Structure Preserving Generative Cross-Domain Learning

Haifeng Xia[♯], Zhengming Ding[†]
[♯]Department of ECE, Indiana University-Purdue University Indianapolis
[†]Department of CIT, Indiana University-Purdue University Indianapolis
{haifxia,zd2}@iu.edu

Abstract

Unsupervised domain adaptation (UDA) casts a light when dealing with insufficient or no labeled data in the target domain by exploring the well-annotated source knowledge in different distributions. Most research efforts on UDA explore to seek a domain-invariant classifier over source supervision. However, due to the scarcity of label information in the target domain, such a classifier has a lack of ground-truth target supervision, which dramatically obstructs the robustness and discrimination of the classifier. To this end, we develop a novel Generative cross-domain learning via Structure-Preserving (GSP), which attempts to transform target data into the source domain in order to take advantage of source supervision. Specifically, a novel cross-domain graph alignment is developed to capture the intrinsic relationship across two domains during target-source translation. Simultaneously, two distinct classifiers are trained to trigger the domain-invariant feature learning both guided with source supervision, one is a traditional source classifier and the other is a source-supervised target classifier. Extensive experimental results on several cross-domain visual benchmarks have demonstrated the effectiveness of our model by comparing with other state-of-the-art UDA algorithms.

1. Introduction

Deep neural networks have achieved an increasing number of successes in computer vision community with a great deal of well-labeled data, which allows deep learning models to easily capture abstract and complex relationship between feature and category [44]. In reality, however, collecting abundant data with annotation becomes too difficult and expensive in many learning tasks. The intuitive motivation to address the realistic issue is to apply knowledge extracted from model trained with available annotated samples into target tasks. Such a strategy frequently tends to be vulnerable for the problem of domain shift [11] as the trained model is more likely to be invalid when assessed on unlabeled target domain having various distribution with training source. Specifically, for visual data, domain shift results from distinctions of light condition occlusions and background [4].

Unsupervised Domain Adaptation (UDA) is a promising technique to train a model obtaining lower risk when evaluated on target domain [13, 8, 9, 41, 18]. Existing UDA methods [28, 21, 7] generally minimize the risk on source data firstly and then employ appropriate statistical property to eliminate cross-domain discrepancy. There are two common manners to measure discrepancy between distributions of two domains, i.e., discrepancy measurement [20, 22] and domain adversarial confusion [44, 19]. Specifically, discrepancy measurement like maximum mean discrepancy employs statistical indication (mean of distribution) to measure cross-domain difference and aligns the distribution of two domains by constraining this indication. While domain adversarial confusion aims to seek a domain invariant feature generator for both domains with a domain confusion discriminator in an adversarial training manner. However, these methods still remain restrictive in the alignment between feature and category due to the neglect of class-level information [23]. They generally suffer from two challenging issues: 1) mis-alignment of cross-domain samples from various classes and 2) the learned classifier would lack of generalization on target domain [14].

To alleviate these disadvantages, target pseudo labels are introduced to effectively enhance class-level alignment during the training process [42, 37]. Moreover, [40] considers the class prior probability defined on two domains as class-specific weight and modifies original MMD with auxiliary weights to promote discriminative ability of classifier for target domain. Similarly, a novel metric measure formulated in [14] includes intra-class domain discrepancy and inter-class domain discrepancy. On the other hand, recent studies [25, 15] pay more attention to the second issue, which attempts to make the learned decision boundary robust for target domain. The common strategy to address this challenge designs two domain-specific classifiers. Subsequently, [30] regards two classifiers as various views for

the same samples of source domain and maximum their distinction to learn a robust classifier for samples from target domain. In addition, [16] develops sliced wasserstein discrepancy (SWD) connecting feature distribution alignment and wasserstein metric to promote the discrimination of target classifier. However, training a target-specific classifier with samples from corresponding domain is inaccessible, which certainly obstructs classification accuracy. This issue stems from the inaccessibility of target label.

In this paper, we propose a Generative cross-domain learning via Structure-Preserving (GSP) model to incorporate samples of target domain into training phase with source supervision (Fig. 1). Specifically, a novel metric discrepancy is defined to measure cross-domain distinction in terms of the topological structure including information of node and edge. In order to minimize cross-domain discrepancy, two-level alignments (i.e., edge-level and node-level) are designed to enhance the mitigation of domain mismatch. The edge-level alignment aims to discover matching relationship between two domains according to node and degree, while the node-level alignment exploits learned matching relationship to restrict feature representation across two domains. Moreover, we develop a source-supervised target classifier which supervises feature learning of target domain with source label. Furthermore, we adopt a symmetrical and adversarial manner to train two domain-specific classifiers, which not only maximize the difference between two classifiers but also extract effective domain invariant features. To this end, our contributions are summarized as following:

- We introduce a novel metric measure in terms of graph distribution and formulate alignments of node-level and edge-level. The edge-level alignment is employed to extract cross-domain matching relation, while node-level operation aims to align feature representation.
- To promote the discriminative ability of classifier, we develop source-supervised target classifier fed with the combination of matching relation and features from target domain. Moreover, we apply symmetric adversarial manner to train two domain-specific classifiers.
- We evaluate our proposed model (GSP) on several visual cross-domain benchmarks. GSP approach outperforms competitive methods in most domain adaptation tasks, demonstrating the effectiveness of solving UDA problem. Extensive analysis illustrates the function of each component in GSP method.

2. Related Work

Gromov-Wasserstein discrepancy (GW) is considered as an effective tool to measure the difference between two spaces [38]. Given two compact metric spaces $(\mathcal{S}, d_s, p_s)$ and $(\mathcal{T}, d_t, p_t)$ where d_s and d_t are two independent metric measures defined on $\mathcal{S}$ and $\mathcal{T}$, respectively, $p_s \in \mathbb{R}^{|\mathcal{S}|}$ ($p_s \mathbb{1}^{|\mathcal{S}|} = 1$) represents a Borel probability measure defined on $\mathcal{S}$ (p_t has the same meaning with p_s), the p-th order Gromov-Wasserstein discrepancy ($p \in [1, \infty)$) has the following formulation:

$$d_{gw} := \inf_{\pi \in \Pi(p_s, p_t)} \Big(\int \int_{\mathcal{S} \times \mathcal{T}} L(d^s_{ij}, d^t_{i'j'})^p d\pi_{ii'} d\pi_{jj'} \Big)^{\frac{1}{p}}, \tag{1}$$

where $L(d^s_{ij}, d^t_{i'j'}) = |d_s(s_i, s_j) - d_t(t_{i'}, t_{j'})|$, and the set of all probability measure is $\Pi(p_s, p_t)$ drawn from $\mathcal{S} \times \mathcal{T}$ with the marginal distributions p_s and p_t. From the above formulation, the loss function in GW discrepancy firstly measures distance between pairs of samples within each compact space and then compares these distances in $\mathcal{S}$ with those in another space $\mathcal{T}$. Due to the property that measure difference between various spaces, GW metric has been successfully applied to measure discrepancy between various graphs [1, 26]. In addition, [38] has theoretically proved that GW discrepancy is a pseudo-metric of graph. Different from these applications only learning matching relation between various graphs, our work not only introduces node-level alignment into learning process but also exploits the learned relation to minimize domain discrepancy. Concretely, GW discrepancy is extended to measure cross-domain distinction in terms of graph distribution and we formulate the novel metric measure as edge-level alignment. According to cross-domain feature representation, we incorporate graph matching relation produced from edge-level alignment into node-level alignment which directly constraints cross-domain feature learning to eliminate domain discrepancy.

Unsupervised Domain Adaptation (UDA) aims to employ robust and generalized model to promote the performance on target domain. Among them, domain-invariant feature learning attempts to generate discriminative feature when aligning distributions of two domains in unsupervised manner [44]. The typical methods to bound the discrepancy of two domains are categorized into two types: domain adversarial training [21, 32], and maximum mean discrepancy [14, 20]. The first category attempts to explore adversarial manner to generate the same feature space for source and target domains, while the other group further constraints properties of generated feature distribution. Specifcally, [20, 22] pay effort to bound target risk by minimizing the difference of distribution mean. In addition, [24, 19, 3] adopt generative adversarial manner to train network architecture. When achieving equilibrium, network system synthesizes domain-invariant feature confusing the discriminator. Moreover, [44] proposes domain symmetric networks (SymNets) incorporating classifier and discriminator into a single frame and training network in symmetric adversarial way. Compared to SymNets, our method GSP introduces feature of target domain into classifier with the supervision of source label. And the symmetric adversarial

training manner is mainly exploited to maximum distinction between two domain-specific classifiers.

3. The Proposed Algorithm

3.1. Preliminaries and Motivation

For UDA, we are generally given source dataset $\mathcal{D}_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ and target dataset $\mathcal{D}_t = \{x_i^t\}_{i=1}^{n_t}$ where $\mathcal{D}_s$ includes n_s data samples $\{x_i^s\}_{i=1}^{n_s}$ with its corresponding label set $\{y_i\}_{i=1}^{n_s}$, and $\mathcal{D}_t$ consists of n_t data instances $\{x_i^t\}_{i=1}^{n_t}$ and the label information for target domain is unknown. Although it is obvious that the same label space is shared by these two domains, the distributions of their data sample sets are different, which limits the performance of the trained model from source to target domain. Minimizing the source risk and bounding the discrepancy between two various distributions effectively improve the performance of model, which has been verified by abundant theoretical analyses.

In this work, we rethink UDA problem from perspective of graph distribution and propose a novel generative model with structure preserving. Concretely, samples within each domain constitute graph structure with information of node, edge and degree. Although there is distribution discrepancy across two domains, topological structures of them are more likely to be similar. Thus, the proposed method matches topological information across two domains through Gromov-Wasserstein (GW) discrepancy [38] defined over graph and leverages the learned relationship to eliminate discrepancy between $\mathcal{D}_s$ and $\mathcal{D}_t$ with cross-domain graph alignment. In addition, we develop a novel source-supervised target classifier jointly with cross-domain alignment to make the trained classifier robust to unlabeled target learning.

3.2. Cross-Domain Generation via Structure Preserving

3.2.1 Cross-Domain Graph Alignment

Existing approaches [17, 5] achieve promising performance by benefiting from deep neural networks, e.g., VGG [31] and ResNet [12]. Those algorithms explore existing deep neural networks as backbone to extract general feature representation and stack cross-domain alignment at the top. Suppose $F_s = \{f_i^s\}_{i=1}^{n_s}$ and $F_t = \{f_j^t\}_{j=1}^{n_t}$ are feature representations from two domains $\mathcal{D}_s$ and $\mathcal{D}_t$, respectively. With extracted features, we define the measurable graphs of source domain and target domain as $G_s(\mathcal{V}_s, A_s, p_s)$ and $G_t(\mathcal{V}_t, A_t, p_t)$, where $\mathcal{V}_s = \{v_i\}_{i=1}^{n_s}$ ($\mathcal{V}_t$) is the set of nodes in the corresponding domain, the similarity or distance between elements in source domain (target domain) is denoted as $A_s = [a_{ij}^s] \in \mathbb{R}^{n_s \times n_s}$ (A_t), and $p_s(p_t)$ represents Borel probability measurement defined on $\mathcal{V}_s(\mathcal{V}_t)$. In practice, p_s (p_t) represents empirical distribution of nodes and it is estimated by normalized node degree.

To effectively match two different domains, we propose two-level cross-domain alignments, i.e., node-level and edge-level. First of all, we explore GW distance to measure the edge similarity across two domains [33]. Metric measures of source domain and target domain are defined as d_s, d_t, respectively. In term of these definitions, we extend GW method to measure the discrepancy of cross-domain topology structure and have the following formulation of edge-level alignment $\mathcal{L}_e$:

$$\begin{aligned}\mathcal{L}_e &= \Big(\sum_{i,j\in\mathcal{V}_s}\sum_{i',j'\in\mathcal{V}_t} |A_{ij}^s - A_{i'j'}^t| A_{i,i'}^{st} A_{j,j'}^{st}\Big)^{\frac{1}{p}} \\ &= \langle L(A_s, A_t, A_{st}), A_{st}\rangle,\end{aligned} \tag{2}$$

where $A_{st} = \{A_{st} \in \mathbb{R}_+^{n_s \times n_t} | A_{st}\mathbb{1}_{n_t} = p_s, A_{st}^{\mathrm{T}}\mathbb{1}_{n_s} = p_t\}$ is the joint distribution of node degree, i.e., $A_{st} \in \Pi(p_s, p_t)$, $L(A_s, A_t, A_{st}) = A_s p_s \mathbb{1}_{n_t}^{\mathrm{T}} + \mathbb{1}_{n_s} p_t^{\mathrm{T}} A_t^{\mathrm{T}} - 2A_s A_{st} A_t^{\mathrm{T}}$ is derived from [26], and $\langle A, B\rangle$ is the inner product of matrices A and B.

To further mitigate the domain mismatch, we bridge the node-level domain gap. In practice, v_i^s (v_j^t) can be represented by the feature f_i^s (f_j^t). Targeting at coupling the relationship between features from various domains, we further exploit the learned structured information to constrain feature representation and reduce discrepancy of two domains. In addition, A_{ij}^{st} also indicates the probability that v_i^s and v_j^t belong to the same category. Thus, we define the node-level alignment as $\mathcal{L}_n$:

$$\mathcal{L}_n = \|F_s - A_{st}F_t\|_{\mathrm{F}}^2, \tag{3}$$

where $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm.

To sum up, our two-level cross-domain graph alignment module is defined by incorporating Eq. (2) and (3) together as follows:

$$\mathcal{L}_g = \mathcal{L}_e + \mathcal{L}_n. \tag{4}$$

Remark: Edge-level alignment in Eq.(2) integrates the distinction between arbitrary edges from various domains and graphs' degree information into a single system. The distance of cross-domain edge reflects domain discrepancy embedded into A_{st}. Optimal A_{st} explores a probabilistic assignment to match the source nodes to the target ones. Compared to edge-level alignment, node-level alignment directly focuses on feature representation. A_{ij}^{st} indicates the probability that the source feature f_i^s and target feature f_j^t belong to the same category. According to Eq. (3), cross-domain samples with the same label tend to be clustered in the shared space with similar feature representation.

3.2.2 Source-Supervised Target Classifier

Due to the lack of label information in target domain, existing methods to solve UDA problem only employ sam-

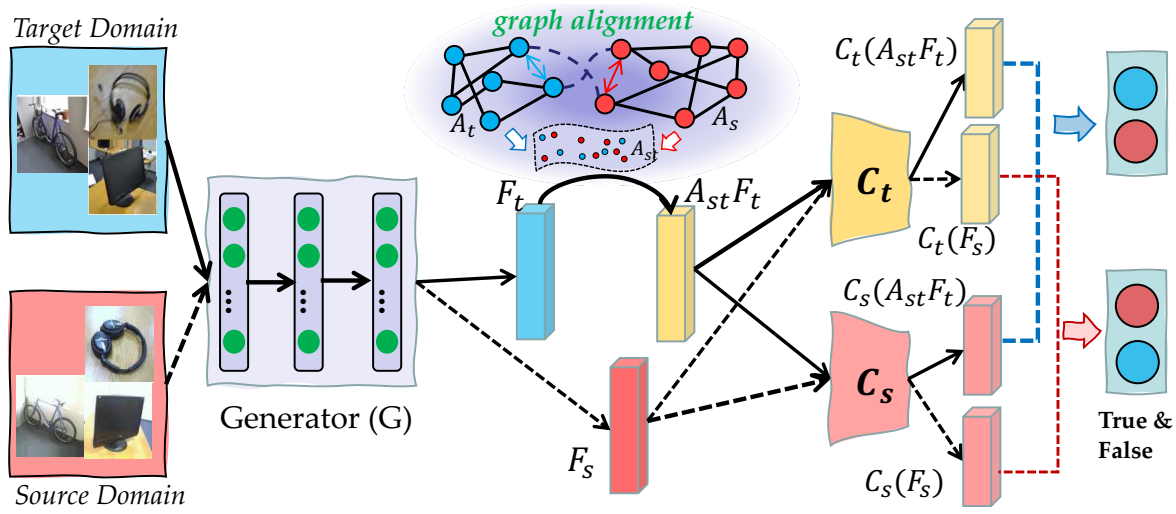


Figure 1: Overview of the proposed architecture, where features F_s and F_t are extracted from raw data through generator (VGG or ResNet), and then we capture matching relationship (blue dotted line) of two domains according to graph distribution. Moreover, two classifiers are built and fed with same input. We adopt domain adversarial training manner to maximum the difference between them.

ples from source domain to train a domain-invariant classifier shared by target domain. Other works [30, 43] alternatively design two classifiers corresponding to two domains and maximize distinction of them. To enhance the generalization ability of the classifiers to target samples, existing works normally explore pseudo labels by involving the target supervision iteratively [42, 37]. However, the fundamental challenge (e.g., to learn a robust classifier for target domain) is still unsolved as ground-truth target label is not accessible. In order to address this issue, we develop a novel source-supervised target classifier $C_t(\cdot)$ with structure preserving, as well as a traditional source-supervised classifier $C_s(\cdot)$ under a symmetric adversarial training manner.

We firstly introduce how to feed unlabeled target samples into the source-supervised target classifier and then present the whole symmetric adversarial architecture. As discussed in section 3.2.1, features F_s extracted from $\mathcal{D}_s$ can be represented by features of target domain F_t under node-level alignment, i.e., $\|F_s - A_{st}F_t\|_{\mathrm{F}}^2$. Without loss of generality, arbitrary f_i^s has the formulation $f_i^s \approx \sum_{j=1}^{n_t} a_{ij}^{st} f_j^t$. The larger a_{ij}^{st} not only demonstrates v_i^s has similar topological structure with v_j^s but also indicates f_i^s and f_j^t come from the same class. This strategy is also considered as a tool extracting samples with larger a_{ij}^{st} from target domain and ignoring influence of other samples to code f_i^s. Most likely, the selected samples share the same label with f_i^s, and are input to train the classifier, which dramatically promote the discriminative ability of classifier for samples in target domain.

Thus, C_s and C_t are developed by taking $\{F_s, Y_s\}$ and $\{A_{st}F_t, Y_s\}$ as input, respectively. Noted that $A_{st}F_t$ shares the same label information with F_s. C_t also learns to identify the interface among various classes in source domain. Interestingly, $C_t(\cdot)$ trained on $A_{st}F_t$ should also be valid to recognize F_t, since $A_{st}F_t$ and F_t share the same feature space. In this sense, we obtain the target classifier with ground-truth source supervision by transforming the target features into source ones. Note that $A_{st}F_t$ can be treated as a bridge to gap the source and target domains.

However, considering that the task of C_t is to trigger more accurate predictions on target domain, the probabilities generated from $C_t(A_{st}F_t)$ and $C_t(F_s)$ should become different. Inspired by [44], symmetric adversarial architecture is exploited to achieve this goal. From Fig. 1, there are two parallel classifiers C_s and C_t sharing the same input F_s and $A_{st}F_t$. And C_s and C_t are built in the same architecture including Fully-Connected (FC) layers and one Softmax layer. For an arbitrary feature input such as f_i^s, the output of C_s and C_t are denoted as $q_s(f_i^s) \in \mathbb{R}^C (q_s \mathbb{1}_C = 1)$ and $q_t(f_i^s) \in \mathbb{R}^C (q_t \mathbb{1}_C = 1)$, where C is the number of classes.

Given features F_s and $A_{st}F_t$, two classifiers generate four types of probabilities: $q_s(F_s)$, $q_s(A_{st}F_t)$, $q_t(F_s)$ and $q_t(A_{st}F_t)$. We train C_s and C_t to make prediction for any input by minimizing the following cross-entropy loss:

$$\begin{aligned}
\mathcal{L}_s = -\frac{1}{n_s}\Big(& \sum_{i=1}^{n_s} y_i^s \log\left(q_s(f_i^s)\right) \\
& + \sum_{i=1}^{n_s} y_i^s \log\left(q_s(\sum_{j=1}^{n_t} a_{ij}^{st} f_j^t)\right)\Big), \\
\mathcal{L}_t = -\frac{1}{n_s}\Big(& \sum_{i=1}^{n_s} y_i^s \log\left(q_t(f_i^s)\right) \\
& + \sum_{i=1}^{n_s} y_i^s \log\left(q_t(\sum_{j=1}^{n_t} a_{ij}^{st} f_j^t)\right)\Big).
\end{aligned} \tag{5}$$

Although C_s and C_t leverage same features as input, they should have various identifying functions. The primary purpose of C_s is to improve prediction accuracy of feature

F_s while C_t pays more attention to the prediction of $A_{st}F_t$. To achieve this goal, we extract feature $H_s(H_t)$ from classifier $C_s(C_t)$ before the Softmax layer and then concatenate features into $H^s_{st} = [H_s(F_s), H_t(F_s)]$ and $H^t_{st} = [H_s(A_{st}F_t), H_t(A_{st}F_t)]$. Subsequently, softmax operation is applied to obtain probability distribution $[q^*_s(F_s), q^*_t(F_s)]$ and $[q^*_s(A_{st}F_t), q^*_t(A_{st}F_t)]$. Alternatively, $q^*_s(F_s)$ should be larger than $q^*_t(F_s)$ but $q^*_s(A_{st}F_t)$ is supposed to have smaller value than $q^*_t(A_{st}F_t)$. We adopt the domain adversarial training manner in [44] by minimizing the following additional cross-entropy losses:

$$\begin{aligned} \mathcal{L}_{s_a} &= -\frac{1}{n_s}\sum\nolimits_{i=1}^{n_s}\log(\sum\nolimits_{k=1}^{C} q^*_{s_k}(f^s_j)), \\ \mathcal{L}_{t_a} &= -\frac{1}{n_s}\sum\nolimits_{i=1}^{n_s}\log(\sum\nolimits_{k=1}^{C} q^*_{t_k}(\sum\nolimits_{j}^{n_t} a^{st}_{ij} f^t_j)). \end{aligned} \tag{6}$$

To this end, we can integrate Eq. (5) and Eq. (6) into the following Eq. (7) to train classifiers by minimizing:

$$\mathcal{L}_c = \mathcal{L}_s + \mathcal{L}_t + \mathcal{L}_{s_a} + \mathcal{L}_{t_a}, \tag{7}$$

Thus, this loss function involves classification task and domain adversarial task.

3.3. Entropy Minimization

Although source-supervised target classifier leverages collaboration of target samples to improve discrimination of classifier, there is no chance for target classifier to access features of target domain directly. To avoid this issue, we adopt Entropy minimization (EM) method widely used in [35] to promote the robustness of classifier. Entropy minimization function aims to simultaneously optimize two classifiers and has the following formulation:

$$\begin{aligned} \mathcal{L}_{em} = &-\frac{1}{n_t}\sum\nolimits_{i=1}^{n_t} q_s(f^t_i)\log(q_s(f^t_i)) \\ &-\frac{1}{n_t}\sum\nolimits_{i=1}^{n_t} q_t(f^t_i)\log(q_t(f^t_i)), \end{aligned} \tag{8}$$

where $q_s(f^t_j)$ indicates the probability of target sample f^t_j and $q_t(f^t_j)$ means the output of target classifier for f^t_j. During the initial training phase, features of target domain lacking of discrimination are simply labeled with incorrect category and are difficult to be identified correctly in the later training phase. According to suggestion in [44], we only employ entropy minimization loss function to train generator instead of updating all parameters in our network.

3.4. Optimization

There are three components: generator, graph alignment and classifier in our proposed model to be optimized iteratively. We provide the following four steps to illustrate the optimization.

Step A: During the initial training phase, we use source instances with corresponding label to train C_s and C_t and update generator G. Although such a simple training manner is difficult to address domain shift problem, generator to some extent learns discriminative features for two domains. In terms of these extracted features, we can calculate cosine distance within each domain as A_s and A_t and then obtain the cross-domain similarity to initialize A_{st}.

Step B: The classifier C_t trained in the first phase produces pseudo label $\hat{Y}_t$ for target domain X_t. We then calculate a mask matrix $\mathcal{M} = Y_s\hat{Y}_t^{\mathrm{T}}$ to filter the irrelevant elements of A_{st} with the formulation as $\mathcal{M} \odot A_{st}$, where $\odot$ means element-wise product operation. Subsequently, we optimize A_{st} according to Eq. (4) and learn optimal cross-domain graph matching relation.

Step C: In this step, we train two classifiers C_s and C_t when fixing generator G. We take F_s and $A_{st}F_t$ as input both with source labels as supervised signal. In addition, classifier loss not only achieves classification task but also minimizes domain adversarial loss. Under this condition, classifiers are updated according to:

$$\min_{C_s, C_t} \mathcal{L}_s + \mathcal{L}_t + \mathcal{L}_{s_a} + \mathcal{L}_{t_a}. \tag{9}$$

Step D: Due to symmetric adversarial training, generator should confuse classifiers with $A_{st}F_t$ and F_s. Concretely, target classifier considers F_s as true while source classifier produces more value for input $A_{st}F_t$. Thus, we define a domain loss as $\mathcal{L}_d = -\frac{1}{n_s}\sum\limits_{i=1}^{n_s}\log(\sum\limits_{k=1}^{C} q^*_{s_k}(\sum\limits_{j}^{n_t} a^{st}_{ij} f^t_j)) - \frac{1}{n_s}\sum\limits_{i=1}^{n_s}\log(\sum\limits_{k=1}^{C} q^*_{t_k}(f^s_j))$. Under this circumstance, generator synthesises domain-invariant features by adversarial training. Specifically, we train generator with fixed classifiers by minimizing objective function:

$$\min_{G} \mathcal{L}_s + \mathcal{L}_t + \lambda_1(\mathcal{L}_n + \mathcal{L}_d) + \lambda_2\mathcal{L}_{em}, \tag{10}$$

where λ_1 and λ_2 control the relative importance of domain alignment and entropy minimization.

Finally, we repeat **Step B**, **Step C** and **Step D** to obtain optimal solution for our model.

4. Experiment

The proposed method is evaluated on three popular benchmark datasets of unsupervised domain adaptation and compared with other state-of-the-art algorithms.

4.1. Experimental Setting

Office-31 is considered as a standard benchmark dataset for UDA problem [29]. It contains 4,110 images collected from three various domains: Amazon Website (**A**), Web camera (**W**) and Digital SLR camera (**D**). Although images of three domains are captured under distinctive conditions, **A**, **W**

Table 1: Top-1 Accuracy (%) on Office-31 dataset for UDA (ResNet-50) and the best result is in bold type.

Method	ResNet-50	DNN	DANN [10]	JAN [22]	SimNet [27]	SymNets [44]	TADA[36]	SAFN [39]	Ours
A→W	68.4	80.5	82.0	85.4	88.6	90.8	**94.3**	90.3	92.9
D→W	96.7	97.1	96.9	97.4	98.2	**98.8**	98.7	98.7	98.7
W→D	99.3	99.6	99.1	98.4	99.7	**100**	99.8	**100**	99.8
A→D	68.9	78.6	79.7	77.8	85.3	93.9	91.6	90.7	**94.5**
D→A	62.5	63.6	68.2	69.5	73.4	74.6	72.9	73.4	**75.9**
W→A	60.7	62.8	67.4	68.9	71.6	72.5	73.0	71.2	**74.9**
Avg	76.1	80.4	82.2	82.9	86.2	88.4	88.4	87.6	**89.5**

Table 2: Top-1 Accuracy (%) on Office-Home dataset for UDA (ResNet-50) and the best result is in bold type.

Method	ResNet-50	DANN [10]	JAN [22]	DSR [2]	SymNets [44]	TADA [36]	SAFN [39]	Ours
Ar→ Cl	34.9	45.6	45.9	53.4	47.8	53.1	52.0	**56.8**
Ar→ Pr	50.0	59.3	61.2	71.6	72.9	72.3	71.7	**75.5**
Ar→ Rw	58.0	70.1	68.9	77.4	78.5	77.2	76.3	**78.9**
Cl→ Ar	37.4	47.0	50.4	57.1	**64.2**	59.1	**64.2**	61.3
Cl→ Pr	41.9	58.5	59.7	66.8	71.3	71.2	**69.9**	69.4
Cl→ Rw	46.2	60.9	61.0	69.3	74.2	72.1	71.9	**74.9**
Pr→ Ar	38.5	46.1	45.8	56.7	**64.2**	59.7	63.7	61.3
Pr→ Cl	31.2	43.7	43.4	49.2	48.8	**53.1**	51.4	52.6
Pr→ Rw	60.4	68.5	70.3	75.7	79.5	78.4	77.1	**79.9**
Rw→ Ar	53.9	63.2	63.9	68.0	**74.5**	72.4	70.9	73.3
Rw→ Cl	41.2	51.8	52.4	54.0	52.6	**60.0**	57.1	54.2
Rw→ Pr	59.9	76.8	76.8	79.5	82.7	82.9	81.5	**83.2**
Avg	46.1	57.6	58.3	64.9	67.6	67.6	67.3	**68.4**

and **D** share the same label space with 31 categories. In addition, the biggest challenge of domain adaptation in this dataset is imbalanced across three domains. Specifically, Amazon domain consists of 2,817 images, while DSLR domain and Webcam domain only contain 498 and 795 images, respectively. We evaluate six domain adaptation tasks in Office-31.

Office-Home is another more challenging dataset for visual domain adaptation [34]. It includes 15,500 images belonging to 65 categories. These images containing various daily objects are captured in office or home scenes. There are four different domains: Artistic images (Ar), Clip Art (Cl), Product images (Pr) and Real-World images (Rw), which forms 12 adaptation tasks.

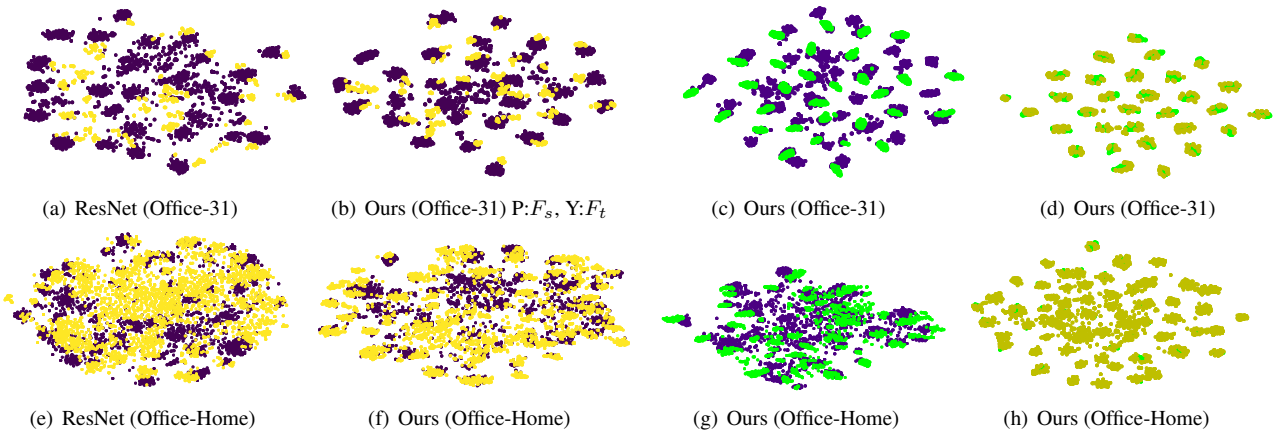
ImageCLEF-DA dataset is another popular standard benchmark for unsupervised domain adaptation including three domains: Caltech-256 (**C**), ImageNet ILSVRC 2012 (**I**) and Pascal VOC 2012 (**P**). Arbitrary domain includes 12 categories and each class contains 50 images. Different from Office-Home and Office-31, three domains in this dataset have the same scale. There are six unsupervised domain adaptation tasks to be evaluated.

Comparisons. We compare our structure preserving method with generative adversarial algorithms: DANN [10], SymNets [44] and maximum mean discrepancy based on approaches: JAN [22] and other deep models like DSR [2], TADA [36], and SAFN [39]. JAN is implemented with the released code. Moreover, we cite the results of DANN, SymNets, DSR, TADA and SAFN directly from corresponding papers [10, 44, 2, 36] for a fair comparison as we adopt the exact the same experimental protocol.

Implementation details. We implement the proposed method on Tensorflow. The ResNet-50 (without the last FC layer) pre-trained on ImageNet dataset [6] is employed to extract features from raw images. We only fine-tune parameters of ResNet-50 on source domain. The architecture in classifier C_s and C_t both include two-layer FC layers with activation function as $Relu$. We adopt Adam optimizer to update all parameters and select the learning rate $\eta_p = \frac{\eta_0}{(1+ap)^b}$, where p is linearly changing from 0 to 1. We set the initial learning rate $\eta_0 = 0.01$, $\alpha = 10$ and $\beta = 0.75$ according to strategy in [44]. λ_1 and λ_2 are selected from $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$. Finally, we obtain the classification accuracy in target domain using C_t.

4.2. Comparison Results

Table 1 shows classification accuracy result of domain adaptation task on Office-31 dataset. The proposed approach overpasses all compared methods in terms of average accuracy. Due to imbalanced condition across three domains, it is difficult for model to transfer knowledge learned in a small-scale dataset into another larger domain. How-



(a) ResNet (Office-31) (b) Ours (Office-31) P:F_s, Y:F_t (c) Ours (Office-31) (d) Ours (Office-31)

(e) ResNet (Office-Home) (f) Ours (Office-Home) (g) Ours (Office-Home) (h) Ours (Office-Home)

Figure 2: Comparison of t-SNE visualization of ResNet-50 and our learned feature representations. (**a**): t-SNE of ResNet (Office-31) with F_s and F_t. (**b**): t-SNE of Ours (Office-31) with F_s and F_t. (**c**): t-SNE of Ours (Office-31) with F_s and $A_{st}F_t$. (**d**): t-SNE of Ours (Office-31) with $A_{st}F_t$ and F_t. (**e**): t-SNE of ResNet (Office-Home) with F_s and F_t. (**f**): t-SNE of Ours (Office-31) with F_s and F_t. (**g**): t-SNE of Ours (Office-Home) with F_s and $A_{st}F_t$. (**h**): t-SNE of Ours (Office-Home) with $A_{st}F_t$ and F_t. We compute t-SNE with the output of the last FC layer on Office-31 task A $\rightarrow$ W and Office-Home task Ar$\rightarrow$Cl. Purple indicates F_s, Yellow denotes F_t and Green represents $A_{st}F_t$.

ever, different from the results of other algorithms in tasks **D** $\rightarrow$ **A** and **W** $\rightarrow$ **A**, our model shows less sensitive to imbalanced circumstance. The main reason for success of our model is that we introduce cross-domain graph information into our method. Alignment with graph discovers similarity of topological structure and utilizes consistency to address domain shift. On the other hand, target classifier with cross-domain graph provides feature learning of target domain with more label information from source domain.

The classification results about 12 domain adaptation tasks on the Office-Home [34] is reported in Table 2. As we all know, since office-Home dataset has more categories than office-31 dataset, it is difficult for the same method to produce better result than its performance in office-31 dataset. Compared to ResNet-50 only fine-tuned in source domain, impressive improvements have been obtained with the mentioned methods. The performance of our method significantly achieves improvements when compared with other algorithms. Although the results of SymNets on tasks **Cl** $\rightarrow$ **Ar**, **Cl** $\rightarrow$ **Pr** and **Rw** $\rightarrow$ **Cl** are higher, our method substantially promotes classification accuracy in most cases and obtains better average performance. Specifically, our model produces higher accuracy with large margin for several difficult tasks such as **Ar** $\rightarrow$ **Cl** and **Ar** $\rightarrow$ **Pr** task. It indicates that the proposed method effectively eliminates domain discrepancy and extracts domain-invariant feature by graph alignment and domain adversarial alignment.

Table 3 reports classification accuracy on ImageCLEF-DA dataset. Different from previous two datasets, each domain in this dataset has the same number of samples. All methods even ResNet-50 totally obtain impressive accuracy. According to comparison with mentioned methods, our model achieves the best performance in most cases e.g., $P \rightarrow C$, $C \rightarrow I$ and $I \rightarrow C$, demonstrating the effectiveness of our proposed method in solving domain adaptation problem. In addition, compared to traditional adversarial training methods (DANN and CDAN), our model and SymNets both perform better results than them, benefiting from symmetric adversarial training manner. Two classifiers in symmetric adversarial method tend to describe the same feature from various perspectives. Thus, the discriminative ability of target classifier is improved dramatically.

Table 3: Top-1 Accuracy (%) on ImageCLEF-DA dataset for UDA (ResNet-50) and the best result is in bold type.

Method	I$\rightarrow$ P	P$\rightarrow$ I	I$\rightarrow$ C	C$\rightarrow$ I	C$\rightarrow$ P	P$\rightarrow$ C
ResNet-50	74.8	83.9	91.5	78	65.5	91.2
DAN	74.5	82.2	92.8	86.3	69.2	89.8
DANN [10]	75	86	96.2	87	74.3	91.5
JAN [22]	76.8	88	94.7	89.5	74.2	91.7
CDAN [21]	76.7	90.6	97	90.5	74.5	93.5
SymNets [44]	**80.2**	93.6	97	93.4	**78.7**	96.4
SAFN [39]	79.3	**93.8**	96.3	91.7	77.6	95.3
Ours	79.4	91.9	**97.9**	**94.1**	76.5	**97.2**

4.3. Ablation Study

4.3.1 t-SNE visualization

To understand the effect of graph alignment, we utilize t-SNE visual technique to observe distribution of features in 2D-space. We compute t-SNE with output of the last FC layer in generator and conduct experiments on Office-31 (A$\rightarrow$W) and Office-Home (Ar$\rightarrow$Cl) for the original ResNet-50 features and our model. According to Fig. 2

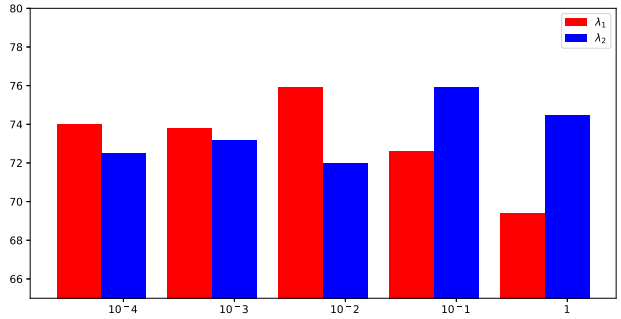
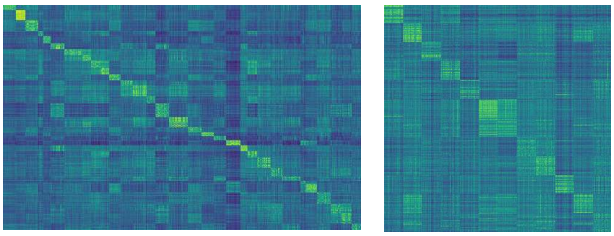


Figure 3: Parameter analysis of our proposed model GSP. We conduct experiment on Office-31 for task $D \rightarrow A$ and investigate classification accuracy with varying parameters λ_1 and λ_2. (Red: λ_1, Blue: λ_2)

(a), there are a few overlaps between target instances (yellow) and samples of source domain (purple), demonstrating cross-domain distribution exists large difference named domain shift. Through feature learning phase with GSP, target samples are embedded into source domain in Fig. 2 (b). When comparing the location of target samples in Fig. 2 (a) and Fig. 2 (b), We also know that there is a phenomenon of translation resulting from the influence of graph alignment which matches target samples with source data points. The comparison between F_s and $A_{st}F_t$ is shown in Fig. 2 (c). Different from F_t, almost all $A_{st}F_t$ are attached to features of source domain. It illustrates that GSP learns cross-domain matching relation and exploits it to transform target domain into source domain. Since source domain (A) contains more samples than target domain (W), space expanded by $A_{st}F_t$ becomes larger than that of F_t in Fig. 2 (d). Thus, reducing domain discrepancy tends to be obstructed with difference between $A_{st}F_t$ and F_t. In addition, focusing on the center of Fig. 2 (e), this area are occupied by abundant target samples with a few source instances. GSP employs graph information to discover cross-domain similarity and transfers data points of target domain into the corresponding instances of source domain in Fig. 2 (f), meaning our model effectively achieves domain adaptation. Similar with Fig. 2 (e), $A_{st}F_t$ mostly are embedded into source domain. The last Fig. 2 (h) shows relationship between $A_{st}F_t$ and F_t on office-home dataset. Abundant overlaps between them means they share the same space. Thus, we transform target domain into source domain through $A_{st}F_t$.

4.3.2 Parameter analysis

In this section, we conduct experiments to observe the performance of our model with parameters λ_1 and λ_2. The control variations method is adopted to investigate experimental results. We select value from $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$. Concretely, when fixing parameter λ_1, we change parameter λ_2 from 10^{-4} to 1. The parameter analysis is conducted on Office-31 ($D \rightarrow A$) and



(a) (b)

Figure 4: Visualization of cross-domain graph A_{st} on (a) Office-31 (D$\rightarrow$ W) with 31 categories and (b) ImageCLEF-DA (P$\rightarrow$ C) with 12 categories.

Fig. 3 reports results. According to Fig. 3, as λ_1 goes up, classification accuracy tend to be improved and then be reduced gradually, illustrating our model is sensitive to parameter λ_1 which adjusts importance of domain adversarial term. However, our method becomes stable when raising the value of λ_2. GSP achieves optimal result with $\lambda_1 = 0.01$ and $\lambda_2 = 0.1$.

4.3.3 Cross-domain Graph Analysis

In addition to t-SNE analysis, we also visualize graph matching A_{st} to observe the performance of edge-level alignment which attempts to discover cross-domain matching relation. Ideally, A^{st}_{ij} has large value when f^s_i and f^t_j belong to the same category, otherwise, A^{st}_{ij} tends to be small. We conduct experiments on Office-31 (**W** $\rightarrow$ **D**) and ImageCLEF-DA (**P** $\rightarrow$ **C**) and extract the optimal A_{st} shown in Fig. 4. The visualization of graph exhibits diagonal block structure which means GSP explores edge-level alignment to capture cross-domain matching information.

5. Conclusion

In this paper, we rethink Unsupervised Domain Adaptation (UDA) from the perspective of graph distribution and propose Generative Cross-domain learning via Structure Preserving (GSP) to address domain shift problem. GSP model mainly contains two important components: graph alignment and source-supervised target classifier. Graph alignment utilizes edge-level alignment to capture cross-domain matching relation and incorporates relation into node-level alignment to eliminate domain shift. Moreover, we introduce matching information into classifiers and develop source-supervised target classifier exploiting label of source domain to supervise feature learning of target domain. To maximize difference of two classifiers, we adopt symmetric adversarial training manner to train neural network. Extensive experimental results and analyses on several cross-domain visual benchmarks have illustrated the effectiveness of GSP model by comparing with other competitive methods.

References

[1] Alexander M Bronstein, Michael M Bronstein, Ron Kimmel, Mona Mahmoudi, and Guillermo Sapiro. A gromov-hausdorff framework with diffusion geometry for topologically-robust non-rigid shape matching. *International Journal of Computer Vision*, 89(2-3):266–286, 2010. 2

[2] Ruichu Cai, Zijian Li, Pengfei Wei, Jie Qiao, Kun Zhang, and Zhifeng Hao. Learning disentangled semantic representation for domain adaptation. In *IJCAI: proceedings of the conference*, volume 2019, page 2060. NIH Public Access, 2019. 6

[3] 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*, pages 1081–1090, 2019. 2

[4] Ziliang Chen, Jingyu Zhuang, Xiaodan Liang, and Liang Lin. Blending-target domain adaptation by adversarial meta-adaptation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2248–2257, 2019. 1

[5] Safa Cicek and Stefano Soatto. Unsupervised domain adaptation via regularized conditional alignment. *arXiv preprint arXiv:1905.10885*, 2019. 3

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 6

[7] Zhengming Ding, Sheng Li, Ming Shao, and Yun Fu. Graph adaptive knowledge transfer for unsupervised domain adaptation. In *Proceedings of the European Conference on Computer Vision*, pages 37–52, 2018. 1

[8] Zhengming Ding, Ming Shao, and Yun Fu. Robust multi-view representation: a unified perspective from multi-view learning to domain adaption. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 5434–5440, 2018. 1

[9] Jiahua Dong, Yang Cong, Gan Sun, Bineng Zhong, and Xiaowei Xu. What can be transferred: Unsupervised domain adaptation for endoscopic lesions segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1

[10] 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. 6, 7

[11] Arthur Gretton, Alex Smola, Jiayuan Huang, Marcel Schmittfull, Karsten Borgwardt, and Bernhard Schölkopf. Covariate shift by kernel mean matching. *Dataset shift in machine learning*, 3(4):5, 2009. 1

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[13] Shuhui Jiang, Zhengming Ding, and Yun Fu. Heterogeneous recommendation via deep low-rank sparse collective factorization. *IEEE transactions on pattern analysis and machine intelligence*, 2019. 1

[14] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4893–4902, 2019. 1, 2

[15] Abhishek Kumar, Prasanna Sattigeri, Kahini Wadhawan, Leonid Karlinsky, Rogerio Feris, Bill Freeman, and Gregory Wornell. Co-regularized alignment for unsupervised domain adaptation. In *Advances in Neural Information Processing Systems*, pages 9345–9356, 2018. 1

[16] Chen-Yu Lee, Tanmay Batra, Mohammad Haris Baig, and Daniel Ulbricht. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10285–10295, 2019. 2

[17] Seungmin Lee, Dongwan Kim, Namil Kim, and Seong-Gyun Jeong. Drop to adapt: Learning discriminative features for unsupervised domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 91–100, 2019. 3

[18] Shuang Li, Chi Harold Liu, Qiuxia Lin, Qi Wen, Limin Su, Gao Huang, and Zhengming Ding. Deep residual correction network for partial domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 1

[19] Hong Liu, Mingsheng Long, Jianmin Wang, and Michael Jordan. Transferable adversarial training: A general approach to adapting deep classifiers. In *International Conference on Machine Learning*, pages 4013–4022, 2019. 1, 2

[20] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I Jordan. Learning transferable features with deep adaptation networks. *arXiv preprint arXiv:1502.02791*, 2015. 1, 2

[21] 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. 1, 2, 7

[22] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2208–2217. JMLR. org, 2017. 1, 2, 6, 7

[23] Yawei Luo, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2507–2516, 2019. 1

[24] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 2

[25] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1406–1415, 2019. 1

[26] Gabriel Peyré, Marco Cuturi, and Justin Solomon. Gromov-wasserstein averaging of kernel and distance matrices. In *International Conference on Machine Learning*, pages 2664–2672, 2016. 2, 3

[27] Pedro O Pinheiro. Unsupervised domain adaptation with similarity learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8004–8013, 2018. 6

[28] Subhankar Roy, Aliaksandr Siarohin, Enver Sangineto, Samuel Rota Bulo, Nicu Sebe, and Elisa Ricci. Unsupervised domain adaptation using feature-whitening and consensus loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9471–9480, 2019. 1

[29] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010. 5

[30] 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. 1, 4

[31] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015. 3

[32] 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. 2

[33] Titouan Vayer, Laetita Chapel, Rémi Flamary, Romain Tavenard, and Nicolas Courty. Fused gromov-wasserstein distance for structured objects: theoretical foundations and mathematical properties. *arXiv preprint arXiv:1811.02834*, 2018. 3

[34] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5018–5027, 2017. 6, 7

[35] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2517–2526, 2019. 5

[36] Ximei Wang, Liang Li, Weirui Ye, Mingsheng Long, and Jianmin Wang. Transferable attention for domain adaptation. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2019. 6

[37] Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 5419–5428, 2018. 1, 4

[38] Hongteng Xu, Dixin Luo, Hongyuan Zha, and Lawrence Carin. Gromov-wasserstein learning for graph matching and node embedding. *arXiv preprint arXiv:1901.06003*, 2019. 2, 3

[39] Ruijia Xu, Guanbin Li, Jihan Yang, and Liang Lin. Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1426–1435, 2019. 6, 7

[40] Hongliang Yan, Yukang Ding, Peihua Li, Qilong Wang, Yong Xu, and Wangmeng Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2272–2281, 2017. 1

[41] Guanglei Yang, Haifeng Xia, Mingli Ding, and Zhengming Ding. Bi-directional generation for unsupervised domain adaptation. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020. 1

[42] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu. Collaborative and adversarial network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3801–3809, 2018. 1, 4

[43] Yang Zhang, Philip David, and Boqing Gong. Curriculum domain adaptation for semantic segmentation of urban scenes. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2020–2030, 2017. 4

[44] Yabin Zhang, Hui Tang, Kui Jia, and Mingkui Tan. Domain-symmetric networks for adversarial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5031–5040, 2019. 1, 2, 4, 5, 6, 7