

Different Set Domain Adaptation for Brain-Computer Interfaces: A Label Alignment Approach

He He^{id} and Dongrui Wu^{id}, *Senior Member, IEEE*

Abstract—A brain-computer interface (BCI) system usually needs a long calibration session for each new subject/task to adjust its parameters, which impedes its transition from the laboratory to real-world applications. Domain adaptation, which leverages labeled data from auxiliary subjects/tasks (source domains), has demonstrated its effectiveness in reducing such calibration effort. Currently, most domain adaptation approaches require the source domains to have the same feature space and label space as the target domain, which limits their applications, as the auxiliary data may have different feature spaces and/or different label spaces. This paper considers different set domain adaptation for BCIs, i.e., the source and target domains have different label spaces. We introduce a practical setting of different label sets for BCIs, and propose a novel label alignment (LA) approach to align the source label space with the target label space. It has three desirable properties: 1) LA only needs as few as one labeled sample from each class of the target subject; 2) LA can be used as a preprocessing step before different feature extraction and classification algorithms; and, 3) LA can be integrated with other domain adaptation approaches to achieve even better performance. Experiments on two motor imagery datasets demonstrated the effectiveness of LA.

Index Terms—Brain-computer interface, EEG, label alignment, Riemannian geometry, domain adaptation, transfer learning.

I. INTRODUCTION

A BRAIN-COMPUTER interface (BCI) system [1], [2] acquires the brain signal, decodes it, and then translates it into control commands for external devices, so that a user can interact with his/her surroundings using thoughts directly, bypassing the normal pathway of peripheral nerves and muscles. Electroencephalogram (EEG) may be the most popular BCI input signal due to its convenience, safety, and

Manuscript received September 25, 2019; revised December 29, 2019, February 11, 2020, and March 6, 2020; accepted March 9, 2020. Date of publication March 12, 2020; date of current version May 8, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61873321 and in part by the Hubei Technology Innovation Platform under Grant 2019AEA171. (Corresponding author: Dongrui Wu.)

The authors are with the Ministry of Education Key Laboratory of Image Processing and Intelligent Control, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: hehe91@hust.edu.cn; drwu@hust.edu.cn).

Digital Object Identifier 10.1109/TNSRE.2020.2980299

low cost. The pipeline for decoding EEG signals usually involves:

- 1) *Signal processing*, which includes band-pass filtering and spatial filtering. Bandpass filtering reduces interferences and noise such as muscle artifacts, eye blinks, and DC drift. Spatial filtering combines different EEG channels to increase the signal-to-noise ratio. Common spatial patterns (CSP) [3]–[6] may be the most frequently used spatial filtering approach.
- 2) *Feature extraction*. Different features, e.g., time domain, frequency domain, time-frequency domain, Riemannian space, could be used.
- 3) *Classification*. Popular classifiers include linear discriminant analysis (LDA) and support vector machine (SVM).

Recently, Barachant *et al.* [7] proposed a novel preprocessing and classification pipeline in the Riemannian space, which integrated spatial filtering and feature extraction into one single step. This Riemannian pipeline uses the covariance matrices of the EEG trials, which are symmetric positive definite and lie on a Riemannian manifold [8]. The covariance matrices encode spatial information of the brain activities, which are useful in many BCI tasks. A popular classifier in the Riemannian space, minimum distance to mean [7], treats the covariance matrices as points on the Riemannian manifold, and uses their Riemannian distances to the class mean for classification. Another more sophisticated approach maps the covariance matrices from the Riemannian space to a Euclidean tangent space (TS) around the Riemannian mean, where the Riemannian space covariance matrices are transformed into Euclidean space vectors, and then used in Euclidean space classifiers as features.

Motor imagery [9] is one of the most frequently used paradigms of BCIs. It is based on the voluntary modulation of the sensorimotor rhythm, which does not need any external stimuli. The imagined movements of different body parts (e.g., hands, feet, and tongue) cause modulations of brain rhythms in the involved cortical areas. So, they can be distinguished by decoding such brain rhythm modulations, and used to control external devices such as powered exoskeletons, wheelchairs, and robots.

Motor imagery-based BCIs were originally designed to help those with neuromuscular impairments [10]. Recent research has extended its application scope to able-bodied users [11], [12]. However, EEG signals are very weak, and easily contaminated by interferences and noise. Moreover, individual differences make it difficult, if not impossible, to build a generic machine learning model optimal for all subjects. Usually a calibration session is needed to collect some subject-specific data for a new subject, which is time-consuming and user-unfriendly.

Researchers have proposed many different approaches [13]–[20] to reduce this calibration effort. One of them is transfer learning [21], or domain adaptation (DA). Its main idea is to leverage the data from auxiliary subjects (called source subjects or source domains) to improve the learning performance for a new subject (called target subject or target domain). A popular idea in DA is to project the source domain and target domain data into low dimensional subspaces where the geometrical shift or/and distribution shift are reduced, such as joint distribution adaptation (JDA) [22], joint geometrical and statistical alignment (JGSA) [23], and manifold embedded distribution alignment (MEDA) [24]. Computational intelligence techniques have also been used in transfer learning, as reviewed by Lu *et al.* [25]. In BCIs, Zanini *et al.* [26] proposed a Riemannian geometry framework to align EEG covariance matrices from different subjects in the Riemannian space. Recently, we [27] proposed a Euclidean alignment (EA) approach, which can be used as a preprocessing step before many Euclidean space feature extraction and pattern recognition algorithms.

However, most existing DA approaches assume that the source domains have the same feature space and label space as the target domain, which may not hold in many real-world applications. There have been some heterogeneous feature space DA approaches [14], [28], [29], which address the problem that the source domains have different feature spaces from the target domain. For example, in BCIs, Wu *et al.* [14] performed transfer learning for heterogeneous feature spaces: the source and target EEG trials are collected from different EEG headsets, with different numbers of channels and channel locations. Its main idea is to select the source domain channels closest to the target domain channels.

There have also been a few heterogeneous label space DA approaches [30]–[33], as shown in Fig. 1. Busto and Gall [30] first proposed the concept of open set DA, assuming the source and target domains have some known classes in common, and also some classes that are different and unknown. Saito *et al.* [31] considered the case that the target domain contains all classes in the source domain, plus an “unknown” class (different from [30], herein the source domain does not contain an “unknown” class). You *et al.* [33] proposed universal DA, which classifies a target domain sample if it belongs to any known class in the source domain, or marks it as “unknown” otherwise. In summary, both open set DA and universal DA train a model to either classify a target domain sample into a known class which has appeared in the source domain, or mark it as “unknown”. An application scenario of open set DA and universal DA is face recognition, where some

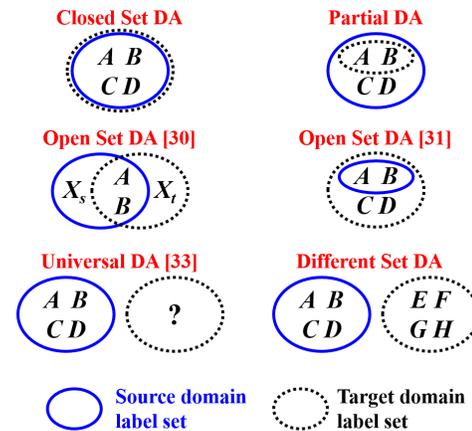


Fig. 1. Different DA scenarios. A, B, \dots, H represent different classes. X_S and X_T are unknown classes in the source domain and the target domain, respectively. In closed set DA, the source domain and target domain have the same classes. In partial DA, the target domain only contains a subset of the source domain classes. In open set DA considered in [30], the source and target domains contain some common classes, but each also contains an “unknown” class. In open set DA considered in [31], the source domain only contains a subset of the target domain classes. In universal DA [33], the target domain may contain some common classes as the target domain, but may also contain some unknown classes. In different set DA considered in this paper, the target domain contains partially or completely different classes from the source domain.

test samples may not appear in the training database and have to be marked as “unknown”.

This paper considers different set DA in BCIs, i.e., the source domains have different label spaces from the target domain, as shown in Fig. 1. For Motor imagery-based BCIs, this means the source subjects and the target subject perform different motor imagery tasks. To our knowledge, no one has studied this problem before.

To address this issue, we propose a label alignment (LA) approach to align EEG covariance matrices of the source subjects to those of the target subject. It first matches each source domain label with a target domain label, then moves the per-class covariance matrices of each source subject to re-center them at the corresponding class means of the target subject. After LA, the distribution discrepancies between the source and the target subjects are reduced, so that a model trained on source subjects can classify each target trial into the category it actually belongs to, even though the source and target subjects have completely different label spaces.

The main contributions of this paper are:

- 1) We introduce a practical setting of different set DA in BCIs: The source and target domains have known and different label sets; we need to classify each target trial into the category it actually belongs to, with the help of the source domain data. This setting is different from existing open set DA and universal DA. To our knowledge, it has not been studied before.
- 2) We propose an effective LA approach for different set DA in BCIs, which has three desirable properties: 1) It only needs as few as one labeled EEG trial from each class of the target subject; 2) It can be used as a preprocessing step in different feature extraction

and classification algorithms; and, 3) It can be integrated with other DA approaches to achieve even better performance.

The remainder of this paper is organized as follows: Section II introduces related background knowledge on the Riemannian space and the EA. Section III proposes the LA. Section IV introduces the datasets used in our experiments. Section V compares the performance of LA with several other DA approaches. Finally, Section VI draws conclusions and points out some future research directions.

II. RELATED WORK

This section introduces some basic concepts of the Riemannian space and its TS, and the EA, a state-of-the-art data alignment approach for BCIs, which also motivated our proposed LA.

A. Riemannian Distance

Each symmetric positive definite matrix can be viewed as a point on a Riemannian manifold. The Riemannian distance between two symmetric positive definite matrices P_1 and P_2 is the length of the *geodesic*, defined as the minimum length curve connecting P_1 and P_2 on the Riemannian manifold:

$$\delta(P_1, P_2) = \|\log(P_1^{-1} P_2)\|_F = \left[\sum_{r=1}^R \log^2 \lambda_r \right]^{\frac{1}{2}}, \quad (1)$$

where the subscript F denotes the Frobenius norm, and λ_r ($r = 1, 2, \dots, R$) are the real eigenvalues of $P_1^{-1} P_2$.

$\delta(P_1, P_2)$ remains unchanged under linear invertible transformations:

$$\delta(W^T P_1 W, W^T P_2 W) = \delta(P_1, P_2), \quad (2)$$

where W is an invertible matrix. This property, called *congruence invariance*, is useful in both EA and LA.

B. Tangent Space (TS) Mapping

Most machine learning approaches are applicable only in the Euclidean space, and cannot be used in the Riemannian space. TS mapping maps the covariance matrices from the Riemannian space to a Euclidean TS, so that they can be used by a Euclidean space classifier.

For each point P on the Riemannian manifold, the TS can be defined by a set of tangent vectors at P . Each tangent vector S_i is defined as the derivative at $t = 0$ of the geodesic between P and the exponential mapping $P_i = \text{Exp}_P(S_i)$:

$$\text{Exp}_P(S_i) = P_i = P^{\frac{1}{2}} \exp(P^{-\frac{1}{2}} S_i P^{-\frac{1}{2}}) P^{\frac{1}{2}}. \quad (3)$$

The inverse mapping is given by the logarithmic mapping:

$$\text{Log}_P(P_i) = S_i = P^{\frac{1}{2}} \log(P^{-\frac{1}{2}} P_i P^{-\frac{1}{2}}) P^{\frac{1}{2}}. \quad (4)$$

TS mapping converts each 2D EEG trial into a 1D feature vector, so that many machine learning algorithms can be used.

C. Euclidean Alignment (EA)

EA [27], [34] is a state-of-the-art DA approach for BCIs, which reduces the individual differences by aligning the EEG covariance matrices.

Some DA approaches [14], [22] first find a proper discrepancy measure between different distributions, then learn a shared subspace where the distribution discrepancy is explicitly minimized. Maximum mean discrepancy [35] is a popular distribution discrepancy measure, which is defined as the distance between the mean feature embeddings of different distributions.

Similar to these maximum mean discrepancy based DA approaches, EA views the covariance matrices as the feature embeddings of different EEG trials, and finds projections to minimize the distance between the mean covariance matrices of different subjects.

For a subject with N trials $\{X_i\}_{i=1}^N$ (each row of X_i is an EEG channel), EA first computes the individual covariance matrices

$$C_i = X_i X_i^T, \quad i = 1, 2, \dots, N \quad (5)$$

and the mean covariance matrix

$$\bar{C} = \frac{1}{N} \sum_{i=1}^N C_i. \quad (6)$$

The projection matrix for the subject is then

$$R = \bar{C}^{-1/2}. \quad (7)$$

Finally, EA performs the following projection for each trial:

$$\tilde{X}_i = R X_i, \quad i = 1, 2, \dots, N. \quad (8)$$

After EA, the mean covariance matrix of the subject becomes an identity matrix:

$$\begin{aligned} \frac{1}{N} \sum_{i=1}^N \tilde{X}_i \tilde{X}_i^T &= \frac{1}{N} \sum_{i=1}^N R X_i X_i^T R \\ &= \bar{C}^{-1/2} \left(\frac{1}{N} \sum_{i=1}^N X_i X_i^T \right) \bar{C}^{-1/2} \\ &= \bar{C}^{-1/2} \bar{C} \bar{C}^{-1/2} = I. \end{aligned} \quad (9)$$

After performing EA for all subjects, they share the same mean covariance matrix, i.e., the distances between the mean covariance matrices of different subjects are minimized (they become zero), and hence data distributions from different subjects become more similar.

We can also understand EA as a correction of data shift. If we view each EEG covariance matrix as a point on a Riemannian manifold, then individual differences cause shifts of these points, although they may entail more than just a simple displacement [26]. In order to correct this shift, EA moves the covariance matrices of each subject to center them at the identity matrix. The *congruence invariance* property makes sure that the distances among the within-subject covariance matrices remain unchanged. So, EA makes the data distributions from different subjects closer, while preserving the local distance information of each subject.

III. LABEL ALIGNMENT (LA) FOR DIFFERENT SET DA

This section introduces our proposed LA for different set DA, and discusses its relationship with EA and CORAL [36].

A. LA

Generally, there are three types of data shift in transfer learning:

- 1) *Covariate shift* [37], [38]: the distributions of the inputs (features) are different.
- 2) *Prior probability shift*: the distributions of the output are different.
- 3) *Concept shift* [39]: the relationships between the inputs and the output are different.

EA considers only the covariate shift but ignores the other two. Although it has been shown to significantly improve the cross-subject classification performance in [27], it only aligns the data in the feature space, and may not work well when the source subjects and the target subject have different label spaces.

This section proposes LA, which extends EA to different label spaces, by simultaneously considering multiple types of data shift. Its main idea is to independently move the per-class covariance matrices of each source subject, to re-center them at the corresponding class center of the target subject.

More specifically, for an M -class classification problem, we assume the source and target subjects have the same number of classes, but their class labels are partially or completely different. Our goal is to use the source data to help the classification of the target trials. LA seeks a transformation matrix A_m for the trials of the m -th class ($m = 1, 2, \dots, M$) from the source subject, such that the distance between the mean covariance matrices of the corresponding class in different domains are minimized:

$$A_m = \arg \min_A \|A \bar{C}_{S,m} A^T - \bar{C}_{T,m}\|_F^2, \quad m = 1, 2, \dots, M. \quad (10)$$

where $\bar{C}_{S,m}$ is the mean covariance matrix of the m -th class of the source subject, and $\bar{C}_{T,m}$ the mean covariance matrix of the m -th class of the target subject. In this paper, we use the *Log-Euclidean mean* [40], which is frequently used for symmetric positive definite matrices and much easier to compute than the Riemannian mean.

We adopt the optimization approach in [36] to solve for A_m in (10):

$$A_m = \bar{C}_{T,m}^{-\frac{1}{2}} \bar{C}_{S,m}^{-\frac{1}{2}}, \quad m = 1, 2, \dots, M. \quad (11)$$

Then, each trial X_j of the source subject is transformed to:

$$\tilde{X}_j = A_m X_j, \quad \text{if } X_j \in \text{Class } m \quad (12)$$

The difference between the mean covariance matrices of the corresponding class between the transformed source subject and the target subject becomes

$$\begin{aligned} A \bar{C}_{S,m} A^T - \bar{C}_{T,m} &= \bar{C}_{T,m}^{-\frac{1}{2}} \bar{C}_{S,m}^{-\frac{1}{2}} \bar{C}_{S,m} \bar{C}_{S,m}^{-\frac{1}{2}} \bar{C}_{T,m}^{-\frac{1}{2}} - \bar{C}_{T,m} \\ &= \bar{C}_{T,m}^{-\frac{1}{2}} I \bar{C}_{T,m}^{-\frac{1}{2}} - \bar{C}_{T,m} = \mathbf{0}, \end{aligned} \quad (13)$$

Algorithm 1 LA for Different Set Domain Adaptation

Input: $\{X_j, y_j\}_{j=1}^{N_s}$, labeled source domain trials;
 $\{y_{S,m}\}_{m=1}^M$, label set of the source domain;
 $\{X_i\}_{i=1}^{N_t}$, unlabeled target domain trials;
 $\{y_{T,m}\}_{m=1}^M$, label set of the target domain;
 k , number of target domain trials to be labeled.

Output: $\{\tilde{X}_j, \tilde{y}_j\}_{j=1}^{N_s}$, aligned source domain trials.

Compute the target domain covariance matrices $\{C_i\}_{i=1}^{N_t}$ by (5);

Perform k -medoids clustering on $\{C_i\}_{i=1}^{N_t}$ using the Riemannian distance;

Label the k medoids;

Compute $\bar{C}_{T,m}$ ($m = 1, 2, \dots, M$), the mean covariance matrix of each target domain class from the k labeled medoids;

Compute $\bar{C}_{S,m}$ ($m = 1, 2, \dots, M$), the mean covariance matrix of each source domain class;

Match each source domain label with a target domain label; assume $y_{S,m}$ is matched with $y_{T,m}$;

for $m = 1, 2, \dots, M$ **do**

 Compute A_m by (11);

 Compute $\{\tilde{X}_j\}_{j=1}^{N_s}$ by (12) and set $\tilde{y}_j = y_{T,m}$,
 $\forall_j y_j = y_{S,m}$;

end

Return $\{\tilde{X}_j, \tilde{y}_j\}_{j=1}^{N_s}$

where $\mathbf{0}$ is an all-zero matrix, i.e., the objective function in (10) is minimized.

A key question in LA is how to obtain $\bar{C}_{T,m}$, which requires some labeled target domain samples. We consider the following offline classification scenario: we have access to the unlabeled EEG trials (the same assumption is also used in EA), and we can label a few of them to estimate $\bar{C}_{T,m}$. To have a good estimate of $\bar{C}_{T,m}$ from only a few labeled trials, we need to select these trials very carefully. In this paper, we perform k -medoids clustering based on the Riemannian distances among the target EEG trials, label the k cluster centers, and then estimate $\bar{C}_{T,m}$ from them. In the rare case that the k centers had fewer than M different labels, we use EA to replace LA.

Another question is how we match the source labels with the target labels. When the source and target label sets partially overlap, for the labels in common, we match each source label with the same target label, and then randomly match each remaining source label with a remaining target label. For example, if the source label set is $\{A, B, C\}$ and the target label set is $\{A, D, E\}$, then we match source label A with target label A , source label B with target label D (or E), and source label C with target label E (or D). If the source and target label sets are completely different, we randomly match the source and target labels.

The pseudo-code of LA is shown in Algorithm 1. We perform LA for each source subject separately if there are multiple source subjects. After LA, the source domain and the target domain have the same label set, and the trials in the same class are aligned. Then, trials from the two domains can

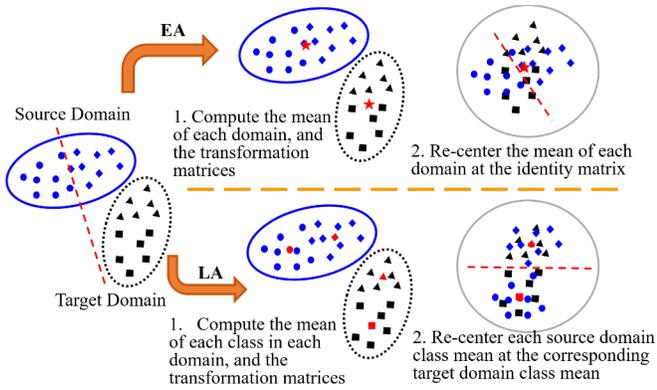


Fig. 2. Illustration of EA and LA.

be combined directly for feature extraction and classification. Or, an additional DA approach can be applied after LA to further improve the transfer learning performance, as shown in Section V.

B. LA Versus EA

The difference between LA and EA is illustrated in Fig. 2. For clarity, binary classification is used, but both EA and LA can be easily extended to multi-class classification, as shown later in this paper. In Fig. 2, each EEG trial is represented by its covariance matrix, as a point on a Riemannian manifold. The source domain (blue points) and target domain (black points) represent two different subjects, who have trials from different motor imagery tasks (indicated by different shapes of the points). Note that the shapes in the target domain are only used to help understand our approach, but not to suggest that we need to know all target labels). Initially, the source and target domains scatter far away from each other, due to the domain gap and also the category gap. If we build a classifier on the source domain (indicated by the red dashed line) and apply it directly to the target domain, it may not work at all. EA and LA alleviate this problem by reducing the gaps between the two domains before classification:

- 1) EA focuses on the domain gap but ignores the category gap completely. It first computes the mean covariance matrix of each domain (indicated by the red stars), from which a transformation matrix of each domain is computed. Using the transformation matrix, EA then re-centers each domain at the identity matrix, and makes the source and target domains overlap with each other, i.e., the domain gap between them is reduced. If we build a classifier in the source domain (the red dashed line) and apply it to the target domain, the classification performance would be improved.
- 2) LA considers the domain gap and the category gap simultaneously. It first computes the mean covariance matrix of each source domain class (indicated by the red circle and the red diamond), and estimates the mean covariance matrix of each target domain class (indicated by the red triangle and the red square). Then, LA re-centers each source domain class at the corresponding estimated class mean of the target domain. If we build a classifier in the source domain (the red dashed line)

and apply it to the target domain, the classification performance would be further improved.

C. LA Versus CORrelation Alignment (CORAL)

Sun *et al.* [36] proposed an unsupervised DA approach, CORrelation ALignment (CORAL), to minimize the domain shift by aligning the second-order statistics of different distributions.

Given a source domain $D_S \in \mathbb{R}^{N_S \times d}$ and a target domain $D_T \in \mathbb{R}^{N_T \times d}$, where N_S and N_T are the number of trials in the source domain and the target domain, respectively, and d the feature dimensionality. CORAL first computes the feature covariance matrix $C_S \in \mathbb{R}^{d \times d}$ in the source domain and $C_T \in \mathbb{R}^{d \times d}$ in the target domain. Then, it finds a linear transformation matrix $A \in \mathbb{R}^{d \times d}$ for the source domain features, so that the Frobenius norm of the difference between the covariance matrices of the two domains is minimized, i.e.,

$$\min_A \|A^T C_S A - C_T\|_F^2 \quad (14)$$

Although (14) seems similar to the objective function of LA in (10), they are different:

- 1) CORAL uses 1D features, and each domain has only one feature covariance matrix, which measures the covariances between different pairs of individual features. LA uses 2D features (EEG trials), and each EEG trial has a covariance matrix, which measures the covariances between different pairs of EEG channels. So, the covariance matrices in CORAL and LA have different meanings.
- 2) CORAL minimizes the distance between the covariance matrices in different domains, whereas LA minimizes the distance between the mean covariance matrices of the corresponding class in different domains.
- 3) CORAL works when the source domain has the same class labels as the target domain, and it finds one transformation matrix for each source domain. LA considers the case that the source and target domains have different class labels (of course, it also works when the two domains have the same class labels), and it finds one transformation matrix for each class of the source domain.

In summary, LA and CORAL have different inputs, different optimization objectives, and also different application scenarios. When the source and target domains have the same class labels, each 2D EEG trial can be mapped from the Riemannian manifold to the tangent space to obtain a 1D feature vector, and hence be plugged into CORAL. However, CORAL cannot be used when the source and target domains have different labels.

IV. DATASETS

This section describes and visualizes the two motor imagery datasets used in our experiments.

A. Datasets and Preprocessing

Both datasets were from BCI Competition IV,¹ and were collected in a cue-based setting. In each experiment, the

¹<http://www.bbc.de/competition/iv/>.

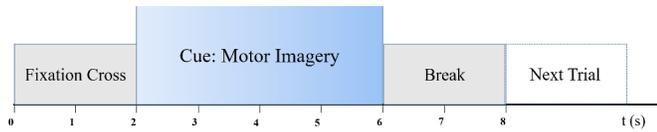


Fig. 3. Timing scheme of the motor imagery tasks.

TABLE I
SUMMARY OF THE TWO MOTOR IMAGERY DATASETS

	Number of				
	Channels	Time samples	Subjects	Classes	Trials/class
Data 1	59	300	7	2	100
Data 2a	22	300	9	4	72

subject was sitting in front of a computer and performed motor imagery tasks at the prompt of visual cues, as shown in Fig. 3. Each trial began when a fixation cross appeared on the black screen ($t = 0$), which prompted the subject to be prepared. After a short period, an arrow pointing to a certain direction was displayed as the visual cue ($t = 2$). The cue was displayed for a few seconds, during which the subject was instructed to perform the desired motor imagery task according to the direction of the arrow. The subject stopped the motor imagery when the visual cue disappeared ($t = 6$). A short break followed, until the next trial began ($t = 8$).

The first dataset² (Dataset 1 [41]) was recorded from seven healthy subjects by 59 EEG channels at 100 Hz. Each subject was instructed to perform two classes of motor imagery tasks, which were selected from three options: left hand, right hand, and feet. The recording of each subject was divided into three sessions: calibration, evaluation, and special feature. This paper only used the calibration data, because they included complete label information. Each subject had 100 trials from each class.

The second dataset³ (Dataset 2a) was recorded from nine healthy subjects by 22 EEG channels and 3 EOG channels at 250 Hz (we downsampled it to 100 Hz, to be consistent with Dataset 1). Each subject was instructed to perform four classes of motor imagery tasks: left hand, right hand, both feet, and tongue, which were represented by labels 1, 2, 3 and 4, respectively. A training session and an evaluation session were recorded on different days for each subject. We only used the 22-channel EEG data in the training session, which included complete label information. Each subject had 72 trials from each class, and 288 trials in total.

For both datasets, the EEG signals were preprocessed using the Matlab EEGLAB toolbox [42], following the guideline in [43]. First, a causal band-pass filter (20-order linear phase Hamming window FIR filter designed by Matlab function `fir1`, with 6dB cut-off frequencies at [8, 30] Hz) was applied to remove muscle artifacts, line-noise contamination and DC drift. Next, we extracted EEG signals between [0.5, 3.5] seconds after the cue appearance as our trials.

Table I summarizes the characteristics of the two datasets.

²http://www.bbci.de/competition/iv/desc_1.html.

³http://www.bbci.de/competition/iv/desc_2a.pdf.

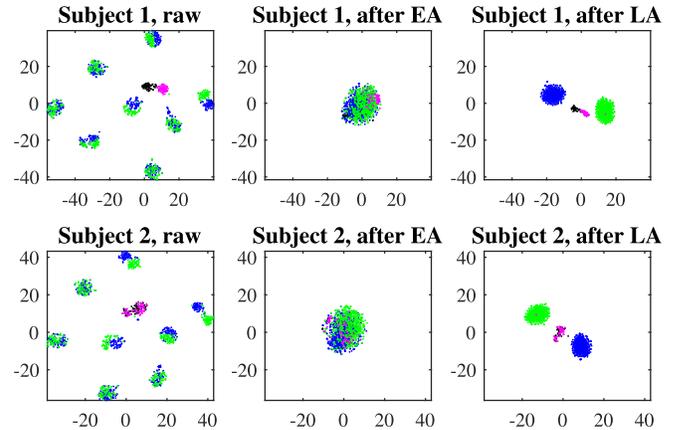


Fig. 4. t -SNE visualization when the first two subjects were used as the target subjects, respectively. Red dots: trials of Label 2 from the target subject; black dots: trials of Label 3 from the target subject; blue dots: trials of Label 1 from the source subjects; green dots: trials of Label 4 from the source subjects. The first column shows the trials without alignment, the second column shows the trials after EA, and the third after LA.

B. Data Visualization

In order to intuitively show how EA and LA reduce the distribution discrepancies between the target and source subjects, we first projected the EEG covariance matrices from the Riemannian manifold into the tangent space, then used the 1D tangent vectors as features to represent the EEG trials, as introduced in Section II-B. Finally, we used t -stochastic neighbor embedding (t -SNE) [44], a technique for dimensionality reduction and high-dimensional dataset visualization, to display the EEG trials (tangent vectors) before and after EA/LA in 2D.

More specifically, we first divided Dataset 2a into two datasets with different label spaces: the source dataset consisted of trials with Labels 1 and 4, and the target dataset with Labels 2 and 3. Then, we picked one subject from the target dataset as the target subject, and the remaining eight subjects from the source dataset as the source subjects. Fig. 4 shows two examples when the first two subjects were used as the target subjects, respectively. The red and black dots are trials of Labels 2 and 3 from the target subject, respectively. The blue and green dots are trials of Labels 1 and 4 from the source subjects, respectively. The first column shows the trials without alignment, the second column shows the trials after EA, and the third after LA.

Observe that trials from the source subjects (blue and green dots) are scattered far away from those of the target subject (red and black dots), when no alignment is performed. However, the target and source trials overlap with each other after EA, since their centers are now identical. After LA, the target and source trials are further aligned according to their labels. It's clear that different classes are more distinguishable after LA.

V. EXPERIMENTS AND RESULTS

This section presents performance comparisons of LA with other approaches on the two datasets. The code is available at <https://github.com/hehe91/LA>.

A. Domain Adaptation (DA) Scenarios

We investigated the problem that the source and target subjects have different label spaces, and considered the following five DA scenarios:

- 1) *Scenario I-a*: The source and target subjects have the same feature space and partially overlapping label spaces (binary classification).
- 2) *Scenario I-b*: The source and target subjects have the same feature space and partially overlapping label spaces (multi-class classification).
- 3) *Scenario II-a*: The source and target subjects have the same feature space and completely different label spaces (binary classification).
- 4) *Scenario II-b*: The source and target subjects have the same feature space and completely different label spaces (multi-class classification).
- 5) *Scenario III*: The source and target subjects have different feature spaces and also different label spaces.

For Scenarios I-a, I-b, II-a and II-b, in each experiment we divided Dataset 2a into two sub-datasets, a source dataset and a target dataset, such that they had the same feature space and different label spaces. Each sub-dataset was named by its label space, for example, sub-dataset “1, 2” consisted of trials with Labels 1 and 2 only, and sub-dataset “3, 4” consisted of trials with Labels 3 and 4 only. Then, “1, 2→3, 4” denotes the experiment that Sub-dataset “1, 2” was used as the source dataset and Sub-dataset “3, 4” the target dataset.

Then, the datasets used in the five DA scenarios were:

- 1) *Scenario I-a*: We divided Dataset 2a into a source sub-dataset and a target sub-dataset, ensuring they had one identical label and one different label. There were 24 such sub-dataset combinations in total, e.g., “1, 2→1, 3” and “1, 2→3, 2”.
- 2) *Scenario I-b*: We divided Dataset 2a into a source sub-dataset and a target sub-dataset, ensuring they had two identical labels and a different label. There were 12 such combinations in total, e.g., “1, 2, 3→1, 2, 4” and “1, 2, 3→1, 4, 3”.
- 3) *Scenario II-a*: We divided Dataset 2a into a source sub-dataset and a target sub-dataset, ensuring they had completely different labels. There were six such combinations in total, e.g., “1, 2→3, 4” and “2, 3→1, 4”.
- 4) *Scenario II-b*: We used the same sub-dataset combinations as in Scenario I-b, but mismatched the labels between the target and source subjects, e.g., “1, 2, 3→2, 1, 4” and “1, 2, 4→2, 1, 3”.
- 5) *Scenario III*: We used Dataset 1 as the source dataset and sub-dataset “3, 4” of Dataset 2a as the target dataset, so that they had different feature spaces (their EEG channels were different) and also different label spaces.

Once a dataset choice was made, each time we picked one subject from the target dataset as the target subject, and the remaining subjects from the source dataset as the source subjects. As the target dataset always had nine subjects, we had nine sub-experiments for each dataset combinations. Table II summarizes the characteristics of all scenarios, where k is the number of labeled target subject trials.

TABLE II

SUMMARY OF THE FOUR DA SCENARIOS, WHERE k IS THE NUMBER OF LABELED TARGET SUBJECT TRIALS

	No. of dataset combinations	No. of sub-experiments	No. of training trials	No. of test trials
Scenario I-a	24	24×9	$144 \times 8 + k$	$144 - k$
Scenario I-b	12	12×9	$216 \times 8 + k$	$216 - k$
Scenario II-a	6	6×9	$144 \times 8 + k$	$144 - k$
Scenario II-b	12	12×9	$216 \times 8 + k$	$216 - k$
Scenario III	1	9	$1400 + k$	$144 - k$

B. Experimental Settings

We first divided the BCI classification pipeline into three stages:

- 1) *Preprocessing*, which first temporally filters the EEG data, then epochs the continuous EEG signals into trials, as described in Section IV-A.
- 2) *Alignment*, which selectively performs different alignments.
- 3) *Classification*, which extracts features and trains classifiers.

In order to emphasize the effect of LA, the algorithms to be compared consisted of the same preprocessing and classification stages, but different alignments. More specifically, three alignment approaches were compared:

- 1) *Raw*, which did not perform any alignment.
- 2) *EA*, which performed EA.
- 3) *LA*, which performed LA.

In each scenario, the experiments were designed to answer the following two questions:

Question 1: Can LA be used as an effective preprocessing step before different feature extraction and classification algorithms?

Question 2: Can LA be integrated with other DA approaches to further improve the classification performance?

For Question 1, we used two feature extraction and classification pipelines:

- 1) CSP-LDA: It spatially filtered the EEG data by CSP, computed the log-variance as features, and then used them in an LDA classifier.
- 2) TS-SVM: It extracted the Riemannian TS features, as introduced in Section II-B, then used them in an SVM classifier.

Combining these two pipelines with the three alignment approaches (Raw, EA, LA), we had $2 \times 3 = 6$ algorithms to be compared. Our goal was to verify whether LA performs the best in both pipelines.

For Question 2, we first extracted the Riemannian TS features, and then used different DA approaches in classification stage (because they need 1D features):

- 1) BL (baseline), which directly applied an SVM classifier to the TS features, without any additional DA approach.
- 2) JDA, which applied JDA to the TS features, and then used them in an SVM classifier.
- 3) JGSA, which applied JGSA to the TS features, and then used them in an SVM classifier.
- 4) MEDA, which applied MEDA to the TS features.

Combining these four approaches with the three alignments (Raw, EA, LA), we had $4 \times 3 = 12$ algorithms to be compared. Our goal was to verify whether “LA+JDA/JGSA/MEDA > LA+BL > Raw+JDA/JGSA/MEDA”, where “>” means “outperform”. For example, “LA+BL > Raw+JDA/JGSA/MEDA” means LA outperforms classical DA approaches such as JDA, JGSA and MEDA, and “LA+JDA/JGSA/MEDA > LA+BL” means the performance could be further improved by integrating LA with other DA approaches, i.e., LA is compatible with and complementary to other DA approaches.

C. Scenario I-A: Same Feature Space and Partially Overlapping Label Spaces in Binary Classification

This subsection considers the binary classification scenario that the source and target subjects have the same feature space and partially overlapping label spaces. As introduced in Section V-A, we had 24 sub-dataset combinations to be tested.

Because in Scenario I-a the source subjects had one identical label and one different label from the target subject, we first matched the identical label, then the remaining labels. For example, in the combination “1, 2 \rightarrow 1, 3”, we matched source Label 1 with target Label 1, and source Label 2 with target Label 3. For algorithms without LA, we directly assigned Label 3 to the source trials with Label 2. For algorithms with LA, we first aligned the source Label 1 trials with the target Label 1 trials, then aligned the source Label 2 trials with the target Label 3 trials, and assigned Label 3 to the source trials of Label 2.

For algorithms involving LA, we considered $k \in \{2, 4, \dots, 20\}$ in k -medoids clustering of LA in Section III. In the rare case that the labeled target trials came from the same class, we cannot perform LA as there was not enough information to estimate the two class means of the target subject; thus, we performed EA instead of LA. No matter whether the labeled target trials were used in the alignment or not, they were always combined with the labeled source trials for feature extraction and classification. All labeled target subject trials were excluded from the test set, so all algorithms had the same training set and test set.

Question 1: Can LA be used as an effective preprocessing step before different feature extraction and classification algorithms?

We compared Raw, EA, and LA in the two classification pipelines to answer this question. Fig. 5 shows the performances of the six algorithms on the 24 different sub-dataset combinations, where each subfigure shows the average accuracies across the nine subjects (each as the target subject once). The last subfigure shows the average performances across the 24 experiments. Observe that:

- 1) EA-CSP-LDA outperformed Raw-CSP-LDA on 20 out of the 24 experiments, and EA-TS-SVM outperformed Raw-TS-SVM on 14 out of the 24 experiments. On average EA-CSP-LDA outperformed Raw-CSP-LDA, and EA-TS-SVM outperformed Raw-TS-SVM. This suggests EA was generally effective, but not always, when the source and target label spaces were different.

TABLE III

SCENARIO I-A: p -VALUES OF PAIRED t -TESTS ON THE AUCs OF THE CLASSIFICATION ACCURACY CURVES IN FIG. 5. THE NULL HYPOTHESIS WAS REJECTED IF $p < \alpha$, WHERE $\alpha = 0.05$

	EA-CSP-LDA	EA-TS-SVM
LA-CSP-LDA	0.0000	
LA-TS-SVM		0.0000

- 2) When k became large, LA-CSP-LDA outperformed Raw-CSP-LDA in all 24 experiments, and LA-TS-SVM also outperformed Raw-TS-SVM in all 24 experiments. These suggest that LA was able to cope well with partially different label spaces.
- 3) When k became large, LA-CSP-LDA outperformed EA-CSP-LDA on all 24 experiments, and LA-TS-SVM also outperformed EA-TS-SVM on all 24 experiments. This suggests LA was more effective and robust than EA.
- 4) Generally, the classification accuracies of LA-CSP-LDA and LA-TS-SVM increased when there were more labeled target trials for estimating the class means, which is intuitive.

We also performed statistical tests to determine if the differences between the LA-based algorithms and others were statistically significant. We first defined an aggregated performance measure called the area under the curve (AUC). For a particular algorithm on a particular subject, the AUC was the area under its accuracy curve when the number of labeled target subject trials increased from 2 to 20. Subjects from all 24 experiments were concatenated, so we had $24 \times 9 = 216$ subjects in total. Each algorithm had 216 AUCs. We then performed paired t -tests on these AUCs. The null hypothesis was that the difference between the paired samples has zero mean, which was rejected if $p < \alpha$, where $\alpha = 0.05$ was used. The results are shown in Table III, where the statistically significant ones are marked in bold. LA-CSP-LDA significantly outperformed EA-CSP-LDA, and LA-TS-SVM significantly outperformed EA-TS-SVM. These results echo the observations from Fig. 5 and answer Question 1: LA can be used as an effective preprocessing step before different feature extraction and classification algorithms.

Question 2: Can LA be integrated with other DA approaches to further improve the classification performance?

As introduced in previous subsection, we had 12 algorithms to be compared. We used the same target and source subjects as introduced in Question 1, which resulted in 24 experiments again. Fig. 6 shows the performances of the 12 algorithms in the 24 experiments, where each subfigure shows the average accuracies across the nine subjects, and the last subfigure shows the average performances across the 24 experiments. Observe that:

- 1) When k was large, LA-BL always outperformed Raw-BL and EA-BL, LA-JDA always outperformed Raw-JDA and EA-JDA, LA-JGSA always outperformed Raw-JGSA and EA-JGSA, and LA-MEDA always outperformed Raw-MEDA and EA-MEDA. These suggest that LA was effective regardless of whether additional DA approaches were used or not.

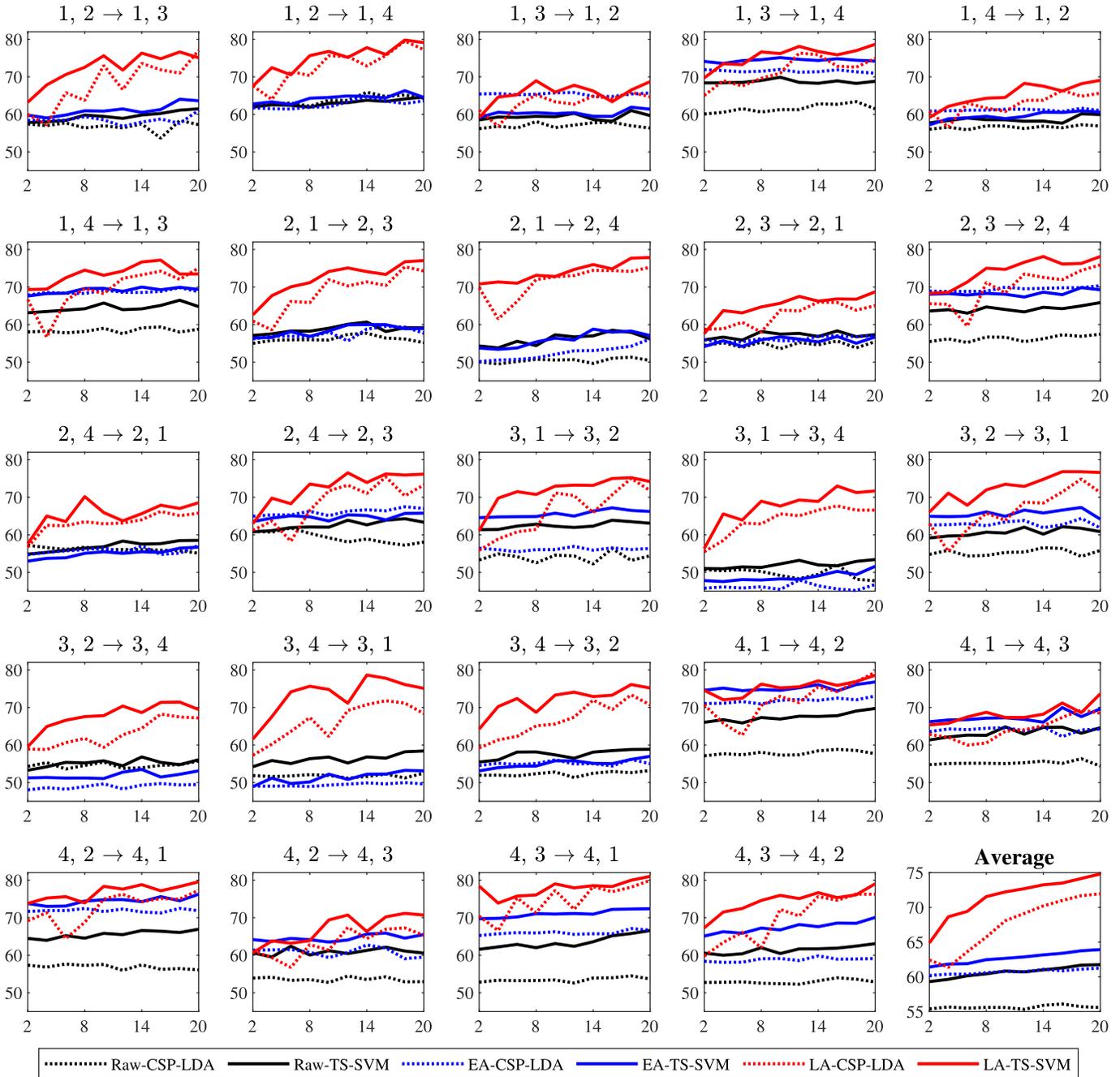


Fig. 5. Average classification accuracies (%) in Scenario I-a. The horizontal axis indicates the number of labeled target subject trials (k in k -medoids clustering in Section III), and the vertical axis the classification accuracies.

- 2) LA-BL always outperformed Raw-JDA and Raw-MEDA, and outperformed Raw-JGSA in 23 out of 24 experiments. These suggest that LA can outperform classical DA approaches such as JDA, JGSA and MEDA.

- 3) Both LA-JDA and LA-MEDA always outperformed LA-BL, and LA-JGSA outperformed LA-BL in 23 out of the 24 experiments. These suggest that it may be advantageous to integrate other DA approaches with LA.

We also performed paired t -tests on the AUCs in Fig. 6. The results are shown in Table IV, which indicate that the algorithms involving LA (i.e., LA-BL, LA-JDA, LA-JGSA, LA-MEDA) significantly outperformed those involving EA

(i.e., EA-JDA, EA-JGSA, EA-MEDA), and the algorithms combining LA and additional DA approaches (i.e., LA-JDA, LA-JGSA, LA-MEDA) significantly outperformed LA-BL. These results echo the observations from Fig. 6 and answer Question 2: LA can not only outperform EA and classical DA approaches, but the classification performance can be further improved when integrated with other DA approaches.

D. Scenario I-B: Same Feature Space and Partially Overlapping Label Spaces in Multi-Class Classification

This subsection considers the multi-class classification scenario that the source and target subjects have the same feature

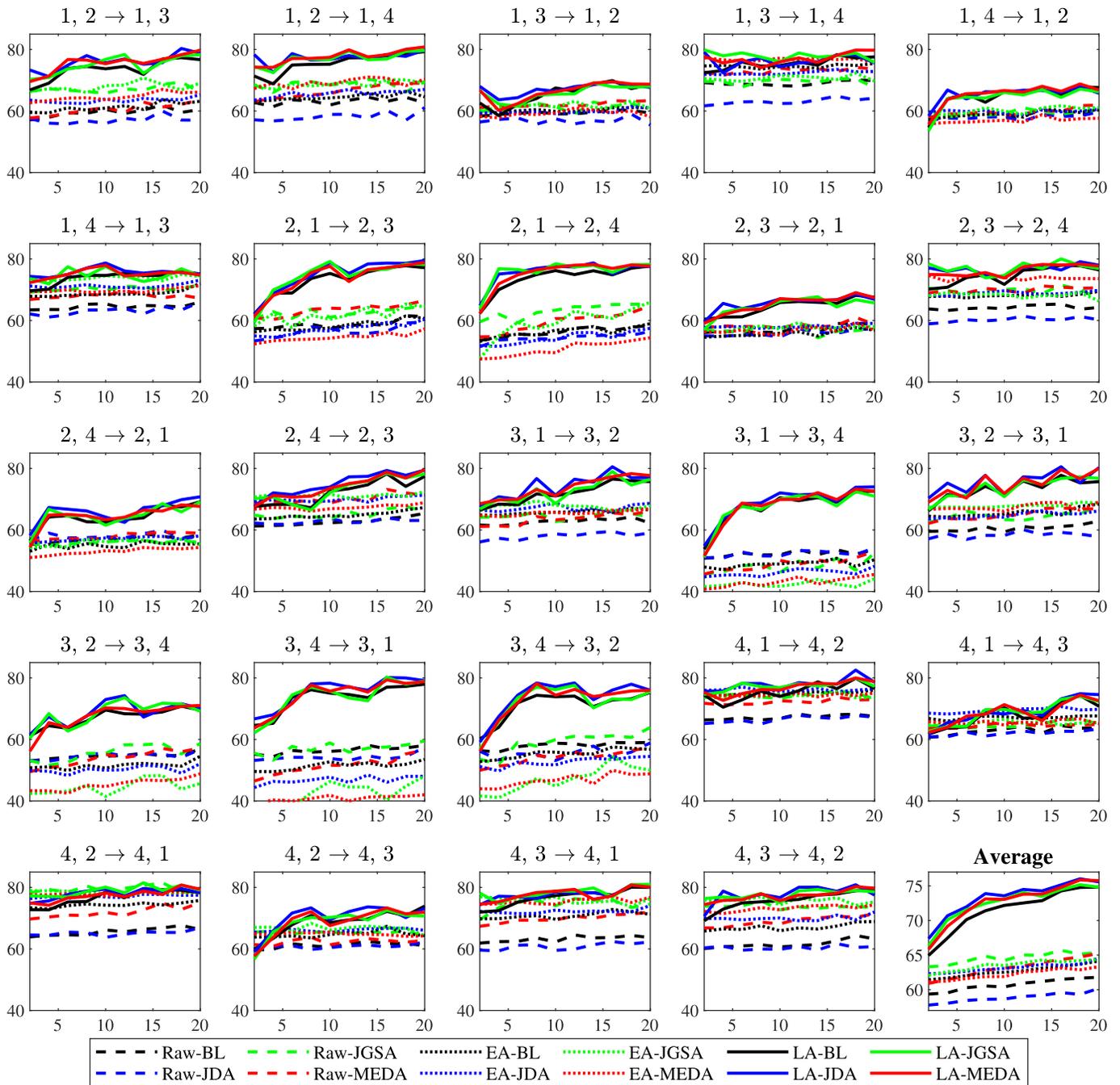


Fig. 6. Average classification accuracies (%) in Scenario I-a, when additional DA approaches were used after LA. The horizontal axis indicates the number of labeled target subject trials (k in k -medoids clustering in Section III), and the vertical axis the classification accuracies.

space and partially overlapping label spaces. As introduced in Section V-A, we had 12 sub-dataset combinations to be tested.

Question 1: Can LA be used as an effective preprocessing step before different feature extraction and classification algorithms?

Again, we compared Raw, EA, and LA in the two classification pipelines to answer this question. CSP filtering was extended from binary classification to multi-class classification by the one-versus-the-rest approach [45]. As we had three class centers of the target subject to be estimated in LA, we considered $k \in \{3, 6, \dots, 30\}$ in k -medoids clustering.

Fig. 7 shows the performances of the six algorithms on the 12 different sub-dataset combinations, where each sub-figure shows the average classification accuracies across the nine subjects (each as the target subject once). The last sub-figure shows the average performances across the 12 experiments. LA-CSP-LDA always outperformed Raw-CSP-LDA and EA-CSP-LDA, and LA-TS-SVM always outperformed Raw-TS-SVM and EA-TS-SVM. These suggest that LA was effective with different feature extraction and classification algorithms.

Paired t -tests on the AUCs in Fig. 7 were also performed to check if the differences between different algorithms were

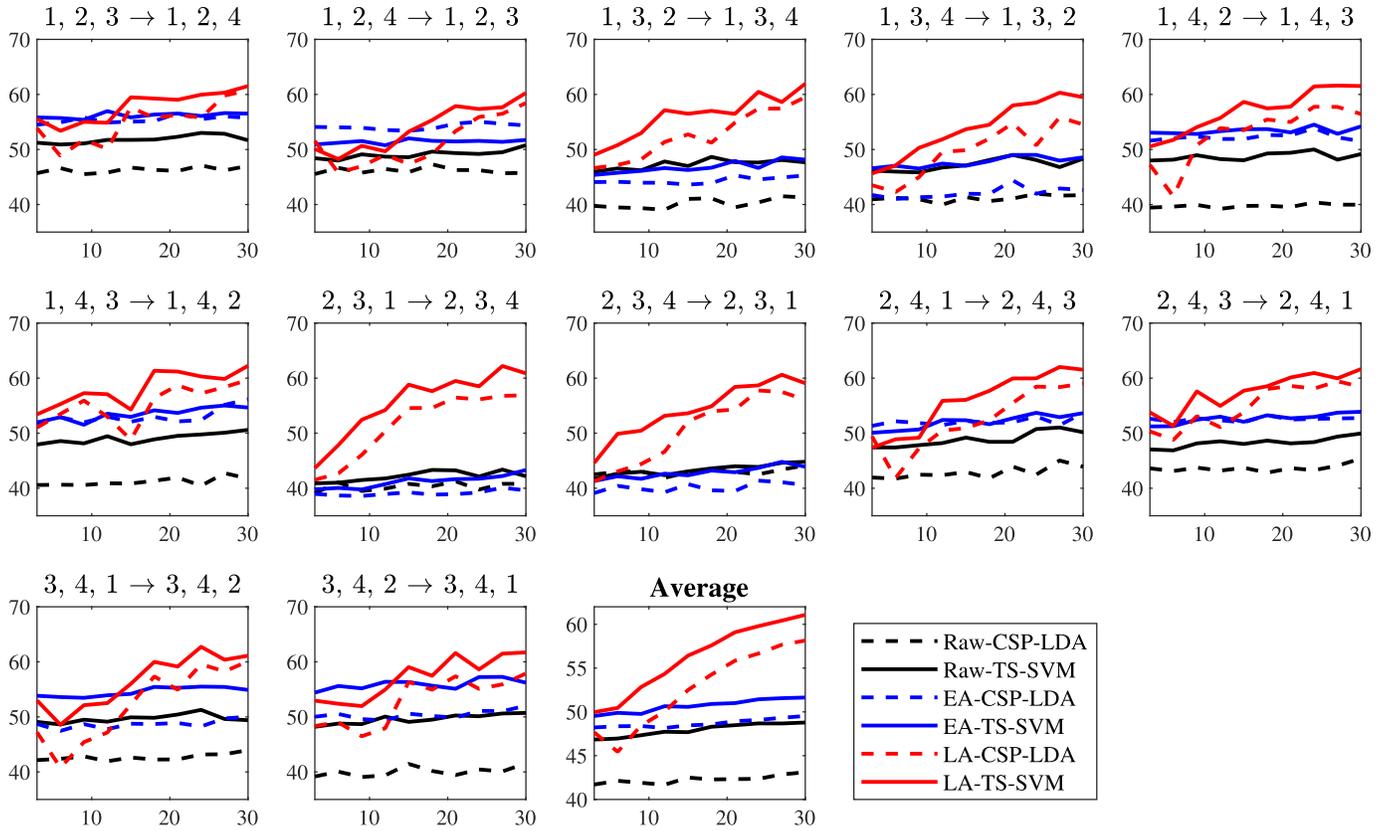


Fig. 7. Average classification accuracies (%) in Scenario I-b. The horizontal axis indicates the number of labeled target subject trials (k in k -medoids clustering in Section III), and the vertical axis the classification accuracies.

TABLE IV

SCENARIO I-A: p -VALUES OF PAIRED t -TESTS ON THE AUCs OF THE CLASSIFICATION ACCURACY CURVES IN Fig. 6. THE NULL HYPOTHESIS WAS REJECTED IF $p < \alpha$, WHERE $\alpha = 0.05$

	LA-BL	EA-JDA	EA-JGSA	EA-MEDA
LA-BL		0.0000	0.0000	0.0000
LA-JDA	0.0000			
LA-JGSA	0.0001		0.0000	
LA-MEDA	0.0000			0.0000

TABLE V

SCENARIO I-B: p -VALUES OF PAIRED t -TESTS ON THE AUCs OF THE CLASSIFICATION ACCURACY CURVES IN Fig. 7. THE NULL HYPOTHESIS WAS REJECTED IF $p < \alpha$, WHERE $\alpha = 0.05$

	EA-CSP-LDA	EA-TS-SVM
LA-CSP-LDA	0.0196	
LA-TS-SVM		0.0010

statistically significant. Here the AUC was the area under the accuracy curve when the number of labeled target subject trials increased from 3 to 30. Each algorithm had $12 \times 9 = 108$ AUCs. The results are shown in Table V, which indicate that LA-CSP-LDA significantly outperformed EA-CSP-LDA, and LA-TS-SVM significantly outperformed EA-TS-SVM.

Question 2: Can LA be integrated with other DA approaches to further improve the classification performance?

Again, we combined Raw, EA, LA with different DA approaches and obtained 12 algorithms to be compared. Fig. 8

shows their performances on the 12 sub-dataset combinations, and the average across the 12 experiments. Observe that:

- 1) When k was large, LA-BL always outperformed Raw-BL and EA-BL, LA-JDA always outperformed Raw-JDA and EA-JDA, LA-JGSA always outperformed Raw-JGSA and EA-JGSA, and LA-MEDA always outperformed Raw-MEDA. These suggest that LA was effective regardless of whether additional DA approaches were used or not.
- 2) When k was large, LA-BL outperformed Raw-JDA, Raw-JGSA and Raw-MEDA in all 12 experiments, suggesting that LA can outperform classical DA approaches such as JDA, JGSA and MEDA.
- 3) Generally, LA-JDA, LA-JGSA and LA-MEDA outperformed LA-BL, suggesting that it may be advantageous to integrate additional DA approaches with LA.

Table VI shows the results of paired t -tests on the AUCs in Fig. 8. The conclusions in binary classification still hold in multi-class classification: LA significantly outperformed EA and classical DA approaches, and its performance can be further significantly improved when integrated with other DA approaches.

E. Scenario II-A: Same Feature Space and Completely Different Label Spaces in Binary Classification

This subsection considers the scenario that the source and target subjects have the same feature space but completely

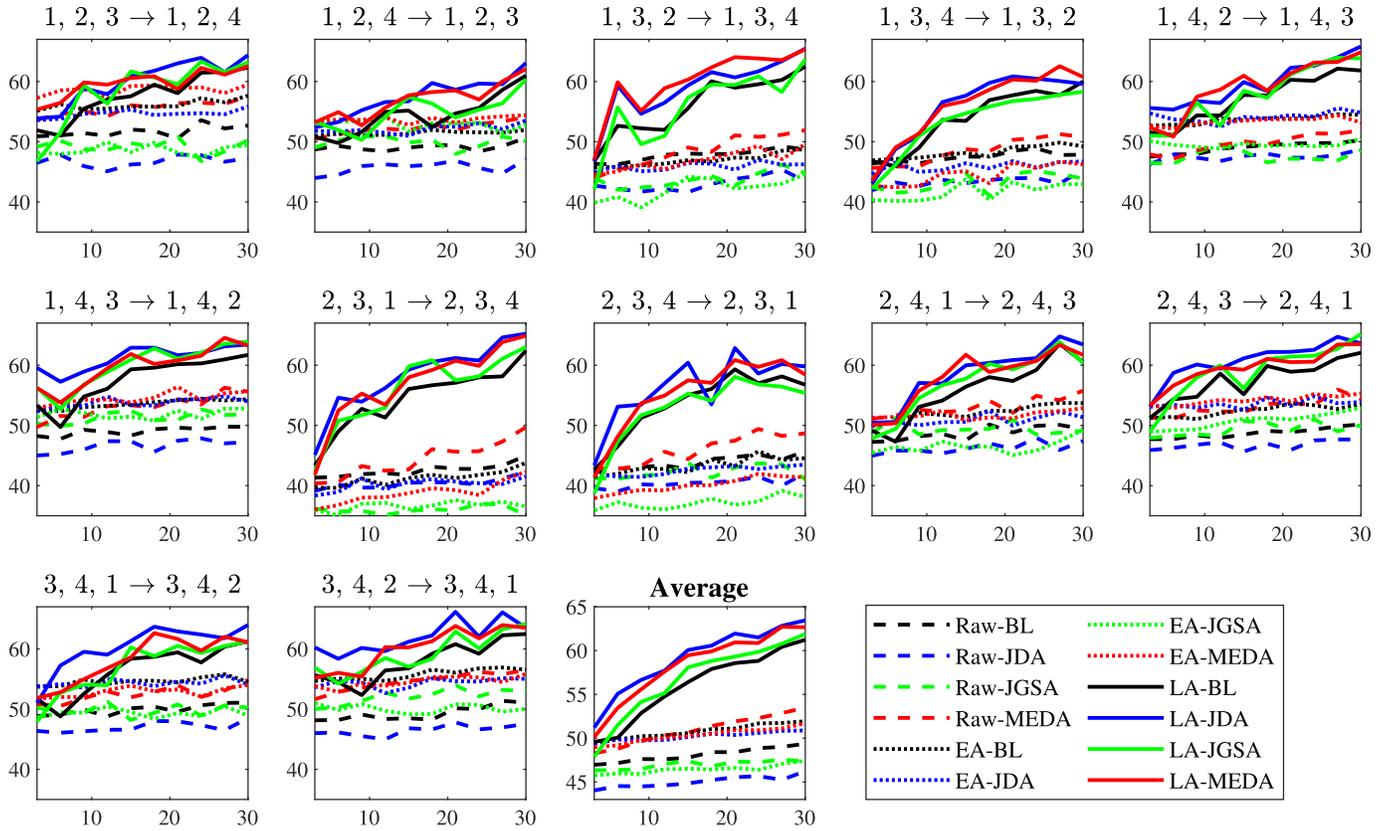


Fig. 8. Average classification accuracies (%) in Scenario I-b, when other DA approaches were used after LA. The horizontal axis indicates the number of labeled target subject trials (k in k -medoids clustering in Section III), and the vertical axis the classification accuracies.

TABLE VI

SCENARIO I-B: p -VALUES OF PAIRED t -TESTS ON THE AUCs IN Fig. 8. THE NULL HYPOTHESIS WAS REJECTED IF $p < \alpha$, WHERE $\alpha = 0.05$

	LA-BL	EA-JDA	EA-JGSA	EA-MEDA
LA-BL		0.0000	0.0000	0.0000
LA-JDA	0.0000			
LA-JGSA	0.0295		0.0000	
LA-MEDA	0.0000			0.0000

different label spaces. As introduced in Section V-A, we had six such sub-dataset combinations to be tested. $k \in \{2, 4, \dots, 20\}$ in k -medoids clustering of LA in binary classification was used.

Because in Scenario II-a the source subjects had completely different labels from the target subject, the source labels and the target labels were randomly matched for LA. For example, in the experiment “1, 2→3, 4”, we could align the trials of Label 1 with those of Label 3, and the trials of Label 2 with those of Label 4. We could also align the trials of Label 2 with those of Label 3, and the trials of Label 1 with those of Label 4. Our experiments showed that LA was effective in both alignment strategies.

Question 1: Can LA be used as an effective preprocessing step before different feature extraction and classification algorithms?

We compared Raw, EA, and LA in the two classification pipelines. Fig. 9 shows the performances of the six algorithms

on the six sub-dataset combinations, and the average. Observe that:

- 1) LA-CSP-LDA always outperformed Raw-CSP-LDA and EA-CSP-LDA, and LA-TS-SVM always outperformed Raw-TS-SVM and EA-TS-SVM. This suggests LA was effective in different feature extraction and classification algorithms.
- 2) Comparing the last subfigure in Fig. 5 with the last one in Fig. 9, we can observe that the performances of Raw-CSP-LDA and Raw-TS-SVM were lower in Fig. 9, which is intuitive, because the label spaces in Scenario II-a had larger discrepancies. However, the performances of LA-CSP-LDA and LA-TS-SVM did not change much, suggesting that LA can cope well with large label space discrepancies.

For the most extreme case that only one labeled target subject trial from each class is available, the average classification accuracies across the nine subjects in the nine experiments are given in Table VII. LA achieved the best performances in all six experiments, regardless of which feature extraction and classification algorithm was used.

We also performed paired t -tests on the AUCs in Fig. 9. Each algorithm had $6 \times 9 = 54$ AUCs. The p -values are shown in Table VIII, where the statistically significant ones are marked in bold. LA-CSP-LDA significantly outperformed EA-CSP-LDA, and LA-TS-SVM significantly outperformed EA-TS-SVM.

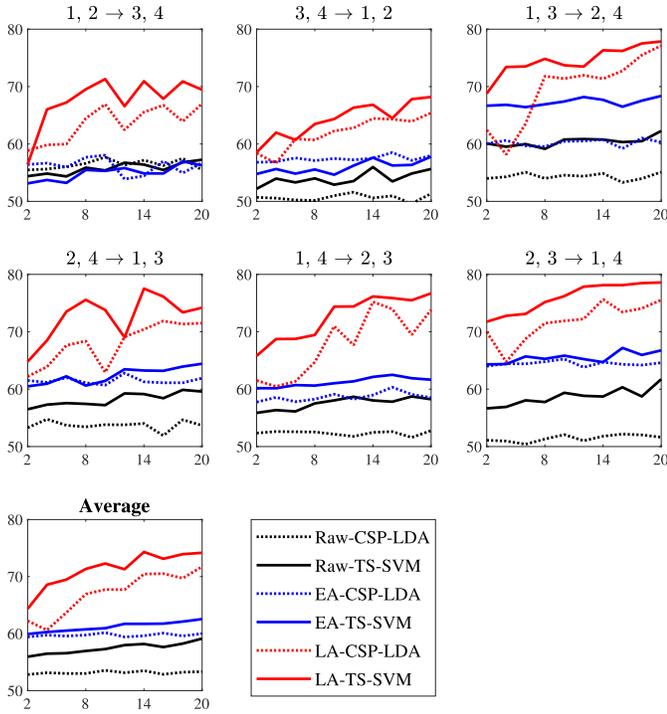


Fig. 9. Average classification accuracies (%) in Scenario II-a. The horizontal axis indicates the number of labeled target subject trials (k in k -medoids clustering in Section III), and the vertical axis the classification accuracies.

TABLE VII

SCENARIO II-A: AVERAGE CLASSIFICATION ACCURACIES (%) ACROSS THE NINE SUBJECTS WHEN ONLY ONE LABELED TARGET SUBJECT TRIAL FROM EACH CLASS IS AVAILABLE

Experiment	Approach	Raw	EA	LA
1, 2 → 3, 4	CSP-LDA	55.48	56.42	58.84
	TS-SVM	54.38	53.13	56.42
3, 4 → 1, 2	CSP-LDA	50.70	56.81	58.37
	TS-SVM	52.19	54.77	58.53
1, 3 → 2, 4	CSP-LDA	53.99	60.02	62.44
	TS-SVM	60.09	66.67	68.78
2, 4 → 1, 3	CSP-LDA	53.29	61.50	62.21
	TS-SVM	56.49	60.49	64.79
1, 4 → 2, 3	CSP-LDA	52.35	57.75	61.50
	TS-SVM	55.87	60.17	65.81
2, 3 → 1, 4	CSP-LDA	51.10	64.01	69.95
	TS-SVM	56.65	64.32	71.75

TABLE VIII

SCENARIO II-A: p -VALUES OF PAIRED t -TESTS ON THE AUCS OF THE CLASSIFICATION ACCURACY CURVES IN Fig. 9. THE NULL HYPOTHESIS WAS REJECTED IF $p < \alpha$, WHERE $\alpha = 0.05$

	EA-CSP-LDA	EA-TS-SVM
LA-CSP-LDA	0.0000	
LA-TS-SVM		0.0000

Question 2: Can LA be integrated with other DA approaches to further improve the classification performance?

Again, we considered the case when there were additional DA approaches after LA. The results are shown in Fig. 10. Observe that:

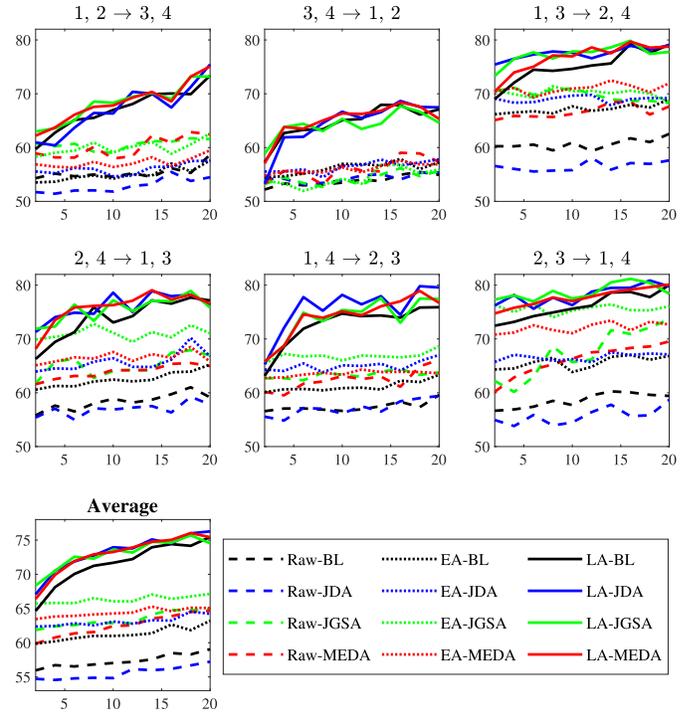


Fig. 10. Average classification accuracies (%) in Scenario II-a, when additional DA approaches were used after LA. The horizontal axis indicates the number of labeled target subject trials (k in k -medoids clustering in Section III), and the vertical axis the classification accuracies.

TABLE IX

SCENARIO II-A: p -VALUES OF PAIRED t -TESTS ON THE AUCS IN Fig. 10. THE NULL HYPOTHESIS WAS REJECTED IF $p < \alpha$, WHERE $\alpha = 0.05$

	LA-BL	EA-JDA	EA-JGSA	EA-MEDA
LA-BL		0.0000	0.0000	0.0000
LA-JDA	0.0109			
LA-JGSA	0.0116		0.0000	
LA-MEDA	0.0009			0.0000

- 1) LA-BL always outperformed Raw-BL and EA-BL, LA-JDA always outperformed Raw-JDA and EA-JDA, LA-JGSA always outperformed Raw-JGSA and EA-JGSA, and LA-MEDA always outperformed Raw-MEDA and EA-MEDA. These suggest that LA was effective regardless of whether an additional DA approach was used or not.
- 2) LA-BL outperformed Raw-JDA, Raw-JGSA and Raw-MEDA in all six experiments, suggesting that LA can outperform classical DA approaches such as JDA, JGSA and MEDA.
- 3) Generally, LA-JDA, LA-JGSA, LA-MEDA outperformed LA-BL, suggesting again that it may be advantageous to integrate an additional DA approach with LA.

The results of paired t -tests on the AUCs in Fig. 10 are shown in Table IX, which are consistent with those in the last two subsections: LA significantly outperformed EA and classical DA approaches, and the classification performance can be further significantly improved when LA was integrated with other DA approaches.

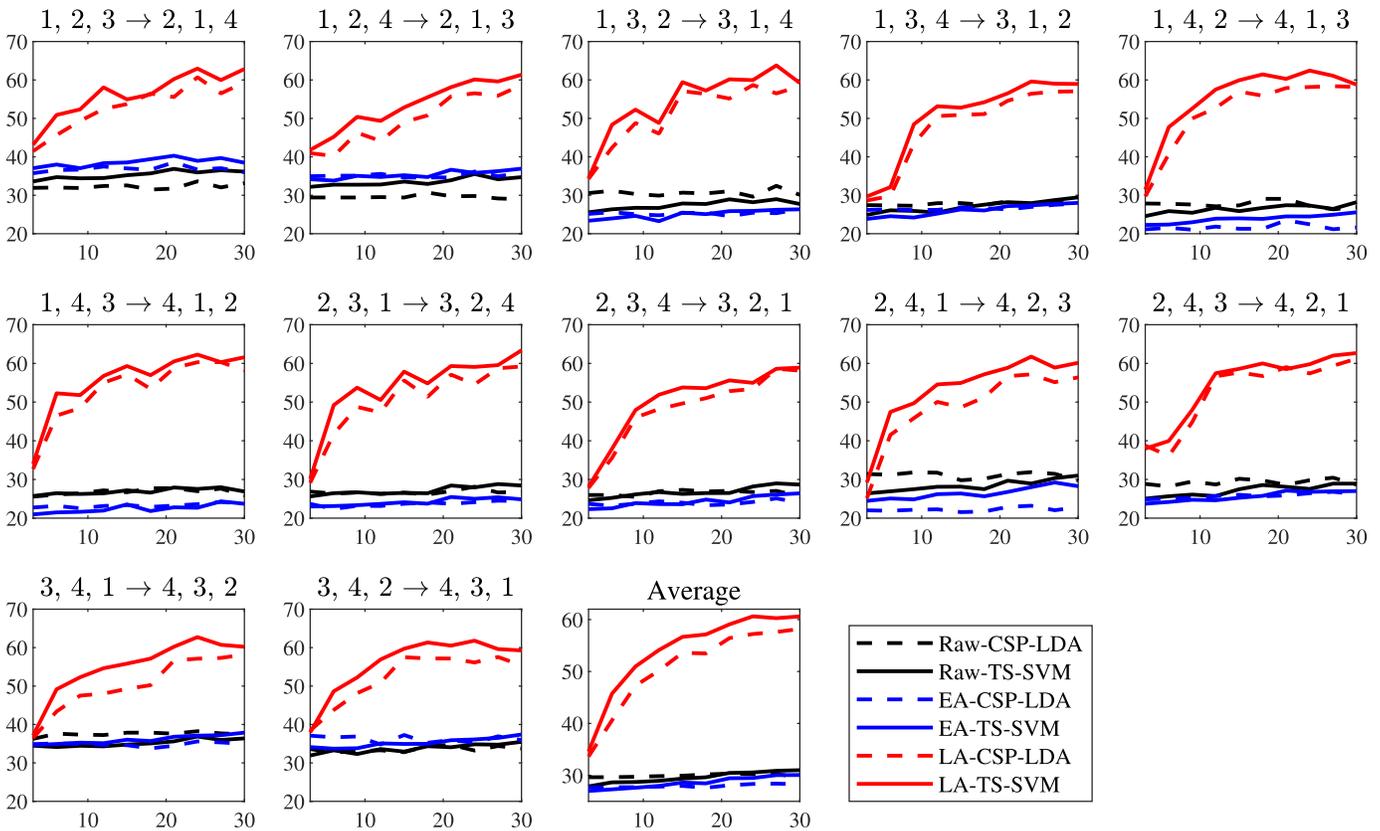


Fig. 11. Average classification accuracies (%) in Scenario II-b. The horizontal axis indicates the number of labeled target subject trials (k in k -medoids clustering in Section III), and the vertical axis the classification accuracies.

F. Scenario II-B: Same Feature Space and Completely Different Label Spaces in Multi-Class Classification

This subsection considers the multi-class classification scenario that the source and the target subjects have the same feature space but completely different label spaces.

Ideally, if Dataset 2a has six or more different classes, we can perform studies like “1, 2, 3 → 4, 5, 6” in multi-class classification. Unfortunately, Dataset 2a only has four classes. So, we mismatched the labels between the target and the source subjects, to simulate completely different label spaces in multi-class classification.

Assume the labels of the source subjects are ‘1’, ‘2’ and ‘3’, and the labels of the target subject are ‘1’, ‘2’ and ‘4’. Then, we match ‘1’ of the source subjects with ‘2’ of the target subject, ‘2’ of the source subjects with ‘1’ of the target subject, and ‘3’ of the source subjects with ‘4’ of the target subject, i.e., ‘1, 2, 3 → 2, 1, 4’. A potential application scenario of this setting is that for the source subjects we know which trials belong to the same class, but do not know the specific class labels. So, we randomly match them to the labels of the target subject.

Question 1: Can LA be used as an effective preprocessing step before different feature extraction and classification algorithms?

Again, we compared Raw, EA, and LA in the two classification pipelines to answer this question. Fig. 11 shows the performances of the six algorithms on 12 different sub-dataset

combinations, where each subfigure shows the average classification accuracies across the nine subjects (each as the target subject once). The last subfigure shows the average performances across the 12 experiments. The title of each subfigure shows the sub-datasets used, and also how we matched the labels between the two sub-datasets. Observe that:

- 1) LA-CSP-LDA always outperformed Raw-CSP-LDA and EA-CSP-LDA, and LA-TS-SVM always outperformed Raw-TS-SVM and EA-TS-SVM. This suggests that was effective in different feature extraction and classification algorithms.
- 2) LA in Fig. 11 achieved much larger performance improvements over Raw and EA than those in Fig. 7. When the labels mismatched, the algorithms without LA (i.e., Raw-CSP-LDA, Raw-TS-SVM, EA-CSP-LDA and EA-TS-SVM) performed very poorly. However, the performances of LA-CSP-LDA and LA-TS-SVM were very consistent, suggesting that LA can cope well with large label space discrepancies.

Paired t -tests on the AUCs in Fig. 11 were also performed to check if the differences between different algorithms were statistically significant. The results are shown in Table X, which indicate that LA-CSP-LDA significantly outperformed EA-CSP-LDA, and LA-TS-SVM significantly outperformed EA-TS-SVM.

Question 2: Can LA be integrated with other DA approaches to further improve the classification performance?

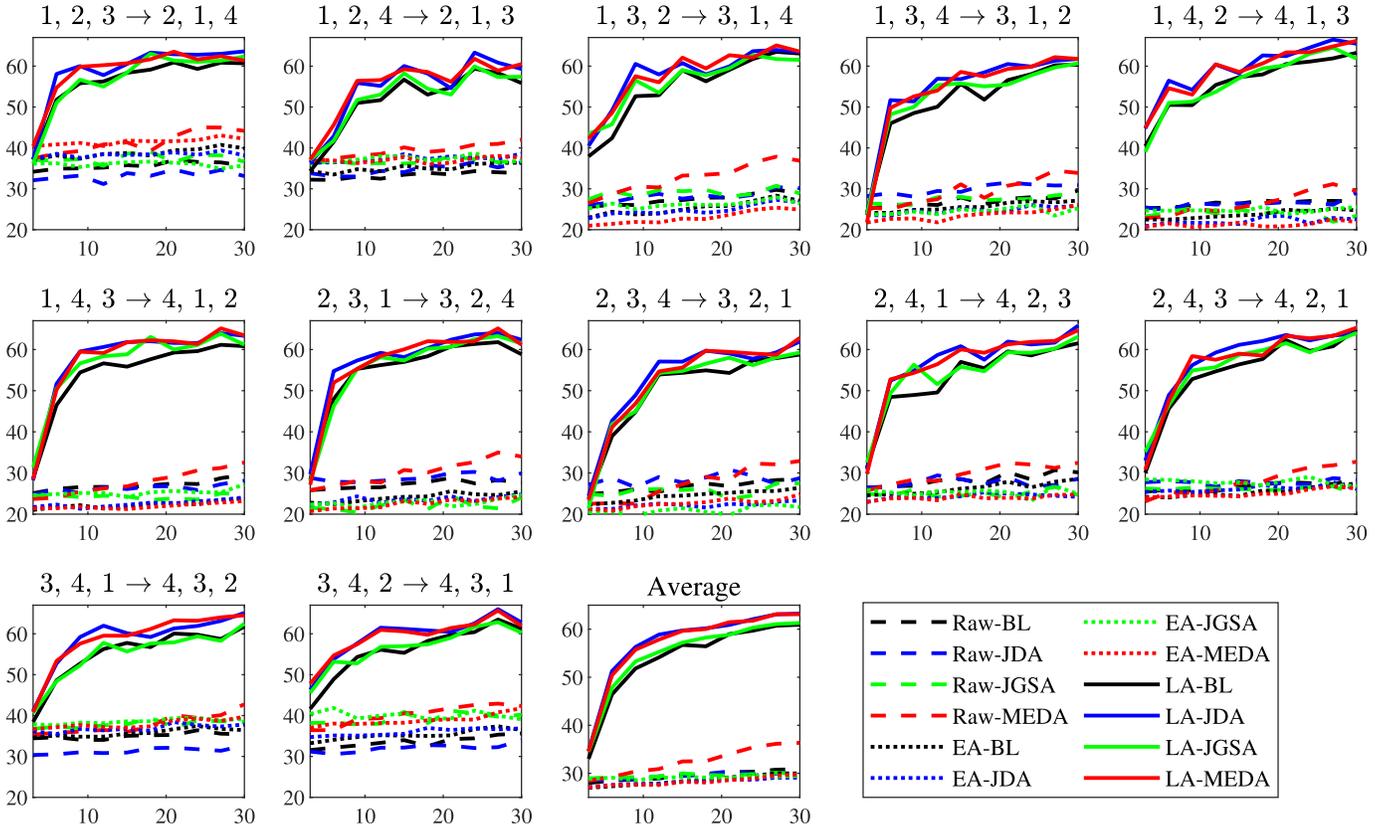


Fig. 12. Average classification accuracies (%) in Scenario II-b, when other DA approaches were used after LA. The horizontal axis indicates the number of labeled target subject trials (k in k -medoids clustering in Section III), and the vertical axis the classification accuracies.

TABLE X

SCENARIO II-B: p -VALUES OF PAIRED t -TESTS ON THE AUCs OF THE CLASSIFICATION ACCURACY CURVES IN FIG. 11. THE NULL HYPOTHESIS WAS REJECTED IF $p < \alpha$, WHERE $\alpha = 0.05$

	EA-CSP-LDA	EA-TS-SVM
LA-CSP-LDA	0.0000	
LA-TS-SVM		0.0000

TABLE XI

SCENARIO II-B: p -VALUES OF PAIRED t -TESTS ON THE AUCs IN FIG. 12. THE NULL HYPOTHESIS WAS REJECTED IF $p < \alpha$, WHERE $\alpha = 0.05$

	LA-BL	EA-JDA	EA-JGSA	EA-MEDA
LA-BL		0.0000	0.0000	0.0000
LA-JDA	0.0000		0.0000	
LA-JGSA	0.0009			
LA-MEDA	0.0000			0.0000

Again, we combined Raw, EA, LA with different DA approaches and obtained 12 algorithms to be compared. Fig. 12 shows their performances on the 12 sub-dataset combinations, and the average across the 12 experiments. Observe that:

- 1) LA-BL always outperformed Raw-BL and EA-BL, LA-JDA always outperformed Raw-JDA and EA-JDA, LA-JGSA always outperformed Raw-JGSA and EA-JGSA, and LA-MEDA always outperformed Raw-MEDA and EA-MEDA. These suggest that LA was effective regardless of whether additional DA approaches were used or not.
- 2) LA-BL always outperformed Raw-JDA, Raw-JGSA and Raw-MEDA, suggesting that LA can outperform classical DA approaches such as JDA, JGSA and MEDA.
- 3) Generally, LA-JDA, LA-JGSA and LA-MEDA outperformed LA-BL, suggesting that it may be advantageous to integrate additional DA approaches with LA.

- 4) When the labels were mismatched, the algorithms without LA performed very poorly. However, the algorithms with LA performed consistently good, suggesting that LA can cope well with large label space discrepancies.

Table XI shows the results of paired t -tests on the AUCs in Fig. 12. The conclusions in binary classification still hold in multi-class classification: LA significantly outperformed EA and classical DA approaches, and its performance can be further significantly improved when integrated with other DA approaches.

G. Scenario III: Different Feature Spaces and Different Label Spaces

This subsection considers the most challenging scenario: the source and target subjects have different feature spaces and also completely different label spaces. We chose “Classes 3, 4” (“feet” and “tongue”) from Dataset 2a as the target dataset,

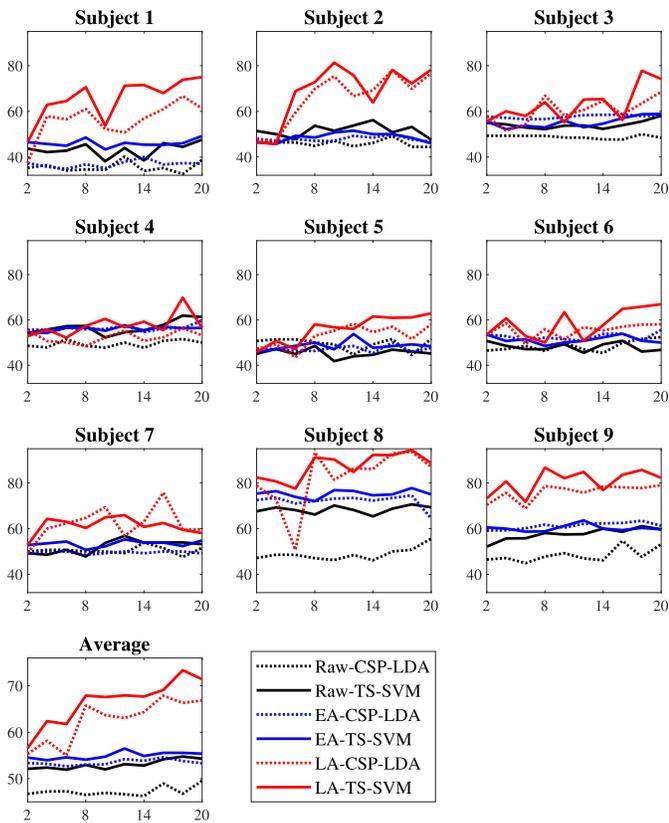


Fig. 13. Average classification accuracies (%) in Scenario III. The horizontal axis indicates the number of labeled target subject trials (k in k -medoids clustering in Section III), and the vertical axis the classification accuracies.

and Dataset 1 as the source dataset. Each time we picked one subject from “Classes 3, 4” as the target subject, and all seven subjects from Dataset 1 as the source subjects. In this scenario, the source dataset and target dataset were collected from different EEG headsets with different numbers of channels at different locations, so they had different feature spaces. In addition, for Dataset 1, Subjects 1 and 6 performed “left hand” and “feet” tasks, whereas other subjects performed “left hand” and “right hand” tasks. So, the source and target subjects also had partially or completely different label spaces.

Question 1: Can LA be used as an effective preprocessing step before different feature extraction and classification algorithms?

We selected the source EEG channels as those closest to the target EEG channels [14], and compared different algorithms. Fig. 13 shows the experimental results when LA was used before different feature extraction and classification algorithms, and Table XII shows the p -values of paired t -tests on the AUCs. LA-CSP-LDA significantly outperformed EA-CSP-LDA, and LA-TS-SVM significantly outperformed EA-TS-SVM. These suggest that LA was effective in different feature extraction and classification algorithms.

Question 2: Can LA be integrated with other DA approaches to further improve the classification performance?

Fig. 14 shows the experimental results with and without additional DA approaches. Generally, LA was effective

TABLE XII

SCENARIO III: p -VALUES OF THE PAIRED t -TESTS ON THE AUCs OF THE ACCURACY CURVES IN Fig. 13. THE NULL HYPOTHESIS WAS REJECTED IF $p < \alpha$, WHERE $\alpha = 0.05$

	EA-CSP-LDA	EA-TS-SVM
LA-CSP-LDA	0.0082	
LA-TS-SVM		0.0006

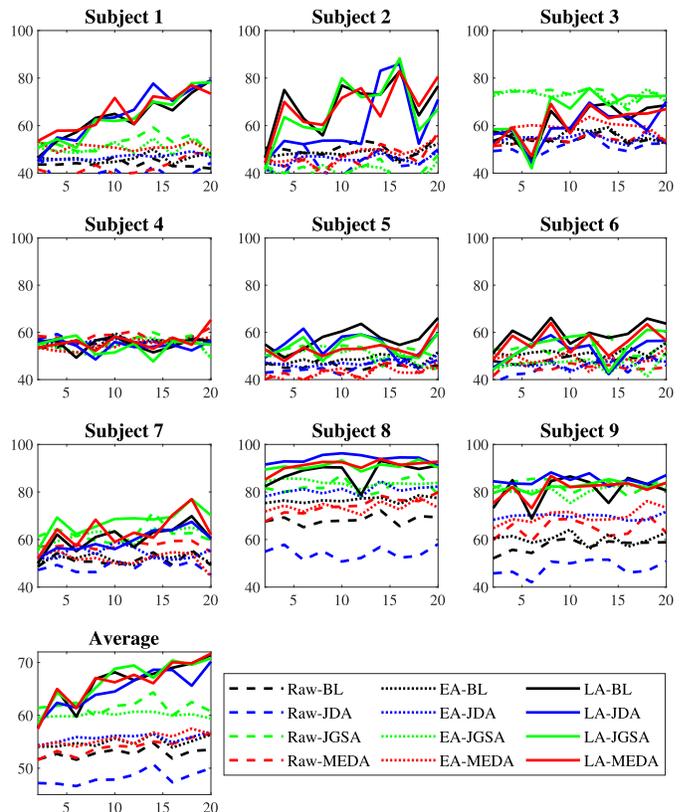


Fig. 14. Average classification accuracies (%) in Scenario III, when additional DA approaches were used after LA. The horizontal axis indicates the number of labeled target subject trials (k in k -medoids clustering in Section III), and the vertical axis the classification accuracies.

regardless of whether additional DA approaches were used or not. Table XIII shows the p -values of paired t -tests on the AUCs in Fig. 14. LA-BL significantly outperformed EA-JDA and EA-MEDA, LA-JDA significantly outperformed EA-JDA, and LA-MEDA significantly outperformed EA-MEDA. However, unlike before, the integration of LA with other DA approaches did not significantly outperform LA-BL. Nevertheless, LA did not degrade the performance of these DA approaches, either.

H. Computational Complexity of LA

The time complexity of LA is $O(N_T^2)$, where N_T is the number of target domain trials. The most time-consuming operation in LA is k -medoids clustering.

We also empirically evaluated the computational cost of LA in Scenario III, by comparing Raw-TS-SVM and LA-TS-SVM. The platform was a Lenovo ThinkPad laptop with Intel

TABLE XIII

SCENARIO III: p -VALUES OF THE PAIRED t -TESTS ON THE AUCs OF THE ACCURACY CURVES IN FIG. 14. THE NULL HYPOTHESIS WAS REJECTED IF $p < \alpha$, WHERE $\alpha = 0.05$

	LA-BL	EA-JDA	EA-JGSA	EA-MEDA
LA-BL		0.0017	0.1335	0.0011
LA-JDA	0.3733	0.0026		
LA-JGSA	0.8449		0.0691	
LA-MEDA	0.9777			0.0012

TABLE XIV

THE COMPUTING TIME (SECONDS) OF RAW-TS-SVM AND LA-TS-SVM IN SCENARIO III

	mean	std
Raw-TS-SVM	2.1963	0.1492
LA-TS-SVM	2.3669	0.3469

Core i5-6200U CPU@2.30GHz, 4GB memory, and 190 GB SSD, running 64-bit Windows 10 and Matlab 2018b. The results are shown in Table XIV, which were averaged across different numbers of labeled target trials (from 2 to 20) and nine target subjects. LA only increased the computing time very slightly.

VI. CONCLUSIONS AND FUTURE RESEARCH

Transfer learning, or domain adaptation, has been successfully used to reduce the subject-specific calibration effort in BCIs. However, most existing DA approaches require the source subjects share the same feature space and also the same label space as the target subject, which may not always hold in real-world applications. This paper has proposed a simple yet effective LA approach to cope with different label spaces. Our experiments demonstrated that: 1) LA only needs as few as one labeled sample from each class of the target subject; 2) LA can be used as a preprocessing step before different feature extraction and classification algorithms; and, 3) LA can be integrated with other DA approaches to achieve even better classification performance.

The current LA may still have some limitations, which will be addressed in our future research:

- 1) The estimation of each class mean in the target domain is very important to the performance of LA. Currently LA uses k -medoids clustering to select a few trials to label, which could be improved.
- 2) LA copes well with different labels spaces, but does not pay special attention to different feature spaces (although it can also be used in this case). This may explain why there were relatively less performance improvements when integrated with other DA approaches in Scenario III. We will specifically consider different feature spaces in the future.
- 3) The current LA approach was specifically designed for EEG trials, and uses 2D covariance matrices as the input features. We will extend it to 1D features so that it can have broader applications in other domains.

REFERENCES

- [1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control," *Clin. Neurophysiol.*, vol. 113, no. 6, pp. 767–791, 2002.
- [2] B. J. Lance, S. E. Kerick, A. J. Ries, K. S. Oie, and K. McDowell, "Brain-computer interface technologies in the coming decades," *Proc. IEEE*, vol. 100, no. 3, pp. 1585–1599, Mar. 2012.
- [3] Z. J. Koles, M. S. Lazar, and S. Z. Zhou, "Spatial patterns underlying population differences in the background EEG," *Brain Topography*, vol. 2, no. 4, pp. 275–284, 1990.
- [4] J. Müller-Gerking, G. Pfurtscheller, and H. Flyvbjerg, "Designing optimal spatial filters for single-trial EEG classification in a movement task," *Clin. Neurophysiol.*, vol. 110, no. 5, pp. 787–798, May 1999.
- [5] H. Ramoser, J. Müller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial EEG during imagined hand movement," *IEEE Trans. Rehabil. Eng.*, vol. 8, no. 4, pp. 441–446, Dec. 2000.
- [6] H. He and D. Wu, "Spatial filtering for brain computer interfaces: A comparison between the common spatial pattern and its variant," in *Proc. IEEE Int. Conf. Signal Process., Commun. Comput. (ICSPCC)*, Qingdao, China, Sep. 2018, pp. 1–6.
- [7] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "Multiclass brain-computer interface classification by Riemannian geometry," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 4, pp. 920–928, Apr. 2012.
- [8] F. Yger, M. Berar, and F. Lotte, "Riemannian approaches in brain-computer interfaces: A review," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 10, pp. 1753–1762, Oct. 2017.
- [9] B. He, B. Baxter, B. J. Edelman, C. C. Cline, and W. W. Ye, "Noninvasive brain-computer interfaces based on sensorimotor rhythms," *Proc. IEEE*, vol. 103, no. 6, pp. 907–925, Jun. 2015.
- [10] G. Pfurtscheller, G. R. Müller-Putz, R. Scherer, and C. Neuper, "Rehabilitation with brain-computer interface systems," *Computer*, vol. 41, no. 10, pp. 58–65, Oct. 2008.
- [11] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain computer interfaces, a review," *Sensors*, vol. 12, no. 2, pp. 1211–1279, 2012.
- [12] J. van Erp, F. Lotte, and M. Tangermann, "Brain-computer interfaces: Beyond medical applications," *Computer*, vol. 45, no. 4, pp. 26–34, Apr. 2012.
- [13] V. Jayaram, M. Alamgir, Y. Altun, B. Scholkopf, and M. Grosse-Wentrup, "Transfer learning in brain-computer interfaces," *IEEE Comput. Intell. Mag.*, vol. 11, no. 1, pp. 20–31, Feb. 2016.
- [14] D. Wu, V. J. Lawhern, W. D. Hairston, and B. J. Lance, "Switching EEG headsets made easy: Reducing offline calibration effort using active weighted adaptation regularization," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 11, pp. 1125–1137, Nov. 2016.
- [15] D. Wu, V. J. Lawhern, S. Gordon, B. J. Lance, and C.-T. Lin, "Driver drowsiness estimation from EEG signals using online weighted adaptation regularization for regression (OwARR)," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 6, pp. 1522–1535, Dec. 2017.
- [16] D. Wu, "Online and offline domain adaptation for reducing BCI calibration effort," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 4, pp. 550–563, Aug. 2017.
- [17] D. Wu, "Active semi-supervised transfer learning (ASTL) for offline BCI calibration," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Banff, AB, Canada, Oct. 2017, pp. 246–251.
- [18] H. He and D. Wu, "Transfer learning enhanced common spatial pattern filtering for brain computer interfaces (BCIs): Overview and a new approach," in *Proc. 24th Int. Conf. Neural Inf. Process.*, Guangzhou, China, Nov. 2017, pp. 811–821.
- [19] H. Kang, Y. Nam, and S. Choi, "Composite common spatial pattern for subject-to-subject transfer," *IEEE Signal Process. Lett.*, vol. 16, no. 8, pp. 683–686, Aug. 2009.
- [20] F. Lotte and C. Guan, "Learning from other subjects helps reducing brain-computer interface calibration time," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Dallas, TX, USA, Mar. 2010, pp. 614–617.
- [21] S. Jialin Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
- [22] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sydney, NSW, Australia, Dec. 2013, pp. 2200–2207.
- [23] J. Zhang, W. Li, and P. Ogunbona, "Joint geometrical and statistical alignment for visual domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 1859–1867.

- [24] J. Wang, W. Feng, Y. Chen, H. Yu, M. Huang, and P. S. Yu, "Visual domain adaptation with manifold embedded distribution alignment," in *Proc. ACM Multimedia Conf. Multimedia Conf. (MM)*, Seoul, South Korea, Oct. 2018, pp. 402–410.
- [25] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: A survey," *Knowl.-Based Syst.*, vol. 80, pp. 14–23, May 2015.
- [26] P. Zanini, M. Congedo, C. Jutten, S. Said, and Y. Berthoumieu, "Transfer learning: A Riemannian geometry framework with applications to brain–computer interfaces," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 5, pp. 1107–1116, May 2018.
- [27] H. He and D. Wu, "Transfer learning for brain-computer interfaces: A Euclidean space data alignment approach," *IEEE Trans. Biomed. Eng.*, vol. 67, no. 2, pp. 399–410, Feb. 2020.
- [28] O. Day and T. M. Khoshgoftaar, "A survey on heterogeneous transfer learning," *J. Big Data*, vol. 4, no. 1, p. 29, Dec. 2017.
- [29] F. Liu, J. Lu, and G. Zhang, "Unsupervised heterogeneous domain adaptation via shared fuzzy equivalence relations," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 6, pp. 3555–3568, Dec. 2018.
- [30] P. P. Busto and J. Gall, "Open set domain adaptation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 754–763.
- [31] K. Saito, S. Yamamoto, Y. Ushiku, and T. Harada, "Open set domain adaptation by backpropagation," in *Proc. Eur. Conf. Comput. Vis.*, Munich, Germany, Sep. 2018, pp. 153–168.
- [32] Z. Fang, J. Lu, F. Liu, J. Xuan, and G. Zhang, "Open set domain adaptation: Theoretical bound and algorithm," 2019, *arXiv:1907.08375*. [Online]. Available: <http://arxiv.org/abs/1907.08375>
- [33] K. You, M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Universal domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 2720–2729.
- [34] H. He and D. Wu, "Channel and trials selection for reducing covariate shift in EEG-based brain-computer interfaces," in *Proc. IEEE Int. Conf. Syst., Man Cybern. (SMC)*, Bari, Italy, Oct. 2019, pp. 3635–3640.
- [35] A. Gretton, K. M. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola, "A kernel method for the two-sample-problem," in *Proc. 19th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, Vancouver, BC, Canada, Dec. 2007, pp. 513–520.
- [36] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," in *Proc. 30th AAAI Conf. Artif. Intell.*, Phoenix, AZ, USA, Feb. 2016, vol. 6, no. 7, pp. 2058–2065.
- [37] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function," *J. Stat. Planning Inference*, vol. 90, no. 2, pp. 227–244, Oct. 2000.
- [38] M. Sugiyama, S. Nakajima, H. Kashima, P. von Büna, and M. Kawanabe, "Direct importance estimation with model selection and its application to covariate shift adaptation," in *Proc. 21st Annu. Conf. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2007, pp. 1433–1440.
- [39] P. E. Utgoff, "Shift of bias for inductive concept learning," in *Machine Learning: An Artificial Intelligence Approach*, vol. 2. San Mateo, CA, USA: Morgan Kaufmann, 1986, pp. 107–148.
- [40] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Geometric means in a novel vector space structure on symmetric positive-definite matrices," *SIAM J. Matrix Anal. Appl.*, vol. 29, no. 1, pp. 328–347, 2007.
- [41] B. Blankertz, G. Dornhege, M. Krauledat, K. R. Müller, and G. Curio, "The non-invasive Berlin brain-computer interface: Fast acquisition of effective performance in untrained subjects," *NeuroImage*, vol. 37, no. 2, pp. 539–550, 2007.
- [42] A. Delorme and S. Makeig, "EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis," *J. Neurosci. Methods*, vol. 134, no. 1, pp. 9–21, Mar. 2004.
- [43] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K.-R. Müller, "Optimizing spatial filters for robust EEG single-trial analysis," *IEEE Signal Process. Mag.*, vol. 25, no. 1, pp. 41–56, 2008.
- [44] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.
- [45] G. Dornhege, B. Blankertz, G. Curio, and K.-R. Müller, "Boosting bit rates in noninvasive EEG single-trial classifications by feature combination and multiclass paradigms," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 993–1002, Jun. 2004.