Representation Learning with Multiple Lipschitz-Constrained Alignments on Partially-Labeled Cross-Domain Data

Songlei Jian,[†] Liang Hu,[‡] Longbing Cao,[‡] Kai Lu[†]
[†]College of Computer, National University of Defense Technology, China
[‡]Advanced Analytics Institute, University of Technology Sydney, Australia
{jiansonglei, kailu}@nudt.edu.cn, rainmilk@gmail.com, longbing.cao@uts.edu.au

Abstract

The cross-domain representation learning plays an important role in tasks including domain adaptation and transfer learning. However, existing cross-domain representation learning focuses on building one shared space and ignores the unlabeled data in the source domain, which cannot effectively capture the distribution and structure heterogeneities in cross-domain data. To address this challenge, we propose a new cross-domain representation learning approach: *MUltiple Lipschitz-constrained AligNments (MULAN)* on partially-labeled cross-domain data. MULAN produces two representation spaces: a *common representation* space to incorporate knowledge from the source domain and a *complementary representation* space to complement the common representation with target local topological information by Lipschitz-constrained representation transformation. MULAN utilizes both unlabeled and labeled data in the source and target domains to address distribution heterogeneity by Lipschitz-constrained adversarial distribution alignment and structure heterogeneity by cluster assumption-based class alignment while keeping the target local topological information in complementary representation by self alignment. Moreover, MULAN is effectively equipped with a customized learning process and an iterative parameter updating process. MULAN shows its superior performance on partially-labeled semi-supervised domain adaptation and few-shot domain adaptation and outperforms the state-of-the-art visual domain adaptation models by up to 12.1%.

Introduction

Learning tasks including domain adaptation (DA) (Rozantsev, Salzmann, and Fua 2018), transfer learning (TL) (Pan and Yang 2010), and image-image translation (Liu et al. 2018a; Lee et al. 2018) aim to acquire knowledge from one domain to enhance that of the other. Typically, such tasks transfer a knowledge representation from a source domain to a target domain by jointly learning a common representation across domains. To date, learning the cross-domain representation is still highly challenging due to the heterogeneities of data distributions and structures between domains, which widely exist in real-world applications. In addition to the common information shared across domains, learning heterogeneous but complementary information between domains for representation is promising yet challenging. It will be very promising if a representation learning approach can effectively learn complementary information from other domains, which enables the knowledge of machines to be self-evolutionary. However, it is still a challenging task for existing cross-domain representation learning methods because only limited consensus data (e.g., labeled data) are available to construct the linkage between domains and the data distributions and structures are usually highly heterogeneous in different domains.

Inspired by this, we propose a novel cross-domain representation learning approach: Multiple Lipschitz-constrained Alignments (MULAN) to construct both common representations and complementary representations for partially-labeled cross-domain data. Different from existing semi-supervised (Motiian et al. 2017; Tzeng et al. 2015) or unsupervised domain adaptation methods (Haeusser et al. 2017; Shu et al. 2018; Xie et al. 2018) in which the fully-labeled source data is available during learning, MULAN utilizes partially labeled data in both source and target domains, which is more common in real-life applications. Also, classic cross-domain representation learning aims to build one common feature space in which the local topological and distribution information of the target domain may be ignored during the alignment with the source domain. In contrast, MULAN involves Lipschitz-constrained representation transformation to generates two representation spaces: a common representation space and a complementary representation space. In the common space, the representations of target data follow the distribution of the source domain, while the representations in the complementary space aim to keep the topological information of the target domain which complements the information loss of the common space.

Our key idea is to build multiple alignments based on the Lipschitz-constrained transformation, i.e., class alignment, distribution alignment and self alignment between source and target domains. The class alignment makes use of the labeled data and is based on the cluster assumption which is widely taken in semi-supervised learning in order to help the source and target alignment in the common space. The distribution alignment uses adversarial learning to generate

a common representation of target domain which follows the distribution of source domain. Moreover, the self alignment between original target feature space and complementary representation space emphasizes the local information in target domain and also constrains the learning of class alignment and distribution alignment. All these three alignments jointly build a training target to learn the representations with a customized parameter learning process. The contributions of this work include:

- A new cross-domain representation learning approach MULAN produces two representation spaces, i.e., common representation and complementary representation for the target domain, which addresses the shortage of existing methods by only learning common representation.
- MULAN only takes partially-labeled data in both source and target domains to accomplish the alignment between two domains while also keeps the local topological information in the complementary representation.
- A customized parameter learning process is designed for MULAN so that the multiple alignments can be jointly and effectively learned in the training process.

We show that our method beats the state-of-the-art DA methods on the representative VisDA dataset in both partially-labeled semi-supervised domain adaptation and limited-labeled few-shot domain adaptation.

Related Work

The most related problems studied on cross-domain representation learning are domain adaptation and representation disentanglement.

Domain Adaptation DA leverages labels in one to multiple source domains to predict the labels in a target domain with inconsistent data distribution. The core aspect of DA is to find a domain invariant representation or embedding space. Existing methods use various metrics to measure the similarity between the representations of source data and target data. Maximum mean discrepancy is a popular metric in most kernel-based DA methods (Kulis, Saenko, and Darrell 2011; Gong et al. 2012). Associative Domain Adaptation (ADA) (Haeusser et al. 2017) utilizes association (Haeusser, Mordvintsev, and Cremers 2017) in semi-supervised training to enforce similar embeddings. Adversarial methods (Ganin et al. 2016; Tzeng et al. 2017; Shen et al. 2018; Long et al. 2018) have received more focus recently. Moreover, to address some limitations of domain adversarial training, the authors in (Shu et al. 2018) proposed a virtual adversarial DA model with Decision-boundary Iterative Refinement Training with a Teacher (DIRT-T). Generative adversarial networks recently show their potential in DA (Yi et al. 2017; Kim et al. 2017) for generating virtual samples. Some Siamese architectures (Tzeng et al. 2015; Sun and Saenko 2016) are proposed to optimize the domain invariance to facilitate domain transfer and learn a discriminative embedding subspace, where the mapped domains are semantically aligned yet maximally separated. The work (Motiian et al. 2017) utilizes few target examples to build Classification and Contrastive Semantic Alignment (CCSA) for domain adaptation and generalization. However, existing DA models only generate one common feature space which may lose information from the target domain. Method in (Bousmalis et al. 2016) builds shared representation space and private representation space for each domain while these private representation cannot complement the target domain learning.

Representation Disentanglement To learn cross-domain representation, representative methods (Liu et al. 2018a; Lee et al. 2018; Liu et al. 2018b) try to disentangle the underlying factors of data from different domains in order to help cross-domain knowledge share and transfer. Recently, most methods (Odena, Olah, and Shlens 2017; Higgins et al. 2017; Chen et al. 2016; Kingma et al. 2014; Liu et al. 2018b) are based on generative adversarial networks (GAN) (Goodfellow et al. 2014) and Variational Autoencoder (VAE) (Rezende, Mohamed, and Wierstra 2014). The method (Liu et al. 2018b) utilizes the fully-labeled source data to perform representation learning and disentanglement in the resulting shared latent space. The work (Gonzalez-Garcia, van de Weijer, and Bengio 2018) disentangles representation from two domains through VAEs and construct common representation. Another work (Liu et al. 2018a) achieves multiple domain confusions in order to customize image generation and translation. However, the representation disentanglement highly depends on a few explanatory factors and assumptions (Rezende, Mohamed, and Wierstra 2014).

Problem Formalization

For a cross-domain representation learning problem, we denote the domain data distribution as $\mathcal{D}$ with a label set $\mathcal{C}$ on the input feature $\mathcal{X}$ and a labeling function $l : \mathcal{X} \mapsto \mathcal{C}$ to retrieve the labels. Let us consider two domains, $(\mathcal{D}_s, l_s)$ denotes the source domain and $(\mathcal{D}_t, l_t)$ denotes the target domain. The inputs $\mathcal{X}_s$ and $\mathcal{X}_t$ are the pre-trained features of source and target domains. $\mathbf{H}_s$ and $\mathbf{H}_t$ are the projected representation spaces w.r.t. the source and target domains. $\mathbf{H}_{t\text{-}s}$ and $\mathbf{H}_{t\text{-}s\text{-}t}$ denote the common representation space and complementary representation space of the target domain respectively. $\mathbf{H}_s$ and $\mathbf{H}_{t\text{-}s}$ are compatible and follow the same distribution. And $\mathbf{h}$ denotes the specific representation of one data sample in corresponding representation space $\mathbf{H}$.

In partially-labeled DA task, we have the labeled data $\mathcal{X}_s^l \in \mathcal{X}_s$ with the labels $\mathcal{Y}_s^l = \{y_s = l_s(\mathbf{x}_s) | \mathbf{x}_s \in \mathcal{X}_s^l\}$ and unlabeled data $\mathcal{X}_s^u \in \mathcal{X}_s$ in the source domain which is more consistent with real-world data. Similarly, we have the labeled data $(\mathcal{X}_t^l, \mathcal{Y}_t^l)$ and unlabeled data $\mathcal{X}_t^u \in \mathcal{X}_t$ in the target domain. Our goal is to find a labeling function h from a hypothesis space $\mathcal{H}$, which can minimize the generalization errors over $\mathcal{X}_t^u$.

Multiple Lipschitz-constrained Alignments

MULAN defines a two-stage representation learning process corresponding to learning common representation and learning complementary representation with the constraints of three alignments, as illustrated in Figure 1. In the first stage,

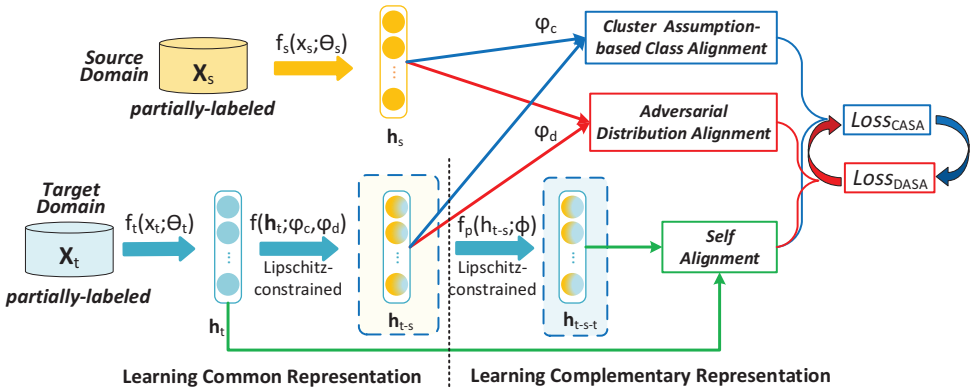


Figure 1: The overview of the working mechanism of MULAN, which generates two representation spaces for the target domain.

the common representation $\mathbf{h}_{t\text{-}s}$ is learned by following the class alignment and distribution alignment with source projected representation $\mathbf{h}_s$ (in terms of the learning objective Eqn. 7 and Eqn. 8). In the second stage, the complementary representation $\mathbf{h}_{t\text{-}s\text{-}t}$ is mainly aligned with the target-projected representation $\mathbf{h}_t$ through self alignment (in terms of learning objective Eqn. 9). Moreover, the three alignments are jointly optimized through two combined loss functions (see the loss function in Eqn. 12). And the two losses are optimized iteratively to update the parameters.

Common Representation Learning

The common representation learning is built on the projected representation spaces, $\mathbf{H}_t$ and $\mathbf{H}_s$, which are encoded from target and source data by the representation projection models, i.e., $f_s(\mathcal{X}_s;\theta_s) : \mathcal{X}_s \mapsto \mathbf{H}_s$ and $f_t(\mathcal{X}_t;\theta_t) : \mathcal{X}_t \mapsto \mathbf{H}_t$, where θ_s and θ_t are the model parameters. Since $\mathbf{H}_t$ and $\mathbf{H}_s$ are from different domains and follow different distributions, they are not directly comparable. Hence, we transform $\mathbf{H}_t$ into the common representation space $\mathbf{H}_{t\text{-}s}$ which complies with the the distribution of $\mathbf{H}_s$ to align source labeling information with the target. And the transformation function is $f_c(\mathbf{h}_t;\varphi_c,\varphi_d) : \mathbf{H}_t \mapsto \mathbf{H}_{t\text{-}s}$, where φ_c and φ_d are the parameters in class alignment and distribution alignment respectively. More specifically, the transformation function $f(\mathbf{h}_t;\varphi_c,\varphi_d)$ can be divided as the following:

$$\mathbf{h}_{t\text{-}s} = f(\mathbf{h}_t;\varphi_c,\varphi_d) = (1-\gamma)f_c(\mathbf{h}_t;\varphi_c) + \gamma f_d((\mathbf{h}_t;\varphi_d) \quad (1)$$

To keep the local geometric information in $\mathbf{h}_t$, we constrain $f_c(\mathbf{h}_t;\varphi_c)$ and $f_d(\mathbf{h}_t;\varphi_d)$ satisfying *1-Lipschitz* which is defined in Definition 1. And γ is the hyper-parameter which will be introduced in detail with the distribution alignment. Due to the convex combination, we can easily verify that $f(\mathbf{h}_t;\varphi_c,\varphi_d)$ also satisfy *1-Lipschitz* in terms of the triangle inequality. According to the definition and properties of Lipschitz continuity, f has bounded first derivative which means the local structure is kept during transformation.

Definition 1. *(Lipschitz continuity). A vector-valued function $f : \mathbb{R}^n \mapsto \mathbb{R}^m$ is K-Lipschitz continuous on $\mathbb{R}^n$ if there is a constant K such that*

$$\|f(\mathbf{x}) - f(\mathbf{y})\| \leq K\|\mathbf{x} - \mathbf{y}\| \quad \textit{for all} \quad \mathbf{x},\mathbf{y} \in \mathbb{R}^n \quad (2)$$

where the smallest K is a Lipschitz constant which equals $\sup_{\mathbf{x}} \|\nabla f(\mathbf{x})\|$.

In particular, we employ the multilayer perceptrons with spectral normalization (SNMLPs) (Miyato et al. 2018) to implement *1-Lipschitz* functions in this work.

During the learning of common representation, the class alignment searches a configuration of $f_c(\mathbf{h}_t;\varphi_c)$ to align target and source with labeled data while the distribution alignment finds a configuration of $f_c(\mathbf{h}_t;\varphi_d)$ by finely aligning the target distribution with the source distribution using both labeled and unlabeled data.

Cluster Assumption-based Class Alignment Although the domain consensus information, i.e., class labels, between source and target domains are limited, it still plays a critical role in aligning two domains, especially when the conditional distribution $\mathcal{D}_t(\mathcal{X}|\mathcal{C})$ is significantly different from $\mathcal{D}_s(\mathcal{X}|\mathcal{C})$, which will be discussed with distribution alignment. *Cluster Assumption* is commonly made to account for the success of semi-supervised learning (SSL) (Ben-David and Urner 2014; Shu et al. 2018), which assumes that the input distribution can be divided into clusters separated by low-density regions, where the data points in the same cluster share almost the same label. According to the cluster assumption, for the cross-domain learning tasks, the limited consensus class information is significantly helpful to align the clusters associated with the same major labels across domains. As a result, we consider the max-margin class separation (MMCS) loss to enlarge the margin of data representation w.r.t. different classes for clearer cluster boundaries. The general form of MMCS w.r.t. two domains is as follows:

$$\mathcal{L}_m^{\mathcal{C}}(\mathcal{A}|\mathcal{B}) = \mathbb{E}_{\mathbf{a}\in\mathcal{A}}\mathbb{E}_{\mathbf{b_i},\mathbf{b_j}\in\mathcal{B}}(\delta_l \cdot [d(\mathbf{a},\mathbf{b_i}) - d(\mathbf{a},\mathbf{b_j}) + m]_0) \quad (3)$$

where $d(\cdot,\cdot)$ is a metric function, and it is considered as Euclidean distance in this paper without further notification, and $[x]_0$ denotes $\max(0,x)$. δ_l is an indicator w.r.t. the labeling function l under the class distribution $\mathcal{C}$:

$$\delta_l = \begin{cases} 1, & \text{if } l(\mathbf{a}) = l(\mathbf{b_i}) \wedge l(\mathbf{a}) \neq l(\mathbf{b_j}) \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

Minimizing MMCS loss separates the input with different labels into different clusters by a margin m. The class alignment minimizes a complex MMCS loss $\mathcal{L}_{CA}$ composed of an intra-domain MMCS loss $\mathcal{L}_{intra}$ within target domain and source domain and an inter-domain MMCS loss $\mathcal{L}_{inter}$ between target and source domains.

$$\begin{aligned} \mathcal{L}_{intra} &= \mathcal{L}_m^{\mathcal{C}}(\mathbf{H}_t|\mathbf{H}_t) + \mathcal{L}_m^{\mathcal{C}}(\mathbf{H}_s|\mathbf{H}_s) \\ &= \mathcal{L}_m^{\mathcal{C}}(f_t(\mathcal{X}_t^l)|f_t(\mathcal{X}_t^l)) + \mathcal{L}_m^{\mathcal{C}}(f_s(\mathcal{X}_s^l)|f_s(\mathcal{X}_s^l)) \end{aligned} \tag{5}$$

$$\begin{aligned} \mathcal{L}_{inter} &= \mathcal{L}_m^{\mathcal{C}}(\mathbf{H}_{t\text{-}s}|\mathbf{H}_s) + \mathcal{L}_m^{\mathcal{C}}(\mathbf{H}_s|\mathbf{H}_{t\text{-}s}) \\ &= \mathcal{L}_m^{\mathcal{C}}(f(f_t(\mathcal{X}_t^l))|f_s(\mathcal{X}_s^l) + \mathcal{L}_m^{\mathcal{C}}(f_s(\mathcal{X}_s^l)|f(f_t(\mathcal{X}_t^l)) \end{aligned} \tag{6}$$

$$\mathcal{L}_{CA} = \mathcal{L}_{intra} + \mathcal{L}_{inter} \tag{7}$$

Minimizing $\mathcal{L}_{intra}$ constructs representation spaces $\mathbf{H}_t$ and $\mathbf{H}_s$ that less violate the cluster assumption in each domain. Due to the *1-Lipschitz* constraint, $\mathbf{H}_{t\text{-}s}$ can preserve the local cluster topology of $\mathbf{H}_t$. Moreover, minimizing $\mathcal{L}_{inter}$ aligns the clusters in $\mathbf{H}_s$ with those in $\mathbf{H}_{t\text{-}s}$ associated with the same labels as in $\mathbf{H}_s$, i.e., inter-domain alignment. As a result, $\mathbf{H}_{t\text{-}s}$ and $\mathbf{H}_s$ are compatible spaces. Note that, we fix the parameters φ_d of f_d in Eqn. 1 when we optimize consensus alignment w.r.t. φ_c of f_c.

Adversarial Distribution Alignment Traditional semi-supervised DA methods only consider labeled data in the source domain. Due to the cluster assumption, the unlabeled data also imply class structures. To exploit unlabeled data (including labeled data without involving their labels), a straightforward way is to align the distributions between target and source. However, arbitrarily aligning distribution through unlabeled data may lead to a wrong alignment between classes. For example, if the conditional target distribution $\mathcal{D}_t(\mathcal{X}_{c_1}|c_1)$ of class c_1 is closer to the conditional source distribution $\mathcal{D}_s(\mathcal{X}_{c_2}|c_2)$ of a heterogeneous class c_2 than to the homogeneous class $c_)$, it tends to be aligned falsely.

Note that common representation transformation function f consists of two transformation functions, f_c and f_d (cf. Eqn. 1), the distribution alignment aims to find an optimal f_d by optimizing parameter φ_d with fixing φ_c of f_c. To find the optimal transformation function, f_d, we minimize the following Wasserstein distance with adversarial learning, where f_c is a *1-Lipschitz* function implemented by SNMLP (Miyato et al. 2018) to serve as the critic:

$$\mathcal{L}_{DA} = \sup_{f_c} \mathbb{E}_{\mathbf{h}_s \in \mathbf{H}_s}[f_c(\mathbf{h}_s;\eta)] - \mathbb{E}_{\mathbf{h}_t \in \mathbf{H}_t}[f_c(\mathbf{h}_{t\text{-}s};\eta)] \tag{8}$$

The distribution alignment and the class alignment alternately optimize the common representation space based on each other's results. Given the common representation space $\mathbf{H}_{t\text{-}s}$ optimized by the class alignment, the distribution alignment searches a $\tilde{\mathbf{h}}_{t\text{-}s}$ in the vicinity of $\mathbf{h}_{t\text{-}s} \in \mathbf{H}_{t\text{-}s}$ within the radius $\gamma\|\mathbf{h}_{t\text{-}s}\|$ (note that γ is the hyper-parameter in Eqn. 1 and f_d is spectral normalized with a unity spectral radius). Since $\mathbf{H}_{t\text{-}s}$ has been optimized under cluster assumption through the MMMC loss (cf. Eqn. 7) with clear margins, the distribution alignment can set a small γ (we set $\gamma = 0.02$ in this paper through empirical test) to constrain the distribution alignment inside a cluster where the target data and source data are associated with the same major labels. Obviously, such cluster-specific distribution alignment avoids the aforementioned inconsistent class alignment issue.

Complementary Representation Learning

The complementary representation learning aims to find a transformation function, i.e., $f_p(\mathbf{h}_{t\text{-}s};\phi) : \mathbf{H}_{t\text{-}s} \mapsto \mathbf{H}_{t\text{-}s\text{-}t}$, which integrates the knowledge from common representation space $\mathbf{H}_{t\text{-}s}$ into $\mathbf{H}_{t\text{-}s\text{-}t}$ without losing original knowledge in the target domain.

Self Alignment To preserve the cluster topology in $\mathbf{H}_{t\text{-}s}$, we also equip f_p with SNMLPs to be a *1-Lipschitz* function. Each $\mathbf{h}_{t\text{-}s\text{-}t} \in \mathbf{H}_{t\text{-}s\text{-}t}$ should be able to retrieve the corresponding $\mathbf{h}_t \in \mathbf{H}_t$ that represents the identical target object. To enable this self alignment, we again apply the MMCS loss where the class labels $\mathcal{C}'$ are the object IDs.

$$\begin{aligned} \mathcal{L}_{SA} &= \mathcal{L}_{m_\varepsilon}^{\mathcal{C}'}(\mathbf{H}_{t\text{-}s\text{-}t}|\mathbf{H}_t) \\ &= \mathcal{L}_{m_\varepsilon}^{\mathcal{C}'}(f_p(f(f_t(\mathcal{X}_t)))|f_t(\mathcal{X}_t) \end{aligned} \tag{9}$$

Minimizing this loss forces the two representation vectors, i.e., $\mathbf{h}_t \in \mathbf{H}_t$ and $\mathbf{h}_{t\text{-}s\text{-}t} \in \mathbf{H}_{t\text{-}s\text{-}t}$, of the identical object to be closer than those of any other objects. We set a small margin $m_\varepsilon = 1e^{-3}$ in Eqn. 9 to avoid overfitting to the target representation space $\mathbf{H}_t$.

Analysis

In this section, we analysis the error bound of classification on common representation space $\mathbf{H}_{t\text{-}s}$ according to the domain adaptation theory (Ben-David and Urner 2014) and the diameter relationship between the projected target representation space, common representation space and complementary representation space.

Let $h : \mathbf{H} \mapsto \mathbb{R}$ be a hypothesis in class $\mathcal{H}$, and $h_{t\text{-}s} = h \circ f \circ f_t : \mathbf{H}_{t\text{-}s} \mapsto \mathbb{R}$ and $h_s = h \circ f_s : \mathbf{H}_s \mapsto \mathbb{R}$ be the compositions, then we have the following error bound on $\mathbf{H}_{t\text{-}s}$:

$$\epsilon_t(h_{t\text{-}s}) \leq \epsilon_s(h_s) + d_{\mathcal{H}\triangle\mathcal{H}}(\mathcal{D}_{\mathbf{H}_s}, \mathcal{D}_{\mathbf{H}_{t\text{-}s}}) + \epsilon_s(h_s^*) + \epsilon_t(h_{t\text{-}s}^*)$$

where $\mathcal{D}_{\mathbf{H}_s}$ and $\mathcal{D}_{\mathbf{H}_{t\text{-}s}}$ denote the corresponding distributions of $\mathbf{H}_s$ and $\mathbf{H}_{t\text{-}s}$ respectively. $d_{\mathcal{H}\triangle\mathcal{H}}(\mathcal{D}_{\mathbf{H}_s}, \mathcal{D}_{\mathbf{H}_{t\text{-}s}})$ denotes the $\mathcal{H} \triangle \mathcal{H}$ distance and we omit the constant factor 2 in the original paper (Ben-David et al. 2010):

$$d_{\mathcal{H}\triangle\mathcal{H}}(\mathcal{D}_{\mathbf{H}_s}, \mathcal{D}_{\mathbf{H}_{t\text{-}s}}) = \sup_{h,h' \in \mathcal{H}} |\epsilon_t(h,h') - \epsilon_s(h,h')|$$

h^* denotes the ideal joint hypothesis which minimizes the combined error:

$$h^* = \arg\min_h [\epsilon_s(h \circ f_s) + \epsilon_t(h \circ f \circ f_t)]$$

Considering the triangle inequality for classification error (Ben-David et al. 2010), for any labeling functions h_1, h_2, h_3, we have $\epsilon(h_1, h_2) \leq \epsilon(h_1, h_3) + \epsilon(h_2, h_3)$. Then

$$
\begin{aligned}
&\epsilon_t(h_{t\text{-}s}) \\
&\leq \epsilon_t(h_{t\text{-}s}, h^*_{t\text{-}s}) + \epsilon_t(h^*_{t\text{-}s}) \\
&= \epsilon_t(h^*_{t\text{-}s}) + \epsilon_s(h_s, h^*_s) + \epsilon_t(h_{t\text{-}s}, h^*_{t\text{-}s}) - \epsilon_s(h_s, h^*_s) \\
&\leq \epsilon_t(h^*_{t\text{-}s}) + \epsilon_s(h_s, h^*_s) + |\epsilon_t(h_{t\text{-}s}, h^*_{t\text{-}s}) - \epsilon_s(h_s, h^*_s)| \\
&\leq \epsilon_t(h^*_{t\text{-}s}) + \epsilon_s(h_s) + \epsilon_s(h^*_s) \\
&\quad + |\epsilon_t(h_{t\text{-}s}, h^*_{t\text{-}s}) - \epsilon_s(h_s, h^*_s)| \\
&= \epsilon_t(h^*_{t\text{-}s}) + \epsilon_s(h_s) + \epsilon_s(h^*_s) \\
&\quad + |\mathbb{E}_{\mathbf{h}\sim f(f_t(\mathcal{X}_t))}[|h(\mathbf{h}) - h^*(\mathbf{h})|] \\
&\qquad - \mathbb{E}_{\mathbf{h}\sim f_s(\mathcal{X}_s)}[|h(\mathbf{h}), h^*(\mathbf{h})|])| \\
&\leq \epsilon_s(h_s) + d_{\mathcal{H}\triangle\mathcal{H}}(\mathcal{D}_{\mathbf{H}_s}, \mathcal{D}_{\mathbf{H}_{t\text{-}s}}) + \epsilon_s(h^*_s) + \epsilon_t(h^*_{t\text{-}s})
\end{aligned}
$$

Optimizing the adversarial distribution alignment (Eqn. 8) can minimize the $d_{\mathcal{H}\triangle\mathcal{H}}(\mathcal{D}_{\mathbf{H}_s}, \mathcal{D}_{\mathbf{H}_{t\text{-}s}})$ according to the Theorem 1 in (Shen et al. 2018). Further, our method utilizes the labeled data in the source domain to minimize the MMCS loss $\mathcal{L}^{\mathcal{C}}_m(\mathbf{H}_s|\mathbf{H}_s)$ (Eqn. 5) for low-density separations which make the classification boundary clearer in $\mathbf{H}_s$ and may lead to lower source error $\epsilon_s(h^*_s)$ given the optimal labeling function h^*. Also the clearer decision boundary and more compact cluster structure in $\mathbf{H}_{t\text{-}s}$ could be achieved according to the diameter relationship between subsets in $\mathbf{H}_t$, $\mathbf{H}_{t\text{-}s}$ and $\mathbf{H}_{t\text{-}s\text{-}t}$. The definition of diameter is as follows:

Definition 2. *(Diameter of a subset). The diameter d of a subset $\mathbf{S}$ of a metric space is the least upper bound of the set of all distances between pairs of points in the subset.*

$$D(\mathbf{S}) = \sup_{\mathbf{x},\mathbf{y}\in\mathbf{S}} \{\|\mathbf{x} - \mathbf{y}\|\} \tag{10}$$

Given any label $c \in \mathcal{C}$, we can obtain the subsets of the projected target representation: $\mathbf{G}^c_t \subset \mathbf{H}_t$, common representation space: $\mathbf{G}^c_{t\text{-}s} \subset \mathbf{H}_{t\text{-}s}$ and complementary representation space: $\mathbf{G}^c_{t\text{-}s\text{-}t} \subset \mathbf{H}_{t\text{-}s\text{-}t}$ associated with c. Recall that transformation functions f and f_p are 1-Lipschitz continuity. For all $\mathbf{h}^1_t, \mathbf{h}^2_t \in \mathbf{G}^c_t$, we have:

$$\|f(\mathbf{h}^1_t) - f(\mathbf{h}^2_t)\| \leq \|\mathbf{h}^1_t - \mathbf{h}^2_t\|$$

Similarly the f_p can keep the local distance relationship of $\mathbf{G}^c_{t\text{-}s}$ in $\mathbf{G}^c_{t\text{-}s\text{-}t}$. Then,

$$D(\mathbf{G}^c_{t\text{-}s\text{-}t}) \leq D(\mathbf{G}^c_{t\text{-}s}) \leq D(\mathbf{G}^c_t) \tag{11}$$

Considering the above diameter relationships and MMCS loss $\mathcal{L}^{\mathcal{C}}_m(\mathbf{H}_t|\mathbf{H}_t)$ (Eqn. 5), the intra-class cluster structure in $\mathbf{H}_{t\text{-}s}$ is more compact than that in $\mathbf{H}_t$ and inter-classes margin is also clearer. Therefore it is easier to find the decision boundary inside $\mathbf{H}_{t\text{-}s}$ and $\mathbf{H}_{t\text{-}s\text{-}t}$ than that inside $\mathbf{H}_t$, which benefits the classification and may lead to lower target error.

Algorithm and Parameter Learning Process

The overall loss of MULAN is:

$$\begin{cases} \mathcal{L}_{CASA} = \mathcal{L}_{CA} + \mathcal{L}_{SA} \\ \mathcal{L}_{DASA} = \mathcal{L}_{DA} + \mathcal{L}_{SA} \end{cases} \tag{12}$$

Algorithm 1 The Learning Process of MULAN

Let $\mathbf{\Omega} = \{\theta_t, \theta_s, \varphi_c, \phi\}$ and $\mathbf{\Phi} = \{\varphi_d, \eta\}$
for $iteration = 1$ to # max-iteration **do**
  Freezing the parameters in $\mathbf{\Phi}$
  for $i = 1$ to # training batches **do**
    Sample a minibatch $\mathcal{B}_t$ and $\mathcal{B}^-_t$ from $\mathcal{X}_t$
    Sample a minibatch $\mathcal{B}_s$ and $\mathcal{B}^-_s$ from $\mathcal{X}_s$
    $\mathbf{H}_t \leftarrow f_t(\mathcal{B}_t; \theta_t), \mathbf{H}_s \leftarrow f_s(\mathcal{B}_s; \theta_s)$
    $\mathbf{H}^-_t \leftarrow f_t(\mathcal{B}^-_t; \theta_t), \mathbf{H}^-_s \leftarrow f_s(\mathcal{B}^-_s; \theta_s)$
    $\mathbf{H}_{t\text{-}s} \leftarrow f(\mathbf{H}_t; \varphi_c, \varphi_d), \mathbf{H}_{t\text{-}s\text{-}t} \leftarrow f_p(\mathbf{H}_{t\text{-}s}; \phi)$
    $\mathcal{L}_{inter} \leftarrow \mathcal{L}^{\mathcal{C}}_m(\mathbf{H}_{t\text{-}s}|\mathbf{H}_s) + \mathcal{L}^{\mathcal{C}}_m(\mathbf{H}_s|\mathbf{H}_{t\text{-}s})$
    $\mathcal{L}_{intra} \leftarrow \mathcal{L}^{\mathcal{C}}_m(\mathbf{H}_t|\mathbf{H}^-_t) + \mathcal{L}^{\mathcal{C}}_m(\mathbf{H}_s|\mathbf{H}^-_s)$
    $\mathcal{L}_{CA} \leftarrow \mathcal{L}_{intra} + \mathcal{L}_{inter}$
    $\mathcal{L}_{SA} \leftarrow \mathcal{L}^{\mathcal{C}}_m(\mathbf{H}_{t\text{-}s\text{-}t}|\mathbf{H}_t)$
    $\mathcal{L}_{CASA} \leftarrow \mathcal{L}_{CA} + \mathcal{L}_{SA}$
    $\mathbf{\Omega} \leftarrow \mathbf{\Omega} - Adam[\nabla_{\mathbf{\Omega}}\mathcal{L}_{CASA}]$
  end for
  Freezing the parameters in $\mathbf{\Omega}$
  for $j = 1$ to # training batches **do**
    for $k = 1$ to # critic sub-iteration **do**
      Sample a minibatch $\mathcal{B}_t$ and $\mathcal{B}_s$
      $\mathbf{H}_{t\text{-}s} \leftarrow f(f_t(\mathcal{B}_t; \theta_t); \varphi_c, \varphi_d), \mathbf{H}_s \leftarrow f_s(\mathcal{B}_s; \theta_s)$
      $\mathcal{L} \leftarrow \sum_{\mathbf{h}_{t\text{-}s}\in\mathbf{H}_{t\text{-}s}} f_c(\mathbf{h}_{t\text{-}s}) - \sum_{\mathbf{h}_s\in\mathbf{H}_s} f_c(\mathbf{h}_s; \eta)$
      $\eta \leftarrow \eta - Adam[\nabla_\eta \mathcal{L}]$
    end for
    Sample a minibatch $\mathcal{B}_t$
    $\mathbf{H}_t \leftarrow f_t(\mathcal{B}_t; \theta_t), \mathbf{H}_{t\text{-}s} \leftarrow f(\mathbf{H}_t; \varphi_c, \varphi_d)$
    $\mathcal{L}_{DA} \leftarrow -\sum_{\mathbf{h}_{t\text{-}s}\in\mathbf{H}_{t\text{-}s}} f_c(\mathbf{h}_{t\text{-}s}; \eta)$
    $\mathbf{H}_{t\text{-}s\text{-}t} \leftarrow f_p(\mathbf{H}_{t\text{-}s}; \phi)$
    $\mathcal{L}_{SA} \leftarrow \mathcal{L}^{\mathcal{C}}_m(\mathbf{H}_{t\text{-}s\text{-}t}|\mathbf{H}_t)$
    $\mathcal{L}_{DASA} \leftarrow \mathcal{L}_{DA} + \mathcal{L}_{SA}$
    $\varphi_d \leftarrow \varphi_d - Adam[\nabla_{\varphi_d}\mathcal{L}_{DASA}]$
  end for
end for
The gradient-based optimization is based on Adam (Kingma and Ba 2014).

These two losses are iteratively learned and the corresponding parameters are updated as shown in Algorithm 1. The learning process consists of two major steps: (1) refining cluster assumption by fixing the parameters in $\mathcal{L}_{DA}$; and (2) aligning domain distributions by fixing the parameters in $\mathcal{L}_{CA}$ and $\mathcal{L}_{SA}$. Note that the MMCS loss (Eqn. 3) is one of the most critical components for MULAN, the empirical estimate of the MMCS loss can be effectively computed with matrix operations over minibatches.

Experiments

Experimental Setup

Datasets and Evaluation Traditional visual DA datasets, such as MNIST, USPS, and SVHN, have been reported that they are over-evaluated to achieve very high accuracy for almost all recent models (Tzeng et al. 2017; Motiian et al. 2017). Therefore, we adopt the latest VisDA Challenge dataset (Peng et al. 2017) in our experiments, which supports object classification of synthetic- and real-object images. To

Table 1: Semi-supervised DA. The classification accuracy (mean $_{\pm std}\%$) of *Synthetic→Real* domain adaptation with 5-fold validation on the VisDA Challenge dataset.

Labeled	Source-only	ADA	DANN	DIRT-T	TADT	CCSA	MULAN$_{Com}$	MULAN
10%	$45.01_{\pm 1.0}$	$47.02_{\pm 1.9}$	$44.87_{\pm 0.8}$	$57.08_{\pm 0.3}$	$69.68_{\pm 0.4}$	$67.99_{\pm 0.3}$	$\mathbf{77.44}_{\pm 0.2}$	$\mathbf{78.11}_{\pm 0.1}$
20%	$42.45_{\pm 0.7}$	$47.90_{\pm 0.9}$	$42.54_{\pm 0.3}$	$59.61_{\pm 0.2}$	$72.97_{\pm 0.2}$	$69.65_{\pm 0.2}$	$\mathbf{78.48}_{\pm 0.2}$	$\mathbf{79.15}_{\pm 0.2}$
50%	$38.82_{\pm 0.5}$	$48.59_{\pm 0.3}$	$41.01_{\pm 0.3}$	$59.88_{\pm 0.2}$	$76.37_{\pm 0.2}$	$71.33_{\pm 0.1}$	$\mathbf{79.76}_{\pm 0.1}$	$\mathbf{80.64}_{\pm 0.1}$

date, this dataset is the largest for cross-domain object classification, with over 280K images across 12 categories. In the synthetic image domain, images were generated by rendering 3D models of the same object categories as in the real data from different angles and under different lighting conditions. This domain contains 152,397 synthetic images. In the real image domain, 55,388 images were collected from MSCOCO (Lin et al. 2014).

Comparison Methods and Experimental Settings We compare our method with the baseline Source-only (i.e., classifier trained only on source domain without adaptation), and the state-of-the-art DA methods: DANN (Ganin et al. 2016), ADA (Haeusser et al. 2017), CCSA (Motiian et al. 2017), TADT (Tzeng et al. 2015), and DIRT-T (Shu et al. 2018). We also construct MULAN$_{Com}$ which only contains the common representation learning without complementary representation learning for the ablation study. The configuration for each model is used as default in original paper. We evaluate MULAN with semi-supervised DA with partially-labeled data in both source and target domains which are more consistent with real data. All the image features, i.e., $\mathcal{X}_t$ and $\mathcal{X}_s$ in these methods are represented by ResNet50 features (He et al. 2016) that are pre-trained on ImageNet.

Semi-supervised DA

Performance Comparison Table 1 shows the classification accuracy of all comparison methods on semi-supervised DA w.r.t. different percentages of labeled data in both source (synthetic images) and target (real images) domains. After the training of MULAN, we train a classifier on the source domain representation $\mathbf{H}_s$ and use the common representation $\mathbf{H}_{t\text{-}s}$ to predict labels for the target domain. Compared with the state-of-the-art methods, MULAN consistently achieves the best performance with all different proportions of labeled data. With the increase of labeled source data, the classifier trained on source domain without adaptation fits better to the source domain while it leads to the poorer performance on the target domain due to significant domain shift. Benefiting from the domain consensus information, semi-supervised DA methods, i.e., CCSA, TADT, and MULAN, outperform unsupervised DA methods, i.e., ADA, DANN, DIRT-T, especially with the increase of labeled data. Moreover, MULAN outperforms the state-of-the-art methods up to 12.1% when 10% labeled data is available, which indicates that MULAN makes better use of labeled data in both domains through the class alignment and distribution alignment without overfitting to the source domain.

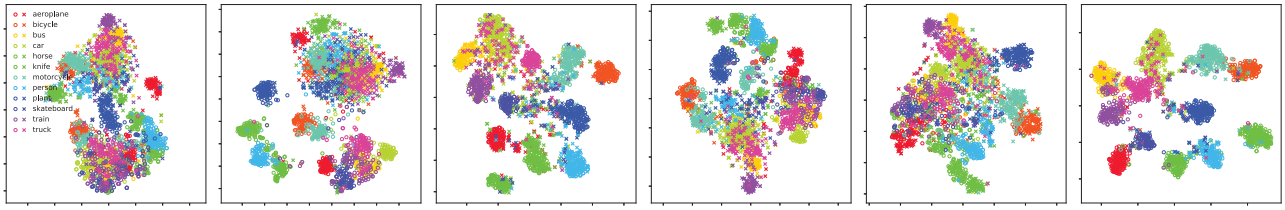
Ablation Study In Table 1, MULAN$_{Com}$ denotes the MULAN model without the complementary representation learning and the loss becomes $\mathcal{L} = \mathcal{L}_{CA} + \mathcal{L}_{DA}$. MULAN$_{Com}$ still outperforms the other comparison methods, which proves that the common representation learning with class alignment and distribution alignment are powerful component in cross-domain learning. Moreover, full MULAN achieves better performance than MULAN$_{Com}$ because the self alignment $\mathcal{L}_{SA}$ helps target domain to keep its own information to avoid overfitting to the source domain. This self alignment not only directly affects the generation of complementary representation but also benefits the learning of common representation.

Visualization Figure 2 visualizes the embedding space of 1,200 randomly sampled source domain instances and 1,200 target domain instances from the methods: Original (ResNet features without adaptation), ADA, DIRT-T, TADT, CCSA, and SPR from MULAN, through t-SNE. The original manifold w.r.t. the source and target features show clear domain shift. In Figure 2g, almost all classes are clearly separated, and the source embedding and the target embedding are well aligned together which reflects that the common representation learned by MULAN well satisfies the cluster assumption. In comparison, all the other methods either much less clearly separate different classes or even fail to adapt source to target. Figure 3 demonstrates the confusion matrices which reflect the distance between classes of the target domain and source domain in the shared or common representation space. And the diagonal blocks in the confusion matrix reflect the distance between the centroid of a target class and the centroid of the corresponding source class. The darker the block color, the smaller the distance is. According to Figure 3f, the source and target data are well aligned in the common representation space learned by MULAN.

Few-shot DA

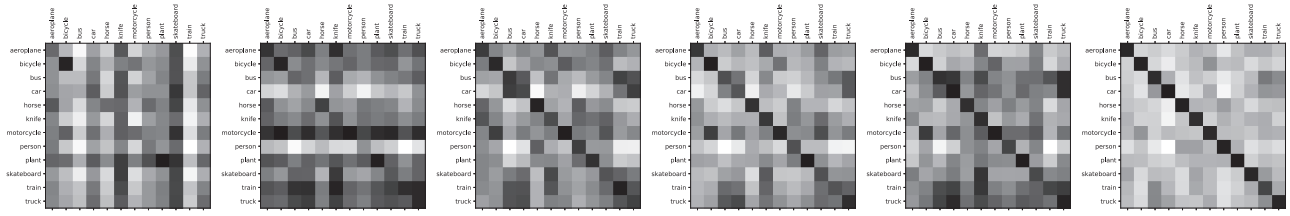
Performance Comparison Table 2 shows the classification performance on few-shot DA where the source domain contains 50% labeled real images and target domain contains few synthetic image instances. In few-shot DA settings, we test two types of classifiers: one is trained with the source domain labeled data and the other is trained with the target domain labeled data. Among all the source classifiers, the classifier using the common representation generated by MULAN achieves the best performance. This is because the labeled real images are limited, these supervised DA models, i.e., CCSA and TADT, cannot learn the domain shift well. Especially, when only 5 labeled instances in each class are available, the common representation learned by MULAN shows its superior performance because MULAN bet-

(a) Data sample from Synthetic and Real domain with legend



(b) Original (c) ADA (d) DIRT-T (e) TADT (f) CCSA (g) MULAN

Figure 2: The t-SNE visualization of shared embedding space of Original representation (ResNet features), ADA, DIRT-T, TADT, CCSA, and MULAN. Note: "o" denotes source synthetic image and "x" denotes target real image as shown in (a).



(a) Original (b) ADA (c) DIRT-T (d) TADT (e) CCSA (f) MULAN

Figure 3: The confusion matrix of source and target domains in the shared embedding space of Original representation (ResNet features), ADA, DIRT-T, TADT, CCSA, and MULAN.

Table 2: Few-shot DA. The classification accuracy of *Real→Synthetic* adaptation.

	Model	20-shot	10-shot	5-shot
Source Classifier	Source only	39.04	39.01	39.51
	ADA	41.31	42.63	40.97
	DANN	38.21	37.34	35.27
	DIRT-T	50.11	49.14	47.72
	TADT	62.07	55.31	52.72
	CCSA	57.34	50.77	49.65
	MULAN(Comm)	**64.40**	**58.17**	**56.02**
Target Classifier	Target-only	56.83	47.24	41.17
	MULAN(Comp)	**66.73**	**62.29**	**57.97**

ter aligns the target distribution to the source distribution with the distribution alignment constrained by cluster alignment through labeled data.

Comparison between Common and Complementary Representations In Table 2, we compare common representation (denoted by MULAN(Comm) in Table 2) and complementary representation (denoted by MULAN(Comp) in Table 2) which are both generated by MULAN, but they have different characteristics and are used in different scenarios. Common representation is compatible with the source domain so that it is evaluated by the source classifier while complementary representation is compatible with the target domain and is evaluated by the target classifier. The complementary representation not only inherits the source domain information from common representation but also emphasises the local information of the target domain. That is why complementary representation achieves the best performance on prediction and significantly outperforms the Target-only classifier when the target labeled data in few-shot DA is quite limited.

Conclusion and Future Work

In this paper, we propose a new cross-domain representation learning method, MULAN, which generates two representation spaces for partially-labeled cross-domain data. We apply MULAN to domain adaptation and demonstrate its superior performance.

There are several future extensions of MULAN. One is to extend MULAN to address multi-domain learning problem since more alignments can be added into the model. Another is to extend MULAN to multi-modal learning by customizing the original feature learning models.

Acknowledgement

This work is partially supported by the National Key Research and Development Program of China (2018YFB0803501), National High-level Personnel for Defense Technology Program (2017-JCJQ-ZQ-013), NSF 61902405 and HUNAN Province Science Foundation 2017RS3045.

References

Ben-David, S., and Urner, R. 2014. Domain adaptation–can quantity compensate for quality? *Annals of Mathematics and Artificial Intelligence* 70(3):185–202.

Ben-David, S.; Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; and Vaughan, J. W. 2010. A theory of learning from different domains. *Machine learning* 79(1-2):151–175.

Bousmalis, K.; Trigeorgis, G.; Silberman, N.; Krishnan, D.; and Erhan, D. 2016. Domain separation networks. In *Advances in neural information processing systems*, 343–351.

Chen, X.; Duan, Y.; Houthooft, R.; Schulman, J.; Sutskever, I.; and Abbeel, P. 2016. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, 2172–2180.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research* 17(1):2096–2030.

Gong, B.; Shi, Y.; Sha, F.; and Grauman, K. 2012. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*, 2066–2073. IEEE.

Gonzalez-Garcia, A.; van de Weijer, J.; and Bengio, Y. 2018. Image-to-image translation for cross-domain disentanglement. In *Advances in Neural Information Processing Systems*, 1294–1305.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *NIPS*, 2672–2680.

Haeusser, P.; Frerix, T.; Mordvintsev, A.; and Cremers, D. 2017. Associative domain adaptation. In *ICCV*, volume 2, 6.

Haeusser, P.; Mordvintsev, A.; and Cremers, D. 2017. Learning by association–a versatile semi-supervised training method for neural networks. In *CVPR*, 89–98.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; and Lerchner, A. 2017. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*.

Kim, T.; Cha, M.; Kim, H.; Lee, J. K.; and Kim, J. 2017. Learning to discover cross-domain relations with generative adversarial networks. *arXiv preprint arXiv:1703.05192*.

Kingma, D., and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P.; Mohamed, S.; Rezende, D. J.; and Welling, M. 2014. Semi-supervised learning with deep generative models. In *Advances in neural information processing systems*, 3581–3589.

Kulis, B.; Saenko, K.; and Darrell, T. 2011. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *CVPR*, 1785–1792. IEEE.

Lee, H.-Y.; Tseng, H.-Y.; Huang, J.-B.; Singh, M.; and Yang, M.-H. 2018. Diverse image-to-image translation via disentangled representations. In *ECCV*, 35–51.

Lin, T.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. 2014. Microsoft coco: Common objects in context. In *ECCV*, 740–755. Springer.

Liu, A. H.; Liu, Y.-C.; Yeh, Y.-Y.; and Wang, Y.-C. F. 2018a. A unified feature disentangler for multi-domain image translation and manipulation. In *Advances in Neural Information Processing Systems*, 2590–2599.

Liu, Y.-C.; Yeh, Y.-Y.; Fu, T.-C.; Wang, S.-D.; Chiu, W.-C.; and Frank Wang, Y.-C. 2018b. Detach and adapt: Learning cross-domain disentangled deep representation. In *CVPR*, 8867–8876.

Long, M.; Cao, Z.; Wang, J.; and Jordan, M. I. 2018. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, 1640–1650.

Miyato, T.; Kataoka, T.; Koyama, M.; and Yoshida, Y. 2018. Spectral normalization for generative adversarial networks. *ICLR*.

Motiian, S.; Piccirilli, M.; Adjeroh, D. A.; and Doretto, G. 2017. Unified deep supervised domain adaptation and generalization. In *ICCV*, volume 2, 3.

Odena, A.; Olah, C.; and Shlens, J. 2017. Conditional image synthesis with auxiliary classifier gans. In *ICML*, 2642–2651. JMLR. org.

Pan, S., and Yang, Q. 2010. A survey on transfer learning. *IEEE TKDE* 22(10):1345–1359.

Peng, X.; Usman, B.; Kaushik, N.; Hoffman, J.; Wang, D.; and Saenko, K. 2017. Visda: The visual domain adaptation challenge. *arXiv preprint arXiv:1710.06924*.

Rezende, D. J.; Mohamed, S.; and Wierstra, D. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, 1278–1286.

Rozantsev, A.; Salzmann, M.; and Fua, P. 2018. Beyond sharing weights for deep domain adaptation. *IEEE TPAMI*.

Shen, J.; Qu, Y.; Zhang, W.; and Yu, Y. 2018. Wasserstein distance guided representation learning for domain adaptation. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Shu, R.; Bui, H. H.; Narui, H.; and Ermon, S. 2018. A dirt-t approach to unsupervised domain adaptation. *arXiv preprint arXiv:1802.08735*.

Sun, B., and Saenko, K. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV*, 443–450. Springer.

Tzeng, E.; Hoffman, J.; Darrell, T.; and Saenko, K. 2015. Simultaneous deep transfer across domains and tasks. In *ICCV*, 4068–4076.

Tzeng, E.; Hoffman, J.; Saenko, K.; and Darrell, T. 2017. Adversarial discriminative domain adaptation. In *CVPR*, volume 1, 4.

Xie, S.; Zheng, Z.; Chen, L.; and Chen, C. 2018. Learning semantic representations for unsupervised domain adaptation. In *ICML*, 5419–5428.

Yi, Z.; Zhang, H. R.; Tan, P.; and Gong, M. 2017. Dualgan: Unsupervised dual learning for image-to-image translation. In *ICCV*, 2868–2876.