Deep Ladder Reconstruction-Classification Network for Unsupervised Domain Adaptation

Wanxia Deng[a], Zhuo Su[b], Qiang Qiu[c], Lingjun Zhao[a], Gangyao Kuang[a], Matti Pietikäinen[b], Huaxin Xiao[d], Li Liu[b,d,*]

[a]*State Key Laboratory of Complex Electromagnetic Environment Effects on Electronics and Information System, College of Electronic Science, National University of Defense technology, Changsha, Hunan 410003, China.*
[b]*University of Oulu, Oulu 90014, Finland*
[c]*Duke University, Durham 27708, USA*
[d]*College of Systems Engineering, National University of Defense technology, Changsha, Hunan 410003, China.*

ABSTRACT

Unsupervised Domain Adaptation aims to learn a classifier for an unlabeled target domain by transferring knowledge from a labeled source domain. Most existing approaches learn domain-invariant features by adapting the entire information of each image. However, forcing adaptation of domain-specific components can undermine the effectiveness of learned features. We propose a novel architecture called Deep Ladder Reconstruction-Classification Network (DLaReC) which is designed to learn cross-domain shared contents by suppressing domain-specific variations. The DLaReC adopts an encoder with cross-domain sharing and a target-domain reconstruction decoder. The encoder and decoder are connected with residual shortcuts at each intermediate layer. By this means, the domain-specific components are directly fed to the decoder for reconstruction, relieving the pressure to learn domain-specific variations at later layers of the shared encoder. Therefore, DLaReC allows the encoder to focus on learning cross-domain shared representations and ignore domain-specific variations. DLaReC is implemented by jointly learning three tasks: supervised classification of the source domain, unsupervised reconstruction of the target domain and cross-domain shared representation adaptation. Extensive experiments on Digit, Office31, ImageCLEF-DA and Office-Home datasets demonstrate the DLaReC outperforms state-of-the-art methods on the whole. The average accuracy on the Digit datasets, for instance, is improved from 95.6% to 96.9%. In addition, the result on Amazon → Webcam obtains significant improvement, *i.e.*, from 91.1% to 94.7%.

1. Introduction

In the past several years, Deep Neural Networks (DNNs) have brought tremendous progress for diverse computer vision tasks such as image classification (Krizhevsky et al., 2012), object detection (Liu et al., 2020), and image segmentation (Garcia-Garcia et al., 2018), showing good generalization ability. The remarkable success of DNNs heavily depends on massive labeled training data. However, labeling data is labor intensive and time consuming. In addition, there are many domains where only limited amounts of annotated training data can be available or collecting labeled training data is too expensive, and sometimes even impossible. To

*Corresponding author: Li Liu
e-mail: li.liu@oulu.fi (Li Liu)

address such problems, an alternative is to leverage labeled data and/or models from a similar domain (source domain) [1] to improve the model for the domain of interest (target domain). However, analogous to other machine learning techniques, DNNs also suffer from the problem of domain shift (Yosinski et al., 2014), (Donahue et al., 2014) *i.e.* predictors trained on a dataset suffer from performance degradation when applied to novel domains. To mitigate this degradation, Domain Adaptation (DA) have been proposed to address the problem of domain shift.

In this paper, we focus on the challenging problem of Unsupervised DA (UDA) that has been introduced by transferring knowledge from a labeled source domain to a fully unlabeled target domain with a related but different distribution. Recently, the combination of DNNs and UDA has achieved remarkable improvements (Wang and Deng, 2018), (Csurka, 2017).

Theoretical studies (Ben-David et al., 2007), (Ben-David et al., 2010) show that learning proper cross-domain features is an efficient path for addressing domain shift. Therefore, most UDA approaches are inspired by this. The metric discrepancy based methods (Long et al., 2015), (Sun and Saenko, 2016) are more popular UDA methods, which explicitly minimize the discrepancy between the feature distributions of two domains. However, despite their efficacy, these methods apply the entire information of domains to adapt. It is obvious that not all regions of an image are transferable. If the domain-specific components are forcefully aligned, the learned domain-invariant features may be vulnerable to negative impact of irrelevant knowledge.

In this paper, we tackle the aforementioned challenges in an improved metric discrepancy learning framework which effectively suppresses the domain-specific variations and learns the cross-domain shared content that are more discriminative for both domains. A novel yet elegant framework called Deep Ladder Reconstruction-Classification Network (DLaReC) is proposed by first introducing the ladder network into UDA, which is motivated by the recent success of the ladder network (Valpola, 2015), (Rasmus et al., 2015), (Pezeshki et al., 2016) for traditional semi-supervised learning and multitask learning for improving generalization. The ladder network is an autoencoder with skipping connections from the encoder to decoder, which is demonstrated that the skip connections can relieve the pressure to represent details in the higher layers of the model, because, through the skip connections, the decoder can recover any details discarded by the encoder Rasmus et al. (2015). Inspired by the insight, we introduce the ladder network for the UDA, which can guarantee to suppress the domain-specific variations and obtains the cross-domain shared representation.

The proposed DLaReC consists of an encoder with cross-domain sharing and a target domain reconstruction decoder. The encoder and decoder layers are connected by lateral residual shortcuts, looking like a ladder architecture. The lateral residual shortcuts are crucial which can alleviate the pressure to learn domain-specific variations at the lateral layers of the shared encoder, because the domain-specific components are directly fed to the reconstruction decoder. By this means, the shared encoder can focus on learning

[1] where large scale labeled training data (*e.g.* synthetic data) is more easily obtained and big enough for training large scale deep models.

cross-domain shared representations for the classification task.

The proposed DLaReC is implemented by jointly learning three tasks: supervised classification of the source domain, unsupervised reconstruction of the target domain and cross-domain shared representation adaptation. To train the three tasks, we introduce a two-stage iterative framework. In the first stage, we reconstruct the target domain using ladder autoencoder, because reconstruction of input samples from low dimensional latent representations at multi-layers of the DCNN is a general way of learning important representation in an unsupervised fashion. In the second stage, we only train the shared encoder and the classification layer, where supervised classification of the source domain and cross-domain shared representation adaptation are conducted simultaneously. We summarize our main contributions as follows:

- We propose a novel framework DLaReC for the problem of UDA. It first integrates the ladder network for UDA, which can learn expressive cross-domain shared representation while suppress domain-specific variations by lateral shortcuts. The lateral residual shortcuts are playing a vital role in DLaReC to the extent that removing them can deteriorate the adaptation performance.

- We introduce an iterative two-stage learning scheme for the supervised classification task to better collaborate with unsupervised reconstruction task. The reconstruction pipeline can function as a good regularizer to obtain the desired regularization using the unlabeled target domain.

2. Related Work

This section reviews mainstream approaches in UDA and focuses on some approaches that are related to our approach. The advantages and disadvantages of some representative algorithms are illustrated as shown in Fig. 1.

Metric Discrepancy based Methods One common approach for UDA is to guide the domain-invariant feature learning by minimizing the domain distribution discrepancy with the metric paradigm. Some representative metric methods include Maximum Mean Discrepancy (MMD), *e.g.*, DDC (Tzeng et al., 2014), DAN (Long et al., 2015), RTN (Long et al., 2016), JAN (Long et al., 2017), DTML (Hu et al., 2015, 2016) and correlation alignment (CORAL) (Sun and Saenko, 2016) which are embedded in the deep convolutional neural networks by adding the adaptation layer. The Central Moment Discrepancy (CMD) (Zellinger et al., 2017) is suggested to match the higher-order central moments of probability distributions. Domain Adaptation method based on Model Uncertainty (MUDA) (Lee and Lee, 2020) minimizes the model uncertainty loss using Monte Carlo dropout sampling to learn domain-invariant features.

Adversarial Learning based Methods Adversarial learning has been widely applied in the domain adaptation to deceive the domain discriminator. DANN (Ganin et al., 2016) and ADDA (Tzeng et al., 2017) add a subnetwork as the domain discriminator.

Methods	Representative Approaches	Main ideas	Strengths	Limitations
Metric Discrepancy based Methods	**DAN** (Long et al., 2015), **JAN** (Long et al., 2017), **D-CORAL** (Long et al., 2017), **CMD** (Zellinger et al., 2017)	They explicitly measure the discrepancy between the source and target domains on the corresponding activation layers of the two network streams.	This type of method directly minimizes differences of both domains, and the idea is more understanding and simple to implement.	They are forced to minimize the differences, when the discrepancy of two domains are large or complex, they perform poor.
Adversarial Learning based Methods	**DANN** (Ganin et al., 2016), **CDAN** (Long et al., 2018), **MCD** (Saito et al., 2018)	They implicitly minimize the domain discrepancy via training a domain critic along with a feature learning network in an adversarial manner.	They can deal with complex domain adaptation scenarios and learn domain invariant features globally.	They cannot guarantee that the two domains are well aligned even if the domain discriminator has been fully confused based on the equilibrium challenge of adversarial learning (Arora et al., 2017).
GAN based Methods	**CoGAN** (Liu and Tuzel, 2016), **CyCADA** (Hoffman et al., 2018), **PixelDA** (Bousmalis et al., 2017)	They explicitly measure the discrepancy between the source and target domains on the corresponding activation layers of the two network streams.	They explicitly measure the discrepancy between the source and target domains on the corresponding activation layers of the two network streams.	They explicitly measure the discrepancy between the source and target domains on the corresponding activation layers of the two network streams.
Reconstruction based Methods	**DRCN** (Ghifary et al., 2016)	It combines the supervised learning of the source domain with the unsupervised reconstruction of the target domain via the shared encoder.	DRCN can mine the intrinsic features of the target domain, and it is easy to operate.	The unsupervised reconstruction of the target domain tries to learn all the information of an image, while the supervised classification of the source domain only learns the task-specific information. The difference between them in capturing information can lead to poor domain-inavraint features.
	DSN (Bousmalis et al., 2016)	It separates the feature into the shared feature and the private feature. These two features are encouraged to be orthogonal while both the features can be decoded back to images.	The work is the first one to introduce the distanglement into deep domain adaptation.	DSN applies multiple architecture to separate the domain-shared features from the domain-specific features, which can lead difficult training and perform worse in face of complex datasets.

Fig. 1. Summary of some representative methods in UDA.

The extracted feature is learned to fool the domain discriminator using a domain-adversarial learning technique. DANN cannot guarantee that the two domains are well aligned even if the domain discriminator has been fully confused based on the equilibrium challenge of adversarial learning (Arora et al., 2017). CDAN (Long et al., 2018) encodes the prediction into deep features, then models the joint distributions of features and labels sharing the spirit of the conditional GANs. WDGRL (Shen et al., 2018) estimates empirical Wasserstein distance between the source and target samples in domain critic network. MCD (Saito et al., 2018) utilizes task-specific classifiers as discriminators, and aligns distributions of source and target domains by the adversarial learning of two task-specific classifiers. MADA (Pei et al., 2018) captures multi-mode structures to enable fine-grained alignment of different data distributions. Domain Adaptation based on Gaussian processes (GPDA) (Kim et al., 2019) maximizes margins and minimizes uncertainty of the class predictions in the target domain. On the basis of DANN, the Cluster Alignment with a Teacher (CAT) (Deng et al., 2019) further applies a deep clustering loss and leverages an implicit ensembling teacher model to uncover the class-conditional structure of domains.

GAN based Methods Recently, there are more popular domain adaptation approaches which incorporate generative modeling into the feature learning process using Generative Adversarial Networks (GAN). PixelDA (Bousmalis et al., 2017) and DTN (Taigman et al., 2016) learns to generate a new version of the source images with the style of target domain perform adaptation in the transferred space. Couple GAN (CoGAN) (Liu and Tuzel, 2016) applies a tuple of GANs, and each is responsible for synthesizing images in one domain. CyCADA (Hoffman et al., 2018), UNIT (Liu et al., 2017) and SBADA-GAN (Russo et al., 2018) constrain

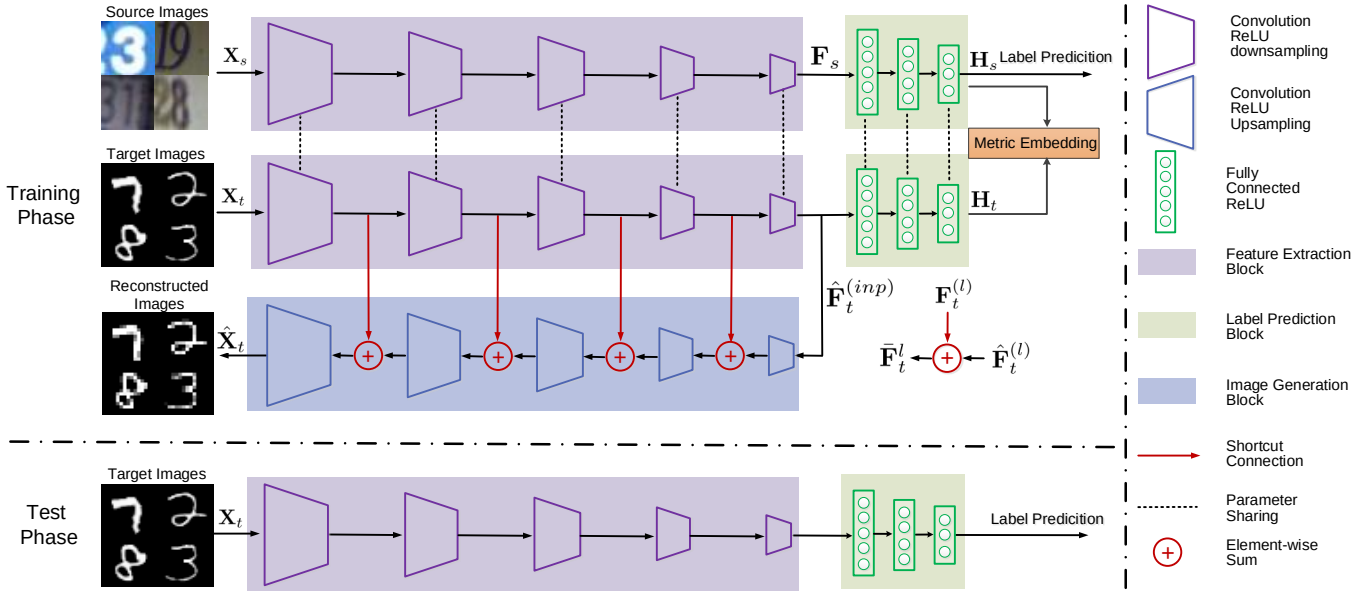


Fig. 2. Illustration of the proposed approach. The training phase consists of two parallel pipelines: (1) classification branch where feature extraction block and label predictor block are updated with the classification loss of the source domain and the discrepancy loss of cross-domain shared features. (2) data reconstruction branch which consists of a ladder autoencoder. The feature extraction block and image generation block are updated using target images reconstruction loss. The red solid lines denote the residual shortcuts. The cross-domain shared features are propagated through the shared encoder for classification, and the domain-specific components are merged into the decoder by lateral shortcuts for reconstruction. In the test phase, the image generation block is removed, and the classification prediction is produced with the feature extraction block and the label predictor block.

the mapping by imposing cycle consistency: the mapping in one direction (source-to-target or target-to-source) should get back where it started. GenToAdapt (Sankaranarayanan et al., 2018) proposes an adversarial image generation approach to learn the feature embedding using a combination of generated source-like images classification loss and an image generation procedure.

Reconstruction based Methods The domain adaptation approaches based on autoencoder reconstruction typically learn the domain-invariant features with supervised classification and unsupervised reconstruction. DRCN (Ghifary et al., 2016) is proposed to jointly learn common encoding representation combining the supervised classification of source domain and unsupervised reconstruction of target domain. The unsupervised reconstruction of the target domain tries to learn all the information of an image, while the supervised classification of the source domain only learns the task-specific information. The contradiction between them in capturing information can lead to poor domain-inavraint features. DSN (Bousmalis et al., 2016) proposes to separate the shared feature from the private feature. The two types of features are encouraged to be orthogonal while both of them can be decoded back to images. DSN applies multiple architecture to separate the domain-shared features from the domain-specific features, which can lead difficult training and perform worse in face of complex datasets. On basis of the DSN, (Liu et al., 2019) proposes that feature orthogonal regularization is applied between private across domains, in addition to private features and shared features in each domain. Our proposed algorithm mainly belongs to this type of method, and it inherits the main motivation of DSN and circumvents the drawbacks of DRCN.

3. Proposed Approach

The overall architecture of our proposed DLaReC is illustrated in Fig. 2. As shown in Fig. 2, we intend to jointly learn three tasks: supervised classification of source domain, adaptation of cross-domain shared representations via metric embedding and unsupervised reconstruction of target domain with a ladder autoencoder. The Ladder network, *i.e.* connecting the shared encoder and the decoder layers with lateral residual shortcuts (the red connections shown in Fig. 2) alleviates the pressure to learn details at the later layers of the shared encoder. The training of DLaReC is firstly performed by utilizing unlabeled target data to reconstruct via the ladder autoencoder, which enables the shared encoder to better focus on learning the cross-domain shared representation. The labeled source data is then utilized to learn discriminative transferable features with the classifier. At the same time, the discrepancy of shared features for target and source domains are reduced by means of metric embedding. We alternate the above two steps in each training iteration and optimize the whole DLaReC end to end. We present the proposed approach in detail below.

3.1. Problem Formulation

In the problem of UDA, we define a source domain probability distribution as P_s on $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ is the input space and $\mathcal{Y}$ is the output label space. Similarly, the target domain distribution is defined as P_t, where $P_s \neq P_t$. The source domain dataset $D_s = \left\{(\mathbf{x}_i^s, y_i^s)\right\}_{i=1}^{N_s}$ of N_s labeled examples is sampled from the distribution P_s. The target domain dataset $D_t = \left\{(\mathbf{x}_i^t)\right\}_{i=1}^{N_t}$ of N_t unlabeled examples is sampled from P_t. Our goal is to learn a transformation function which maps the $\mathcal{X}$ space to the $\mathcal{F}$ space. In $\mathcal{F}$, the distributions of the two domains are minimized to obtain the domain-invariant feature so that the target domain images can be classified into $\mathcal{Y}$.

3.2. Ladder Reconstruction Network

The proposed DLaReC is based on the assumption that only adapting the cross-domain shared features can obtain better domain-invariant features. There are two main branches: one branch is utilized for supervised classification and cross-domain shared representations adaptation, and the other is utilized for reconstruction of target domain images. These two branches share the encoder network. We add a lateral shortcut from the shared encoder to the decoder at each intermediate layer to encourage the shared encoder focusing on cross-domain shared features and suppressing domain-specific variations for classification. As shown in Fig. 2, in order to learn effective domain-invariant features, we design a ladder network for unsupervised reconstruction of the target images where the encoder and decoder layers are connected with residual shortcuts. The lateral shortcuts are deliberately introduced in order to feed the domain-specific details at multi-levels directly into the decoder layers to facilitate the reconstruction goal. Therefore, the ladder architecture makes the shared encoder easily focus on learning cross-domain shared features and suppress domain-specific variations.

We define the shared encoder as G_{enc} with parameter Θ_{enc}, the decoder as G_{dec} with parameter Θ_{dec}, and the classification block as

G_{cls} with parameter Θ_{cls}. For simplicity, we denote $\mathbf{X}_s$ and $\mathbf{X}_t$ as source and target examples respectively. The label of the source example is $\mathbf{y}_s$. For the target domain, the forward propagation of each layer is described more formally as follows:

$$\mathbf{F}_t^{(l)} = G_{enc}^{(l)}(\mathbf{X}_t), \quad for \quad l = 1 \tag{1}$$

$$\mathbf{F}_t^{(l)} = G_{enc}^{(l)}(\mathbf{F}_t^{(l-1)}), \quad for \quad 2 \leqslant l \leqslant L \tag{2}$$

$$\begin{cases} \bar{\mathbf{F}}_t^{(l)} = Combinator(\mathbf{F}_t^{(l)}, \hat{\mathbf{F}}_t^{(l+1)}) \\ \hat{\mathbf{F}}_t^{(l)} = G_{dec}^{(l)}(\bar{\mathbf{F}}_t^{l}) \end{cases}, \quad for \quad 1 \leqslant l < L \tag{3}$$

$$\begin{cases} \hat{\mathbf{F}}_t^{(inp)} = \mathbf{F}_t^{(l)} \\ \hat{\mathbf{F}}_t^{(l)} = G_{dec}^{(l)}(\hat{\mathbf{F}}_t^{(inp)}) \end{cases}, \quad for \quad l = L \tag{4}$$

$$\hat{\mathbf{X}}_t = \hat{\mathbf{F}}_t^{(l)}, \quad for \quad l = 1 \tag{5}$$

where L represents the total number of layers of the encoder or decoder. The (l) refers to variables on encoder or decoder layer l. $G_{enc}^{(l)}$ and $G_{dec}^{(l)}$ denote the feature transformation on the $l-th$ layer of encoder and decoder, respectively. $G_{enc}^{(l)}$ consists of the convolutional, ReLu, or downsampling layers. $G_{dec}^{(l)}$ consists of the convolutional, ReLu, or upsampling layers. $\mathbf{F}_t^{(l)}$ indicates the $l-th$ layer feature of the target domain in the encoder. $\hat{\mathbf{F}}_t^{(l)}$ indicates the $l-th$ layer reconstruction feature of the target domain in the decoder. $\hat{\mathbf{F}}_t^{(inp)}$ means the input of the decoder. $\hat{\mathbf{X}}_t$ is the output of the decoder, *i.e.*, it is the reconstruction of $\mathbf{X}_t$. $\bar{\mathbf{F}}_t^{(l)}$ is the representation of the combination of $\mathbf{F}_t^{(l)}$ and $\hat{\mathbf{F}}_t^{(l+1)}$, where the combinator is defined as follows:

$$Combinator(\mathbf{F}_t^{(l)}, \hat{\mathbf{F}}_t^{(l+1)}) = \frac{1}{2}[\hat{\mathbf{F}}_t^{(l+1)} + (\mathbf{F}_t^{(l)})] \tag{6}$$

The simple combination is a residual connection, which plays an important role in focusing on cross-domain shared features and suppressing domain-specific variations.

3.3. Classification and Cross-Domain Shared Feature Adaptation

The source domain classification is carried out via the shared encoder and classification layer. Meanwhile, the adaptation of cross-domain shared features is conducted via embedding the discrepancy metric at the last layer of the classification block G_{cls}. The forward propagation is defined as follows:

$$\begin{cases} \mathbf{F}_s = G_{enc}(\mathbf{X}_s) \\ \mathbf{H}_s = G_{cls}(\mathbf{F}_s) \end{cases} \tag{7}$$

$$\tilde{\mathbf{y}}_s = Softmax(\mathbf{H}_s) \tag{8}$$

$$\begin{cases} \mathbf{F}_t = G_{enc}(\mathbf{X}_t) \\ \mathbf{H}_t = G_{cls}(\mathbf{F}_t) \end{cases} \tag{9}$$

Here, $\mathbf{F}_s$ and $\mathbf{F}_t$ are the representations of source and target domains in the encoder's final layer, respectively. $\mathbf{H}_s$ and $\mathbf{H}_t$ are the representations of source and target domains in the network's final hidden layer, respectively. The $\tilde{\mathbf{y}}_s$ denotes the network's prediction of $\mathbf{X}_s$.

There are multiple choices for distance function to measure the feature discrepancy, such as MMD (Borgwardt et al., 2006) computing the norm of difference between two domain means and CORAL (Sun et al., 2016) computing the distance of two domains. We explore the two different measurements in the experiment to verify the compatibility and generalization of the proposed DLaReC. Specifically, CORAL is to align the second-order statistics (covariances) of the source and target features:

$$d_{coral} = \frac{1}{4d^2} \left\| cov(\mathbf{H}_s) - cov(\mathbf{H}_t) \right\|_F^2 \tag{10}$$

where d is the feature dimension of $\mathbf{H}_s$ and $\mathbf{H}_t$. $\|\cdot\|_F^2$ denotes the squared matrix Frobenius norm. The covariance matrices of the source and target domain features are given by:

$$cov(\mathbf{H}_s) = \frac{1}{n_s - 1}(\mathbf{H}_s^T \mathbf{H}_s - \frac{1}{n_s}(\mathbf{1}^T \mathbf{H}_s)^T (\mathbf{1}^T \mathbf{H}_s)) \tag{11}$$

$$cov(\mathbf{H}_t) = \frac{1}{n_t - 1}(\mathbf{H}_t^T \mathbf{H}_t - \frac{1}{n_t}(\mathbf{1}^T \mathbf{H}_t)^T (\mathbf{1}^T \mathbf{H}_t)) \tag{12}$$

where $\mathbf{1}$ is a column vector with all elements equal to 1. n_s and n_t are the samples number of $\mathbf{H}_s$ and $\mathbf{H}_t$, respectively.

For complex cross-domain scenarios, we consider the variant of MMD, multiple kernel MMD (MK-MMD) (Gretton et al., 2012), which can leverage different kernels to enhance MK-MMD test, leading to a principled method for optimal kernel selection. The discrepancy is computed with respect to a particular representation ϕ, which is a function mapping the latent feature representation to Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$. An empirical approximation to the discrepancy is computed as follows:

$$d_{mmd}^2(P_s, P_t; \phi) = \left\| E_{P_s}[\phi(\mathbf{H}_s)] - E_{P_t}[\phi(\mathbf{H}_t)] \right\|_{\mathcal{H}}^2 \tag{13}$$

where E_{P_s} and E_{P_t} are the means of distribution P_s and P_t in RKHS $\mathcal{H}$, respectively. ϕ corresponds to a kernel function $k(\mathbf{H}_s, \mathbf{H}_t) :=< \phi(\mathbf{H}_s), \phi(\mathbf{H}_t) >_{\mathcal{H}}$, which is defined as the convex combination of m PSD kernels k_u,

$$\mathcal{K} = \left\{ k = \sum_{u=1}^m \beta_u k_u : \sum_{u=1}^m \beta_u = 1, \beta_u \geq 0, \forall u \right\} \tag{14}$$

where the constraints on coefficients β_u are imposed to guarantee that the derived multi-kernel k is characteristic.

3.4. Network Training

The training algorithm is a two-stage framework which is described as follows.

In the first stage, the target domain images are constructed using the ladder autoencoder. We define the total reconstruction loss as the sum of the perceptual loss and pixel reconstruction loss to generate visually indistinguishable images with input images:

$$L_{rec}(\Theta_{enc}, \Theta_{dec}) = \frac{1}{M_p} \left\| \hat{\mathbf{X}}_t - \mathbf{X}_t \right\|_2^2 + \gamma \sum_{l=2}^{L-1} \left\| \mathbf{F}_t^{(l)} - \hat{\mathbf{F}}_t^{(l)} \right\| \tag{15}$$

where $\|\cdot\|_2^2$ is the squared L_2 norm. γ is the weight to balance perceptual loss and pixel reconstruction loss. M_p is the number of pixels of the input image.

In the second stage, supervised classification of the source domain, and the adaption of cross-domain shared features of the two domains, are conducted simultaneously. Our goal is to minimize the following objective:

$$L_{cls}(\Theta_{enc}, \Theta_{cls}) + \lambda L_{adapt}(\Theta_{enc}, \Theta_{cls}) \tag{16}$$

where λ is a weight that controls the interaction of the losses. $L_{cls}(\Theta_{enc}, \Theta_{cls})$ represents a typical cross-entropy loss for classification:

$$L_{cls}(\Theta_{enc}, G_{cls}) = E_{\mathbf{X}_s, \mathbf{y}_s} - \sum_{c=1}^{C} \mathbb{1}[\mathbf{y}_s = c] log(\tilde{\mathbf{y}}_s) \tag{17}$$

where $\mathbb{1}$ is the indicator function. C is the number of classes. $L_{adapt}(\Theta_{enc}, \Theta_{cls})$ denotes the adaptation loss of source and target domains:

$$L_{adapt}(\Theta_{enc}, \Theta_{cls}) = d_{coral} \tag{18}$$

or

$$L_{adapt}(\Theta_{enc}, \Theta_{cls}) = d^2_{mmd}(P_s, P_t; \phi) \tag{19}$$

As mentioned above, the first and second stages are iterated to train the DLaReC. In this way, we can learn domain-invariant cross-domain shared features to classify effectively.

4. Experiments and Results

We evaluate the proposed method on four datasets including Digits, Office31 (Saenko et al., 2010), ImageCLEF-DA[2] and Office-Home (Venkateswara et al., 2017) and compare the results with other methods. DLaReC_C and DLaReC_M mean alignment of cross-domain shared features using Coral and MK-MMD, respectively.

4.1. Digit Datasets Adaptation

We first focus on Digit image across three Digit datasets, including the MNIST (LeCun et al., 1998), USPS (Hull, 1994), and Street View House Numbers (SVHN) (Netzer et al., 2011). Each dataset contains digit images of 10 classes.

Parameter Setting We empirically set $\lambda = 0.2$ and $\gamma = 0.1$. For applying MK-MMD to minimize the cross-domain shared feature discrepancy, we apply $m = 5$ Gaussian kernels. The networks are trained using the ADAM optimizer with learning rate 0.0001 and betas 0.9 and 0.999.

Results The classification accuracy on the Digit datasets for UDA is shown in Table 1. We use MN, US and SV to denote MNIST,

[2]http://imageclef.org/2014/adaptation

Table 1. Classification accuracies (%) of various methods on Digit datasets for UDA. Red and bold numbers denote the best and second best results respectively for each column.

	MN→ US	US→MN	SV→MN	Average
Source only	85.6	65.8	62.3	71.2
Target only	96.5	99.2	99.5	98.4
Metric Discrepancy based Methods				
DAN (Long et al., 2015)	81.1	–	71.1	–
D-CORAL (Sun and Saenko, 2016)	81.7	–	63.1	–
Adversarial Learning based Methods				
DANN (Ganin et al., 2016)	85.1	73.0	73.9	77.3
ADDA (Tzeng et al., 2017)	89.4	90.1	76.0	85.2
MCD (Saito et al., 2018)	96.5	94.1	96.2	95.6
GAN based Methods				
CoGAN (Liu and Tuzel, 2016)	91.2	89.1	not conv.	–
PixelDA (Bousmalis et al., 2017)	95.9	–	–	–
DTN (Taigman et al., 2016)	–	–	84.4	–
UNIT (Liu et al., 2017)	95.9	93.5	90.5	93.3
GenToAdapt (Sankaranarayanan et al., 2018)	92.5	90.8	84.7	89.3
SBADA-GAN (Russo et al., 2018)	97.6	95.0	76.1	89.6
CyCADA (Hoffman et al., 2018)	94.8	95.7	88.3	92.9
I2I (Murez et al., 2018)	95.1	92.2	92.1	93.1
Reconstruction based Methods				
DRCN (Ghifary et al., 2016)	91.8	73.7	82.0	82.5
DSN (Bousmalis et al., 2016)	91.3	–	82.7	–
DLaReC_M (ours)	95.8	**97.3**	94.0	**95.7**
DLaReC_C (ours)	**97.3**	97.8	**95.5**	96.9

Table 2. Classification accuracies (%) of DLaReC variants on Digit datasets for UDA.

	MN→US	US→MN	SV→MN
DLaReC_M (ours)	95.8	97.3	94.0
DLaReC_M (w/o lad.)	94.1	96.5	90.1
DLaReC_M (w/o rec.)	93.8	96.2	89.1
DLaReC_C (ours)	97.3	97.8	95.5
DLaReC_C (w/o lad.)	95.4	95.9	91.2
DLaReC_C (w/o rec.)	95.5	95.8	90.9
DLaReC (w/o dis.)	95.0	96.0	84.7

Table 3. Classification accuracies (%) on Office31 dataset for UDA. All models utilize ResNet-50 as base architecture.

	A→W	D→W	W→D	A→D	D→A	W→A	Average
ResNet-50 (He et al., 2016)	68.4	96.7	99.3	68.9	62.5	60.7	76.1
DAN (Long et al., 2015)	80.5	97.1	99.6	78.6	63.6	62.8	80.4
RTN (Long et al., 2016)	84.5	96.8	99.4	77.5	66.2	64.8	81.6
DANN (Ganin et al., 2016)	82.0	96.9	99.1	79.7	68.2	67.4	82.2
ADDA (Tzeng et al., 2017)	86.2	96.2	98.4	77.8	69.5	68.9	82.9
JAN (Long et al., 2017)	85.4	97.4	**99.8**	84.7	68.6	**70.0**	84.3
GPDA (Kim et al., 2019)	83.9	97.3	**100.0**	85.5	**72.3**	68.8	84.6
MADA (Pei et al., 2018)	90.0	97.4	99.6	87.8	70.3	66.4	85.2
MUDA (Lee and Lee, 2020)	88.2	**98.7**	99.8	90.0	71.2	**69.0**	86.1
CAT (Deng et al., 2019)	91.1	98.6	99.6	**90.6**	70.4	66.5	86.1
DLaReC_M (w/o lad.)	92.4	97.9	99.8	88.8	64.2	65.9	84.8
DLaReC_M (Ours)	**94.7**	98.2	99.8	89.6	68.3	68.2	**86.5**

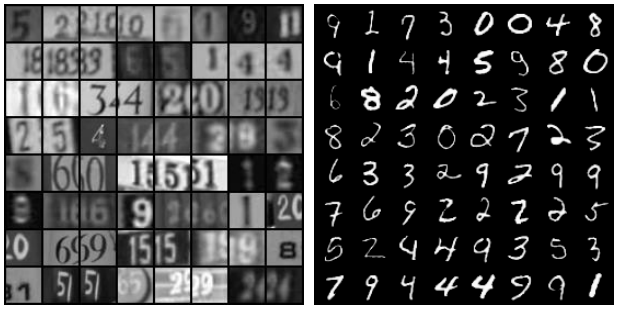
Table 4. Classification accuracies (%) on ImageCLEF-DA dataset for UDA. All models utilize ResNet-50 as base architecture.

	I→P	P→I	I→C	C→I	C→P	P→C	Average
ResNet-50 (He et al., 2016)	74.8	83.9	91.5	78.0	65.5	91.2	80.7
DAN (Long et al., 2015)	75.0	86.2	93.3	84.1	69.8	91.3	83.3
RTN (Long et al., 2016)	75.6	86.8	95.3	86.9	72.7	**92.2**	84.9
DANN (Ganin et al., 2016)	75.0	86.0	**96.2**	87.0	74.3	91.5	85.0
JAN (Long et al., 2017)	76.8	88.0	94.7	89.5	74.2	91.7	85.8
MADA (Pei et al., 2018)	75.0	87.9	96.0	88.8	75.2	**92.2**	85.8
CAT (Deng et al., 2019)	76.7	89.0	94.5	89.8	74.0	**93.7**	86.3
DLaReC_M (w/o lad.)	76.7	88.3	94.7	89.8	75.3	91.7	86.1
DLaReC_M (Ours)	**76.9**	**89.8**	95.9	**90.2**	**76.7**	92.2	**87.0**

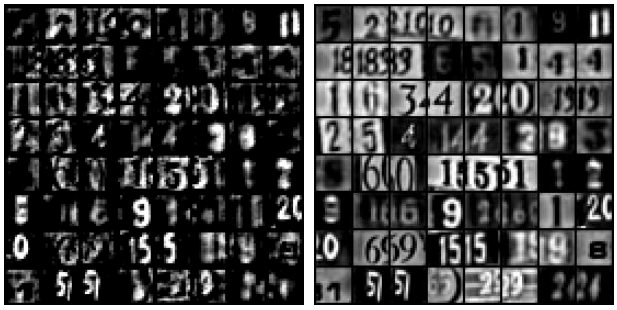
Table 5. Classification accuracies (%) on Office-Home dataset for UDA. All models utilize ResNet-50 as base architecture.

	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Average
ResNet-50 (He et al., 2016)	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DAN (Long et al., 2015)	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3	56.3
DANN (Ganin et al., 2016)	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
JAN (Long et al., 2017)	45.9	61.2	68.9	50.4	59.7	61.0	45.8	43.4	70.3	63.9	52.4	76.8	58.3
DWT (Roy et al., 2019)	**50.8**	**72.0**	**75.8**	58.9	65.6	60.2	57.2	**49.5**	78.3	70.1	**55.3**	78.2	64.3
DLaReC_M (w/o lad.)	48.3	68.9	74.5	60.2	65.8	66.8	59.5	47.7	77.3	69.6	53.3	80.6	64.4
DLaReC_M (Ours)	49.8	69.6	75.7	**60.6**	**67.4**	**68.3**	**60.5**	48.9	**79.0**	**70.5**	54.6	**81.3**	**65.5**

USPS and SVHN respectively. From Table 1, we can see the proposed method performs better overall. Notably on the US→MN task, DLaReC_M achieves the highest performance and outperforms CyCADA by about 2%. In the cases of MN→US and SV→MN,



(a) Source (SVHN) (b) Target (MNIST)



(c) DLaReC_M (w/o lad.) (d) DLaReC_M

Fig. 3. Data reconstruction after training from SVHN to MNIST. (a)-(b) depict the original input images, and (c)-(d) show the reconstructed source images (SVHN).

DLaReC_M is second only to the best algorithm. Even so, our proposed method outperforms the state-of-the-art approaches on the whole. Compared with DRCN, our algorithm has obtained significant improvement, mainly because our proposed DLaReC can pass some irrelevant features directly to the decoder, so that the shared encoder can only learn informative features. DRCN learns task-specific features of the source domain and all the features of the target domain through a shared encoder iteratively, however, the incompatibility of the features will influence the performance of the adaptation. Although DSN has the same motivation with the proposed DLaReC, DSN introduces two private encoder for both domains, which increases the complexity of the model and is difficult to process complex scenarios.

Ablation Studies To disentangle the contributions behind the success of DLaReC, we conduct an ablation study on Digit datasets as shown in Table 2. DLaReC_M (w/o lad.) and DLaReC_C (w/o lad.) denotes there is no ladder connection in the reconstruction. DLaReC_M (w/o rec.) and DLaReC_C (w/o rec.) denotes there is no reconstruction structure, which are similar to DAN and D-CORAL. In case there is a difference about the structure of the network, so we conduct this set of experiments. DLaReC (w/o dis.) means there is no discrepancy metric to adapt cross-domain shared features. Experimental results reveal that only adapting cross-domain shared features have positive effects on classification accuracy. Comparing DLaReC_M and DLaReC_M (w/o lad.), DLaReC_C and DLaReC_C (w/o lad.), we can find the ladder connections plays an important role in obtaining domain-shared features, especially on the complicated SV→MN task. Specially, using discrepancy metric to adapt the cross-domain shared features can obtain significant improvement, which proves that discrepancy metric can help align extracted cross-domain shared features.

In order to further analyze the ladder network, we reconstruct the source domain with the trained model as shown in Fig. 3.

Notably, all trained models only reconstruct the target domain images during the training process. Fig. 3 (a) and (b) denote images of the source domain (SVHN) and the target domain (MNIST), respectively. Fig. 3 (c) shows the reconstruction result of source domain with DLaReC_M (w/o lad.). We can see some digits can be reconstructed but some details are lost. Fig. 3 (d) shows the reconstruction result of DLaReC_M, completely reconstructing the information of the source domain, which demonstrates the shared encoder learn the domain-invariant features for source and target domains, and suppress the domain-specific features.

4.2. Office31, ImageCLEF-DA and Office-Home Datasets Adaptation

Office-31 consists of three different domains: AMAZON (A), DSLR (D), and WEBCAM (W), including 4,652 images in 31 classes. **ImageCLEF-DA** is a benchmark dataset for ImageCLEF 2014 domain adaptation challenge, which is organized by selecting 12 common classes shared by three public datasets (domains): Caltech-256 (C), ImageNet ILSVRC 2012 (I), and Pascal VOC 2012 (P). For each domain, there are 50 images in each category. Each dataset contains 6 domain adaptation tasks. **Office-Home** is a larger dataset with around 15,500 images and contains images of 65 object categories. The dataset contains 4 domains of distinct styles: Artistic (Ar), Clip Art (Cl), Product (Pr) and Real-World (Rw), and we can get 12 domain adaptation tasks. All datasets are down sampled to 256×256 pixels and then a random crop of size 224×224 pixels is used.

Experiment Setting In this section, we apply multi-layer metric embedding to deal with complex domain adaptation tasks. For our encoder branch, we use a ResNet-50 (He et al., 2016) pretrained on ImageNet. The final classification branch contains a linear layer after global average pooling. For the reconstruction pipeline, we choose to reconstruct from the output of the first convolutional layer of ResNet-50 and apply 4 3×3 convolutional layers with feature dimension 1024, 512, 256, 64 in the decoder. Each convolutional layer is followed by a ReLU nonlinearity($f(x) = max(0, x)$) and an upsampling layer except the last convolutional layer. To prevent overfitting, the dropout layer is added after the first two upsampling layers. The network is trained using the mini-batch SGD optimizer with the momentum 0.9. We empirically set $\lambda = 1$ and $\gamma = 0.1$. We apply $m = 5$ Gaussian kernels for MK-MMD. For Office31 and Office-Home datasets, the learning rate is set to 0.0001 for the encoder and decoder, and 0.001 for the classifier layer. For ImageCLEF-DA dataset, the corresponding learning rates are 0.0003 and 0.003 respectively. The learning rate annealing strategy is adopted as (Ganin et al., 2016): $\eta_p = \eta_0(1 + \alpha p)^{-\beta}$, where p changes from 0 to 1, $\alpha = 0.0003$, $\beta = 0.75$ and η_0 is the initial learning rate.

Results The experiment results on Office31, ImageCLEF-DA and Office-Home datasets are reported in Table 3, Table 4 and Table 5 respectively. For fair comparison, results of other methods are directly reported from their original papers. The DLaReC_M (w/o lad.) denotes there is no ladder connection. Our DLaReC_M outperforms all state-of-the-art methods on these three benchmark datasets, highly affirming the effectiveness of our proposed method in focusing on the adaptation of the cross-domain shared features and suppressing the domain-specific variations. It is compelling that our DLaReC_M substantially enhances the classifica-

tion accuracy on the A→W and Cl→Rw task. Comparing DLaReC_M and DLaReC_M (w/o lad.), we can see that adding ladder structures can truly improve performance. It is desirable that DLaReC_M improves the performance on most adaptation tasks, demonstrating the efficiency of our proposed only adapting the cross-domain shared features. Generally speaking, the progress of the proposed DLaReC is mainly due to solving two bottlenecks. The first one is that many existing methods neglect exploring the structure information due to the absence of the target domain. However, the intrinsic information of the target domain is useful and worthy of mining, and can help obtain the domain-shared features. The second one is that existing approaches apply the entire information of each image to adapt without considering the negative effect of the domain-specific variations. Our proposed DLaReC can not only mine the information of the target domain, but also realize the inhibition of domain-specific variations.

5. Conclusion

In this paper, we propose DLaReC for the problem of UDA. It aims at extracting expressive cross-domain shared representations and suppressing domain-specific variations that may hurt target domain classification performance. DLaReC performs multitask learning: data reconstruction, label prediction and cross-domain shared features alignment. Our strong performance improvement clearly shows the effectiveness of DLaReC. Importantly, our study shows that the lateral residual shortcuts connecting the shared encoder and the decoder play a vital role in learning better cross-domain shared features. In our future work, we intend to conduct the combination of DLaReC with the adversarial learning based methods.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant 61701508, and Hunan Provincial Natural Science Foundation of China under Grant 2018JJ3613, and China Scholarship Council.

References

Arora, S., Ge, R., Liang, Y., Ma, T., Zhang, Y., 2017. Generalization and equilibrium in generative adversarial nets (gans), in: ICML, PMLR. pp. 224–232.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., Vaughan, J.W., 2010. A theory of learning from different domains. Machine learning 79, 151–175.

Ben-David, S., Blitzer, J., Crammer, K., Pereira, F., 2007. Analysis of representations for domain adaptation, in: NeurIPS, pp. 137–144.

Borgwardt, K.M., Gretton, A., Rasch, M.J., Kriegel, H.P., Schölkopf, B., Smola, A.J., 2006. Integrating structured biological data by kernel maximum mean discrepancy. Bioinformatics 22, e49–e57.

Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., Krishnan, D., 2017. Unsupervised pixel-level domain adaptation with generative adversarial networks, in: CVPR, pp. 3722–3731.

Bousmalis, K., Trigeorgis, G., Silberman, N., Krishnan, D., Erhan, D., 2016. Domain separation networks, in: NeurIPS, pp. 343–351.

Csurka, G., 2017. A comprehensive survey on domain adaptation for visual applications, in: Domain Adaptation in Computer Vision Applications. Springer. Advances in Computer Vision and Pattern Recognition, pp. 1–35.

Deng, Z., Luo, Y., Zhu, J., 2019. Cluster alignment with a teacher for unsupervised domain adaptation, in: ICCV, pp. 9944–9953.

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T., 2014. Decaf: A deep convolutional activation feature for generic visual recognition, in: ICML, pp. 647–655.

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

Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., Martinez-Gonzalez, P., Garcia-Rodriguez, J., 2018. A survey on deep learning techniques for image and video semantic segmentation. Applied Soft Computing 70, 41–65.

Ghifary, M., Kleijn, W.B., Zhang, M., Balduzzi, D., Li, W., 2016. Deep reconstruction-classification networks for unsupervised domain adaptation, in: ECCV, Springer. pp. 597–613.

Gretton, A., Sejdinovic, D., Strathmann, H., Balakrishnan, S., Pontil, M., Fukumizu, K., Sriperumbudur, B.K., 2012. Optimal kernel choice for large-scale two-sample tests, in: NeurIPS, pp. 1205–1213.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: CVPR, pp. 770–778.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.Y., Isola, P., Saenko, K., Efros, A.A., Darrell, T., 2018. Cycada: Cycle-consistent adversarial domain adaptation. ICML .

Hu, J., Lu, J., Tan, Y.P., 2015. Deep transfer metric learning, in: CVPR, pp. 325–333.

Hu, J., Lu, J., Tan, Y.P., Zhou, J., 2016. Deep transfer metric learning. IEEE Trans. Image Process. 25, 5576–5588.
Hull, J.J., 1994. A database for handwritten text recognition research. IEEE Transactions on pattern analysis and machine intelligence 16, 550–554.
Kim, M., Sahu, P., Gholami, B., Pavlovic, V., 2019. Unsupervised visual domain adaptation: A deep max-margin gaussian process approach, in: CVPR, pp. 4380–4390.
Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks, in: NeurIPS, pp. 1097–1105.
LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al., 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE 86, 2278–2324.
Lee, J., Lee, G., 2020. Model uncertainty for unsupervised domain adaptation, in: ICIP, pp. 1841–1845.
Liu, L., Ouyang, W., Wang, X., Fieguth, P.W., Chen, J., Liu, X., Pietikäinen, M., 2020. Deep learning for generic object detection: A survey. Int. J. Comput. Vis. 128, 261–318.
Liu, M.Y., Breuel, T., Kautz, J., 2017. Unsupervised image-to-image translation networks, in: NeurIPS, pp. 700–708.
Liu, M.Y., Tuzel, O., 2016. Coupled generative adversarial networks, in: NeurIPS, pp. 469–477.
Liu, Y., Tian, X., Li, Y., Xiong, Z., Wu, F., 2019. Compact feature learning for multi-domain image classification, in: CVPR, pp. 7193–7201.
Long, M., Cao, Y., Wang, J., Jordan, M.I., 2015. Learning transferable features with deep adaptation networks. ICML .
Long, M., Cao, Z., Wang, J., Jordan, M.I., 2018. Conditional adversarial domain adaptation, in: Advances in Neural Information Processing Systems, pp. 1640–1650.
Long, M., Zhu, H., Wang, J., Jordan, M.I., 2016. Unsupervised domain adaptation with residual transfer networks, in: NeurIPS, pp. 136–144.
Long, M., Zhu, H., Wang, J., Jordan, M.I., 2017. Deep transfer learning with joint adaptation networks, in: Proceedings of the 34th ICML-Volume 70, JMLR. org. pp. 2208–2217.
Murez, Z., Kolouri, S., Kriegman, D., Ramamoorthi, R., Kim, K., 2018. Image to image translation for domain adaptation, in: CVPR, pp. 4500–4509.
Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y., 2011. Reading digits in natural images with unsupervised feature learning .
Pei, Z., Cao, Z., Long, M., Wang, J., 2018. Multi-adversarial domain adaptation, in: AAAI.
Pezeshki, M., Fan, L., Brakel, P., Courville, A., Bengio, Y., 2016. Deconstructing the ladder network architecture, in: ICML, pp. 2368–2376.
Rasmus, A., Berglund, M., Honkala, M., Valpola, H., Raiko, T., 2015. Semi-supervised learning with ladder networks, in: NeurIPS, pp. 3546–3554.
Roy, S., Siarohin, A., Sangineto, E., Bulo, S.R., Sebe, N., Ricci, E., 2019. Unsupervised domain adaptation using feature-whitening and consensus loss, in: CVPR, pp. 9471–9480.
Russo, P., Carlucci, F.M., Tommasi, T., Caputo, B., 2018. From source to target and back: symmetric bi-directional adaptive gan, in: CVPR, pp. 8099–8108.
Saenko, K., Kulis, B., Fritz, M., Darrell, T., 2010. Adapting visual category models to new domains, in: ECCV, Springer. pp. 213–226.
Saito, K., Watanabe, K., Ushiku, Y., Harada, T., 2018. Maximum classifier discrepancy for unsupervised domain adaptation, in: CVPR, pp. 3723–3732.
Sankaranarayanan, S., Balaji, Y., Castillo, C., Chellappa, R., 2018. Generate to adapt: Aligning domains using generative adversarial networks, in: CVPR, pp. 8503–8512.
Shen, J., Qu, Y., Zhang, W., Yu, Y., 2018. Wasserstein distance guided representation learning for domain adaptation. AAAI .
Sun, B., Feng, J., Saenko, K., 2016. Return of frustratingly easy domain adaptation, in: AAAI.
Sun, B., Saenko, K., 2016. Deep coral: Correlation alignment for deep domain adaptation, in: ECCV, Springer. pp. 443–450.
Taigman, Y., Polyak, A., Wolf, L., 2016. Unsupervised cross-domain image generation. ICLR .
Tzeng, E., Hoffman, J., Saenko, K., Darrell, T., 2017. Adversarial discriminative domain adaptation, in: CVPR, pp. 7167–7176.
Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., Darrell, T., 2014. Deep domain confusion: Maximizing for domain invariance. Computer Science .
Valpola, H., 2015. From neural pca to deep unsupervised learning, in: Advances in Independent Component Analysis and Learning Machines. Elsevier, pp. 143–171.
Venkateswara, H., Eusebio, J., Chakraborty, S., Panchanathan, S., 2017. Deep hashing network for unsupervised domain adaptation, in: CVPR, pp. 5018–5027.
Wang, M., Deng, W., 2018. Deep visual domain adaptation: A survey. Neurocomputing 312, 135–153.
Yosinski, J., Clune, J., Bengio, Y., Lipson, H., 2014. How transferable are features in deep neural networks?, in: NeurIPS, pp. 3320–3328.
Zellinger, W., Grubinger, T., Lughofer, E., Natschläger, T., Saminger-Platz, S., 2017. Central moment discrepancy (CMD) for domain-invariant representation learning .