issue of inter-subject variability, and can be further adapted. This paper addresses this challenge by studying 5 different subject adaptation schemes for a deep CNN. We experimented with different percentages of adaptation data and different learning



(a) Scheme 4, 60% adaptation rate.



(b) Scheme 4, 100% adaptation rate.



(c) Scheme 4, 80% adaptation rate.

Fig. 5. Subject-level comparison of accuracy between subject-independent model and different adaptation configurations. The bar edges indicate the performance with (orange marker) and without adaptation (blue marker). The green bars represent subjects with performance increase using adapted model and the red bars indicate fall in performance. The subjects are sorted in decreasing order of difference in performance.

rates for fine-tuning the base model from subject-independent classification. In Section 4.1.2, we have seen that using 80% of the available adaptation data for a subject (80 trials), we can have a significant ($p = 0.005$) 3.21% increase in accuracy by adapting the last 4 convolution-pooling blocks of the deep CNN model (scheme 4). This shows that by fine-tuning a subset of the CNN parameters with data from the target subject can significantly increase the average classification accuracy. This bridges the gaps in cross-subject transfer of deep learning models and tackles inter-subject variability.

Further exploration of our adaptation schemes sheds some light into the optimal different adaptation data and learning rates for effective adaptation. We notice that except for $\alpha = 1$, the best performing adaptive models all come from scheme 4 configurations. On the one hand, it shows that the convolutional layers can be further adapted with a lower learning rate as demonstrated in Tables 4–6. On the other hand, Table 5 illustrates that the

improvements are significant when comparing to other adaptation configurations with fewer adaptation data. We also showed that adapting the first convolutional block can lead to overfitting as there is a significant decrease in accuracy. This indicates that the first layer is adequately trained in the subject-independent model, whereas the features extracted from the deeper convolutional layers in Schirrmeister et al. (2017)'s architecture are still susceptible to inter-subject variations.

In our current adaptation schemes, the weight for the changes in each layer is the same α. However, different layers may have different importance when adapting the distribution of the source domain to the target domain. The techniques like Maximum Mean Discrepancy (MMD) (Rozantsev, Salzmann, & Fua, 2018) and Correlation Alignment (CORAL) (Sun, Feng, & Saenko, 2016) can be used to further regularize the adaptation of individual CNN layer. Furthermore, the fact that all the deeper layers except for the first convolutional-pooling block are sensitive to inter-subject variability indicates that a more domain-invariant feature is yet to be discovered. Using Generative Adversarial Networks (GANs) (Ganin & Lempitsky, 2014; Ghifary, Kleijn, Zhang, Balduzzi, & Li, 2016; Liu & Tuzel, 2016) may be a possible direction to address this issue. Also, based on the CNN model, an online classification model can be obtained by either classifying repeatedly using features from a sliding window with a fully-connected layer or developing models with Recurrent Neural Networks (RNN) to better exploit the temporal relationships.

Leveraging the learning capacity of the deep CNN means that a lot of time is devoted to training the subject-independent base model due to a large amount of data and trainable parameters. With the base model, however, the adaptation is relatively fast. It also means that the features it has learned may be less obvious, but producing *interpretable* features is equally important. Robinson et al. (2019), Schirrmeister et al. (2017) provided useful visualizations based on known band power features, and Lawhern et al. (2018) reported the relevance of individual features on the resulting classification decision using DeepLIFT (Shrikumar, Greenside, & Kundaje, 2017). However, little is discovered beyond the features we have already known. In future, we consider to extend this work to explore the yet unknown features that are learned by the network.

6. Conclusion

In this paper, we studied 5 adaptation schemes of a deep CNN for EEG-based motor imagery classification. By comparing the two baseline models (subject-specific and subject-independent models), we observed a significant ($p < 10^{-16}$) 32.50% increase in accuracy and showed that more data is crucial when using deep learning methods in EEG-based BCI systems. The reported performance of 84.19 % for subject-independent MI classification is the highest compared to state-of-the-art methods in literature. We further propose improvement in the subject-independent model to address the inter-subject variability that might impact performance of a target subject. We showed that using scheme 4 with a lower learning rate, a significant ($p = 0.005$) 3.21% increase in accuracy can be achieved using 80% of adaptation data. By comparing our proposed adaptation scheme with other state-of-the-art machine learning techniques for MI classification, we showed that our scheme can produce higher average accuracy with lower variability (manifested in both lower standard deviation and range). The minimum classification accuracy among all 54 subjects is also the highest among all methodologies.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was partially supported by the RIE2020 Advanced Manufacturing and Engineering (AME) Programmatic Fund, Singapore (No. A20G8b0102).

This work was partially supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (No. 2017-0-00451, and No. 2019-0-00079).

The computational work for this project was partially performed on resources of the National Supercomputing Centre, Singapore (https://www.nscc.sg).

References

Ang, K. K., Chin, Z. Y., Zhang, H., & Guan, C. (2008). Filter bank common spatial pattern (FBCSP) in brain-computer interface. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)* (pp. 2390–2397). IEEE.

Arvaneh, M., Guan, C., Ang, K. K., & Quek, C. (2013). EEG data space adaptation to reduce intersession nonstationarity in brain-computer interface. *Neural Computation*, *25*(8), 2146–2171.

Azab, A. M., Ahmadi, H., Mihaylova, L., & Arvaneh, M. (2020). Dynamic time warping-based transfer learning for improving common spatial patterns in brain–computer interface. *Journal of Neural Engineering*, *17*(1), Article 016061. http://dx.doi.org/10.1088/1741-2552/ab64a0.

Blankertz, B., Tomioka, R., Lemm, S., Kawanabe, M., & Muller, K.-R. (2007). Optimizing spatial filters for robust EEG single-trial analysis. *IEEE Signal Processing Magazine*, *25*(1), 41–56.

Craik, A., He, Y., & Contreras-Vidal, J. L. (2019). Deep learning for electroencephalogram (EEG) classification tasks: A review. *Journal of Neural Engineering*, *16*(3), 31001.

Fahimi, F., Zhang, Z., Goh, W. B., Lee, T.-S., Ang, K. K., & Guan, C. (2019). Inter-subject transfer learning with an end-to-end deep convolutional neural network for EEG-based BCI. *Journal of Neural Engineering*, *16*(2), Article 026007. http://dx.doi.org/10.1088/1741-2552/aaf3f6.

Ganin, Y., & Lempitsky, V. (2014). Unsupervised domain adaptation by backpropagation. arXiv:1409.7495, arXiv e-prints, (p. arXiv:1409.7495).

Ghifary, M., Kleijn, W. B., Zhang, M., Balduzzi, D., & Li, W. (2016). Deep reconstruction-classification networks for unsupervised domain adaptation. In *European conference on computer vision* (pp. 597–613). Springer.

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2015). Region-based convolutional networks for accurate object detection and segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *38*(1), 142–158.

Goldberger, A. L., Amaral, L. A. N., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., et al. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation*, *101*(23), e215–e220.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv:1502.03167, arXiv e-prints, (p. arXiv:1502.03167).

Kwon, O., Lee, M., Guan, C., & Lee, S. (2019). Subject-independent brain-computer interfaces based on deep convolutional neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 1–14. http://dx.doi.org/10.1109/TNNLS.2019.2946869.

Lawhern, V. J., Solon, A. J., Waytowich, N. R., Gordon, S. M., Hung, C. P., & Lance, B. J. (2018). EEGNet: A compact convolutional neural network for EEG-based brain–computer interfaces. *Journal of Neural Engineering*, *15*(5), 56013.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444.

Lee, M.-H., Kwon, O.-Y., Kim, Y.-J., Kim, H.-K., Lee, Y.-E., Williamson, J., et al. (2019). EEG dataset and OpenBMI toolbox for three BCI paradigms: An investigation into BCI illiteracy. *GigaScience*, *8*(5), giz002.

Lemm, S., Blankertz, B., Curio, G., & Muller, K.-R. (2005). Spatio-spectral filters for improving the classification of single trial EEG. *IEEE Transactions on Biomedical Engineering*, *52*(9), 1541–1548.

Liu, M.-Y., & Tuzel, O. (2016). Coupled generative adversarial networks. In *Advances in neural information processing systems* (pp. 469–477).

Liyanage, S. R., Guan, C., Zhang, H., Ang, K. K., Xu, J., & Lee, T. H. (2013). Dynamically weighted ensemble classification for non-stationary EEG processing. *Journal of Neural Engineering*, *10*(3), Article 036007.

Loshchilov, I., & Hutter, F. (2016). SGDR: Stochastic gradient descent with warm restarts. arXiv:1608.03983, arXiv e-prints, (p. arXiv:1608.03983).

Loshchilov, I., & Hutter, F. (2017). Decoupled weight decay regularization. arXiv:1711.05101, arXiv e-prints, (p. arXiv:1711.05101).

Lotte, F., Congedo, M., Lécuyer, A., Lamarche, F., & Arnaldi, B. (2007). A review of classification algorithms for EEG-based brain–computer interfaces. *Journal of Neural Engineering*, *4*(2), R1.

Pfurtscheller, G., & Da Silva, F. H. L. (1999). Event-related EEG/MEG synchronization and desynchronization: Basic principles. *Clinical Neurophysiology*, *110*(11), 1842–1857.

Pfurtscheller, G., & Neuper, C. (2001). Motor imagery and direct brain-computer communication. *Proceedings of the IEEE*, *89*(7), 1123–1134.

Pfurtscheller, G., Stancak, A., Jr., & Edlinger, G. (1997). On the existence of different types of central beta rhythms below 30 Hz. *Electroencephalography and Clinical Neurophysiology*, *102*(4), 316–325.

Ramoser, H., Muller-Gerking, J., & Pfurtscheller, G. (2000). Optimal spatial filtering of single trial EEG during imagined hand movement. *IEEE Transactions on Rehabilitation Engineering*, *8*(4), 441–446.

Raza, H., Cecotti, H., Li, Y., & Prasad, G. (2016). Adaptive learning with covariate shift-detection for motor imagery-based brain–computer interface. *Soft Computing*, *20*(8), 3085–3096.

Robinson, N., Lee, S.-W., & Guan, C. (2019). EEG representation in deep convolutional neural networks for classification of motor imagery. In *2019 IEEE international conference on systems, man and cybernetics* (pp. 1322–1326). IEEE.

Rodrigues, P. L. C., Jutten, C., & Congedo, M. (2018). Riemannian Procrustes analysis: Transfer learning for brain–computer interfaces. *IEEE Transactions on Biomedical Engineering*, *66*(8), 2390–2401.

Rozantsev, A., Salzmann, M., & Fua, P. (2018). Beyond sharing weights for deep domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*(4), 801–814.

Sakhavi, S., & Guan, C. (2017). Convolutional neural network-based transfer learning and knowledge distillation using multi-subject data in motor imagery BCI. In *2017 8th international IEEE/EMBS conference on neural engineering* (pp. 588–591). IEEE.

Sakhavi, S., Guan, C., & Yan, S. (2018). Learning temporal information for brain-computer interface using convolutional neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, *29*(11), 5619–5629.

Schirrmeister, R. T., Springenberg, J. T., Fiederer, L. D. J., Glasstetter, M., Eggensperger, K., Tangermann, M., et al. (2017). Deep learning with convolutional neural networks for EEG decoding and visualization. *Human Brain Mapping*, *38*(11), 5391–5420.

Shrikumar, A., Greenside, P., & Kundaje, A. (2017). Learning important features through propagating activation differences. In *Proceedings of the 34th international conference on machine learning - volume 70* (pp. 3145–3153). JMLR.org.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(56), 1929–1958.

Suk, H.-I., & Lee, S.-W. (2012). A novel Bayesian framework for discriminative feature extraction in brain-computer interfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(2), 286–299.

Sun, B., Feng, J., & Saenko, K. (2016). Return of frustratingly easy domain adaptation. In *Proceedings of the thirtieth AAAI conference on artificial intelligence* (pp. 2058–2065). AAAI Press.

Tangermann, M., Müller, K.-R., Aertsen, A., Birbaumer, N., Braun, C., Brunner, C., et al. (2012). Review of the BCI competition IV. *Frontiers in Neuroscience*, *6*, 55.

Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., & Vaughan, T. M. (2002). Brain–computer interfaces for communication and control. *Clinical Neurophysiology*, *113*(6), 767–791.

Wronkiewicz, M., Larson, E., & Lee, A. K. (2015). Leveraging anatomical information to improve transfer learning in brain–computer interfaces. *Journal of Neural Engineering*, *12*(4), Article 046027.

Xu, L., Xu, M., Ke, Y., An, X., Liu, S., & Ming, D. (2020). Cross-dataset variability problem in EEG decoding with deep learning. *Frontiers in Human Neuroscience*, *14*, 103. http://dx.doi.org/10.3389/fnhum.2020.00103.

Zhang, W., & Wu, D. (2020). Manifold embedded knowledge transfer for brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *28*(5), 1117–1127.

Zhang, X., Yao, L., Wang, X., Monaghan, J., Mcalpine, D., & Zhang, Y. (2019). A survey on deep learning based brain computer interface: Recent advances and new frontiers. arXiv:1905.04149, arXiv e-prints, (p. arXiv:1905.04149).