GCAN: Graph Convolutional Adversarial Network for Unsupervised Domain Adaptation

Xinhong Ma[1,2,3], Tianzhu Zhang[1,2,4] and Changsheng Xu[1,2,3]
[1] National Lab of Pattern Recognition (NLPR),
Institute of Automation, Chinese Academy of Sciences (CASIA)
[2] University of Chinese Academy of Sciences (UCAS)
[3] Peng Cheng Laboratory, ShenZhen, China
[4] University of Science and Technology of China
{xinhong.ma, csxu}@nlpr.ia.ac.cn, tzzhang10@gmail.com

Abstract

To bridge source and target domains for domain adaptation, there are three important types of information including data structure, domain label, and class label. Most existing domain adaptation approaches exploit only one or two types of the above information and cannot make them complement and enhance each other. Different from existing methods, we propose an end-to-end Graph Convolutional Adversarial Network (GCAN) for unsupervised domain adaptation by jointly modeling data structure, domain label, and class label in a unified deep model. The proposed GCAN model enjoys several merits. First, to the best of our knowledge, this is the first work to model the three kinds of information jointly in a deep model for unsupervised domain adaptation. Second, the proposed model has designed three effective alignment mechanisms including structure-aware alignment, domain alignment, and class centroid alignment, which can learn domain-invariant and semantic representations effectively to reduce the domain discrepancy for domain adaptation. Extensive experimental results on five standard benchmarks demonstrate that the proposed GCAN algorithm performs favorably against state-of-the-art unsupervised domain adaptation methods.

1. Introduction

Deep learning approaches can learn discriminative representations and have significantly improved the state of the arts for a wide variety of machine-learning tasks and computer vision applications [4, 10, 19, 33, 34, 43, 48, 83, 85, 87, 88, 3]. Unfortunately, the impressive performance gains come only when massive amounts of labeled data are available for deep model training. In practice, manual labeling of such sufficient training data is often prohibitive or impossible to collect, especially for a target task short of labeled data. Therefore, there is a strong motivation to build effective learners that can leverage rich labeled data from a different source domain [61]. However, due to the phenomenon known as dataset bias or domain shift, predictive models trained on one large scale dataset do not generalize well to new datasets and tasks [61]. As a result, this learning paradigm suffers from the shift in data distributions across different domains, which poses a huge obstacle for adapting predictive models to the target task [69, 61].

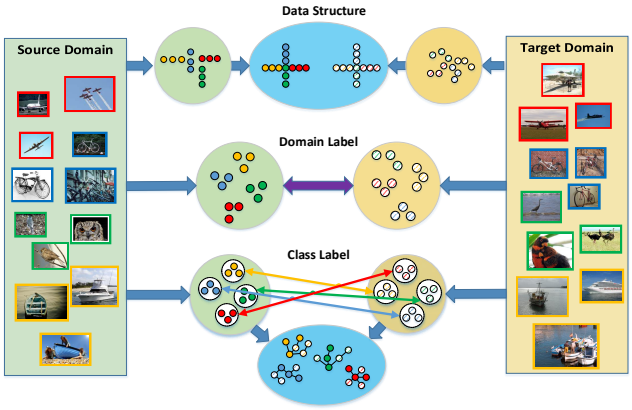


Figure 1. Illustration of our motivation. The data structure, domain label and class label are three important types of information to bridge source and target domains for domain adaptation.

Learning a discriminative classifier or other predictors in the presence of the shift between training and test distributions is known as transfer learning or domain adaptation [61]. The main technical difficulty of domain adaptation is how to formally reduce the distribution discrepancy across different domains, usually labeled source data and unlabeled target data. To address this issue, a variety of domain adaptation approaches have been proposed [15, 61, 76]. Generally, these methods can be categorized into three major categories including instance-

based domain adaptation [14, 37, 11, 13, 39, 7, 8, 12], parameter-based domain adaptation [20, 62], feature-based domain adaptation [53, 14, 37, 40, 61, 32]. Among existing methods, Maximum Mean Discrepancy (MMD) [32] is one of the most widely used strategies to measure the distribution difference between source and target domains [50, 38, 57, 45]. Later on, numerous domain adaptation approaches have been proposed by designing a revised class-wise MMD, such as, class-wise MMD [38, 79], multi-kernel MMD [74, 49]. Recently, numerous adversarial adaptation methods [1, 23, 47, 82, 41, 73] have been proposed, which is analogous to generative adversarial networks [30]. A domain classifier is trained to tell whether the sample comes from source domain or target domain. The feature extractor is trained to minimize the classification loss and maximize the domain confusion loss. Domain-invariant yet discriminative features are seemingly obtainable through the principled lens of adversarial training.

In the above mentioned methods, three types of information play crucial roles in bridging the labeled source and unlabeled target domains, namely data structure, domain label, and class label, as shown in Figure 1. **Data structure** generally reflects the inherent properties of dataset including marginal or conditional data distributions [51], data statistics information [86], geometric data structure [66, 78] and so on. **Domain label** is used in adversarial domain adaptation methods [1, 23, 73, 7, 8, 12], and can help train a domain classifier to model the global distribution of source and target domains. **Class labels**, specifically target pseudo labels, are usually adopted to enforce the semantic alignment [18, 77], which can guarantee that samples from different domains with the same class label will be mapped nearby in the feature space. In summary, the three types of information help reduce the domain discrepancy in different aspects, and they can complement and enhance each other for domain adaptation. It naturally comes into minds that *how to effectively leverage data structure, domain label, and class label jointly in a unified network for unsupervised domain adaptation*. As we know, most previous methods only exploit one or two kinds of information into consideration. In [77], the deep adversarial adaptation method only enforces the alignment of global domain statistics, and the crucial semantic class label information for each category may be lost. Even with perfect confusion alignment, there is no guarantee that samples from different domains with the same class label will be mapped nearby in the feature space. Therefore, some semantic transfer methods [77, 55, 58, 72] have been proposed and can propagate the class label information into the deep adversarial adaptation network to address the above limitations. For traditional data structure related methods [54, 86, 81, 78], they can reduce the distributional divergence between domains while preserving data properties in original spaces. However, it is difficult to model and integrate the data structure information into existing deep networks effectively.

To deal with the above limitations, we propose an end-to-end Graph Convolutional Adversarial Network (GCAN) for unsupervised domain adaptation by jointly modeling data structure, domain label, and class label in a unified deep model. To align domain distributions robustly, we design three effective alignment mechanisms including structure-aware alignment, domain alignment, and class centroid alignment, which play important roles in reducing the domain discrepancy for domain adaptation. In the structure-aware alignment, data structures of source and target domains are exploited so that the structure discrepancy can be minimized to reduce domain shift. To model data structures under the deep network, we use the CNN features of samples to construct a dense-connected instance graph based on the similarity of structural characteristics of samples. Each node corresponds to CNN features of a sample, which is extracted by a standard convolutional network, *e.g*., the AlexNet. Then, the Graph Convolution Network (GCN) is applied on the instance graph, which allows the structure information to be propagated along the weighted graph edge which can be learned from a designed network. In the domain alignment, global domain statistics from different domains are excavated to match each other. The divergence of domain statistics measured by the adversarial similarity loss is used to guide the feature extractor to learn domain-invariant representations. In the class centroid alignment, we constrain class centroids from different domains to move closer with iteration increasing so that the learned representations can be encoded with the class label information. Thereby, samples with the same category label can be embedded nearby in the feature space. Our model conducts a class alignment loss to achieve the idea and a moving centroid strategy is applied to suppress the influence of false pseudo-labels. By modeling the three alignment mechanisms, the deep network can generate domain-invariant and discriminative semantic representations.

The major contributions of this work can be summarized as follows. (1) We propose an end-to-end Graph Convolutional Adversarial Network for unsupervised domain adaptation by modeling data structure, domain label, and class label jointly in a unified network. To the best of our knowledge, this is the first work to model the three kinds of information jointly in a deep model for unsupervised domain adaptation. (2) The proposed alignment mechanisms (structure-aware alignment, domain alignment, and class centroid alignment) can learn domain-invariant and semantic representations effectively to reduce the domain discrepancy for domain adaptation. (3) Extensive experimental results on five standard benchmarks demonstrate that the proposed GCAN algorithm performs favorably against state-of-the-art unsupervised domain adaptation methods.

2. Related Work

In this section, we briefly overview methods that are related to domain adaptation and graph neural networks.

Domain Adaptation. A large number of domain adaptation methods have been proposed over the recent years. Generally, they can be mainly categorized into three groups: (1) Instance-based domain adaptation aims to identify the training samples that are most relevant to the target domain by instance reweighting and importance sampling. The reweighted source instances are then used for training a target domain model. Here, training on the reweighted source samples guarantees classifiers with transferability [11, 13, 39, 7, 8, 12]. (2) Parameter-based domain adaptation assumes that models of the source and target domain share the same prior parameters. It is designed to transfer knowledge by shared or regularized parameters of source and target domain models, or through combining multiple reweighted source models to form an improved target model [20, 62]. (3) Feature-based domain adaptation is designed to map different domains into a common shared space and make their feature distributions as close as possible [14, 37, 40, 61, 32]. In addition, [76] categorizes feature-based domain adaptation methods into two groups: asymmetric feature-based methods and symmetric feature-based methods. Asymmetric feature-based methods transform the features of one domain to more closely match another domain [40, 37] while symmetric feature-based methods [54, 86, 18, 54] map different domains to a common latent space where the feature distributions are close.

Recently, deep learning has been regarded as a powerful way to learn feature representations for domain adaptation. Among existing methods, Maximum Mean Discrepancy (MMD) [32] is one of the most widely used strategies to measure the distribution difference between source and target domains [50, 38, 57, 45]. The MMD is a nonparametric metric that measures the distribution divergence between the mean embedding of two distributions in Reproducing Kernel Hilbert Space (RKHS). The deep domain confusion (DDC) method [74] utilizes the MMD metric in the last fully connected layer in addition to the regular classification loss to learn representations that are both domain invariant and discriminative. In [49], the deep adaptation network (DAN) is proposed to enhance the feature transferability by minimizing a multi-kernel MMD in several task-specific layers. Currently, most methods use an adversarial objective to reduce domain discrepancy [5, 22, 47, 49, 53, 59, 73]. In [1, 23], the domain adversarial neural network (DANN) is proposed to learn domain invariant features by a minimax game between the domain classifier and the feature extractor. In order to back-propagate the gradients computed from the domain classifier, the DANN employs a gradient reversal layer. On the other hand, a general framework [73] is proposed for adversarial adaptation by choosing adversarial loss type with respect to the domain classifier and the weight sharing strategy. Our proposed GCAN model can also be viewed as an adversarial adaptation method. The difference is that our model can leverage data structure, domain label and class label jointly in a unified network for unsupervised domain adaptation.

Graph Neural Networks (GNN). The GNN is designed to use deep learning architectures on graph-structured data, which is in fact natural generalizations of convolutional networks to non-Euclidean graphs. The GNN is first proposed in [31, 64] as a trainable recurrent message passing whose fixed points could be adjusted discriminatively. In [46, 70], the GNN model is relaxed by untying the recurrent layer weights and using several nonlinear updates through gating mechanisms. In [6, 35], the models are designed to learn smooth spectral multipliers of the graph Laplacian, albeit with high computational cost. In [16, 42], the computational bottleneck is resolved by learning polynomials of the graph Laplacian, thus avoiding the computation of eigenvectors and completing the connection with GNNs.

Among above graph neural networks, the Graph Convolutional Network (GCN) has been applied to many applications [80, 65, 44, 84, 24, 25, 26]. The principle of constructing GCNs on graph generally follows two streams: (1) the spectral perspective, where the locality of the graph convolution is considered in the form of spectral analysis [21, 35, 42]. (2) the spatial perspective, where the convolution filters are applied directly on the graph nodes and their neighbors [60]. Our works is based on the spectral perspective line [42]. The proposed model exploits the GCN to operate on a dense-connected instance graph so that data structure information can be jointly complemented with domain label and class label information in a unified deep network.

3. Our Approach

In this section, we provide details of the proposed model for unsupervised domain adaptation.

3.1. Notations and Definitions

We give some notations and definitions following [15, 61, 76]. In unsupervised domain adaptation, we are given n_s labeled samples $\left\{\left(x_S^{(i)}, y_S^{(i)}\right)\right\}_{i=1}^{n_s}$ from the source domain D_S, where $x_S^{(i)} \in \mathcal{X}_S$ and $y_S^{(i)} \in \mathcal{Y}_S$. $\mathcal{X}_S$ and $\mathcal{Y}_S$ are defined as the source data space and source label space, respectively. Additionally, we are also given n_t unlabeled target samples $\left\{\left(x_T^{(i)}\right)\right\}_{i=1}^{n_t}$, where $x_T^{(i)} \in \mathcal{X}_T$, and the $\mathcal{X}_T$ represents target data space. The $\mathcal{X}_S$ and $\mathcal{X}_T$ are assumed to be different but related (referred as covariate shift in the literature [67]). The target task is assumed to be the same with the source task, which means the source label space $\mathcal{Y}_S$ is shared with the target label space $\mathcal{Y}_T$. Our ultimate goal is to develop a deep neural network f: $\mathcal{X}_T \rightarrow \mathcal{Y}_T$ that

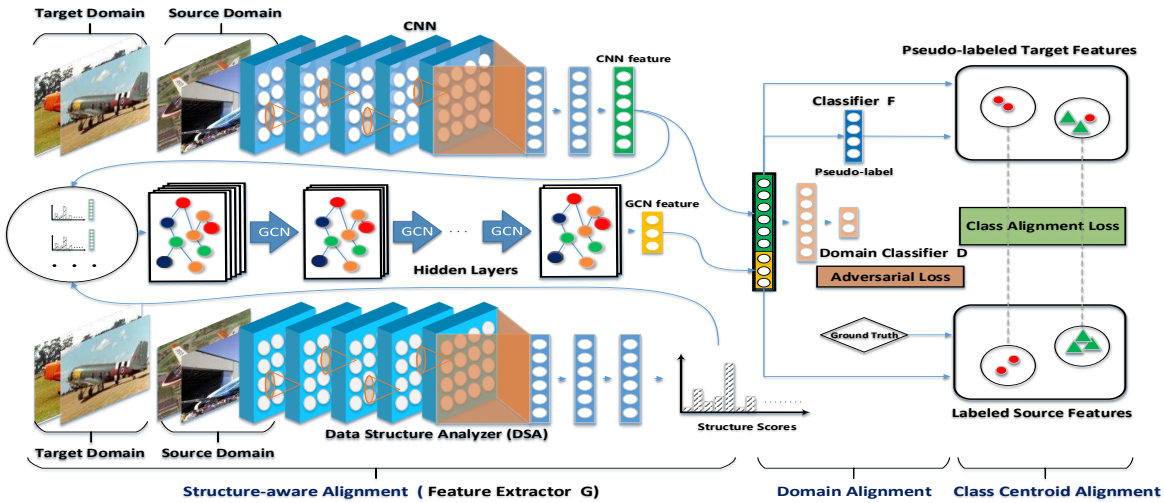


Figure 2. The architecture of the proposed Graph Convolutional Adversarial Network (GCAN). Our GCAN consists of three alignment mechanisms including structure-aware alignment, domain alignment, and class centroid alignment. In the structure-aware alignment, the Data Structure Analyzer network generates structure scores encoded with source data structure information while the CNN features are extracted by CNNs. Then, the structure scores and CNN features are used to construct dense-connected instance graphs for the GCN. The concatenated CNN and GCN features are input to the domain alignment and class centroid alignment modules. In the domain alignment, a domain alignment loss is applied to match the global domain statistics. In the class centroid alignment, pseudo-labeled target features and labeled source features are used to calculate the class centroid alignment loss to ensure that samples with same class from different domains can be embedded closely. For more details, please refer to the text.

is able to predict labels for samples from target domain.

3.2. Graph Convolutional Adversarial Network

The architecture of our proposed Graph Convolutional Adversarial Network is shown in Figure 2. For unsupervised domain adaptation, in the presence of domain shift, a label prediction function f is trained by minimizing the overall objective as shown in Eq. (1):

$$\begin{aligned}\mathcal{L}\left(\mathcal{X}_S, \mathcal{Y}_S, \mathcal{X}_T\right) = \mathcal{L}_C\left(\mathcal{X}_S, \mathcal{Y}_S\right) + \lambda \mathcal{L}_{DA}\left(\mathcal{X}_S, \mathcal{X}_T\right) \\ + \gamma \mathcal{L}_{CA}\left(\mathcal{X}_S, \mathcal{Y}_S, \mathcal{X}_T\right) + \eta \mathcal{L}_T,\end{aligned} \tag{1}$$

The classification loss $\mathcal{L}_C\left(\mathcal{X}_S, \mathcal{Y}_S\right)$ is shown in Eq.(2).

$$\mathcal{L}_C\left(\mathcal{X}_S, \mathcal{Y}_S\right) = \mathbb{E}_{(x,y)\sim D_S}\left[J\left(f\left(x\right), y\right)\right] \tag{2}$$

The $J\left(\cdot,\cdot\right)$ is typically a cross entropy loss. The λ, γ, and η are the balance parameters. The $\mathcal{L}_{DA}$, $\mathcal{L}_{CA}$ and $\mathcal{L}_T$ represent the domain alignment loss, the class alignment loss, and the triplet loss for the structure-aware alignment, respectively. The details are introduced as follows.

3.2.1 Domain Alignment

Here, we use the domain adversarial similarity loss as the domain alignment loss as shown in Eq. (3). Specifically, we employ an additional domain classifier $\mathbf{D}$ to tell whether the features from the feature extractor $\mathbf{G}$ arise from source or target domain while the $\mathbf{G}$ is trained to fool the $\mathbf{D}$. This two-player minimax game is expected to reach an equilibrium where the features from $\mathbf{G}$ are domain-invariant.

$$\begin{aligned}\mathcal{L}_{DA}\left(\mathcal{X}_S, \mathcal{X}_T\right) = \mathbb{E}_{x\in D_S}\left[\log\left(1 - D\left(G\left(x\right)\right)\right)\right] \\ + \mathbb{E}_{x\in D_T}\left[\log\left(D\left(G\left(x\right)\right)\right)\right]\end{aligned} \tag{3}$$

3.2.2 Structure-aware Alignment

The domain alignment mechanism only enforces the alignment of global domain statistics but ignores the structure information of a mini-batch samples. In fact, previous methods [54, 86] focus on modeling the data structure information for unsupervised domain adaptation and have gained impressive performance, which further emphasizes the importance of data structure information. In order to model the data structure from a mini-batch source and target samples in a deep network, we propose a structure-aware alignment mechanism for unsupervised domain adaptation.

For the structure-aware alignment, we first use a Data Structure Analyzer (DSA) network to generate **Structure Scores** for a mini-batch samples. Then, the obtained structure scores and the learnt CNN features of samples are employed to construct densely-connected instance graphs. The Graph Convolutional Network (GCN) is then operated on the instance graphs to learn GCN Features encoded with the data structure information. Before introducing our method, we first give a brief introduction of the GCN proposed in [42]. The GCN aims to learn the layerwise propagation operations that can be applied directly on graphs. Given an undirected graph with m nodes, a set of edges between nodes, and an adjacency matrix $\mathbf{A} \in R^{m\times m}$, we formulate a linear transformation of graph convolution as the multiplication of a graph signal $\mathbf{X} \in R^{k\times m}$ (the column vector $\mathbf{X}_i \in R^k$ is the feature representation of the i-th node) with a filter $\mathbf{W} \in R^{k\times c}$:

$$\mathbf{Z} = \hat{\mathbf{D}}^{-\frac{1}{2}}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-\frac{1}{2}}\mathbf{X}^T\mathbf{W}, \tag{4}$$

where $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, $\mathbf{I}$ is the identity matrix, and $\hat{\mathbf{D}}_{ii} = \sum_j \hat{\mathbf{A}}_{ij}$. In this formulation, the output is a $c \times m$ matrix $\mathbf{Z}$. Note that the GCN can be constructed by stacking multiple graph convolutional layers as the form in Eq. (4), and each layer is followed by a non-linear operation (such as ReLU).

Next, we show how to build densely-connected instance graphs for the GCN, i.e. the graph signal $\mathbf{X}$ and the adjacency matrix $\mathbf{A}$ in Eq. (4). Each node in the instance graph represents the feature of a sample, which is extracted by a standard convolutional network, *e.g.*, the AlexNet [43] or VGG [68]. Thus, the graph signal $\mathbf{X}$ can be obtained by:

$$\mathbf{X} = CNN(\mathbf{X}_{batch}), \tag{5}$$

where $\mathbf{X}_{batch}$ represents a mini-batch samples. To construct the adjacency matrix $\hat{\mathbf{A}}$, the same mini-batch samples are fed into a **Data Structure Analyzer (DSA)** network to generate structure scores $\mathbf{X}_{sc}$. With these structure scores, the adjacency matrix $\hat{\mathbf{A}}$ can be constructed by:

$$\hat{\mathbf{A}} = \mathbf{X}_{sc}\mathbf{X}_{sc}^T, \tag{6}$$

where $\mathbf{X}_{sc} \in R^{w \times h}$, w is the batch size, and h is the dimension of the structure scores. It is noticed that structure scores from source domain can be further constrained by the triplet loss [36] as in Eq. (7), which can guide the **Data Structure Analyzer** network to generate structure scores by modeling the structure similarity information of data samples.

$$\mathcal{L}_T = \max\left(\left\|\mathbf{X}_{sc_a} - \mathbf{X}_{sc_p}\right\|^2 - \left\|\mathbf{X}_{sc_a} - \mathbf{X}_{sc_n}\right\|^2 + \alpha_T, 0\right) \tag{7}$$

Where $\mathbf{X}_{sc_a}$ is randomly sampled from source domain. The $\mathbf{X}_{sc_p}$ is chosen from the same category with $\mathbf{X}_{sc_a}$, and $\mathbf{X}_{sc_n}$ is from a different category. The threshold α_T is a margin. Given the graph signal $\mathbf{X}$ and an adjacency matrix $\hat{\mathbf{A}}$, we can obtain the GCN feature according to the Eq. (4). As source and target CNN features are domain-discriminative in the early training, simultaneously constructing graphs may influence the stability of network training. Therefore, source and target graphs are individually constructed and fed into the parameters-shared GCNs to learn representations.

3.2.3 Class Centroid Alignment

The domain invariance and structure consistency are not equal to discriminability. For example, features of target category "laptops" can be mapped near features of source category "screens" while satisfying the condition of domain invariance and structure consistency. Separately, it has been shown that supervised domain adaptation (SDA) method improves unsupervised domain adaptation (UDA) by making use of class label information, since the SDA can ensure features of the same class from different domains to be mapped nearby [59]. This key observation motivates us to model class label information for the UDA via a class centroid alignment as in [77].

To design the class centroid alignment module, we follow the basic idea in [77]. Specifically, we firstly assign pseudo labels by using a target classifier $\mathbf{F}$ and obtain a pseudo-labeled target domain. The labeled and pseudo-labeled samples are used together to compute the centroid for each class. The centroid alignment objective for unsupervised domain adaptation is as follows.

$$\mathcal{L}_{CA}\left(\mathcal{X}_S, \mathcal{Y}_S, \mathcal{X}_T, \mathcal{Y}_T\right) = \sum_{k=1}^{K} \phi\left(\mathcal{C}_S^k, \mathcal{C}_T^k\right) \tag{8}$$

Where $\mathcal{C}_S^k$ and $\mathcal{C}_T^k$ are centroids of class k in the source and target domain respectively, and $\phi(\cdot,\cdot)$ is any proper distance measure function. We use the squard Euclidean distance $\phi(x, x') = \|x - x'\|^2$ in our experiments. Through explicitly restricting the distance between class centroids from different domains, we can ensure that features in the same class will be mapped nearby.

3.3. Discussion

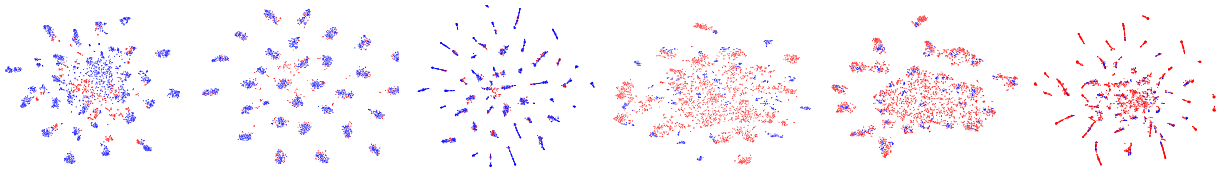
In this section, we show the differences among our model and two relevant methods including GAKT [18] and MSTN [77]. (1) The GAKT jointly optimizes target labels and domain-free features in a unified shallow framework. Meanwhile, the semi-supervised knowledge adaptation and a label propagation on target data are coupled to benefit each other. Different from the GAKT, our proposed model propagates the class label information by a simple class alignment loss term which is quite more comprehensive and effective. (2) The MSTN [77] learns semantic representations for unlabeled target samples by aligning labeled source centroid and pseudo-labeled target centroid. Both the MSTN and our model adopt the similar centroid alignment, but our model can employ structure representations to calculate class centroids, which is more accurate to measure the class-level domain divergency. (3) Unlike the GAKT and MSTN, our model not only utilizes the domain label and class label information, but also models the data structure information into a deep network. Overall, three types of vital information in unsupervised domain adaptation are jointly fused to learn domain-invariant representations and boost the performance of target task.

4. Experiments

In this section, we first illustrate the datasets, baseline methods, and implementation details. Then, we show extensive experimental results and analysis.

4.1. Datasets

Office-31 [63] is a benchmark dataset for domain adaptation, comprising 4,110 images in 31 classes collected from three distinct domains: Amazon (A), which contains images downloaded from amazon.com, Webcam (W) and DSLR (D), which contain images taken by web camera and



(a) AlexNet: $A \rightarrow W$ (b) RevGrad: $A \rightarrow W$ (c) GCAN: $A \rightarrow W$ (d) AlexNet: $W \rightarrow A$ (e) RevGrad: $W \rightarrow A$ (f) GCAN: $W \rightarrow A$

Figure 3. The representation visualization over transfer tasks $A \rightarrow W$ and $W \rightarrow A$. Here, we demonstrate the effectiveness of our method through the learned representation visualization using the t-distributed stochastic neighbor embedding (t-SNE) [56]. Blue points are source samples and red ones are target samples. (a) and (d) are trained without any adaptation. (b) and (e) are trained with previous adversarial domain adaptation methods. (c) and (f) are trained by our proposed method. Compared to the non-adapted methods, the adversarial adaptation methods successfully fuse the source and target domain features. However, the class label information is ignored and ambiguous features are generated near to the class boundary, which is catastrophic for the classification task. Different from existing methods, our model attempts to fuse features in the same class while separating features in different classes.

digital SLR camera with different photographical settings, respectively. To enable the unbiased evaluation, we evaluate all methods on all six transfer tasks A→W, D→W, W→D, A→D, D→A and W→A.

ImageCLEF-DA[1] is a benchmark dataset for ImageCLEF 2014 domain adaptation challenge, which is organized by selecting the 12 common categories shared by the following three public datasets. Here, each dataset is considered as a domain: *Caltrch-256* (**C**), *ImageNet ILSVRC 2012* (**I**), and *Pascal VOC 2012* (**P**). There are 50 images in each category and 600 images in each domain. We build 6 transfer tasks: I→P, P→I, I→C, C→I, C→P and P→C.

Office-Home [75] contains 4 domains, each with 65 categories including daily objects. Specifically, Art (Ar) denotes artistic depictions for object images, Clipart (Cl) means picture collection of clipart, Product (Pr) shows object images with a clear background and is similar to Amazon category in Office-31, and Real-World (Rw) represents object images collected with a regular camera. We use all domain combinations and build 12 transfer tasks.

4.2. Baseline Methods

On the Office-31 and ImageCLEF-DA datasets, we first use a stand deep learning network trained on source data, *e.g.*, Alexnet, to predict samples on target data, which provides a low bound of target performance. Then, we compare with state-of-art transfer learning methods including DDC [74], DRCN [27], RevGrad [22], RTN [52], JAN [53], AutoDIAL [9], and MSTN [77]. All baseline results are directly cited from these published papers. On the Office-Home dataset, we mainly compare with seven state-of-the-art shallow domain adaptation approaches including GFK [29], JDA [50], CCSL [57], LSC [38], JGSA [86], PUnDA [28], and GAKT [18]. We further compare to several deep domain adaptation models, *e.g.*, DAN [49], DHN [75], and WDAN [79]. All results are directly cited from [18] except the MSTN whose results are obtained by running the released code[2] by ourselves.

[1]https://www.imageclef.org/2014/adaptation

[2]https://github.com/Mid-Push/Moving-Semantic-Transfer-Network

4.3. Implementation Details

We follow the standard evaluation protocols for unsupervised domain adaptation as in [22, 49, 53]. We use all labeled source samples and all unlabeled target samples. We repeat each transfer task three times and report the mean accuracy and the standard error. The image random flipping and cropping strategies are adopted as in JAN [53]. The stochastic gradient decent with 0.9 momentum is used, and the learning rate is annealed by $\mu_p = \frac{\mu_0}{(1+\alpha \cdot p)^\beta}$, where $\mu_0 = 0.01$, $\alpha = 10$, and $\beta = 0.75$ [22]. We set the learning rate for finetuned layers to be 0.1 times of that from scratch. We set the batch size to 128 for each domain. The domain adversarial loss is scaled by 0.1 following [22].

Network Architecture. There are mainly four modules in our model, i.e., CNNs, Data Structure Analyzer (DSA), Domain Classifier, and GCNs. As for CNNs, we adopt the AlexNet architecture in all experiments. Following the RTN [52] and Revgrad [22], a bottleneck layer *fcb* with 256 units is added after the *fc7* layer in the AlexNet. For a fair comparison with other methods, we also finetune *conv1, conv2, conv3, conv4, conv5, fc6, fc7* layers in the pretrained AlexNet model. The Data Structure Analyzer (DSA) is implemented as the original pretrained AlexNet with a 1000-dimensional output. It is finetuned on source data during training with the triplet loss. For the architecture of GCNs, we only use one GCN. The dimension of node representation is 256 and the output dimension is 150. As for the domain classifier, we use the same architecture as the RevGrad [22], i.e. x→1024→1024→1. The dropout strategy is also used. In our implementation, all models and methods are implemented with Tensorflow.

Hyper-parameters Tuning. There are five hyper-parameters in our model, namely weight balance parameters λ, γ, η, the threshold α_T, and the moving average coefficient θ. For a fair comparison with other methods, we design the parameters λ, γ, θ in our experiments as the same values in [77]. The λ and γ are set as $\frac{2}{1+\exp(-k \cdot p)}$, where the k is set to 10, and the p is changing from 0 to 1 within the training process. Here, the λ and γ are optimized by [22] to

Table 1. Classification accuracy (%) on the Office-31 dataset.

Method	$A \rightarrow W$	$D \rightarrow W$	$W \rightarrow D$	$A \rightarrow D$	$D \rightarrow A$	$W \rightarrow A$	Avg
AlexNet	61.6±0.5	95.4±0.3	99.0±0.2	63.8±0.5	51.1±0.6	49.8±0.4	70.1
DDC [74]	61.8±0.4	95.0±0.5	98.5±0.4	64.4±0.3	52.1±0.6	52.2±0.4	70.6
DRCN [27]	68.7±0.3	96.4±0.3	99.0±0.2	66.8±0.5	56.0±0.5	54.9±0.5	73.6
RevGrad [22]	73.0±0.5	96.4±0.3	99.2±0.3	72.3±0.3	53.4±0.4	51.2±0.5	74.3
RTN [52]	73.3±0.3	96.8±0.2	99.6±0.1	71.0±0.2	50.5±0.3	51.0±0.1	73.7
JAN [53]	74.9±0.3	96.6±0.2	99.5±0.2	71.8±0.2	58.3±0.3	55.0±0.4	76.0
AutoDIAL [50]	75.5	96.6	99.5	73.6	58.1	59.4	77.1
MSTN [77]	80.5±0.4	96.9±0.1	**99.9±0.1**	74.5±0.4	62.5±0.4	60.0±0.6	79.1
GCAN	**82.7±0.1**	**97.1±0.1**	99.8±0.1	**76.4±0.5**	**64.9±0.1**	**62.6±0.3**	**80.6**

suppress the noisy signal from the discriminator at the early stages of training, and the noise brought by false labels. We also set $\theta = 0.7$, $\eta = 0.001$, $\alpha_T = 1$.

4.4. Results and Analysis

We next discuss experimental results on the Office-31, ImageCLEF-DA and Office-Home datasets. Due to the limited space, more results and analysis on other datasets can be found in the **supplementary material**.

Office-31. We follow the fully transductive evaluation protocol in [22]. Results are shown in Table 1. The proposed model outperforms all comparison methods on most transfer tasks. It is noteworthy that the proposed GCAN can effectively improve four hard transfer tasks including $A \rightarrow W, A \rightarrow D, W \rightarrow A$, and $D \rightarrow A$. On these difficult tasks, our method promotes classification accuracies substantially. The encouraging improvement on the hard transfer tasks proves the importance of modeling structure-aware alignment, domain alignment and class centroid alignment jointly for unsupervised domain adaptation. Furthermore, the results demonstrate that our model can effectively exploit the information of data structure, domain label and class label to reduce the domain discrepancy. In addition, the results reveal several interesting observations. (1) Deep transfer learning methods outperform standard deep learning methods, *e.g.*, the AlexNet. It validates that domain shift cannot be removed by deep networks. (2) The DRCN trains an extra decoder to enforce the extracted features containing semantic information and thus outperforms the Alexnet by about 4%. This improvement also indicates the importance to learn semantic representations. (3) Separately, the distribution matching methods RevGrad, RTN and JAN also bring significant improvement over the AlexNet trained only on source data. Our method exploits the advantages of the DRCN and distribution matching methods in a very simple form. In particular, in contrast to using a decoder to extract semantic information, our method leverages the class label information by the class centroid alignment so that the features in same classes from different domains are mapped nearby. (4) Compared with the state-of-the-art MSTN, our GCAN significantly improves the performance, which proves the effectiveness of modeling data structure in a deep network for domain adaptation.

For completeness, as shown in Figure 3, we also conduct representation visualizations for our model and the RevGrad over transfer tasks $A \rightarrow W$ and $W \rightarrow A$. From the comparison between Figure 3(c) and 3(f), we can clearly observe two interesting experimental phenomena: (1) The target representations in different classes are more dispersed in our method, which illustrates that the learned target representations are more discrimitative. (2) The source and target representations are perfectly matched in our methods, which proves the learned representations are domain-invariant. The above two points illustrate that our model can achieve better performance than the RevGrad.

Table 2. Classification accuracy (%) on ImageCLEF-DA dataset.

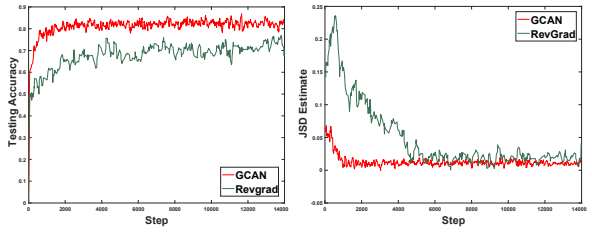
Method	$I \rightarrow P$	$P \rightarrow I$	$I \rightarrow C$	$C \rightarrow I$	$C \rightarrow P$	$P \rightarrow C$	Avg
AlexNet	66.2±0.2	70.0±0.2	84.3±0.2	71.3±0.4	59.3±0.5	84.5±0.3	73.9
RTN [52]	67.4±0.3	81.3±0.3	89.5±0.4	78.0±0.2	62.0±0.2	89.1±0.1	77.9
RevGrad [22]	66.5±0.5	81.8±0.4	89.0±0.5	79.8±0.5	63.5±0.4	88.7±0.4	78.2
JAN [53]	67.2±0.5	82.8±0.4	91.3±0.5	80.0±0.5	63.5±0.4	91.0±0.4	79.3
MSTN [77]	67.3±0.3	82.8±0.2	91.5±0.1	81.7±0.3	65.3±0.2	91.2±0.2	80.0
GCAN	**68.2±0.5**	**84.1±0.2**	**92.2±0.1**	**82.5±0.1**	**67.2±0.2**	**91.3±0.1**	**80.9**

ImageCLEF-DA. Results are shown in Table 2. From the results, we can draw the following conclusions: (1) The three domains on the ImageCLEF-DA dataset are more balanced than those on the Office-31 dataset. With these more balanced transfer tasks, we can testify whether transfer learning can be improved when domain sizes do not change. From the results, our model can still outperform existing methods in all transfer tasks but with less improvement compared to the results on the Office-31 dataset, which validates the hypothesis that the domain size may cause domain shift [53]. (2) Our GCAN model outperforms all baseline methods on most transfer tasks. In particular, the GCAN substantially improves the accuracy on the hard transfer tasks, e.g. $A \rightarrow W$ and $C \rightarrow W$, where the source and target domains are very different, and achieves comparable accuracy on the easy transfer tasks, $D \rightarrow W$ and $W \rightarrow D$, where two domains are similar. These results suggest that the GCAN is able to learn more adaptive classifiers and transferable features for the safer domain adaptation. (3) The GCAN models outperform previous methods and sets a new state of the art record. Compared with the JAN which adapts the joint distributions of network activations in all domain-specific layers to fully correct the shifts in joint distributions across domains, our GCAN shows better performance by a large margin, which further illustrates the effectiveness of the three alignment mechanisms.

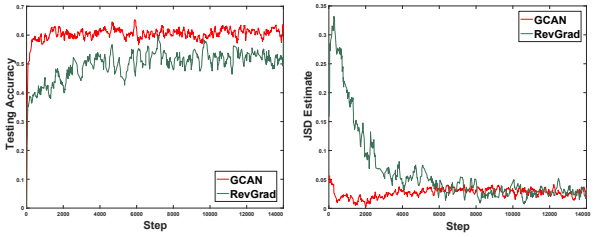
Office-Home. Results on the Office-Home dataset are shown in Table 3. The proposed model outperforms all baseline methods on all transfer tasks by a large margin and even exceeds the recent state-of-the-art MSTN about 3%. From the results, we have the following observations. (1) Comparing with shallow methods and deep methods, *e.g.*, PUnDA and DAN, the performance of PUnDA and DAN is approximately at the same level, which confirms that data structure matching is vital in domain adaptation problem. (2) It has been proved that very deep convolutional network-

Table 3. Recognition accuracies (%) for cross-domain experiments on the Office+Home dataset.

Source Target	Ar Cl	Ar Pr	Ar Rw	Cl Ar	Cl Pr	Cl Rw	Pr Ar	Pr Cl	Pr Rw	Rw Ar	Rw Cl	Rw Pr	Avg
GFK [29]	21.60	31.72	38.83	21.63	34.94	34.20	24.52	25.73	42.92	32.88	28.96	50.89	32.40
JDA [50]	25.34	35.98	42.94	24.52	40.19	40.90	25.96	32.72	49.25	35.10	35.35	55.35	36.97
CCSL [57]	23.51	34.12	40.02	22.54	35.69	36.04	24.84	27.09	46.36	34.61	31.75	52.89	34.12
LSC [38]	31.81	39.42	50.25	35.46	51.19	51.43	30.46	39.54	59.74	43.98	42.88	62.25	44.87
RTML [17]	27.57	36.20	46.09	29.49	44.69	44.66	28.21	36.12	52.99	38.54	40.62	57.80	40.25
JGSA [86]	28.81	37.57	48.92	31.67	46.30	46.76	28.72	35.90	54.47	40.61	40.83	59.16	41.64
PUnDA [28]	29.99	37.76	50.17	33.90	48.91	48.71	30.31	38.69	56.91	42.25	44.51	61.05	43.60
DAN [49]	30.66	42.17	54.13	32.83	47.59	49.78	29.07	34.05	56.70	43.58	38.25	62.73	43.46
DHN [75]	31.64	40.75	51.73	34.69	51.93	52.79	29.91	39.63	60.71	44.99	45.13	62.54	45.54
WDAN [79]	32.26	43.16	54.98	34.28	49.92	50.26	30.82	38.27	56.87	44.32	39.35	63.34	44.82
GAKT [18]	34.49	43.63	55.28	36.14	52.74	53.16	31.59	40.55	61.43	45.64	44.58	64.92	47.01
MSTN [77]	34.87	46.20	56.77	36.63	54.97	55.41	33.27	41.66	60.62	46.94	45.90	68.25	48.46
GCAN	**36.43**	**47.25**	**61.08**	**37.90**	**58.25**	**57.00**	**35.77**	**42.66**	**64.47**	**50.08**	**49.12**	**72.53**	**51.05**



(a) Accuracy $A \rightarrow W$ (b) JSD $A \rightarrow W$



(c) Accuracy $W \rightarrow A$ (d) JSD $W \rightarrow A$

Figure 4. (a) and (c): Comparison of different models. (b) and (d): Comparison of the Revgrad and our GCAN on the Jensen Shannon divergence (JSD) estimation during training. Our model stabilizes and accelerates the adversarial learning process.

s, *e.g.*, VGGnet [68], GoogLeNet [71] and ResNet [34], is able to not only learn better representations for general vision tasks, but also learn more transferable representations for domain adaptation [53]. It is noticed that all methods use the VGG as their base feature extractor, while the MSTN and GCAN use the AlexNet for feature extraction. Surprisingly, the MSTN and GCAN can still surpass other shallow and deep methods, which can verify the great power of the class centroid alignment. (3) Unlike the GAKT and MSTN, our model not only utilizes domain label and class label information but also models data structure information into a deep network. The results can illustrate the effectiveness of the three alignment mechanisms.

4.5. Further Remarks

Convergence. As our model involves the adversarial adaptation module, we testify its performance on the convergence from two different aspects. The first is the testing accuracy as shown in Figure 4(a) and 4(c). Our model has similar convergence speed as the RevGrad. Since the adversarial module in our model and the RevGrad are analogous to the GAN [30], we will evaluate our model from the perspective of the GAN. It has been proved that when the discriminator is optimal, the generator involved in the min-max game can reduce the Jenson-Shannon Divergence (JSD). For the discriminator in the adversarial adaptation, it is trained to maximize $\mathcal{L}_D = \mathbb{E}_{x\sim D_S}\left[\log\left(1-D\left(x\right)\right)\right] + \mathbb{E}_{x\sim D_T}\left[\log D\left(x\right)\right]$, which is a low bound of $2JS\left(D_S, D_T\right) - 2\log 2$ [2]. Therefore, following [2], we plot the quantity of $\frac{1}{2}JS\left(D_S, D_T\right) + \log 2$, which is the lower bound of the JS distance. Results are shown in Figure 4(b) and 4(d). We can make the following observations: (1) Different from the vanishing generator gradient problem in traditional GANs, the manifold, where features generated by the adversarial adaptation methods lie, seems to be perfectly aligned. Therefore, the gradients for the feature extractor will not vanish but lead to reduce the JS distance. This justifies the feasibility for the adversarial domain adaptation methods. (2) Compared to the RevGrad, our model is more stable and can accelerate the minimization process for the JSD. It indicates that our method stabilizes the notorious unstable adversarial training through the three alignment mechanisms.

5. Conclusion

In this paper, we propose a novel method to jointly leverage the data structure, domain label and class label information in a unified deep network for unsupervised domain adaptation. To match source and target domain distributions robustly, we design three effective alignment mechanisms including structure-aware alignment, domain alignment and class centroid alignment. These three alignment mechanisms can enhance and complement each other to learn domain-invariant and discriminative representations for target task. Experiments on standard domain adaptation datasets verify the effectiveness of the proposed model.

Acknowledgements

This work is supported in part by the National Natural Science Foundation of China under Grants 61432019, 61572498, 61532009, 61728210, 61721004, 61751211, 61720106006 and U1705262, and the Key Research Program of Frontier Sciences, CAS, Grant NO. QYZDJ-SSW-JSC039, the Beijing Natural Science Foundation 4172062, and Youth Innovation Promotion Association CAS 2018166.

References

[1] Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, and Mario Marchand. Domain-adversarial neural networks. *arXiv preprint arXiv:1412.4446*, 2014.

[2] Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862*, 2017.

[3] Yan Bai, Yihang Lou, Feng Gao, Shiqi Wang, Yuwei Wu, and Ling-Yu Duan. Group-sensitive triplet embedding for vehicle reidentification. *IEEE Transactions on Multimedia*, 20(9):2385–2399, 2018.

[4] Slawomir Bak, Peter Carr, and Jean-Francois Lalonde. Domain adaptation through synthesis for unsupervised person re-identification. In *ECCV*, pages 189–205, 2018.

[5] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. In *NIPS*, pages 343–351, 2016.

[6] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013.

[7] Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Michael I. Jordan. Partial transfer learning with selective adversarial networks. In *CVPR*, pages 2724–2732, 2018.

[8] Zhangjie Cao, Lijia Ma, Mingsheng Long, and Jianmin Wang. Partial adversarial domain adaptation. In *ECCV*, pages 135–150, 2018.

[9] Fabio Maria Cariucci, Lorenzo Porzi, Barbara Caputo, Elisa Ricci, and Samuel Rota Bulo. Autodial: Automatic domain alignment layers. In *ICCV*, pages 5077–5085, 2017.

[10] Jiaxin Chen and Yi Fang. Deep cross-modality adaptation via semantics preserving adversarial learning for sketch-based 3d shape retrieval. In *ECCV*, pages 605–620, 2018.

[11] Minmin Chen, Kilian Q Weinberger, and John Blitzer. Co-training for domain adaptation. In *NIPS*, pages 2456–2464, 2011.

[12] Qingchao Chen, Yang Liu, Zhaowen Wang, Ian Wassell, and Kevin Chetty. Re-weighted adversarial adaptation network for unsupervised domain adaptation. In *CVPR*, pages 7976–7985, 2018.

[13] Wen-Sheng Chu, Fernando De la Torre, and Jeffery F Cohn. Selective transfer machine for personalized facial action unit detection. In *CVPR*, pages 3515–3522, 2013.

[14] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1853–1865, 2017.

[15] Gabriela Csurka. Domain adaptation for visual applications: A comprehensive survey. *arXiv preprint arXiv:1702.05374*, 2017.

[16] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *NIPS*, pages 3844–3852, 2016.

[17] Zhengming Ding and Yun Fu. Robust transfer metric learning for image classification. *IEEE Transactions on Image Processing*, 26(2):660–670, 2017.

[18] Zhengming Ding, Sheng Li, Ming Shao, and Yun Fu. Graph adaptive knowledge transfer for unsupervised domain adaptation. In *ECCV*, pages 37–52, 2018.

[19] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, pages 647–655, 2014.

[20] Lixin Duan, Dong Xu, and Shih-Fu Chang. Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach. In *CVPR*, pages 1338–1345, 2012.

[21] David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. In *NIPS*, pages 2224–2232, 2015.

[22] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, pages 1180–1189, 2015.

[23] 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.

[24] Hongyang Gao, Zhengyang Wang, and Shuiwang Ji. Large-scale learnable graph convolutional networks. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1416–1424, 2018.

[25] Junyu Gao, Tianzhu Zhang, and Changsheng Xu. Watch, think and attend: End-to-end video classification via dynamic knowledge evolution modeling. In *ACM MM*, 2018.

[26] Junyu Gao, Tianzhu Zhang, and Changsheng Xu. I know the relationships: Zero-shot action recognition via two-stream graph convolutional networks and knowledge graphs. In *AAAI*, 2019.

[27] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, David Balduzzi, and Wen Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In *ECCV*, pages 597–613, 2016.

[28] Behnam Gholami, Ognjen Rudovic, and Vladimir Pavlovic. Punda: Probabilistic unsupervised domain adaptation for knowledge transfer across visual categories. In *ICCV*, pages 3601–3610, 2017.

[29] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*, pages 2066–2073, 2012.

[30] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.

[31] Marco Gori, Gabriele Monfardini, and Franco Scarselli. A new model for learning in graph domains. In *IJCNN*, pages 729–734, 2005.

[32] Arthur Gretton, Karsten M Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex J Smola. A kernel method for the two-sample-problem. In *NIPS*, pages 513–520, 2007.

[33] Junwei Han, Xiang Ji, Xintao Hu, Dajiang Zhu, Kaiming Li, Xi Jiang, Guangbin Cui, Lei Guo, and Tianming Liu. Representing and retrieving video shots in human-centric brain imaging space. *IEEE Transactions on Image Processing*, 22(7):2723–2736, 2013.

[34] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[35] Mikael Henaff, Joan Bruna, and Yann LeCun. Deep convolutional networks on graph-structured data. *arXiv preprint arXiv:1506.05163*, 2015.

[36] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

[37] Judy Hoffman, Erik Rodner, Jeff Donahue, Brian Kulis, and Kate Saenko. Asymmetric and category invariant feature transformations for domain adaptation. *International journal of computer vision*, 109(1-2):28–41, 2014.

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

[39] Jiayuan Huang, Arthur Gretton, Karsten M Borgwardt, Bernhard Schölkopf, and Alex J Smola. Correcting sample selection bias by unlabeled data. In *NIPS*, pages 601–608, 2007.

[40] Melih Kandemir. Asymmetric transfer learning with deep gaussian processes. In *ICML*, pages 730–738, 2015.

[41] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. In *ICML*, pages 1857–1865, 2017.

[42] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[43] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

[44] Sofia Ira Ktena, Sarah Parisot, Enzo Ferrante, Martin Rajchl, Matthew Lee, Ben Glocker, and Daniel Rueckert. Distance metric learning using graph convolutional networks: Application to functional brain networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 469–477, 2017.

[45] Jingjing Li, Ke Lu, Zi Huang, Lei Zhu, and Heng Tao Shen. Transfer independently together: A generalized framework for domain adaptation. *IEEE Transactions on Cybernetics*, 2018.

[46] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493*, 2015.

[47] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In *NIPS*, pages 469–477, 2016.

[48] Si Liu, Zhen Wei, Yao Sun, Xinyu Ou, Junyu Lin, Bin Liu, and Ming-Hsuan Yang. Composing semantic collage for image retargeting. *IEEE Transactions on Image Processing*, 27(10):5032–5043, 2018.

[49] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *ICML*, pages 97–105, 2015.

[50] Mingsheng Long, Guiguang Ding, Jianmin Wang, Jiaguang Sun, Yuchen Guo, and Philip S Yu. Transfer sparse coding for robust image representation. In *CVPR*, pages 407–414, 2013.

[51] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S Yu. Transfer joint matching for unsupervised domain adaptation. In *CVPR*, pages 1410–1417, 2014.

[52] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *NIPS*, pages 136–144, 2016.

[53] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *ICML*, pages 2208–2217, 2017.

[54] Lingkun Luo, Xiaofang Wang, Shiqiang Hu, and Liming Chen. Robust data geometric structure aligned close yet discriminative domain adaptation. *arXiv preprint arXiv:1705.08620*, 2017.

[55] Zelun Luo, Yuliang Zou, Judy Hoffman, and Li F Fei-Fei. Label efficient learning of transferable representations acrosss domains and tasks. In *NIPS*, pages 165–177, 2017.

[56] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[57] Tzu Ming Harry Hsu, Wei Yu Chen, Cheng-An Hou, Yao-Hung Hubert Tsai, Yi-Ren Yeh, and Yu-Chiang Frank Wang. Unsupervised domain adaptation with imbalanced cross-domain data. In *ICCV*, pages 4121–4129, 2015.

[58] Saeid Motiian, Quinn Jones, Seyed Iranmanesh, and Gianfranco Doretto. Few-shot adversarial domain adaptation. In *NIPS*, pages 6670–6680, 2017.

[59] Saeid Motiian, Marco Piccirilli, Donald A Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *ICCV*, pages 5715–5725, 2017.

[60] Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov. Learning convolutional neural networks for graphs. In *ICML*, pages 2014–2023, 2016.

[61] Sinno Jialin Pan, Qiang Yang, et al. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.

[62] Artem Rozantsev, Mathieu Salzmann, and Pascal Fua. Beyond sharing weights for deep domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(4):801–814, 2019.

[63] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *ECCV*, pages 213–226, 2010.

[64] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009.

[65] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference*, pages 593–607, 2018.

[66] Ming Shao, Dmitry Kit, and Yun Fu. Generalized transfer subspace learning through low-rank constraint. *International Journal of Computer Vision*, 109(1-2):74–93, 2014.

[67] Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

[68] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[69] Masashi Sugiyama, Neil D Lawrence, Anton Schwaighofer, et al. *Dataset shift in machine learning*. 2017.

[70] Sainbayar Sukhbaatar, Rob Fergus, et al. Learning multiagent communication with backpropagation. In *NIPS*, pages 2244–2252, 2016.

[71] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015.

[72] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *ICCV*, pages 4068–4076, 2015.

[73] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *CVPR*, pages 7167–7176, 2017.

[74] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014.

[75] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *CVPR*, pages 5385–5394, 2017.

[76] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big Data*, 3(1):9, 2016.

[77] Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *ICML*, pages 5419–5428, 2018.

[78] Yong Xu, Xiaozhao Fang, Jian Wu, Xuelong Li, and David Zhang. Discriminative transfer subspace learning via low-rank and sparse representation. *IEEE Transactions on Image Processing*, 25(2):850–863, 2016.

[79] 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 *CVPR*, pages 2272–2281, 2017.

[80] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. *arXiv preprint arXiv:1801.07455*, 2018.

[81] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *CVPR*, pages 3166–3173, 2013.

[82] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *ICCV*, pages 2868–2876, 2017.

[83] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *NIPS*, pages 3320–3328, 2014.

[84] Jing Yu, Yuhang Lu, Zengchang Qin, Weifeng Zhang, Yanbing Liu, Jianlong Tan, and Li Guo. Modeling text with graph convolutional network for cross-modal information retrieval. In *Pacific Rim Conference on Multimedia*, pages 223–234, 2018.

[85] Dingwen Zhang, Deyu Meng, and Junwei Han. Co-saliency detection via a self-paced multiple-instance learning framework. *IEEE transactions on pattern analysis and machine intelligence*, 39(5):865–878, 2017.

[86] Jing Zhang, Wanqing Li, and Philip Ogunbona. Joint geometrical and statistical alignment for visual domain adaptation. In *CVPR*, pages 5150–5158, 2017.

[87] Yuhui Zheng, Le Sun, Shunfeng Wang, Jianwei Zhang, and Jifeng Ning. Spatially regularized structural support vector machine for robust visual tracking. *IEEE transactions on neural networks and learning systems*, (99):1–11, 2018.

[88] Yang Zou, Zhiding Yu, B.V.K. Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, pages 289–305, 2018.