Transfer Learning for Motor Imagery Based Brain-Computer Interfaces: A Complete Pipeline

Dongrui Wu, Ruimin Peng, Jian Huang*, Zhigang Zeng*

Key Laboratory of the Ministry of Education for Image Processing and Intelligent Control, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China

Abstract

Transfer learning (TL) has been widely used in motor imagery (MI) based brain-computer interfaces (BCIs) to reduce the calibration effort for a new subject, and demonstrated promising performance. After electroencephalogram (EEG) signal acquisition, a closed-loop MI-based BCI system also includes signal processing, feature engineering, and classification blocks before sending out the control signal to an external device, whereas previous approaches only considered TL in one or two such components. This paper proposes that TL could be considered in all three components (signal processing, feature engineering, and classification) of MI-based BCIs. Furthermore, it is also very important to specifically add a data alignment component before signal processing to make the data from different subjects more consistent, and hence to facilitate subsequential TL. Offline calibration experiments on two MI datasets verified our proposal. Especially, integrating data alignment and sophisticated TL approaches can significantly improve the classification performance, and hence greatly reduce the calibration effort.

Keywords: Brain-computer interface, EEG, transfer learning, Euclidean alignment, motor imagery

*Corresponding authors

Email addresses: drwu@hust.edu.cn (Dongrui Wu), rmpeng2019@hust.edu.cn (Ruimin Peng), huang_jan@hust.edu.cn (Jian Huang), zgzeng@hust.edu.cn (Zhigang Zeng)

1. Introduction

A brain-computer interface (BCI) [14, 38] enables a user to communicate directly with an external device, e.g., a computer, using his/her brain signals, e.g., electroencephalogram (EEG). It can benefit both patients [26] and able-bodied people [36, 23].

The flowchart of a closed-loop EEG-based BCI system is shown in Figure 1. It consists of the following main components [44]:

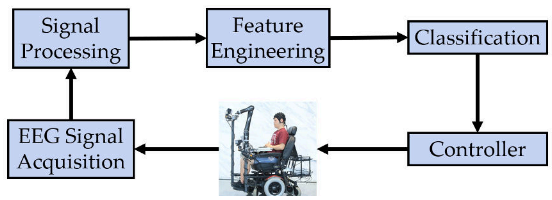


Figure 1: A closed-loop EEG-based BCI system, using motor imagery as an example.

1. *Signal acquisition*, which uses a headset to collect EEG signal from the scalp, while the user is performing MI tasks.
2. *Signal processing* [20]. Because EEG signals are weak, and easily contaminated by artifacts and interferences, e.g., muscle movements, eye blinks, heartbeats, powerline noise, etc., sophisticated signal processing approaches must be used to increase the signal-to-noise ratio. Both temporal filtering and spatial filtering are usually performed. Temporal filtering may include notch filtering to remove the 50Hz or 60Hz powerline interference, and then bandpass filtering, e.g., $[8, 30]$ Hz, to remove DC drift and high frequency noise. Spatial filters [40] include the basic ones, e.g., common average reference [35], Laplacian filters [13], principal component analysis [11], etc., and more sophisticated ones, e.g., independent component analysis [7], xDAWN [30], canonical correlation analysis [32], common spatial patterns (CSP) [28], etc.
3. *Feature engineering*, which includes feature extraction, and sometimes also feature selection. Time domain, frequency domain, time-frequency domain, Riemannian space [45], and/or topoplot features [16] could be used.

4. *Classification* [21], which uses a machine learning algorithm to decode the EEG signal from the extracted features. Commonly used classifiers include linear discriminant analysis (LDA) and support vector machine (SVM).
5. *Controller*, which sends a command to an external device, e.g., a wheelchair, according to the decoded EEG signal.

Motor imagery (MI) [27] is a common paradigm in EEG-based BCIs, and also the focus of this paper. In MI-based BCIs, the user imagines the movements of his/her body parts, which activates different areas of the motor cortex of the brain, e.g., top-left for right-hand MI, top-right for left-hand MI, and top-central for feet MI. A classification algorithm can then be used to decode the recorded EEG signals and map the corresponding MI to a command for the external device.

Because of individual differences and non-stationarity of EEG signals, an MI-based BCI usually needs a long calibration session for a new subject, from 20-30 minutes [33] to hours or even days. This lengthy calibration significantly reduces the utility of BCI systems. Hence, many sophisticated signal processing and machine learning approaches have been proposed recently to reduce or eliminate the calibration [6, 8, 9, 10, 31, 37, 39, 44, 46, 47].

One of the most promising such approaches is transfer learning (TL) [24], which uses data/knowledge from source domains (existing subjects) to help the calibration in the target domain (new subject). However, previous TL approaches for BCIs usually considered only one or two components of the closed-loop system in Figure 1, particularly, classification, as introduced in our latest survey [44]. For example, to consider TL in signal processing, Dai *et al.* [6] proposed transfer kernel CSP to integrate kernel CSP [1] and transfer kernel learning [19] for EEG trial filtering. To consider TL in classification, Jayaram *et al.* [10] proposed a multi-task learning (which is a subfield of TL) framework for cross-subject MI classification. To consider TL in feature engineering, Chen *et al.* [4] extended ReliefF [12] and minimum redundancy maximum relevancy (mRMR) [25] feature selection approaches to class separation and domain fusion (CSDF)-ReliefF and CSDF-mRMR, which optimized both the class separability and the domain similarity simultaneously. They then further integrated CSDF-ReliefF and CSDF-mRMR with an adaptation regularization-based TL classifier [18].

In this paper, we claim that TL should be considered in as many components of a BCI system as possible, and propose a complete TL pipeline for MI-based BCIs, shown in Figure 2:

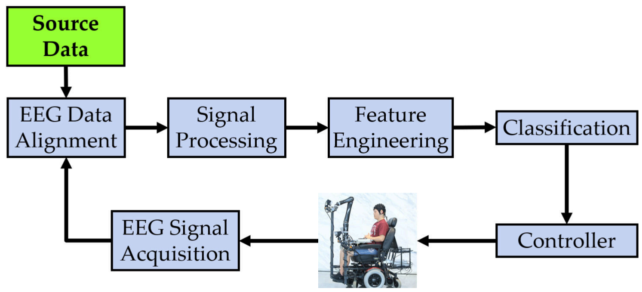


Figure 2: A complete TL pipeline for closed-loop MI-based BCI systems.

1. *Data alignment*, which aligns EEG trials from the source domains and the target domain so that their distributions are more consistent. This is a new component, which does not exist in Figure 1, but will greatly facilitate TL in sequential components.
2. *Signal processing*, where TL can be used to design better filters, especially when the amount of target domain labeled data is small.
3. *Feature engineering*, where TL may be used to extract or select more informative features.
4. *Classification*, where TL can be used to design better classifiers or regression models, especially when there are no or very few target domain labeled data.

We will introduce some representative TL approaches in data alignment, signal processing and classification, and demonstrate using two MI datasets that incorporate TL in all these components can indeed achieve better classification performance than not using TL, or using TL in only a subset of the components.

Our main contributions are:

1. We propose a complete TL pipeline for closed-loop MI-based BCI systems, as shown in Figure 2, and point out that explicitly including a data alignment component before signal processing is very important to the TL performance.

2. We verify through experiments that usually considering TL in more components in Figure 2 can result in better classification performance, and more sophisticated TL approaches are usually more beneficial than simple TL approaches, or not using TL at all.

The remainder of this paper is organized as follows: Section 2 introduces some representative TL approaches at different components of a BCI system. Section 3 evaluates the performance of the complete TL pipeline in offline MI classification. Section 4 discusses how TL could be used in feature engineering, an overlooked component so far, and to handle EEG non-stationarity. Finally, Section 5 draws conclusions and points out some future research directions.

2. TL Approaches

This section introduces the basic concepts of TL, and how TL could be used in data alignment, signal processing, and classification of an MI-based BCI system.

We consider offline binary classification only, and would like to use labeled EEG trials from a source subject to help classify trials from a target subject. When there are multiple source subjects, we can combine data from all source subjects and then view that as a single source subject, or perform TL for each source subject separately and then aggregate them.

Assume the source subject has N_s labeled samples $\{X_s^n, y_s^n\}_{n=1}^{N_s}$, where $X_s^n \in \mathbb{R}^{c\times t}$ is the n-th EEG trial and y_s^n the corresponding class label, in which c is the number of EEG channels, and t the number of time domain samples. The target subject has N_l labeled samples $\{X_t^n, y_t^n\}_{n=1}^{N_l}$, and N_u unlabeled samples $\{X_t^n\}_{n=N_l+1}^{N_l+N_u}$.

2.1. TL

TL [24] uses data/knowledge from a source domain to help solve a task in a target domain. A domain consists of a feature space $\mathcal{X}$ and its associated marginal probability distribution $P(X)$, i.e., $\{\mathcal{X}, P(X)\}$, where $X \in \mathcal{X}$. Two domains are different if they have different feature spaces, and/or different $P(X)$. A task consists of a label space $\mathcal{Y}$

and a prediction function $f(X)$, i.e., $\{\mathcal{Y}, f(X)\}$. Two tasks are different if they have different label spaces, and/or different conditional probability distributions $P(y|X)$.

For BCI calibration, TL usually means to use labeled EEG trials from an existing subject to help the calibration for a new subject. This paper considers the scenario that both subjects have the same feature space and label space, but different $P(X)$ and $P(y|X)$, i.e., the subjects perform the same MIs using the same BCI system. This is the most commonly encountered TL scenario in BCI calibration.

A very simple idea of TL in classifier training is illustrated in Figure 3. Assume the target domain has only four training samples belonging to two classes (represented by different shapes), whereas the source domain has more. Without TL, we can build a classifier in the target domain using only its own four training samples. Since the number of training samples is very small, this classifier is usually unreliable. With TL, we can combine samples from the source domain with those in the target domain to train a classifier. Since the two domains may not be completely consistent, e.g., the marginal probability distributions may be different, we may assign the source domain samples smaller weights than the target domain samples. If optimized properly, the resulting classifier can usually achieve better generalization performance.

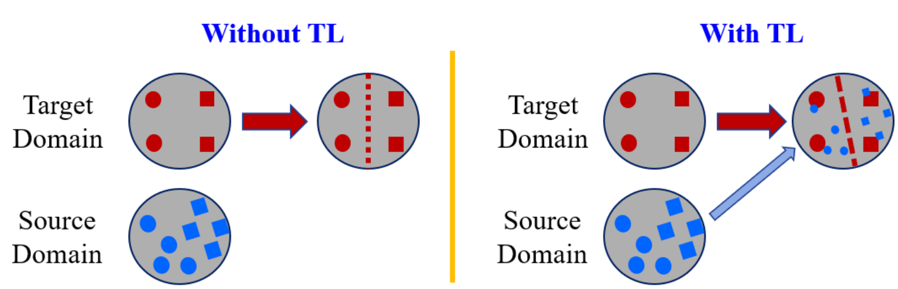


Figure 3: Illustration of simple TL in classification.

Figure 3 illustrates maybe the simplest TL approach in classification. Similar approaches may also be used in signal processing and feature engineering components in Figure 2. We will introduce some of them next.

2.2. Euclidean Alignment (EA)

Due to individual differences, the marginal probability distributions of the EEG trials from different subjects are usually (significantly) different; so, it is very beneficial to perform data alignment to make different domains more consistent, before other operations in Figure 2.

Different EEG trial alignment approaches have been proposed recently [8, 9, 31, 46, 47]. A summary and comparison of them is given in [44]. Among them, Euclidean alignment (EA), proposed by He and Wu [9] and illustrated in Figure 4, is easy to perform and completely unsupervised (does not need any labeled data from any subject). So, it is used as an example in this paper.

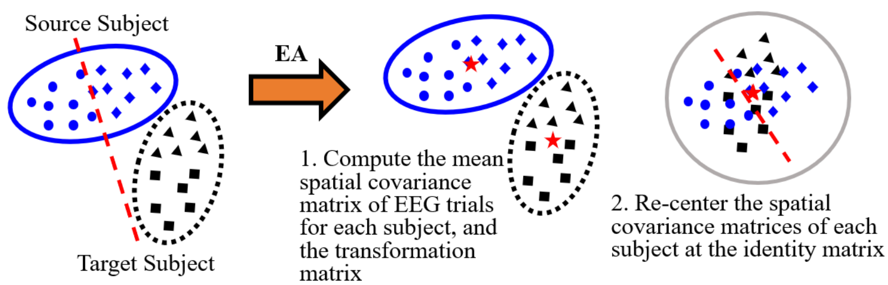


Figure 4: EA for aligning EEG trials from different subjects (domains).

For the source subject, EA first computes

$$\bar{R}_s = \frac{1}{N_s} \sum_{n=1}^{N_s} X_s^n \left(X_s^n\right)^\top , \tag{1}$$

i.e., the arithmetic mean of all spatial covariance matrices from the source subject, then performs the alignment by

$$\tilde{X}_s^n = \bar{R}_s^{-1/2} X_s^n. \tag{2}$$

Similarly, for the target subject, EA computes the arithmetic mean of all $N_l + N_u$ spatial covariance matrices and then performs the alignment.

After EA, the aligned EEG trials are whitened [47], and their mean spatial covariance matrix from each subject equals the identity matrix [9]; hence, the distributions of EEG trials from different subjects become more consistent. This will greatly benefit TL in subsequent steps.

2.3. CSP

CSP [3, 28] performs supervised spatial filtering for EEG trials, aiming to find a set of spatial filters to maximize the ratio of variance between two classes.

The traditional CSP uses data from the target subject only. For Class $k \in \{-1, 1\}$, CSP tries to find a spatial filter matrix $W_k^* \in \mathbb{R}^{c \times f}$, where f is the number of spatial filters, to maximize the variance ratio between Class k and Class $-k$:

$$W_k^* = \arg \max_{W \in \mathbb{R}^{c \times f}} \frac{\mathrm{tr}(W^\top \bar{C}_t^k W)}{\mathrm{tr}(W^\top \bar{C}_t^{-k} W)}, \tag{3}$$

where $\bar{C}_t^k \in \mathbb{R}^{c \times c}$ is the mean spatial covariance matrix of the N_l labeled EEG trials in Class k, and tr the trace of a matrix. The solution W_k^* is the concatenation of the f leading eigenvectors of $(\bar{C}_t^{-k})^{-1} \bar{C}_t^k$.

Then, CSP concatenates the $2f$ spatial filters from both classes to obtain the complete filter matrix:

$$W^* = \begin{bmatrix} W_{-1}^* & W_1^* \end{bmatrix} \in \mathbb{R}^{c \times 2f}, \tag{4}$$

and computes the spatially filtered X_t^n by:

$$\tilde{X}_t^n = W^{*\top} X_t^n \in \mathbb{R}^{2f \times t}. \tag{5}$$

Finally, the log-variances of $\tilde{X}_t^n$ can be extracted as features $\boldsymbol{x}_t^n \in \mathbb{R}^{1 \times 2f}$ in later classification:

$$\boldsymbol{x}_t^n = \log \left(\frac{\mathrm{diag}\left(\tilde{X}_t^n \left(\tilde{X}_t^n \right)^\top \right)}{\mathrm{tr}\left(\tilde{X}_t^n \left(\tilde{X}_t^n \right)^\top \right)} \right), \tag{6}$$

where diag means the diagonal elements of a matrix, and $\log$ is the logarithm operator.

2.4. Combined CSP (CCSP)

Because the target subject has very few labeled samples, i.e., N_l is small, W^* computed above may not be reliable. The source domain samples can be used to improve W^*.

In the combined CSP (CCSP), we simply concatenate the N_s source domain labeled samples and N_l target domain labeled samples to compute W^*. Note that all samples have the same weight, i.e., source domain and target domain samples are treated equally.

CCSP may be the simplest TL-based CSP approach.

2.5. Regularized CSP (RCSP)

Regularized CSP (RCSP) [22] was specifically proposed to handle the problem that the target domain has very few labeled samples. Though the original paper did not mention TL, it actually used the idea of TL.

RCSP computes the regularized average spatial covariance matrix for Class k as:

$$\hat{C}^k(\beta,\gamma) = (1-\gamma)\hat{C}^k(\beta) + \frac{\gamma}{c}\mathrm{tr}(\hat{C}^k(\beta))I, \tag{7}$$

where β and γ are two parameters in $[0,1]$, $I \in \mathbb{R}^{c\times c}$ is an identity matrix, and

$$\hat{C}^k(\beta) = \frac{\beta N_l \bar{C}_t^k + (1-\beta) N_s \bar{C}_s^k}{\beta N_l + (1-\beta) N_s}. \tag{8}$$

$\hat{C}^k(\beta,\gamma)$ can then be used to replace $\bar{C}_t^k$ in (3) to compute the CSP filter matrix.

Note that when $\beta = 1$ and $\gamma = 0$, RCSP becomes the traditional CSP. When $\beta = 0.5$ and $\gamma = 0$, RCSP becomes CCSP.

2.6. LDA

LDA is a popular linear classifier for binary classification. It assumes that the feature covariance matrices (not to be confused with the spatial covariance matrix of an EEG trial) from the two classes have full rank and are both equal to Σ_t. The classification for a new input $\boldsymbol{x}$ is then

$$\mathrm{sign}\left(\boldsymbol{w}\boldsymbol{x}^\top - \theta\right), \tag{9}$$

where

$$\boldsymbol{w} = \Sigma_t^{-1}(\bar{\boldsymbol{x}}_{t,1} - \bar{\boldsymbol{x}}_{t,-1}), \tag{10}$$

$$\boldsymbol{\theta} = \frac{1}{2}\boldsymbol{w}(\bar{\boldsymbol{x}}_{t,1} + \bar{\boldsymbol{x}}_{t,-1}), \tag{11}$$

in which $\bar{\boldsymbol{x}}_{t,-1}$ and $\bar{\boldsymbol{x}}_{t,1}$ are the mean feature vector of Class -1 and Class 1 computed from the N_l target domain labeled samples, respectively.

2.7. Combined LDA (CLDA)

When N_l is small, the above LDA classifier may not be reliable. The combined LDA (CLDA) is a simple TL approach, which concatenates labeled samples from both the source domain and the target domain to train an LDA classifier. All samples from both domains are treated equally.

2.8. wAR

Wu [39] proposed weighted adaptation regularization (wAR), a TL approach for cross-subject EEG trial classification. It can be used in both online and offline calibration. Though the original experiments were conducted for event-related potential classification, wAR can also be used for MI classification.

wAR learns a classifier f^* by minimizing the following regularized loss function:

$$\begin{aligned} f^* = \underset{f}{\operatorname{argmin}} & \sum_{n=1}^{N_s} w_s^n \ell(f(\boldsymbol{x}_s^n), y_s^n) + w_t \sum_{n=1}^{N_l} w_t^n \ell(f(\boldsymbol{x}_t^n), y_t^n) \\ & + \lambda_1 \|f\|_K^2 + \lambda_2 D_{f,K}(P_s(\boldsymbol{x}_s), P_t(\boldsymbol{x}_t)) \\ & + \lambda_3 D_{f,K}(P_s(\boldsymbol{x}_s|y_s), P_t(\boldsymbol{x}_t|y_t)) \end{aligned} \tag{12}$$

where ℓ is the classification loss, w_t is the overall weight of samples from the target subject, w_s^n and w_t^n are the weights for the n-th sample from the source subject and the target subject, respectively, K is a kernel function, $P_s(\boldsymbol{x}_s)$ and $P_t(\boldsymbol{x}_t)$ are the marginal probability distribution of features from the source subject and the target subject, respectively, $P_s(\boldsymbol{x}_s|y_s)$ and $P_t(\boldsymbol{x}_t|y_t)$ are the conditional probability distribution from the source subject and the target subject, respectively, and λ_1, λ_2 and λ_3 are non-negative regularization parameters.

Briefly speaking, the five terms in (12) minimize the classification loss for the source subject, the classification loss for the target subject, the structural risk of the classifier, the distance between the marginal probability distributions of the two subjects, and the distance between the conditional probability distributions of the two subjects, respectively.

Although it looks complicated, (12) has a closed-form solution when the squared loss $\ell(f(\boldsymbol{x}) - y) = (y - f(\boldsymbol{x}))^2$ is used [39].

3. Experiments and Results

This section evaluates the offline cross-subject calibration performances of different combinations of TL approaches on two MI datasets.

3.1. MI Datasets

Two MI datasets from BCI Competition IV[1] were used in this study. They were also used in our previous research [8, 9, 47].

In each experiment, the subject sat in front of a computer and performed visual cue based MI tasks, as shown in Figure 5. A fixation cross on the black screen ($t = 0$) prompted the subject to be prepared, and marked the start of a trial. After two seconds, a visual cue, which was an arrow pointing to a certain direction, was displayed for four seconds, during which the subject performed the instructed MI task. The visual cue disappeared at $t = 6$ second, and the MI also stopped. After a two-second break, the next trial started.

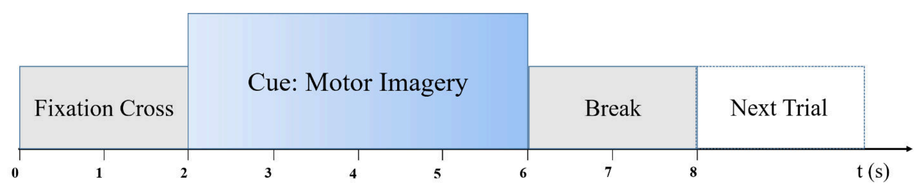


Figure 5: Timing scheme of the MI tasks.

The first dataset[2] (Dataset 1 [2]) consisted of seven healthy subjects. Each subject performed two types of MIs, selected from three classes: left-hand, right-hand, and foot. We used the 59-channel EEG data collected from the calibration phase, which included complete marker information. Each subject had 100 trials per class.

The second MI dataset[3] (Dataset 2a) included nine heathy subjects. Each subject performed four different MIs: left-hand, right-hand, both feet, and tongue. We used

[1] http://www.bbci.de/competition/iv/.
[2] http://www.bbci.de/competition/iv/desc_1.html.
[3] http://www.bbci.de/competition/iv/desc_2a.pdf.

the 22-channel EEG data and two classes of MIs (left-hand and right-hand) collected from the training phase. Each subject had 72 trials per class.

EEG data preprocessing steps were identical to those in [9]. A causal [8, 30] Hz band-pass filter was used to remove muscle artifacts, powerline noise, and DC drift. Next, we extracted EEG signals between $[0.5, 3.5]$ seconds after the cue onset as our trials for both datasets.

3.2. Algorithms

We compared the following 13 different algorithms, with various different configurations of TL components:

1. *CSP-LDA*, which used only the target domain labeled data to train the CSP filters, and then performed LDA classification. No source data was used at all, i.e., no TL was used at all.
2. *CSP-CLDA*, which used only the target domain labeled data to train the CSP filters, and then performed CLDA classification by using labeled data from both domains, i.e., only the classifier used a simple TL approach.
3. *CSP-wAR*, which used only the target domain labeled data to train the CSP filters, and then performed wAR classification by using data from both domains, i.e., only the classifier used a sophisticated TL approach.
4. *CCSP-CLDA*, which concatenated labeled data from both domains to train the CCSP filters and performed CLDA classification, i.e., both signal processing and classification used a simple TL approach.
5. *CCSP-wAR*, which concatenated labeled data from both domains to train the CCSP filters, and then performed wAR classification, i.e., signal processing used a simple TL approach, whereas classification used a sophisticated TL approach.
6. *RCSP-CLDA*, which used labeled data from both domains to train the RCSP filters, and then performed CLDA classification, i.e., signal processing used a sophisticated TL approach, whereas classification used a simple TL approach.
7. *RCSP-wAR*, which used labeled data from both domains to train the RCSP filters, and then performed wAR classification, i.e., both signal processing and classification used a sophisticated TL approach.

8. *EA-CSP-CLDA*, which performed EA before CSP-LDA, i.e., only the classifier used a simple TL approach, after EA.
9. *EA-CSP-wAR*, which performed EA before CSP-wAR, i.e., only the classifier used a sophisticated TL approach, after EA.
10. *EA-CCSP-CLDA*, which performed EA before CCSP-CLDA, i.e., both signal processing and classification used a simple TL approach, after EA.
11. *EA-CCSP-wAR*, which performed EA before CCSP-wAR, i.e., signal processing used a simple TL approach, whereas classification used a sophisticated TL approach, after EA.
12. *EA-RCSP-CLDA*, which performed EA before RCSP-CLDA, i.e., signal processing used a sophisticated TL approach, whereas classification used a simple TL approach, after EA.
13. *EA-RCSP-wAR*, which performed EA before RCSP-wAR, i.e., both signal processing and classification used a sophisticated TL approach, after EA.

Six (a typical number [29]) filters were used in all CSP algorithms. $\beta = \gamma = 0.1$ were used in RCSP. $w_t = 10$, $\lambda_1 = 0.1$, $\lambda_2 = \lambda_3 = 10$ were used in wAR, as in [39, 42], except that w_t was increased from 2 to 10 because the combined source domain has much more labeled samples than the target domain. All source code is available at https://github.com/drwuHUST/TLBCI.

By comparing between different pairs of the above algorithms, we can individually study the effect of TL in different components of Figure 2.

3.3. Experimental Settings and Results

For each dataset, we sequentially selected one subject as the target subject and all remaining ones as the source subjects, i.e., we performed cross-subject evaluations. As in [9], we combined all source subjects as a single source domain, and performed the corresponding TL. This procedure was repeated for each subject, so that each one became the target subject once.

The number of labeled samples in the target domain (N_l) increased from zero to 20, with a step of 4. We selected a random starting point in the EEG trial sequence of the target subject, and then sampled 20 trials from there continuously. Because there was randomness involved, we repeated this process 30 times and report the average results. Note that for algorithms whose signal processing component did not involve TL, e.g., those with *CSP*-, when $N_l = 0$, no CSP filters can be trained, and hence no model can be built. All other algorithms used TL in CSP, and hence the source domain labeled data can be used to train the CSP filters even when $N_l = 0$.

The cross-subject classification accuracies, averaged over 30 random runs, are shown in Figure 6. The average performances over all subjects are shown in the last panel of each subfigure.

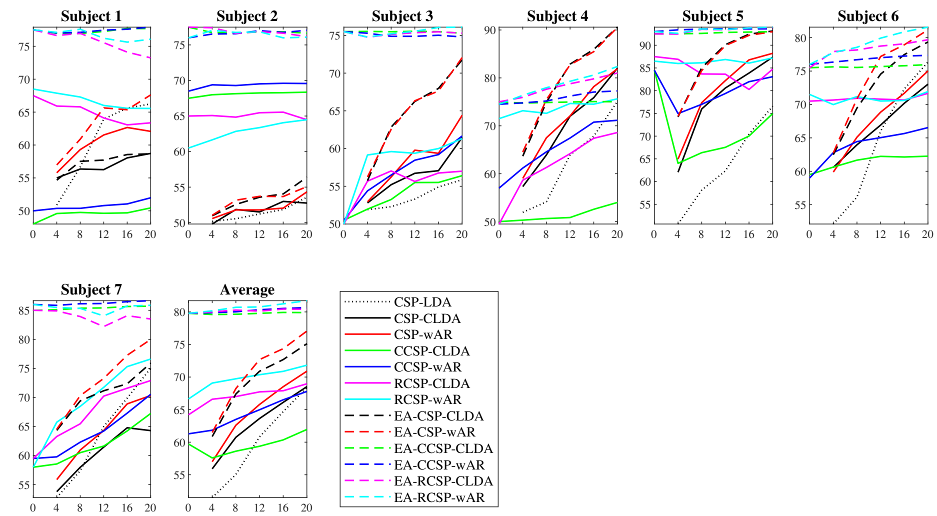
3.4. The General Effect of TL

In Figure 6, by comparing CSP-LDA, which did not use TL at all, with the other 12 algorithms, which used simple or sophisticated TL in one or more components of Figure 2, we can see that when N_l was small, TL almost always resulted in better performance, no matter how much TL was used. However, when N_l increased, CSP-LDA gradually outperformed certain simple TL approaches, e.g., CSP-CLDA and CCSP-CLDA, whereas sophisticated TL approaches, e.g., EA-RCSP-wAR, almost always outperformed CSP-LDA. These results suggest that sophisticated TL may always be beneficial.

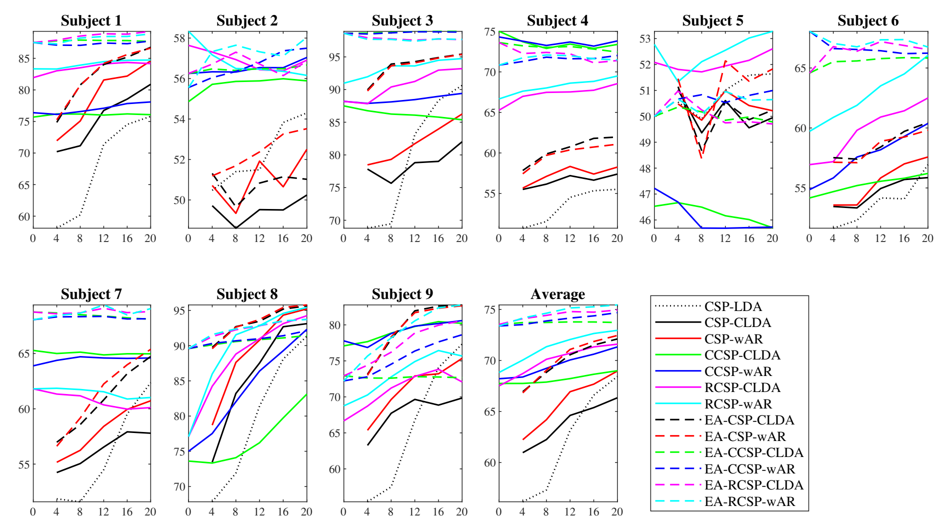
To quantitatively study the general effect of TL, we computed the mean classification accuracies of different approaches when N_l increased from 4 to 20 (we did not use $N_l = 0$ because certain approaches did not work in this case), and compared them with that of CSP-LDA. The results are shown in Table 1, which confirm again that generally more sophisticated TL approaches achieved more performance improvements.

3.5. The Effect of EA

In Figure 6, comparing algorithms without EA and their counterparts with EA, e.g., CSP-CLDA and EA-CSP-CLDA, we can observe that every EA version almost always significantly outperformed its non-EA counterpart, suggesting that a data alignment approach such as EA should always be included as a TL preprocessing step in a BCI system.



(a)



(b)

Figure 6: Offline cross-subject classification accuracies (vertical axis) on the MI datasets, with different N_l (horizontal axis). (a) Dataset 1; (b) Dataset 2a.

Table 1: Cross-subject classification accuracies of different TL approaches, and their improvements over CSP-LDA.

Algorithm	Dataset 1		Dataset 2a	
	Accuracy (%)	Improvement (%)	Accuracy (%)	Improvement (%)
CSP-CLDA	66.95	0.43	65.62	-1.75
CSP-wAR	69.31	3.96	68.12	1.99
CCSP-CLDA	61.91	-7.14	69.04	3.37
CCSP-wAR	67.56	1.33	71.13	6.50
RCSP-CLDA	68.92	3.38	71.10	6.46
RCSP-wAR	71.73	7.60	72.51	8.57
EA-CSP-CLDA	73.55	10.33	71.57	7.16
EA-CSP-wAR	75.19	12.78	71.89	7.65
EA-CCSP-CLDA	80.03	20.04	73.91	10.67
EA-CCSP-wAR	80.75	21.13	74.68	11.82
EA-RCSP-CLDA	80.58	20.87	74.91	12.17
EA-RCSP-wAR	81.59	22.38	75.34	12.81

To quantitatively verify the above conclusion, we also show the mean classification accuracies of algorithms without and with EA in Table 2. Clearly, EA significantly improved the classification accuracies, especially on Dataset 1.

Table 2: Mean cross-subject classification accuracies of algorithms without and with EA, and the improvements of the latter over the former.

Dataset	Algorithm	without EA (%)	with EA (%)	Improvement (%)
1	CSP-CLDA	66.95	73.55	9.86
	CSP-wAR	69.31	75.19	8.48
	CCSP-CLDA	61.91	80.03	29.28
	CCSP-wAR	67.56	80.75	19.53
	RCSP-CLDA	68.92	80.58	16.92
	RCSP-wAR	71.73	81.59	13.74
2a	CSP-CLDA	65.62	71.57	9.07
	CSP-wAR	68.12	71.89	5.54
	CCSP-CLDA	69.04	73.91	7.06
	CCSP-wAR	71.13	74.68	4.99
	RCSP-CLDA	71.10	74.91	5.36
	RCSP-wAR	72.51	75.34	3.90

3.6. The Effect of TL in Signal Processing

In Figure 6, comparing algorithms without TL in signal processing (CSP), with simple TL in signal processing (CCSP), and with sophisticated TL in signal processing (RCSP), e.g., CSP-CLDA, CCSP-CLDA and RCSP-CLDA, we can observe that simple TL in signal processing may not always work (e.g., CCSP-LDA had worse performance than CSP-LDA on Dataset 1, but better performance on Dataset 2a), but sophisticated TL in signal processing was almost always beneficial (e.g., RCSP-CLDA outperformed both CSP-CLDA and CCSP-CLDA on both datasets). So, sophisticated TL approaches, such as RCSP, should be used in signal processing in a BCI system.

To quantitatively verify the above conclusion, we also show the mean classification accuracies of algorithms without and with TL in signal processing in Table 3. Clearly, RCSP (sophisticated TL in signal processing) always outperformed the corresponding CSP (no TL in signal processing) and CCSP (simple TL in signal processing) versions.

Table 3: The effect of TL in signal processing.

Dataset	Algorithm	No TL	Simple TL		Sophisticated TL	
		CSP	CCSP	Imp. (%)	RCSP	Imp. (%)
1	CSP-CLDA	66.95	61.91	-7.54	68.92	2.94
	CSP-wAR	69.31	67.56	-2.53	71.73	3.50
	EA-CSP-CLDA	73.55	80.03	8.80	80.58	9.56
	EA-CSP-wAR	75.19	80.75	7.40	81.59	8.52
2a	CSP-CLDA	65.62	69.04	5.21	71.10	8.35
	CSP-wAR	68.12	71.13	4.42	72.51	6.45
	EA-CSP-CLDA	71.57	73.91	3.27	74.91	4.67
	EA-CSP-wAR	71.89	74.68	3.88	75.34	4.79

3.7. The Effect of TL in the Classifier

In Figure 6, comparing algorithms with simple and sophisticated TL in the classifier, e.g., CCSP-CLDA and CCSP-wAR, we can observe that sophisticated TL in the classifier almost always outperformed simple TL, regardless of whether TL was used in other components or not. So, sophisticated TL approaches, such as wAR, should be used in the classifier in a BCI system.

To quantitatively verify the above conclusion, we also show the mean classification accuracies of algorithms without and with TL in the classifier in Table 4. Clearly, on average wAR (sophisticated TL in the classifier) always outperformed CLDA (simple TL in the classifier).

Interestingly, when EA was used, the performance improvement of wAR over CLDA became smaller, because EA reduced the discrepancy between the source and target domain data, and hence made classification easier.

Table 4: The effect of TL in the classifier.

Dataset	Algorithm	Accuracy	Algorithm	Accuracy	Imp. (%)
1	CSP-CLDA	66.95	CSP-wAR	69.31	3.52
	CCSP-CLDA	61.91	CCSP-wAR	67.56	9.13
	RCSP-CLDA	68.92	RCSP-wAR	71.73	4.08
	EA-CSP-CLDA	73.55	EA-CSP-wAR	75.19	2.22
	EA-CCSP-CLDA	80.03	EA-CCSP-wAR	80.75	0.90
	EA-RCSP-CLDA	80.58	EA-RCSP-wAR	81.59	1.25
2a	CSP-CLDA	65.62	CSP-wAR	68.12	3.81
	CCSP-CLDA	69.04	CCSP-wAR	71.13	3.03
	RCSP-CLDA	71.10	RCSP-wAR	72.51	1.98
	EA-CSP-CLDA	71.57	EA-CSP-wAR	71.89	0.45
	EA-CCSP-CLDA	73.91	EA-CCSP-wAR	74.68	1.04
	EA-RCSP-CLDA	74.91	EA-RCSP-wAR	75.34	0.57

3.8. Summary

In summary, we can conclude that:

1. Generally, using TL in different components of Figure 2 can achieve better classification performance than not using it.
2. Generally, a more sophisticated TL approach outperformed a simple one.

3. Data alignment is a very important preprocessing step in TL.
4. TL in different components of Figure 2 could be complementary to each other, so integrating them can further improve the classification performance.

4. Discussion

Some additional discussions are provided in this section.

4.1. The Effect of TL in Feature Engineering

In Introduction, we mention that TL may also be used in the feature engineering block of the BCI system to improve its performance. However, we did not consider that in the previous section. This is because there were very few BCI feature engineering approaches that considered TL. In fact, to our knowledge, the only reference is [4], which proposed CSDF-ReliefF and CSDF-mRMR for EEG feature selection, by optimizing both the class separability and the domain similarity simultaneously. Unfortunately, we were not able to obtain improved performance using CSDF-ReliefF in our study. So, they are not introduced in this paper.

Nevertheless, we did perform experiments to show that considering TL in feature selection is more beneficial than not considering it at all, on Dataset 1. More specifically, we compared the following four algorithms:

1. *EA-RCSP6-wAR*, which was the best-performing EA-CSP-wAR algorithm in the previous section. RCSP trained six CSP filters.
2. *EA-RCSP20-wAR*, which was similar to EA-RCSP6-wAR, except that 20 RCSP filters were used.
3. *EA-RCSP20-ReliefF6-wAR*, which was similar to EA-RCSP20-wAR, except that after RCSP, ReliefF [12] was used to select the six best features, using labeled data from the target subject only.
4. *EA-RCSP20-CReliefF6-wAR*, which was similar to EA-RCSP20-ReliefF6-wAR, except that CReliefF was used to replace ReliefF. CReliefF combined labeled samples from both domains, and then performed ReliefF to select the six best features.

The cross-subject classification results are shown in Figure 7. We can observe that:

1. On average, EA-RCSP20-wAR achieved slightly worse performance than EA-RCSP6-wAR, suggesting that more CSP filters are not necessarily better. In practice, it is common to use only 6-10 CSP filters.
2. EA-RCSP20-ReliefF6-wAR almost always gave the worst performance, suggesting that using limited target domain labeled samples only in ReliefF was not adequate to select the best features.
3. EA-RCSP20-CReliefF6-wAR significantly outperformed EA-RCSP20-ReliefF6-wAR, suggesting that it is indeed beneficial to consider TL in feature selection, even though the TL idea is very simple.
4. On average, EA-RCSP20-CReliefF6-wAR had comparable performance as EA-RCSP6-wAR, maybe slightly better when N_l was large. The former used simple TL in ReliefF to select the six best features, whereas the latter used directly the six leading CSP filters, and hence was simpler to implement.

In summary, using TL in feature engineering is better than not using it at all; however, more research on more sophisticated TL in feature engineering is needed.

4.2. TL for Handling EEG Non-stationarity

It is well-known that EEG signals are non-stationary [17], i.e., EEG responses to the same stimulus from the same subject in different sessions are usually varying. This subsection evaluates how the proposed TL pipeline can be used to handle EEG non-stationarity in MI-based BCIs.

More specifically, we performed cross-session MI classification, in contrast to cross-subject classifications in the previous section. In Dataset 2a, each of the nine subjects had two sessions (training and evaluation), collected on different days. For each subject, we used the training session as the source domain, and the evaluation session as the target domain. Other experimental settings were identical to those in the previous section, except that we used $w_t = 2$ in wAR, as in cross-session TL the number of labeled source domain samples were much smaller than those in cross-subject TL.

The results are shown in Figure 8. Some subjects, e.g., Subjects 1, 3, 7, 8 and 9, demonstrated very stable classification performance when N_l increased, indicating

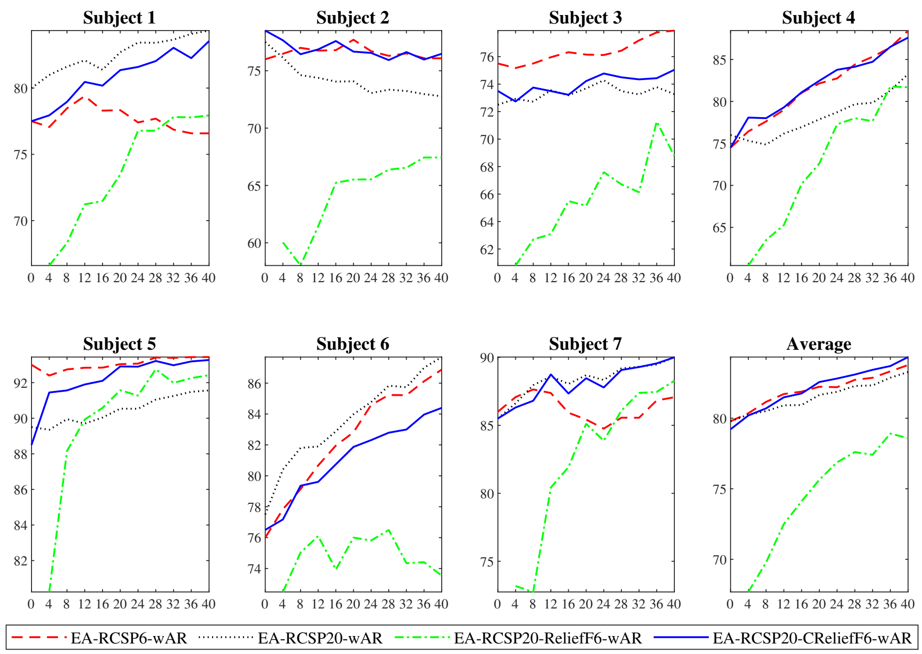


Figure 7: Offline cross-subject classification accuracies (vertical axis) on Dataset 1, with different N_l (horizontal axis), when TL in feature engineering was considered.

that their EEG signals were quite stationary, at least in the two experimental sessions. However, the remaining four subjects's EEG signals were more non-stationary, and hence the classification performance had large variations.

EA-CCSP-wAR, which integrates data alignment and TL in two components (signal processing and classification), achieved the best average performance. EA-RCSP-wAR, which was the best performer in cross-subject transfers in the previous section, was slightly worse, but still better than 10 other approaches with or without TL. On average, every approach with EA always outperformed its counterpart without EA, suggesting again the importance and necessity of explicitly adding a data alignment block before TL.

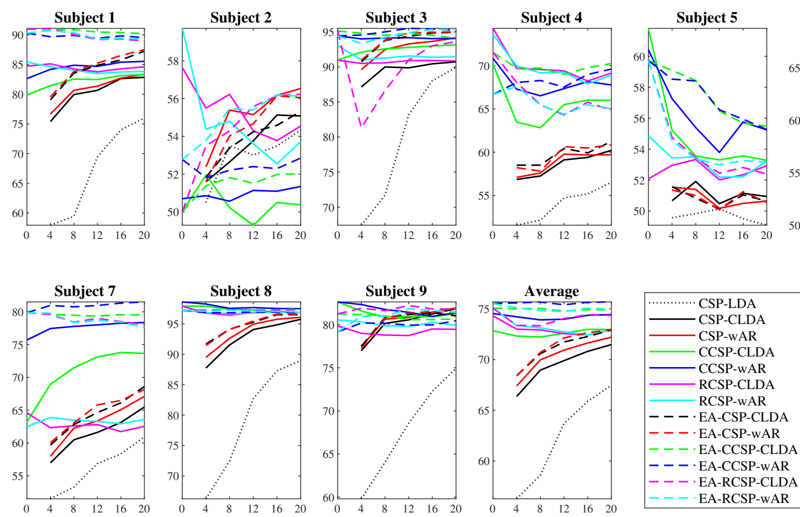


Figure 8: Offline cross-session classification accuracies (vertical axis) on Dataset 2a, with different N_l (horizontal axis).

In summary, TL is also effective in handling non-stationarity of EEG signals in MI-based BCIs, and considering TL in more components of the classification pipeline is generally more beneficial.

5. Conclusions and Future Research

Transfer learning has been widely used in MI-based BCIs to reduce the calibration effort for a new subject, and demonstrated promising performance. A closed-loop MI-based BCI system includes signal processing, feature engineering, and classification blocks before sending out the control signal to an external device, whereas previous research only considered TL in one or two such components. This paper proposes that TL could be considered in all three components, and it is also very important to specifically add a data alignment component before signal processing to make the source domain and target domain data more consistent. Offline calibration experiments on two MI datasets verified our proposal.

The following directions will be considered in our future research:

1. As mentioned in Section 4, compared with other components, not enough attention has been paid to TL in feature engineering of the BCI system. We will develop more sophisticated TL approaches for feature engineering in the future, and also other components in Figure 2.
2. Deep learning has started to find successful applications in BCIs [15, 34]. It's interesting to study if data alignment can also significantly benefit deep learning, and how to better use TL in deep learning, in addition to the currently widely used fine-tuning approach.
3. This paper considers only offline MI classification problems in BCIs. We will extend the analysis to other BCI classification paradigms, e.g., event-related potentials and steady-state visual evoked potentials, and also BCI regression problems, e.g., driver drowsiness estimation [5, 41] and user reaction-time estimation [43]. Furthermore, we will also consider TL in online calibration.
4. It has been shown [42] that integrating TL with active learning in the classifier can further improve the offline classification performance. It is interesting to study if TL and active learning can be integrated in other components of the BCI system, e.g., signal processing and feature engineering.

Acknowledgements

This research was supported by the Hubei Province Funds for Distinguished Young Scholars (2020CFA050), the National Natural Science Foundation of China (U1913207 and 61873321), the Hubei Technology Innovation Platform (2019AEA171), and the International Science and Technology Cooperation Program of China (2017YFE0128300).

References

[1] H. Albalawi, X. Song, A study of kernel CSP-based motor imagery brain computer interface classification, in: Proc. IEEE Signal Processing in Medicine and Biology Symposium, New York City, NY, 1–4, 2012.

[2] B. Blankertz, G. Dornhege, M. Krauledat, K. R. Muller, G. Curio, The non-invasive Berlin Brain-Computer Interface: Fast acquisition of effective performance in untrained subjects, NeuroImage 37 (2) (2007) 539–550.

[3] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, K. R. Muller, Optimizing Spatial filters for Robust EEG Single-Trial Analysis, IEEE Signal Processing Magazine 25 (1) (2008) 41–56.

[4] L.-L. Chen, A. Zhang, X.-G. Lou, Cross-subject driver status detection from physiological signals based on hybrid feature selection and transfer learning, Expert Systems with Applications 137 (2019) 266–280.

[5] Y. Cui, Y. Xu, D. Wu, EEG-Based Driver Drowsiness Estimation Using Feature Weighted Episodic Training, IEEE Trans. on Neural Systems and Rehabilitation Engineering 27 (11) (2019) 2263–2273.

[6] M. Dai, D. Zheng, S. Liu, P. Zhang, Transfer kernel common spatial patterns for motor imagery brain-computer interface classification, Computational and Mathematical Methods in Medicine 2018.

[7] A. Delorme, S. Makeig, EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis, Journal of Neuroscience Methods 134 (2004) 9–21.

[8] H. He, D. Wu, Different Set Domain Adaptation for Brain-Computer Interfaces: A Label Alignment Approach, IEEE Trans. on Neural Systems and Rehabilitation Engineering 28 (5) (2020) 1091–1108.

[9] H. He, D. Wu, Transfer Learning for Brain-Computer Interfaces: A Euclidean Space Data Alignment Approach, IEEE Trans. on Biomedical Engineering 67 (2) (2020) 399–410.

[10] V. Jayaram, M. Alamgir, Y. Altun, B. Scholkopf, M. Grosse-Wentrup, Transfer learning in brain-computer interfaces, IEEE Computational Intelligence Magazine 11 (1) (2016) 20–31.

[11] I. Jolliffe, Principal Component Analysis, Wiley Online Library, 2002.

[12] I. Kononenko, Estimating attributes: Analysis and extensions of RELIEF, in: Proc. European Conf. on Machine Learning, Catania, Italy, 171–182, 1994.

[13] T. D. Lagerlund, F. W. Sharbrough, N. E. Busacker, Spatial Filtering of Multichannel Electroencephalographic Recordings Through Principal Component Analysis by Singular Value Decomposition, Journal of Clinical Neurophysiology 14 (1) (1997) 73–82.

[14] B. J. Lance, S. E. Kerick, A. J. Ries, K. S. Oie, K. McDowell, Brain-Computer Interface Technologies in the Coming Decades, Proc. of the IEEE 100 (3) (2012) 1585–1599.

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

[16] C.-T. Lin, C. Chuang, Y. Hung, C. Fang, D. Wu, Y.-K. Wang, A Driving Performance Forecasting System Based on Brain Dynamic State Analysis Using 4-D Convolutional Neural Networks, IEEE Trans. on Cybernetics In press.

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

[18] M. Long, J. Wang, G. Ding, S. J. Pan, P. S. Yu, Adaptation Regularization: A General Framework for Transfer Learning, IEEE Trans. on Knowledge and Data Engineering 26 (5) (2014) 1076–1089.

[19] M. Long, J. Wang, J. Sun, S. Y. Philip, Domain invariant transfer kernel learning, IEEE Trans. on Knowledge and Data Engineering 27 (6) (2015) 1519–1532.

[20] F. Lotte, Signal Processing Approaches to Minimize or Suppress Calibration Time in Oscillatory Activity-Based Brain-Computer Interfaces, Proc. of the IEEE 103 (6) (2015) 871–890.

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

[22] H. Lu, H.-L. Eng, C. Guan, K. N. Plataniotis, A. N. Venetsanopoulos, Regularized common spatial pattern with aggregation for EEG classification in small-sample setting, IEEE Trans. on Biomedical Engineering 57 (12) (2010) 2936–2946.

[23] L. F. Nicolas-Alonso, J. Gomez-Gil, Brain computer interfaces, a review, Sensors 12 (2) (2012) 1211–1279.

[24] S. J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. on Knowledge and Data Engineering 22 (10) (2010) 1345–1359.

[25] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. on Pattern Analysis and Machine Intelligence 27 (8) (2005) 1226–1238.

[26] G. Pfurtscheller, G. R. Müller-Putz, R. Scherer, C. Neuper, Rehabilitation with brain-computer interface systems, Computer 41 (10) (2008) 58–65.

[27] G. Pfurtscheller, C. Neuper, Motor imagery and direct brain-computer communication, Proc. of the IEEE 89 (7) (2001) 1123–1134.

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

[29] R. P. Rao, R. Scherer, Chapter 10 - Statistical Pattern Recognition and Machine Learning in Brain-Computer Interfaces, in: K. G. Oweiss (Ed.), Statistical Signal Processing for Neuroscience and Neurotechnology, Academic Press, Oxford, 335–367, 2010.

[30] B. Rivet, A. Souloumiac, V. Attina, G. Gibert, xDAWN algorithm to enhance evoked potentials: application to brain-computer interface, IEEE Trans. on Biomedical Engineering 56 (8) (2009) 2035–2043.

[31] P. L. C. Rodrigues, C. Jutten, M. Congedo, Riemannian Procrustes Analysis: Transfer Learning for Brain–Computer Interfaces, IEEE Trans. on Biomedical Engineering 66 (8) (2019) 2390–2401.

[32] R. N. Roy, S. Bonnet, S. Charbonnier, P. Jallon, A. Campagne, A comparison of ERP spatial filtering methods for optimal mental workload estimation, in: Proc. 37th Annual Int'l Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC), 7254–7257, 2015.

[33] S. Saha, K. I. U. Ahmed, R. Mostafa, L. Hadjileontiadis, A. Khandoker, Evidence of Variabilities in EEG Dynamics During Motor Imagery-Based Multiclass Brain–Computer Interface, IEEE Trans. on Neural Systems and Rehabilitation Engineering 26 (2) (2018) 371–382.

[34] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, T. Ball, Deep learning with convolutional neural networks for EEG decoding and visualization, Human Brain Mapping 38 (11) (2017) 5391–5420.

[35] M. Teplan, Fundamentals of EEG measurement, Measurement Science Review 2 (2) (2002) 1–11.

[36] J. van Erp, F. Lotte, M. Tangermann, Brain-Computer Interfaces: Beyond Medical Applications, Computer 45 (4) (2012) 26–34, ISSN 0018-9162.

[37] P. Wang, J. Lu, B. Zhang, Z. Tang, A Review on Transfer Learning for Brain-Computer Interface Classification, in: Proc. 5th Int'l Conf. on Information Science and Technology, Changsha, China, 2015.

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

[39] D. Wu, Online and Offline Domain Adaptation for Reducing BCI Calibration Effort, IEEE Trans. on Human-Machine Systems 47 (4) (2017) 550–563.

[40] D. Wu, J.-T. King, C.-H. Chuang, C.-T. Lin, T.-P. Jung, Spatial Filtering for EEG-Based Regression Problems in Brain-Computer Interface (BCI), IEEE Trans. on Fuzzy Systems 26 (2) (2018) 771–781.

[41] D. Wu, V. J. Lawhern, S. Gordon, B. J. Lance, C.-T. Lin, Driver Drowsiness Estimation from EEG Signals Using Online Weighted Adaptation Regularization for Regression (OwARR), IEEE Trans. on Fuzzy Systems 25 (6) (2017) 1522–1535.

[42] D. Wu, V. J. Lawhern, W. D. Hairston, B. J. Lance, Switching EEG headsets made easy: Reducing offline calibration effort using active wighted adaptation regularization, IEEE Trans. on Neural Systems and Rehabilitation Engineering 24 (11) (2016) 1125–1137.

[43] D. Wu, V. J. Lawhern, B. J. Lance, S. Gordon, T.-P. Jung, C.-T. Lin, EEG-Based User Reaction Time Estimation Using Riemannian Geometry Features, IEEE Trans. on Neural Systems and Rehabilitation Engineering 25 (11) (2017) 2157–2168.

[44] D. Wu, Y. Xu, B.-L. Lu, Transfer Learning for EEG-Based Brain-Computer Interfaces: A Review of Progress Made Since 2016, IEEE Trans. on Cognitive and Developmental Systems In press.

[45] F. Yger, M. Berar, F. Lotte, Riemannian approaches in brain-computer interfaces: a review, IEEE Trans. on Neural Systems and Rehabilitation Engineering 25 (10) (2017) 1753–1762.

[46] P. Zanini, M. Congedo, C. Jutten, S. Said, Y. Berthoumieu, Transfer Learning: a Riemannian geometry framework with applications to Brain-Computer Interfaces, IEEE Trans. on Biomedical Engineering 65 (5) (2018) 1107–1116.

[47] W. Zhang, D. Wu, Manifold Embedded Knowledge Transfer for Brain-Computer Interfaces, IEEE Trans. on Neural Systems and Rehabilitation Engineering 28 (5) (2020) 1117–1127.