Multi-Source Decentralized Transfer for Privacy-Preserving BCIs

Wen Zhang*, Ziwei Wang, and Dongrui Wu*, Senior Member, IEEE

Abstract—Transfer learning, which utilizes labeled source domains to facilitate the learning in a target model, is effective in alleviating high intra- and inter-subject variations in electroencephalogram (EEG) based brain-computer interfaces (BCIs). Existing transfer learning approaches usually use the source subjects’ EEG data directly, leading to privacy concerns. This paper considers a decentralized privacy-preserving transfer learning scenario: there are multiple source subjects, whose data and computations are kept local, and only the parameters or predictions of their pre-trained models can be accessed for privacy-protection; then, how to perform effective cross-subject transfer for a new subject with unlabeled EEG trials? We propose an offline unsupervised multi-source decentralized transfer (MSDT) approach, which first generates a pre-trained model from each source subject, and then performs decentralized transfer using the source model parameters (in gray-box settings) or predictions (in black-box settings). Experiments on two datasets from two BCI paradigms, motor imagery and affective BCI, demonstrated that MSDT outperformed several existing approaches, which do not consider privacy-protection at all. In other words, MSDT achieved both high privacy-protection and better classification performance.

Index Terms—Brain-computer interfaces, transfer learning, privacy protection, deep learning.

I. INTRODUCTION

A BRAIN-COMPUTER interface (BCI) [1], [2] enables direct communications between a user’s brain and external devices, by analyzing neural activities of the brain, typically recorded with multi-channel electroencephalogram (EEG) signals.

BCIs can be categorized into active and passive ones according to whether the brain signals are generated intentionally or unintentionally [3]. A representative active BCI paradigm is motor imagery (MI) [4], which usually seeks to recognize a user’s motor intention, e.g., hand, feet, or tongue movements. A representative negative BCI paradigm is affective BCI (aBCI) [5], which usually aims to identify the emotion states, e.g., positive, neutral, and negative, induced by videos, audio, images, etc. These two paradigms have been widely used in human-machine interaction, health management, neural rehabilitation, etc., and are the focus of this paper.

The conventional pipeline for processing EEG signals in MI or aBCI consists of signal preprocessing, feature extraction, and model training. The latter two can also be integrated into a single neural network. Signal preprocessing and model training procedures in MI and aBCI are similar. Since discriminative information in MI is mainly spatial, commonly extracted features include log-variance of the spatially filtered EEGs, e.g., common spatial pattern (CSP) [6], [7], and tangent space features from the covariance matrices of EEG trials [8]. For aBCI, commonly used features include time domain, frequency domain [9], and entropy features [10].

A major challenge of BCIs is that EEG signals are non-stationary, with high variations across sessions, subjects, tasks, and devices [11]. Transfer learning (TL) [11] is a promising approach to alleviate this problem. Various TL approaches have been proposed for BCI in the last decade, e.g., adaptive CSP [12], data alignment [13], [14], instance-based TL [15], [16], feature-based TL [17], [18], and deep TL [19]. For aBCIs, existing TL approaches mainly include feature-based TL [20] and adversarial-based deep TL [21], [22].

However, most existing TL approaches require access to the source EEG data or features [17], [18], [19], [20]. EEG signals contain rich private information [23], [24], e.g., health status, emotion, psychological state, personal identity, etc. Consequently, the source EEG data may not be directly shared due to regulations such as the European General Data Protection Regulation1 (GDPR), China Personal Information Protection Law, or user privacy concerns. For example, the raw EEG signals or extracted features can be used to perform personal identification or authentication [25]. Additionally, due to high data collection costs, EEG data are usually organized distributedly in small datasets from different groups, instead of a large centralized dataset like ImageNet [26] in computer vision.

To protect data privacy in TL, we consider a decentralized scenario: all data and computations of each source subject are kept local, and only the parameters or predictions of the pre-trained source models are accessible to the new subject.

1https://gdpr-info.eu/
A. Feature Extraction for MI and aBCI

For MI, the discriminative information is mainly spatial, which can be obtained from the covariance matrices of the EEG trials. These covariance matrices are symmetric positive definite (SPD) and lie on a Riemannian manifold [8]. Tangent space mapping can map a Riemannian space SPD matrix $P_t \in \mathbb{R}^{c \times c}$, where $c$ is the number of EEG channels, to a Euclidean tangent space vector $x_t$ around an SPD matrix $M$, which is usually the Riemannian or Euclidean mean:

$$x_t = \text{upper}(\log_M(M_{\text{ref}}P_tM_{\text{ref}})),$$

where upper takes the upper triangular part of a $c \times c$ SPD matrix and forms a vector $x_t \in \mathbb{R}^{1 \times (c+1)/2}$, and $M_{\text{ref}}$ is a reference matrix, which can be the Riemannian mean of all covariance matrices. This tangent space vector contains discriminative information on MI, and hence is frequently used as features in classification.

For aBCI, differential entropy (DE) [10] is a commonly used feature:

$$\text{DE} = -\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma^2} e^{\frac{(x-\mu)^2}{2\sigma^2}} \log\left(\frac{1}{\sqrt{2\pi}\sigma^2} e^{\frac{(x-\mu)^2}{2\sigma^2}}\right) dx$$

$$= \frac{1}{2} \log 2\pi e \sigma^2,$$

where $x$ is the time series, and $\sigma$ is the variance of $x$. DE is a simple measure of the complexity of a time series in a certain frequency band, which is equivalent to the logarithm energy spectrum. For each trial, we compute DE in different frequency bands.

B. Privacy-Preserving Machine Learning

Privacy-preserving machine learning [27] addresses the increasing privacy concerns in real-world applications, and can be realized by federated learning [28], differential privacy [29], etc. However, several challenges hinder the broad adoptions of these techniques in BCIs, e.g., technology immaturity, performance degradation, and deployment difficulties.

Recently, there is a trend in the computer vision community to use data-free TL to balance the cross-domain learning performance and source privacy protection. In data-free TL, knowledge can be transferred from the source model parameters or predictions. Various such approaches have been proposed, e.g., deep hypothesis transfer [30, 31], virtual domain construction [32], knowledge distillation [33], etc.

In BCIs, the privacy of each source subject should be individually addressed, requiring all data and computations to be kept local. This paper considers MSDT, which only accesses the parameters or predictions of the pre-trained source models, to protect source privacy and improve the target learning performance.

C. Transfer Learning for MI and aBCI

TL in BCIs usually includes multiple source subjects. Multiple TL approaches have been proposed [11], which can be categorized into source combination ones and multi-source ones.

Source combination TL approaches include traditional TL and deep TL:
**Traditional TL:** It includes data alignment and shallow approaches. Data alignment seeks to mitigate the marginal distribution discrepancies of different domains by aligning the EEG covariance matrices [13], [14], [34]. The shallow approaches accomplish knowledge transfer through features, instances, etc. Zhang and Wu [17] proposed a manifold embedded knowledge transfer approach for MI, which combines data alignment and feature adaptation in the tangent space. Xu et al. [15] proposed an instance-based selective TL approach for MI in the Riemannian tangent space, which utilizes labeled samples from the source and target subjects.

**Deep TL:** It has also been well studied in MI and aBCI. The main idea is to minimize the gap between different domains, represented by discrepancy-based losses [19], [35] or adversarial losses [21], [22], [36]. Zhang et al. [37] applied fine-tuning to adapt a pre-trained source model on different subsets of model parameters. Tang et al. [36] proposed a conditional domain adaptation framework, which utilizes a conditional domain discriminator and a label classifier to learn common features across subjects. Xu et al. [38] demonstrated that data alignment combined with adaptive batch normalization can improve the generalization ability of deep models. Chen et al. [39] applied distribution adaptation to multi-view features to obtain more generalizable representations.

The multi-source TL strategy considers each source domain separately and is more practical and challenging. For MI, Jeon et al. [40] proposed a source subject selection strategy to remove dissimilar subjects; Azab et al. [41] proposed a parameter-based TL approach, which regularizes the target model using parameters of the source models; and, Ju et al. [42] adopted federated averaging to reduce the discrepancies among different subjects, based on the EEG spatial covariance matrices. For aBCI, Li et al. [20] proposed a multi-source TL framework in aBCI by assuming there are some labeled calibration instances from the new subject; Gu et al. [43] recently combined joint subspace learning with dictionary learning to learn a common subspace between subjects; and, Chen et al. [44] applied multiple feature space adaptation network (MFSAN) [45] to aBCI, by simultaneously aligning multiple specific feature spaces and the classifier outputs.

In summary, most above approaches are offline unsupervised ones, and require private source domain data, except [41] and [42]. The parameter-based TL approach in [41] needs calibration trials from the target subject, which limits its application. The federated averaging strategy in [42] incurs high information transmission cost, and is only applicable to MI.

### III. MSDT

Unlike existing multi-source TL approaches in BCIs, our proposed MSDT only needs parameters or predictions from each pre-trained source model. For the target subject, MSDT only requires some unlabeled trials, rather than labeled trials in many other approaches.

Assume there are $M$ source subjects $S = \{S_m\}_{m=1}^M$. For each source subject, we have $n_s$ labeled EEG trials $S_m = \{(X_{s,i}, y_{s,i})\}_{i=1}^{n_s}$, where $X_{s,i} \in \mathbb{R}^{c \times t}$ is the $i$-th feature matrix, and $y_{s,i} \in \{1, \ldots, K\}$ the corresponding label, in which $c$, $t$ and $k$ denote the number of channels, time domain samples, and classes, respectively. Assume also the target domain has $n_t$ unlabeled EEG trials $T_u = \{(X_{t,i}, y_{t,i})\}_{i=1}^{n_t}$, where $X_{t,i} \in \mathbb{R}^{c \times t}$. The data of each source domain are only accessible to train a local model. The goal of MSDT is to learn a good target model $\theta_t$ using some unlabeled target domain data $T_u$ and pre-trained source models $\{\theta_m\}_{m=1}^M$.

MSDT consists of two steps:

1. **Source model pre-training**, which contains three sub-steps: data augmentation (optional), feature extraction, and deep model generation.
2. **Decentralized transfer**, where there are some unlabeled target data $T_u$ and a set of pre-trained source models $\{\theta_m\}_{m=1}^M$. The parameters of the source models are available in gray-box settings, and the source models are only available as application programming interfaces (APIs) in black-box settings.

#### A. Source Model Pre-Training

Source model pre-training contains three sub-steps: data augmentation (optional), feature extraction, and deep model generation.

1. **EEG Data Augmentation:** In MI, the number of trials per subject is usually small, which may be insufficient to train a neural network. Data augmentation [46] on the raw EEG data can be used to alleviate this issue. The data augmentation strategies shown in Table I were used for MI classification, where $C_{\text{noise}}$, $C_{\text{mult}}$ and $C_{\text{freq}}$ are three hyper-parameters set as 2, 0.05 and 0.2, respectively, according to [47].

| Strategy       | Formula                                                                 |
|----------------|-------------------------------------------------------------------------|
| Noise Injection| $X_{s,i} + \text{rand} \times \text{std}(X_{s,i})/C_{\text{noise}}$    |
| Data Flipping  | $\max(X_{s,i} - X_{s,i})$                                              |
| Data Scaling   | $X_{s,i} \times (1 \pm C_{\text{scale}})$                             |
| Frequency Shift| $F_{\text{shift}}(X_{s,i}, \pm C_{\text{freq}})$                       |

More specifically, noise injection adds uniform noise to each EEG trial, data flipping flips the EEG signal of each channel, data scaling multiplies the original EEG signal by a coefficient close to 1, and frequency shift uses the Hilbert transform [48] to shift the frequency of the EEG signal. Some example EEG trials before and after augmentation are shown in Fig. 2.

2. **Feature Extraction:** After data augmentation, we extract commonly used features from EEG signals [11], [17], such as tangent space vectors for MI, and differential entropy features for aBCI (Section II-A). The extracted features of each subject are denoted as $X$ in this paper.
3) Deep Model Generation: With the extracted \( \{X_s, Y_s\} \) from each source subject, we further use a deep neural network (DNN) to transform them into low-dimensional features. For model generation, we employ two linear fully connected layers, each followed by layer normalization [49], as the backbone feature extractor \( g_m \), and a linear fully connected layer with weight normalization [30] as the classifier \( f_m \) (the detailed architecture is introduced in Section IV-B). Thus, the complete source model is \( \theta_m = (g_m \circ f_m) \). The source training loss is:

\[
L_{ce} = \mathbb{E}_{(x_i, y_i) \in S_m} \ell_{CE}(\theta_m(x_i), \tilde{y}_i),
\]

where \( \ell_{CE} \) is the cross-entropy loss, and \( \tilde{y}_i = (1 - \alpha) y_i + \alpha / K \) is the smoothed label after label smoothing [50], in which \( \alpha \) is a hyper-parameter empirically set to 0.1, and \( K \) the number of classes.

Note that all above procedures are executed locally in each source domain, avoiding the risk of privacy leakage.

B. Gray-Box MSDT

As shown in Fig. 3, in gray-box decentralized transfer, we have a set of pre-trained source models \( \{\theta_m\}_{m=1}^{M} \) and some unlabeled target data \( T_n \). For each source model, we construct a corresponding target model \( \theta'_m = (g_m \circ f_m) \) with the same architecture as \( \theta_m \), where the feature extractor \( g_m \) is initialized from \( g_m \) of the source model, and the classifier \( f_m \) is forced to be the same as \( f_m \) of the source model to ensure hypothesis transfer [30]. The final target model is a weighted ensemble of the adapted models

\[
\theta_t(X_t) = \sum_{m=1}^{M} \alpha_m \theta'_m(X_t),
\]

where \( \alpha_m \in (0, 1) \) is the weight of the \( m \)-th target model adapted from the corresponding source model, and \( \sum_{m=1}^{M} \alpha_m = 1 \). Intuitively, \( \alpha_m \) represents the \( m \)-th source model’s transferability.

Details of the training losses and source transferability estimation are introduced next.

1) Information Maximization: The ideal predictions should be similar to one-hot coding and not fall into only one class, which can be facilitated by information maximization [30], [51]. For the adaptation of each \( g'_m \), we minimize the following negative mutual information of predictions from model \( \theta'_m \):

\[
-I(x_t, \hat{Y}_t) = H(\hat{Y}_t | X_t) - H(\hat{Y}_t) = \mathbb{E}_{x_i \in X_t} h(\theta'_m(x_i)) - h(\mathbb{E}_{x_i \in X_t} \theta'_m(x_i)),
\]

where \( h(p) = -\sum p_i \log p_i \) is the information entropy on the predicted probabilities \( p \). Minimizing the conditional entropy \( H(\hat{Y}_t | X_t) \) and maximizing the marginal entropy \( H(\hat{Y}_t) \) make the target predictions more class-balanced and diverse, respectively. When there are multiple source models to adapt, we minimize the sum of the negative mutual information of all adapted source models, i.e.,

\[
L_{im} = \frac{1}{M} \sum_{m=1}^{M} \left[ \mathbb{E}_{x_i \in X_t} h(\theta'_m(x_i)) \right] + \sum_{k=1}^{K} \tilde{p}_k \log \tilde{p}_k,
\]

where \( \tilde{p} = \mathbb{E}_{x_i \in T_n} [\sigma (\theta_m(x_i))] \) is the average class probability in \( T_n \), and \( \sigma \) is the softmax function.

By minimizing the above loss, predictions of the adapted models are forced to resemble one-hot encoding.

2) Source Consistency Regularization: If a target sample \( x_{s, i} \) is well adapted to multiple source models, then the conditional probabilities \( p(y | x_{s, i}) \) predicted by different source models should be similar. More specifically, for a particular target sample \( x_{s, i} \), we define the source consistency loss as:

\[
L_{sc} = \frac{1}{KM} \sum_{k=1}^{K} \left[ \sum_{m=1}^{M} \theta'_m(x_{s, i}) - \frac{1}{M} \sum_{m'=1}^{M} \theta'_m(x_{s, i}) \right]^2,
\]
where $\theta'_m(x_{i,j})_k$ is the probability of $x_{i,j}$ belonging to the $k$-th class, predicted by $\theta'_m$, and both $m$ and $m'$ are the model indices.

The training process minimizes the class variance of predictions from different $\theta_m$ so that different adapted models $\theta'_m$ make more consistent predictions.

3) Source Transferability Estimation: When there are multiple source subjects, the source transferability determines the contribution of each source domain. Subjects with low transferability usually have low correlations with the new subject, and direct transfer from them may result in negative transfer [52].

Assume there are $M$ pre-trained source models $\{\theta_m\}_{m=1}^M$. We compute the mutual information of the predictions from each adapted source model to estimate its transferability, which is used as the weight of the model $\theta'_m$:

$$
\alpha_m = \frac{I \left( X_i, \theta'_m(X_i) \right)}{\sum_{m=1}^M I \left( X_i, \theta'_m(X_i) \right)},
$$

where $I(\cdot)$ is the information entropy in (5). The higher the information entropy is, the closer the predictions are to the true labels. Since $\theta'_m$ is optimized in the training process, the learned weights are dynamically adjusted during the model updating process.

With this strategy, we can assign each source subject a weight $\alpha_m$, according to his/her similarity to the target subject.

4) Mixup Regularization: Mixup [53] is a popular approach in data augmentation to improve the generalization ability of neural networks [54]. Its main idea is to generate random linear interpolations of the original data and labels by $\text{Mix}_x(x, y) = \lambda x + (1 - \lambda) y$, where $x$ and $y$ are features or labels of two random instances, and $\lambda$ is a constant parameter sampled from a Beta distribution [53] ($\beta(0.3, 0.3)$ was used in this paper).

To further increase the target model generalization, we employ the mixup strategy on $x_{i,j}$ and aggregate predictions $\theta_i(x_{i,j})$ of the target domain, then minimize their cross-entropy loss:

$$
\mathcal{L}_{\text{mix}}(\theta_i, X_t) = \mathbb{E}_{X_{t,i}, x_{i,j} \in X_t} \mathbb{E}_{\text{Mix}_x(X_{t,i}, X_{t,j})} \left[ -\sum_{m=1}^M \alpha_m \theta_m(y_{i,j}) \log(g_t(x_{i,j})) \right],
$$

where $\theta_i(x_{i,j})$ is $\sum_{m=1}^M \alpha_m \theta'_m(x_{i,j})$. By optimizing the mixup loss, the target model is forced to generalize well on the interpolated samples.

5) The Final Loss Function: Putting everything together, the final loss function is

$$
\mathcal{L}_{\text{all}} = \mathcal{L}_{\text{im}} + \beta \mathcal{L}_{\text{sc}} + \gamma \mathcal{L}_{\text{mix}}.
$$

$\beta = 0.1$ and $\gamma = 1$ were used in our experiments.

C. Black-Box MSDT

The aforementioned gray-box MSDT approach requires each source model to have its parameters available to the new subject. A more secure and privacy-preserving setting is black-box transfer, where the source domain models are only available as black-box APIs for querying. This subsection extends the gray-box MSDT approach to black-box MSDT using knowledge distillation [55], which is frequently used to transfer knowledge from a pre-trained model.

Our goal is to learn a single student model $\theta_i$ by querying multiple source APIs. This is different from MSDT-G, in which we learn a set of target adapted models $\{\theta'_m\}_{m=1}^M$. Additionally, MSDT-G requires to know the parameters of each source model, whereas MSDT-B only needs to query the source models for their outputs (the model parameters are not required).

We randomly initialize the student model $\theta_i = (g_t \circ f_t)$, and replace the source consistency regularization by the following unsupervised knowledge distillation loss between the weighted queries and the student model predictions,

$$
\mathcal{L}_{kd} = \mathbb{E}_{X_t \in X_t} \ell_{\text{KL}} \left( \sum_{m=1}^M \alpha_m \theta_m(x_t) \| \theta_i(x_t) \right),
$$

where $\ell_{\text{KL}}$ is the Kullback-Leibler (KL) divergence [56] loss, $\theta_m$ is a black-box model API, and $\alpha_m$ is the source transferability estimated by (11) on the source API predictions of the target data. By minimizing the knowledge distillation loss, the target model can learn the responses of the source APIs.

Note that, since there is only one student model, the information maximization term becomes:

$$
\mathcal{L}_{im} = \mathbb{E}_{X_t \in X_t} h(\theta_i(x_t)) + \sum_{k=1}^K \tilde{p}_k \log \tilde{p}_k.
$$

The overall loss function for black-box MSDT is

$$
\mathcal{L}_{\text{all}} = \mathcal{L}_{kd} + \mathcal{L}_{im} + \mathcal{L}_{\text{mix}}.
$$

The pseudocode of MSDT in gray-box and black-box settings is summarized in Algorithm 1.

IV. EXPERIMENTS

In this section, we conducted extensive cross-subject classification experiments in MI and aBCI paradigms to evaluate the effectiveness of MSDT in both gray-box and black-box settings.
were three classes of emotions (positive, neutral, and negative) recorded at 1000 Hz from 15 healthy subjects. There was a current drift. A bandpass filter was used to remove muscle artifacts and direct current drift. It is recommended in [58], i.e., a 4th-order [8, 32] Hz Butterworth filter. We recommend using the Mother of all BCI Benchmark 2 repository [58] (Mother of all BCI Benchmark 2) to increase the number of subjects, with 72 trials per class (e.g., left hand and right hand MIs). EEG signals were imported from an open-source database.

A. Datasets and Preprocessing

We used one MI dataset and one aBCI dataset in the experiments. Their statistics are summarized in Table II. For the MI dataset, subjects sat still in front of a computer screen. An arrow pointing to a certain direction appeared on the screen as a visual cue for a few seconds, during which the subject performed the corresponding MI task. The MI dataset was BCI Competition IV Dataset 2a (MI2) [57]. 22-channel EEGs were recorded at 250 Hz from nine healthy subjects, with 72 trials per class (e.g., left hand and right hand MIs). EEG signals were imported from an open-source repository [58] (Mother of all BCI Benchmark 2) to increase the reproducibility. EEG data processing followed the settings recommended in [58], i.e., a 4th-order [8, 32] Hz Butterworth bandpass filter was used to remove muscle artifacts and direct current drift.

The second dataset was SEED3 [59]. 62-channel EEGs were recorded at 1000 Hz from 15 healthy subjects. There were three classes of emotions (positive, neutral, and negative) induced by short video clips, with five sessions per class. The first session was selected for each subject. The EEG signals were downsampled to 200 Hz, filtered with a 75 Hz low-pass filter, and divided into a series of 1-second non-overlapping segments. Each segment was viewed as a sample, resulting in 3,394 samples per subject. The pre-computed 310-dimension DE features provided in the SEED dataset were used. We also used z-score normalization to scale each feature dimension to mean 0 and standard deviation 1.

B. Implementation Details

We considered unsupervised cross-subject transfers, i.e. one subject was used as the target, and all others as the sources. The classification accuracy was used as the performance measure.

1) Baselines: We compared MSDT with three types of baselines for BCI classification.

a) Classical approaches for MI and aBCI.

- CSP-LDA/DE-LR, which uses six CSP filters to extract log-variance features [7] and linear discriminative analysis (LDA) [60] classifier for MI, and uses DE [10] features with linear regression (LR) [60] classifier for aBCI.
- EA-CSP-LDA, which performs first Euclidean alignment [14] on the raw EEG data before CSP-LDA. This is a simple TL approach for MI.
- RA-MDRM, which performs Riemannian alignment (RA) [13] on the covariance matrices before the minimum distance to Riemannian mean (MDRM) classifier [8]. This is a TL approach for MI.
- EEGNet [61] and ShallowCNN [62], which are two popular deep learning approaches for MI.

b) Source combination approaches, which combine data from all source subjects as the training set.

- DNN, which trains a DNN model and tests on the target data directly.
- DAN, which implements the deep adaptation network (DAN) [63] by accessing the source data.
- DANN, which implements the domain adversarial neural network (DANN) [64] by accessing the source data.
- CDAN, which implements the conditional domain adversarial network (CDAN) [65]. This is an improvement of DANN.
- MCC, which implements minimum class confusion (MCC) [66] by accessing the source data. This is a deep TL approach.
- SHOT, which first trains a DNN model on the combined source data, and then uses source hypothesis transfer (SHOT) [30] by accessing parameters of the source DNN model.

c) Multi-source approaches, which utilize data or models of each source subject with data augmentation.

- Ensemble, which generates a pre-trained model on each source subject and votes their outputs as the final prediction.
- MFSAN [45], which is a multi-source TL approach by accessing the source data.

---

### Table II

**Statistics of the MI Dataset and Emotion Dataset**

| Dataset | # Subjects | # Channels | # Trails | # Classes |
|---------|------------|------------|----------|-----------|
| MI2     | 9          | 22         | 144      | 2         |
| SEED    | 15         | 62         | 3,394    | 3         |

---

2https://github.com/NeuroTechX/moabb
3https://bcmi.sjtu.edu.cn/ seed/seed.html
TABLE III
NEURAL NETWORK ARCHITECTURE

| Module       | Configuration                                      |
|--------------|---------------------------------------------------|
| Backbone     | Linear fully connected layer from input dimensionality to 100 |
|              | Layer normalization, ReLU                         |
|              | Linear fully connected layer from 100 to 50       |
|              | Layer normalization, ReLU                         |
| Classifier   | Linear map from 50 to the number of classes       |
|              | Weight normalization                              |

- MS-MDA [44], which is another multi-source TL approach by accessing the source data.
- SHOT-Ens [30], which first generates a pre-trained model for each source subject, then adapts each source model with unlabeled target data by SHOT, and finally votes their predictions.

2) Network Architecture: All models involving neural networks used the lightweight architecture shown in Table III. The backbone network consisted of two fully connected layers, each followed by layer normalization and ReLU activation. The classifier was a fully connected layer with weight normalization [30].

3) Source Model Training: Each source model was trained on the corresponding source subject’s data locally, without data transmission or information communication. The training batch size was 8 for the MI2 dataset, and 32 for SEED due to its larger number of samples. All source models were trained for 100 epochs. The optimizer was stochastic gradient descent (SGD) with an initial learning rate 0.01 and momentum 0.9. The last 10% samples of each source subject were selected as validation data for MI, and random 10% for aBCI. The models with the best validation performance were saved as the pre-trained source models.

4) Hyper-Parameters: For fair comparison, all algorithms used the same network architecture in Table III. Hyper-parameters of all baselines were set according to their corresponding publications. For the source combination approaches, 50 training epochs were used; the batch size was 8 for MI2, and 32 for SEED. For MSDT, target model adaptation shared the same batch size and optimizer with source model training, but used a different epoch number, 10, for fast convergence.

C. Classification Results

Tables IV and V show the cross-subject classification accuracies on MI2 and SEED, respectively. Each column represents the task when the specified subject was selected as the target domain with all other subjects as source domains. For all tables in this paper, the best accuracy in each task is marked in bold, and the second-best underlined.

According to Table IV, compared with the best baseline using source data (MCC), our MSDT-G improved the average accuracy by 2.23%; compared with the data-free TL approach (SHOT), our MSDT-G improved the average accuracy by 4.32%; compared with the best multi-source baseline (Ensemble), our MSDT-G improved the average accuracy by 2.0%. According to Table V, the corresponding improvements were 0.57%, 3.62% and 5.74%, respectively.

Although the average accuracy of MSDT-G was lower than MS-MDA on SEED, MS-MDA needs to know the EEG data of each individual user, i.e., it may leak the identify and EEG data privacy of the source users. Additionally, MS-MDA performed poorly on MI2, maybe due to its structure specifically designed for aBCI tasks.

Table VI compares the cross-subject classification accuracies under gray-box and black-box settings. MSDT-G and MSDT-B always achieved the best or the second best performances, while protecting the privacy of each source subject.

D. Privacy-Protection Capability

This subsection discusses the privacy-protection capabilities of different approaches.

Private knowledge of the source subjects may be in the form of raw EEG data, extracted features, model parameters, or model APIs. Table VII compares the privacy risk and memory requirement of different approaches.

Source data based approaches need to access the source EEG data or features, which contain rich private information, e.g., emotion, health state, and personal identity. The literature [25] has shown that at least personal identify can easily leak. Model parameter based approaches, e.g., SHOT, SHOT-Ens and MSDT-G, have lower privacy risks. Computer vision applications have shown that data inversion [67] and membership inference [68] can still be used to steal private information, though no similar studies have been performed in EEG-based BCIs. Model API based approaches, e.g., Ensemble and MSDT-B, have the lowest privacy risk. To our knowledge, there has been only one privacy attack [69] to black-box APIs. In summary, MSDT can transfer from source model parameters or APIs and has low memory requirement.

We also performed person identification and authentication experiments to verify the source privacy protection ability of MSDT. Person identification finds out who the subject is. It models each subject as a separate class, and tries to identify the subject’s category. Person authentication seeks to prove or disprove the subject’s claimed identity [25]. A binary classifier is used to admit or reject the claimed identity.

Specifically, for source data based approaches, we extracted EEG features and then adopted support vector machine (SVM) [70] as the base classifier for person identification and authentication. For identification, we selected 80% samples from each source subject as a single class, combined them all-together as the training data, and used the remaining samples as the test data. For M source subjects, the learning task is an M-class classification problem. For authentication, the target trials were used as test data. If the maximum prediction probability of a target trial \( x_t \) was below 2/M, then \( x_t \) was considered not belonging to the training set.

The results are shown in Table VIII. The person identification accuracy was over 99% from the raw source data. Model parameter or API based approaches do not suffer from person identification attacks. All approaches had low privacy risk in person authentication.
TABLE IV
CROSS-SUBJECT CLASSIFICATION ACCURACIES (%) ON MI2

| Approach | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | Avg |
|----------|----|----|----|----|----|----|----|----|----|-----|
| CSP-LDA  | 66.67 | 54.86 | 90.97 | 65.28 | 44.44 | 54.17 | 61.81 | 89.58 | 75.00 | 66.98 |
| EA-CSP   | 86.11 | 53.47 | 91.67 | 70.14 | 55.56 | 67.36 | 63.89 | 90.97 | 77.19 | 72.92 |
| CA-TSM   | 81.94 | 54.17 | 91.67 | **75.69** | 56.94 | 67.36 | 56.94 | 89.58 | 73.61 | 71.99 |
| EegNet   | 79.86 | 50.69 | 89.58 | 59.03 | 46.53 | **71.53** | 52.08 | 92.36 | 80.56 | 69.14 |
| ShallowCNN | 77.78 | 53.47 | 81.94 | 52.65 | 52.08 | 66.67 | 58.33 | 90.28 | 79.17 | 68.44 |

TABLE V
CROSS-SUBJECT CLASSIFICATION ACCURACIES (%) ON SEED

| Approach | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10 | 11 | 12 | 13 | 14 | 15 | Avg |
|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| Classical | 69.18 | 73.87 | **74.31** | 85.71 | 70.30 | 63.64 | 73.51 | 64.44 | 74.13 | 74.34 | 79.43 | 56.22 | 69.53 | 79.35 | 72.42 | 70.03 |
| Combination | 72.13 | 62.79 | 63.08 | 64.50 | 62.40 | 71.27 | 60.49 | 69.62 | 76.52 | 75.37 | 66.12 | 48.82 | 68.44 | 69.71 | 64.82 | 66.41 |
| DNN | 63.76 | 76.16 | 60.14 | 76.87 | 64.47 | 65.29 | 64.20 | **71.63** | 68.39 | 67.47 | 73.84 | 51.24 | 71.36 | 77.34 | 65.03 | 69.15 |
| DAN | 71.92 | 77.61 | 73.36 | 88.98 | 68.03 | 67.35 | 72.30 | 69.30 | 79.46 | 74.37 | *81.79* | 53.03 | **77.46** | 89.28 | 67.27 | 74.10 |
| DANN | 74.75 | 78.98 | 64.97 | **91.43** | 73.90 | **91.54** | 73.73 | 54.46 | 64.98 | 58.43 | 91.57 | 66.17 | 69.65 |
| MCC | 61.52 | 68.55 | 73.72 | 86.39 | 81.35 | 67.62 | 72.22 | 63.05 | **81.76** | **80.67** | 76.31 | 49.65 | 70.68 | **88.75** | 69.80 | **72.69** |
| SHOT | 78.49 | 63.85 | 55.10 | 74.22 | **85.59** | 81.56 | 66.88 | 65.50 | 71.98 | 71.27 | 73.92 | **71.92** | 75.99 | 81.59 | 84.94 | 73.34 |
| Multi-Source | 68.27 | 52.71 | 66.38 | 90.75 | 63.46 | 50.27 | 67.99 | 62.70 | 71.77 | 77.40 | **86.53** | 62.93 | 76.10 | 85.74 | 80.73 | 71.50 |
| MFSAN | 54.50 | 78.17 | 83.74 | 88.89 | 85.30 | **87.54** | **95.37** | **72.92** | 69.33 | 80.17 | **90.93** | 64.50 | **78.90** | **90.54** | 100.00 | **82.05** |
| MS-MDA | 74.45 | 67.95 | 55.57 | 74.93 | 86.27 | 81.82 | 66.93 | 64.67 | 72.30 | 68.90 | 74.39 | 73.01 | 75.96 | 81.91 | 87.89 | 73.50 |
| SHOT-Ens | **79.73** | **65.82** | **53.62** | **74.10** | **84.41** | **87.04** | **67.56** | **70.24** | **78.37** | **73.54** | **76.05** | **73.66** | **77.22** | **83.59** | **99.65** | **76.31** |

TABLE VI
CROSS-SUBJECT CLASSIFICATION ACCURACIES (%) IN GRAY-BOX AND BLACK-BOX SETTNGS ON THE TWO BCI DATASETS

| Approach | MI2 | SEED | Avg |
|----------|-----|------|-----|
| Gray-box | SHOT | 72.22 | 72.69 | 72.46 |
| SHOT-Ens | 74.07 | 73.50 | 73.79 |
| MSDT-G | **76.54** | **76.31** | **76.43** |
| Black-box | Ensemble | 74.54 | 73.34 | 73.94 |
| MSDT-B | **75.54** | **74.73** | **75.15** |

E. Network Architecture

The above experiments used hand-crafted features and a lightweight multi-layer perceptron (MLP) classifier. We also validated the effectiveness of MSDT when end-to-end deep learning models, e.g., EEGNet and ShallowCNN, were used.

Table IX shows the average classification accuracies of several source data free approaches on MI2. SEED was not included since manually extracted DE features were very suitable for it. For EEGNet, all transfer learning approaches performed much better than Ensemble, and MSDT-G achieved the best performance. For ShallowCNN, the four approaches had similar low performance, possibly because the training data were too small to learn good feature representation.

In summary, for a specific subject with a small number of training samples, hand-crafted features followed by an MLP classifier may be a good choice.
TABLE VIII
AVERAGE ACCURACIES (%) IN PERSON IDENTIFICATION AND AUTHENTICATION

| Source Domain Type | Identification | Authentication |
|--------------------|----------------|----------------|
|                    | MI2 SEED       | MI2 SEED       |
| Data               | 99.83          | 99.24          |
| Model parameters   | 3.94           | 0.00           |
| Model APIs         | 3.94           | 0.00           |

TABLE IX
AVERAGE ACCURACIES (%) OF DIFFERENT NETWORK ARCHITECTURES ON MI2

| Network Architecture | ShallowCNN | EEGNet | MLP |
|----------------------|------------|--------|-----|
| Ensemble             | 60.88      | 64.04  | 74.54 |
| SHOT-Ens             | 61.19      | 71.68  | 74.07 |
| MSDT-G               | 62.35      | 73.15  | 76.54 |
| MSDT-B               | 61.27      | 68.06  | 75.54 |

TABLE X
ABLATION ANALYSIS FOR GRAY-BOX MSDT

| Approach               | MI2  | SEED | Avg  |
|------------------------|------|------|------|
| MSDT (w/o $L_{深入}$) | 74.85| 72.59| 73.72|
| MSDT (w/o $L_{深入}$) | 75.77| 75.65| 75.71|
| MSDT (w/o $L_{混合}$) | 76.15| 75.12| 75.64|
| MSDT (w/o STE)         | 75.93| 76.06| 76.00|
| MSDT                   | 76.54| 76.31| 76.43|

F. Ablation Analysis

We conducted an ablation analysis to evaluate how each loss in MSDT contributed to the final performance.

Table X shows the average classification accuracies on the two datasets. Clearly, each of the four strategies in (10) contributed to the improved performance of MSDT, and all four together achieved the best performance.

Fig. 5 shows the correlation between the estimated weight and the source model performance, tested on a target subject in MI2. Generally, better-performing source models had larger weights, indicating that source transferability estimation was effective.

G. Effect of Data Augmentation

Fig. 6 shows the average classification accuracies when the pre-trained models used a single or a combination of data augmentation strategies. Using all four strategies together achieved the best performance, though not all individual strategies were beneficial.

H. Computational Cost

Table XI compares the computational cost (seconds) of several approaches on MI2. All computations were carried out on a single GeForce RTX 3090 GPU. Our proposed MSDT-G and MSDT-B had the best performance, but the smallest computational cost, compared with three existing transfer learning approaches.

V. DISCUSSION

This work considered decentralized transfer learning in BCIs, without accessing source data or features. We proposed MSDT-G, which uses the source model parameters in gray-box settings, and MSDT-B, which uses source model APIs in black-box settings. Cross-subject transfer learning performances in Tables IV-VI show that MSDT-G and MSDT-B achieved the best or second-best performance. We also investigated whether MSDT can protect the identity of the source subjects. Result in Table VIII showed that using source model parameters or APIs is much safer than using source EEG data or features.

MSDT still has some limitations. First, its performance on SEED could be improved. Second, it needs some unlabeled EEG trials from the new subject, which may limit its application to online BCIs. Additionally, developing a practical BCI system needs to consider also the training data size, model
generalization, etc. Thus, our future research will: 1) introduce more data augmentation techniques to enlarge the training data size; 2) develop a well-generalizable classifier for the source subjects, especially in emotion classification; and, 3) extend MSDT to online privacy-preserving transfer.

VI. CONCLUSION

Transfer learning is popular in EEG-based BCI s to cope with variations among different subjects, sessions, tasks, and/or devices. This paper focuses on decentralized TL: there are multiple source subjects, whose data and computations are kept local, and only the parameters or predictions of their pre-trained models can be accessed for privacy-protection; then, how to perform offline unsupervised cross-subject transfer for a new subject with unlabeled EEG trials? We proposed MSDT, which first generates a pre-trained model from each source subject, and then performs decentralized transfer using the source model parameters in gray-box settings or their predictions in black-box settings. Experiments on two datasets from two BCI paradigms (MI and aBCI) demonstrated that MSDT outperformed several existing transfer learning approaches, which do not consider privacy-protection at all. In other words, MSDT achieved both high privacy-protection ability and better classification performance.

REFERENCES

[1] J. Wolpaw, N. Birbaumer, D. McFarland, G. Pfurtscheller, and T. Vaughan, “Brain-computer interfaces for communication and control,” Clin. Neurophysiol., vol. 113, no. 6, pp. 767–791, 2002.
[2] R. P. Rao, Brain-Computer Interfacing: An Introduction. Cambridge, U.K.: Cambridge Univ. Press, 2013.
[3] T. O. Zander, C. Kothe, S. Jatzev, and M. Gaertner, “Enhancing human–computer interaction with input from active and passive brain-computer interfaces,” in Brain-Computer Interfaces. London, U.K.: Springer, 2010, pp. 181–199.
[4] G. Pfurtscheller and C. Neuper, “Motor imagery activates primary sensorimotor area in humans,” Neurosci. Lett., vol. 239, nos. 2–3, pp. 65–68, 1997.
[5] K. H. Kim, S. W. Bang, and S. R. Kim, “Emotion recognition system using short-term monitoring of physiological signals,” Med. Biol. Eng. Comput., vol. 42, pp. 419–427, Jun. 2004.
[6] Z. J. Koles, M. S. Lazar, and S. Z. Zhou, “Spatial patterns underlying population differences in the background EEG,” Brain Topogr., vol. 2, no. 4, pp. 275–284, 1990.
[7] D. Wu, J.-T. King, C.-H. Chuang, C.-T. Lin, and T.-P. Jung, “Spatial filtering for EEG-based regression problems in brain-computer interface (BCI),” IEEE Trans. Fuzzy Syst., vol. 26, no. 2, pp. 771–781, Mar. 2018.
[8] 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, Oct. 2012.
[9] R. Jenke, A. Peer, and M. Buss, “Feature extraction and selection for emotion recognition from EEG,” IEEE Trans. Affect. Comput., vol. 5, no. 3, pp. 327–339, Jul. 2014.
[10] R.-N. Duan, J.-Y. Zhu, and B.-L. Lu, “Differential entropy feature for EEG-based emotion classification,” in Proc. 6th Int. IEEE/EMBS Conf. Neural Eng. (NER), Nov. 2013, pp. 81–84.
[11] D. Wu, Y. Xu, and B.-L. Lu, “Transfer learning for EEG-based brain–computer interfaces: A review of progress made since 2016,” IEEE Trans. Cognit. Develop. Syst., vol. 14, no. 1, pp. 4–19, Jul. 2021.
[12] 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.
[13] 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. 1107–1117, May 2018.
[14] H. He and D. Wu, “Transfer learning for brain–computer interfaces: An Euclidean space data alignment approach,” IEEE Trans. Biomed. Eng., vol. 67, no. 2, pp. 399–410, Apr. 2020.
[15] Y. Xu, X. Huang, and Q. Lan, “Selective cross-subject transfer learning based on Riemannian tangent space for motor imagery brain-computer interface,” Frontiers Neurosci., vol. 15, p. 1487, Sep. 2021.
[16] D. Wu, “Online and offline domain adaptation for reducing BCI calibration effort,” IEEE Trans. Hum.-Mach. Syst., vol. 47, no. 4, pp. 550–563, Aug. 2017.
[17] W. Zhang and D. Wu, “Manifold embedded knowledge transfer for brain-computer interfaces,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 28, no. 5, pp. 1117–1127, Apr. 2020.
[18] P.-Y. Jeng, C.-S. Wei, T.-P. Jung, and L.-C. Wang, “Low-dimensional subject representation-based transfer learning in EEG decoding,” IEEE J. Biomed. Health Inform., vol. 25, no. 6, pp. 1915–1925, Jun. 2021.
[19] X. Hong et al., “Dynamic joint domain adaptation network for motor imagery classification,” IEEE Trans. Cybern., vol. 50, no. 7, pp. 3281–3293, Oct. 2019.
[20] J. Li, S. Qiu, Y.-Y. Shen, C.-L. Liu, and H. He, “Multisource transfer learning for cross-subject EEG emotion recognition,” IEEE Trans. Cybern., vol. 50, no. 7, pp. 556–565, 2021.
[21] Y. Luo, S.-Y. Zhang, W.-L. Zheng, and B.-L. Lu, “WGAN domain adaptation for EEG-based emotion recognition,” in Proc. Int. Conf. Neural Inf. Process., Siem Reap, Cambodia, Dec. 2018, pp. 275–286.
[22] B.-Q. Ma, H. Li, W.-L. Zheng, and B.-L. Lu, “Reducing the subject variability of EEG signals with adversarial domain generalization,” in Proc. Int. Conf. Neural Inf. Process., Sydney, NSW, Australia, Dec. 2019, pp. 30–42.
[23] F. Su, L. Xia, A. Cai, Y. Wu, and J. Ma, “EEG-based personal identification: From proof-of-concept to a practical system,” in Proc. 20th Int. Conf. Pattern Recognit., Istanbul, Turkey, Aug. 2010, pp. 3728–3731.
[24] X. Gu et al., “EEG-based brain-computer interfaces (BCIs): A survey of recent studies on signal sensing technologies and computational intelligence approaches and their applications,” IEEE/ACM Trans. Comput. Biol. Bioinf., vol. 18, no. 5, pp. 1645–1666, Oct. 2021.
[25] A. J. Bidgoly, H. J. Bidgoly, and Z. Arzoumand, “A survey on methods and challenges in EEG based authentication,” Comput. Secur., vol. 93, Jun. 2020, Art. no. 101788.
[26] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Miami Beach, FL, USA, Jun. 2009, pp. 248–255.
[27] R. Xu, N. Baracaldo, and J. Yoshi, “Privacy-preserving machine learning: Methods, challenges and directions,” 2021, arXiv:2108.04417.
[28] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, “Federated learning: Strategies for improving communication efficiency,” 2016, arXiv:1610.05492.
[29] A. Agarwal et al., “Protecting privacy of users in brain-computer interface applications,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 27, no. 8, pp. 1546–1555, Feb. 2019.
[30] J. Liang, D. Hu, and J. Feng, “Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation,” in Proc. Int. Conf. Mach. Learn., Vienna, Austria, Jul. 2020, pp. 6028–6039.
[31] S. M. Ahmed, D. S. Raychaudhuri, S. Paul, S. Oymak, and A. K. Roy-Chowdhury, “Unsupervised multi-source domain adaptation without access to source data,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Nashville, TN, USA, Jun. 2021, pp. 10103–10112.
[32] J. Tian, J. Zhang, W. Li, and D. Xu, “VDM-DA: Virtual domain modeling for source data-free domain adaptation,” 2021, arXiv:2103.14357.
[33] J. Liang, D. Hu, J. Feng, and R. He, “DINE: Domain adaptation from single and multiple black-box predictors,” 2021, arXiv:2104.01539.
[34] D. Wu, X. Jiang, and R. Peng, “Transfer learning for motor imagery based brain-computer interfaces: A tutorial,” Neural Netw., vol. 153, pp. 235–253, Sep. 2022.
[35] H. Li, Y.-M. Jin, W.-L. Zheng, and B.-L. Lu, “Cross-subject emotion recognition using deep adaptation networks,” in Proc. Int. Conf. Neural Inf. Process., Siem Reap, Cambodia, Dec. 2018, pp. 403–413.
[36] X. Tang and X. Zhang, “Conditional adversarial domain adaptation neural network for motor imagery EEG decoding,” Entropy, vol. 22, no. 1, p. 96, 2020.
[37] K. Zhang, N. Robinson, S.-W. Lee, and C. Guan, “Adaptive transfer learning for EEG motor imagery classification with deep convolutional neural network,” Neural Netw., vol. 136, pp. 1–10, Jun. 2021.
[38] L. Xu, M. Xu, Z. Ma, K. Wang, T.-P. Jung, and D. Ming, “Enhancing transfer performance across datasets for brain-computer interfaces using a combination of alignment strategies and adaptive batch normalization,” *J. Neural Eng.*, vol. 18, no. 4, 2021, Art. no. 046055.

[39] C. Chen, C.-M. Vong, S. Wang, H. Wang, and M. Pang, “Easy domain adaptation for cross-subject multi-view emotion recognition,” *Knowl.-Based Syst.*, vol. 239, Sep. 2022, Art. no. 107982.

[40] E. Iacon, W. Ko, and H.-I. Suk, “Domain adaptation with source selection for motor-imagery based BCI,” in *Proc. Int. Winter Conf. Brain-Comput. Interface*, Gangwon, South Korea, Feb. 2019, pp. 1–4.

[41] A. M. Azab, L. Mihaylova, K. K. Ang, and M. Arvaneth, “Weighted transfer learning for improving motor imagery-based brain-computer interface,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 7, pp. 1352–1359, Jun. 2019.

[42] C. Ju, D. Gao, R. Mane, B. Tan, Y. Liu, and C. Guan, “Federated transfer learning for EEG signal classification,” in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Montreal, QC, Canada, Jul. 2020, pp. 3040–3045.

[43] X. Gu, W. Cai, M. Gao, Y. Jiang, X. Ning, and P. Qian, “Multi-source domain transfer discriminative dictionary learning modeling for electroencephalogram-based emotion recognition,” *IEEE Trans. Computat. Social Syst.*, early access, Mar. 8, 2022, doi: 10.1109/TCSS.2022.3153660.

[44] H. Chen, M. Jin, Z. Li, C. Fan, J. Li, and H. He, “MS-MDA: Multisource marginal distribution adaptation for cross-subject and cross-session EEG emotion recognition,” *Frontiers Neurosci.*, vol. 15, pp. 6094–6097, Sep. 2021.

[45] Y. Zhu, F. Zhang, and D. Wang, “Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources,” in *Proc. AAAI*, vol. 33, Jul. 2019, pp. 5989–5996, doi: 10.1609/aaai.v33i01.33015898.

[46] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Lake Tahoe, NV, USA, vol. 25, Dec. 2012, pp. 1097–1105.

[47] D. Freer and G.-Z. Yang, “Data augmentation for self-paced motor imagery classification with C-LSTM,” *J. Neural Eng.*, vol. 17, no. 1, 2020, Art. no. 016041.

[48] W. Freeman, “Hilbert transform for brain waves,” *Scholarpedia*, vol. 2, no. 1, p. 1338, 2007.

[49] J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” 2016, arXiv:1607.06450.

[50] R. Müller, S. Kornblith, and G. Hinton, “When does label smoothing help?” in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2019, pp. 4694–4703.

[51] R. Gomes, A. Krause, and P. Perona, “Discriminative clustering by regularized information maximization,” in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2010, pp. 775–783.

[52] W. Zhang, L. Deng, L. Zhang, and D. Wu, “A survey on negative transfer,” *IEEE/CAA J. Automat. Sinica*, to be published.

[53] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, “mixup: Beyond empirical risk minimization,” in *Proc. Int. Conf. Learn. Represent.*, Toulon, France, Apr. 2017, pp. 1–13.

[54] V. Verma, K. Kawaguchi, A. Lamb, J. Kannala, Y. Bengio, and D. Lopez-Paz, “Interpolation consistency training for semi-supervised learning,” in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Macao, China, Aug. 2019, pp. 3635–3641.

[55] L. Wang and K.-J. Yoon, “Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 3048–3068, Jan. 2022.

[56] S. Kullback and R. A. Leibler, “On information and sufficiency,” *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.

[57] M. Tangermann et al., “Review of the BCI competition IV,” *Frontiers Neurosci.*, vol. 6, no. 1, p. 55, 2012.

[58] V. Jayaram and A. Barachant, “MOABB: Trustworthy algorithm benchmarking for BCIs,” *J. Neural Eng.*, vol. 15, no. 6, 2018, Art. no. 066011.

[59] D. Steyr, R. Scherer, J. Faller, and G. R. Müller-Putz, “Random forests in non-invasive sensorimotor rhythm brain-computer interfaces: A practical and convenient non-linear classifier,” *Biomed. Eng., Biomed. Tech.*, vol. 61, no. 1, pp. 77–86, 2016.

[60] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.

[61] V. Lawhern, A. Solon, N. Waytowich, S. M. Gordon, C. Hung, and B. J. Lance, “EEGNet: A compact convolutional neural network for EEG-based brain–computer interfaces,” *J. Neural Eng.*, vol. 15, no. 5, 2018, Art. no. 056013.

[62] R. T. Schirrmeister et al., “Deep learning with convolutional neural networks for EEG decoding and visualization,” *Hum. Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, Aug. 2017.

[63] M. Long, Y. Cao, J. Wang, and M. Jordan, “Learning transferable features with deep adaptation networks,” in *Proc. ICML*, Lille, France, Feb. 2015, pp. 97–105.

[64] G. Yaroslav and V. Lempitsky, “Unsupervised domain adaptation by backpropagation,” in *Proc. ICML*, Lille, France, Jul. 2015, pp. 1180–1189.

[65] M. Long, Z. Cao, J. Wang, and M. I. Jordan, “Conditional adversarial domain adaptation,” in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2018, pp. 1645–1653.

[66] Y. Jin, X. Wang, M. Long, and J. Wang, “Minimum class confusion for versatile domain adaptation,” in *Proc. Eur. Conf. Comput. Vis.*, Aug. 2020, pp. 464–480.

[67] H. Yin et al., “Dreaming to distill: Data-free knowledge transfer via deep inversion,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 8715–8724.

[68] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, “Membership inference attacks against machine learning models,” in *Proc. IEEE Symp. Secur. Privacy*, San Jose, CA, USA, May 2017, pp. 3–18.

[69] U. Aïvodji, S. Gambs, and T. Thér, “GAMIN: An adversarial approach to black-box model inversion,” 2019, arXiv:1909.11835.

[70] J. C. J. Burges, “A tutorial on support vector machines for pattern recognition,” *Data Mining Knowl. Discovery*, vol. 2, no. 2, pp. 121–167, Jun. 1998.