Unsupervised Domain Adaptation with Hierarchical Gradient Synchronization

Lanqing Hu[1,2] Meina Kan[1,2] Shiguang Shan[1,2,3] Xilin Chen[1,2]
[1] Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China
[2] University of Chinese Academy of Sciences, Beijing 100049, China
[3] CAS Center for Excellence in Brain Science and Intelligence Technology, Shanghai, 200031, China
lanqing.hu@vipl.ict.ac.cn {kanmeina,sgshan,xlchen}@ict.ac.cn

Abstract

Domain adaptation attempts to boost the performance on a target domain by borrowing knowledge from a well established source domain. To handle the distribution gap between two domains, the prominent approaches endeavor to extract domain-invariant features. It is known that after a perfect domain alignment the domain-invariant representations of two domains should share the same characteristics from perspective of the overview and also any local piece. Inspired by this, we propose a novel method called Hierarchical Gradient Synchronization to model the synchronization relationship among the local distribution pieces and global distribution, aiming for more precise domain-invariant features. Specifically, the hierarchical domain alignments including class-wise alignment, group-wise alignment and global alignment are first constructed. Then, these three types of alignment are constrained to be consistent to ensure better structure preservation. As a result, the obtained features are domain invariant and intrinsically structure preserved. As evaluated on extensive domain adaptation tasks, our proposed method achieves state-of-the-art classification performance on both vanilla unsupervised domain adaptation and partial domain adaptation.

1. Introduction

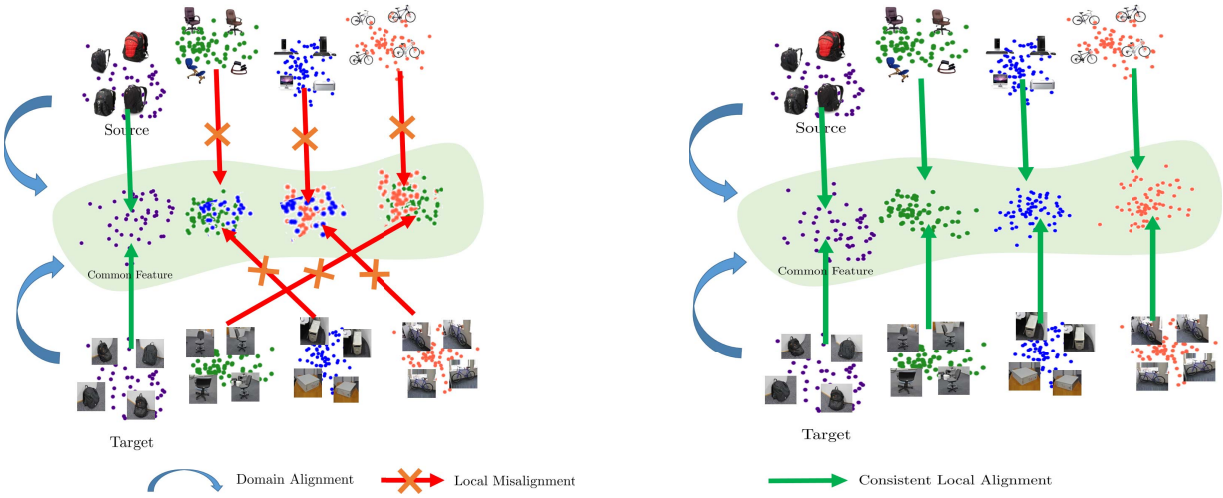
The general hypothesis of machine learning is that the training and testing data share similar distribution, which makes the model trained on a large scale labeled data perform well on the test data. However, in many real world applications, we usually only have access to limited amount of labeled training data sharing similar distribution with the testing data, which is insufficient for training a good enough model. Domain adaptation has shown promising effect on such a challenge by borrowing knowledge from a sophisticated set (i.e., source domain) which has a large number of labeled data but lies in a different distribution with the test data (i.e., target domain).

According to the scale of labeled data in target domain, domain adaptation can be categorized into supervised, semi-supervised and unsupervised domain adaptation. This paper mainly concentrates on the unsupervised domain adaptation where there is only unlabeled data in target domain. Most existing works deal with the domain adaptation problem by alleviating marginal distribution discrepancy (i.e., the distribution of data X) or conditional distribution discrepancy (i.e., distribution of data X given classes labeled with Y). Besides, there are also some works attempting to tackle both the marginal and conditional distribution simultaneously.

In the early days, most methods endeavor to align the marginal distribution of source and target domains by using instance re-weighting, such as sample selection bias [45, 7, 19] and co-variate shift [39, 1]. These approaches are suitable for those scenarios where the source and target domains share the same support, thus they cannot achieve satisfactory performance in the wild scenarios.

For better handling the complicated scenarios, the common subspace methods focusing on extracting domain invariant representation came up [14, 13, 37, 12, 38]. These methods mainly attempt to minimize the gap between marginal distributions of two domains. In the approach of Geodesic Flow Kernel (GFK) [13], an infinite number of the subspaces is integrated to model domain shift between source and target domain. In [12], a set of landmarks, i.e., a subset of labeled data from the source domain that have the most similar distribution as the target domain, are uncovered to bridge the source and target domain. The methods proposed in [24] and [28] embed deep features into Reproducing Kernel Hilbert spaces (RKHS) and minimize the maximum mean discrepancy (MMD) of the features for distribution adaptation. JGSA [46] and PUnDA [11] mitigate the geometrical structure gap and distribution shift jointly. The method in [48] handles the domain shift by aligning the

2575-7075/20/$31.00 ©2020 IEEE
DOI 10.1109/CVPR42600.2020.00410



(a) Local misalignment (b) Expected consist alignment

Figure 1. Illustration of (a) local misalignment in methods only with global distribution alignment, and (b) expected alignment on both global domain and local classes. Best viewed in color.

RKHS covariance matrix across domains.

In these conventional approaches, the distribution discrepancy is usually measured by the metrics like MMD, K-L divergence and Bregman divergence. Recently, the adversarial loss as a more powerful metric has caught a lot of attentions. The works in [9, 10, 41] handle the domain shift by augmenting a gradient reversal layer or employing adversarial objective on target domain features. As a result, the features confusing the domain classifier are generally domain invariant. Afterwards, many methods based on domain transformation via adversarial learning [40, 23, 22, 49, 35] attain quite promising performance on distribution alignment and domain invariant feature extraction.

The above methods only consider the gap between marginal distributions of two domains. In other words, these methods only align the two domains globally, but without considering whether the alignment of local piece is correct or not. As a result, there may happen that two domains are well aligned, but the local pieces (e.g., categories) of two domains are mismatched as shown in Figure 1(a).

In recent years, *a few methods attempt to minimize the gap between conditional distributions (i.e., class-wise distribution) of two domains, for better alignment of the categories between two domains*. Specifically, in WDAN [44], class-specific auxiliary weight for each class is introduced into the original MMD metric for utilizing the class prior on source and target domains. MADA [31] exploits multiple adversarial learning, one for each class, gaining much more performance improvement on target domain. Further, based on this multiple adversarial framework, CDAN [25] novelly designs multi-linear conditioning, i.e., conducting adversarial learning on the covariance between feature representations and classifier predictions, to implicitly align the conditional distribution of source and target domains, which handles the domain distribution alignment more elaborately. Similarly, the methods specialized for partial domain adaptation including SAN [2] and PADA [3] also show the advantages of considering class-wise distribution alignment.

There are some other methods [26, 17, 36, 33, 5, 18, 30, 47] directly predict the category labels of unlabeled samples in target domain as pseudo labels during training process as pseudo-labels. With the pseudo category labels of target domain samples and those known true labels of source domain samples, the samples from distinct domains but the same category are implicitly pulled close to share the same distribution. In the proposed SymNets in [47], the domain discrimination and confusion are stacked upon the concatenated classifiers of source and target domains, thus facilitating the domain-level and category-level feature distribution confusion. MCDDA [34] and CAN [21] are both approaches concentrating on explicitly calibrating the category-level distribution of both domains. MCDDA [34] plays the min-max game between feature encoder and two different classifiers to optimize the decision boundary and then alleviate the intra-class domain discrepancy. CAN [21] explicitly minimizes the intra-class discrepancy and simultaneously maximizes the inter-class discrepancy between domains according to the labels of source and predicted labels of target domain.

Generally, these recently proposed methods consider the alignment of both global distribution (domain-level) discrepancy and local distribution (category-level) discrepancy, thus achieving promising performance. However, in these methods the global alignment and local alignment are implemented in a separate manner, e.g., minimizing weighted sum of domain-level and category-level discrepancy [6, 31, 25]. As a result, the obtained results are only a trade-off of the global and local distribution alignment, and

inconsistent distribution alignment still exists.

As observed from Figure 1(b), in a perfect domain alignment, the calibration of local category and the global domain distribution are consistent, i.e. the calibration direction are roughly the same. To elaborately consider the intrinsic relation between local and global distribution alignment, in this work we propose a new method that can consistently align the local and global distribution by constraining the gradient of local and global alignment to be synchronous, referred to as Domain Adaptation with Hierarchical Gradient Synchronization (GSDA).

Briefly, the contributions of this work are in two folds: (1) we propose a novel method that considers consistency of the global and local distribution alignment, to preserve the intrinsic structures of both domain distributions for better domain adaptation. To the best of our knowledge, it is the first work to explicitly model the intrinsic relation between global and local distribution alignment. (2) The consistency of the global and local distribution alignment is achieved by a newly designed a hierarchical gradient synchronization module. (3) This method achieves state-of-the-art classification accuracy in unsupervised domain adaptation and partial domain adaptation scenarios experimentally.

2. Method

For clear description, we first give some definitions. The labeled source domain images and the unlabeled target domain images are denoted as $X^s = \{(x_i^s, y_i^s)\}_{i=1}^n$ and $X^t = \{x_j^t\}_{j=1}^m$, respectively. In unsupervised domain adaptation, the source and target domains, i.e., X^s and X^t, generally follow different distributions but share the same categories. The samples in source domain are labeled, with category label denoted as $y_i^s \in C^s = \{1, 2, \cdots, r\}$, while the samples in the target domain are unlabeled. In the unsupervised domain adaptation the source and target domains share exactly the same categories, i.e., $C^t = C^s$, where C^t and C^s are r classes in target and source domains. There is also a special scenario where the C^t is a subset of C^s, i.e., $C^t \subset C^s$, called as partial unsupervised domain adaptation. Our method is applicable for both unsupervised domain adaptation and partial unsupervised domain adaptation. For easier understanding we introduce the formulation in the scenario of unsupervised domain adaptation, while evaluate both tasks in the experiments section. Unless otherwise specified, the symbols s and t used in the superscript or subscript denote the source domain and target domain, respectively.

The whole framework of our method is shown in Figure 2, which is equipped with a feature extractor $\mathcal{E}$, an object classifier $\mathcal{C}$ and three types of adversarial discriminators $\mathcal{D} = \{\mathcal{D}^{dom}, \mathcal{D}^{grp}, \mathcal{D}^{cls}\}$. Here, $\mathcal{D}^{dom}$ denotes the adversarial discriminator for globally domain distribution alignment, namely, the domain-level alignment. $\mathcal{D}^{cls}$ denotes adversarial discriminators for locally class-wise distribution alignment. And $\mathcal{D}^{grp}$ represents adversarial discriminators for group-wise alignment where each group is composed of several classes. The feature extractor $\mathcal{E}$ is fed with both the source and target domain data and outputs the features f which are expected to be domain invariant. Afterwards, the features are fed into the classifier $\mathcal{C}$ for classification and also into the adversarial discriminators $\mathcal{D}$ for domain shift reduction. The feature extractor $\mathcal{E}$ and the discriminators $\mathcal{D}$ play a two-player min-max game to make the features from $\mathcal{E}$ domain invariant. In other words, the features from $\mathcal{E}$ should be domain invariant if they successfully fool the domain discriminators $\mathcal{D}$.

2.1. Feature Extraction and Classification

The feature extractor $\mathcal{E}$ encodes the input source or target samples x^s and x^t into a common feature space as follows:

$$f^s = \mathcal{E}(x^s), f^t = \mathcal{E}(x^t), \tag{1}$$

where $\mathcal{E}$ can be any kind of network architecture such as several successive convolutional layers. Then $f \in \{f^s, f^t\}$ is fed into the classifier $\mathcal{C}$ to ensure feature f to be discriminative. The parameter of feature extractor $\mathcal{E}$ and classifier $\mathcal{C}$ are denoted as $\theta_{\mathcal{E}}$ and $\theta_{\mathcal{C}}$, respectively. The output of object classifier $\mathcal{C}$ is denoted as below:

$$p_i^s = \mathcal{C}(f_i^s), p_j^t = \mathcal{C}(f_j^t), \tag{2}$$

where p_i^s is the softmax output of $\mathcal{C}$ with x_i^s as input, and p_j^t is the softmax output of $\mathcal{C}$ with x_j^t as input. Considering that true category labels are available for source domain, the cross entropy loss of classification is directly applied and formulated as below:

$$L_c^s = \sum_{x_i^s \in X^s} H\Big(\mathcal{C}\big(\mathcal{E}(x_i^s)\big), y_i^s\Big), \tag{3}$$

where $H(\cdot, \cdot)$ represents the cross entropy loss.

For target domain samples, the category labels are unavailable, and thus conventional cross entropy loss is inapplicable. Therefore, following [15], the conditional entropy loss is exploited to enhance the certainty of prediction, i.e., force only one element in p_j^t to be dominant while the rest suppressed. Formally, the conditional entropy loss L_c^t for unlabeled target domain samples is as below:

$$L_c^t = \sum_{x_j^t \in X^t} \hat{H}\Big(\mathcal{C}\big(\mathcal{E}(x_j^t)\big)\Big), \tag{4}$$

where $\hat{H}(\cdot)$ is the conditional entropy loss with $\hat{H}(p_j^t) = -\sum_{k=1}^r p_j^t(k) \log p_j^t(k)$. The k^{th} element $p_j^t(k)$ in p_j^t indicates the probability of x_j^t being assigned to the k^{th} class.

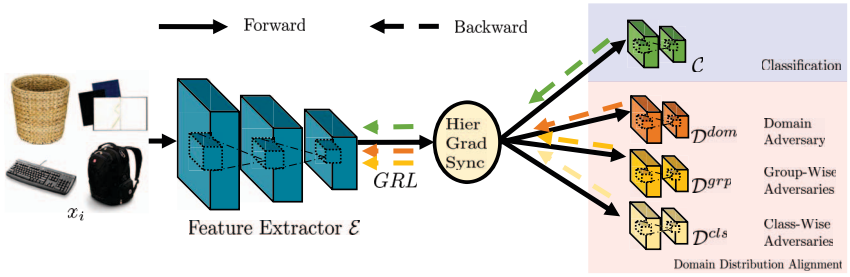


Figure 2. Illustration of the overall framework of our GSDA method. An input sample x_i from source or target domain is firstly encoded by the common feature extractor $\mathcal{E}$. Based on the extracted feature, the classifier $\mathcal{C}$ is designed for object classification, and the adversarial discriminators including $\mathcal{D}^{dom}$, $\mathcal{D}^{grp}$ and $\mathcal{D}^{cls}$ are designed for distribution alignment from perspective of domain-level, group-level and category-level respectively. Furthermore, a hierarchical gradient synchronization between the three types of adversarial discriminators is constructed to constrain the consistency between global and local alignment for better structure preservation. Best viewed in color.

Overall, the object classification loss of both domains is obtained as below:

$$L^c = L_c^s + \alpha L_c^t, \tag{5}$$

constraining the common feature f to be discriminative, benefitting the classification task.

2.2. Domain Distribution Alignment

Besides the categorial discriminability, the feature f from $\mathcal{E}$ should be also domain invariant to potentiate knowledge transfer from source domain to target domain. In a perfect domain-invariant feature space, not only the global structure of both domains but also any local piece such as every group or even every class should be well aligned. Aiming for this goal, three types of adversarial discriminators are introduced for domain-level, group-level, and class-level distribution alignment respectively. Furthermore, the consistency of the three types of alignment are constrained by a novel hierarchical gradient synchronization module. This synchronization module ensures the alignment of any local piece is consistent with the global alignment structure, leading to a more informative domain alignment.

Global Adversarial Discriminator The global adversarial discriminator, i.e., domain-level adversarial discriminator $\mathcal{D}^{dom}$ is designed to distinguish the source domain from target domain with cross entropy loss as follows:

$$\begin{aligned} L^g &= \sum_{x_i \in X^s \cup X^t} H(\mathcal{D}^{dom}(\mathcal{E}(x_i)), d_i), with \\ d_i &= \begin{cases} 1, & if \quad x_i \in X^s, \\ 0, & if \quad x_i \in X^t, \end{cases} \end{aligned} \tag{6}$$

where d_i represents the domain label of each sample x_i.

By playing min-max adversarial optimization between $\mathcal{E}$ and this discriminator $\mathcal{D}^{dom}$ whose parameter is denoted as $\theta_{\mathcal{D}^{dom}}$, the whole distributions of two domains from $\mathcal{E}$ will become nonseparable globally.

Local Adversarial Discriminators Even if the global distribution is well aligned, the distribution of each class in two domains may be misaligned as shown in Figure 1(a), e.g., the i^{th} category of source domain may be aligned to $k^{th}(i \neq k)$ category of target domain although the two domains are globally well aligned. This is because that the global domain migration constraint merely considers the whole domain discrepancy but not the discrepancy in any local piece. Therefore, the local adversarial discriminators are established to deal with the distribution discrepancy in local regions of source and target domains, which consist of two kinds of local adversarial discriminators: class-wise ones and group-wise ones.

Firstly and straightforwardly, class-wise adversarial discriminators are constructed to tackle the discrepancy within each category between the source and target domain, i.e., the i^{th} category of source domain should be aligned to the i^{th} category of target domain rather than other categories in target domain. Formally, the class-wise adversarial discriminator for the k^{th} category is denoted as $\mathcal{D}_k^{cls}$ and its domain discrimination loss is formulated as follows:

$$\begin{aligned} L_k^{cls} &= \sum_{x_i \in X^s \cup X^t} p_i^k H\Big(\mathcal{D}_k^{cls}\big(\mathcal{E}(x_i)\big), d_i\Big), \quad with \\ d_i &= \begin{cases} 1, & if \quad x_i \in X^s \\ 0, & if \quad x_i \in X^t, \end{cases} \end{aligned} \tag{7}$$

where d_i is the domain label, similar with that in global adversarial discriminator, $k \in \{1, 2, \cdots, r\}$ denotes the index of k^{th} class-wise adversarial discriminator and p_i^k is the loss weight of sample x_i representing its probability of belonging to k^{th} class, i.e., the k^{th} dimension output of p_i^s and p_i^t in Equation (2). Note that if $x_i \in X^s$ and it belongs to the k^{th} class, $p_i^k = 1$ and $p_i^j|_{j \neq k} = 0$ because the label of $x_i \in X^s$ is definite. While for $x_i \in X^t$, as its label is unavailable, the corresponding p_i^k is the predicted probability of $x_i \in X^t$ to be classified into the k^{th} class by classifier C in Equation (2).

Likewise, by playing min-max adversarial optimization

with the objective above, the distribution of two domains is well aligned for each category. The parameter of each class-wise local discriminator $\mathcal{D}_k^{cls}$ is denoted as $\theta_{\mathcal{D}_k^{cls}}$.

Besides each class, any local group consisting of several classes should be also well aligned in a perfect domain alignment. Thus, the local alignment can be reinforced by establishing group-level adversarial discriminators. Similar as the class-wise adversarial discriminators, the group-wise adversarial discriminators $\mathcal{D}_q^{grp}$ for the q^{th} group with domain discrimination loss is formulated as follows:

$$L_q^{grp} = \sum_{x_i \in X^s \cup X^t} p_i^q H\Big(\mathcal{D}_q^{grp}\big(\mathcal{E}(x_i)\big), d_i\Big), with$$
$$d_i = \begin{cases} 1, & if \quad x_i \in X^s \\ 0, & if \quad x_i \in X^t, \end{cases} \tag{8}$$

where $q \in \{1, 2, \cdots, b\}$ denotes the index of q^{th} group-wise adversarial discriminator, the p_i^q denotes the probability of x_i belonging to the q^{th} group. The groups here are simply achieved as random divisions of all classes that are defined in Equation (7). Correspondingly, the category grouping probability of the q^{th} group p_i^q can be easily obtained as $p_i^q = \sum_{k \in q} p_i^k$. Generally, the classes in different groups are allowed to overlap with each other, while in this work all groups are simply randomly divided without overlap. What is worth mentioning is that, when the number of classes is large, these groups could be hierarchically structured groups rather than flat structured ones.

Similarly, by playing min-max adversarial optimization with the objective above, the distribution of two domains is well aligned locally in each group. The parameter of each group-wise local discriminator $\mathcal{D}_q^{grp}$ is denoted as $\theta_{\mathcal{D}_k^{cls}}$. The parameter of each group-wise local discriminator $\mathcal{D}_q^{grp}$ is denoted as $\theta_{\mathcal{D}_q^{grp}}$.

Then the overall parameters of all discriminators are denoted as $\theta_{\mathcal{D}} = \{\theta_{\mathcal{D}^{dom}}, \theta_{\mathcal{D}^{cls}}, \theta_{\mathcal{D}_q^{grp}}\}$. By summing up all the local adversarial discriminators, the objective for local distribution alignment is obtained as:

$$L^l = \sum_{q=1}^{b} L_q^{grp} + \sum_{k=1}^{r} L_k^{cls}, \tag{9}$$

where b stands for the number of groups and r represents the number of classes.

Overall, the three types of distribution alignment including domain-level, group-wise, and class-wise domain distribution alignment form a hierarchical aligning structure, aiming for better alignment between source and target domains globally as well as locally.

2.3. Hierarchical Gradient Synchronization

The preceding global and local adversarial discriminators deal with the distribution alignment between domains from global and local perspective, but in an independent manner. This may cause inconsistency among the global and local alignments, which would compromise the aligning direction of global and local alignment leading to inaccurate distribution alignment.

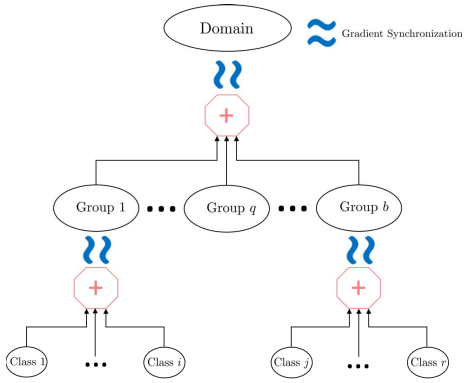


Figure 3. Illustration of the hierarchical distribution alignments and hierarchical gradient synchronization among them.

Actually, in a perfect global alignment, any local piece should be also well aligned, or vice versa: a perfect alignment of each local piece also forms an optimal global alignment. Specifically, the aligning direction and magnitude of each local piece should be consistent with that of the whole domain. So intuitively the consistency between the global and local domain alignment could be used to verify if two domains are well aligned or not. In return, it would benefit the domain alignment if this consistency is formulated into the process of distribution alignment. With this in mind, a novel constraint on the gradient is designed as the *Hierarchical Gradient Synchronization* term, which is presented in Figure 3 and specifically formulated in Equations (10) and (11) below.

Specifically, Hierarchical Gradient Synchronization consists of gradient synchronization among the three levels of adversarial discriminators, i.e., domain-level, group-level, and class-level discriminators, forming a hierarchical manner. The gradient synchronization between class-wise alignment and group-wise alignment is designed as below:

$$L_{grp \sim cls}^{syn} = \left| \sum_{\substack{x_i \in \\ X^s \cup X^t}} \left|\left| \frac{\partial L_q^{grp}}{\partial \mathcal{E}(x_i)} \right|\right|_2 - \sum_{k \in grp_q} \sum_{\substack{x_i \in \\ X^s \cup X^t}} \left|\left| \frac{\partial L_k^{cls}}{\partial \mathcal{E}(x_i)} \right|\right|_2 \right|. \tag{10}$$

The gradient synchronization objective in the above Equation (10) attempts to make the magnitude of the aligning direction of each group to be consistent with the sum of that of each class within this group. Here, the first term denotes the gradient magnitude of discriminator for the q^{th} group, and the second term denotes the gradients magnitude of discriminators for each class in q^{th} group in the domain. Note that here we only use constraint on magnitude as it can affect both the direction and magnitude, while the sum of gradients direction will neutralize the difference. Note that in the second term, $x_i \in X^s \cup X^t$ still means the samples from the k^{th} class because the sampling probability is

included in L_k^{cls}.

Similarly, the gradient synchronization between group-wise and the whole domain alignment is formulated as follows:

$$L_{dom\sim grp}^{syn} = \left| \sum_{\substack{x_i \in \\ X^s \cup X^t}} \left|\left| \frac{\partial L^g}{\partial \mathcal{E}(x_i)} \right|\right|_2 - \sum_{q=1}^{b} \sum_{\substack{x_i \in \\ X^s \cup X^t}} \left|\left| \frac{\partial L_q^{grp}}{\partial \mathcal{E}(x_i)} \right|\right|_2 \right|, \tag{11}$$

where the first term denotes the gradients magnitude of discriminator for the whole domain, and the second term denotes the gradients magnitude of discriminators for each group. Note that in the second term, $x_i \in X^s \cup X^t$ still means the samples from the q^{th} group because the sampling probability is included in L_q^{grp}.

Note that although Equations (10) and (11) are the losses with regard to the gradients, they are first-order derivatives optimization rather than second-order ones which is efficient. This is because that the gradients in Equations (10) and (11) are with regard to the input features, but not with regard to the network parameters.

Afterwards, piling all the layers together, the overall 3-layer hierarchical gradient synchronization constraint is naturally obtained as below:

$$L^{syn} = \frac{1}{b} \sum_{q=1}^{b} L_{grp\sim cls}^{syn} + L_{dom\sim grp}^{syn}. \tag{12}$$

With this constraint, the directions and magnitude of gradient descent for both global and local alignment are expected to be kept in synchronization with each other. As a result, the distributions of two domains can be aligned more accurately.

With the global alignment, local alignment, and gradient synchronization defined in Equations (6), (9) and (12), the overall objective function of the discriminators $\mathcal{D}$ is finally formulated as follows:

$$L^d = L^g + L^l + \beta L^{syn}. \tag{13}$$

With the objective in above Equation (13), the source and target domain are aligned globally and locally, with consistency between the global and local distribution alignment. As as result, the two domains are well aligned and also the discriminative structure are well preserved.

2.4. Overall Objective and Optimization

The overall objective function is optimized by alternatively optimizing $\{\mathcal{E}, \mathcal{C}\}$ and $\mathcal{D}$ following the adversarial learning mechanism, which are detailed in the following.

Given $\{\mathcal{E}, \mathcal{C}\}$, the adversarial discriminators $\mathcal{D}$ are optimized to distinguish the source domain from target domain by minimizing the domain discrimination loss:

$$\min_{\theta_{\mathcal{D}^g}, \theta_{\mathcal{D}^l}} L^d = L^g + L^l + \beta L^{syn}, \tag{14}$$

with the parameters updated as below:

$$\begin{aligned} \theta_{\mathcal{D}^g} &\leftarrow \theta_{\mathcal{D}^g} - \eta \frac{\partial (L^g + \beta L^{syn})}{\partial \theta_{\mathcal{D}^g}}, \\ \theta_{\mathcal{D}_k^l} &\leftarrow \theta_{\mathcal{D}_k^l} - \eta \frac{\partial (L^l + \beta L^{syn})}{\partial \theta_{\mathcal{D}_k^l}}, \end{aligned} \tag{15}$$

where η is the learning rate.

Given $\mathcal{D}$, the feature extractor $\mathcal{E}$ and classifier $\mathcal{C}$ are optimized to make the features from $\mathcal{E}$ are discriminative and domain invariant. This is achieved by minimizing the object classification loss and confusing the adversarial discriminators by the min-max game as follows:

$$\min_{\theta_{\mathcal{C}}, \theta_{\mathcal{E}}} \left(L^c + \beta L^{syn} - (L^g + L^l) \right), \tag{16}$$

with the parameters updated as:

$$\begin{aligned} \theta_{\mathcal{C}} &\leftarrow \theta_{\mathcal{C}} - \eta \frac{\partial L^c}{\partial \theta_{\mathcal{C}}}, \\ \theta_{\mathcal{E}} &\leftarrow \theta_{\mathcal{E}} - \eta \left(\frac{\partial L^c}{\partial \theta_{\mathcal{C}}} \times \frac{\partial \theta_{\mathcal{C}}}{\partial \theta_{\mathcal{E}}} + \beta \frac{\partial L^{syn}}{\partial \theta_{\mathcal{E}}} - \frac{\partial (L^g + L^l)}{\partial \theta_{\mathcal{D}}} \times \frac{\partial \theta_{\mathcal{D}}}{\partial \theta_{\mathcal{E}}} \right). \end{aligned} \tag{17}$$

3. Experiments

We evaluate the proposed method and other related works on both unsupervised domain adaptation (source and target domains share the same categories) and partial domain adaptation (the categories of target domain is a subset of that of source domain) benchmarks of object classification, of which *the partial domain adaptation results will be given in supplementary materials*. Besides, ablation study is carefully done for analysing the contributions of each part of the proposed method.

3.1. Datasets and Experimental Setting

Three standard benchmarks for unsupervised domain adaptation and one for partial domain adaptation, respectively, are employed for the evaluation.

Office-31-DA Office-31 [20] is a classical and widely used benchmark for domain adaptation with 31 categories, consisting of 3 different domains including Amazon (A) with 2817 images, Webcam (W) with 795 images, and D-SLR (D) with 498 images. Following the commonly used protocol defined in [13, 26, 28, 31], all 31 categories from the three domains are used for evaluation of unsupervised domain adaptation, forming 6 transfer tasks.

Office-Home [42] Office-Home is another classical dataset with 65 categories, consisting of 4 different domains including Artistic images (Ar), Clip Art images (Cl), Product images (Pr) and Real-World images (Rw). Following the commonly utilized protocol defined in [13, 26, 28, 31], all 65 categories from the four domains are used for evaluation of unsupervised domain adaptation, forming 12 transfer tasks.

Table 1. Ablation study of our GSDA for domain adaptation on Office-31 (ResNet50).

Glb	Cls	Grp	Grad Sync	A→W	D→W	W→D	A→D	D→A	W→A	Avg
✓				87.9	98.2	**100.0**	85.5	66.4	64.1	83.7
✓	✓			91.7	98.4	**100.0**	87.1	68.9	67.2	85.6
✓	✓	✓		93.1	99.0	**100.0**	91.4	71.5	67.0	87.0
✓	✓	✓	✓	**95.7**	**99.1**	**100.0**	**94.8**	**73.5**	**74.9**	**89.7**

VisDA-2017 VisDA-2017 [32] is a more challenging simulation-to-real task, with two distinct domains: synthetic object images rendered from 3D models and real object images. It contains 152397 training images and 55388 validation images across 12 classes. Following the training and testing protocol in [34, 25], the model is trained on labeled training and unlabeled validation set and tested on the validation set in unsupervised domain adaptation.

Office-31-PDA Recently, a new protocol for *partial domain adaptation* is built on Office-31 [20]. As defined in [2, 3], the same three domains as that for the standard unsupervised domain adaptation are used but with different categories for source and target domains: all 31 categories from the three domains are used as source domains, denoted as A31, D31, and W31, respectively, while the 10 common categories shared between Office-31 and Caltech-256 are used as target domains, denoted as A10 (958 images), W10 (295 images) and D10 (157 images), respectively.

Implementation Details For fair comparison, on each setting we use the same network architecture as the compared methods. Specifically, we use ResNet50 as the backbone in all the experiments. In Office-31-DA and Office-Home, the hyper-parameter α in Equation (5) and β in Equation (13) is set as 0.02, and 1.0. In VisDA-2017 and Office-31-PDA, α and β are set as 0.2 and 10.0, respectively. In Office-31-DA and Office-31-PDA, classes are divided into 6 groups. In Office-Home, they are divided into 13 groups. And in VisDA-2017, they are divided into 4 groups. For clearer explanation of hyper-parameter selection, the sensitivity analysis about hyper-parameters is presented in supplementary materials. For stable training of GSDA, those categories with fewer samples are augmented by randomly re-sampling images to make all categories have roughly the same number of images to avoid the data imbalance problem stated in [50]. For target domain, the class labels are unavailable, so only those samples with highly confident pseudo labels are used as training samples.

3.2. Ablation Study

The ablation study is conducted on unsupervised domain adaptation setting (Office-31-DA) to investigate the necessity of each component in GSDA. Briefly, our GSDA consist of three parts, global alignment, local alignment, and the hierarchical gradient synchronization between them. As shown in Table 1, the method with only global domain alignment (*Glb*) performs worse than that added with class-wise alignment (*Cls*), showing that the local alignment is important for keeping discriminative structure during adaptation. Then constructed with group-wise alignment (*Grp*), the model has further improvement because the discriminative structure is captured more elaborately by randomly combining several classes as a group. Furthermore, by considering hierarchical gradient synchronization between global alignment and local alignment, our GSDA (with gradient synchronization denoted as *Grad Sync*) achieves significant improvement indicating its effectivenesswhich also illustrates the necessity of consistency between global and local distribution alignment. Clearly, our main contributions, i.e., group-wise alignment and hierarchical alignment synchronization, shows promising benefit for domain adaptation.

Table 2. Object classification accuracy on Office-31-DA (ResNet50). All methods follow the same settings, so most results are directly from the original works except MCDDA which is tuned using the released codes.

Method	A ↓ W	D ↓ W	W ↓ D	A ↓ D	D ↓ A	W ↓ A	Avg
ResNet50 [16]	68.4	96.7	99.3	68.9	62.5	60.7	76.1
TCA [29]	72.7	96.7	99.6	74.1	61.7	60.9	77.6
GFK [13]	72.8	95.0	98.2	74.5	63.4	61.0	77.5
DAN [24]	80.5	97.1	99.6	78.6	63.6	62.8	80.4
RTN [28]	84.5	96.8	99.4	77.5	66.2	64.8	81.6
JAN [27]	85.4	97.4	99.8	84.7	68.6	70.0	84.3
DANN [10]	82.0	96.9	99.1	79.7	68.2	67.4	82.2
ADDA [41]	86.2	96.2	98.4	77.8	69.5	68.9	82.9
MCDDA [34]	82.6	98.9	99.8	84.3	66.2	66.3	83.0
MADA [31]	90.0	97.4	99.6	87.8	70.3	66.4	85.2
CDAN [25]	94.1	98.6	**100.0**	92.9	71.0	69.3	87.7
SymNets [47]	90.8	98.8	**100.0**	93.9	**74.6**	72.5	88.4
SAFN [43]	90.3	98.7	**100.0**	92.1	73.4	71.2	87.6
BSP [4]	93.3	98.2	**100.0**	93.0	73.6	72.6	88.5
GSDA (Ours)	**95.7**	**99.1**	**100.0**	**94.8**	73.5	**74.9**	**89.7**

3.3. Unsupervised Domain Adaptation

Unsupervised domain adaptation is the most typical setting for domain adaptation, and there are many related works such as the conventional methods TCA [29] and GFK [13], the deep adaptation works based on MMD criterion like DAN [24], RTN [28] and JAN [27], and the adversarial learning based approaches including DANN [10], ADDA [41], MADA [31], CDAN [25] and SymNets [47]. All these methods are compared with our method on Office-31-DA, Office-Home and VisDA-2017 introduced in Section 3.1. The experiment results are shown in Tables 2, 3 and 4.

As can be seen, the baseline without adaptation and the conventional non-deep methods perform the worst, while

Table 3. Object classification accuracy on Office-Home dataset (ResNet50). All methods follow the same settings, so all the results are directly copied from the original works.

Method	Ar → Cl	Ar → Pr	Ar → Rw	Cl → Ar	Cl → Pr	Cl → Rw	Pr → Ar	Pr → Cl	Pr → Rw	Rw → Ar	Rw → Cl	Rw → Pr	Avg
ResNet50 [16]	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DAN [24]	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3	56.3
DANN [10]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
JAN [27]	45.9	61.2	68.9	50.4	59.7	61.0	45.8	43.4	70.3	63.9	52.4	76.8	58.3
CDAN [25]	50.7	70.6	76.0	57.6	70.0	70.0	57.4	50.9	77.3	70.9	56.7	81.6	65.8
SymNets[47]	47.7	72.9	78.5	64.2	71.3	74.2	64.2	48.8	79.5	**74.5**	52.6	82.7	67.6
SAFN[43]	54.4	73.3	77.9	65.2	71.5	73.2	63.6	52.6	78.2	72.3	58.0	82.1	68.5
GSDA (Ours)	**61.3**	**76.1**	**79.4**	**65.4**	**73.3**	**74.3**	**65.0**	**53.2**	**80.0**	72.2	**60.6**	**83.1**	**70.3**

Table 4. Object classification accuracy on VisDA-2017 task (ResNet50). All methods follow the same settings and encoder architecture except the methods marked with † using ResNet101. The results are directly copied from the original works. The underlined results mean the highest accuracies of the four marked methods with deeper networks or multiple data augmentations (S-En).

Method	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	Avg
ResNet50 [16]	70.6	51.8	55.8	68.9	77.9	7.6	93.3	34.5	81.1	27.9	88.6	5.6	55.3
DAN[24]	61.7	54.8	77.7	32.2	75.0	80.8	78.3	46.9	66.9	34.5	79.6	29.1	59.8
DANN[10]	75.9	70.5	65.3	17.3	72.8	38.6	58.0	77.2	72.5	40.4	70.4	44.7	58.6
MCDDA † [34]	87.0	60.9	83.7	64.0	88.9	79.6	84.7	76.9	88.6	40.3	83.0	25.8	71.9
TPN [30]	**93.7**	**85.1**	69.2	81.6	93.5	61.9	89.3	**81.4**	**93.5**	81.6	**84.5**	**49.9**	80.4
S-En* [8]	96.3	87.9	84.7	55.7	95.9	95.2	88.6	77.4	93.3	92.8	87.5	38.2	82.8
BSP †[4]	92.4	61.0	81.0	57.5	89.0	80.6	90.1	77.0	84.2	77.9	82.1	38.4	75.9
SAFN† [43]	93.6	61.3	84.1	70.6	94.1	79.0	91.8	79.6	89.9	55.6	89.0	24.4	76.1
GSDA (Ours)	93.1	67.8	**83.1**	**83.4**	**94.7**	**93.4**	**93.4**	79.5	93.0	**88.8**	83.4	36.7	**81.5**

the deep methods with MMD criterion such as DAN [24] and JAN [27] perform much better benefited from the favorable non-linearity of the deep networks. Furthermore, the adversarial learning based methods including DANN [10], ADDA [41], MADA [31] and ours perform even better than those MMD based deep methods attributing to the more powerful capability of adversarial learning for reducing distribution discrepancy.

Among the adversarial-based methods, DANN [10] and ADDA [41] are early ones only concentrating on global distribution alignment which outperform the MMD-based methods but with limited improvement. MADA [31], CDAN [25] and SymNets [47] further consider class-level alignment achieving more promising adaptation. However, they do not consider the intrinsic relation between local and global alignment, so some misalignment may still appear. Go a further step, our proposed method GSDA considers not only global and local (i.e., class-wise and group-wise) alignment, but also the hierarchical gradient synchronization relation between them, leading to better adaptation.

Moreover, BSP [4] and SAFN [43] are recently proposed methods with different perspectives from feature distribution alignment. BSP penalizes the largest singular values of feature eigenvectors to enhance the discriminality and SAFN improves the transferablity by magnifying norm of features. Compared with these two novel methods, our method still achieves the best performance, demonstrating the advantage and necessity of considering the relation between the global and local distribution alignment.

4. Conclusion and Future Work

Aiming for better unsupervised domain adaption, we propose a novel method named GSDA aligning the distribution of two different domains globally and locally as well, with gradient synchronization between them. The hierarchical gradient synchronization module is established to ensure the consistency between global and local distribution alignment for better structure preservation. The extensive experiments verify the superiority of our method.

The gradient synchronization between global and local domain alignment has achieved promising improvement in this work, and this also implies that the relation between global and local distribution alignment deserves deeper analysis and exploration in future.

ACKNOWLEDGEMENT

This work is partially supported by National Key R&D Program of China (No. 2017YFA0700800), Natural Science Foundation of China (No. 61772496) and UCAS Joint PhD Training Program.

References

[1] S. Bickel, M. Brückner, and T. Scheffer. Discriminative learning under covariate shift. *Journal of Machine Learning Research (JMLR)*, 10(9):2137–2155, 2009.

[2] Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Michael I. Jordan. Partial transfer learning with selective adversarial networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[3] Zhangjie Cao, Lijia Ma, Mingsheng Long, and Jianmin Wang. Partial adversarial domain adaptation. In *European Conference on Computer Vision (ECCV)*, 2018.

[4] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *International Conference on Machine Learning (ICML)*, 2019.

[5] Z. Ding and Y. Fu. Robust transfer metric learning for image classification. *IEEE Transactions on Image Processing (TIP)*, 26(2):660–670, 2016.

[6] Zhengming Ding, Sheng Li, Ming Shao, and Yun Fu. Graph adaptive knowledge transfer for unsupervised domain adaptation. In *European Conference on Computer Vision (ECCV)*, 2018.

[7] M. Dudık, R. E. Schapire, and S. J. Phillips. Correcting sample selection bias in maximum entropy density estimation. In *Neural Information Processing Systems (NeurIPS)*, 2005.

[8] Geoffrey French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. In *International Conference on Representation Learning (ICLR)*, 2018.

[9] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine learning (ICML)*, 2015.

[10] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, and et al. Domain-adversarial training of neural networks. *Journal of Machine Learning Research (JMLR)*, 17(1):2096–2030, 2016.

[11] Behnam Gholami, Ognjen Rudovic, and Vladimir Pavlovic. Punda: Probabilistic unsupervised domain adaptation for knowledge transfer across visual categories. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[12] B. Gong, K. Grauman, and F. Sha. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In *International Conference on Machine learning (ICML)*, 2013.

[13] B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[14] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *IEEE International Conference on Computer Vision (ICCV)*, 2011.

[15] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Neural Information Processing Systems (NeurIPS)*, 2005.

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[17] Cheng An Hou, Yao Hung Hubert Tsai, Yi Ren Yeh, and Yu Chiang Frank Wang. Unsupervised domain adaptation with label and structural consistency. *IEEE Transactions on Image Processing (TIP)*, 25(12):5552–5562, 2016.

[18] Lanqing Hu, Meina Kan, Shiguang Shan, and Xilin Chen. Duplex generative adversarial network for unsupervised domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[19] J. Huang, A. J. Smola, A. Gretton, K. M. Borgwardt, B. Schölkopf, and et al. Correcting sample selection bias by unlabeled data. In *Neural Information Processing Systems (NeurIPS)*, 2007.

[20] M. Fritz K. Saenko, B. Kulis and T. Darrell. Adapting visual category models to new domains. In *European Conference on Computer Vision (ECCV)*, 2010.

[21] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G. Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[22] M. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. In *Neural Information Processing Systems (NeurIPS)*, 2017.

[23] M. Liu and O. Tuzel. Coupled generative adversarial networks. In *Neural Information Processing Systems (NeurIPS)*, 2016.

[24] M. Long, Y. Cao, J. Wang, and M. I. Jordan. Learning transferable features with deep adaptation networks. In *International Conference on Machine learning (ICML)*, 2015.

[25] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *Neural Information Processing Systems (NeurIPS)*, 2018.

[26] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S. Yu. Transfer feature learning with joint distribution adaptation. In *IEEE International Conference on Computer Vision (ICCV)*, 2014.

[27] Mingsheng Long, Jianmin Wang, and Michael I. Jordan. Deep transfer learning with joint adaptation networks. In *International Conference on Machine learning (ICML)*, 2017.

[28] M. Long, H. Zhu, J. Wang, and M. I. Jordan. Unsupervised domain adaptation with residual transfer networks. In *Neural Information Processing Systems (NeurIPS)*, 2016.

[29] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks (TNN)*, 22(2):199–210, 2010.

[30] Yingwei Pan, Ting Yao, Yehao Li, Yu Wang, Chong-Wah Ngo, and Tao Mei. Transferrable prototypical networks for unsupervised domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[31] Z. Pei, Z. Cao, M. Long, and J. Wang. Multi-adversarial domain adaptation. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[32] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. *CoRR*, abs/1710.06924, 2017.

[33] K. Saito, Y. Ushiku, and T. Harada. Asymmetric tri-training for unsupervised domain adaptation. In *International Conference on Machine learning (ICML)*, 2017.

[34] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[35] Swami Sankaranarayanan, Yogesh Balaji, Carlos D. Castillo, and Rama Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[36] O. Sener, H. O. Song, A. Saxena, and S. Savarese. Learning transferrable representations for unsupervised domain adaptation. In *Neural Information Processing Systems (NeurIPS)*, 2016.

[37] M. Shao, C. Castillo, Z. Gu, and Y. Fu. Low-rank transfer subspace learning. In *International Conference on Data Mining (ICDM)*, 2012.

[38] M. Shao, D. Kit, and Y. Fu. Generalized transfer subspace learning through low-rank constraint. *International Journal of Computer Vision (IJCV)*, 109(1-2):74–93, 2014.

[39] M. Sugiyama, M. Krauledat, and K-B. MÃžller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research (JMLR)*, 8(5):985–1005, 2007.

[40] Y. Taigman, A. Polyak, and L. Wolf. Unsupervised cross-domain image generation. 2017.

[41] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[42] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[43] Ruijia Xu, Guanbin Li, Jihan Yang, and Liang Lin. Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[44] 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)*, 2017.

[45] B. Zadrozny. Learning and evaluating classifiers under sample selection bias. In *International Conference on Machine learning (ICML)*, 2004.

[46] Jing Zhang, Wanqing Li, and Philip Ogunbona. Joint geometrical and statistical alignment for visual domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[47] Yabin Zhang, Hui Tang, Kui Jia, and Mingkui Tan. Domain-symmetric networks for adversarial domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[48] Zhen Zhang, Mianzhi Wang, Yan Huang, and Arye Nehorai. Aligning infinite-dimensional covariance matrices in reproducing kernel hilbert spaces for domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[49] Jun Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[50] Yang Zou, Zhiding Yu, B.V.K. Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *European Conference on Computer Vision (ECCV)*, 2018.