Deep Adversarial Attention Alignment for Unsupervised Domain Adaptation: The Benefit of Target Expectation Maximization

Guoliang Kang[1], Liang Zheng[1,2], Yan Yan[1], and Yi Yang[1](✉)

[1] CAI, University of Technology Sydney, Ultimo, Australia
{Guoliang.Kang,Yan.Yan-3}@student.uts.edu.au, Yi.Yang@uts.edu.au
[2] Research School of Computer Science, Australian National University,
Canberra, Australia
liangzheng06@gmail.com

Abstract. In this paper, we make two contributions to unsupervised domain adaptation (UDA) using the convolutional neural network (CNN). First, our approach transfers knowledge in all the convolutional layers through attention alignment. Most previous methods align high-level representations, *e.g.,* activations of the fully connected (FC) layers. In these methods, however, the convolutional layers which underpin critical low-level domain knowledge cannot be updated directly towards reducing domain discrepancy. Specifically, we assume that the discriminative regions in an image are relatively invariant to image style changes. Based on this assumption, we propose an attention alignment scheme on all the target convolutional layers to uncover the knowledge shared by the source domain. Second, we estimate the posterior label distribution of the unlabeled data for target network training. Previous methods, which iteratively update the pseudo labels by the target network and refine the target network by the updated pseudo labels, are vulnerable to label estimation errors. Instead, our approach uses category distribution to calculate the cross-entropy loss for training, thereby ameliorating the error accumulation of the estimated labels. The two contributions allow our approach to outperform the state-of-the-art methods by +2.6% on the Office-31 dataset.

Keywords: Domain adaptation · CycleGAN · Attention · EM

1 Introduction

This paper focuses on unsupervised domain adaptation (UDA) for visual classification task. We aim to adapt the knowledge from a source network, trained

Electronic supplementary material The online version of this chapter (https://doi.org/10.1007/978-3-030-01252-6_25) contains supplementary material, which is available to authorized users.

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11215, pp. 420–436, 2018.
https://doi.org/10.1007/978-3-030-01252-6_25

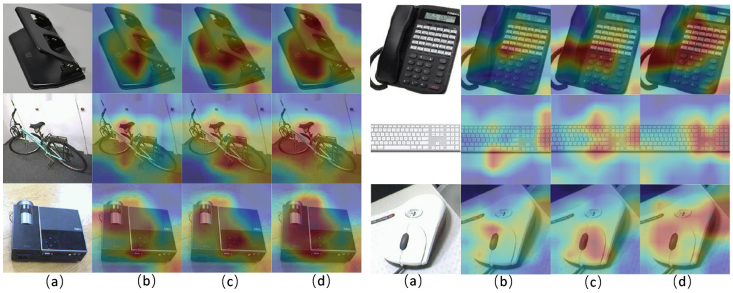

Fig. 1. Attention visualization of the last convolutional layer of ResNet-50. The original *target* input images are illustrated in (**a**). The corresponding attentions of the source network, the target network trained on labeled target data, and the target network adapted with adversarial attention alignment are shown in (**b**), (**c**), and (**d**) respectively.

by the source domain data, to the training of a target network, which will be used for making predications in the target domain. Note that in UDA the *target domain is unlabeled.* The increasing popularity of UDA arises from the fact that the performance of a model trained on one domain may degenerate heavily on another when their underlying data distributions are different.

In the community of UDA, many deep learning methods attempt to minimize the discrepancy across domains on the top layers, such as the fully connected layers, of the neural network via explicitly imposing penalty terms [16,17,23,26] or in an adversarial way [7,24,25]. While the modifications at the fully connected layers can be back-propagated in principle, it may decay after a few layers, especially when gradient explosion or vanishing takes place. Consequently, the convolutional layers may be under-constrained. However, the domain discrepancy may emerge at the start from the convolutional layers, which makes any adjustment purely at the tail of the network less effective.

We investigate the domain discrepancy of the convolutional layers by visualizing their attention mechanisms. In essence, the attention mechanism is emphasized as a key ingredient for CNN, suggested by a number of studies [20,22,27–30,32]. Zagoruyko *et al.* [28] find that the model performance is highly correlated with the attention mechanism: a stronger model always owns better aligned attention than a weaker one. From Fig. 1, suppose we have networks trained on labeled data from source and target domains respectively, we observe distinct attention patterns exhibited by the convolutional layers for the same target domain image. The attention mechanism degenerates when directly applying the source network to the target domain data, which may exert negative influence on the classification performance. Therefore, this paper expects the attention of the convolutional layers to be *invariant to the domain shift.*

Based on the above discussions, this paper takes the domain discrepancy of the convolutional layers directly into account by aligning the attention of the target network with the source network. Our assumption is that no matter how domain varies, the discriminative parts of an image should be insensitive to the changes of image style. Previous discrepancy measures (*e.g.*, MMD [16] and JMMD [17]) which work effectively on high-level *semantic* representations cannot be trivially transferred to measure the attention discrepancy of the convolutional layers where low-level *structure* information is critical. In this paper, we propose using CycleGAN [33] to build the data correspondence across domains, *i.e.*, translating the data from one domain to another without modifying its underlying content. Then, for the paired samples (*e.g.* real source (or target) image and synthetic target (or source) image), we explicitly penalize the distances between attentions of the source and the target networks.

Additionally, we train our target network with real and synthetic data from both source and target domains. For source domain and its translated data, we impose the cross-entropy loss between the predictions and the ground-truth labels. For target domain and its translated source domain data, due to the lack of ground-truth labels, we make use of their underlying category distributions which provide insight into the target data. In a nutshell, we adopt the modified Expectation Maximization (EM) steps to maximize the likelihood of target domain images and update the model. Training iterations improve both the label posterior distribution estimation and the discriminative ability of the model.

Our contributions are summarized below,

- We propose a deep attention alignment method which allows the target network to mimic the attention of the source network. Taking advantage of the pairing nature of CycleGAN, no additional supervision is needed.
- We propose using EM algorithm to exploit the unlabeled target data to update the network. Several modifications are made to stabilize training and improve the adaptation performance.
- Our method outperforms the state of art in all the six transfer tasks, achieving +2.6% improvement in average on the real-world domain adaptation dataset Office-31.

2 Related Work

Unsupervised Domain Adaptation. Various methods have been proposed for unsupervised domain adaptation [7,16,17,26]. Many works try to make the representations at the tail of neural networks invariant across domains. Tzeng *et al.* [26] propose a kind of domain confusion loss to encourage the network to learn both semantically meaningful and domain invariant representations. Similarly, Long *et al.* [16] minimize the MMD distance of the fully-connected activations between source and target domain while sharing the convolutional features. Ganin *et al.* [7] enable the network to learn domain invariant representations in an adversarial way by adding a domain classifier and back-propagating inverse gradients. JAN [17] penalizes the JMMD over multiple fully-connected layers to minimize the domain discrepancy coming from both the data distribution and the label distribution. Further, JAN-A [17], as a variant of JAN, trains the network in an adversarial way with JMMD as the domain adversary.

DSN [3] explicitly models domain-specific features to help improve networks' ability to learn domain-invariant features. Associative domain adaptation (ADA) [8] reinforces associations across domains directly in embedding space to extract statistically domain-invariant and class discriminative features. Few works pay attention to the domain shift coming from the convolutional layers. In this paper, we notice that the attention mechanism cannot be preserved when directly applying the model trained on the source domain to the target domain. To alleviate this problem, we constrain the training of convolutional layers by imposing the attention alignment penalty across domains.

Attention of CNNs. There exist many ways to define and visualize the attention mechanisms learned by CNNs. Zeiler and Fergus [29] project certain features back onto the image through a network called "deconvnet" which shares the same weights as the original feed-forward network. Simonyan *et al.* [22] propose using the gradient of the class score *w.r.t* the input image to visualize the CNN. Class activation maps (CAMs), proposed by [32], aim to visualize the class-discriminative image regions used by a CNN. Grad-CAM [20] combines gradient based attention method and CAM, enabling to obtain class-discriminative attention maps without modifying the original network structure as [32].

Zagoruyko *et al.* [28] define attention as a set of spatial maps indicating which area the network focuses on to perform a certain task. The attention maps can also be defined *w.r.t* various layers of the network so that they are able to capture both low-, mid-, and high-level representation information. They propose that attention mechanism should be a kind of knowledge transferred across different *network architectures*. Zaogruyko *et al.* [28] align the attention across different architectures for exactly the same image during the training process and aim to transfer the knowledge from a large model to a smaller one. Different to [28], our method aligns the attention across different *data domains* where images across domains are unpaired and aims to promote the model adaptation performance.

Unpaired Image-to-Image Translation. Unpaired image-to-image translation aims to train a model to map image samples across domains, under the absence of pairing information. It can be realized through GAN to pair the real source (or target) and synthetic target (or source) images [2,11,12,14,15,19,21,33]. Generating synthetic images can be beneficial for various vision tasks [5,6,18,31]. In this paper, we concentrate on maximizing the utility of given paired real and synthetic samples. And we choose CycleGAN [33] to perform such adversarial data pairing.

3 Deep Adversarial Attention Alignment

Our framework is illustrated in Fig. 2. We train a source CNN which guides the attention alignment of the target CNN whose convolutional layers have the same architecture as the source network. The target CNN is trained with a mixture of real and synthetic images from both source and target domains. For source and synthetic target domain data, we have ground-truth labels and use them to train the target network with cross-entropy loss. On the other hand, for the target and synthetic source domain data, due to the lack of ground-truth labels, we optimize the target network through an EM algorithm.

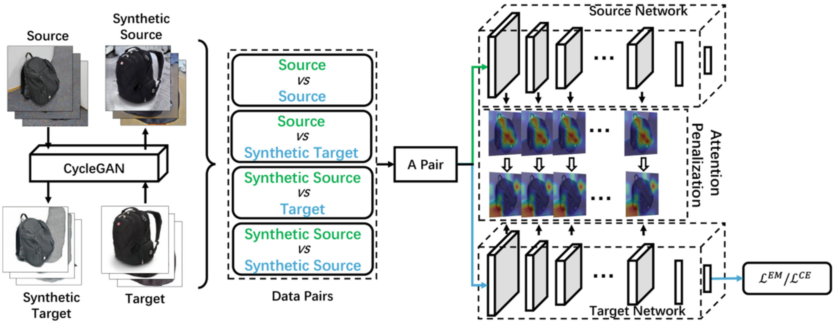

Fig. 2. The framework of deep adversarial attention alignment. We train a source network and fix it. The source network guides the attention alignment of the target network. The target network is trained with real and synthetic images from both domains. For labeled real source and synthetic target data, we update the network by computing the cross-entropy loss between the predictions and the ground-truth labels. For unlabeled real target and synthetic source images, we maximize the likelihood of the data with EM steps. The attention distance for a pair of images (as illustrated in the "Data Pairs" block) passing through the source network and the target network, respectively, is minimized.

3.1 Adversarial Data Pairing

We use CycleGAN to translate the samples in the source domain S to those in the target domain T, and vice versa. The underlying assumption to obtain meaningful translation is that there exist some relationships between two domains. For unsupervised domain adaptation, the objects of interest across domains belong to the same set of category. So it is possible to use CycleGAN to map the sample in the source domain to that in the target domain while maintaining the underlying object-of-interest.

The Generative Adversarial Network (GAN) aims to generate synthetic images which are indistinguishable from real samples through an adversarial loss,

$$\mathcal{L}^{GAN}(G^{ST}, D^T, X^S, X^T) = \mathbb{E}_{x^T}[\log D^T(x^T)] + \mathbb{E}_{x^S}[1 - \log D^T(G^{ST}(x^S))], \tag{1}$$

where x^S and x^T are sampled from source domain S and target domain T, respectively. The generator G^{ST} mapping X^S to X^T strives to make its generated synthetic outputs $G^{ST}(x^S)$ indistinguishable from real target samples x^T for the domain discriminator D^T.

Because the training data across domains are unpaired, the translation from source domain to target domain is highly under-constrained. CycleGAN couples the adversarial training of this mapping with its inverse one, *i.e.* the mapping from S to T and that from T to S are learned concurrently. Moreover, it introduces a cycle consistency loss to regularize the training,

$$\mathcal{L}^{cyc}(G^{ST}, G^{TS}) = \mathbb{E}_{x^S}[\|G^{TS}(G^{ST}(x^S)) - x^S\|_1] + \mathbb{E}_{x^T}[\|G^{ST}(G^{TS}(x^T)) - x^T\|_1], \quad (2)$$

Formally, the full objective for CycleGAN is,

$$\begin{aligned}\mathcal{L}^{cyc}(G, F, D_X, D_Y) = {} & \mathcal{L}^{GAN}(G^{ST}, D^T, X^S, X^T) + \mathcal{L}^{GAN}(G^{TS}, D^S, X^T, X^S) \\ & + \lambda \mathcal{L}^{cyc}(G^{ST}, G^{TS}),\end{aligned} \quad (3)$$

where the constant λ controls the strength of the cycle consistency loss. Through CycleGAN, we are able to translate an image in the source domain to that in the target domain in the context of our visual domain adaptation tasks (Fig. 3).

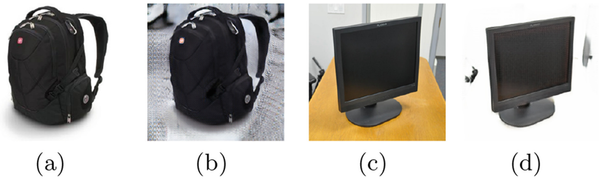

Fig. 3. Paired data across domains using CycleGAN. (**a**) and (**c**): real images sampled from source and target domain, respectively. (**b**): a synthetic target image paired with (**a**) through G^{ST}. (**d**): a synthetic source image paired with a real target image (**c**) through G^{TS}.

As illustrated in Fig. 1, the target model pays too much attention to the irrelevant background or less discriminative parts of the objects of interest. This attention misalignment will degenerate the model's performance. In this paper, we propose to use the style-translated images as natural image correspondences to guide the attention mechanism of the target model to mimic that of the source model, to be detailed in Sect. 3.2.

3.2 Attention Alignment

Based on the paired images, we propose imposing the attention alignment penalty to reduce the discrepancy of attention maps across domains. Specifically, we represent attention as a function of spatial maps *w.r.t* each convolutional layer [28]. For the input x of a CNN, let the corresponding feature maps *w.r.t* layer l be represented by $F_l(x)$. Then, the attention map $A_l(x)$ *w.r.t* layer l is defined as

$$A_l(x) = \sum_c |F_{l,c}(x)|^2, \quad (4)$$

where $F_{l,c}(x)$ denotes the c-th channel of the feature maps. The operations in Eq. (4) are all element-wise. Alternative ways to represent the attention maps include $\sum_c |F_{l,c}|$, and $\max |F_{l,c}|$, *etc.* We adopt Eq. (4) to emphasize the salient parts of the feature maps.

We propose using the source network to guide the attention alignment of the target network, as illustrated in Fig. 2. We penalize the distance between the vectorized attention maps between the source and the target networks to minimize their discrepancy. In order to make the attention mechanism invariant to the domain shift, we train the target network with a mixture of real and synthetic data from both source and target domains.

Formally, the attention alignment penalty can be formulated as,

$$\begin{aligned}\mathcal{L}^{AT} = \sum_l \{ \sum_i \| \frac{A_l^S(x_i^S)}{\|A_l^S(x_i^S)\|_2} - \frac{A_l^T(x_i^S)}{\|A_l^T(x_i^S)\|_2} \|_2 + \sum_j \| \frac{A_l^S(x_j^S)}{\|A_l^S(x_j^S)\|_2} - \frac{A_l^T(\tilde{x}_j^T)}{\|A_l^T(\tilde{x}_j^T)\|_2} \|_2 \\ + \sum_m \| \frac{A_l^S(\tilde{x}_m^S)}{\|A_l^S(\tilde{x}_m^S)\|_2} - \frac{A_l^T(\tilde{x}_m^S)}{\|A_l^T(\tilde{x}_m^S)\|_2} \|_2 + \sum_n \| \frac{A_l^S(\tilde{x}_n^S)}{\|A_l^S(\tilde{x}_n^S)\|_2} - \frac{A_l^T(x_n^T)}{\|A_l^T(x_n^T)\|_2} \|_2 \},\end{aligned} \tag{5}$$

where the subscript l denotes the layer and i, j denote the samples. The A_l^S and A_l^T represent the attention maps *w.r.t* layer l for the source network and the target network, respectively. x^S and x^T are real source and real target domain data, respectively. The synthetic target data $\tilde{x}_i^T$ and synthetic source data $\tilde{x}_n^S$ satisfy $\tilde{x}_i^T = G^{ST}(x_i^S)$ and $\tilde{x}_n^S = G^{TS}(x_n^T)$, respectively.

Through Eq. (5), the distances of attention maps for the paired images (*i.e.*, $(x_j^S, \tilde{x}_j^T)$ and $(x_n^T, \tilde{x}_n^S)$) are minimized. Moreover, we additionally penalize the attention maps of the same input (*i.e.*, x_i^S and $\tilde{x}_m^S$) passing through different networks. The attention alignment penalty $\mathcal{L}^{AT}$ allows the attention mechanism to be gradually adapted to the target domain, which makes the attention mechanism of the target network invariant to the domain shift.

Discussion. On minimizing the discrepancy across domains, our method shares similar ideas with DAN [16] and JAN [17]. The difference is that our method works on the convolutional layers where the critical structure information is captured and aligned across domains; in comparison, DAN and JAN focus on the FC layers where high-level semantic information is considered. Another notable difference is that our method deals with the image-level differences through CycleGAN data pairing, whereas DAN and JAN consider the discrepancy of feature distributions.

In DAN and JAN, MMD and JMMD criteria are adopted respectively to measure the discrepancy of feature distributions across domains. Technically, MMD and JMMD can also be used as attention discrepancy measures. However, as to be shown in the experiment part, MMD and JMMD yield inferior performance to the L_2 distance enabled by adversarial data pairing in our method. The reason is that MMD and JMMD are distribution distance estimators: they map the attention maps to the Reproducing Kernel Hilbert Space (RKHS) and lose the structure information. So they are not suitable for measuring the attention discrepancy across domains.

3.3 Training with EM

To make full use of the available data (labeled and unlabeled), we train the target-domain model with a mixture of real and synthetic data from both source and target domains, as illustrated in Fig. 2. For the source and its translated synthetic target domain data, we compute the cross-entropy loss between the predictions and ground-truth labels to back-propagate the gradients through the target network. The cross-entropy loss for the source and corresponding synthetic target domain data can be formulated as follows,

$$\mathcal{L}^{CE} = -[\sum_i \log p_\theta(y_i^S|x_i^S) + \sum_j \log p_\theta(y_j^S|\tilde{x}_j^T)], \tag{6}$$

where $y^S \in \{1, 2, \cdots, K\}$ denotes the label for the source sample x^S and the translated synthetic target sample $\tilde{x}^T$. The probability $p_\theta(y|x)$ is represented by the y-th output of the target network with parameters θ given the input image x. $\tilde{x}_j^T = G^{ST}(x_j^S)$.

For the unlabeled target data, due to the lack of labels, we employ the EM algorithm to optimize the target network. The EM algorithm can be split into two alternative steps: the (**E**)xpectation computation step and the expectation (**M**)aximization step. The objective is to maximize the log-likelihood of target data samples,

$$\sum_i \log p_\theta(x_i^T), \tag{7}$$

In image classification, our prior is that the target data samples belong to K different categories. We choose the underlying category $z_i \in \{1, 2, \cdots, K\}$ of each sample as the hidden variable, and the algorithm is depicted as follows (we omit the sample subscript and the target domain superscript for description simplicity).

(i) The Expectation step. We first estimate $p_{\theta_{t-1}}(z|x)$ through,

$$p_{\theta_{t-1}}(z|x) = \frac{p_{\theta_{t-1}}(x|z)p(z)}{\sum_z p_{\theta_{t-1}}(x|z)p(z)}, \tag{8}$$

where the distribution $p_{\theta_{t-1}}(z|x)$ is modeled by the target network. θ_{t-1} is the parameters of the target-domain CNN at last training step $t-1$. We adopt the uniform distributions to depict $p(z)$ (*i.e.,* assuming the occurrence probabilities of all the categories are the same) and $p(x)$ (*i.e.,* assuming all possible image instantiations are distributed uniformly in the manifold of image gallery). In this manner, $p_{\theta_{t-1}}(z|x) = \alpha p_{\theta_{t-1}}(x|z)$ where α is a constant.

(ii) The Maximization step. Based on the computed posterior $p_{\theta_{t-1}}(z|x)$, our objective is to update θ_t to improve the lower bound of Eq. (7),

$$\sum_z p_{\theta_{t-1}}(z|x) \log p_{\theta_t}(x|z) \tag{9}$$

Note that we omit $\sum_z p_{\theta_{t-1}}(z|x) \log p(z)$ because we assume $p(z)$ subjects to the uniform distribution which is irrelevant to θ_t. Also, because $p_\theta(z|x) = p_\theta(x|z)$, Eq. (9) is equivalent to,

$$\sum_z p_{\theta_{t-1}}(z|x) \log p_{\theta_t}(z|x). \tag{10}$$

Moreover, we propose to improve the effectiveness and stability of the above EM steps through three aspects

(A) Asynchronous update of $p(z|x)$. We adopt an independent network M^{post} to estimate $p(z|x)$ and update M^{post} asynchronously, *i.e.,* M^{post} synchronizes its parameters θ^{post} with the target network every N steps: $\theta_t^{post} = \theta_{\lfloor t/N \rfloor \times N}$. In this manner, we avoid the frequent update of $p(z|x)$ and make the training process much more stable.

(B) Filtering the inaccurate estimates. Because the estimate of $p(z|x)$ is not accurate, we set a threshold p_t and discard the samples whose maximum value of $p(z|x)$ over z is lower than p_t.

(C) Initializing the learning rate schedule after each update of M^{post}. To accelerate the target network adapting to the new update of the distribution $p(z|x)$, we choose to initialize the learning rate schedule after each update of M^{post}.

Note that for synthetic source data $\tilde{x}^S = G^{TS}(x^T)$, we can also apply the modified EM steps for training. Because G^{TS} is a definite mapping, we assume $p(z|\tilde{x}^S) = p(z|x^T)$.

To summarize, when using the EM algorithm to update the target network with target data and synthetic source data, we first compute the posterior $p(z|x^T)$ through network M^{post} which synchronizes with the target network every N steps. Then we minimize the loss,

$$\mathcal{L}^{EM} = -\{\sum_i \sum_{z_i} p_{\theta^{post}}(z_i|x_i^T) \log p_\theta(z_i|x_i^T) + \sum_j \sum_{z_j} p_{\theta^{post}}(z_j|x_j^T) \log p_\theta(z_j|\tilde{x}_j^S)\}. \tag{11}$$

In our experiment, we show that these modifications yield consistent improvement over the basic EM algorithm.

3.4 Deep Adversarial Attention Alignment

Based on the above discussions, our full objective for training the target network can be formulated as,

$$\min_\theta \mathcal{L}^{full} = \mathcal{L}^{CE} + \mathcal{L}^{EM} + \beta \mathcal{L}^{AT} \tag{12}$$

where β determines the strength of the attention alignment penalty term $\mathcal{L}^{AT}$.

Discussion. Our approach mainly consists of two parts: attention alignment and EM training. On the one hand, attention alignment is crucial for the success of EM training. For EM training, there originally exists no constraint that the estimated hidden variable Z is assigned with the semantic meaning aligned with the ground-truth label, *i.e.* there may exist label shift or the data is clustered in an undesirable way. Training with labeled data (*e.g.* source and synthetic target data) and synchronizing θ^{post} with θ, the above issue can be alleviated. In addition, attention alignment further regularizes the training process by encouraging the network to focus on the desirable discriminative information.

On the other hand, EM benefits attention alignment by providing label distribution estimations for target data. EM approximately guides the attention of target network to fit the target domain statistics, while attention alignment regularizes the attention of target network to be not far from source network. These two seemingly adversarial counterparts cooperate to make the target network acquire the attention mechanism which is invariant to the domain shift.

Note that both parts are promoted by the use of adversarial data pairing which provides natural image correspondences to perform attention alignment. Thus our method is named "deep adversarial attention alignment".

4 Experiment

4.1 Setup

Datasets. We use the following two UDA datasets for image classification.

(1) Digit datasets from **MNIST** [13] (60,000 training + 10,000 test images) to **MNIST-M** [7] (59,001 training + 90,001 test images). MNIST and MNIST-M are treated as the source domain and target domain, respectively. The images of MNIST-M are created by combining MNIST digits with the patches randomly extracted from color photos of BSDS500 [1] as their background.

(2) **Office-31** is a standard benchmark for real-world domain adaptation tasks. It consists of 4,110 images subject to 31 categories. This dataset contains three distinct domains, (1) images which are collected from the Amazon website (**A**mazon domain), (2) web camera (**W**ebcam domain), and (3) digital SLR camera (**D**SLR domain) under different settings, respectively. The dataset is also imbalanced across domains, with 2,817 images in **A** domain, 795 images in **W** domain, and 498 images in **D** domain. We evaluate our algorithm for six transfer tasks across these three domains, including **A** → **W**, **D** → **W**, **W** → **D**, **A** → **D**, **D** → **A**, and **W** → **A**.

Competing Methods. We compare our method with some representative and state-of-the-art approaches, including RevGrad [7], JAN [17], JAN-A [17], DSN [3] and ADA [8] which minimize domain discrepancy on the FC layers of CNN. We compare with the results of these methods reported in their published papers with identical evaluation setting. For the task MNIST → MNIST-M, we also compare with PixelDA [2], a state-of-the-art method on this task. Both CycleGAN and PixelDA transfer the source style to the target domain without modifying its content heavily. Therefore, PixelDA is an alternative way to generate paired images across domains and is compatible to our framework. We emphasize that a model capable of generating more genuine paired images will probably lead to higher accuracy using our method. The investigation in this direction can be parallel and reaches beyond the scope of this paper.

4.2 Implementation Details

MNIST → MNIST-M. The source network is trained on the MNIST training set. When the source network is trained, it is fixed to guide the training of the target network. The target and the source network are made up of four convolutional layers, where the first three are for feature extraction and the last one acts as a classifier. We align the attention between the source and target network for the three convolutional layers.

Office-31. To make a fair comparison with the state-of-the-art domain adaptation methods [17], we adopt the ResNet-50 [9,10] architecture to perform the adaptation tasks on Office-31 and we start from the model pre-trained on ImageNet [4]. We first fine-tune the model on the source domain data and fix it. The source model is then used to guide the attention alignment of the target network. The target network starts from the fine-tuned model and is gradually trained to adapt to the target domain data. We penalize the distances of the attention maps *w.r.t* all the convolutional layers except for the first convolutional layer.

Detailed settings of training are demonstrated in the supplementary material.

4.3 Evaluation

MNIST → MNIST-M. The classification results of transferring MNIST to MNIST-M are presented in Table 1. We arrive at four observations. First, our method outperforms a series of representative domain adaptation methods (*e.g.*, RevGrad, DSN, ADA) with a large margin, all of which minimize the domain discrepancy at the FC layers of neural networks. Moreover, we achieve competitive accuracy (95.6%) to the state-of-the-art result (98.2%) reported by PixelDA. Note that technically, PixelDA is compatible to our method, and can be adopted to improve the accuracy of our model. We will investigate this in the future. Second, we observe that the accuracy of the source network drops heavily when transferred to the target domain (from 99.3% on source test set to 45.6% on target test set), which implies the significant domain shift from MNIST to MNIST-M. Third, we can see that the distribution of synthetic target data is much closer to real target data than real source data, by observing that training with synthetic target data improves the performance over the source network by about +30%. Finally, training with a mixture of source and synthetic target data is beneficial for learning domain invariant features, and improves the adaptation performance by +3.5% over the model trained with synthetic target data only.

Table 1 demonstrates that our EM training algorithm is an effective way to exploit unlabeled target domain data. Moreover, imposing the attention alignment penalty $\mathcal{L}^{AT}$ always leads to noticeable improvement.

Table 1. Classification accuracy (%) for MNIST → MNIST-M. "CNN" denotes the source and target network (Sect. 4.2). The "S" and "T_f" represent labeled source data and synthetic target data, respectively. The "T" and "S_f" denote unlabeled target data and synthetic source data, respectively

Method	Train Data	Accuracy (%)
RevGrad [7]	S+T	81.5
DSN [3]	S+T	83.2
ADA [8]	S+T	85.9
PixelDA [2]	S+T+T_f	**98.2**
Ours (wo $\mathcal{L}^{AT}$)	S+T_f+T+S_f	93.5
Ours (w $\mathcal{L}^{AT}$)	S+T_f+T+S_f	**95.6**

Method	Train Data	Accuracy (%)
CNN	S	45.6
CNN	T_f	75.0
CNN	S+T_f	78.5
CNN + $\mathcal{L}^{AT}$	S+T_f	85.7
Ours (wo $\mathcal{L}^{AT}$)	S+T_f+T+S_f	93.5
Ours (w $\mathcal{L}^{AT}$)	S+T_f+T+S_f	**95.6**

Table 2. Classification accuracy (%) on the Office-31 dataset based on ResNet-50

Method	Train Data	A → W	D → W	W → D	A → D	D → A	W → A	Average
ResNet-50	S	68.4 ± 0.2	96.7 ± 0.1	99.3 ± 0.1	68.9 ± 0.2	62.5 ± 0.3	60.7 ± 0.3	76.1
RevGrad [7]	S+T	82.0 ± 0.4	96.9 ± 0.2	99.1 ± 0.1	79.7 ± 0.4	68.2 ± 0.4	67.4 ± 0.5	82.2
JAN [17]	S+T	85.4 ± 0.3	97.4 ± 0.2	99.8 ± 0.2	84.7 ± 0.3	68.6 ± 0.3	70.0 ± 0.4	84.3
JAN-A [17]	S+T	86.0 ± 0.4	96.7 ± 0.3	99.7 ± 0.1	85.1 ± 0.4	69.2 ± 0.4	70.7 ± 0.5	84.6
ResNet-50	T_f	81.1 ± 0.2	98.5 ± 0.2	99.8 ± 0.0	83.3 ± 0.3	61.0 ± 0.2	60.2 ± 0.3	80.6
ResNet-50	S+T_f	81.9 ± 0.2	98.5 ± 0.2	99.8 ± 0.0	83.7 ± 0.3	66.5 ± 0.2	64.8 ± 0.3	82.5
Ours (wo $\mathcal{L}^{AT}$)	T_f+T	86.2 ± 0.2	**99.3** ± 0.1	**100** ± 0.0	86.5 ± 0.6	69.9 ± 0.6	70.2 ± 0.2	85.4
Ours (w $\mathcal{L}^{AT}$)	T_f+T	86.8 ± 0.2	**99.3** ± 0.1	**100** ± 0.0	87.2 ± 0.5	71.7 ± 0.5	71.8 ± 0.1	86.1
Ours (wo $\mathcal{L}^{AT}$)	S+T_f+T+S_f	**87.1** ± 0.3	**99.3** ± 0.1	**100** ± 0.0	87.1 ± 0.2	72.3 ± 0.2	72.2 ± 0.2	86.3
Ours (w $\mathcal{L}^{AT}$)	S+T_f+T+S_f	86.8 ± 0.2	**99.3** ± 0.1	**100** ± 0.0	**88.8** ± 0.4	**74.3** ± 0.2	**73.9** ± 0.2	**87.2**

Office-31. The classification results based on ResNet-50 are shown in Table 2. With identical evaluation setting, we compare our methods with previous transfer methods and variants of our method. We have three major conclusions.

First, from Table 2, it can be seen that our method outperforms the state of art in all the transfer tasks with a large margin. The improvement is larger on harder transfer tasks, where the source domain is substantially different from and has much less data than the target domain, *e.g.* **D** → **A**, and **W** → **A**.

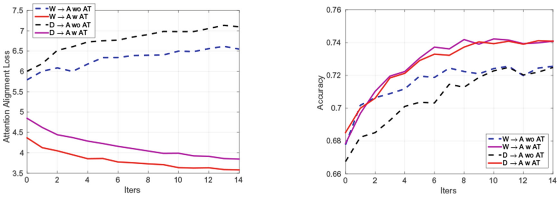

Fig. 4. Analysis of the training process (EM is implemented). **Left**: The trend of $\mathcal{L}^{AT}$ during training with and without imposing the $\mathcal{L}^{AT}$ penalty term. **Right**: The curves of test accuracy on the target domain. The results of tasks **W** → **A** and **D** → **A** are presented. The results for other tasks are similar. One iteration here represents one update of the network M^{post} (see Sect. 3.3).

Table 3. Variants of the EM algorithm with and without $\mathcal{L}^{AT}$. The EM algorithm without asynchronous update of M^{post} is denoted by EM-A, while that without filtering the noisy data is denoted by EM-B. EM-C represents EM training without initializing the learning rate schedule when M^{post} is updated

Method	Train Data	A → W	A → D	D → A	W → A	Average
ResNet-50	S	68.4 ± 0.2	68.9 ± 0.2	62.5 ± 0.3	60.7 ± 0.3	65.1
EM-A	S+T$_f$+T+S$_f$	68.6 ± 0.3	73.5 ± 0.3	62.7 ± 0.3	52.8 ± 0.3	64.4
EM-A + $\mathcal{L}^{AT}$	S+T$_f$+T+S$_f$	80.4 ± 0.2	79.1 ± 0.2	66.4 ± 0.2	58.4 ± 0.2	71.1
EM-C	S+T$_f$+T+S$_f$	86.4 ± 0.3	87.0 ± 0.3	69.5 ± 0.3	71.4 ± 0.3	78.6
EM-C + $\mathcal{L}^{AT}$	S+T$_f$+T+S$_f$	86.2 ± 0.2	86.6 ± 0.3	71.8 ± 0.3	73.7 ± 0.2	79.6
EM-B	S+T$_f$+T+S$_f$	*very low*	*very low*	*very low*	*very low*	*very low*
EM-B + $\mathcal{L}^{AT}$	S+T$_f$+T+S$_f$	*very low*	*very low*	*very low*	*very low*	*very low*
Ours (wo $\mathcal{L}^{AT}$)	S+T$_f$+T+S$_f$	**87.1** ± 0.3	87.1 ± 0.2	72.3 ± 0.2	72.2 ± 0.2	79.7
Ours (w $\mathcal{L}^{AT}$)	S+T$_f$+T+S$_f$	86.8 ± 0.2	**88.8** ± 0.4	**74.3** ± 0.2	**73.9** ± 0.2	**80.9**

Table 4. Comparison of different attention discrepancy measures on Office-31

Measure	**A → W**	**A → D**	**D → A**	**W → A**	Average
L_1-norm	*very low*	*very low*	*very low*	*very low*	*very low*
MMD	84.7	84.1	66.2	64.5	74.9
JMMD	85.9	85.3	70.1	71.1	78.1
Ours	**86.8**	**88.8**	**74.3**	**73.9**	**80.9**

Specifically, we improve over the state of art result by +2.6% on average, and by +5.1% for the difficult transfer task **D** → **A**.

Second, we also compare our method with and without the adversarial attention alignment loss $\mathcal{L}^{AT}$. Although for easy transfer tasks, the performance of these two variants are comparable, when moving to much harder tasks, we observe obvious improvement brought by the adversarial attention alignment, *e.g.,* training with adversarial attention alignment outperforms that without attention alignment by +2% for the task **D** → **A**, and +1.7% for the task **W** → **A**. This implies that adversarial attention alignment helps reduce the discrepancy across domains and regularize the training of the target model.

Third, we validate that augmenting with synthetic target data to facilitate the target network training brings significant improvement of accuracy over source network. This indicates that the discrepancy between synthetic and real target data is much smaller. We also notice that in our method, the accuracy of the network trained with real and synthetic data from both domains is much better than the one purely trained with real and synthetic target data. This verifies the knowledge shared by the source domain can be sufficiently uncovered by our framework to improve the target network performance.

Figure 4 illustrates how the attention alignment penalty $\mathcal{L}^{AT}$ changes during the training process with and without this penalty imposed. Without attention alignment, the discrepancy of the attention maps between the source and target network is significantly larger and increases as the training goes on. The improvement of accuracy brought by adding $\mathcal{L}^{AT}$ penalty to the objective can be attributed to the much smaller discrepancy of attention maps between the source and the target models, *i.e.,* better aligned attention mechanism. The testing accuracy curves on the target domain for tasks $\mathbf{D} \rightarrow \mathbf{A}$ and $\mathbf{D} \rightarrow \mathbf{A}$ are also drawn in Fig. 4. It can be seen that the test accuracy steadily increases and the model with $\mathcal{L}^{AT}$ converges much faster than that without any attention alignment.

Visualization of the attention maps of our method is provided in Fig. 1. We observe that through attention alignment, the attention maps of the target network adapt well to the target domain images, and are even better than those of the target model trained on labeled target images.

4.4 Ablation Study

Table 3 compares the accuracy of different EM variants. We conduct ablation studies by removing one component from the system at a time (three components are considered which are defined in Sect. 3.3). For each variant of EM, we also evaluate the effect of imposing $\mathcal{L}^{AT}$ by comparing training with and without $\mathcal{L}^{AT}$. By comparing the performances of EM-A, EM-B, EM-C and full method we adopted, we find that the three modifications all contribute considerably to the system. Among them, filtering the noisy data is the most important factor. We also notice that for EM-A and EM-C, training along with $\mathcal{L}^{AT}$ always leads to a significant improvement, implying performing attention alignment is an effective way to improve the adaptation performance.

4.5 Comparing Different Attention Discrepancy Measures

In this section, we provide a method comparison in measuring the attention discrepancy across domains which is discussed in Sect. 3.2. This paper uses the L_2 distance, and the compared methods include the L_1 distance, MMD [16] and JMMD [17]. Results are presented in Table 4.

We find that our method achieves the best results among the four measures. The L_1 distance fails in training a workable network because it is misled by the noise in the attention maps. Our method outperforms MMD/JMMD by a large margin, because our method preserves the structure information, as discussed in Sect. 3.2.

5 Conclusion

In this paper, we make two contributions to the community of UDA. First, from the *convolutional layers*, we propose to align the attention maps of the source network and target network to make the knowledge from source network better adapted to the target one. Second, from an *EM perspective*, we maximize the likelihood of unlabeled target data, which enables target network to leverage more training data for better domain adaptation. Both contributions benefit from the unsupervised image correspondences provided by CycleGAN. Experiment demonstrates that the two contributions both have positive effects on the system performance, and they cooperate together to achieve competitive or even state-of-the-art results on two benchmark datasets.

Acknowledgement. We acknowledge the Data to Decisions CRC (D2D CRC) and Cooperative Research Centres Programme for funding the research.

References

1. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **33**(5), 898–916 (2011)
2. Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., Krishnan, D.: Unsupervised pixel-level domain adaptation with generative adversarial networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
3. Bousmalis, K., Trigeorgis, G., Silberman, N., Krishnan, D., Erhan, D.: Domain separation networks. In: Advances in Neural Information Processing Systems, pp. 343–351 (2016)
4. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 248–255. IEEE (2009)
5. Ding, M., Fan, G.: Multilayer joint gait-pose manifolds for human gait motion modeling. IEEE Trans. Cybern. **45**(11), 2413–2424 (2015)
6. Dong, X., Yan, Y., Ouyang, W., Yang, Y.: Style aggregated network for facial landmark detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 379–388, June 2018
7. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: International Conference on Machine Learning, pp. 1180–1189 (2015)
8. Haeusser, P., Frerix, T., Mordvintsev, A., Cremers, D.: Associative domain adaptation. In: International Conference on Computer Vision (ICCV), vol. 2, p. 6 (2017)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
10. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 630–645. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_38
11. Hoffman, J., et al.: Cycada: Cycle-consistent adversarial domain adaptation. arXiv preprint arXiv:1711.03213 (2017)
12. Kim, T., Cha, M., Kim, H., Lee, J., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: International Conference on Machine Learning (2017)
13. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
14. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: Advances in Neural Information Processing Systems, pp. 700–708 (2017)

15. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. In: Advances in Neural Information Processing Systems, pp. 469–477 (2016)
16. Long, M., Cao, Y., Wang, J., Jordan, M.: Learning transferable features with deep adaptation networks. In: International Conference on Machine Learning, pp. 97–105 (2015)
17. Long, M., Wang, J., Jordan, M.I.: Deep transfer learning with joint adaptation networks. In: ICML (2017)
18. Luc, P., Couprie, C., Chintala, S., Verbeek, J.: Semantic segmentation using adversarial networks. In: NIPS Workshop on Adversarial Training (2016)
19. Russo, P., Carlucci, F.M., Tommasi, T., Caputo, B.: From source to target and back: symmetric bi-directional adaptive GAN. arXiv preprint arXiv:1705.08824 (2017)
20. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-CAM: visual explanations from deep networks via gradient-based localization. In: ICCV, pp. 618–626 (2017)
21. Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., Webb, R.: Learning from simulated and unsupervised images through adversarial training. In: CVPR (2017)
22. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: visualising image classification models and saliency maps. arXiv preprint arXiv:1312.6034 (2013)
23. Sun, B., Saenko, K.: Deep CORAL: correlation alignment for deep domain adaptation. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9915, pp. 443–450. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49409-8_35
24. Tzeng, E., Hoffman, J., Darrell, T., Saenko, K.: Simultaneous deep transfer across domains and tasks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4068–4076 (2015)
25. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: Computer Vision and Pattern Recognition (CVPR) (2017)
26. Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., Darrell, T.: Deep domain confusion: maximizing for domain invariance. arXiv preprint arXiv:1412.3474 (2014)
27. Wei, Y., Feng, J., Liang, X., Cheng, M.M., Zhao, Y., Yan, S.: Object region mining with adversarial erasing: a simple classification to semantic segmentation approach. In: IEEE CVPR (2017)
28. Zagoruyko, S., Komodakis, N.: Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer. In: ICLR (2017)
29. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 818–833. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10590-1_53
30. Zhang, X., Wei, Y., Feng, J., Yang, Y., Huang, T.: Adversarial complementary learning for weakly supervised object localization. In: IEEE CVPR (2018)
31. Zheng, Z., Zheng, L., Yang, Y.: Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In: Proceedings of the IEEE International Conference on Computer Vision (2017)
32. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2921–2929 (2016)

33. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networkss. In: 2017 IEEE International Conference on Computer Vision (ICCV) (2017)