Journal of Network Intelligence
Taiwan Ubiquitous Information
©2022 ISSN 2414-8105 (Online)
Volume **7**, Number **3**, August **2022**

Unsupervised Domain Adaptation Using Attention Network and New Loss Function

Li-Quan Zhao

Key Laboratory of Modern Power System Simulation and Control & Renewable Energy Technology, Ministry of Education
Northeast Electric Power University
Jilin 132012, China
zhaoliquan@neepu.edu.cn

Si-Ying Zhou

Key Laboratory of Modern Power System Simulation and Control & Renewable Energy Technology, Ministry of Education
Northeast Electric Power University
Jilin 132012, China
865660265@qq.com

Zi-Ming Teng

College of information and communication
Jilin University
Changchun, 130012, China
tengziming2002@163.com

Zi-Cong Jiang

Graduate school of Information Science and Electrical Engineering
Kyushu University
Fukuoka 819-0395, Japan
jiangzicong1234@gmail.com

Yan-Fei Jia*

School of Electrical and Information Engineering
Beihua University
Jilin 132013, China

*Corresponding author: Yan-Fei Jia

Received March 11, 2022, revised April 22, 2022, accepted June 1, 2022.

Abstract. *An unsupervised domain adaptation method is proposed based on the stepwise adaptive feature norm method to improve transfer learning accuracy. The feature extraction network consists of channel attention and spatial attention modules, which enhance feature extraction ability. To reduce the influence of dimension change on channel weight, a convolution layer with the same kernel size is used to replace the fully connected layer in the bottleneck layer used in the common channel attention module. In addition, a smooth loss function is also proposed. The curve of the proposed function is smoother than the L2 function that is the least square error function. The proposed function's gradient changes more slowly than the L2 function. The proposed function is used as the activate function of classification prediction regression to reduce the risk of gradient explosion. Compared with a selection of existing methods (Resnet50, conditional domain adversarial networks method, deep adaptive network method, joint adaptation network method, domain adaptation neural network method, and stepwise adaptive feature norm method), the proposed transfer learning method has the highest accuracy than other methods on different datasets for all tasks.*
Keywords: Unsupervised domain adaptation; Artificial intelligence; Transfer learning; Adversarial network

1. **Introduction.** Although artificial intelligence (AI) technology has been widely used in real applications, the conventional artificial intelligence method requires many training samples for the training of the neural networks model [1]. In practice, this arrangement is expensive, and labeling training samples is an exhaustive process that can yield defective samples in some applications (such as defective insulator samples and defective samples of industrial products) [2]. Further, for each AI task the model parameters must be structured on related tasks. To solve this problem, a transfer learning method has been proposed. The proposed method can transform previously learned knowledge into a new model. Domain adaptation network model is one example of a special transfer learning method with higher accuracy. It firstly maps the data in the target domain and source domain to feature space. Secondly, it adjusts the parameters in the source domain to the target domain by transforming the objective function to reduce the distance between features in different domains. Domain adaptation is widely used in target re-identification [3], medical image segmentation [4], machinery fault diagnosis [5].

There are three parts to the domain adaptive network structure: feature extraction, source domain classification, and domain discrepancy metric. The common methods of domain discrepancy are adversarial generation network and maximum mean discrepancy (MMD). Although the adversarial generation network method can obtain a better domain transfer effect, the whole network is required to be trained over a long time period due to the long adversarial time between the generator and discriminator. The overall model of the MMD method is lightweight, and the training time is short, but the domain discrepancy is dictated by selected inter-domain measurement indicators, so a high level of accuracy cannot be guaranteed.

Adaptive Feature Norm (AFN) method [6] introduces the difference of feature norm between target and source domains into the loss function, thus eliminating the errors affecting the domain adaptation network model. It uses the feature norm adaptation method to reduce the transfer effect of domain shift. The rationale of this method is that the larger the feature norm, the easier the classification. However, the adaptive feature norm method only considers the distribution of the feature norm. It ignores the effects of feature selection on the size of the feature norm during the feature extraction stage, as well as the influence of outlier samples on the overall model for choosing L2 function in regression. Both weaknesses can undermine the classification accuracy. In order to extract

more effective feature information and reduce the effect of outlier samples, we propose an improved unsupervised domain adaptation based on the adaptive feature norm method, which can improve classification accuracy.

The main contributions of this paper are summarized as follows:

1. We design a channel attention module that consists of two global maximum pooling layers, two global average pooling layers, and a shared convolution connected layer. We introduce the designed channel attention module and spatial attention module to the backbone. The channel attention module can extract more useful feature information that is related to classification tasks, thus suppressing redundant information.

2. We propose a new activate function, one which is smoother than incumbent functions. Compared with the L2 function that is used in the AFN method, this enhanced function offers better robustness, with a gradient that descends more slowly. We use the proposed function to construct a new loss function in order to measure the feature norm discrepancy between the target domain and the source domain. It can solve the problem of possible gradient explosion associated with the loss function in the conventional method.

In this section, we have provided an introduction to transfer learning and our contributions to the field. In section 2, we introduce the significant developments related to domain adaptation methods. In section 3, we describe our proposed unsupervised domain adaptation method. In section 4, we analyze and discuss the simulation results. In section 4, the conclusions are summarized.

2. **Related Work.**

There are three categories of domain adaptation methods that are semi-supervised domain adaptation method, weakly-supervised domain adaptation method, and unsupervised domain adaptation method, according to whether the samples in the target domains and source domain have labels. Our proposed method can be described as an unsupervised domain adaptation method. Therefore, we mainly introduce unsupervised domain adaptation methods in the following. Unsupervised domain adaptation methods, each of which is based on the adversarial network, can be further divided into two groups: methods based on adversarial discrimination networks and ones based on adversarial generation networks.

The method based on adversarial discrimination employs feature-level domain adaptation. It transfers the measured differences between different domains to the feature space by adding an antagonistic target to the domain discriminator in order to realize feature confusion between the target domain and the source domain. Bousmalis et al. [7] designed the domain adaptation neural network (DANN) model. In this model, the samples of the target domain and the source domain are mapped in the same manner, and then the source discriminator and the target discriminator by means of the cross-entropy loss function. This can make the feature extractor extract the domain invariant features in order to classify the input samples in the target domain. However, the DANN method cannot extract features satisfactorily where there exists large variation between features, which in turn produces a reduction of many domain invariant features. Tzeng et al. [8] proposed adversarial discriminative domain adaptation (ADDA), which is based on the DANN method. ADDA introduces the concept of an adversary into the sample mapping process of the target domain so that different domains can be distinguished by the discriminator. It discriminates the target domain samples by identifying them as source domain samples to realize data distribution alignment between the target domain and the source domain. However, the ADDA method does not take into account the multimodal structure of data distribution and cannot ensure good alignment of the target domain and source domain. Long et al. [9] designed the conditional domain adversarial network

(CDAN). This employs the multilinear conditioning method to solve the multimodal distribution problem, and it adjusts the transferability of the classifier by using the classifier prediction approach.

Adversarial generation is a means of pixel-level domain adaptation, which learns common features between different learning domains from images generated of the source domain and images of the source domain. Hoffman et al. [10] proposed the cyclic consistent adversarial domain adaptation (CyCADA) method. It is important to note that the domain adaptation of CyCADA is realized at both the feature level as well as at the pixel level. The method retains the consistency of global and local feature distribution of the transformed source domain images by optimizing the pixel level reconstruction loss function and the semantic loss function of the labels for predicting the domain label based on domain data. Li et al. [11] devised an improved transfer network with higher accuracy based on CyCADA by combining the cyclic consistency loss method with CADA methods. To get features that are domain invariant, it swaps the features of the target and source domains by making use of the predicted covariance that is obtained by the classifier and introducing feature converters into conditional adversarial networks.

Existing adversarial methods are concerned with aligning the domain distribution in the potential space; however, this undermines the generative process of capturing the alignment distribution. Zhou et al. [12] proposed the deep cycle autoencoder (DCA) method, which uses the annotation data and trains the classifier by reconstructing the source image capture alignment to better generalize the target sample using the source classifier. The unsupervised domain adaptation approach is based on distance measurement. It maps some different metrics between the target domain and the source domain into the same regenerated Hilbert space and then minimizes the metrics between the domains to reduce the domain offset of the target domain and the source domain, and thus aligns the data distribution of the target domain with the data distribution of the source domain. Zhuang et al. [13] used the symmetric form of KL divergence to measure the difference between domains. Data distributions between the source domain and target domain are more similar when KL divergence value is low. Pan et al. [14] used KL divergence, whereby they reshaped the distance from the prototype of each class for classification and trained via minimization of KL divergence among the three types of data output distribution (source domain, target domain, and source-target). Shen et al. [15] proposed a new unsupervised domain adaptation approach named WDGRL, which calculates the difference between domains using the Wasserstein distance and measures the difference between distributions by determining the basic geometric properties of probability space. WDGRL solves the problem of the gradient disappearing during the training process when the distance between domains is large. However, the generalization effect of WDGRL is less than satisfactory if the data samples have structured output space. For this reason, Lee et al. [16] proposed sliced Wasserstein dispersion (SWD) to detect target samples far away from the source domain and align the data distribution through end-to-end training. It is widely used to measure a distribution's differences between target domains and source domains through the maximum mean discrepancy(MMD) [17]. Pan et al. [18] proposed the Transfer Component Analysis (TCA) method by calculating the edge distribution of MMD learning across the domain transfer feature. Li et al. [19] proposed the locality preserving joint transfer for domain adaptation (LPJT) method, introduced manifold learning based on TCA, and minimized the edge distribution of the source domain and the conditional distribution of the target domain by training a feature change matrix. Long et al. [20] introduced the idea of conditional distribution into the TCA method and proposed Joint Distribution Adaptation (JDA) to reduce inter-domain distance by hitting the pseudo-labels in the target domain data.

The methods listed above employ single kernel MMD; however, a single kernel-based solution does not adequately address the domain adaptation problem. Long et al. [21] proposed the deep adaptive network (DAN), which adds MMDs to the last three fully connected layers of the network. This MK-MMD weighted calculation solves the one-sided problem of the single fixed kernel and makes better use of the migration characteristics of the deep network. However, DAN does not take into account the migration of conditional distribution. Long et al. [22] proposed the joint adaptation network (JAN), which fully utilizes the relationship between different layers and avoids the relationship assumption between edge distribution and conditional distribution by maximizing the mean discrepancy of the joint distribution. Li et al. [23] proposed a graph-based landmark selection algorithm, DTN, that focuses on sample geometric relationships by learning multiple mappings and mapping high-dimensional features into a shared subspace. Thus it does not require pre-processing of the data, a cumbersome process upon which previous methods have relied. However, the DTN method only uses globally varying geometric fields and fixed unchanged features, and so the scope for real-world application in current technology is restricted. Kim et al. [24] proposed a recursive transformation network (RTN) to directly estimate the transformation between image pairs and to refine transformation estimation and feature representation in a recursive way, thus enhancing matching quality. Li et al. [25] proposed a new distance activate named maximum density scatter, which is applied to adversarial domain adaptation. Their proposed method measures the distribution difference between the two domains by converging the adversarial network and maximum density divergence to optimize an additional loss function.

Xu et al. [6] in a series of experiments demonstrated that the size of the feature norm in the target domain is smaller than the size of the feature norm in the source domain and that this disparity is a major cause of model degradation. None of the above methods are designed to monitor the influence of the size of the feature norm on the domain shift; therefore, current metrics probably do not provide an accurate description of domain migration. Xu et al. proposed a maximum mean feature norm discrepancy (MMFND) method, which is based on MMD. A better transfer is achieved by minimizing the mean norm distance between the target domain and the source domain. To reduce the difference in terms of feature norm between the target and source domains, they proposed a stepwise adaptive feature norm (SAFN) method, which offers improved accuracy by increasing the feature norm of the target domain. However, the SAFN method does not account for the effects of feature selection on the size of the feature norm during the feature extraction stage or the effects of outlier samples on the overall model.

3. **Proposed Unsupervised domain adaptation.**

3.1. **Loss Function.**

In the stepwise adaptive feature norm method, the L2 loss function is utilized to limit the discrepancy of the feature norm between the target domain and the source domain. The loss function of the stepwise adaptive feature norm method is expressed as follows:

$$L_D = \frac{\lambda}{n_s+n_D} \sum_{x_i \in D_s \cup D_t} L_d \left(h(x_i;\theta_1) + \Delta r, h(x_i;\theta_2) \right) \tag{1}$$

where λ is a hyperparameter; n_t is the number of samples that are unlabeled within the target domain; n_s is the number of labeled samples that are unlabeled within the target domain; the D_t expresses the target domain; the D_s expresses source domain; x_i represents a sample within the domain of the target or source; $h(x)$ represents the mapping function determined by the depth representation module and L2 norm; L_d represents the

L2 distance; θ_2 is current iteration's model parameter; θ_1 is the current iteration's model parameter; As a positive scale scalar, Δr is utilized to gradually increase the characteristic norm of the target domain.

The L2 function and its derivative are expressed as follows:

$$L_2(x) = x^2 \tag{2}$$

$$\frac{dL_2(x)}{dx} = 2x \tag{3}$$

where x expresses the distance between the domains of the target and source, between the previous iteration and the current iteration, the SAFN method uses the gradient descent algorithm to optimize the loss function; this is based on the L2 function. When x is small, the gradient of the L2 loss function gradually decreases so that the trained model can easily obtain the optimal solution. By contrast, when x is large, the derivative of the L2 function descends too fast, leading to gradient explosion, thus undermining the stability and accuracy of the network. Further, if x is an outlier sample that is larger than 1, the output of the L2 function will be much too larger than the input and, therefore, the function will be more sensitive to outlier samples.

To solve this problem, a new smooth L2 function is proposed. The proposed smooth L2 function and its derivative are expressed as follows:

$$SL_2(x) = \begin{cases} 0.5x^2, |x| \leq 1 \\ |x| \ln |x| + 0.5, |x| > 1 \end{cases} \tag{4}$$

$$\frac{SL_2(x)}{dx} = \begin{cases} x, |x| \leq 1 \\ |sign(x) \times [\ln(|x| + 1)], |x| > 1 \end{cases} \tag{5}$$

The proposed smooth L2 function is a piecewise function. When x (that is, the distance or difference between the domains of the target and source, between the previous iteration and the current iteration) is smaller than 1, we use $0.5x^2$ as the output of the smooth L2 function. $0.5x^2$ is almost the same as the L2 function. To the reduce the sensibility of the loss function to outlier samples, we use a smoother function when $|x| > 1$, whereby we propose to use $|x| \ln(|x|) + 0.5$ as the output of the smooth L2 function. When $|x| > 1$, the value of $\ln(|x|)$ is smaller than $|x|$ and variation of is also smaller than x. The absolute values of gradient of $\ln(|x|)$ and $|x|$ are $1/|x|$ and 1 when $|x| > 1$, respectively. Therefore, the absolute value of gradient of $\ln(|x|)$is smaller than the absolute value of gradient of $|x|$ when $|x| > 1$. This means that the change of $ln(|x|)$is more slowly than $|x|$ when $|x| > 1$. The change of $|x| \ln(|x|)$ is also more slowly than $|x|^2$ when $|x| > 1$. Therefore, we use $|x| \ln(|x|)$ instead of x^2 that was used in the original L2 function to reduce the effectiveness of outliers.

The L2 function and our proposed smooth L2 function and their derivatives are shown in Figure 4. The curve of the proposed function is gentler than the L2 function and, therefore, the proposed function has better robustness than the L2 function. In the back-propagation process of the network, according to the chain derivation rule, the derivative is obtained by the products of initializing weight and multiple gradients of the activation function. If the initialized weight and gradient of activation function are larger, the obtained derivative is also larger. Therefore, the gradient of activation function is larger may lead to a larger derivative in backpropagation, and a larger derivative will greatly update the weight that causes instability of network training. The gradient of the proposed function descends more slowly than the L2 function, thus lowering the risk of gradient explosion and improving the accuracy of the network.

The loss function based on the proposed smooth L2 function for improving the SAFN method is expressed as follows:

$$L(x_i^s, y_i^s, \theta_g, \theta_f, \theta_y) = \frac{1}{n_s} \sum_{(x_i, y_i) \in D_s} L_y(x_i, y_i) + \frac{\lambda}{n_s + n_t} \sum_{x_i \in D_s \cup D_t} SL_2(h(x_i; \theta_1) + \Delta r, h(x_i; \theta_2)) \tag{6}$$

where x_i^s and y_i^s are images and labels in the source domain corresponding to the ith input image, respectively; x_i represents a sample in the target domain or source domain, and y_i denotes the source domain label of a sample; L_y represents the source classification loss function obtained through the softmax function; D_t and D_s represents the target domain and source domain, respectively; and, θ_g, θ_f, θ_y are parameters of the feature extraction network, penultimate fully connected layer and the last fully connected layer, respectively.

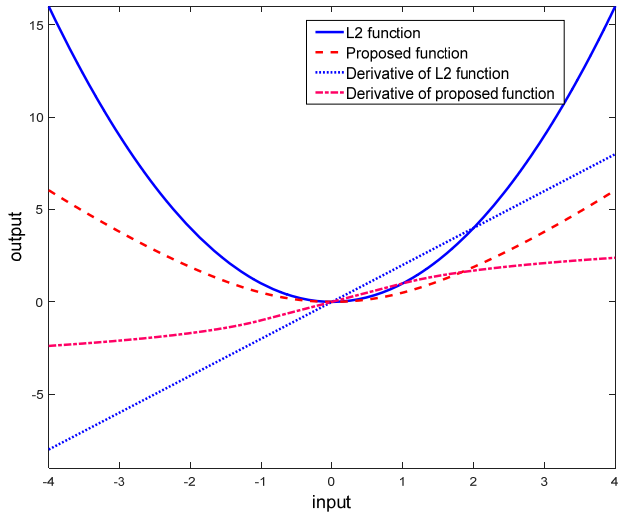

FIGURE 1. Input-output curves of the four activate functions

3.2. **Feature Extraction network.** The SAFN method uses the residual network to extract features. It uses the relationship between local features to update extracted features. In residual networks, because the convolution layers and pooling layers operate on the feature maps of all channels, in the same way, the feature maps of all channels receive the same degree of attention. This means that the extracted features related to the classification task and extracted features unrelated to the classification task each receive the same degree of attention. This setting will affect the accuracy of classification. If a featured network focuses more attention on the feature maps that contain more effective classification information and comparatively less attention on feature maps that contain less useful classification information, the feature extraction ability of the feature network will be improved. Based on this idea, we introduce the attention mechanism to the feature extraction network of the SAFN method. The attention mechanism is designed to focus more attention on more important features of the object. It emphasizes where learning is required and where data can be ignored. Therefore, it can enable feature extract networks to extract more useful feature information from images. The attention mechanism consists of the channel attention mechanism and the spatial attention mechanism. The former mechanism can make the feature network commit different levels of attention to feature

maps of different channels by means of weight adjustments. If the feature map contains more useful information on classification, it will be assigned a larger weight.

The common channel attention modules use two fully connected layers and a bottleneck layer to calculate the weight. The bottleneck layer is used to reduce the dimension of the first connected layer. Although it can reduce the complexity of the network, it also blurs the relationship between channels and corresponding weightings. The values of weights are decided by the kernel size of one-dimensional convolutions that are used in the bottleneck layer. The kernel sizes are obtained by computing a large number of hyperparameters. The number of hyperparameters is the product of the kernel size of one-dimensional convolution and the number of channels. The number of hyperparameters is larger, and their values are changed with each variation in the number of channels. Therefore, a large number of hyperparameters directly affect weight, rendering inefficient the process of capturing the information interleaving between channels via dimensionality reduction.

To solve this problem, we have designed an improved channel attention module (shown in Figure 2) inspired by the CBMA method [27] and ECA-Net [28] , which consists of two global average pooling layers and two global maximum pooling layers, and a shared connected layer. The first global average pooling layer extracts the average value of the feature map in each channel and provides feedback on each pixel on the feature map. The first global maximum pooling layer is used to extract the maximum value of the feature map in each channel. For maximum global pooling, only the pixel with the largest response in the feature map has gradient feedback in gradient backpropagation. This shared one-dimensional convolution layer is used to capture the interaction information between adjacent k channels and generate a feature map with the same size as the original number of channels [28]. Using this one-dimensional convolution layer, we avoid the non-correspondence between channel and weight caused by dimensionality reduction and thus lower the number of calculations and parameters.

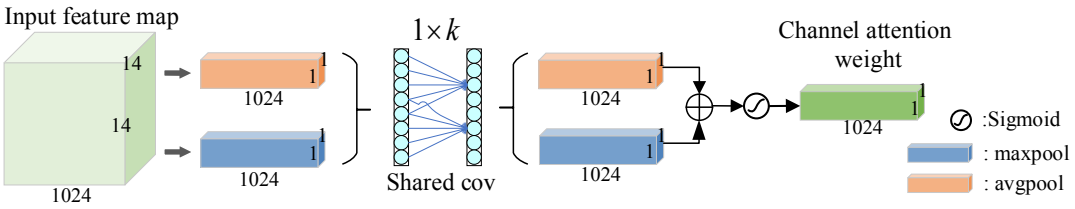

FIGURE 2. Proposed channel attention module

The shared connected layer is composed of a one-dimensional convolution with the same kernel size k, which in turn is adaptively determined by the function of the channel dimension. k can be expressed as follows [28]:

$$k = \psi(c) = \left| \frac{\log_2(c)}{2} + \frac{1}{2} \right|_{odd} \tag{7}$$

where c is the number of channels, and $\left| \frac{\log_2(c)}{2} + \frac{1}{2} \right|_{odd}$ denotes the nearest odd number to $(\frac{\log_2(c)}{2} + \frac{1}{2})$. The shared convolution layer is used to generate a feature map with the same size as the original number of channels. In contrast with the conventional channel attention module, ours uses the shared connected layer instead of the two fully connected layers used in the conventional channel attention module. In the traditional method, it is indirect correspondence relation between the channels and weights caused by descending dimension operation. The weight is decided by the kernel size of a bottleneck. The

number of parameters that are used to decide the kernel size is decided by the number of the one-dimension convolution kernel and channel. It is direct correspondence relation between the channels and weights by using our proposed shared connected layers. The weight is directly decided by kernel size. If the number of channels is expressed as c and the number of the kernel is k, the number of parameters of the traditional method and our proposed method is $k \times c$ and k, respectively. Therefore, the number of parameters is reduced from $k \times c$ to k, and so dimension reduction for the two shared connected layers can be avoided.

We firstly extract texture features and background features of each channel by the maximum global pooling and average pooling operation, respectively. Secondly, we use the one-dimension convolution to fuse texture features of adjacent channels and background features of adjacent channels, respectively. The stride of one-dimension convolution is one, and the kernel size of one-dimension convolution is k, so the features of k adjacent channels (channel i to channel $i + k$) are fused. (For example, It firstly fuses the features of channel 1 to channel k. Secondly, it fuses the features of channel 2 to channel k+1.). Each fused feature contains features of k adjacent channel. Therefore, it can effectively capture cross-channel interaction information and thus improve accuracy. The second global average pooling layer and global maximum pooling layer are used to extract the average value of the feature map and maximum value of the feature map in each channel, respectively. Finally, we can obtain the channel weight by using the active function to process the sum of two pooling layers. Therefore, different channels may have different channel weights. If a feature map in one channel is necessary for the classification task, then the channel weight will be larger and more attention shall be directed to the feature map.

The channel attention mechanism only focuses on the differences between feature maps of different channels and does not consider each pixel point. The same pixel point in different feature maps, therefore, receives the same attention. Therefore, the pixel point that is more important for the classification task receives the same attention as any other pixel point. The spatial attention mechanism focuses on pixel differences for the feature map. It can assign a larger weight to the pixel, which is considered more important for classification tasks on the feature map. Therefore, the pixels that are more useful for classification will receive more attention on the feature map. The spatial attention module is shown in Figure 3. This consists of a global maximum pooling layer, global average pooling layer, convolution layer, and sigmoid for activating the function. The global maximum pooling layer and global average pooling layer are used to extract the maximum feature value and average feature value of pixels in different channels, respectively. The convolution layer is used to compute the weight of each pixel. The spatial attention module varies the weight of the pixel according to the pixel importance as regards classification. Therefore, the module can extract more effective features. The network structure of the proposed method is shown in Figure 4.

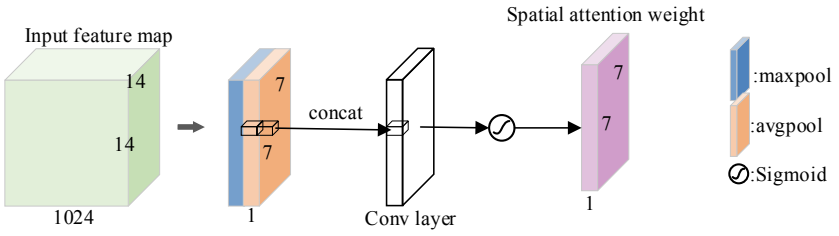

FIGURE 3. The spatial attention module

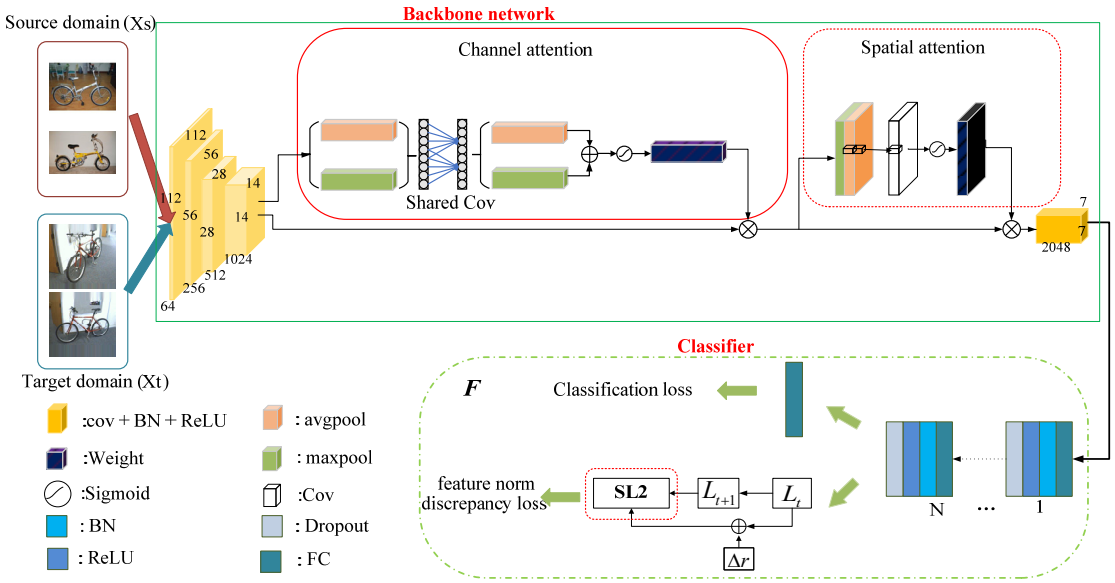

FIGURE 4. Proposed network structure

The proposed method's network structure (shown in Figure 4) is composed of the backbone network and classifier. x_t and x_s, which are the images of the target and source domains, respectively, serve as the network input. The backbone network, which is used to extract feature information, consists of a channel attention module, convolution layer, Batch Normalization, ReLu activate function layer, and spatial attention module. In order to extract more accurate feature information from the feature map, we introduce the proposed channel attention module and spatial attention module into the backbone network. The classifier network is used to realize classification with N layers, each of which is organized in the fully connected layer, Batch Normalization layer, ReLU activate function-Dropout layer order. The last fully-connected layer with softmax operation is used to calculate classification probabilities. The output of the final classification is used to construct the classification loss, that is, one of all losses. An adaptive layer is added before the last fully connected layer to compute the distance between the target domain and the source domain. We use the proposed smooth L2 function, whereby SL2 is the activate function (instead of the L2 function), that is used in the original loss function so as to avoid gradient explosion. The output of SL2 is used to construct the feature norm discrepancy loss in order to determine the distance between the target domain and the source domain. The feature norm discrepancy loss and classification loss together are treated as a complete loss.

4. **Experimental results and analysis.** We follow a standard protocol in which the source domain samples have labels, and the target domain samples have no labels. We conducted experiments using three data sets: Office-31, imageCLEF-DA, and Office-Home. In an effort to ensure a fair comparison with our method, we have selected the same hyperparameter set as the competitor's method. Our proposed method is parameter-free. Therefore, all parameters are set in the same way as in the SAFN method. We select training times (epoch) according to the relationship between accuracy and epoch. On Office-31 and image CHEF-DA datasets, the curves of all methods are gentle at 60. On Office-Home dataset, the curves of all methods are gentle at 120. Therefore, we set the training times to 60 for using Office-31 and imageCLEF-DA datasets to train all models

and 120 for using Office-Home dataset to train all models. Each dataset contains many sets. For example, the Office-31 dataset contains three sets that are A(Amazon) set, W(Webcam) set and D(DSLR) set. The numbers of sample A, W, and D are 958, 295 and 157, respectively. In the training process on Office-31, we firstly randomly select A and W set to train the network model. The A set with label is used as source domain set and W set without label is used as target domain set. Although the samples of W set have label, we discard the labels when W set is used as target domain set. The source domain and target domain are used as the model input to train the model. Then, we use the W set with label as test set to test the performance of trained model for A to W. Secondly, we use W set with label is used as source domain set and A set without label is used as target domain set. The source domain and target domain are also used as the input of model to train the model. Then, we use the A set with label as test set to test the performance of trained model of W to A. Based on above process, we can train and test the model for A to D, D to A, W to D and D to W.

4.1. **Heat map of attention mechanisms.** We randomly select six pictures from the data sets and use the Grad-CAM method to generate a heat map. The importance of the pixel points in terms of classification is expressed by the color depth in the heat map. The greater the color depth of the pixel point, the more important the pixel point. For the sake of brevity, we name the method based on our proposed channel attention module as CSAFN and the method based on our proposed channel attention module and spatial attention module as CSSAFN. The Grad-cam visualization results are shown in Figure 5. The images of the rows from the first to the fourth are original images, heat maps generated by SAFN, CSAFN and CSSAFN method, respectviely. In Figure 5 (a), the bird is the object that needs to be detected. The SAFN method not only extracts the features of the bird but also extracts the features of surrounding branches. Excessive useless features will yield errors during the process of image classification. Compared with the SAFN method, CSAFN and CSSAFN methods reduce the feature extraction of the surrounding environment more effectively and extract the features of the bird more accurately. In Figure 5 (b), the flowers are the objects needing to be detected. The SAFN method only extracts a few flower features located at the left of the image, and many of the extracted features do not facilitate flower detection. The CSAFN method successfully extracts the flower features located at the center of the image. The CSSAFN method extracts the flower features located at the bottom right-hand of the image. In Figure 5 (c), flowers are again the objects which must be detected. Although the flower features are extracted by the SAFN method, more features about branches are also extracted. Compared with the SAFN method, the CSAFN method and CSSAFN method pay more attention to flower features. In Figure 5 (d), the object of interest is the television. The SAFN method not only extracts the television feature but also extracts the features of land, rivers, and the objects that are shown on the television. Compared with the SAFN method, the CSAFN method and CSSAFN extract fewer valuable features from the rivers and other objects displayed on television. Excessive useful features can directly undermine object detection. In Figure 5 (e), the objects of interest are the people and horses. The SAFN method doesn't extract the horse feature. Figure 5 (f) shows the keyboard's object of interest. The CSAFN method pays more attention to removing desk features than SAFN method and our proposed method. The CSSAFN method pays more attention to monitors and keyboards than others. The above analysis indicates that the proposed attention module enables the feature extraction network to focus more on the object of interest and less on irrelevant objects. Therefore, the feature extraction network based on the proposed attention module can extract more relevant features.

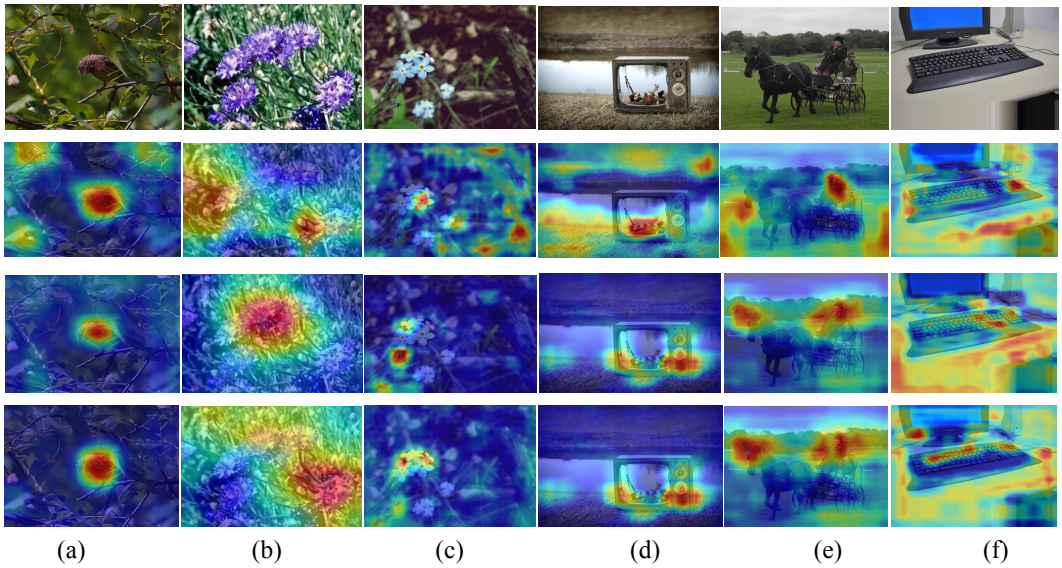

FIGURE 5. Grad-cam visualization images

4.2. **Accuracy.** As mentioned, the acronym of the improved SAFN method based on our proposed smooth L2 function is LSAFN; the acronym of the method based on our proposed smooth L2 function and proposed channel attention module is LCSAFN; and the acronym of the method based on our proposed smooth L2 function, channel attention module, and spatial attention module is LCSSAFN. LCSSAFN is also our complete proposed method. We compare our proposed methods with the following existing methods: LSAFN, LCSAFN and LCSSAFN with SAFN [6], DANN [7], CDAN [9], DAN [21], JAN [22], and Resnet50 [26]. Table 1 details the accuracies of different domain adaptive methods with the Office-31 dataset. The LCSSAFN method has the highest accuracy for all transfer tasks, followed by the LCSAFN method and LSAFN method. The proposed three methods still attain higher accuracy results than others for the average accuracy. The SAFN has the highest average accuracy of all methods, except for our proposed methods. Compared with the SAFN method, for tasks W to D, D to W, D to A, A to D, A to W, W to A, with LSAFN the accuracy increased by 0.2%, 0.2%, 2.1%, 2.6%, 1.6%, and 0.9%, respectively; with LCSAFN the accuracy increased by 0.2%, 0.5%, 2.9%, 3.7%, 1.8% and 2.2%, respectively; and, with LCSSAFN the accuracy increased by 0.2%, 0.5%, 2.9%, 4.2%, 2.1% and 2.8%, respectively. The highest average accuracy of our proposed method (87.8%) is 2.1% higher than the SAFN method (85.7%) and 2.9% higher than the CDAN method (84.9%).

Table 2 shows the accuracies of different domain adaptive methods with the ImageCLEF-DA dataset. The LCSSAFN method still has the highest accuracy for all transfer tasks, followed by the LCSAFN method and LSAFN method. The three proposed methods still attain higher accuracy than others regarding average accuracy. SAFN has the highest average accuracy, except for our proposed methods. Compared with the SAFN method, for tasks P to C, C to P, P to I, I to P, C to I, I to C, with LSAFN the accuracy increased by 0.6%, 0.9%, 0.6%, 0.4 %, 0.5%, and 0.7%, respectively; for LCSAFN the accuracy increased by 0.7%, 0.4%, 0.8%, 0.5%, 0.4% and 0.6%, respectively; and with LCSSAFN the accuracy increased by 1.2%, 2.7%, 1.7%, 3.3%, 2.4% and 2.1%, respectively. The highest average accuracy of our proposed method (90.4%) is 2.3% higher than the SAFN method (88.1%) and 3.3% higher than the CDAN method (87.1%).

TABLE 1. Accuracies of different methods on office-31 dataset.

Task	W to D	D to W	D to A	A to D	A to W	W to A	Avg
Resnet50	99.3	96.7	65.2	68.9	68.4	60.7	76.1
JAN	99.8	97.4	68.6	84.7	85.4	70.0	84.3
DAN	99.6	97.1	63.6	78.6	80.5	62.8	80.4
CDAN	100	98.2	70.1	89.8	83.1	68.0	84.9
DANN	99.1	96.9	68.2	79.7	82.0	67.4	82.2
SAFN	99.8	98.4	69.8	87.7	88.8	69.7	85.7
LSAFN	100	98.6	71.9	90.3	90.4	70.6	87.0
LCSAFN	100	98.9	72.7	91.4	90.6	71.9	87.5
LCSSAFN	100	98.9	72.7	91.9	90.9	72.5	87.8

TABLE 2. Accuracies of different methods on ImageCLEF-DA dataset.

Method	P to C	C to P	P to I	I to P	C to I	I to C	Avg
Resnet50	91.2	65.5	83.9	74.8	78.0	91.5	80.8
JAN	91.7	74.2	88.0	76.8	89.5	94.7	85.8
DAN	89.8	69.2	82.2	74.5	86.3	92.8	82.5
CDAN	93.5	74.5	90.6	76.7	90.5	97.0	87.1
DANN	91.5	74.3	86.0	75.0	87.0	96.2	85.0
SAFN	94.7	77.0	91.7	78.0	91.1	96.2	88.1
LSAFN	95.4	77.4	92.5	78.5	91.5	96.8	88.7
LCSAFN	95.4	78.6	92.6	78.9	92.4	97.2	89.2
LCSSAFN	95.9	79.7	93.4	81.3	93.5	98.3	90.4

Table 3 details the accuracies of different domain adaptive methods with the Office-Home dataset. The LCSSAFN method still attains the highest accuracy for all transfer tasks, followed by the LCSAFN method and LSAFN method. The proposed three methods still have better accuracy than others regarding average accuracy. SAFN has the best average accuracy, except for our proposed methods. Compared with the SAFN method, for tasks Ar to Pr, Pr to Ar, CI to Ar, Ar to CI, Rw to Ar, Ar to RW, CI to Pr, Pr to CI, CI to RW, RW to CI, Pr to Rw, Rw to Pr with LSAFN the accuracy is increased by 0.9%, 0.5%, 0.7%, 1.1%, 1.3%, 1.2%, 1.5%, 2.5%, 0.5%, 1.6%, 1.5%, and 1.2%, respectively; with LCSAFN the accuracy is increased by 1.2%, 1.2%, 2.0%, 1.2%, 1.4%, 1.5%, 2.1%, 2.3%, 1.3%, 2.3%, 1.7% and 1.1%, respectively; and, with LCSSAFN the accuracy is increased by 2.7%, 2.5%, 2.2%, 2.5%, 2.2%, 1.9%, 2.1%, 2.9%, 2.2%, 2.6%, 2.6% and 2.2%, respectively. The highest average accuracy of our proposed method (69.7%) is 2.4% higher than that of the SAFN method (67.3%).

TABLE 3. Accuracies of different methods on Office-Home dataset

Method	Resnet50	JAN	DAN	CDAN	DANN	SAFN	LSAFN	LCSAFN	LCSSAFN
Ar to Pr	60.9	61.2	57.0	69.3	59.3	71.7	72.6	72.9	74.4
Pr to Ar	52.9	45.8	44.0	55.6	46.1	63.7	64.2	64.9	66.2
CI to Ar	75.2	50.4	45.8	54.4	47.0	64.2	64.9	66.2	66.4
Ar to CI	38.6	45.9	43.6	49.0	45.6	52.0	53.1	53.2	54.5
Rw to Ar	65.4	63.9	63.1	68.4	63.2	70.9	72.2	72.3	73.1
Ar to Rw	58.0	68.9	67.9	74.5	70.1	76.3	77.5	77.8	78.2
CI to Pr	39.9	59.7	56.5	66.0	58.5	69.9	71.4	72.0	72.0
Pr to CI	31.0	43.4	43.6	48.3	43.7	51.4	53.9	53.7	54.3
CI to Rw	48.1	61.0	60.4	68.4	60.9	71.9	72.4	73.2	74.1
Rw to CI	41.8	52.4	51.5	55.4	51.8	57.1	58.7	59.4	59.7
Pr to Rw	70.8	70.3	67.7	75.9	68.5	77.1	78.6	78.8	79.7
Rw to Pr	70.4	76.8	74.3	80.5	76.8	81.5	82.7	82.6	83.7
Average	53.7	58.3	56.3	63.8	57.6	67.3	68.5	68.9	69.7

In summary, it is evident that the LSAFN method, LCSAFN method, and LCSSAFN method achieve higher accuracy than the SAFN method and other methods for all transfer tasks on the Office-31, imageCLEF-DA, and Office-Home datasets. Therefore, our proposed smooth L2 function and attention module are more accurate and effective than the SAFN method.

5. **Conclusions.** An improved method of transfer learning based on SAFN is presented in this paper. The proposed method consists of an improved channel attention module composed of two convolution layers instead of two fully connected layers and a bottleneck layer used in the conventional channel attention module to reduce the influence of dimension change on accuracy. The proposed channel attention module and spatial channel attention module are included in the feature extraction network to extract relevant features more effectively. Further, we have also offered a smooth L2 function to avoid gradient explosion, which is contained in the classification prediction regression loss function. Simulations were performed using three different datasets to compare our proposed method with a selection of existing methods. Our proposed method achieves higher accuracy than Resnet50, DAN, DANN, JAN, CDAN, and SAFN for all transfer tasks. The highest average accuracies of our proposed method are 2.1%, 2.3% and 2.4% higher than the SAFN method on office-31 dataset, ImageCLEF-DA dataset and Office-Home dataset, respectively.

Acknowledgment. This work was supported by the National Natural Science Foundation of China (61271115), Research Foundation of Education Bureau of Jilin Province (JJKH20220054KJ, JJKH20210095KJ).

REFERENCES

[1] Y. Longze, X. Bai, S. Liguo, "Conditional Depth Convolution Generation of Confrontation Network Method for Scenery Output Scenario Generation," *Journal of Northeast Electric Power University*, vol. 41, no. 6, pp. 90-99, 2021.

[2] W. Yijun, C. Peipei, W. Xuesong, Y. Xingyu, "Research on Insulator Self Explosion Detection Method Based on Deep Learning," *Journal of Northeast Electric Power University*, vol. 40, no. 3, pp. 33-40, 2020.

[3] H. Liu , F. Guo, D. Xia, "Domain adaptation with structural knowledge transfer learning for person re-identification," *Multimedia Tools and Applications*, vol. 80, no. 19, pp. 29321-29337, 2021.
[4] L. Y. Liu, Z. D. Zhang, S. Li, K. Ma, Y. F. Zheng, "S-CUDA: Self-Cleansing Unsupervised Domain Adaptation for Medical Image Segmentation," *Medical Image Analysis*, vol. 74, 102214, 2021.
[5] N. Xu, X. Li, "Intelligent fault diagnosis methodology under varying operating conditions using multi-layer domain adversarial learning strategy," *International Journal of Dynamics and Control*, vol. 9, pp. 1370-1380, 2021.
[6] R. J. Xu, G. B. Li, J. H. Yang, "Larger Norm More Transferable: An Adaptive Feature Norm Approach for Unsupervised Domain Adaptation," *The 12th International Conference on Computer Vision Systems*, pp. 1426-1435, 2019.
[7] K. Bousmalis, G. Trigeorgis, N. Silberman, "Domain separation networks," *The 30th International Conference on Neural Information Processing Systems*, pp. 343-351, 2016.
[8] E. Tzeng, J. Hoffman, K. Saenko, "Adversarial discriminative domain adaptation," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2962-2971, 2017.
[9] M. Long, J. Cao, M. Jordan, "Conditional adversarial domain adaptation," *The 32nd International Conference on Neural Information Processing Systems*, pp. 1647-1657, 2018.
[10] J. Hoffman, E. Tzeng, T. Park, "CyCADA: Cycle-consistent adversarial domain adaptation," *International Conference on Machine Learning*, pp. 1989-1998, 2018.
[11] J. Li, E. Chen, Z. Ding, "Cycle-consistent conditional adversarial transfer networks," *The 27th ACM International Conference on Multimedia*, pp. 747-755, 2019.
[12] Q. Zhou, W. Zhou, B. Yang, "Deep cycle autoencoder for unsupervised domain adaptation with generative adversarial networks," *IET Computer Vision*, vol. 13, no. 7, pp. 659-665, 2019.
[13] F. Zhuang, X. Cheng, P. Luo, "Supervised representation learning: Transfer learning with deep autoencoders," *The 24th International Joint Conference on Artificial Intelligence*, pp. 4119-4125, 2015.
[14] Y. Pan, T. Yao, Y. Li, "Transferrable prototypical networks for unsupervised domain adaptation," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2239-2247, 2019.
[15] J. Shen, Y. Qu, W. Zhang, Y. Yu, "Wasserstein Distance Guided Representation Learning for Domain Adaptation," *The 32nd AAAI Conference on Artificial Intelligence*, pp. 4058-4065, 2018.
[16] C. Y. Lee, T. Batry, M. H. Baig, "Sliced wasserstein discrepancy for unsupervised domain adaptation," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10285-10295, 2019.
[17] A. Gretton , K. M. Borgwardt, M. Rasch," "A Kernel Two-Sample Test, *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 723-773, 2012.
[18] S. J. Pan, I. W. Tsang, J. T. Kwok, "Domain adaptation via transfer component analysis," *IEEE Transactions on Neural Networks*, vol. 22, no.2, pp. 199-210, 2010.
[19] J. LI, M. JING, K. LU , "Locality preserving joint transfer for domain adaptation," *IEEE Transactions on Image Processing*, vol. 28, no. 12, pp. 6103-6115, 2019.
[20] M. Long, J. Wang, G. Ding, "Transfer feature learning with joint distribution adaptation," *IEEE International Conference on Computer Vision*, pp. 2200-2207, 2013.
[21] M. Long, Y. Cao, J. Wang, "Learning Transferable Features with Deep Adaptation Networks," *The 32nd International Conference on Machine Learning*, vol. 37, pp. 97-105, 2015.
[22] M. Long, H. Zhu, J. Wang, "Deep Transfer Learning with Joint Adaptation Networks," *The 34nd International Conference on Machine Learning*, vol. 70, pp. 2208-2217, 2017.
[23] J. LI, K. LU, Z. Huang, "Transfer independently together: A generalized framework for domain adaptation," *IEEE Transactions on Cybernetics*, vol. 49, no. 6, pp. 2144-2155, 2018.
[24] S. Kim , S. Lin , S. R. Jeon, "Recurrent Transformer Networks for Semantic Correspondence," *The 32nd Conference on Neural Information Processing Systems*, vol. 31, pp. 6126-6136, 2018.
[25] J. LI, E. Chen, Z. Ding, "Maximum density divergence for domain adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 3918-3930, 2021.
[26] K. He, X. Zhang, S. Ren, "Deep residual learning for image recognition," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, 2016.
[27] S. Woo, J. Park, J. Y. Lee, I. S. Kweon, "CBMA: Convolutional block attention module," *European Conference on Computer Vision*, pp. 3-19, 2018.
[28] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, Q. Hu "ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11531-11539, 2020.