Domain Attention Consistency for Multi-Source Domain Adaptation

Zhongying Deng[1, 2]
z.deng@surrey.ac.uk
Kaiyang Zhou[3]
k.zhou.vision@gmail.com
Yongxin Yang[1]
yongxin.yang@surrey.ac.uk
Tao Xiang[1, 2]
t.xiang@surrey.ac.uk

[1] University of Surrey
Guildford, UK
[2] iFlyTek-Surrey Joint Research Center on Artificial Intelligence
[3] Nanyang Technological University
Singapore

Abstract

Most existing multi-source domain adaptation (MSDA) methods minimize the distance between multiple source-target domain pairs via feature distribution alignment, an approach borrowed from the single source setting. However, with diverse source domains, aligning pairwise feature distributions is challenging and could even be counter-productive for MSDA. In this paper, we introduce a novel approach: transferable attribute learning. The motivation is simple: although different domains can have drastically different visual appearances, they contain the same set of classes characterized by the same set of attributes; an MSDA model thus should focus on learning the most transferable attributes for the target domain. Adopting this approach, we propose a domain attention consistency network, dubbed DAC-Net. The key design is a feature channel attention module, which aims to identify transferable features (attributes). Importantly, the attention module is supervised by a consistency loss, which is imposed on the distributions of channel attention weights between source and target domains. Moreover, to facilitate discriminative feature learning on the target data, we combine pseudo-labeling with a class compactness loss to minimize the distance between the target features and the classifier's weight vectors. Extensive experiments on three MSDA benchmarks show that our DAC-Net achieves new state of the art performance on all of them.

1 Introduction

The domain shift problem has been one of the main obstacles for large-scale deployment of machine learning systems in real-world applications [22, 43]. This is because in practice, we often need to apply a trained model to a new target environment where the test data follow a different distribution from the training data. As a result, the performance of the model typically drops significantly. This problem has persisted in the deep learning era, even when deep convolutional neural networks (CNNs) have demonstrated great successes in solving many recognition tasks [26, 43]. As a key solution to overcome the domain shift problem, unsupervised domain adaptation (UDA) has been extensively studied [2, 6, 7, 13, 19, 27, 42]. UDA aims to transfer the knowledge learned from one or multiple labeled source domains to a target domain in which only unlabeled data are given for model adaptation.

Early UDA work has been focused on the single-source setting [20, 27]. However, in real world, the source training data can often be collected from multiple domains (see Figure 1). This leads to a new UDA setting known as multi-source domain adaptation (MSDA), which has received increasing attention in recent years [23, 32, 37, 42]. Most existing MSDA methods are based on aligning the feature distribution of the unlabeled target domain data with those of the source domains. Feature alignment has been widely used for tackling domain adaptation [4] and is adopted by most single-source UDA methods [7, 9, 30]. However, this approach seems to be much less successful when applied to MSDA. This is not surprising: from Figure 1, it is evident that objects of the same class can have drastically different appearances across different domains. Aligning the feature distribution of a target domain to all the source domains requires a set of features that are completely domain-invariant. This is extremely difficult to achieve; and forcing it can be counter-productive—it has been shown that enforcing such an alignment across multiple source domains can lead to performance inferior to that of using a single source domain [23].

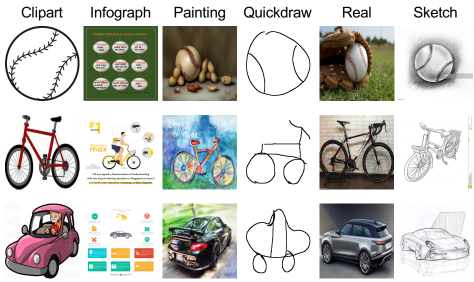

Figure 1: Example images from the MSDA benchmark DomainNet [23]. Each row contains object images of the same class but from different domains.

In this work, we introduce a new approach based on *transferable attribute learning* as an alternative to the existing feature distribution alignment based approach to MSDA. The motivation is simple: although different domains can have drastically different visual appearances, they contain the same set of classes, which can be explained using the same set of attributes. For example, Figure 1 (mid-row) shows that though a bicycle can be depicted in very different ways across the six domains (e.g., in different image styles), it always consists of wheels, frame, seat, handle bar, etc. Some of these attributes are even shared across classes (e.g., cars also have wheels). Therefore, we argue that an MSDA model should focus on learning from source domains the most transferable attributes for the target domain, which can be achieved by enforcing consistency on attributes used by different domains.

To realize transferable attribute learning, we propose a novel domain attention consistency network, dubbed DAC-Net. We follow the conventional model design adopted in most papers [23, 38], which consists of a feature embedding network and a classifier (a softmax-activated fully-connected layer) shared by both the source and target domains. However, instead of aligning feature distributions via some distance metrics, we propose to enforce domain attention consistency to identify transferable attributes, each represented by a CNN feature channel. To that end, we first construct a feature channel attention module to encourage the DAC-Net to use a small set of features (latent attributes) to represent each image. Then, to ensure these attributes to be transferable, a novel domain attention consistency loss is introduced, which minimizes the distribution divergence of channel attention weights between each pair of source and target domains. To facilitate discriminate feature learning on the target data, we further combine pseudo-labeling [15, 29] and a class compactness loss to minimize the distance between the target features and the classifier's weight vectors.

Our contributions are summarized as follows: **(1)** We propose a new transferable attribute learning based approach to tackle MSDA. The main idea is to learn, in each source domain, the most transferable attributes/features for the target domain. **(2)** We propose a novel domain attention consistency network (DAC-Net), which aims to align the distributions of channel-wise attention weights in each pair of source-target domains for learning transferable latent attributes. **(3)** To facilitate discriminative feature learning, we combine pseudo-labeling with a class compactness loss to pull together the target features and the classifier's weight vectors. **(4)** Extensive experiments on three MSDA datasets, including DomainNet [23], Digit-Five [6, 14, 21] and PACS [16], show that DAC outperforms the state of the art on all datasets, often by significant margins (e.g., 3.8% on the largest DomainNet).

2 Related Work

Single-source domain adaptation has been extensively researched. The main stream of domain adaptation methods has been devoted to reducing the distribution mismatch between source and target domains, mostly at the feature level [3, 7, 9, 18]. Direct distance metrics like maximum mean discrepancy (MMD) [9] and its kernelized version have been used in [9, 18] for distribution divergence minimization. Inspired by generative adversarial network (GAN) [8], adversarial learning has been used to align the feature distributions between source and target domains [7, 30]. Recent work has further taken into account the class information, and focused on class-wise feature alignment across domains by using bi-classifiers [20, 27] or aligning class centroids [12, 36].

More related to our work are attention alignment-based methods [11, 33]. In [11], the spatial attentions summarized across channels between each source domain and the translated pseudo-target domain (via CycleGAN [44]) are aligned. In [33], transferable image regions are identified based on adversarial networks. Different from these methods, our design of DAC-Net aims to identify the most transferable attributes (feature channels) by aligning the *distributions of channel attentions* between each pair of source and target domains.

Multi-source domain adaptation (MSDA) assumes access to multi-source data, compared with the single-source setting. Most existing MSDA methods are still based on feature alignment. Xu et al. [37] developed deep cocktail network (DCTN), which extends the domain-adversarial learning [7, 30] by learning a domain discriminator for each source-target pair. Li et al. [17] chose a relevant subset of each domain to apply feature alignment. Zhu et al. [45] proposed to align each source domain's distributions with that of target domain in multiple domain-specific feature spaces. Peng et al. [23] introduced M^3SDA, which minimizes moment-based distribution distances between each pair of source-target domains, as well as between each source-source pair. To facilitate feature alignment, Peng et al. [24] measured domain similarity by using a DOMAIN2VEC model to output vectorial representation for each domain. Zhou et al. [42] leveraged complementary information from multiple domain-specific classifiers to form an ensemble for the target domain. Pernes et al. [25] weighed the importance of each source domain for feature alignment. Yang et al. [38] developed curriculum manager for source selection (CMSS), which aims to learn which source domains are more suitable to be aligned with the target domain. Wang et al. [32] investigated interactions between different domains and developed a knowledge graph-based method called LtC-MSDA to promote information propagation from source domains to the target one.

Our DAC-Net differs from the existing MSDA methods in that no feature distribution alignment is attempted. Instead, we first introduce a channel feature attention module to encourage the learned features to capture a set of domain-transferable latent attributes. Then we design a consistency loss to minimize the divergence between the distributions of channel attention weights of each source-target domain pair. This is a much softer constraint than feature distribution alignment, and it is also much more effective (see Table 1).

Attention mechanism was initially introduced to focus on specific words in one language when translating a word in the other language [1]. In computer vision, attention has been used for CNN architecture design. Hu et al. [10] investigated attention from the channel dimension instead of the spatial dimension. They designed a squeeze-and-excitation network (SENet), which introduces a light-weight module that produces channel-wise attention values for a CNN layer. Woo et al. [35] further developed convolutional block attention mechanism (CBAM), which combines spatial attention [31] with channel attention [10]. In this work, for the first time, we exploit attention for *transferable attribute learning* for domain adaptation—a channel attention network is learned to attend to features that are transferable between source domains and the target domain, supervised by a novel DAC loss.

3 Methodology

3.1 Problem Formulation

In multi-source domain adaptation (MSDA), we are provided with labeled source data from K different domains, $\{\mathcal{S}_1, ..., \mathcal{S}_K\}$. The training data from the k-th source domain are denoted by $\mathcal{S}_k = \{(x_i^{\mathcal{S}_k}, y_i^{\mathcal{S}_k})\}_{i=1}^{N_{\mathcal{S}_k}}$ where x and y denote data (image) and label respectively. We also have access to unlabeled data from the target domain, $\mathcal{T} = \{x_i^{\mathcal{T}}\}_{i=1}^{N_{\mathcal{T}}}$. In this paper, we focus on image classification problems and assume a shared label space for the source and target domains. The goal is to train a classification model leveraging $\{\mathcal{S}_k\}_{k=1}^{K}$ and $\mathcal{T}$ so that the model can work well on an unseen test set in the target domain.

3.2 Domain Attention Consistency

To address MSDA, we propose a *domain attention consistency* network (DAC-Net). The motivation behind DAC-Net is to learn to attend to features that are transferable between multiple source domains and the target domain. Each feature (a CNN feature channel) represents a particular attribute. Therefore, in essence we aim to identify attributes that are transferable to the target domain.

The architecture of our DAC-Net is illustrated in Figure 2. It adopts a common architecture design used by existing MSDA models [23, 38]. Concretely, the model consists of a feature embedding CNN sub-network followed by a classification layer, both of which are shared across the source and target domains. To encourage the feature embedding CNN to learn a set of features that correspond to transferable latent attributes, we introduce a channel attention module inserted into different layers of the embedding network. With attention modeling for each image, DAC-Net is encouraged to use a subset of the feature channels to explain the image content, therefore facilitating the discovery of transferable latent attributes.

However, without proper supervision, simply inserting attention modules to a CNN network would not help knowledge transfer from source domains to the target (see Table 2, #3 vs. #2). We therefore propose a novel domain attention consistency (DAC) loss, which minimizes the distance between the *distributions* of attention weights used by source domains and the target domain. Below we detail the design of the attention module and the DAC loss.

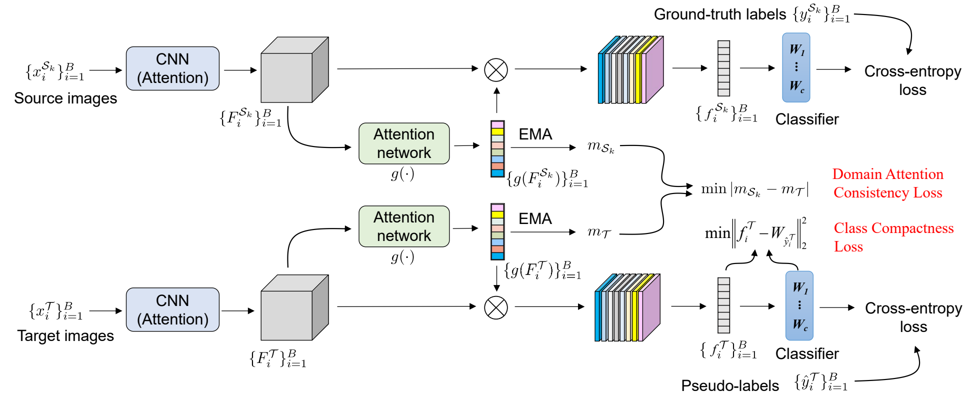

Figure 2: Overview of our DAC-Net, designed to attend to features that are transferable from multiple source domains to the target domain. This is achieved by optimizing a domain attention consistency loss that minimizes the ℓ_1 distance between the exponential moving average (EMA) of attention weights of each source domain ($m_{\mathcal{S}_k}$) and that of the target domain ($m_{\mathcal{T}}$). In implementation, we apply the attention network to multiple layers in a CNN. All the parameters are shared across domains.

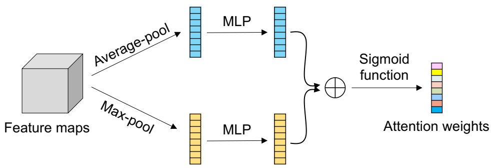

Figure 3: Architecture of our attention module

Attention module. Let $F \in \mathbb{R}^{C\times H\times W}$ denote feature maps extracted by a CNN, where C, H and W denote channel depth, height and width, respectively. The attention network $g(\cdot)$ takes as input F and produces a vector of attention weights spanning the channel dimension, $g(F) \in \mathbb{R}^C$. To make $g(\cdot)$ light-weight, we follow the design of CBAM [35] when constructing $g(\cdot)$. The architecture is detailed in Figure 3. The two branches share the same multi-layer perceptron (MLP), which consists of two fully connected layers with the same dimension for input and output. Notably, the hidden dimension in the MLP is reduced from C to $\frac{C}{r}$, where r is a reduction ratio (fixed to 16), to reduce parameter overhead.

Domain attention consistency loss. As shown in Figure 2, we first extract feature maps $\{F_i^{\mathcal{S}_k}\}_{i=1}^B$ and $\{F_i^{\mathcal{T}}\}_{i=1}^B$ from the k-th source domain images $\{x_i^{\mathcal{S}_k}\}_{i=1}^B$ and the target domain images $\{x_i^{\mathcal{T}}\}_{i=1}^B$, respectively. Here B denotes the batch size. The feature maps are then forwarded to the attention module $g(\cdot)$. In this work, we simply use the attention weight vector $g(F)$ averaged in each domain to represent the domain-level attention distribution. This average could be easily obtained at each mini-batch, but it will be an inaccurate measure of the domain-wise attribute attention statistics. We thus use exponential moving average (EMA) over mini-batches. Specifically, the EMA attention weights are computed as

$$m_{\mathcal{S}_k} = \alpha m_{\mathcal{S}_k} + (1-\alpha)\frac{1}{B}\sum_{i=1}^{B} g(F_i^{\mathcal{S}_k}), \tag{1}$$

$$m_{\mathcal{T}} = \alpha m_{\mathcal{T}} + (1-\alpha)\frac{1}{B}\sum_{i=1}^{B} g(F_i^{\mathcal{T}}), \tag{2}$$

where α is fixed to 0.999. Using EMA results in a more accurate estimation of the mean attention weights of the entire population (within each domain), while adding negligible computational overhead.

The domain attention consistency loss is computed as the ℓ_1 distance between $m_{\mathcal{S}_k}$ and $m_{\mathcal{T}}$. This is done for each pair of source and target domains. Formally, the loss is defined as

$$L_d = \frac{1}{K}\sum_{k=1}^{K} |m_{\mathcal{S}_k} - m_{\mathcal{T}}|. \tag{3}$$

We have also tried alternative distance measures such as MMD but found that the EMA-based ℓ_1 distance works better (see Table 2, #8 vs. #4).

3.3 Discriminative Feature Learning

The DAC loss in Eq. (3) is designed for learning transferable features from multiple source domains. Here we turn to the discriminative feature learning part for classification tasks.

Supervised learning for labeled source data. We use the cross-entropy loss to exploit labeled source data for learning discriminative features:

$$L_s = -\frac{1}{KB}\sum_{k=1}^{K}\sum_{i=1}^{B} \log p^{\mathcal{S}_k}_{i,y_i^{\mathcal{S}_k}}, \tag{4}$$

where $p^{\mathcal{S}_k}_{i,y_i^{\mathcal{S}_k}}$ means the predicted probability on the $y_i^{\mathcal{S}_k}$-th class (the ground truth) for $x_i^{\mathcal{S}_k}$.

Pseudo-labeling for unlabeled target data. To overcome the absence of labels for the target data, we resort to pseudo-labeling—a widely used technique in semi-supervised learning (SSL) [6, 15, 29]. Specifically, we follow the recently proposed FixMatch [29] but use the pseudo labels for MSDA rather than SSL. Given a target image, its pseudo-label is obtained by feeding the weakly augmented version of the image to the CNN model and picking the predicted class $\hat{y}_i^{\mathcal{T}}$ that has the maximum probability. A threshold $\tau = 0.95$ is used to filter out low-confidence predictions. The cross-entropy loss is then imposed on the model's output for the strongly augmented version of the image, defined as

$$L_t = -\frac{1}{B}\sum_{i=1}^{B} \mathbb{1}(q(\hat{y}_i^{\mathcal{T}}) \geq \tau) \log p^{\mathcal{T}}_{i,\hat{y}_i^{\mathcal{T}}}, \tag{5}$$

where $q(\hat{y}_i^{\mathcal{T}})$ is the predicted probability on pseudo-label $\hat{y}_i^{\mathcal{T}}$, and $\mathbb{1}(\cdot)$ the indicator function.

Enforcing class compactness on unlabeled target data. To further promote discriminative feature learning on the target data, we design a class compactness loss to encourage the target features to be close to the corresponding classification weight vectors, which can be seen as class prototypes [28]. Let W_j be the weight vector for class j in the last fully-connected layer, and $f_i^{\mathcal{T}}$ the features of $x_i^{\mathcal{T}}$, the class compactness loss is formulated as

$$L_c = \frac{1}{B}\sum_{i=1}^{B} \mathbb{1}(q(\hat{y}_i^{\mathcal{T}}) \geq \tau) ||f_i^{\mathcal{T}} - W_{\hat{y}_i^{\mathcal{T}}}||_2^2. \tag{6}$$

3.4 Training

For training the classification CNN model, we combine the losses in Eqs. (3), (4), (5) and (6):

$$L = L_s + L_t + \lambda_c L_c + \lambda_d L_d, \tag{7}$$

where λ_c and λ_d are hyper-parameters. The final CNN model trained with Eq. (7) is called domain attention consistency network, or DAC-Net.

Table 1: Results on three MSDA benchmark datasets where our DAC-Net achieves state-of-the-art performance on all datasets, with a clear margin over other competitors.

(a) DomainNet.

Methods	Clipart	Infograph	Painting	Quickdraw	Real	Sketch	Avg
Source-only [23]	47.6±0.52	13.0±0.41	38.1±0.45	13.3±0.39	51.9±0.85	33.7±0.54	32.9
DANN [7]	45.5±0.59	13.1±0.72	37.0±0.69	13.2±0.77	48.9±0.65	31.8±0.62	32.6
DCTN [37]	48.6±0.73	23.5±0.59	48.8±0.63	7.2±0.46	53.5±0.56	47.3±0.47	38.2
MCD [27]	54.3±0.64	22.1±0.70	45.7±0.63	7.6±0.49	58.4±0.65	43.5±0.57	38.5
M^3SDA [23]	58.6±0.53	26.0±0.89	52.3±0.55	6.3±0.58	62.7±0.51	49.5±0.76	42.6
CMSS [38]	64.2±0.18	28.0±0.20	53.6±0.39	16.0±0.12	63.4±0.21	53.8±0.35	46.5
LtC-MSDA [32]	63.1±0.50	**28.7**±0.70	56.1±0.50	16.3±0.50	66.1±0.60	53.8±0.60	47.4
DAC-Net (*ours*)	**72.5**±0.04	27.6±0.10	**57.8**±0.06	**23.0**±0.14	**66.7**±0.10	**59.5**±0.12	**51.2**

(b) Digit-Five.

Methods	MNIST	USPS	MNIST-M	SVHN	Synthetic	Avg
Source-only [38]	92.3±0.91	90.7±0.54	63.7±0.83	71.5±0.75	83.4±0.79	80.3
DANN [7]	97.9±0.83	93.4±0.79	70.8±0.94	68.5±0.85	87.3±0.68	83.6
DCTN [37]	96.2±0.80	92.8±0.30	70.5±1.20	77.6±0.40	86.8±0.80	84.8
MCD [27]	96.2±0.81	95.3±0.74	72.5±0.67	78.8±0.78	87.4±0.65	86.1
M^3SDA [23]	98.4±0.68	96.1±0.81	72.8±1.13	81.3±0.86	89.6±0.56	87.6
CMSS [38]	99.0±0.08	97.7±0.13	75.3±0.57	88.4±0.54	93.7±0.21	90.8
LtC-MSDA [32]	99.0±0.40	98.3±0.40	85.6±0.80	83.2±0.60	93.0±0.50	91.8
DAC-Net (*ours*)	**99.2**±0.03	**98.7**±0.11	**86.0**±0.44	**91.6**±0.16	**97.1**±0.18	**94.5**

(c) PACS.

Methods	ArtPainting	Cartoon	Sketch	Photo	Avg
Source-only	81.22	78.54	72.54	95.45	81.94
MDAN [40]	83.54	82.34	72.42	92.91	82.80
DCTN [37]	84.67	86.72	71.84	95.60	84.71
M^3SDA [23]	84.20	85.68	74.62	94.47	84.74
MDDA [41]	86.73	86.24	77.56	93.89	86.11
LtC-MSDA [32]	90.19	90.47	81.53	97.23	89.85
DAC-Net (*ours*)	**91.39**	**91.39**	**84.97**	**97.93**	**91.42**

4 Experiments

4.1 Experimental Setting

We apply the attention module to multiple layers in our DAC-Net: on Digit-Five, the attention module is applied after the 2nd and 3rd convolution layers; on PACS and DomainNet, where the ResNet architecture is used, we apply the attention module after the `conv4_x` and `conv5_x` blocks (i.e. last two residual blocks). We will evaluate this design choice later. Throughout the experiments, λ_d is set to 0.3 and λ_c is 0.1 (unless otherwise specified).

More settings such as the datasets, protocols, and other training details can be found in the Supplementary Material. The code will be available at https://github.com/Zhongying-Deng/DAC-Net.

4.2 Main Results

In this section, we compare our DAC-Net with the current state of the art on three MSDA benchmark datasets, namely DomainNet, Digit-Five and PACS. The results are shown in Table 1. Below we discuss the results in detail.

DomainNet is the most challenging dataset among the three due to its large scale. Among the compared methods, the most related to ours are those based on the idea of domain alignment, including M^{3}SDA and DCTN. In particular, M^{3}SDA minimizes the moment distance between each pair of source-target domains and each pair of source-source domains, while DCTN applies adversarial learning (similar to that used by DANN) to align the feature distribution in each pair of source-target domains. DAC-Net significantly outperforms both M^{3}SDA and DCTN with a significant margin of more than 8.6%. This improvement demonstrates that aligning attention weights for identifying transferable features is much more useful than aligning feature distributions for MSDA. Compared with the latest methods, i.e. CMSS and LtC-MSDA (pseudo-labeling [39] is also adopted in LtC-MSDA when estimating class prototypes for the target domain), DAC-Net is also clearly better—with more than 3.8% improvement over them. It is noteworthy that the biggest improvements over CMSS and LtC-MSDA are obtained on Quickdraw and Sketch, which are drastically different from the other domains where images are mostly colorized with rich textures (see Figure 1). This suggests that the learned transferable latent attributes are more robust against large domain shift.

On Infograph domain, our method fails to beat some other methods [32, 38]. This can explained by the existence of irrelevant content in Infograph's images, as shown in Figure 1, which may result in very noisy pseudo labels for discriminative feature learning. To improve the quality of pseudo labels, some regularization methods [46] can be introduced.

Digit-Five and PACS. It is clear that DAC-Net achieves the best performance on all target domains on these two datasets, which further justifies our design of domain attention consistency. The other conclusions drawn above also hold: significant gaps exist between DAC-Net and the most related M^{3}SDA and DCTN; the margins over CMSS and LtC-MSDA are also clear, particularly on those challenging domains (over 3% improvement) like SVHN in Digit-Five and Sketch in PACS.

4.3 Ablation Study

In this section, we conduct ablation studies on PACS to evaluate the main components in our DAC-Net. Note that all variants are trained using exactly the same training parameters as DAC-Net for fair comparison. The results are reported in Table 2. Overall, our final model, DAC-Net, brings the largest improvement of 9.48% over the source-only baseline.

Table 2: Ablation study on PACS. $\mathcal{A}$: attention network. Δ: accuracy difference versus the source-only baseline.

#	Methods	Avg	Δ
1	L_s	81.94	-
2	+ L_t	88.86	+6.92
3	+ L_t + $\mathcal{A}$	88.34	+6.40
4	+ L_t + $\mathcal{A}$ + L_d	90.79	+8.85
5	+ L_t + $\mathcal{A}$ + L_d + L_c (final model)	**91.42**	**+9.48**
6	+ L_t + $\mathcal{A}$ + L_d + CenterLoss	87.52	+5.58
7	+ L_t + $\mathcal{A}$ + L_d w/o EMA	88.61	+6.67
8	+ L_t + $\mathcal{A}$ + MMD-based L_d	90.23	+8.29

Significance of L_d. We first apply the attention network to the pseudo-labeling baseline (#2), and compare the results to see whether the attention network brings any improvement. From the comparison of #2 vs. #3, we observe that adding the attention network even brings an adverse effect—the accuracy drops from 88.86% to 88.34%. However, by incorporating our domain attention consistency (DAC) loss L_d in model training, the performance is significantly improved from 88.34% to 90.79% (#3 vs. #4). The results confirm that the DAC is essential for learning transferable features.

Importance of the class compactness loss. Our class compactness loss in Eq. (6) essentially pulls together the target features and the classifier's weight vectors to facilitate discriminative target feature learning. By comparing #5 with #4, we observe that the accuracy is improved by 0.63% with the class compactness loss. We also compare with the center loss [34], which enforces class compactness using parameterized class centers. Similar to our class compactness loss, we discard low-confidence pseudo-labels when updating the parameterized centers for the center loss. By replacing our class compactness loss with the center loss, i.e. #5 vs. #6, we observe a sharp decrease in accuracy (-3.9%). This result suggests that parameterized class centers cannot be properly learned, possibly close to the decision boundary due to the noisy target pseudo-label; as such, one should rely more on the classification weight vectors to enforce class compactness—the weight vectors updated/dominated by labeled source data can be less noisy and probably far away from the decision boundary. More experimental results are provided in the Supplementary Material.

Effectiveness of EMA in L_d. We train a variant of DAC by removing the EMA part when computing L_d so that the mean attention weights are computed based merely on the current mini-batches, i.e. $m_{\mathcal{D}} = \frac{1}{B}\sum_{i=1}^{B} g(F_i^{\mathcal{D}})$. As a result, the performance drops from 90.79% (#4) to 88.61% (#7). This result justifies the use of EMA statistics for computing the DAC loss.

Alternative distance function for L_d. In Eq. (3), the distance is measured based on the ℓ_1 distance between the EMA of domain attention weights. Here we try an alternative distance function based on maximum mean discrepancy (MMD) [9]. Comparing with the EMA version (#4), we observe a decrease in performance for the MMD version (#8).

Sensitivity of λ_d and λ_c. Recall that λ_d and λ_c control the weights on L_d (domain attention consistency loss) and L_c (class compactness loss) in Eq. (7), respectively. To evaluate how sensitive the performance is to λ_d and λ_c, we first set λ_c to 0 and linearly increase λ_d from 0.1 to 1, which covers a wide value range. The results are shown in Figure 4. It can be seen that the accuracy is generally stable with different values for λ_d (blue solid line), with the best performance achieved at $\lambda_d = 0.3$. Then we fix λ_d to 0.3 and adjust λ_c. The results (red dashed line) indicate that increasing λ_c seems to result in a (smooth) downward trend in performance, with $\lambda_c = 0.1$ being the best choice.

Where to apply the DAC loss? We evaluate four variants of DAC in the first four rows of Table 3 where domain attention consistency loss is applied after different numbers of residual block at different places. The findings are summarized as follows. 1) Applying the loss after the last two residual blocks gives the best performance. 2) Applying the loss to lower layers worsens the performance. This is expected: we aim to discover transferable latent attributes which are semantic/abstract concepts that only emerge in the top layers of a CNN.

Attention alignment vs. feature alignment. We compare our DAC based approach with the most popular feature distribution alignment (FDA) method in Table 3 (see the last row). With everything else identical including the attention network, for the FDA method, we apply the same domain consistency loss L_d on the final features (the features used by the classifier for classification) to directly align feature distributions. It is clear that FDA results in inferior performance to corresponding DAC variants (see Supplementary Material for more experimental results). This observation supports the main claim of the paper: instead of aligning feature distributions, aligning attribute attention weights is more effective for MSDA.

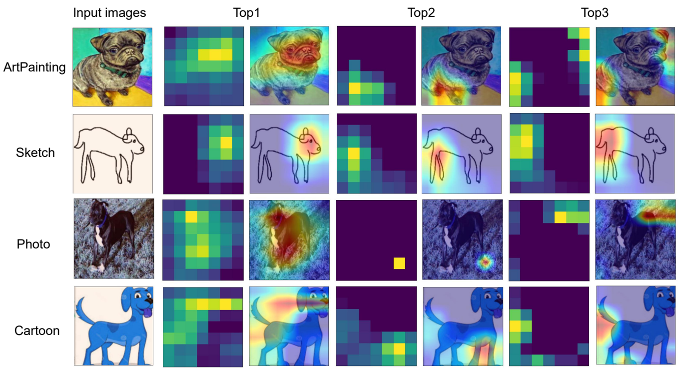

Figure 5: Attended feature maps of DAC-Net and their masked input images. We show the feature maps corresponding to the top-3 attention weights across a random testing subset of PACS, and project the feature maps to the input images. These feature maps or masked images are from the same class of dog but different domains.

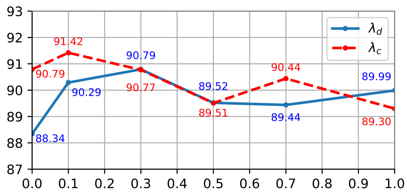

Figure 4: Sensitivity of λ_d and λ_c in Eq. (7).

Table 3: Ablation study on where to apply the DAC loss L_d on PACS. FDA: Feature distribution alignment.

Methods	Apply L_d after	**Avg**
DAC	Last residual block	90.48
	Last two residual blocks	**90.79**
	Last three residual blocks	88.46
	All four residual blocks	88.06
FDA	Only final features	89.27

4.4 Visualization

In this section, we provide visualization of attended feature maps to help understand why our DAC-Net works. We visualize examples of the top-3 attended feature maps and their corresponding masked images in Figure 5. We can see that the feature maps with high attentions focus on some semantic attributes, e.g. the head or leg of a dog, even though their appearance varies greatly across domains. These semantic attributes are discriminative features for classification, and more importantly are transferable across domains. The ability to discovering them thus underpins the good performance of our DAC-Net.

Visualizations on feature distributions can be found in the Supplementary Material.

5 Conclusion

In this paper, we introduced a novel DAC-Net to learn transferable latent attributes for MSDA. It incorporates an attention module and a domain attention consistency loss applied on the exponential moving average (EMA) of the attention weights of each source domain and that of the target domain. We also proposed a class compactness loss to pull together the target features and the classification weight vectors (class prototypes). Extensive experiments on three MSDA benchmark datasets demonstrated that our DAC-Net significantly outperforms the current state-of-the-art competitors.

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015.

[2] Mahsa Baktashmotlagh, Mehrtash T Harandi, Brian C Lovell, and Mathieu Salzmann. Unsupervised domain adaptation by domain invariant projection. In *ICCV*, 2013.

[3] Yogesh Balaji, Rama Chellappa, and Soheil Feizi. Normalized wasserstein for mixture distributions with applications in adversarial learning and domain adaptation. In *ICCV*, 2019.

[4] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *ML*, 2010.

[5] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*. 2019.

[6] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, 2015.

[7] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *JMLR*, 2016.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

[9] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *JMLR*, 2012.

[10] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018.

[11] Guoliang Kang, Liang Zheng, Yan Yan, and Yi Yang. Deep adversarial attention alignment for unsupervised domain adaptation: the benefit of target expectation maximization. In *ECCV*, 2018.

[12] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *CVPR*, 2019.

[13] Guoliang Kang, Lu Jiang, Yunchao Wei, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for single-and multi-source domain adaptation. *TPAMI*, 2020.

[14] Y Lecun and L Bottou. Gradient-based learning applied to document recognition. *IEEE*, 1998.

[15] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML Workshop*, 2013.

[16] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *CVPR*, 2017.

[17] Yitong Li, Michael Murias, Samantha Major, Geraldine Dawson, and David E Carlson. Extracting relationships by multi-domain matching. In *NeurIPS*, 2018.

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

[19] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *NeurIPS*, 2016.

[20] Zhihe Lu, Yongxin Yang, Xiatian Zhu, Cong Liu, Yi-Zhe Song, and Tao Xiang. Stochastic classifiers for unsupervised domain adaptation. In *CVPR*, 2020.

[21] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NeurIPS-W*, 2011.

[22] S.J. Pan and Q. Yang. A survey on transfer learning. *TKDE*, 2010.

[23] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *ICCV*, 2019.

[24] Xingchao Peng, Yichen Li, and Kate Saenko. Domain2vec: Domain embedding for unsupervised domain adaptation supplementary material. In *ECCV*, 2020.

[25] Diogo Pernes and Jaime S Cardoso. Tackling unsupervised multi-source domain adaptation with optimism and consistency. *arXiv preprint arXiv:2009.13939*, 2020.

[26] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *ICML*, 2019.

[27] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *CVPR*, 2018.

[28] Jake Snell, Kevin Swersky, and Richard S Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017.

[29] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*, 2020.

[30] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *CVPR*, 2017.

[31] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *CVPR*, 2017.

[32] Hang Wang, Minghao Xu, Bingbing Ni, and Wenjun Zhang. Learning to combine: Knowledge aggregation for multi-source domain adaptation. In *ECCV*, 2020.

[33] Ximei Wang, Liang Li, Weirui Ye, Mingsheng Long, and Jianmin Wang. Transferable attention for domain adaptation. In *AAAI*, 2019.

[34] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, 2016.

[35] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, 2018.

[36] Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *ICML*, 2018.

[37] Ruijia Xu, Ziliang Chen, Wangmeng Zuo, Junjie Yan, and Liang Lin. Deep cocktail network: Multi-source unsupervised domain adaptation with category shift. In *CVPR*, 2018.

[38] Luyu Yang, Yogesh Balaji, Ser-Nam Lim, and Abhinav Shrivastava. Curriculum manager for source selection in multi-source domain adaptation. In *ECCV*, 2020.

[39] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu. Collaborative and adversarial network for unsupervised domain adaptation. In *CVPR*, 2018.

[40] Han Zhao, Shanghang Zhang, Guanhang Wu, José MF Moura, Joao P Costeira, and Geoffrey J Gordon. Adversarial multiple source domain adaptation. In *NeurIPS*, 2018.

[41] Sicheng Zhao, Guangzhi Wang, Shanghang Zhang, Yang Gu, Yaxian Li, Zhichao Song, Pengfei Xu, Runbo Hu, Hua Chai, and Kurt Keutzer. Multi-source distilling domain adaptation. In *AAAI*, 2020.

[42] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain adaptive ensemble learning. *arXiv preprint arXiv:2003.07325*, 2020.

[43] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *arXiv preprint arXiv:2103.02503*, 2021.

[44] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017.

[45] Yongchun Zhu, Fuzhen Zhuang, and Deqing Wang. Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources. In *AAAI*, 2019.

[46] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. Confidence regularized self-training. In *ICCV*, 2019.