Gradient Distribution Alignment Certificates Better Adversarial Domain Adaptation

Zhiqiang Gao[1], Shufei Zhang[1], Kaizhu Huang*[1], Qiufeng Wang[1], and Chaoliang Zhong[2]

[1] Xi'an Jiatong-Liverpool University, Suzhou, China [2] Fujitsu R&D Centre, Beijing, China

Abstract

The latest heuristic for handling the domain shift in unsupervised domain adaptation tasks is to reduce the data distribution discrepancy using adversarial learning. Recent studies improve the conventional adversarial domain adaptation methods with discriminative information by integrating the classifier's outputs into distribution divergence measurement. However, they still suffer from the equilibrium problem of adversarial learning in which even if the discriminator is fully confused, sufficient similarity between two distributions cannot be guaranteed. To overcome this problem, we propose a novel approach named feature gradient distribution alignment (FGDA)[1]*. We demonstrate the rationale of our method both theoretically and empirically. In particular, we show that the distribution discrepancy can be reduced by constraining feature gradients of two domains to have similar distributions. Meanwhile, our method enjoys a theoretical guarantee that a tighter error upper bound for target samples can be obtained than that of conventional adversarial domain adaptation methods. By integrating the proposed method with existing adversarial domain adaptation models, we achieve state-of-the-art performance on two real-world benchmark datasets.*

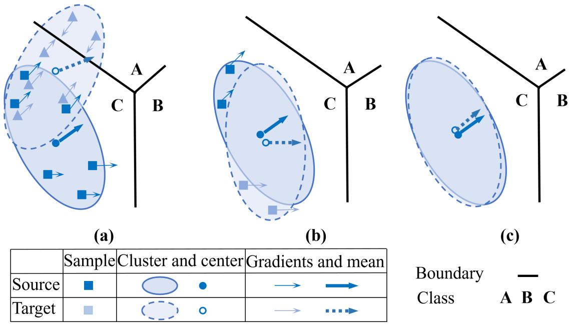


Figure 1. Illustration of Feature Gradient Distribution Alignment (FGDA). **(a)-(c)**: When features of two domains distribute very differently due to large domain shift, their gradients in non-overlapping regions may disperse in distinct parts of the highly complicated decision boundary, which leads to a large gradient distribution discrepancy. With the gradient alignment, the overlapping region is enlarged to reduce the domain shift. **(a)-(b)**: Domain shift measured by the conventional adversarial domain adaptation method tends to be zero when two mean features are close enough. In this case, conventional methods fail to further reduce the domain shift. **(b)-(c)**: Even if the distance between two mean features is small, the domain shift measured by our method (in terms of feature gradient discrepancy) can be still observed due to obvious different gradients in non-overlapping regions. FGDA can certificate a further domain shift reduction.

1. Introduction

Deep Neural Networks (DNNs) have achieved impressive performance for various applications such as image classification [12, 16], object detection [11, 26] and semantic segmentation [21, 24]. However, DNNs may not generalize well on new data due to the data distribution shift problem which manifest in many different ways, such as sample selection bias [5], class distribution shift [19], and covariate shift [29]. Unsupervised Domain Adaptation (UDA) aims to address domain shift with access to labeled source data and unlabeled target data [8]. The fundamental objective is to infer the domain-invariant representations [32].

Among current deep architectures, the adversarial domain adaptation (ADA) approaches [25, 22, 30, 15] are widely investigated and achieve state-of-the-art performance. As one seminal work, Domain-Adversarial Neural Networks (DANN) integrates adversarial learning and domain adaptation into a mini-max game [8]. A domain discriminator is learned to distinguish the source distribution from the target one, while a deep classification model learns transferable representations that are indistinguishable for the domain discriminator. Recent successful methods

*Corresponding author: Kaizhu Huang (kaizhu.huang@xjtlu.edu.cn).
[1] The codes is available at https://github.com/gzqhappy/FGDA.

revealed that a discriminative distribution alignment enables a better domain adaptation [25, 22, 30, 15, 27, 38]. The key idea of those studies is leveraging the discriminative information delivered by the classifier's outputs or predictions for target discriminative representation learning.

Although the discriminative information can help promote the performance on domain adaptation, we argue that the domain shift still presents one major challenge which limits further performance improvement. Such drawback comes from the equilibrium challenge of adversarial learning [2] in which even if the discriminator is fully confused, there is no guarantee that two distributions are sufficiently similar, as shown in Fig. 1 (a) and (b).

To tackle this problem, we propose a novel method called feature gradient distribution alignment (FGDA) in order to further reduce domain shift. Specifically, FGDA learns to reduce distribution discrepancy of feature gradient between two domains in the manner of the adversarial learning between the feature extractor and the discriminator. When the equilibrium is reached, the value of the feature distribution discrepancy can be minimal.

We borrow the insight of adversarial perturbations from [10, 1, 37] to simply describe the principle of the proposed method. Input gradients of samples can be considered as sensitive directions which perturb the input least in order to change the model's output most [1]. Intuitively, the feature gradient direction of one sample may tend to point to the region of its nearest decision boundary. Furthermore, feature gradients apart from each other are probably different significantly since they point to the distinct part of the highly complicated decision boundary (as typically seen in DNNs); the feature gradients closer to each other may share a similar direction. Therefore, aligning the feature gradients encourages learning the latent representations which enforce the two domain distributions to stay closer. As such, the feature distribution discrepancy can be reduced. For the merit of our method, compared with conventional domain adaptation methods, our method can further reduce the domain shift even if the mean features of two domains are close to each other as shown in Fig. 1, which is also theoretically analyzed as later seen in Section 3.7. Importantly, we further prove that aligning the feature gradients leads to a tighter upper bound than conventional adversarial domain adaptation methods with respect to the expected error on target samples.

In a nutshell, our key contributions are listed as follows:

- We propose a novel method FGDA where adversarial learning is adopted to align the feature gradients for reducing distribution discrepancy. Compared with conventional methods, our model can further reduce the domain shift even if the means of the source and target distributions are close to each other.
- We prove the efficacy of our method both theoretically and empirically. In particular, we show that our method can obtain a tighter upper bound than conventional domain adaptation methods.
- We conduct extensive experiments to show that the proposed approach is not only able to reduce domain discrepancy but also offers improvement consistently over current feature-based adversarial domain adaptation methods. Particularly, our approach achieves state-of-the-art performances on tasks of UDA.

2. Related Work

The adversarial domain adaptation (ADA) approaches [7, 34, 35], which not only provides theoretical guarantee but also achieves state-of-the-art performance, are widely studied recently. Inspired by Generative Adversarial Networks (GANs) [9], these methods learn a domain-invariant feature in a mini-max game, where a feature extractor learns to fool a domain discriminator while the discriminator struggles to be not fooled [8]. On par with these feature-level approaches, generative pixel-level adaptation models perform distribution alignment in the raw pixel space, by translating source data to a target domain using image-to-image translation techniques [40, 20, 14, 28].

Despite their general efficacy for various tasks ranging from classification [7, 35, 20] to segmentation [28, 33, 14], they have to face the equilibrium problem of adversarial learning [2]. As a result, when complex multi-modal structures of data distributions exists, vanilla ADA methods may fail to capture such multi-modal structures for a discriminative alignment of distributions without mode mismatch [22].

To further reduce the domain shift, recent successful ADA methods are devoted to achieving a discriminative distribution alignment. One of the widely studied methods is named class-conditional ADA conditioning the domain classifier on features and corresponding predictions simultaneously [25, 22, 30, 15]. These approaches aim to approximate a joint distribution alignment between two domains to enable discriminative target features. Another line of ADA researches relies on two classifiers to measure the distribution discrepancy of two domains. The disagreement of two classifiers' predictions is utilized for detecting the non-discriminative features which do not clearly belong to some categories. By playing a mini-max game with a feature extractor, two classifiers optimize the decision boundaries for alleviating intra-class domain discrepancy [27, 38].

Although the above efforts promote a better distribution alignment, the inevitable equilibrium challenge of adversarial training still limits the performance of current ADA methods. To alleviate this problem, we take a further step on this line of research and propose a feature gradient distribution alignment method which can certificate a further distribution discrepancy reduction with a theoretical tighter error upper bound. It is noted that a concurrent work [6] coincides with our idea and investigates a similar approach. We independently exploit gradient alignment but from a different perspective and with different theoretical analysis.

3. Methodology

3.1. Preliminaries

For the vanilla unsupervised domain adaptation (UDA) task, we are given n_s labeled examples $\{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ from source domain $\mathcal{D}_s$ where $x_i^s \in \mathcal{X}_s, y_i^s \in \mathcal{Y}_s$ and n_t unlabeled examples $\left\{x_j^t\right\}_{j=1}^{n_t}$ from the target domain $\mathcal{D}_t$ where $x_j^t \in \mathcal{X}_t$. The goal of UDA is to learn a classification model to predict target domain labels $\left\{y_j^t\right\}_{j=1}^{n_t}$ where $y_j^t \in \mathcal{Y}_t$. The whole classification model, composed of a feature extractor $G(\cdot)$ and task classifier $C(\cdot)$, is expected to ensure a low target risk $\mathbb{E}_{(\mathbf{x}^t, y^t) \sim \mathcal{D}_t}\left[\mathcal{L}\left(C\left(G\left(\mathbf{x}^t\right)\right), y^t\right)\right]$ for a classification criterion $\mathcal{L}(\cdot, \cdot)$.

The feature extractor $G(\cdot)$ encodes $\boldsymbol{x}^s$ and $\boldsymbol{x}^t$ into a common feature space by $\boldsymbol{f}^s = G\left(\boldsymbol{x}^s\right)$, $\boldsymbol{f}^t = G\left(\boldsymbol{x}^t\right)$, where $G(\cdot)$ can be an arbitrary type of neural networks, and $\boldsymbol{f}^s, \boldsymbol{f}^t \in \mathbb{R}^D$ represent the D-dimensional feature vector for source and target domain. For a K-way classification task, $\boldsymbol{f}^s$ and $\boldsymbol{f}^t$ are fed into task classifier for prediction $\boldsymbol{z}^s = C\left(\boldsymbol{f}^s\right)$, $\boldsymbol{z}^t = C\left(\boldsymbol{f}^t\right)$, where $\boldsymbol{z}^s, \boldsymbol{z}^t \in \mathbb{R}^k$ are the score vectors. With the true labeled data in source domain, the classification model is trained by minimizing the standard cross-entropy loss:

$$\mathcal{L}_{\text{src}} = -\mathbb{E}_{\boldsymbol{x}^s \in \mathcal{X}_s, y^s \in \mathcal{Y}_s} \sum_{k=1}^{K} q_k \log \delta_k\left(\boldsymbol{z}^s\right), \tag{1}$$

where the $\delta_k(\boldsymbol{z}^s) = \frac{\exp(z_k^s)}{\sum_l \exp\left(z_l^s\right)}$ represents the k^{th} element in the softmax output of a vector $\boldsymbol{z}^s$, and q is the one-of-K encoding of y^s where q_k is '1' for the correct class and '0' for the rest. The goal is to predict labels of the target samples denoted as $\hat{y}^t = \operatorname{argmax}_k(\delta(\boldsymbol{z}_k^t))$.

3.2. Framework Overview

We propose a framework named feature gradient distribution alignment (FGDA) for reducing domain shift, as shown in Fig. 2. Our core component is the part of adversarial learning for aligning feature gradient distributions of two domains where the feature extractor and gradient discriminator compete with each other. To further promote our proposed method, we propose the Jacobian regularization term $\|J(\boldsymbol{f}^s)\|_F^2$ and self-supervised pseudo-labeling mechanism to improve the model generalization and pseudo-labels quality respectively. Moreover, we will indicate how to deploy our method on conventional ADA methods, e.g. DANN [8] CDAN [22] and MDD [38], for further reducing distribution discrepancy.

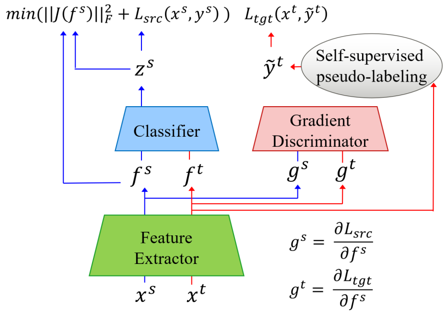

$min(||J(f^s)||_F^2 + L_{src}(x^s, y^s))$
$L_{tgt}(x^t, \tilde{y}^t)$
z^s
$\tilde{y}^t$
Self-supervised pseudo-labeling
Classifier
Gradient Discriminator
f^s
f^t
g^s
g^t
Feature Extractor
x^s
x^t
$g^s = \frac{\partial L_{src}}{\partial f^s}$
$g^t = \frac{\partial L_{tgt}}{\partial f^s}$

Figure 2. Structure of Feature Gradient Distribution Alignment. The whole structure is composed of several mechanisms: adversarial learning (classifier, gradient discriminator, and feature extractor), self-supervised pseudo-labeling, and Jacobian regularization $\|J(\boldsymbol{f}^s)\|_F^2$ where the last two mechanisms are used to promote adversarial learning.

3.3. Feature Gradient Distribution Alignment

To obtain the feature gradients of two domains, we need to compute the loss for target samples. During the training stage, we use classification model's predictions $\hat{y}^t$ as online pseudo-labels to calculate the loss for target sample $\mathcal{L}_{\text{tgt}}\left(C\left(G\left(\mathbf{x}^t\right)\right), \hat{y}^t\right)$. Note that the losses in the target domain are only used for gradient calculation rather than training the whole classification model.

With the back-propagation mechanism, the feature gradients vectors in source and target domain can be computed as:

$$\boldsymbol{g}(x^s, G) := \left[\frac{\partial \mathcal{L}_{\text{src}}}{\partial G(x^s)_1} \cdots \frac{\partial \mathcal{L}_{\text{src}}}{\partial G(x^s)_d} \cdots \frac{\partial \mathcal{L}_{\text{src}}}{\partial G(x^s)_D}\right], \tag{2}$$

$$\boldsymbol{g}(x^t, G) := \left[\frac{\partial \mathcal{L}_{\text{tgt}}}{\partial G(x^t)_1} \cdots \frac{\partial \mathcal{L}_{\text{tgt}}}{\partial G(x^t)_d} \cdots \frac{\partial \mathcal{L}_{\text{tgt}}}{\partial G(x^t)_D}\right], \tag{3}$$

where $G(x^s)_d$ and $G(x^t)_d$ represent the d^{th} elements of the feature vectors (f_d^s and f_d^t are used for convenience). $\boldsymbol{g}(x^s, G)$ and $\boldsymbol{g}(x^t, G)$ represent corresponding gradient vectors ($\boldsymbol{g}^s$ and $\boldsymbol{g}^t$ are used for convenience).

To achieve the goal of gradient distribution alignment, adversarial learning is adopted where the feature extractor and the discriminator (served as divergence estimator) compete with each other. Specifically, the discriminator is to predict the domain labels for the feature gradients of the source and target domain while the feature extractor learns to confuse the discriminator. When the equilibrium is reached, the minimal value of the feature distribution discrepancy is achieved.

The main objective of feature gradient alignment in our proposed method can be formulated as:

$$\min_{G} \max_{D_g} \mathcal{L}_{adv} = \mathbb{E}_{\boldsymbol{x}^t \in X^t}\left[\log D_g\left(g(\boldsymbol{x}^t, G)\right)\right] + \mathbb{E}_{\boldsymbol{x}^s \in X^s}\left[\log\left(1 - D_g\left(g(\boldsymbol{x}^s, G)\right)\right)\right] \tag{4}$$

where $D_g(\cdot)$ is the discriminator which outputs the probability that a gradient vector comes from the target domain.

The principle behind the proposed method can be seen in Fig. 3. In the beginning, due to the large domain shift, two sets of points from different domains are distributed very differently and feature gradients point to different parts of the highly complicated decision boundary in high dimensional space. Thus, most feature gradients of the two domains are obviously different leading that the feature gradient distribution discrepancy is large. Normally, the features which are close to each other or in the small region share similar gradients. Therefore, gradient alignment can enforce these two sets of points to move towards each other and stay in the small regions, *e.g.* red dashed circles in Fig. 3 where the gradients are similar. In other words, the domain shift can be reduced. To further show domain shift leads to large feature gradient discrepancy, we provide an analysis in Fig. 3, where the mean gradient distance for each class between two domains are plotted (note that normalization is applied, thus the distance reflects the variance of gradient direction). The plot shows gradient distribution discrepancy is decreased gradually during feature gradient alignment.

In comparison, for conventional feature-based ADA methods, as shown in Fig. 1, the domain shift is measured by the discriminator D_g: $dis = |E_{\boldsymbol{f} \sim \tilde{\mathcal{D}_\mathcal{S}}} D_g(\boldsymbol{f}) - E_{\boldsymbol{f} \sim \tilde{\mathcal{D}_\mathcal{T}}} D_g(\boldsymbol{f})|$ where $\tilde{\mathcal{D}_\mathcal{S}}$ and $\tilde{\mathcal{D}_\mathcal{T}}$ are induced distributions in feature space of $\mathcal{D}_s$ and $\mathcal{D}_t$. There exists a constant α such that $dis = \alpha|D_g(E_{\boldsymbol{f} \sim \tilde{\mathcal{D}_\mathcal{S}}} \boldsymbol{f}) - D_g(E_{\boldsymbol{f} \sim \tilde{\mathcal{D}_\mathcal{T}}} \boldsymbol{f})|$. When the mean features $E_{\boldsymbol{f} \sim \tilde{\mathcal{D}_\mathcal{S}}} \boldsymbol{f}$ and $E_{\boldsymbol{f} \sim \tilde{\mathcal{D}_\mathcal{T}}} \boldsymbol{f}$ are close, the domain shift measured by conventional method dis tends to be zero such that the corresponding gradients back propagated to feature extractor tends to be zeros. Therefore, the domain shift cannot be further reduced. Differently, for our method, the domain shift is measured by: $dis_{our} = |E_{\boldsymbol{f} \in \tilde{\mathcal{D}_\mathcal{S}}} D_g(\nabla_{\boldsymbol{f}} \mathcal{L}) - E_{\boldsymbol{f} \in \tilde{\mathcal{D}_\mathcal{T}}} D_g(\nabla_{\boldsymbol{f}} \mathcal{L})|$. Similarly, there exists a constant β such that $dis_{our} = \beta|D_g(E_{\boldsymbol{f} \sim \tilde{\mathcal{D}_\mathcal{S}}} \nabla_{\boldsymbol{f}} \mathcal{L}) - D_g(E_{\boldsymbol{f} \sim \tilde{\mathcal{D}_\mathcal{T}}} \nabla_{\boldsymbol{f}} \mathcal{L})|$. According to Fig. 1, even if the distance between two mean features is small, the domain shift measured by our method dis_{our} can be still large due to obvious different gradients in non-overlapping regions. Therefore, the domain shift can be further reduced.

3.4. Feature-level Jacobian Regularization

Several works showed that the discriminative features can help improve the performance of distribution alignment [30, 27, 35]. To learn more discriminative features, in this paper, the gradient regularization method is utilized for maximizing the classification margin [23, 13]. To this end, Hoffman *et al.* proposed to minimize the norm of an input-output gradient matrix, named Jacobian matrix [13]. Similarly, we adopt the Jacobian regularization at the feature level so that the feature extractor can learn more discriminative features far away from the decision boundary and the classifier can enlarge the classification margin simultaneously.

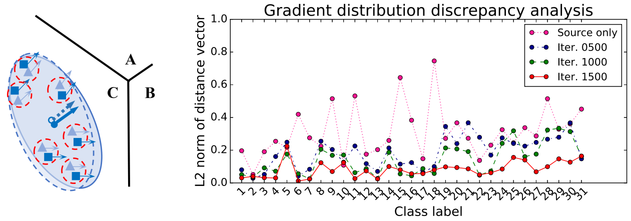


Figure 3. Geometric interpretation: Gradient alignment enforces features of two domains to stay in the small regions (red dashed circles) where the gradients are similar. Numerical analysis: Gradient discrepancy analysis on D→A task of office-31.

The input-output Jacobian matrix is defined as:

$$J_{k;d}(\boldsymbol{f}^s) \equiv \frac{\partial z_k}{\partial f_d^s}(\boldsymbol{f}^s), \tag{5}$$

where z_k^s and f_d^s denote the k-*th* score value $\boldsymbol{z}^s$ and d-*th* element of feature $\boldsymbol{f}^s$ in source domain. Then, Jacobian regularization is defined as:

$$\min_{G,C} L_{jr} = \|J(\boldsymbol{f}^s)\|_{\mathrm{F}}^2 \equiv \left\{\sum_{d,k} [J_{k;d}(\boldsymbol{f}^s)]^2\right\}. \tag{6}$$

3.5. Self-supervised Pseudo-labeling

Although employing online pseudo-labels predicted by the model for gradient alignment is feasible, the incorrect predictions will produce a sub-optimal gradient distribution that prevents the gradient alignment from achieving the optimal performance.

To obtain high-quality pseudo-labels in the target domain, we integrate an unsupervised approach, capturing the target distributions of different classes, termed self-supervised pseudo-labeling [18] into our framework. After the initial training stage, this strategy will be executed for every fixed number of iterations for generating an offline pseudo-label set $\widetilde{\mathcal{Y}}_t$ (offline pseudo-labels of the current sample are predicted with the feature centroids). Once offline pseudo-labels are obtained, $\tilde{y}_t \in \widetilde{\mathcal{Y}}_t$ will replace the online pseudo-label (online pseudo-labels are predictions for the current samples with classifier) $\hat{y}_t$ for feature gradient calculation of each target sample.

The centroid of feature distribution for each class is also considered as one prototype representation and its distribution should emerge in the region where a large number of samples are predicted confidently by the classifier. The update of the centroid is similar to k-means clustering by weighting each target feature with the classifier's confidence:

$$\boldsymbol{c}_k^{(0)} = \frac{\sum_{\boldsymbol{x}^t \in \mathcal{X}_t} \delta_k \left(\tilde{C}(\tilde{G}(\boldsymbol{x}^t)) \right) \tilde{G}(\boldsymbol{x}^t)}{\sum_{\boldsymbol{x}^t \in \mathcal{X}_t} \delta_k \left(\tilde{C}(\tilde{G}(\boldsymbol{x}^t)) \right)}, \tag{7}$$

where $\tilde{G}(\cdot)$ and $\tilde{C}(\cdot)$ have been trained in the previous iteration to predict online pseudo-labels. The offline pseudo-label of each target sample is assigned with the label of the nearest centroid:

$$\tilde{y}^t = \arg\min_k M_f \left(\tilde{G}\left(\boldsymbol{x}_t\right), \boldsymbol{c}_k^{(0)} \right), \tag{8}$$

where $M_f(\boldsymbol{a}, \boldsymbol{b})$ is cosine distance metric between $\boldsymbol{a}$ and $\boldsymbol{b}$. Finally, the category centroids will be determined according to new pseudo labels:

$$\begin{aligned} \boldsymbol{c}_k^{(1)} &= \frac{\sum_{\boldsymbol{x}^t \in \mathcal{X}_t} \mathbb{I}\left(\tilde{y}^t = k\right) \tilde{G}(\boldsymbol{x}_t)}{\sum_{\boldsymbol{x}^t \in \mathcal{X}_t} \mathbb{I}\left(\tilde{y}^t = k\right)}, \\ \tilde{y}^t &= \arg\min_k M_f \left(\tilde{G}\left(\boldsymbol{x}^t\right), \boldsymbol{c}_k^{(1)} \right), \end{aligned} \tag{9}$$

where $\mathbb{I}$ is a binary indicator function. When $\tilde{y}^t = k$, the function outputs 1. Within Eq. 9, the class centroids and pseudo-labels are alternately updated for multiple rounds. As observed practically, even these parameters are updated once, it can still enhance the quality of pseudo-labels.

3.6. Overall Learning Objective

To sum up the previous components, we present the total training loss in the following. At the initial training stage, the model is trained with feature gradient distribution alignment and feature-level Jacobian regularization. The corresponding objective can be formulated as:

$$\min_{G,C} \max_{D_g} \left(\mathcal{L}_{\text{src}} + \lambda_1 \mathcal{L}_{adv} + \lambda_2 \mathcal{L}_{jr} \right), \tag{10}$$

where $\lambda_1, \lambda_2 \geq 0$ are two balancing parameters. Initially, the online pseudo-labels $\hat{y}^t$ are employed for calculating feature gradient $\mathbf{g}^t$ of target samples to obtain the $\mathcal{L}_{adv}$ of Eq. 4. After a fixed number iteration, the self-supervised pseudo-labeling involves in training. We then change $\mathcal{L}_{adv}$ to $\tilde{\mathcal{L}}_{adv}$ where $\mathbf{g}^t$ is given by offline pseudo-labels $\tilde{y}^t$ and $\mathcal{L}_{\text{tgt}}\left(\cdot, \tilde{y}^t\right)$. In summary, the complete loss is shown as:

$$\min_{G,C} \max_{D_g} \mathcal{L}_{FGDA} = \mathcal{L}_{\text{src}} + \lambda_1 \tilde{\mathcal{L}}_{adv} + \lambda_2 \mathcal{L}_{jr}. \tag{11}$$

To show the advantage of our method, we couple proposed FGDA with some conventional feature-based ADA methods, such as DANN [8], CDAN [22] and MDD [38]. The simple way for the combination is adding a gradient discriminator directly and reusing their architectures. The combined training loss is formulated as:

$$\mathcal{L}_{FGDA+fada} = \mathcal{L}_{\text{src}} + \lambda_1 \tilde{\mathcal{L}}_{adv} + \lambda_2 \mathcal{L}_{jr} + \lambda_3 \mathcal{L}_{fada}, \tag{12}$$

where $\mathcal{L}_{fada}$ is the adversarial loss of feature-based ADA model and λ_3 is its balancing parameter.

3.7. Model Analysis

In this section, we provide both the theoretical and empirical analysis for the proposed FGDA method. We first consider the feature $\boldsymbol{f} = G(\boldsymbol{x})$ and a family of source classifiers C on a fixed representation space $\mathcal{F}$, and hypothesis space $\mathcal{H}$ respectively. The error of a hypothesis $C \in \mathcal{H}$ on source domain is $\epsilon_S(C) = \mathrm{E}_{\boldsymbol{f} \sim \tilde{\mathcal{D}_S}} \left[C(\boldsymbol{f}) \neq y \right]$, where $\tilde{\mathcal{D}_S}$ denotes the induced feature distribution of source data distribution $\mathcal{D}_S$ and y is the label of feature f. The disagreement between hypotheses $C_1, C_2 \in \mathcal{H}$ is given by $\epsilon_S(C_1, C_2) = \mathrm{E}_{\boldsymbol{f} \sim \tilde{\mathcal{D}_S}} \left[C_1(\boldsymbol{f}) \neq C_2(\boldsymbol{f}) \right]$. To estimate distribution divergence from unlabeled data, one ideal joint hypothesis that minimizes the combined error on two domains is introduced as $C^* = \operatorname{argmin}_C \epsilon_S(C) + \epsilon_T(C)$. Then, the probabilistic bound of target error is given by

$$\epsilon_T(C) \leq \epsilon_S + \lambda + \left| \epsilon_T\left(C, C^*\right) - \epsilon_S\left(C, C^*\right) \right|, \tag{13}$$

where $\lambda = \epsilon_S\left(C^*\right) + \epsilon_T\left(C^*\right)$ is the error of the ideal joint hypothesis.

To prove the efficacy of our proposed method FGDA, we show that our method can obtain a tighter upper bound for the target domain error than conventional domain adaptation methods. The main theory is presented in Theorem 1 and 2.

Theorem 1 Let G be a fixed representation function from $\mathcal{X}$ to $\mathcal{F}$, and $\mathcal{H}$ be a hypothesis space of VC-dimension d. If a random labeled sample of size m is generated by applying G to a $\mathcal{D}_s$ - i.i.d. The feature f is drawn from $\tilde{\mathcal{D}_S}$ or $\tilde{\mathcal{D}_T}$ with corresponding label y. Denote that $\tilde{\mathcal{U}_S}$, $\tilde{\mathcal{U}_T}$ are the set of unlabeled samples of size m' each, drawn from $\tilde{\mathcal{D}_S}$ and $\tilde{\mathcal{D}_T}$ respectively. Then with the probability at least $1 - \delta$ (over the choice of the samples), for every $C \in \mathcal{H}$:

$$\begin{aligned} \epsilon_T(C) \leq \quad & \hat{\epsilon}_S(C) + \lambda + d_\nabla \left(\tilde{\mathcal{U}_S}, \tilde{\mathcal{U}_T} \right) \\ & + \frac{4}{m} \sqrt{\left(d \log \frac{2em}{d} + \log \frac{4}{\delta} \right)} \\ & + 4 \sqrt{\frac{d \log\left(2m'\right) + \log\left(\frac{4}{\delta}\right)}{m'}} \\ & = const + d_\nabla \left(\tilde{\mathcal{U}_S}, \tilde{\mathcal{U}_T} \right) \end{aligned} \tag{14}$$

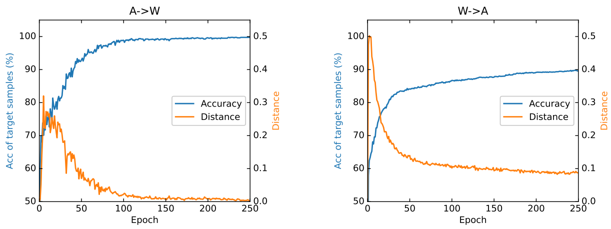


Figure 4. Test accuracy and ∇-distance during training progress on A→W and W→A of Office-31.

where $\hat{\epsilon}_S(C)$ is empirical error of source samples, λ is a very small constant, e represents the base of the natural logarithm, $d_\nabla\left(\tilde{\mathcal{U}_S},\tilde{\mathcal{U}_T}\right) = a\sup_{D_g\mathcal{H}_D}\left|E_{\boldsymbol{f}\in\tilde{\mathcal{U}_S}}D_g(\nabla_{\boldsymbol{f}}\mathcal{L}) - E_{\boldsymbol{f}\in\tilde{\mathcal{U}_T}}D_g(\nabla_{\boldsymbol{f}}\mathcal{L})\right|$ is the introduced ∇-distance, D_g is the discriminator and $a = \frac{1}{\min_{C(\boldsymbol{f})\in[0,1]}\nabla_C\mathcal{L}(C(\boldsymbol{f}),y)}$. Here $\mathcal{L}(\cdot)$ denotes the loss function.

Theorem 2 When $a \leq 1$, our method can obtain a tighter upper bound than traditional domain adaptation methods:

$$const + d_\nabla\left(\tilde{\mathcal{U}_S},\tilde{\mathcal{U}_T}\right) \leq const + d_\mathcal{H}\left(\tilde{\mathcal{U}_S},\tilde{\mathcal{U}_T}\right), \text{where}$$

$d_\mathcal{H}\left(\tilde{\mathcal{U}_S},\tilde{\mathcal{U}_T}\right) = \sup_{D_g\in\mathcal{H}_D}\left|E_{\boldsymbol{f}\in\tilde{\mathcal{U}_S}}D_g(\boldsymbol{f}) - E_{\boldsymbol{f}\in\tilde{\mathcal{U}_T}}D_g(\boldsymbol{f})\right|$. Proof of Theorem 1-2 can be seen in the supplementary document.

Theorem 1-2 show that feature gradient distribution discrepancy can help bound the test error. In other words, feature gradient alignment can certificate to reduce the test error. More importantly, a tighter bound ensures the superiority of our proposed method to the conventional domain adaptation methods.

From the empirical perspective, it may however be difficult to calculate the ∇-distance directly. In this work, we resort to the domain discriminator D_g to approximate it. Specifically, D_g attempts to distinguish which domain feature gradient $\nabla_{\boldsymbol{f}}\mathcal{L}$ comes from. $\mathcal{H}_g$ is a ∇ hypothesis space over the feature gradient $\nabla_{\boldsymbol{f}}\mathcal{L}$. Hence, the domain discrepancy $\left|\epsilon_T\left(C, C^*\right) - \epsilon_S\left(C, C^*\right)\right|$ can be upper bounded by ∇-distance. The detail proof can be found in the supplemental material.

We further show the correlation between ∇-distance and test error in Fig. 4. We consider an ideal situation where all the feature gradients of target samples are calculated with their true target labels. As observed, there is an explicit negative correlation between them. The decrease of ∇-distance leads that the test accuracy increases directly for different training epochs consistently. Until the convergence of ∇-distance, the accuracy of A→W and W→A are 100% and 90% respectively. It indicates that ∇-distance is highly correlated with the test error, and aligning feature gradient distribution between two domains is analogous to train the model with the target pseudo-labels softly.

4. Experiments

4.1. Datasets

Office-31. Office-31 is a widely used dataset for evaluating visual domain adaptation algorithms. It includes 4,652 images and 31 categories, which are collected from three distinct domains: Amazon (A), Webcam (W), and DSLR (D). We evaluate all the methods on six transfer tasks A→W, D→W, W→D, A→D, D→A, and W→A.

Office-Home. Office-Home is a more challenging dataset. It consists of 15,500 images in 65 classes in office and home settings. On four extremely dissimilar domains: Artistic images (A), Clip Art (C), Product images (P), and Real-World images (R), we evaluate all the transfer tasks.

Implementation Details. Following standard evaluation protocols for UDA [3], all labeled source and unlabeled target instances are used as training data. For fair comparisons, we exploit the same network structure as the compared methods. Concretely, ResNet-50 [12] is used as the backbone network in all the experiments, and a linear layer followed by the softmax function is taken as a category classifier. For our gradient discriminator, it consists of two hidden layers that are a fully connected layer followed by ReLu activation function and BatchNorm layer, and a domain classifier that linearly transforms the hidden feature and then activates it with the sigmoid function. The adversarial learning algorithm is implemented similar to the original DANN [8], where a reverse gradient layer is applied on feature gradients. In Eq. 11, when FGDA is examined individually, λ_1 is set to 1. When FGDA is combined with MDD [38] as shown in Eq. 12, λ_3 is fixed as 0.5, and $\lambda_1 = 1$ and $\lambda_1 = 0.5$ are used for Office-31 and Office-Home respectively. For λ_2, it is searched from $[0.05, 0.10, 0.15, 0.20, 0.25]$ so as to achieve the best results.

Method	A→W	D→W	A→D	D→A	W→A	Avg
ResNet-50 [12]	68.4	96.7	68.9	62.5	60.7	76.1
DANN [8]	82.0	96.9	79.7	68.2	67.4	82.2
ADDA [35]	86.2	96.2	77.8	69.5	68.9	82.9
MADA [25]	90.0	97.4	87.8	70.3	66.4	85.2
CDAN [22]	94.1	98.6	92.9	71.0	69.3	87.7
MCDDA [27]	82.6	98.9	84.3	66.2	66.3	83.0
MDD [38]	94.5	98.4	93.5	74.6	72.2	88.9
SymNets [22]	90.8	98.8	93.9	74.6	72.5	88.4
ALDA [4]	95.6	97.7	94.0	72.2	72.5	88.7
DADA [30]	92.3	**99.2**	93.9	74.4	74.2	89.0
DCAN [17]	95.0	97.5	92.6	77.2	74.9	89.5
GSDA [15]	**95.7**	99.1	94.8	73.5	74.9	89.7
FGDA	93.3	99.1	93.2	73.2	72.7	88.6
FGDA+MDD	95.1	98.7	**95.4**	**78.1**	**76.5**	**90.6**

Table 1. Accuracy (%) on Office-31 for UDA (ResNet-50)

Method	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Avg
ResNet-50 [12]	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
DANN [8]	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
CDAN [22]	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 [39]	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
MDD [38]	54.9	73.7	77.8	60.0	71.4	71.8	61.2	53.6	78.1	72.5	60.2	82.3	68.1
ALDA [4]	53.7	70.1	76.4	60.2	72.6	71.5	56.8	51.9	77.1	70.2	56.3	82.1	66.6
DCAN [17]	54.5	75.7	**81.2**	67.4	74.0	**76.3**	**67.4**	52.7	80.6	74.1	59.1	83.5	70.5
GSDA [15]	**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
FGDA	52.3	77.0	78.2	64.6	75.5	73.7	64.0	49.5	80.7	70.1	52.3	81.6	68.3
FGDA+MDD	57.1	**77.5**	81.0	**68.4**	**77.2**	75.9	65.8	**55.8**	**81.0**	74.3	60.5	**83.6**	**71.5**

Table 2. Accuracy (%) on Office-Home for UDA (ResNet-50)

4.2. Results

We compare the proposed method with a number of previous ADA methods. Results are reported in Table 1 and 2. All the results of W→D on Office-31 are hidden, but participate in the calculation of the average result, because most of models achieve 100% on this task. MADA [25], CDAN [22] and GSDA [15] condition on a single or multiple discriminators on classifier outputs to improve DANN [8]. SymNets [39] and DADA [30] further achieve domain discrimination and confusion by relying on classifier outputs and creative adversarial learning mechanisms. MCDDA [27] and MDD [38] consider disagreement of two classifiers for alleviating intra-class distribution discrepancy. As observed, FGDA achieves competitive results with MDD, which indicates that applying feature gradient alignment individually is feasible for distribution discrepancy reduction. Furthermore, FGDA+MDD outperforms the best results of the comparison models by 0.9% on Office-31 dataset and by 1.0% on Office-Home dataset respectively.

5. Further Analysis

5.1. Ablation Study

Our study starts with setting a non-adaptation model as the first baseline, which simply fine-tunes RestNet-50 [12] on source data. To demonstrate the advantage of our method towards representation-based ADA method, DANN [8], CDAN-E [22] and MDD [38] will be respectively combined with our method and considered as the other baselines. To investigate how high-quality pseudo-labels would improve the performance, we remove the self-supervised pseudo-labeling (SPL) method from our approach; to examine performance of the gradient regularization, we also remove the feature-level Jacobian regularization (FJR); to validate the feasibility of feature gradient alignment for reducing the distribution divergence, we remove both FJR and SPL. To observe the influence of loss function, when only feature gradient alignment is applied, the loss function for calculating gradients in the target domain is changed from cross-entropy to conditional entropy minimization (Ent.).

As reported in Table 3, with the same network structure, FGDA (w/o FJR, SPL) improves over DANN, showing that employing gradient as the representation is not only able to reduce distribution divergence, but also results in a better distribution alignment. FGDA (w/o FJR) and FGDA (w/o SPL) are observed to be able to improve over FGDA (w/o FJR, SPL) significantly , which demonstrates the effectiveness of our each component. Meanwhile, FGDA (w/ Ent.) applied on DANN still leads to significant performance gain though not as evident as FGDA+DANN.

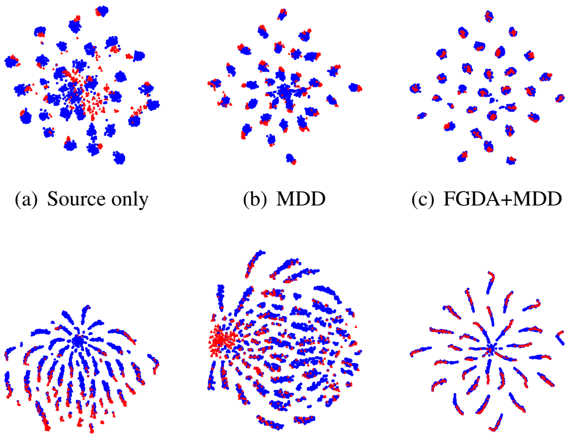
Regarding FGDA, it leads to comparable performance with MDD and outperforms DANN and CDAN in a large margin. Moreover, when FGDA is combined with DANN, CDAN-E, and MDD respectively, the results confirm that unitizing gradient as representation for measuring the distribution divergence can indeed improve representation-based ADA consistently.

Method	A-W	D-W	A-D	D-A	W-A	Avg
ResNet-50 [12]	68.4	96.7	68.9	62.5	60.7	76.1
DANN [8]	82.0	96.9	79.7	68.2	67.4	82.2
MDD [38]	94.5	98.4	93.5	74.6	72.2	88.9
CDAN-E [22]	94.1	98.6	92.9	71.0	69.3	87.7
FGDA	93.3	99.1	93.2	73.2	72.7	88.6
w/o FJR	93.5	98.6	93.2	72.7	71.8	88.3
w/o SPL	91.3	98.6	89.8	69.1	67.2	86.0
w/o FJR, SPL	89.2	98.0	89.2	69.0	64.8	85.0
w/ Ent. +DANN	94.8	98.4	90.6	71.6	70.1	87.6
FGDA+DANN	92.6	**99.1**	94.2	73.9	73.7	88.9
FGDA+CDAN-E	92.6	98.7	95.0	74.7	74.4	89.2
FGDA+MDD	**95.1**	98.7	**95.4**	**78.1**	**76.5**	**90.6**

Table 3. Accuracy (%) on Office-31 for UDA (ResNet-50)

5.2. Visualization Analysis

To have an intuitive understanding of FGDA and show that gradient alignment can surely help reduce the distribution discrepancy, we visualize the features and their gradients of the source and target domains on Office-31 with t-SNE [36]. As seen in Fig. 5, in ResNet-50 (source only), both feature and gradient distributions of source and target domains are distributed differently due to the large domain shift. When MDD is employed to align feature distributions, the gradient distribution discrepancy is still obvious, even though the feature distribution discrepancy has


(a) Source only (b) MDD (c) FGDA+MDD

(d) Source only (e) MDD (f) FGDA+MDD

Figure 5. t-SNE visualizations of feature distribution and gradient distribution for source only model (RestNet-50), MDD and FGDA+MDD on task A→W of Office-31. Blue and red points denote the source and target domain samples respectively.

already reduced. By contrast, FGDA+MDD shows superiority over MMD with smaller distribution discrepancies for both the features and gradients. Moreover, it is observed that the target features are as discriminative as source features, where the target distribution is inter-class separated and intra-class clustered after feature gradient alignment is conducted. Note that the visualization results for CDAN-E [22], FGDA (w/o FJR, SPL), and FGDA are provided in the supplementary document.

5.3. Sensitivity Analysis

In Table 4, we empirically show the influence of the balancing parameter λ_2 of FJR in FGDA (w/o SPL). By choosing λ_2 from $[0.05, 0.10, 0.15, 0.20, 0.25]$, all results of Office-31 are presented for sensitivity analysis. Although there is not a consistent trend for setting λ_2, we observe that the chosen parameter range, from 0.05 to 0.25, could generally cover most of the best results. Sensitivity analysis on other parameters λ_1 and λ_3 can be seen in the supplementary document.

λ_2	A-W	D-W	A-D	D-A	W-A
0.005	85.0	**98.6**	87.1	68.2	65.7
0.010	**91.3**	98.5	88.0	68.9	67.0
0.015	89.4	98.6	88.4	68.9	67.0
0.020	87.4	98.5	88.2	**69.1**	**67.2**
0.025	89.8	98.6	**89.8**	68.8	67.0

Table 4. Accuracy (%) of FGDA (w/o SPL) on Office-31 (ResNet-50)

To investigate how pseudo-label noise in the target domain affects the performance, a case study for FGDA+MDD is conducted in Fig. 6. Notably, FJR, SPL, and MDD are applied after 900 iterations. Initially, even

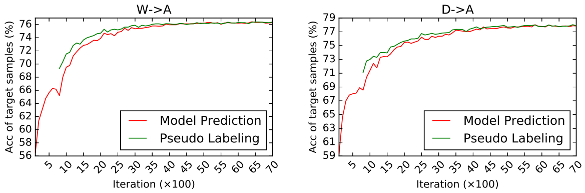


Figure 6. Accuracy of model prediction and pseudo-labelling of target domain on Office-31 for FGDA+MDD.

without high-quality pseudo-labeling, gradient alignment appears effective as well. Once SPL is involved, a more accurate gradient distribution of the target domain promotes the performance obviously. When training with gradient alignment, a more separable feature space for target samples is obtained so that the accuracy of pseudo-labeling and model prediction increase alternately. Thus, gradient alignment can benefit from pseudo-labeling and vice versa. Additionally, there are other methods to achieve high quality pseudo-labels, such as the temporal ensembling approach named mean teacher model [31].

6. Conclusion

In this work, we show that present adversarial domain adaptation methods have an inherent drawback in which even if the discriminator is fully confused, sufficient similarity between two distributions cannot be guaranteed. To cope with this problem, we propose a novel method named feature gradient distribution alignment which can certificate a further distribution discrepancy reduction between the source and target domain. We show that the distribution discrepancy can be reduced by aligning feature gradients theoretically and empirically. More importantly, our proposed novel framework enjoys a theoretical guarantee that a tighter error upper bound on the target domain can be attained than that of the existing adversarial domain adaptation methods. Extensive experiments validate that our proposed novel framework can achieve state-of-the-art performance on two real-world benchmark data quantitatively and qualitatively.

Acknowledgement

We would like to thank Dr. Rui Zhang from School of Science, XJTLU for the valuable discussion as well as theoretical analysis on the research in this paper. The work was partially supported by the following: National Natural Science Foundation of China under no.61876155 and 61876154; Jiangsu Science and Technology Programme (Natural Science Foundation of Jiangsu Province) under no. BE2020006-4B, K20181189, BK20181190; Key Program Special Fund in XJTLU under no. KSF-T-06, KSF-E-26.

References

[1] Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. Towards better understanding of gradient-based attribution methods for deep neural networks. In *6th International Conference on Learning Representations*, 2018.

[2] Sanjeev Arora, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. Generalization and equilibrium in generative adversarial nets (gans). In *International Conference on Machine Learning*, pages 224–232. PMLR, 2017.

[3] Ruichu Cai, Zijian Li, Pengfei Wei, Jie Qiao, Kun Zhang, and Zhifeng Hao. Learning disentangled semantic representation for domain adaptation. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 2060–2066. ijcai.org, 2019.

[4] Minghao Chen, Shuai Zhao, Haifeng Liu, and Deng Cai. Adversarial-learned loss for domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3521–3528, 2020.

[5] Corinna Cortes, Mehryar Mohri, Michael Riley, and Afshin Rostamizadeh. Sample selection bias correction theory. In *International conference on algorithmic learning theory*, pages 38–53. Springer, 2008.

[6] Zhekai Du, Jingjing Li, Hongzu Su, Lei Zhu, and Ke Lu. Cross-domain gradient discrepancy minimization for unsupervised domain adaptation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

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

[8] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27:2672–2680, 2014.

[10] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations, year = 2015,*.

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[13] Judy Hoffman, Daniel A Roberts, and Sho Yaida. Robust learning with jacobian regularization. *arXiv preprint arXiv:1908.02729*, 2019.

[14] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. PMLR, 2018.

[15] Lanqing Hu, Meina Kan, Shiguang Shan, and Xilin Chen. Unsupervised domain adaptation with hierarchical gradient synchronization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4043–4052, 2020.

[16] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[17] Shuang Li, Chi Liu, Qiuxia Lin, Binhui Xie, Zhengming Ding, Gao Huang, and Jian Tang. Domain conditioned adaptation network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11386–11393, 2020.

[18] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *Proceedings of the 37th International Conference on Machine Learning*, pages 6028–6039. PMLR, 2020.

[19] Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In *International conference on machine learning*, pages 3122–3130. PMLR, 2018.

[20] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *Advances in neural information processing systems*, pages 700–708, 2017.

[21] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[22] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *Advances in neural information processing systems*, pages 1640–1650, 2018.

[23] Chunchuan Lyu, Kaizhu Huang, and Hai-Ning Liang. A unified gradient regularization family for adversarial examples. *2015 IEEE International Conference on Data Mining*, pages 301–309, 2015.

[24] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1520–1528, 2015.

[25] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[26] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

[27] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3723–3732, 2018.

[28] Swami Sankaranarayanan, Yogesh Balaji, Carlos D Castillo, and Rama Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8503–8512, 2018.

[29] Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(5), 2007.

[30] Hui Tang and Kui Jia. Discriminative adversarial domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5940–5947, 2020.

[31] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, pages 1195–1204, 2017.

[32] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE, 2011.

[33] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7472–7481, 2018.

[34] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4068–4076, 2015.

[35] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017.

[36] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[37] Shufei Zhang, Zhuang Qian, Kaizhu Huang, Qiufeng Wang, Rui Zhang, and Xinping Yi. Towards better robust generalization with shift consistency regularization. In *Proceedings of the 38th International Conference on Machine Learning*, pages 12524–12534. PMLR, 2021.

[38] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*, pages 7404–7413, 2019.

[39] Yabin Zhang, Hui Tang, Kui Jia, and Mingkui Tan. Domain-symmetric networks for adversarial domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5031–5040, 2019.

[40] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.