WLE+ MSFRAN: A Novel Multi-source Domain Adaptation for EEG-based Emotion Recognition

Shitong Shao, and Wei Huan,

***Abstract*—Electroencephalogram (EEG), as an important physiological signal, has been widely applied in many fields. Among them, EEG-based emotion recognition has gradually become a research hotspot. However, the huge distribution differences of EEG signals across subjects make the current research of emotion recognition stuck in a dilemma. In order to solve this problem, we choose the transfer learning method to seek the inspiration for solutions. In this paper, we propose Wide Linear Extractor (WLE) and Multi-source Feature Representation and Alignment Network (MSFRAN) in one framework. The former relies on simple fully-connected layers to learn domain-invariant features and class-separable features. The latter adopts and improves the strategy of multi-source domain adaptation, which not only align the distribution of each pair of source and target domains but also solve the problem of ignoring relevant information between multiple source domains. To verify the validity of WLE+MSFRAN, we carry out cross-subject experiments on two public databases SEED and DEAP. Extensive experimental results show that our model beats other competitors, and achieves great performance and generalization ability for EEG-based emotion recognition.**

***Index Terms*—EEG, Emotion recognition, Wide linear extractor, Multi-source feature extractors, Classifiers alignment**

I. Introduction

EMOTION is an essential physiological response of human beings. Different emotions have different degrees of influence on normal activities of people's daily life. A positive mood is the premise to maintain the normal operation of human physiological function, while a long-term negative mood is more likely to lead to various diseases. The emotional evaluation models can be described from two aspects: the discrete model and the dimensional model. The former consists of multiple discrete states, including happiness, sadness, surprise, anger, disgust, and fear [8]. The latter is composed of multiple continuous dimensionalities, like arousal, valence and dominance [30]. When emotion occurs, it is always accompanied by some external manifestation such as facial expression, intonation, gesture, posture and so on. Meanwhile, current research indicates that emotion is related to the synergy of cerebral cortex and subcortical nerves [39]. It is spontaneous for nervous systems stimulated by emotions to produce reactions.

Therefore, Electroencephalogram (EEG) signals from human brains are more reliable and authentic to reflect the change of emotions. This kind of signal is collected by many brain electrodes made of special material. With the rapid development of wearable devices, it becomes easier and more convenient to record EEG signals. In recent years, EEG-based emotional state recognition has attracted more and more attention from researchers in the field of brain-computer interfaces (BCIs). In this research, signal preprocessing of EEG is the first and very indispensable step. Souvik *et al.* [29] have made a detailed summary about these. This process includes removing noise and artifacts from the EEG and extracting different emotional features from time-domain, frequency-domain, time-frequency-domain and spatial-domain. After that, feature learning and classifier learning are also necessary and important steps and have inspired researchers to explore further.

At present, there are still a lot of difficulties and challenges to be solved in EEG-based emotion recognition. Firstly, different EEG signals of the same subject may quickly change even under the same emotion. Secondly, EEG signals from different subjects under the same emotional state may have huge differences. Thirdly, EEG signals generated by different emotional stimuli often have certain similarities. [21] Because of the above problems, it is hard to design a universally applicable model that can fit different subjects and categorize emotions correctly. In other words, we need to attach importance to the differences between different subjects and consider more realistic scenario, i.e. cross-subject emotion recognition. And for feature extractors based on deep learning, the theoretical foundations of current these extractors are relatively complicated in the EEG-based emotion recognition. And the performance of some extractors is almost the same as that of the full connection layers. Therefore, above these predicaments inspire us to design a simple but efficient feature extractor and a new transfer learning method to reduce data distribution differences between different subjects. In the feature extraction, we use fully-connected layers to construct a effective feature extractor named wide linear extractor (WLE). In the transfer learning, we introduce the concept of multi-source domain adaptation. In view of the shortcomings of multi-source feature space adaptation network (MFSAN) [46] (These will be described in the section II part B), we have made some improvements for network structure and achieve good results in the experiments.

Overall, our contributions are fourfold in the following.

(1) We complete the exhaustive analysis of the depth and width of a fully connected network. This has made a significant preliminary foundation for us to designing new model.

(2) We design a novel and simple feature extractor named wide linear extractor (WLE) for emotion recognition. It surpasses other relevant comparison methods on

This work was supported in part by the Basic Research Project of Leading Technology of Jiangsu Province under Grant BK20192004.

Manuscript received April 19, 2022; revised August 26, 2015.

SEED and DEAP(Valence) and achieves great performance on DEAP(Arousal).

(3) We make some breakthroughs base on MFSAN, which include adding constraints between source domains by random obfuscation to better focus on relevant information across different source domains and selecting the results of partial classifiers with high reliability to predict the labels of target samples.

(4)We integrate new feature extractor and new multi-source domain adaptation into a framework. Experimental results testify that our overall model outperforms other relevant approaches.

The rest of this paper is arranged as follows. Section II investigates existing feature extractors for EEG-based emotion recognition and relevant transfer learning algorithms. Section III makes theoretical analysis for designing a new extractor, and then describes the inner mechanism of its. Section IV illustrates the principle of new multi-source transfer learning method. Section V introduces experimental background, discusses and analyzes the the results of multi-group comparative experiments. Section VI concludes this paper.

II. Related Work

In this section, we will introduce the related work from two perspectives: Feature Extractors, Domain Adaptation Methods.

A. Feature Extractors

For the EEG-based emotion recognition, researchers have conducted many relative studies about feature extraction. They can be mainly divided into two categories: traditional machine learning methods and deep learning methods. In traditional machine learning, Support Vector Machine (SVM) [1], Decision Tree [16], Random Forest [13], [38] and other classifiers have been widely applied to emotion classification. However, the disadvantage of them is that these classifiers rely on manually extracted feature, which will take a lot of time and effort. And in deep learning, feature extraction and classifier learning are usually integrated into a single structure. Marjit *et al* [26] completed EEG-based emotion recognition using Multi-layer Perceptron (MLP) and adopted genetic algorithm to optimize the MLP. Shen *et al* [31] proposed a novel 4D convolutional recurrent neural network which combines the convolutional neural network (CNN) with long short term memory (LSTM) in order to extract the frequency, spatial and temporal information of multi-channel EEG signals. As the attention mechanism has become more popular, Tao *et al* [35] presented an attention-based convolutional recurrent neural network (ACRNN) to explore EEG information in channel and time. Liu *et al* [22] apply temporal and spatial attention simultaneously in conventional Transformer structure [37] to learn spatiospectral feature from EEG. Song *et al* [33] used dynamical graph convolutional neural networks (DGCNN) to dynamically study the connections between different channels in EEG signals so as to overcome the limitations of GCNN method [7]. The design of these extractors is derived from complex theoretical derivation and contains knowledge from many fields. These need to take time and efforts to understand and learn. Based on this point, we hope to design a simple and effective feature extractor for subsequent research.

B. Domain Adaptation Methods

In order to solve data distribution difference and find invariant features of source and target domains, researchers try to devise new domain adaptation methods from different angles. Some studies seek for a mapping function ϕ to minimize the distance of two domain, i.e. $\|\phi(P(X_s)) - \phi(P(X_t))\|$, where X_s and X_t represent the source and target samples, respectively. There are many metrics to measure the distance, including Kullback-Leibler divergence [15], Jensen-Shannon divergence [9] and so on. In the field of transfer learning, Maximum Mean Discrepancy(MMD) [2] has been widely used. Transductive Component Analysis (TCA), as a classical domain adaptation method, was firstly proposed by Pan *et al.* [27]. It uses MMD to learn some transfer components across domains in a Reproducing Kernel Hilbert Space, and then projects source and target domain data into common low-dimensional feature subspace spanned by these transfer components. And then Tzeng *et al.* [36] added an adaptive layer and domain confusion loss based on MMD in the convolutional neural network architecture, named Deep Domain Confusion(DDC) to automatically learning domain invariance. To avoid ignoring class weight bias across domains, Yan *et al.* [40] devised a weighted-MMD which assigns class-specific auxiliary weights to explore the class prior probability between source and target domains based on MMD. Zhang *et al.* [44] utilized the DDC method with Electrode-Frequency Distribution Maps(EFDMs) of EEG as input data to realize the emotion classification. Long *et al.* [23] further proposed Deep Adaptation Networks(DAN), which uses multi-kernel MMD (MK-MMD)and multiple adaptive layers to enhance transferability of features so as to greatly overcome the limitation of DDC. And Sun *et al.* [34] proposed another metric loss named CORAL loss, which is aimed at aligning the second-order statistics of the source and target data distributions, so as to minimize the difference of feature covariances between domains.

Some studies focus on the feature selection between two domains. Yin *et al.* [41] suggested a cross-subject feature selection algorithm named Transfer Recursive Feature Elimination(TRFE) for emotion recognition. This method makes use of different features extracted from the temporal and frequency domain, chooses the most robust sharing features and ignores the uncommon features between domains. Some studies concentrate on the subspace learning method. Chai *et al.* [6] integrated auto-encoder network with subspace alignment to construct a unified framework after considering fully nonlinear transformation and a consistency constraint. Chai *et al.* [4] proposed a strategy named Adaptive Subspace Feature Matching. They developed a linear transformation function to reduce the marginal and conditional distribution discrepancies.

Motivated by the framework of Generative Adversarial Nets (GAN) [11], now more and more researchers begin to introduce the adversarial learning strategy into domain adaptation to automatically achieve common invariant features. Jin *et al.* [14] firstly introduce the domain adaptation network(DAN) which is suggested by Ganin *et al.* [10] into the EEG-based emotion recognition. DAN consists of four parts: feature extractor, domain classifier, class predictor and gradient reversal layer. By adding a gradient reversal layer into domain classifier, feature extractor seeks for a feature mapping to maximize the loss of domain classifier(make the feature distribution of two domains as similar as possible) while domain classifier minimize the loss of domain(make the feature distribution of two domains as different as possible) by back propagation. Through adversarial learning of two parts, domain invariant features can be acquired during the training process. Pei *et al.* [28] presented a multi-adversarial domain adaptation (MADA), which captures multi-mode structures to achieve more fine-grained data alignment based on multiple domain discriminators. Yu *et al.* [42] combined global domain confrontation with local domain confrontation named Dynamic Adversarial Adaptation Network(DAAN). It can adaptively assign the weights to adjust the marginal and conditional distribution. Luo *et al.* [25] intorduced a Wasserstein generative adversarial network(GAN) into domain adaptation. This method applies Wasserstein GAN gradient penalty loss to narrow the gap between source and target domain in adversarial-training. Li *et al.* [19] simultaneously considered the condition and marginal distribution of EEG samples. They used adversarial strategy(gradient reversal) in the shallow layers of network to adjust the marginal distributions and adapted association reinforcement to solve the conditional distributions in the deep layers.

However, these methods mainly focus on single-source domain adaptation methods. In a real practical scenario, data may come from different places, thus there are huge distribution differences from each other. Thus combining these data into one single source domain can be detrimental to the follow-up researches. Meanwhile, in the emotion recognition, EEG datasets usually consist of multiple diverse subjects' data samples. Different subjects' individual differences can lead to the huge distribution differences of the collected EEG signals. Therefore, multi-source domain adaptation methods are more suitable for the EEG-based emotion recognition. Many researchers set out to start relevant research for this conundrum. Li *et al* [20] suggested a multi-source transfer learning for cross-subject EEG Emotion Recognition. The first step is to search for the appropriate sources that fit the target. The second step is to utilize the style transfer mapping to reduce the differences between each source and target. Cao *et al* [3] put forward a model named multi-source and multi-representation adaptation (MSMRA) with two stages. This method firstly splits the EEG data from diverse subjects and sessions into multiple domains and then aligns the distribution of multiple representations. Gu *et al* [12] integrated the transfer learning and dictionary learning into one model named multi-source domain transfer discriminative dictionary learning modeling (MDTDDL). This model first projects EEG signal of multiple domains into one transferable subspace by the domain-specific transformation matrix, then uses the dictionary learning to explore the latent correlations between multiple source domains and target domain. Therefore, in view of the current research situation, we introduce multi-source transfer learning method to effectively solve the problem of domain differences. Zhu *et al* [46] proposed a Multiple Feature Spaces Adaptation Network (MFSAN) with two alignment stages. It not only aligns the data distribution of each pair of source and target domains, but also aligns the outputs of classifier by the way of domain-specific decision boundaries. Since MFSAN is first proposed for the task of image classification, the number of source domains is relatively small. And this method also fails to consider relevant information across multiple source domains. Therefore, we will make adaptive improvements base on MFSAN so as to better apply this improved method to realize the emotion classification of multi-source domains.

III. Feature extractor

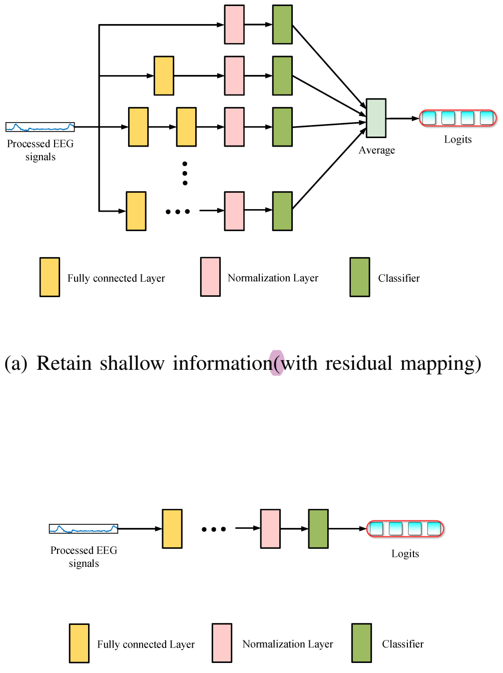
In deep learning networks, many researchers have designed some feature extractors for EEG-based emotion classification, such as the aforementioned CNN-LSTM, DGCNN, ACRNN and so on. But the theoretical basis of these structures is relatively complex, which requires a lot of time for some researchers to understand and realize. Moreover, the performance of these carefully designed extractors is not significantly enhanced in cross-subject emotion recognition. So far few researchers have explored the effect of simple network structure on emotion recognition. Inspired by this, we regard the fully connected network as the base model to analyze the performance changes caused by different width and depth. In addition, we have excavated a simple effective feature extractor named wide linear extractor(WLE) according to the analysis results.

A. Width and Depth Analysis of Fully Connected Networks

In this subsection, we take the depth and width of network constructed by stacked fully connected layer as the research variables, and make a reasonable analysis by contrast experiments.

We first make the definitions on some concepts. When nonlinear function do not exist between the fully connected layers, we use $\{\mathcal{F}_i\}_{i=1}^N$ to represent fully connected layers of sequential connections, where N is the number of the fully connected layers. And $\mathcal{W}_i$ and b_i represent the weight matrix and the bias of $\mathcal{F}_i$, respectively. $\mathcal{Y}_{i-1}$ is both the input of $\mathcal{F}_i$ and the output of $\mathcal{F}_{i-1}$. In the data preprocessing stage, we usually normalize the EEG signal. So we can assume that input sample $\mathcal{X}$ follow a standard normal distribution $\mathcal{N}(0,1)$. When $\mathcal{X}$ is entered into the $\mathcal{F}_1$, the mapping formula is $\mathcal{Y}_1 = \mathcal{W}_1\mathcal{X} + b_1$. In the process of training, $\mathcal{W}_1$ follows $\mathcal{N}(0,\sigma_w^2)$ where σ_w^2 is a minimal quantity, and $\mathcal{Y}_1$ follows $\mathcal{N}(\mu_y,\sigma_y^2)$. For the sake of simplicity, we ignore the variable b_1 in the mapping formula, so μ_y, σ_y^2 can be expressed as

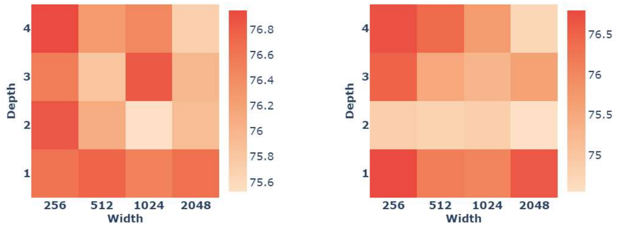
$$
\begin{aligned}
\mu_y &= \frac{0 * \sigma_w^2 + 0 * 1}{1 + \sigma_w^2} = 0, \\
\sigma_y^2 &= \frac{\sigma_w^2 * 1}{1 + \sigma_w^2} \qquad \approx \sigma_w^2
\end{aligned}
\tag{1}
$$



(a) Retain shallow information(with residual mapping)

(b) Non-retain shallow information(without residual mapping)

Fig. 1: Compare the performance of models with different depth and width



(a) Retain shallow information (with residual mapping)

(b) Non-retain shallow information (without residual mapping)

Fig. 2: Accuracy thermodynamic diagram based on test benchmark

Similarly, when data flow through $\mathcal{F}_i$, the variance of its output $\mathcal{Y}_i$ can be expressed by the Equation 2.

$$\sigma^2(\mathcal{Y}_i) = \frac{\sigma^2(\mathcal{W}_i)\sigma^2(\mathcal{Y}_{i-1})}{\sigma^2(\mathcal{W}_i)+\sigma^2(\mathcal{Y}_{i-1})} < \min(\sigma^2(\mathcal{W}_i),\sigma^2(\mathcal{Y}_{i-1})) \tag{2}$$

Therefore, as shown in the Equation 3, we can compare the variances of the output $\mathcal{Y}_i$ of each fully connected layer $\mathcal{F}_i$ and input sample $\mathcal{X}$. Then we can find that the variance of $\mathcal{Y}_i$ is getting smaller and smaller.

$$\sigma^2(\mathcal{Y}_N) < \cdots < \sigma^2(\mathcal{Y}_2) < \sigma^2(\mathcal{Y}_1) < \sigma^2(\mathcal{X}). \tag{3}$$

Next, we will use norms to analyze the impact of this phenomenon that the variance of each layer's output will be smaller and smaller in the forward. We can get Equation 4 through simple math:

$$\|\mathcal{Y}_N\|_2 = \left(\sum_{i=1}^{N'} \left(\mathcal{Y}_N^i\right)^2\right)^{\frac{1}{2}} \tag{4}$$

where N' refers to the number of elements in $\mathcal{Y}_N$ and $\mathcal{Y}_N^i$ refers to an element in $\mathcal{Y}_N$. When $N \to +\infty$, the Equation 4 can be converted into an integral:

$$\begin{aligned}
\|\mathcal{Y}_N\|_2 &\approx \left(\int_{-\infty}^{+\infty} N'\rho\left(x\right)x^2 dx\right)^{\frac{1}{2}}, x \in (-\infty, +\infty) \\
&\approx \left(2\mathrm{N}\sigma(\mathcal{Y}_N)^2 \int_{-\infty}^{+\infty} exp\left(\frac{-x^2}{2\sigma(\mathcal{Y}_N)^2}\right)\frac{x^2}{2\sigma(\mathcal{Y}_N)^2} d\left(\frac{x}{\sqrt{2}\sigma(\mathcal{Y}_N)}\right)\right)^{\frac{1}{2}} \\
&\approx \left(2\mathrm{N}\sigma(\mathcal{Y}_N)^2 \left(\frac{exp(-x^2)\left(\sqrt{\pi}exp(x^2)erf(x)-2x\right)}{4}\right)\Bigg|_{-\infty}^{+\infty}\right)^{\frac{1}{2}} \\
&\approx \pi^{\frac{1}{4}} N^{\frac{1}{2}} \sigma(\mathcal{Y}_N).
\end{aligned} \tag{5}$$

Observing at the conclusion of Equation 5, we can easily find that $\|\mathcal{Y}_N\|_2$ is proportional to the standard deviation of the distribution it follows. The above phenomenon causes gradient attenuation, which means that our analysis experiments are unfair to the deeper model.

Based on the above theoretical analysis, to ensure the experimental fairness, we need to add normalization operation before classification layer of each comparison model. Therefore, two groups of the experiment have been carried out, which are shown in figure 1. One preserves shallow information while increasing depth and width of model and the other does not, as shown in figure 1(a) and figure 1(b). Specifically speaking, width represents the number of neurons in each full connected layer. In the first group, depth represents the number of branches of fully connected layers and the number of fully connected layers in each branch. In the second group, it can be seen as a further analysis based on part of the first set of experiments. Depth only represents the number of fully connected layers. We adopt the control variable method to conduct the experiments. The results are shown in figure 2(a) and 2(b) respectively.

As can be seen from the figure 2, increasing the depth of the model can improve the performance to some extent when preserving the shallow branches information. However, if the shallow information is deserted, model depth increasement will not bring very good result. This means that major features of EEG signals can be studied at the shallow layers, thus there is no need to increase the model depth for depth features by consuming a lot of computational costs. Therefore, we focus primary attention on shallow layer information and design a simple and effective feature extraction structure based on it.

B. Wide Linear Extractor

Based on the above analysis, the feature extractor we newly constructed, named Wide Linear Extractor (WLE), is shown in figure 3. It enlarges the network width and can

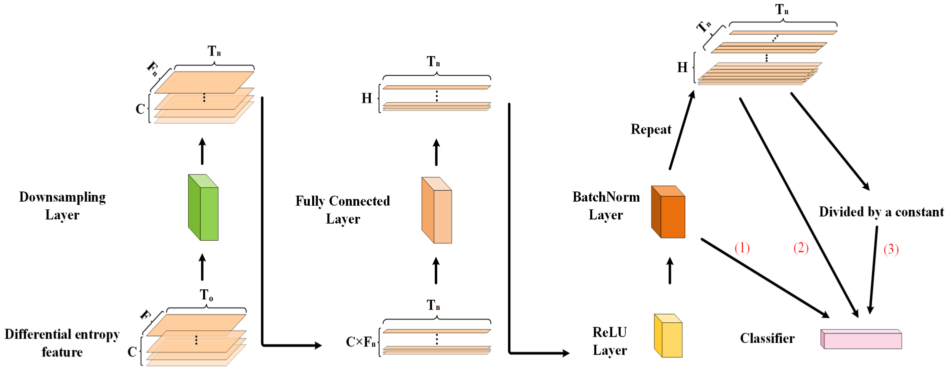


Fig. 3: The structure of Wide Linear extractor

stabilize convergence by reducing the step size of gradient update. The whole structure includes a downsampling layer, a fully connected layer, activation function(ReLU) layer and batchnorm layer. First of all, differential entropy feature is extracted from EEG signal. Its data form $\mathcal{X}$ can be described as $\mathcal{X} \in \mathbb{R}^{C \times F \times T_o}$, where C represents the number of electrodes, F represents the number of frequency band in frequency domain and T_o represents the length of EEG duration. The downsampling operator is used to deal with dimensions F and T_o, which will smooth the EEG signal and reduces the impact of data noise on model performance. These two dimensions will be mapped to the same value T_n. Then we associate the information of electrodes and frequency band through the fully connected layer. Our remarkable contribution is the processing in front of the classifier. As is shown in figure 3, most models feed the output of the feature extractor directly to the classifier to get the logits. However, our proposed method first duplicates the output T_n times, divides it by a constant greater than 1, and finally feeds it into classifier. Using this simple and effective method can effectively improves the performance compared to not using this method.

As stated in the [24], a wider model can be better able to find the global optimal solution. Repeating the input of the classifier can effectively increase the width. But copy operation can cause a potential problem. That is to increase the norm of the input of the classifier.

$$
\begin{aligned}
\mathcal{O}(\mathcal{X}_{original}) &= \mathcal{O}(\{X_i\}_{i=1}^{N}) = \left(\sum_{i=1}^{N} X_i^2\right)^{\frac{1}{2}}, \\
\|\mathcal{X}_{repeat}\|_2 &= (T_n \times \mathcal{O}(\mathcal{X}_{original}))^{\frac{1}{2}} \\
&= \sqrt{T_n} \times \|\mathcal{X}_{original}\|_2, \\
\|\mathcal{X}_{repeat}\|_1 &= \quad T_n \times \|\mathcal{X}_{original}\|_1,
\end{aligned}
\tag{6}
$$

where $\mathcal{X}_{original}$ refers to the input vector of the classifier, N refers to the length of the $\mathcal{X}_{original}$. $\|\mathcal{X}_{repeat}\|_1$ denotes l_1 norm. From the formula 6, we can see that the norm of the copied input has greatly increased. In order to update the gradient more smoothly in back propagation, we need to divide the copied input by a constant g. It is also worth noting that g is chosen from $\sqrt{T_n}$ and T_n in experiments with different datasets. Details of the experiments are given in the part B of Section IV.

IV. Multi-source feature representation and alignment network

Beforehand, we first introduce some definitions for better describing this method. We regard the all EEG samples data of every subject as an independent domain. In this case, we choose one subject as the target domain and the remaining subjects as the source domain. In addition, after further considering the differences between different subjects in the source domain, each subject in the source domain can be treated as an independent sub-source domain. Thus, multiple source domains and target domain can be denoted as $\{\mathcal{D}_{si}\}_{i=1}^{K} = \{X_{si}, Y_{si}\}_{i=1}^{K}$ and $\mathcal{D}_t = \{X_t\}$, respectively. Where K represents the number of source domains, X_{si} represents the data samples from i-th source domain $\mathcal{D}_{si}$, Y_{si} represents the corresponding ground-truth labels. In the target domain $\mathcal{D}_t$, only unlabeled data samples X_t are included. In the cross-subject emotion recognition, due to the existence of individual differences, it is difficult for feature extractor to learn common domain-invariant feature from all domains. On the contrary, it is relatively easy to achieve the domain-invariant features between each pair of source and target domains. However, as the number of source domains increases, if we assign a feature extractor for each pair of source and target domains, the demand for computing resources will be enormous. And the large distribution difference between source domains also need to be considered and solved. In addition, the labels of the same sample under the prediction of multiple classifiers are often different in multi-source domain adaptation. How to solve these problems is worth studying. Under this situation, we improve the structure of MFSAN and form new method named multi-source feature representation and alignment network(MSFRAN). As is shown in figure 4, MSFRAN is composed of four parts: common feature extractor, multi-domain feature extractors, classifiers alignment and multi-domain classifiers. Meanwhile, taking into account the difference in the number of samples from different subjects will affect the final model performance. Thus, we design a sampler to solve the problem of samples imbalance during data input process. The sampler can ensures that the number of samples from every subject is the same. Specifically, each domain contains m randomly selected samples from every subject. Then multiple source domains include $K \times m$ samples

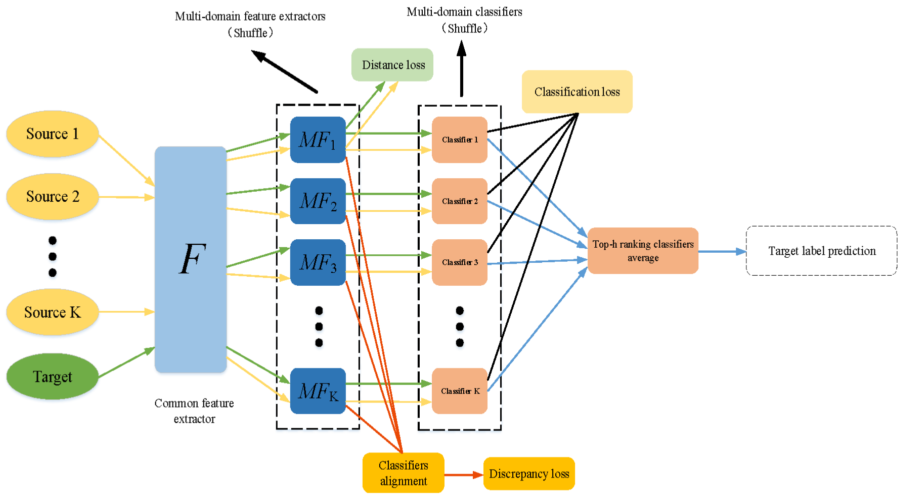


Fig. 4: The structure of Multi-source Feature Space Alignment Network

and target domain consists of m samples in one batch.

A. The acquisition of multiple domain-invariant features

Specifically, the acquisition of multiple domain-invariant features mainly depends on common feature extractor and multi-domain feature extractors. Firstly, adopting the common feature extractor $F(\cdot)$ can learn some common domain-invariant representations in advance. Theoretically, this block can be replaced by other proposed networks. Multi-domain feature extractors $\{MF_i(\cdot)\}_{i=1}^{K}$ consist of K small sub-networks that do not share parameters. The function of $MF_i(\cdot)$ is to map i-th source doamin and target domain into a specific feature space after the process of $F(\cdot)$. This will contribute to learning specific domain-invariant feature of each pair of source and target domains. These two parts will greatly reduce time consumption and demand for computing resources. To measure the distance between each source domain and target domain, we choose the MK-MMD to calculate. The distance formula can be described as

$$M_k(\mathcal{D}_s,\mathcal{D}_t) = \frac{1}{n_s^2}\sum_{i=1}^{n_s}\sum_{j=1}^{n_s}k(x_i^s,x_j^s)+\frac{1}{n_t^2}\sum_{i=1}^{n_t}\sum_{j=1}^{n_t}k(x_i^t,x_j^t) - \frac{2}{n_sn_t}\sum_{i=1}^{n_s}\sum_{j=1}^{n_t}k(x_i^s,x_j^t) \tag{7}$$

Where k stands for the characteristic kernel, $k(x_i,x_j) = \langle\phi(x_i),\phi(x_j)\rangle$, $\langle\cdot,\cdot\rangle$ represents inner product of vectors, n_s, n_t represent the number of source domain samples and target domain samples respectively. Therefore, the final distance loss of K MMD loss can be expressed as:

$$\mathcal{L}_{dist} = \frac{1}{K}\sum_{i=1}^{K} M_k(MF_i(F(X_{si})), MF_i(F(X_t))) \tag{8}$$

Meanwhile, we select the shuffle operator for multi-domain feature extractors during the training process. Specifically speaking, for the data of each pair of source and target domains after $F(\cdot)$, we do not choose a fixed multi-domain feature extractor for them. After creating $\{MF_i(\cdot)\}_{i=1}^{K}$, we first randomly shuffle their order. Then, these data are trained to extract features by utilizing a randomly selected extractor, such as j-th extractor $MF_j(\cdot)$. We initially set the number of batch to 0, and then we continuously accumulate the number of batch in each epoch. When the number satisfies our preset condition (i.e. The number of batch is an integer multiple of 10), we will select an extractor at random again, for example k-th extractor $MF_k(\cdot)$ to process the data from the same pair of source and target domains. The main purpose of this operation is to strengthen the correlation between the multiple source domains. Carefully speaking, due to the huge individual differences of EEG signals, the distribution of different subjects can be quite different. This leads us to attach full importance to the distributions of multiple source domains. If we adopt the fixed multi-domain feature extractors, each pair of source and target domains is relatively independent. This will ignore relevant information among diverse source domains and is unfavorable to extract some common features between source domains. And it is inadequate for common feature extractor to learn domain-invariant features of all domains. Thus, shuffle operator can further solve this problem effectively to some extent.

B. Classifiers alignment

Since K source domains produce K classifiers and these classifiers can predict different labels for the same target sample, especially those at the classification boundary. In order to be able to predict the label more accurately in the following process, we refer to the method [46] to reduce the discrepancy among multiple domain classifiers. We use softmax function $S(\cdot)$ to process the output of every classifiers. Then the discrepancy of target domain samples between multi-domain classifiers can be expressed as:

$$\mathcal{L}_{disc} = \frac{1}{K(K-1)} \sum_{i=1}^{K-1} \sum_{j=i+1}^{K} \Big| S(MF_i(F(X_t))) - S(MF_j(F(X_t))) \Big| \tag{9}$$

where $|\cdot|$ represents the absolute value operation.

C. Multi-domain Classifiers

After going through multi-domain feature extractors, multiple feature representations are fed into multi-domain classifiers $\{C_i(\cdot)\}_{i=1}^{K}$. In this part, we still use the randomly shuffle operator for multiple domain classifiers. The condition for shuffle operator is the same as that of multi-domain feature extractors (i.e. The number of batch is an integer multiple of 10). While there are fewer shuffle operations in this part compared with those in the multi-domain feature extractors. This is because we do not want to overly confuse multiple classifiers. Our main goal is to make the classifier focus primarily on the feature representation of each pair of source and target domains. But we also hope the classifier can also take into account the relationships between the source domains extracted in the previous step. So we set up a small number of shuffle operations. For each classifier, we calculate the classification loss by cross entropy $J(\cdot)$. The final classification loss of multi-domain classifiers can be expressed by the following formula:

$$\mathcal{L}_{class} = \frac{1}{K} \sum_{i=1}^{K} J(C_i(MF(F(X_{si}))), Y_{si}) \tag{10}$$

Because the classifier and the multi-domain extractor are not one-to-one correspondence, when calculating the cross entropy loss of each classifier, the multi-source extractor may be any one of extractors. Thus, in formula 10, we use $MF(\cdot)$ without subscript to describe. For the probability output of all classifiers, we do not uses the average of them to predict the labels of target samples. In this paper, we first adopt softmax function to process the output of K classifiers. Then we can achieve the maximum value on the category dimension of each classifier's output. The number of maximum values is K. After that we sort the these maximum values in descending order and get maximum value's index. The index represents which classifier maximum value belongs to. And then we select the corresponding classifiers according to the top h maximum values. We average the output of top h classifiers as final output of our model to predict the labels of target samples.

The total loss of MSFRAN can be defined as equation 11.

$$\mathcal{L}_{total} = \mathcal{L}_{class} + \lambda(\xi_1 \mathcal{L}_{dist} + \xi_2 \mathcal{L}_{disc}) \tag{11}$$

λ is trade-off parameter, which can be expressed as equation 12, where p is a hyperparameter set based on batch size. ξ_1 and ξ_2 are the weights of their loss functions, respectively. By minimizing the $\mathcal{L}_{total}$, MSFRAN can learn multiple domain-invariant features, relevant information between source domains and more accurate classification results.

$$\lambda = \frac{2}{1+e^{-10p}} - 1 \tag{12}$$

The training process of MSFRAN is shown in Algorithm 1

Algorithm 1 Overview of MSFRAN

```
Input: EEG data of $K$ source domains $\mathcal{D}_{si}$ and target domain
    $\mathcal{D}_t$; $T$: The number of training iterations; $B$: The number
    of batch and it starts at 0;
Output: The labels of target domain samples $X_t$
for t in 1: $T$ do
    Randomly selected $m$ samples $X_{si}^m$ from each source
    domain $\mathcal{D}_{si}$;
    Randomly selected $m$ samples $X_t^m$ from target domain
    $\mathcal{D}_t$;
    Feed the data of source domains and target domain into
    common feature extractor $F(\cdot)$ to learn some domain-
    invariant feature $F(X_{si}^m)$ and $F(X_t^m)$;
    if B%10==0 then
        Shuffle the order of multi-domain feature extractors
        and multi-domain classifiers to get $MF_j(\cdot)$ and $C_l(\cdot)$,
        respectively;
        Feed the $F(X_{si}^m)$ and $F(X_t^m)$ into the $MF_j(\cdot)$;
        Use the equation [8] to calculate the distance loss of
        $MF_j(F(X_{si}^m))$ and $MF_j(F(X_t^m))$;
        Feed the $MF_j(F(X_{si}^m))$ into the $C_l(\cdot)$ and use the
        equation [10] to calculate the classification loss after
        $C_l(\cdot)$;
        Use the equation [9] to calculate the discrepancy of
        multi-domain classifiers;
    else
        Shuffle the order of multi-domain feature extractors
        to get $MF_k(\cdot)$;
        Feed $F(X_{si}^m)$ and $F(X_t^m)$ into the $MF_k(\cdot)$;
        Use the equation [8] to calculate the distance loss of
        $MF_k(F(X_{si}^m))$ and $MF_k(F(X_t^m))$;
        Feed $MF_k(F(X_{si}^m))$ and $MF_k(F(X_t^m))$ into the
        $C_l(\cdot)$ and use the equation [10] to calculate the
        classification loss after $C_l(\cdot)$;
        Use the equation [9] to calculate the discrepancy of
        multi-domain classifiers;
    end if
    Update the parameters of common feature extractor
    $F(\cdot)$, multi-domain feature extractors $\{MF(\cdot)\}_{i=1}^{K}$ and
    multi-domain classifiers $\{C_i(\cdot)\}_{i=1}^{K}$
end for
```

V. Experiments and discussion

A. Background of experiments

1) Dataset Discription: In this paper, we use two publicly available datasets to test the validity of our proposed model for emotion recognition,which is respectively called SJTU Emotion EEG Dataset (SEED) and DEAP.

The SEED dataset [45] consists of 15 chinese subjects (7 males and 8 females; mean age: 23.27). Each subject is asked to participate in experiment three times at intervals of one week or longer. There are 15 chinese film clips (induce positive, neutral and negative emotions) in each experiment. The duration of each film is about four minutes. Every subject need to make self-evaluation after the film ends. The

EEG signals were recorded by a 62- channel ESI NeuroScan System, at a sampling rate of 1000 Hz and downsampled to 200 Hz. And then a bandpass filter from 0 - 75 Hz was applied to remove noise and interference.

The DEAP dataset [17] includes 32 subjects (16 males and 16 females; mean age: 26.9). Each subject is required to watch 40 1-min music video clips and make self-assessment from five dimensions: 1) valence (associated with pleasantness level); 2) arousal (associated with excitation level); 3) dominance (associated with control power); 4) liking (associated with preference); and 5) familiarity (associated with the knowledge of the stimulus) [18]. The EEG signals were recorded by a 48-channel (32 EEG channels, 12 peripheral channels, 3 unused channels and 1 status channel) Biosemi Active Two devices at a sampling rate of 512 Hz and downsampled to 128 Hz. Then signals go through a bandpass filter between 4 and 45 Hz.

2) Evaluation Protocol: In this paper, we adopt leave-one-subject-out cross validation(LOSOCV) to evaluate performance results in all experiments. Specifically, we take one subject as target domain data and regard other subjects as the source domain data. All results in our experiments are shown in Mean (Standard deviation). We report evaluation results at the end of training. At the same time, in order to ensure the reliability and authenticity of the experiment, the LOSOCV experiment will be repeated **5** times and averaged. We also use bold to highlight the overall best results.

3) Data Preprocessing: On all datasets used in our experiments, the input EEG features are differential entropy (DE) features. DE features have been widely adopted in many researches of transfer learning on EEG-based emotion recognition [5], [18], [20]. The differential entropy feature is defined as follows. Where the time series T obey the Gauss Distribution $\mathcal{N}(\mu, \sigma^2)$. In [32], it has proved that when T is filtered by band pass filter, the time-series of the sub-band signals follow the Gaussian distribution.

$$
\begin{aligned}
DE(T) &= -\int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right) ds \\
&= \frac{1}{2}\log(2\pi\sigma^2)
\end{aligned}
\tag{13}
$$

Let us define $\mathcal{X}$ as a data sample, $\mathcal{X} \in \mathbb{R}^{C \times T}$, where C is the number of channels and T is the length of time sequence length in the frequency domain. In the SEED, C is set to 62. The DE is calculated over five frequency bands (δ: 1-4Hz, θ: 4-8Hz, α: 8-13Hz, β: 13-30Hz, γ: 30-50Hz). We choose each 180-second EEG signal as one sample and normalize each frequency band on dimension T to obtain the normalization data of five frequency bands. In the DEAP, C is set to 32. The DE is calculated over four frequency bands except δ frequency band. We choose 60-second EEG signal in each trail as one sample and then cut each second in half, thus T is set to 120. Normalization processing is implemented like SEED.

B. Experiments on feature extractor

On the SEED dataset, we compare the effects of different g on model performance. By contrast experiment in Table I, we find setting g to T_n can achieve the best result. If only the data is copied without dividing g, then its performance is almost the same as that without copying. However, if the feature vector is divided by g before entering the classifier, even if g is $\sqrt{T_n}$, a certain improvement in test accuracy can be obtained. This can also be demonstrated on other datasets. To achieve the best result, g is set to $\sqrt{T_n}$ on DEAP (Valence). g is set to T_n on DEAP (Arousal). The conclusion that can be drawn is that for different datasets, the selected g is not the same. We infer that this is due to the characteristic of the dataset itself.

TABLE I: Comparison of different processing methods of output of wide linear extractor

Methods	1	2	3	4	5	Mean
(1)	75.71(6.81)	78.22(8.86)	79.85(7.76)	77.93(9.09)	76.89(7.82)	77.72(8.07)
(2)	77.19(9.95)	80.00(8.55)	78.37(5.59)	77.18(7.68)	76.89(6.68)	77.93(7.69)
(3) constant=$\sqrt{T_n}$	76.59(8.15)	78.37(9.58)	78.52(7.85)	80.15(8.09)	77.48(7.25)	78.22(8.19)
(3) constant=T_n	**82.37(6.10)**	**82.96(6.42)**	**83.41(5.50)**	**84.15(5.79)**	**82.22(7.12)**	**83.02(6.18)**

The results are shown in Mean (Standard deviation) on SEED dataset. 1, 2, 3, 4, 5 represent the sequence of experiments. The methods (1), (2) and (3) stands for different processing operations to the output of the feature extractor in figure 3. And the constant represents the number to be divided by the output of the feature extractor.

TABLE II: Comparison of different feature extractors

Methods	SEED	DEAP	
		Valence	Arousal
Random Forest	65.925(6.964)	56.406(8.522)	**57.890(15.812)**
Decision Tree [16]	70.814(9.175)	58.437(7.874)	55.625(11.110)
SVM	78.074(7.475)	59.297(7.348)	57.578(15.693)
4D-CRNN [31]	40.191(4.378)	50.414(8.509)	49.625(9.388)
MLP	79.881(8.008)	58.938(8.005)	55.063(11.505)
Transformer [37]	79.526(7.656)	56.203(7.959)	54.453(10.216)
WLE (ours)	**83.022(6.184)**	**59.313(10.035)**	57.031(10.028)

Our proposed WLE achieves the best performance on SEED and DEAP (Valence), with a difference of only 0.8% on DEAP (Arousal)

Then we make a comparison of different feature extractors without transfer learning method. Specifically, we only use some classic machine learning or deep learning models as baseline models. For all machine learning methods, we use grid search to select the best result from 50∼100 different sets of hyperparameter combinations; for all deep learning methods, we use experience-based principles to tune parameters and achieve the best results. All experimental results are shown in Table II, we can easily find that our proposed WLE beats the counterparts and gets the best performance on SEED and DEAP (Valence).

TABLE III: Different classifiers used by WLE on DEAP (Arousal)

Method	rbf	linear
WLE(ours)	**58.361(14.793)**	57.031(10.028)

This table shows the performance of our proposed WLE on DEAP (Arousal) when using rbf and linear classifier, respectively.

Although WLE does not achieve the best performance on DEAP (Arousal), the combination of hyperparameters used by these top-performing machine learning methods can be changed according to different datasets. Specifically, for SVM we find the best result can be acquired by using a linear kernel function as the kernel function on DEAP (Valence), and

Table IV: Comparison of domain adaptation methods on different feature extractors on SEED and DEAP

Extractors TL methods	MLP			Transformer [37]			4D-CRNN [31]			WLE(Ours)		
	SEED	DEAP (V)	DEAP (A)	SEED	DEAP (V)	DEAP (A)	SEED	DEAP (V)	DEAP (A)	SEED	DEAP (V)	DEAP (A)
Baseline	79.881 (8.008)	58.938 (8.005)	55.063 (11.505)	79.526 (7.656)	56.203 (7.959)	54.344 (10.312)	40.191 (4.378)	50.414 (8.509)	49.625 (9.388)	83.022 (6.184)	59.313 (10.035)	57.031 (10.028)
DDC [2]	76.118 (8.941)	58.875 (7.656)	55.047 (11.007)	78.489 (6.964)	55.938 (8.249)	52.593 (10.733)	**44.845** **(5.023)**	51.628 (2.321)	52.431 (4.798)	81.600 (6.424)	56.875 (8.556)	55.109 (8.213)
CORAL [34]	77.630 (9.095)	59.250 (7.538)	55.078 (11.237)	80.741 (7.361)	55.969 (8.145)	53.906 (10.018)	43.460 (4.787)	50.224 (3.054)	51.033 (1.992)	83.170 (6.544)	58.062 (8.321)	55.344 (9.236)
JDA [19]	78.036 (9.033)	58.034 (8.326)	55.166 (10.383)	80.698 (6.184)	56.387 (5.856)	53.180 (7.591)	42.424 (5.069)	50.589 (4.286)	51.506 (6.228)	83.293 (5.968)	58.502 (7.557)	56.226 (11.995)
DANN [10]	77.778 (8.763)	59.218 (7.930)	55.391 (11.229)	80.593 (7.646)	56.688 (8.038)	51.781 (8.087)	41.235 (5.486)	50.304 (4.469)	51.449 (3.723)	83.230 (6.326)	58.281 (8.152)	55.469 (9.613)
MADN [28]	77.719 (8.864)	58.641 (7.999)	54.656 (11.034)	80.948 (7.270)	56.781 (7.010)	54.047 (10.334)	40.363 (4.339)	50.327 (4.412)	52.969 (5.744)	81.630 (6.750)	59.047 (8.485)	55.938 (10.698)
DAAN [43]	76.889 (8.479)	58.734 (7.979)	52.359 (8.264)	80.859 (7.603)	56.453 (7.652)	51.500 (8.479)	39.536 (3.360)	49.956 (4.371)	50.672 (4.516)	80.711 (7.395)	57.125 (8.157)	55.078 (10.804)
MSFRAN (Ours)	**80.247** **(9.123)**	**59.266** **(8.278)**	**55.297** **(9.780)**	**82.617** **(7.337)**	**59.828** **(7.884)**	**54.781** **(10.967)**	42.485 (4.249)	**52.854** **(4.746)**	**52.527** **(4.220)**	**85.481** **(5.238)**		

using a radial basis function as the kernel function on DEAP (Arousal). As shown in Table V, the linear kernel can achieve the highest performance on SEED and DEAP (Valence), but not the highest accuracy on DEAP (Arousal). Meanwhile, the rbg kernel behaves exactly opposite to the linear kernel.

In consideration of the above experiments, we define two different classifiers as follows:

$$\begin{aligned} &\mathcal{F}_{linear}\left(X\right) = W \times X + b, \\ &c_1, \quad c_2, \cdots, c_S \in \mathbb{R}^N, \\ &\sigma_1, \quad \sigma_2, \cdots, \sigma_S \in \mathbb{R}, \\ &w_1, \quad w_2, \cdots, w_S \in \mathbb{R}, \\ &\mathcal{F}_{rbf}\left(X\right) \quad = \left(\sum_{i=1}^{S} w_i e^{-\frac{\|\mathcal{X}-c_i\|^2}{2\sigma_i^2}}\right) + b, \end{aligned} \tag{14}$$

where $W \in \mathbb{R}^N$ and $b \in \mathbb{R}$. Here N is the number of features of the sample and S is the number of hidden layer nodes. As shown in Table III, if we use the rbf classifier instead of the linear classifier to rerun the experiment on DEAP (Arousal), the accuracy of ours is improved.

From the above experiments, on the one hand, we demonstrate that our feature extractor model is effective and outperforms all other related methods. On the other hand, a model can achieve excellent performance on a specific dataset. But that doesn't mean it still works well when the dataset is changed.

C. Experiments on transfer learning methods

In order to verify the effectiveness of our proposed transfer learning method, we have carried out detailed comparative experiments. Specifically, we selected several representative feature extractors (MLP, Transformer, 4D-CRNN and Wide linear extractor) as the baseline extractor respectively. Specifically speaking, MLP has one hidden layer and residual connection is used to process the input and output of MLP. In the Transformer, we used only the absolute position coding and one encoder structure. And in the 4D-CRNN, we adopted its integral structure. Then we adopted different transfer learning methods to compare with ours.

TABLE V: Hyperparameter setting of SVM

Datasets	Regularization	Kernel	Gamma	Accuracy
SEED	0.015	linear	-	**78.074(7.475)**
	0.15	linear	-	**78.074(7.475)**
	0.015	rbf	$\frac{1}{number(\mathcal{X})}$	65.630(8.349)
	1.5	rbf	$\frac{1}{number(\mathcal{X})}$	76.444(9.137)
	0.015	rbf	$\frac{1}{number(\mathcal{X})var(\mathcal{X})}$	65.630(8.349)
	1.5	rbf	$\frac{1}{number(\mathcal{X})var(\mathcal{X})}$	76.444(9.137)
DEAP(V)	0.015	linear	-	**59.297(7.348)**
	0.15	linear	-	**59.297(7.348)**
	0.015	rbf	$\frac{1}{number(\mathcal{X})}$	55.313(9.137)
	0.15	rbf	$\frac{1}{number(\mathcal{X})}$	55.313(9.137)
	0.015	rbf	$\frac{1}{number(\mathcal{X})var(\mathcal{X})}$	55.313(9.137)
	0.15	rbf	$\frac{1}{number(\mathcal{X})var(\mathcal{X})}$	55.313(9.137)
DEAP(A)	0.015	linear	-	53.125(10.606)
	0.15	linear	-	53.125(10.606)
	0.015	rbf	$\frac{1}{number(\mathcal{X})}$	**57.578(15.693)**
	0.15	rbf	$\frac{1}{number(\mathcal{X})}$	**57.578(15.693)**
	0.015	rbf	$\frac{1}{number(\mathcal{X})var(\mathcal{X})}$	**57.578(15.693)**
	0.15	rbf	$\frac{1}{number(\mathcal{X})var(\mathcal{X})}$	**57.578(15.693)**

This table shows the different hyperparameters used by SVM to achieve the best performance on different datasets, where DEAP(V) refers to DEAP (Valence), and DEAP(A) refers to DEAP (Arousal). And linear refers to linear kernel function and rbf refers to radial basis function. In addition, $number(\mathcal{X})$ represents the number of features in the input sample $\mathcal{X}$, and $var(\mathcal{X})$ represents the variance of $\mathcal{X}$.

In Table IV, we have done the comparative experiments on SEED, DEAP(Valence) and DEAP(Arousal). On the whole, we can find that applying the same domain adaptation method to different feature extractors can obtain different performances in each dataset. This means that feature extractor will have important influence on final performance. And we also find that the convergence rate of different extractors is quite different. Among the selected feature extractors, combining WLE and other domain adaptation methods can achieve more excellent results on SEED. However, on DEAP(V) and DEAP(A), it is not outstanding for the WLE-based domain adaptation methods. This is most likely due to the differences of data distribution between different datasets. For 4D-CRNN-based domain adaptation, we can recognize that although it has remarkable results in the experiments based on the same subject, it does not have great cross-subject emotion recognition capability. When different domain adaptive methods are added to 4D-CRNN, the performance will get a certain degree of improvement. On the premise of the same feature extractor, methods based on adversarial strategy are relatively difficult to adjust parameters and hard to ensure the data separability and transferability simultaneously. For the methods of measuring the distance between source and target domain, CORAL can achieve better results compared with DDC. The effects of DANN and MADN on the same extractor is not much different in EEG signal. In contrast, DAAN which absorb the advantages of DANN and MADN do not get expected effect in emotion classification. This shows that there is a great difference between image data and EEG data. Original domain adaptation methods in the image field need to be improved according to the characteristics of EEG. And in the Table IV, it is worth noting that different baseline methods suffer from a slight performance degradation when partial transfer learning methods are added. This indicates the occurrence of negative transfer. On the WLE feature extractor, MSFRAN can significantly improve classification performance by 2-3 percentage points on different datasets.

D. Ablation experiments

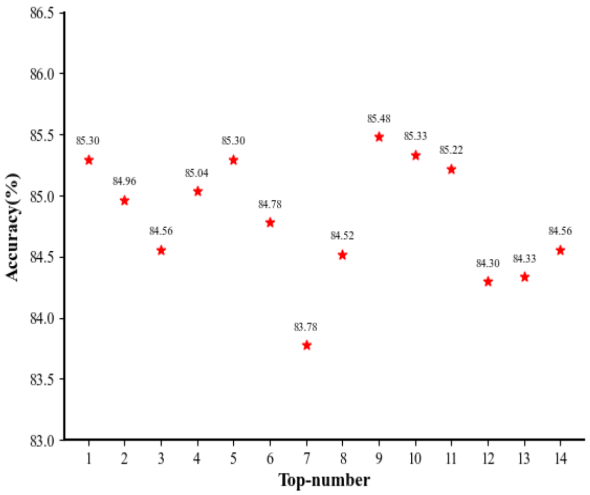


Fig. 5: The number of Top multi-domain classifiers

In the MSFRAN, we need to select the number of optimal classifiers to achieve the best classification result. Thus we carry out five times experiments based on WLE+MSFRAN for different numbers of top-classifiers, respectively. As shown in Figure 5, as the number of top-classifiers increases , the accuracy of top-classifiers presents a fluctuating change. As we can see, it's not that the more number of top-classifiers, the better accuracy. The best accuracy can be achieved when the number of top-classifiers is set to 5.

VI. Conclusion

In this paper, we mainly design a simple and effective feature extractor (WLE) and improve multi-source domain adaptation method named MSFRAN. We combine these two innovations into a new model and prove its effectiveness and superiority. This model not only learn domain-invariant features between each pair of source and target domains, but also study the relevant information between source domains and establish correlation constraint between each other. Meanwhile, it also effectively alleviates the discrepancy between multiple classifiers. Extensive experimental results have demonstrated that our model is effective and exceeds other relevant methods in cross-subject emotion recognition. In future studies, we will continue to explore new transfer learning methods to further solve the problem of differences between subjects and achieve higher performance across different datasets.

References

[1] Omid Bazgir, Zeynab Mohammadi, and Seyed Amir Hassan Habibi. Emotion recognition with machine learning using EEG signals. In *25th National and 3rd International Iranian Conference on Biomedical Engineering (ICBME)*, pages 1–5, Qom, Iran, Nov. 2018. IEEE.

[2] Karsten M. Borgwardt, Arthur Gretton, Malte J. Rasch, Hans-Peter Kriegel, Bernhard Schölkopf, and Alex J. Smola. Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(14):49–57, 2006.

[3] Jiangsheng Cao, Xueqin He, Chenhui Yang, Sifang Chen, Zhangyu Li, and Zhanxiang Wang. Multi-source and multi-representation adaptation for cross-domain electroencephalography emotion recognition. *Frontiers in psychology*, 12:809459, 2021.

[4] Xin Chai, Qisong Wang, Yongping Zhao, Yongqiang Li, Dan Liu, and Ou Bai. A fast, efficient domain adaptation technique for cross-domain electroencephalography (EEG)-based emotion recognition. *Sensors*, 17(5):1014(1)–1014(21), 2017.

[5] Xin Chai, Qisong Wang, Yongping Zhao, Yongqiang Li, Dan Liu, and Ou Bai. A fast, efficient domain adaptation technique for cross-domain electroencephalography EEG-based emotion recognition. *Sensors*, 17(5):1014(1)–1014(21), 2017.

[6] Xin Chai, Qisong Wang, Yongping Zhao, Xin Liu, Ou Bai, and Yongqiang Li. Unsupervised domain adaptation techniques based on auto-encoder for non-stationary EEG-based emotion recognition. *Computers in Biology and Medicine*, 79:205–214, 2016.

[7] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in neural information processing systems*, volume 29, Barcelona, Spain, Dec. 2016. NIPS.

[8] Paul Ekman, Wallace V. Friesen, Maureen O"Sullivan, Anthony Chan, Irene Diacoyanni-Tarlatzis, Karl Heider, Rainer Krause, William Ayhan LeCompte, Tom Pitcairn, Pio E. Ricci-Bitti, Klaus Scherer, and Masatoshi Tomita. Universals and cultural differences in the judgments of facial expressions of emotion. *Journal of Personality and Social Psychology*, 53(4):712, 1987.

[9] Bent Fuglede and Flemming Topsoe. Jensen-shannon divergence and hilbert space embedding. In *International Symposium on Information Theory, 2004. ISIT 2004. Proceedings.*, page 30, Chicago, IL, USA, Jun.-Jul. 2004. IEEE.

[10] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189, Lille, France, Jul. 2015. PMLR.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *International Conference on Neural Information Processing Systems*, pages 2672–2680, Montreal, Canada, Dec. 2014. MIT Press.

[12] Xiaoqing Gu, Weiwei Cai, Ming Gao, Yizhang Jiang, Xin Ning, and Pengjiang Qian. Multi-source domain transfer discriminative dictionary learning modeling for electroencephalogram-based emotion recognition. *IEEE Transactions on Computational Social Systems*, pages 1–9, 2022.

[13] Vipin Gupta, Mayur Dahyabhai Chopda, and Ram Bilas Pachori. Cross-subject emotion recognition using flexible analytic wavelet transform from eeg signals. *IEEE Sensors Journal*, 19(6):2266–2274, 2019.

[14] Yiming Jin, Yudong Luo, Weilong Zheng, and Baoliang Lu. EEG-based emotion recognition using domain adaptation network. In *International Conference on Orange Technologies*, pages 222–225, Singapore, Singapore, Dec. 2017. IEEE.

[15] James M. Joyce. *Kullback-Leibler Divergence*, pages 720–722. Springer, 2011.

[16] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30, 2017.

[17] Sander Koelstra, Christian Muhl, Mohammad Soleymani, Jong-Seok Lee, Ashkan Yazdani, Touradj Ebrahimi, Thierry Pun, Anton Nijholt, and Ioannis Patras. DEAP: A database for emotion analysis using physiological signals. *IEEE Transactions on Affective Computing*, 3(1):18–31, 2012.

[18] Zirui Lan, Olga Sourina, Lipo Wang, Reinhold Scherer, and Gernot R. Müller-Putz. Domain adaptation techniques for EEG-based emotion recognition: A comparative study on two public datasets. *IEEE Transactions on Cognitive and Developmental Systems*, 11(1):85–94, 2019.

[19] Jinpeng Li, Shuang Qiu, Changde Du, Yixin Wang, and Huiguang He. Domain adaptation for EEG emotion recognition based on latent representation similarity. *IEEE Transactions on Cognitive and Developmental Systems*, 12(2):344–353, 2020.

[20] Jinpeng Li, Shuang Qiu, Yuanyuan Shen, Chenglin Liu, and Huiguang He. Multisource transfer learning for cross-subject EEG emotion recognition. *IEEE Transactions on Cybernetics*, 50(7):3281–3293, 2020.

[21] Wei Li, Wei Huan, Bowen Hou, Ye Tian, Zhen Zhang, and Aiguo Song. Can emotion be transferred? ¨c a review on transfer learning for eeg-based emotion recognition. *IEEE Transactions on Cognitive and Developmental Systems*, pages 1–14, 2021.

[22] Jiyao Liu, Li Zhang, Hao Wu, and Huan Zhao. Transformers for eeg emotion recognition. *arXiv preprint arXiv:2110.06553*, 2021.

[23] Mingsheng Long and Jianmin Wang. Learning transferable features with deep adaptation networks. In *International Conference on Machine Learning (ICML)*, Lille, France, Jul. 2015. International Machine Learning Society (IMLS).

[24] Zhou Lu, Hongming Pu, Feicheng Wang, Zhiqiang Hu, and Liwei Wang. The expressive power of neural networks: A view from the width. *Advances in neural information processing systems*, 30, 2017.

[25] Yun Luo, Siyang Zhang, Weilong Zheng, and Baoliang Lu. WGAN domain adaptation for EEG-based emotion recognition. In *International Conference on Neural Information Processing – Neural Information Processing*, volume 11305, pages 275–286, Siem Reap, Cambodia, Dec. 2018. Springer.

[26] Shyam Marjit, Upasana Talukdar, and Shyamanta M Hazarika. EEG-based emotion recognition using genetic algorithm optimized multi-layer perceptron. In *International Symposium of Asian Control Association on Intelligent Robotics and Industrial Automation*, pages 304–309, 2021.

[27] Sinno Jialin Pan, Ivor W. Tsang, James T. Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2011.

[28] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. In *AAAI conference on artificial intelligence*, pages 1–8, New Orleans, Louisiana, USA, Feb. 2018. American Association for Artificial Intelligence.

[29] Souvik Phadikar, Nidul Sinha, and Rajdeep Ghosh. A survey on feature extraction methods for eeg based emotion recognition. In *Intelligent Techniques and Applications in Science and Technology*, volume 12, pages 31–45, Siliguri, India, Sept. 2019. Springer.

[30] James A. Russell. A circumplex model of affect. *Journal of Personality and Social Psychology*, 39(6):1161–1178, 1980.

[31] Fangyao Shen, Guojun Dai, Guang Lin, Jianhai Zhang, Wanzeng Kong, and Hong Zeng. EEG-based emotion recognition using 4D convolutional recurrent neural network. *Cognitive Neurodynamics*, 14(6):815–828, 2020.

[32] Lichen Shi, Yingying Jiao, and Baoliang Lu. Differential entropy feature for eeg-based vigilance estimation. In *35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 6627–6630, Osaka, Japan, Jul. 2013. IEEE.

[33] Tengfei Song, Wenming Zheng, Peng Song, and Zhen Cui. EEG emotion recognition using dynamical graph convolutional neural networks. *IEEE Transactions on Affective Computing*, 11(3):532–541, 2020.

[34] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450, Amsterdam, The Netherlands, Oct. 2016. Springer.

[35] Wei Tao, Chang Li, Rencheng Song, Juan Cheng, Yu Liu, Feng Wan, and Xun Chen. EEG-based emotion recognition via channel-wise attention and self attention. *IEEE Transactions on Affective Computing*, pages 1–12, 2020.

[36] Eric Tzeng, Judy Hoffmanand Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *Computer Science*, 2014.

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, volume 30, Long Beach Convention Center, Long Beach, Dec. 2017. Curran Associates, Inc.

[38] Gnana Keerthi Priya Veeramallu, Yamuna Anupalli, Sravan kumar Jilumudi, and Abhijit Bhattacharyya. Eeg based automatic emotion recognition using EMD and random forest classifier. In *International Conference on Computing, Communication and Networking Technologies*, pages 1–6, Honolulu, Hawaii, USA, Feb. 2019. IEEE.

[39] Anand Venkatraman, Brian L. Edlow, and Mary Helen Immordino-Yang. The brainstem in emotion: A review. *Frontiers in Neuroanatomy*, 11:15(1)–15(12), 2017.

[40] Hongliang Yan, Yukang Ding, Peihua Li, Qilong Wang, Yong Xu, and Wangmeng Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *IEEE Conference on*

Computer Vision and Pattern Recognition (CVPR), pages 1–10, Hawaii, America, Feb. 2017. IEEE.

[41] Zhong Yin, Yongxiong Wang, Li Liu, Wei Zhang, and Jianhua Zhang. Cross-subject EEG feature selection for emotion recognition using transfer recursive feature elimination. *Frontiers in Neurorobotics*, 11:19(1)–19(16), 2017.

[42] Chaohui Yu, Jindong Wang, Yiqiang Chen, and Meiyu Huang. Transfer learning with dynamic adversarial adaptation network. In *IEEE International Conference on Data Mining*, pages 778–786, Beijing, China, Nov. 2019. IEEE.

[43] Chaohui Yu, Jindong Wang, Yiqiang Chen, and Meiyu Huang. Transfer learning with dynamic adversarial adaptation network. In *IEEE International Conference on Data Mining*, pages 778–786, Beijing, China, Nov. 2019. IEEE.

[44] Weiwei Zhang, Fei Wang, Yang Jiang, Zongfeng Xu, Shichao Wu, and Yahui Zhang. Cross-subject EEG-based emotion recognition with deep domain confusion. In *International Conference on Intelligent Robotics and Applications*, pages 558–570, Shenyang, China, Aug. 2019. Chinese Academy of Sciences.

[45] Wei-Long Zheng and Bao-Liang Lu. Investigating critical frequency bands and channels for EEG-based emotion recognition with deep neural networks. *IEEE Transactions on Autonomous Mental Development*, 7(3):162–175, 2015.

[46] Yongchun Zhu, Fuzhen Zhuang, and Deqing Wang. Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5989–5996, Hawaii, USA, Jan.-Feb. 2019.

PLACE
PHOTO
HERE

Michael Shell Biography text here.

John Doe Biography text here.

Jane Doe Biography text here.