IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 18, 2019, accepted March 29, 2019, date of publication April 15, 2019, date of current version April 25, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2908851

Domain Transfer Multiple Kernel Boosting for Classification of EEG Motor Imagery Signals

MENGXI DAI[1], SHUAI WANG[2], DEZHI ZHENG[1,3], RUI NA[1], AND SHUAILEI ZHANG[1]

[1]School of Instrumentation and Optoelectronic Engineering, Beihang University, Beijing 100191, China
[2]School of Computer Science and Engineering, Beihang University, Beijing 100191, China
[3]Beijing Advanced Innovation Center for Big Data-based Precision Medicine, Beihang University, Beijing 100191, China

Corresponding authors: Shuai Wang (wangshuai@buaa.edu.cn) and Dezhi Zheng (zhengdezhi@buaa.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61873021, in part by the Youth Talent Support Program of Beihang University, and in part by the National Key Scientific Instrument and Equipment Development Project under Grant 2014YQ350461.

ABSTRACT The application of wireless sensors in the brain–computer interface (BCI) system provides great convenience for the acquisition of electroencephalography (EEG) signals. However, a large amount of training data is needed to build the classification architectures used in motor imagery (MI) brain–computer interface (BCI), which is time-consuming to generate. To address this issue, transfer learning has gained significant attention in a small sample setting BCI system. The transfer learning methods have shown promising results by leveraging labeled patterns from the source domain to learn robust classifiers for the target domain, which has only a limited number of labeled samples. However, the successful application of such approaches in a motor imagery BCI remains limited. In this paper, we present a novel framework called domain transfer multiple kernel boosting (DTMKB), which extends the DTMKL algorithms by applying boosting techniques for learning kernel-based classifiers with the transfer of multiple kernels. Based on the proposed framework, we examined their empirical performance in comparison to several state-of-the-art algorithms on two MI task datasets. DTMKB yields the best performance for all datasets and achieves the best average classification accuracy 87.60%, 76.00%, 74.66%, and 74.13%, respectively. In particular, the proposed framework can be applied successfully in a small sample of EEG motor imagery signals.

INDEX TERMS Brain-computer interface EEG, transfer learning, boosting, domain transfer multiple kernel boosting.

I. INTRODUCTION

The brain computer interface (BCI) is an alternative method of communication between a user and system that depends on neither the brain's normal output nerve pathways nor the muscles [1], [2]. Electroencephalography (EEG) is an important tool for recording functional brain activity. It is suited for investigating the mechanism of brain functions. The EEG obtains time series data with multiple variants recorded at several sensors pressed on the scalp. It thereby presents electrical potentials under the induction of brain activities. Studies have proved that EEG measurements taken during the mental imagining of different movements can be translated into different commands. In Motor Imagery(MI)-based BCI systems, the imagination of body movements is accompanied by a circumscribed event-related synchronization/desynchronization(ERS/ERD). Thus, motor imagery has been widespread used as a major approach in BCI systems [3], [4].

In recent years, the classification performance of BCI systems based on an EEG has faced significant challenges. For one, wireless BCI devices make BCI systems be wearable easily [5]–[7]. For another one, it is necessary for a fresh subject to conduct a lengthy calibration session for a sufficient collection of training samples to establish the classifiers. In a recent study on BCI, it was shown to be very important to reduce the number training sessions owing to the tedious, time-consuming process of a calibration session. As a result, conducting a performance promotion using a scarce labeled set is more desirable when compared with a large one. Nevertheless, suitable methods must be identified to strengthen the performance because a short calibration session means

The associate editor coordinating the review of this manuscript and approving it for publication was Qilian Liang.

the availability of only a few training samples for the target users, which may result in an overfitting or suboptimal feature classifiers or extractors.

To address the above problems, transfer learning is a promising approach [8], [9]. It applies data represented in various feature spaces or obtained from various distributions for compensating insufficiently labeled data. In the BCI field, transfer learning has attracted considerable attention because it enables the establishment of subject-independent spatial classifiers and/or filters, and lowers the calibration times. MI has been the most widely used paradigm for the testing of transfer learning methods, probably owing to the availability of datasets from BCI competitions [10]–[16].

A particularly productive research study has focused on the building of spatial filters based on EEG signals. Common spatial patterns (CSP) and spatial filters have the potential to achieve the rapid learning of proper training data, but do not perform well with a large amount of heterogeneous data recorded from other subjects or other sessions [17]. A regularization strategy in this case is effective [10]. A more relevant approach is to directly regularize the CSP objective function rather than the covariance matrices [11], [12], [21]. Furthermore, the use of a subspace strategy, which can construct a common feature space from other subjects, is a good idea. [13]–[16], [19]–[20] Deep transfer learning is a new development trend [22].

Many transfer learning methods are also based on finding classification approaches for two or more domains. Fazli et al. [18] built a subject-independent classifier for movement imagination detection. They first extracted an ensemble of features (spatial and frequency filters) and then applied LDA classifiers across all subjects. They compared various ways of combining these classifiers to classify the data of a new subject. In addition, a classic strategy is to match the distributions of the source and target domains. These transformations can either be linear or non-linear, for instance, based on the kernel methods applied [23], [24]. Some methods for combining the features and classifiers within ensembles have been developed [25]. The first concern when considering an ensemble is to guarantee the quality of the features and classifiers from the source domain.

In this study, we used filter bank regularized common spatial patterns (FBRCSP) as the feature extraction method [12]. Herein, we propose a transfer learning classification, referred to as domain transfer multiple kernel boosting (DTMKB), which adapts the boosting techniques for classifiers used in domain transfer multiple kernels learning (DTMKL) [24]. DTMKL can evaluate a distribution mismatch between the target and source domains. The intuitive idea of multiple kernel boosting is to employ a boosting framework to learn an ensemble of multiple base kernel classifiers, each of which is equipped with one base kernel. The weights for both the kernel and classifier combination can be easily determined through the boosting process. As a result, we can efficiently learn a classifier with the transfer of multiple kernels and a boosting strategy.

The rest of this paper is organized as follows. Section II describes the proposed method of domain transfer multiple kernels learning. Section III details the BCI competition III IVa dataset and our own dataset, as well as the experiments conducted. Section IV discusses the empirical study. Section V provides some concluding remarks regarding the proposed method.

II. METHOD

In this section, we present the novel framework of DTMKB, which adapts the idea of boosting for DTMKL. Before presenting the DTMB algorithms, we first introduce the problems and review the DTMKL. Based on the proposed framework, we further propose a method using a support vector learning (SVM) classifier.

A. DTMKL

Let us denote the data set patterns from the target domain as $D^T = \left(x_i^T, y_i^T\right)|_{i=1}^T$, where y_i^T is the label of x_i^T. We also define D^T as the dataset from the target domain under the marginal data distribution P, and $D^S = \left(x_i^S, y_i^S\right)|_{i=1}^S$ as the dataset from the source domain under the marginal data distribution Q. The training data can be from both domains ($D = D^T \bigcup D^S$).

In transfer learning, it is crucial to reduce the difference between the data distributions of the source and target domains. Many parametric criteria (e.g., KL divergence) have been used to measure the distance between data distributions. However, an intermediate density estimate process is usually required. To avoid such a nontrivial task, Borgwardt *et al.* [26] proposed an effective nonparametric criterion, referred to as the Maximum Mean Discrepancy (MMD), to compare data distributions based on the distance between the means of the samples from two domains in a kernel k induced reproducing kernel Hilbert space (RKHS) H, namely,

$$\mathrm{DIST}_k\left(D^S, D^T\right) = \left\| \frac{1}{n_s}\sum_{i=1}^{n_s} \phi\left(x_i^S\right) - \frac{1}{n_T}\sum_{i=1}^{n_T} \phi\left(x_i^T\right) \right\|_H \tag{1}$$

In previous cross-domain learning methods [27], [28], the weights or the kernel matrix of the samples are learned separately using the MMD criterion in (2) without considering any label information. However, it is usually beneficial to utilize label information during kernel learning. Instead of using the two-step approaches as in [27], [28], we propose a unified cross-domain learning framework, DTMKL, to learn the decision function for the target domain

$$f(x) = \omega' \phi(x) + b = \sum_{i=1}^{n} \beta_i k(x_i, x) + b \tag{2}$$

as well as the kernel function k simultaneously, where ω is the weight vector in the feature space, and b is the bias term. Note that β_i is the coefficient of the kernel expansion for the decision function $f(x)$ using the representer theorem [25]. In practice, DTMKL minimizes the distance between the data

distributions of the source and target domains, as well as the structural risk function of any kernel method. The learning framework of DTMKL is then formulated as

$$[k, f] = \underset{k,f}{\operatorname{argmin}}\, \Omega\left(\mathrm{DIST}_k^2\left(D^S, D^T\right)\right) + \theta R\left(k, f, D\right) \quad (3)$$

where $\Omega(\cdot)$ is any monotonic increasing function, and $\theta > 0$ is a tradeoff parameter used to balance a mismatch between data distributions of two domains and the structural risk function $R\left(k, f, D\right)$ defined on the labeled patterns.

The first objective in DTMKL is to minimize the mismatch between data distributions of two domains using the MMD criterion defined in (2). Let $\Phi = [\phi\left(x_1^S\right), \cdots, \phi\left(x_{n_s}^S\right), \phi\left(x_1^T\right), \cdots, \phi\left(x_{nT}^T\right)]$ be the kernel matrix after feature mapping, and $\frac{1}{n_S}\sum_{i=1}^{n_S}\phi\left(x_i^S\right) - \frac{1}{n_T}\sum_{i=1}^{n_T}\phi\left(x_i^T\right)$ in (2) is then simplified as Φ_S. We define a column vector $\mathrm{s} = [1/n_1, \cdots, 1/n_S, -1/n_1, \cdots, -1/n_T]$. Thus, the criterion in (2) can be rewritten as

$$\mathrm{DIST}_k^2\left(D^S, D^T\right) = \|\Phi_S\|^2 = \mathrm{tr}\left(\Phi^{'}\Phi\mathbf{S}\right) = \mathrm{tr}\left(\mathbf{KS}\right) \quad (4)$$

where

$$\mathbf{S} = s^{'}s \in R^{(n_S+n_T)\times(n_S+n_T)},$$
$$\mathbf{K} = \Phi^{'}\Phi = \begin{bmatrix}\mathbf{K}^{S,S} & \mathbf{K}^{S,T}\\ \mathbf{K}^{S,T} & \mathbf{K}^{T,T}\end{bmatrix} \in R^{(n_S+n_T)\times(n_S+n_T)},$$

$\mathbf{K}^{S,S} \in R^{n_S\times n_S}$, $\mathbf{K}^{T,T} \in R^{n_T\times n_T}$, and $\mathbf{K}^{S,T} \in R^{n_S\times n_T}$ are the kernel matrices defined for the source domain, target domain, and cross domain from the source domain to the target domain, respectively.

Instead of learning a nonparametric kernel matrix $\mathbf{K}$ in (4) for cross-domain learning, as in [21], following [29], we assume kernel k as a linear combination of a set of base kernels k_ms, namely,

$$k = \sum_{m=1}^{M} d_m\mathbf{K}_m \quad (5)$$

where $d_m \geq 0, \sum_{m=1}^{M} d_m = 1$. We further assume the first objective $\Omega(\mathrm{tr}(\mathbf{KS}))$ in (4) is

$$\Omega\left(\mathrm{tr}\left(\mathbf{KS}\right)\right) = \frac{1}{2}\left(tr\left(\mathbf{KS}\right)\right)^2 = \frac{1}{2}\left(tr\left(\sum\nolimits_{m=1}^{M} d_m\mathbf{K}_m\mathbf{S}\right)\right)^2 = \frac{1}{2}\mathbf{d}^{'}\mathbf{p}\mathbf{p}^{'}\mathbf{d} \quad (6)$$

where $\mathbf{p} = [p_1, \cdots, p_M]'$, $p_m = tr(\mathbf{K}_m\mathbf{S})$, $\mathrm{K}_m = \left[k_m\left(x_i, x_j\right)\right] \in R^{(n_S+n_T)\times(n_S+n_T)}$, $\mathbf{d} = [d_1, \cdots, d_M]'$. Moreover, from (3) we have $f\left(x\right) = \sum_{m=1}^{M} d_m\omega_m^{'}\phi_m\left(x\right) + b$, where $\omega_m = \sum_{i=1}^{n}\beta_i\phi_m(x_i)$.

Thus, the optimization problem in (4) can be rewritten as

$$\underset{k,f}{\operatorname{argmin}}\Omega\ \left(\mathrm{DIST}_k^2\left(D^S, D^T\right)\right) + \theta R\left(k, f, D\right) \quad (7)$$

$$\min_{k,f}\frac{1}{2}\mathbf{d}^{'}\mathbf{p}\mathbf{p}^{'}\mathbf{d} + \min\theta J\left(\mathbf{d}\right) \quad (8)$$

where $J\left(\mathbf{d}\right) = R(\mathbf{d}, f, D)$, $D = \{\mathbf{d}|\mathbf{d} \geq 0, \mathbf{d}'1_M = 1\}$ is the feasible set of $\mathbf{d}$, and f is the target decision function.

Following [29], we developed an efficient and effective reduced gradient descent procedure to iteratively update different variables (e.g., $\mathbf{d}$ and f) in (5) to obtain the optimal solution. In detail, the algorithm updates the decision function f. With a fixed d, only the structural risk functional $R(\mathbf{d}, f, D)$ in (5) depends on f. We can solve the decision function f by minimizing $R(\mathbf{d}, f, D)$.

For updating the kernel coefficients $\mathbf{d}$, when the decision function f is fixed, (7) can be updated using the reduced gradient descent method, as suggested in [29]. Specifically, the gradient of h in (7) is

$$\nabla h = \mathbf{p}\mathbf{p}^{'}\mathbf{D} + \theta\nabla J \quad (9)$$

where ∇J is the gradient of J in (6). Furthermore, the Hessian matrix can be derived as

$$\nabla^2 h = \mathbf{p}\mathbf{p}^{'} + \theta\nabla^2 J \quad (10)$$

Compared with first-order gradient-based methods, second-order derivative-based methods usually converge more quickly. Thus, we define $g = (\nabla^2 h)^{-1}\nabla h$ as the updating direction. To maintain $\mathbf{d} \in D$, the updating direction g is reduced as in [29], and thus the updated weight of multiple base kernels is

$$\mathrm{d}_{t+1} = \mathrm{d}_t - \eta_t\mathrm{g}_t \in D \quad (11)$$

where d_t and g_t are the linear combination coefficient vector $\mathbf{d}$ and the reduced updating direction g at the t-th iteration, respectively, and η_t is the learning rate. The overall procedure of the proposed DTMKL is summarized in Algorithm 1.

Algorithm 1 DTMKL Algorithm

(1) INPUT: training data: $D = D^T \bigcup D^S$;
kernel functions: $\mathbf{K}^{S,S} \in R^{n_S\times n_S}$, $\mathbf{K}^{T,T} \in R^{n_T\times n_T}$, $\mathbf{K}^{S,T} \in R^{n_S\times n_T}$;
initial distribution: $\mathbf{d} = \frac{1}{M}1_M$

(2) for $t = 1, \cdots, T_{max}$

(3) solve the target classifier f in the objective function in $\theta\nabla J$:
$\min_{\mathbf{d}\in D} = \min_{\mathbf{d}\in D}\frac{1}{2}\mathbf{d}'\mathbf{p}\mathbf{p}'\mathbf{d} + \min_f\theta\nabla J(\mathbf{d})$

(4) update the linear combination coefficient vector d of multiple base kernels:
$\mathrm{d}_{t+1} = \mathrm{d}_t - \eta_t\mathrm{g}_t \in D$
end

(5) OUTPUT: $f\left(x\right) = \mathrm{sgn}(\sum_{i=1}^{M}\beta_i K\left(x_i, x\right) + b$
$\mathrm{K} = \sum_{m=1}^{M} d_m\mathbf{K}_m$

As mentioned before, any structural risk function of a kernel method can be employed in the learning framework of DTMKL. For the preliminary conference version of this approach [30], we proposed the use of the hinge loss in an SVM. Here, the risk function becomes an SVM, which is the first formulation in the present paper. Moreover, inspired by the utilization of source classifiers for cross-domain learning, we also propose another formulation that considers the decision values from the base classifiers on the unlabeled patterns in the target domain.

B. DTMKB

The general idea of DTMKB is to apply boosting techniques to learn a classifier using the transfer of multiple kernels. For this purpose, we follow the typical procedure of a boosting algorithm, i.e., Adaptive Boosting, known as "Adaboost" [31], which is a popular and successful boosting technique. Figure 1 shows the schematic illustration of the Adaboost framework.

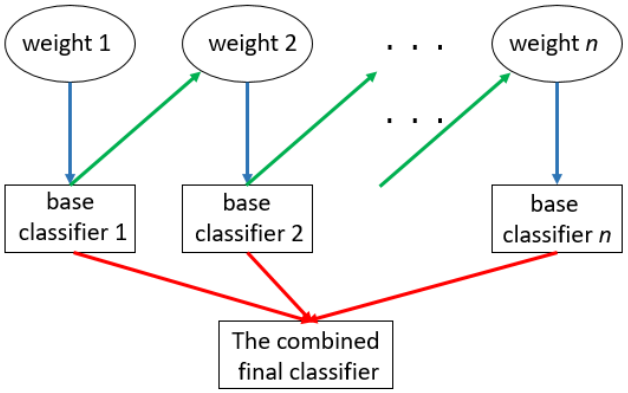

FIGURE 1. Schematic illustration of the boosting framework. Each base classifier is trained on a weighted form of the training set (blue arrows) in which the weights depend on the performance of the previous base classifier (green arrows). Once all base classifiers have been trained, they are combined to give the final classifier (red arrows).

The goal of our multiple kernel boosting task is to learn a kernel-based classifier f, which is an ensemble of kernel classifiers using a collection of M kernel classifiers from the given training data examples. Typically, we can express such a kernel classifier as follows:

$$f(x_i) = \sum_{i=t}^{T} \alpha_t f_t(x_i) \tag{12}$$

where f_t is a hypothesis learned from a boosting trial, α_t is its associated weight in the final classifier, and T is the total number of boosting trials. The main challenge of multiple kernel boosting is how to develop an effective boosting scheme to learn the optimal hypothesis f_t and its combination weight α_t at each boosting trial.

In particular, we repeatedly learn some kernel classifiers with multiple kernels f_t through a series of boosting trials ($t = 1, \cdots, T$), where T denotes the total number of boosting trials. For each boosting trial, a distribution of weights D_t is engaged to indicate the importance of the training examples for the classification. At each boosting trial, we increase the weights of the wrongly classified examples and/or decrease the weights of those correctly classified examples in order to focus on those examples that are difficult to be correctly classified.

During each boosting trial, we first sample a subset of n training examples according to distribution D_t. The major concern involves the method for learning the kernel-based classifier f_t from these training data. The first method is to learn one classifier $h_t^j t$ with every kernel k_j from the set of M kernels using the transfer kernel method used in our study. Based on M base kernels, we are able to further measure the misclassification of each classifier h_t^j with kernel $\mathbf{K}_j$ ($\mathbf{K}_m$ in (5))over distribution D_t for all training data accumulated:

$$\epsilon_t^j = \epsilon\left(h_t^j\right) = \sum_{i=1}^{N} D_t(i)\left(h_t^j(x_i) \neq y_i\right) \tag{13}$$

As a result, we can construct the classifier f_t for the t-th boosting trial via choosing the best classifier with the smallest misclassification rate:

$$h_t = \arg\min_{j \in 1,\ldots,j} \epsilon\left(h_t^j\right) \tag{14}$$

The last step of each boosting trial is to update the weight $D_{t+1}(i)$:

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t}, & h_t(x_i) = y_i \\ e^{\alpha_t}, & h_t(x_i) \neq y_i \end{cases} \tag{15}$$

where $\alpha_t = \frac{1}{2}\ln\frac{1-\epsilon_t}{\epsilon_t}$ and Z_t is a normalization factor to make D_{t+1} a distribution. Finally, after T boosting trials, the final classifier is as (16). Algorithm 2 shows a summary of a complete DTMKB procedure. And Figure 1 shows the schematic illustration of the DTMKB framework.

$$h(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right) \tag{16}$$

Algorithm 2 DTMKB Algorithm

(1) INPUT: training data: $D = D^T \bigcup D^S$;
transfer kernel functions: $\mathbf{K}_j, J = 1, \cdots, J$
initial:$D_1(i) = 1/N, i = 1, \cdots, N$
(2) for $t = 1, \cdots, T$
sample n examples using distribution D_t
(3) for $t = 1, \cdots, T$
train weak classifier with kernel $\mathbf{K}_j$:
$h_t^j : X \to \{-1, +1\}$
(4) compute the training error over D_t:
$\epsilon_t^j = \epsilon\left(h_t^j\right) = \sum_{i=1}^{N} D_t(i)(h_t^j(x_i) \neq y_i)$
end
(5) select the best classifier with the error rate:
$h_t = \arg\min_{j \in 1,\ldots,j} \epsilon\left(h_t^j\right)$
(6) update $D_{t+1}(i)$:
$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t}, h_t(x_i) = y_i \\ e^{\alpha_t}, h_t(x_i) \neq y_i \end{cases}$
end
(7) OUTPUT: $f(x) = \text{sgn}(\sum_{i=1}^{T} \alpha_t h_t(\text{x})$

III. DATASETS

In this study, we employed the IVa dataset from BCI Competition III (BCIC III) [32], because this dataset is widely used in transfer learning of EEG signals [10]–[13]. This dataset includes EEG data containing a classification task of motor

imagery with two levels: 1) imagery movement of the right hand (denoted by "R") and 2) imagery movement of the right foot (denoted by "F"). 118 electrodes were employed to measure the EEG signals during every trial from five different subjects, each of whom participated in a total of 280 trials. In addition, the 118 EEG electrodes is shown in Figure 1(a).

The second dataset is for five male subjects (aged 23 to 30). During the experiment, these subjects were seated in front of a computer screen and followed the instructions to conduct two MI tasks (right hand and feet), and each of them participated in a total of 200 trials. The EEG data used in this paper were sampled at a rate of 250 Hz. The EEG used for processing was recorded from Ag-AgCl electrodes placed according to the extended international 10-20 system. The 64 channels shown in Fig. 1(b) were employed.

During this experiment, the dataset included three sessions for the training set. Each trial was structured as shown in Figure 2. First, a "rest" icon was shown on the screen to remind the subjects to rest for 3s with their body relaxed, followed by a "prepare" indicator allowing the subjects to prepare for the task for 3s. Then, a "begin" icon was shown on screen for 0.5s. Finally, one of these two patterns (i.e., right hand or feet) was displayed for 6s while the subjects imaged to move their hands or feet. (i.e., the MI tasks.)

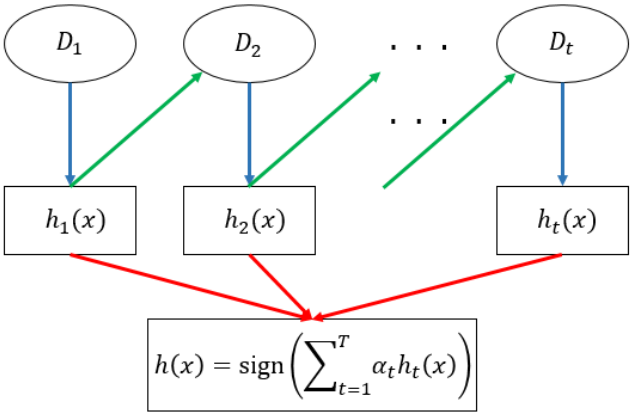

FIGURE 2. Schematic illustration of the DTMKB framework. Each transfer kernel base classifier $\mathbf{h_t(x)}$ is trained on a weighted form of the training set (blue arrows) in which the weights $\mathbf{D_t}$ depend on the performance of the previous transfer kernel base classifier $\mathbf{h_{t-1}(x)}$ (green arrows). Once all base classifiers have been trained, they are combined to give the final classifier h(t) (red arrows).

According to [33], [34], to measure the performance of the transfer learning framework, the training set size ought to be different. Table 1 shows a summary of the IVa dataset from BCI Competition III, in which the number of training samples of subjects ay, aw, and av are fewer than those of the test samples. The ratio of the number of training trials to the number of test trials of five subjects (aa, al, av, aw, ay) are respectively 1.50, 4.00, 0.43, 0.25 and 0.11. For our own dataset, among the 200 trials for every subject on our own dataset, different number of trials selected for the training set, with the remaining trials assigned to the test set. As shown

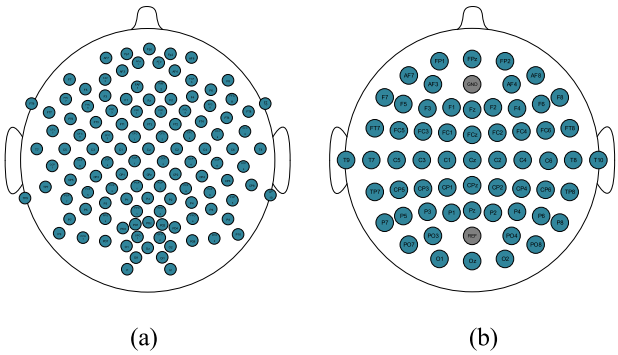

FIGURE 3. Positions of EEG channels for both datasets. (a) is the position for BCI Competition III IVa dataset; (b) is the position for our own dataset.

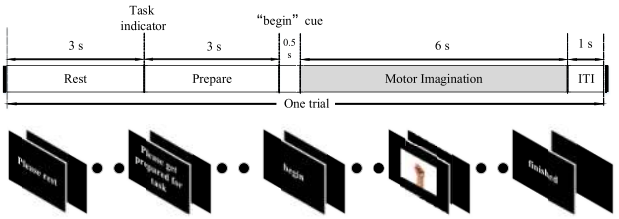

FIGURE 4. Diagram of a trial and timings during a session for our own dataset.

TABLE 1. Data description for BCI Competition III IVa dataset.

Subject	aa	al	av	aw	ay
Number of training samples	168	224	84	56	28
Number of test samples	112	56	196	224	252

in Table 2, our own dataset is divided into three groups (our own dataset I, II and III) due to the different size of training samples. The ratio of the number of training trials to the number of test trials for our own dataset I, II, III are respectively 1.22, 0.67 and 0.33. Each trial was considered an $M \times T$ matrix E_i, in which M represents the number of electrodes, and T is the number of time points sampled. The EEG signals measured were band-pass decomposed (at 8 to 30 Hz). FBRCSP was utilized as the feature extraction method.

TABLE 2. Data description for our own dataset.

Subject	I D,Z,L,X,S	II D,Z,L,X,S	III D,Z,L,X,S
Number of training samples	110	80	50
Number of test samples	90	120	150

Our establishment of a dataset (containing the target and source domains) for cross-domain classification is described as follows. Each dataset of each subject could become the target domain, whereas the datasets of other subjects could become the source domain. This dataset construction strategy ensured the relevance between the domains of the unlabeled and labeled data, as they were located in the same top-level categories. Accordingly, $C_4^1 + C_4^2 + C_4^3 + C_4^4 = 15$ datasets of

the source domains were generated for the target domains of every dataset. It was possible to generate five dataset groups, with a total of $5 \times 15 = 75$ datasets.

IV. RESULTS AND DISCUSSIONS

A. RESULTS

In this section, DTMKB and three competitive methods are evaluated based on the classification accuracy. There are five subjects in each dataset, and each subject can correspond to a dataset group. Each dataset can establish five dataset groups, each dataset group has one target subject and four source subjects ($C_4^1 + C_4^2 + C_4^3 + C_4^4 = 15$ source domains). Table 3, Table 4, Table 5 and Table 6 show the five dataset groups of BCI Competition III Iva, our own dataset I, our own dataset II, our own dataset III respectively. The first columns of the Tables show the different source domains, and The first rows of the Tables show the different target domains.

TABLE 3. Classification accuracy of DTMKB on BCI Competition III IVa dataset.

source \ target	aa(%)	al(%)	av(%)	aw(%)	ay(%)
aa	-	80.42	50.00	77.68	88.10
al	**91.96**	-	65.30	**81.25**	84.92
av	50.00	82.20	-	70.54	56.35
aw	58.17	**96.43**	57.65	-	64.68
ay	50.00	87.54	62.24	62.50	-
aa+al	-	-	57.65	75.00	**92.86**
aa+av	-	84.28	-	69.20	72.22
aa+aw	-	86.06	62.24	-	65.47
aa+ay	-	84.28	-	71.88	72.62
al+av	59.06	-	-	71.43	50.00
al+aw	85.76	-	62.24	-	54.76
al+ay	75.89	-	**75.52**	66.52	-
av+aw	50.00	86.06	-	-	50.00
av+ay	50.00	89.62	-	65.62	-
aw+ay	62.50	91.40	61.22	-	-
aa+al+av	-	-	-	75.45	74.60
aa+al+aw	-	-	50.00	-	63.89
aa+al+ay	-	-	58.67	70.09	-
aa+av+aw	-	82.50	-	-	50.00
aa+av+ay	-	84.28	-	68.30	-
aa+aw+ay	-	89.62	50.00	-	-
al+av+aw	63.39	-	-	-	50.00
al+av+ay	50.00	-	-	66.96	-
al+aw+ay	72.32	-	58.67	-	-
av+aw+ay	50.00	91.40	-	-	-
al+av+aw+ay	86.61	-	-	-	-
aa+av+aw+ay	-	86.06	-	-	-
aa+al+aw+ay	-	-	60.71	-	-
aa+al+av+ay	-	-	-	71.88	-
aa+al+av+aw	-	-	-	-	57.94

For the setup of our experiments, we follow the typical approach used in the previous DTMKL studies in literature. Specifically, we used four types of kernel: linear (i.e., $k(x_i, x_j) = x_i' x_j$), Gaussian (i.e., $k(x_i, x_j) = e^{-\gamma \|x_i - x_j\|^2}$), Laplacian (i.e., $(x_i, x_j) = e^{-\sqrt{\gamma}\|x_i - x_j\|}$), and inverse square distance (i.e., $k(x_i, x_j) = \frac{1}{\gamma \|x_i - x_j\|^2 + 1}$) kernels. In total, we have 4 kinds base kernels for all methods.

For the BCI Competition III IVa dataset, we established five dataset groups, each of which includes four source subjects as a source domain and one target subject as a target domain. When one subject is a target domain, it will no longer appear in the source domains. Thus, each target domain corresponds to 15 different source domains. Table 3 shows the classification accuracy for the BCI Competition III dataset, IVa. The first column of Table 3 shows the different source domains, and the first row shows the different target domains. The highest classification accuracy of the target domain aa was 85.76%, where the corresponding source domain was al; the highest classification accuracy of the target domain al was 96.43%, where the corresponding source domain was aw; the highest classification accuracy of the target domain av was 75.52%, where the corresponding source domain was al + ay; the highest classification accuracy of the target domain aw is 81.25%, where the corresponding source domain is al; and the highest classification accuracy of the target domain ay is 92.06%, where the corresponding source domain is aa + al.

TABLE 4. Classification accuracy of DTMKB on our own dataset I.

source \ target	D(%)	Z(%)	L(%)	X(%)	S(%)
D	-	71.33	50.00	74.44	70.00
Z	65.55	-	63.33	**82.22**	72.22
L	50.00	70.00	-	70.00	50.00
X	62.22	**81.11**	55.56	-	73.33
S	50.00	76.67	58.89	70.00	-
D+Z	-	-	55.56	77.78	70.00
D+L	-	75.55	-	72.22	50.00
D+X	-	74.44	50.00	-	73.33
D+S	-	74.44	-	74.44	-
Z+L	56.67	-	-	80.00	56.67
Z+X	**70.00**	-	61.11	-	**75.56**
Z+S	57.78	-	**72.22**	70.00	-
L+X	53.33	76.67	-	-	60.00
L+S	50.00	76.67	-	66.67	-
X+S	50.00	78.89	63.33	-	-
D+Z+L	-	-	-	81.11	60.00
D+Z+X	-	-	50.00	-	68.89
D+Z+S	-	-	56.67	73.33	-
D+L+X	-	73.44	-	-	50.00
D+L+S	-	71.11	-	74.44	-
D+X+S	-	75.55	50.00	-	-
Z+L+X	62.22	-	-	-	64.44
Z+L+S	50.00	-	-	67.78	-
Z+X+S	64.44	-	58.89	-	-
L+X+S	50.00	72.22	-	-	-
Z+L+X+S	63.33	-	-	-	-
D+L+X+S	-	73.33	-	-	-
D+Z+X+S	-	-	60.00	-	-
D+Z+L+S	-	-	-	72.22	-
D+Z+L+X	-	-	-	-	61.11

Table 4 shows the classification accuracy on our own dataset. The highest classification accuracy of the target domain D was 70.00%, where the corresponding source domain was Z+X; the highest classification accuracy of the target domain Z was 81.11%, where the corresponding source domain was X; the highest classification accuracy of the target domain L was 72.22%, where the corresponding source domain was Z+S; the highest classification accuracy of the target domain X is 82.22%, where the corresponding source

domain is Z; and the highest classification accuracy of the target domain S is 75.56%, where the corresponding source domain is Z+X.

Table 5 shows the classification accuracy on our own dataset I. The highest classification accuracy of the target domain D was 68.33%, where the corresponding source domain was Z+X; the highest classification accuracy of the target domain Z was 79.98%, where the corresponding source domain was X; the highest classification accuracy of the target domain L was 69.17%, where the corresponding source domain was Z+S; the highest classification accuracy of the target domain X is 81.67%, where the corresponding source domain is Z; and the highest classification accuracy of the target domain S is 74.17%, where the corresponding source domain is Z+X.

TABLE 5. Classification accuracy of DTMKB on our own dataset II.

source \ target	D(%)	Z(%)	L(%)	X(%)	S(%)
D	-	73.33	50.83	74.17	69.17
Z	65.83	-	63.33	**81.67**	72.50
L	50.00	71.67	-	70.00	55.00
X	61.67	**79.98**	55.00	-	73.33
S	50.00	77.50	60.83	67.75	-
D+Z	-	-	54.17	75.83	70.83
D+L	-	75.00	-	70.83	50.00
D+X	-	75.83	51.76	-	72.50
D+S	-	73.33	-	73.33	-
Z+L	59.27	-	-	77.50	57.50
Z+X	**68.33**	-	61.67	-	**74.17**
Z+S	57.50	-	**69.17**	68.33	-
L+X	52.50	73.33	-	-	60.00
L+S	50.00	74.17	-	64.17	-
X+S	50.00	76.67	60.83	-	-
D+Z+L	-	-	-	78.33	62.50
D+Z+X	-	-	50.00	-	67.75
D+Z+S	-	-	56.67	70.00	-
D+L+X	-	72.50	-	-	50.00
D+L+S	-	70.83	-	69.17	-
D+X+S	-	74.17	50.00	-	-
Z+L+X	62.50	-	-	-	63.33
Z+L+S	50.00	-	-	65.00	-
Z+X+S	63.33	-	58.33	-	-
L+X+S	55.83	71.67	-	-	-
Z+L+X+S	63.33	-	-	-	-
D+L+X+S	-	71.67	-	-	-
D+Z+X+S	-	-	60.83	-	-
D+Z+L+S	-	-	-	71.67	-
D+Z+L+X	-	-	-	-	59.17

Table 6 shows the classification accuracy on our own dataset III. The highest classification accuracy of the target domain D was 68.00%, where the corresponding source domain was Z+X; the highest classification accuracy of the target domain Z was 79.98%, where the corresponding source domain was X; the highest classification accuracy of the target domain L was 69.17%, where the corresponding source domain was Z+S; the highest classification accuracy of the target domain X is 81.67%, where the corresponding source domain is Z; and the highest classification accuracy of the target domain S is 74.17%, where the corresponding source domain is Z+X.

TABLE 6. Classification accuracy of DTMKB on our own dataset III.

source \ target	D(%)	Z(%)	L(%)	X(%)	S(%)
D	-	71.33	50.00	72.67	67.33
Z	65.33	-	60.67	**80.67**	72.00
L	50.00	70.00	-	70.00	50.00
X	60.67	**80.00**	52.00	-	70.67
S	50.00	76.67	56.67	66.67	-
D+Z	-	-	51.33	74.67	68.67
D+L	-	74.00	-	69.33	50.00
D+X	-	72.00	50.00	-	71.33
D+S	-	71.33	-	70.00	-
Z+L	56.67	-	-	77.33	55.33
Z+X	**68.00**	-	58.00	-	**73.33**
Z+S	56.00	-	**68.67**	66.67	-
L+X	52.67	73.33	-	-	60.00
L+S	50.00	74.00	-	62.67	-
X+S	50.00	76.67	60.00	-	-
D+Z+L	-	-	-	77.33	58.67
D+Z+X	-	-	50.00	-	67.33
D+Z+S	-	-	52.00	69.33	-
D+L+X	-	72.00	-	-	50.00
D+L+S	-	68.67	-	69.33	-
D+X+S	-	73.33	50.00	-	-
Z+L+X	60.67	-	-	-	62.00
Z+L+S	50.00	-	-	64.00	-
Z+X+S	61.33	-	55.33	-	-
L+X+S	50.00	69.33	-	-	-
Z+L+X+S	61.33	-	-	-	-
D+L+X+S	-	69.33	-	-	-
D+Z+X+S	-	-	56.67	-	-
D+Z+L+S	-	-	-	70.67	-
D+Z+L+X	-	-	-	-	60.67

TABLE 7. Classification accuracy of DTMKB on all datasets.

subject \ method		DTMKB(%)	MKB(%)	DTMKL(%)	SVM(%)
BCIC III IVa	aa	**91.96**	87.50	91.07	82.14
	al	**96.43**	**96.43**	94.64	92.85
	av	**75.52**	68.75	71.94	67.35
	aw	81.25	76.79	**82.59**	74.55
	ay	**92.86**	82.14	88.89	85.83
	mean	**87.60**	82.32	85.83	79.33
Our own dataset I	D	**70.00**	67.78	67.78	65.56
	Z	**81.11**	78.89	80.00	75.56
	L	**72.22**	70.00	70.00	67.28
	X	81.11	**82.22**	81.11	78.89
	S	**75.56**	73.33	72.22	70.00
	mean	**76.00**	74.44	74.22	71.56
Our own dataset II	D	**68.33**	66.67	66.67	62.50
	Z	**79.98**	75.00	76.67	71.67
	L	**69.17**	65.83	65.00	63.33
	X	81.67	81.67	**83.33**	78.33
	S	**74.17**	69.17	72.50	67.75
	mean	**74.66**	71.67	72.83	68.72
Our own dataset III	D	**68.00**	63.33	64.00	60.00
	Z	**80.00**	74.00	75.33	71.33
	L	**68.67**	66.00	66.00	63.33
	X	**80.67**	74.67	76.67	72.00
	S	**73.33**	69.33	71.33	66.67
	mean	**74.13**	69.47	70.67	66.67

Table 7 lists the classification precisions of three comparison approaches (multiple kernel learning (MKL), DTMKL, and SVM) and DTMKB for the above datasets. The performance achieved by DTMKB is significantly better than

that achieved by the three comparison approaches. Several observations can be made from these results.

For the IVa dataset, the number of training trials for aa and al is greater than the number of test trials, and the number of training trials for aw and ay is smaller than the number of test trials. The classification performance of DTMKB for aa and al was 91.96% and 96.43%, respectively. The former performance was higher than that of the three comparison approaches, and the latter achieved the same performance as MKL, with the best performance among all methods. For av, and ay, DTMKB achieved results of 75.52% and 92.86% respectively, which are better than the performance of the other three comparison approaches. DTMKL achieved the best performance for subject av at 82.59%, although DTMKB achieved a performance very close to this at 81.25%. Moreover, DTMKB achieved an average classification accuracy for these datasets of 87.60%, providing a significant performance improvement over the other competitive approaches.

For our own dataset I, the number of training trials was more than the number of test trials for all subjects. For subject X, DTMKB achieved the best classification results for the four subjects, D, Z, L, and S, with performances of 70.00%, 81.11%, 72.22%, and 75.56%, respectively. For subject X, DTMKB produced a result of 81.11%, which was less than the 83.33% result of the MKB method. In addition, DTMKB achieved an average classification accuracy of 76.00% for our own dataset I, with significant performance improvements over the competing methods.

For our own dataset II, the number of training trials was less than the number of test trials for all subjects. For subject X, DTMKB achieved the best classification results for the four subjects, D, Z, L, and S, with performances of 68.33%, 79.98%, 69.17%, and 74.17%, respectively. For subject X, DTMKB produced a result of 81.67%, which was less than the 83.33% result of the DTMKL method. In addition, DTMKB achieved an average classification accuracy of 74.66% for our own dataset, which was the best performance among the competing methods.

For our own dataset III, the number of training trials was less than the number of test trials for all subjects. For subject X, DTMKB achieved the best classification results for all five subjects, D, Z, L, X and S, with performances of 68.00%, 80.00%, 68.67%, 80.67% and 73.33%, respectively. Obviously, DTMKB achieved a superior average classification accuracy of 74.66% compared to the three other approaches.

As shown in Table 7 and the above description, the smaller the training dataset, the higher the average classification accuracy of the DTMKB method compared to other methods. This proves that the DTMKB method achieves good results on small samples.

B. DISCUSSIONS

The most significant parameter in the boosting strategy is the number of boost rounds, (i.e., the number of iterations). The optimal performance of the boosting model can be achieved by carefully choosing the best parameters. To test the impact of the parameters, we conducted a series of experiments to evaluate their impact on accuracy and efficiency performance in the classification task.

In our experiments, to evaluate the influence of the total number of boosting trials for the proposed DTMKB algorithm, we examine the experimental results by varying the boosting parameter T from 1 to 50. Figure 5(a) and Figure 5(b) show the influence of the number of boost parameter T on the classification accuracy and learning time cost respectively for BCIC III IVa dataset. In addition, Figure 6(a), Figure 6(b) and Figure 6(c) show the evaluation result for the impact of the boosting parameter T on the classification accuracy for our own dataset I, II and III. Figure 7(a), Figure 7(b) and Figure 7(c) show the evaluation result for the impact of the boosting parameter T on the learning time cost for our own dataset I, II and III.

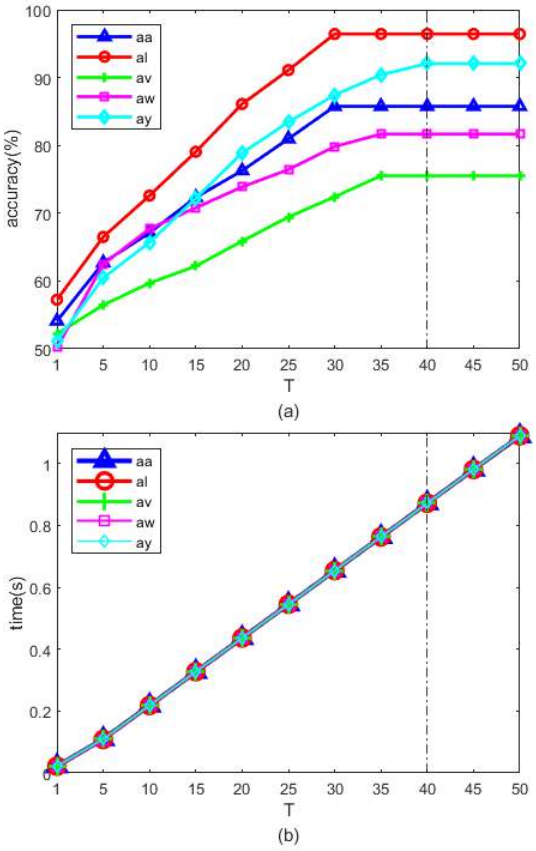

FIGURE 5. Evaluation of DTMKB classification accuracy and learning time cost with respect to boosting parameter T respectively. (a) is for classification accuracy on BCIC III IVa dataset; (b) is for learning time cost on BCIC III IVa dataset.

It was observed that, in terms of the classification accuracy, increasing the total value of T generally improves the accuracy for all datasets. On the other hand, we found that increasing the value of T results in a linear increase in the time cost required for all dataset. And when it comes to different datasets, the time cost is almost the same. This is not surprising because the DTMKB algorithm have linear

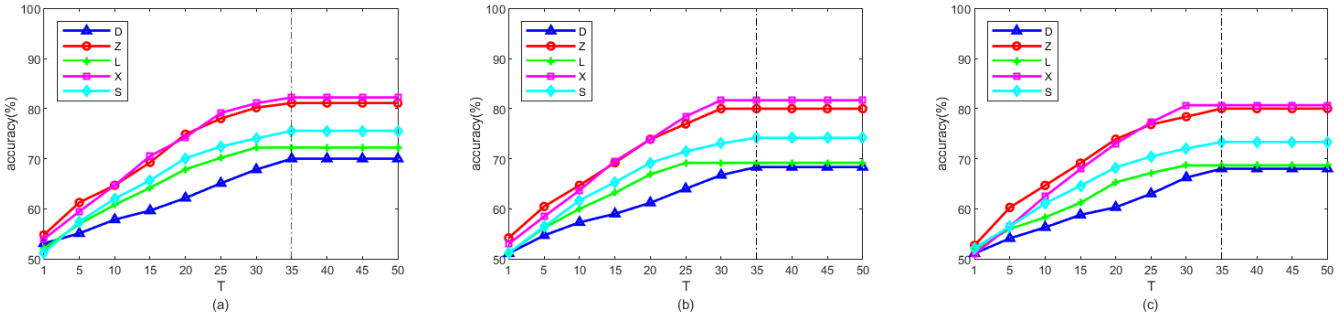

FIGURE 6. Evaluation of DTMKB classification accuracy with respect to boosting parameter T. (a) is for classification accuracy on our own dataset I; (b) is for classification accuracy on our own dataset II; (c) is for classification accuracy on our own dataset III.

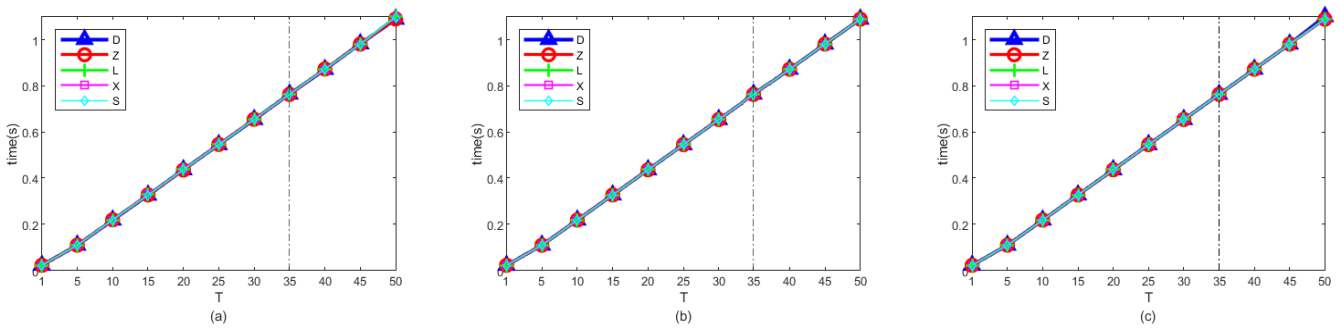

FIGURE 7. Evaluation of DTMKB learning time cost with respect to boosting parameter T. (a) is for learning time cost on on our own dataset I; (b) is for learning time cost on our own dataset II; (c) is for learning time cost on on our own dataset III.

time complexity with respect to the total number of boosting trials T.

In addition to the impact on the accuracy performance, the parameter T also affects the time efficiency of the DTMKB algorithm. To avoid a waste in the learning time cost, T should be as small as possible. Because the datasets have different sizes and features, the number of iterations during the learning process may be different. The DTMKB model was set using $T = 40$ for the BCIC III Iva dataset, and $T = 35$ for our own dataset I, II and III.

As shown in Figure 5(b) and Figure 7, DTMKB algorithm have a linear time complexity with respect to the total number of the parameter T. In particular, the base learning time cost of the DTMKB algorithm the underlying algorithms used in training kernel based classifiers. We adopt SVM for training the base kernel classifiers, the SVM training complexity is $O(n^{2.3})$. So the training cost for our proposed DTMKB is $T \times O(n^{2.3})$.

V. CONCLUSION

In this paper, we employed a novel framework of DTMKB for classification with the domain transfer of multiple kernels. DTMKL learns the kernel function and target classifier simultaneously by minimizing the structural risk function and the distribution mismatch between the samples from the source and target domains. The DTMKB technique is simple and can easily be used to learn effective classification models with the domain transfer of multiple kernels by taking advantage of efficient boosting techniques.

The results of this study show that the average classification accuracies for the BCI Competition III dataset, IVa, and our own dataset are 87.60% and 74.66% respectively. The above approach yields a superior average classification accuracy compared to the three other approaches applied. When the number of training trials was less than the number of test trials, DTMKL clearly showed a better classification performance. In conclusion, DTMKB is a novel and promising framework for classification on small samples of MI tasks.

ACKNOWLEDGMENT

The authors would like to thank the laboratory of brain-computer interfaces at the Graz University of Technology for providing their data.

REFERENCES

[1] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain computer interfaces, a review," *Sensors*, vol. 12, no. 2, pp. 1211–1279, Jan. 2012.

[2] J. J. Shih, D. J. Krusienski, and J. R. Wolpaw, "Brain-computer interfaces in medicine," *Mayo Clinic Proc.*, vol. 87, no. 3, pp. 268–279, Mar. 2012.

[3] G. Pfurtscheller and F. L. H. Da Silva, "Event-related EEG/MEG synchronization and desynchronization: Basic principles," *Clin. Neurophysiol.* vol. 110, no. 11, pp. 57–1842, Nov. 1999.

[4] S. Lemm, C. Schäfer, and G. Curio, "BCI Competition 2003–Data set III: Probabilistic modeling of sensorimotor μ rhythms for classification of imaginary hand movements," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1077–1080, Jun. 2004.

[5] Y.-H. Yu *et al.*, "An inflatable and wearable wireless system for making 32-channel electroencephalogram measurements," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 7, pp. 806–813, Jul. 2016.
[6] Q. Liang, X. Cheng, S. C. Huang, and D. Chen, "Opportunistic sensing in wireless sensor networks: Theory and application," *IEEE Trans. Comput.*, vol. 63, no. 8, pp. 2002–2010, Aug. 2014.
[7] X. Liu, M. Jia, Z. Na, W. Lu, and F. Li, "Multi-modal cooperative spectrum sensing based on dempster-shafer fusion in 5G-based cognitive radio," *IEEE Access*, vol. 6, pp. 199–208, 2018.
[8] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[9] F. Lotte *et al.*, "A review of classification algorithms for EEG-based brain-computer interfaces: A 10 year update," *J. Neural Eng.*, vol. 15, no. 3, Apr. 2018, Art. no. 031005.
[10] 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.
[11] F. Lotte and C. Guan, "Regularizing common spatial patterns to improve BCI designs: Unified theory and new algorithms," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 2, pp. 62–355, Feb. 2011.
[12] S.-H. Park, D. Lee, and S.-G. Lee, "Filter bank regularized common spatial pattern ensemble for small sample motor imagery classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 2, pp. 498–505, Feb. 2018.
[13] M. Alamgir, M. Grosse-Wentrup, and Y. Altun, "Multitask learning for brain-computer interfaces," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2010, pp. 17–24.
[14] M. Arvaneh, C. Guan, K. K. Ang, and C. Quek, "EEG data space adaptation to reduce intersession nonstationarity in brain-computer interface," *Neural Comput.*, vol. 25, pp. 2146–2171, Aug. 2013.
[15] H. Cho, M. Ahn, K. Kim, and S. C. Jun, "Increasing session-to-session transfer in a brain-computer interface with on-site background noise acquisition," *J. Neural Eng.*, vol. 12, no. 6, Dec. 2015, Art. no. 066009.
[16] 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.
[17] H. Ramoser, J. Müller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial EEG during imagined hand movement," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 8, no. 4, pp. 441–446, Dec. 2000.
[18] S. Fazli, F. Popescu, M. Danóczy, B. Blankertz, K. R. Müller, and C. Grozea, "Subject-independent mental state classification in single trials," *Neural Netw.*, vol. 22, no. 9, pp. 12–1305, Nov. 2009.
[19] W. Samek, F. C. Meinecke, and K.-R. Müller, "Transferring subspaces between subjects in brain–computer interfacing," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 8, pp. 2289–2298, Aug. 2013.
[20] Y. Jin, M. Mousavi, and V. R. de Sa, "Adaptive CSP with subspace alignment for subject-to-subject transfer in motor imagery brain-computer interfaces," in *Proc. 6th Int. Conf. Brain-Comput. Interface (BCI)*, Jan. 2018, pp. 1–4.
[21] R. Zhang *et al.*, "A new motor imagery EEG classification method FB-TRCSP+RF Based on CSP and random forest," *IEEE Access*, vol. 6, pp. 44944–44950, 2018.
[22] M. Völker *et al.*, "Deep transfer learning for error decoding from non-invasive EEG," in *Proc. 6th Int. Conf. Brain-Comput. Interface (BCI)*, Jan. 2018, pp. 1–6.
[23] M. Dai, Z. Zheng, S. Liu, and P. Zhang, "Transfer kernel common spatial patterns for motor imagery brain-computer interface classification," *Comput. Math. Methods Med.*, vol. 2018, Mar. 2018, Art. no. 9871603.
[24] L. Duan, I. W. Tsang, and D. Xu, "Domain transfer multiple kernel learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 465–479, Mar. 2012.
[25] W. Tu and S. Sun, "A subject transfer framework for EEG classification," *Neurocomputing*, vol. 82, pp. 109–116, Apr. 2012.
[26] K. M. Borgwardt, A. Gretton, M. J. Rasch, H.-P. Kriegel, B. Schölkopf, and A. J. Smola, "Integrating structured biological data by kernel maximum mean discrepancy," *Bioinformatics*, vol. 22, no. 14, pp. e49–e57, Jul. 2006.
[27] J. Huang, A. J. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf, "Correcting sample selection bias by unlabeled data," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 19, pp. 601–608, Dec. 2007.
[28] S. J. Pan, J. T. Kwok, and Q. Yang, "Transfer learning via dimensionality reduction," in *Proc. 33rd Assoc. Adv. Artif. Intell.*, Jul. 2008, pp. 677–682.
[29] A. Rakotomamonjy, F. R. Bach, S. Canu, and Y. Grandvalet, "SimpleMKL," *J. Mach. Learn. Res.*, vol. 9, pp. 2491–2521, Nov. 2008.
[30] L. Duan, I. W. Tsang, D. Xu, and S. J. Maybank, "Domain transfer SVM for video concept detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1375–1381.
[31] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997.
[32] B. Blankertz *et al.*, "The BCI competition. III: Validating alternative approaches to actual BCI problems," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 14, no. 2, pp. 153–159, Jun. 2006.
[33] Electrode Position Nomenclature Committee, "Guideline thirteen: Guidelines for standard electrode position nomenclature," *J. Clin. Neurophysiol.*, vol. 11, pp. 111–113, Jan. 1994.
[34] R. Raina, B. Alexis, H. Lee, B. Packer, and A. Y. Ng, "Self-taught learning: Transfer learning from unlabeled data," in *Proc. 24th Int. Conf. Mach. Learn. (ICML)*, Jun. 2007, pp. 759–766.

MENGXI DAI received the B.E. degree from the Hebei University of Technology, in 2011, and the M.S. degree from the Xi'an University of Technology, in 2014. He is currently pursuing the Ph.D. degree with the School of Instrumentation and Optoelectronic Engineering, Beihang University. His research interests include brain–computer interface, biomedical signal processing, and machine learning.

SHUAI WANG, photograph and biography not available at the time of publication.

DEZHI ZHENG, photograph and biography not available at the time of publication.

RUI NA, photograph and biography not available at the time of publication.

SHUAILEI ZHANG, photograph and biography not available at the time of publication.