Neurocomputing 499 (2022) 1–10

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Attention-based cross-layer domain alignment for unsupervised domain adaptation

Xu Ma [a], Junkun Yuan [a], Yen-wei Chen [b], Ruofeng Tong [a], Lanfen Lin [a,*]

[a] College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China
[b] College of Information Science and Engineering, Ritsumeikan University, Kusatsu 5250058, Japan

ARTICLE INFO

Article history:
Received 15 September 2021
Revised 25 March 2022
Accepted 17 April 2022
Available online 2 May 2022
Communicated by Zidong Wang

Keywords:
Unsupervised domain adaptation
Cross-layer semantic alignment
Attention
Visual recognition

ABSTRACT

Unsupervised domain adaptation (UDA) aims to learn transferable knowledge from a labeled source domain and adapts a trained model to an unlabeled target domain. To bridge the gap between source and target domains, one prevailing strategy is to minimize the distribution discrepancy by aligning their semantic features extracted by deep models. The existing alignment-based methods mainly focus on reducing domain divergence in the same model layer. However, the same level of semantic information could distribute across model layers due to the domain shifts. To further boost model adaptation performance, we propose a novel method called Attention-based Cross-layer Domain Alignment (ACDA), which captures the semantic relationship between the source and target domains across model layers and calibrates each level of semantic information automatically through a dynamic attention mechanism. An elaborate attention mechanism is designed to reweight each cross-layer pair based on their semantic similarity for precise domain alignment, effectively matching each level of semantic information during model adaptation. Extensive experiments on multiple benchmark datasets consistently show that the proposed method ACDA yields state-of-the-art performance.

1. Introduction

Deep learning has achieved remarkable progress in diverse areas of computer vision like visual recognition [1] and many more. The training of deep learning models heavily relies on the independent and identical distributed (i.i.d.) assuming that training and test datasets should have the same statistical distribution. However, in real world applications, usually models trained on one dataset face test datasets which have totally different and distinct data distributions. This results in severe performance degradation because the features of the datasets are completely different from each other. Unsupervised domain adaptation (UDA) [2–5] is introduced to tackle the distribution/domain shift problem by learning transferable knowledge from a labeled source (training) domain and adapting the model to an unlabeled target (test) domain.

One prevailing strategy for UDA is to minimize the distribution discrepancy of the source and target data by aligning their semantic features extracted by deep models. However, the existing alignment-based algorithms [6–9] mostly reduce domain divergence by matching semantic features of the source and target data in the same layer of model, while recent studies [10,11] have shown that the same level of semantic information could distribute across model layers due to domain shifts, which we call the semantic dislocation problem. This problem makes different levels of semantic information be mismatched during the same-layer feature alignment process in the previous methods, bringing negative transfer gain to model adaptation performance. In comparison, we propose to match the same level of semantic information of the source and target domains across model layers for precise domain alignment, as shown in Fig. 1.

In order to further facilitate accurate semantic alignment and minimize domain divergence, we propose a novel method called *Attention-based Cross-layer Domain Alignment (ACDA)*. This method captures and calibrates the semantic relationship between the source and target domains across the different layers of the model. Specifically, we first minimize divergence between each pair of the extracted semantic features of the source and target data. To calibrate each level of semantic information, a dynamic attention mechanism is designed to reweight the divergence minimization loss of each pair of the cross-layer on the basis of their semantic similarities. In this way, different levels of semantic information are aligned automatically, effectively minimizing the distribution discrepancy between the source and target domains and improving

* Corresponding author.
E-mail address: llf@zju.edu.cn (L. Lin).

https://doi.org/10.1016/j.neucom.2022.04.086
0925-2312/

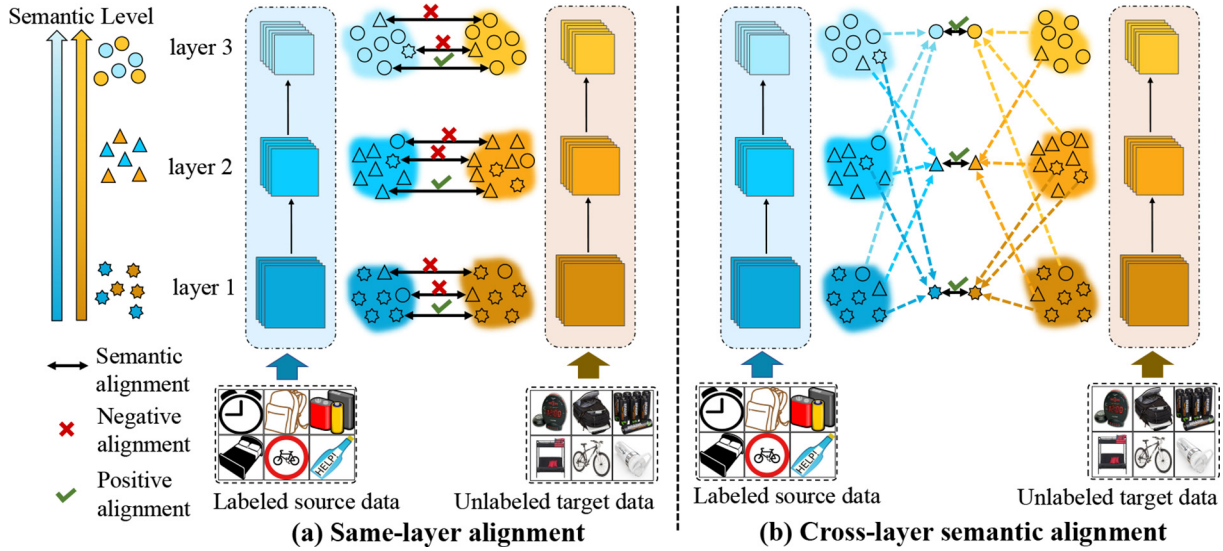

Fig. 1. Comparison between (a) same-layer alignment adopted by previous methods and (b) cross-layer semantic alignment introduced by this work. Instead of directly aligning same-layer features, we match the same level of semantic information across model layers of source and target to further facilitate domain divergence minimization and improve model adaptation performance.

model adaptation capability. Extensive experiments on various standard domain adaptation benchmark datasets, i.e., Office-31, Office-Home, ImageCLEF-DA, and VisDA-2017, show that the proposed method ACDA outperforms other state-of-the-art UDA methods.

In summary, this paper has the following contributions:

- We point out the semantic dislocation problem that each level of semantic information can be distributed across different layers of the model due to the domain shifts, which could bring negative transfer gain to previous methods with same-layer semantic feature alignment.
- In order to address the above problem, we propose a novel method called attention-based cross-layer domain alignment to match the same level of semantics by reweighting each cross-layer pair through a semantic similarity based dynamic attention mechanism.
- Extensive experiments show the superior performance of our method in comparison to other state-of-the-art UDA approaches on multiple standard benchmark datasets.

The rest of this paper is organized as follows. In Section 2, related works about unsupervised domain adaptation and attention mechanism are briefly introduced. In Section 3, the framework and algorithm of the proposed method are described. In Section 4, the results of experiments are reported and discussed. In Section 5, we conclude the investigation with a future research recommendation.

2. Related Work

2.1. Unsupervised Domain Adaptation

Unsupervised domain adaptation (UDA) [3–5,12–24] aims to adapt the model trained on a labeled source domain to an unlabeled target domain when there is distinct domain divergence. A series of UDA algorithms [25–30] have been proposed by employing an adversarial learning strategy where the semantic features of the source and target data are aligned for reducing domain divergence. For example, a representative framework DANN [31] utilizes the generator to extract domain-invariant semantic features that can fool the discriminator with a gradient reversal layer. Long et al. [25] extend this framework and reduce domain divergence by considering conditional probability distributions. Further, Shao et al.[32] use pixel-level adversarial adaptation as a constraint for feature-level adaptation, which can avoid the image quality degradation problem while mitigating the low-level domain variance. Yang et al.[33]. developed an adversarial network to learn graph-aligned representations with similar distributional structure in the source and target domains, which not only learns domain-invariant representations, but also retains structural information in each domain. However, the model adaptation performance of these methods rely on the carefully designed network structure and adversarial training process, which could be unstable and inefficient [34].

Directly minimizing domain divergence by aligning the semantic features of the source and target domains extracted by deep models is the usual way to address the domain shifts, much attention has been paid to this approach [6–9]. In these approaches the discrepancy of the semantic features of the source and target domains is reduced using the distance matrix, such as maximum mean discrepancy (MMD) [6] and multi-kernel MMD (MK-MMD) [7–9]. For example, Long et al. [6] propose to learn a generalizable model by matching the semantic feature embeddings of different domains. However, most of these previous alignment-based methods consider the semantic relationship in the same layer of the model, while each level of semantic information can be distributed across the layers due to the domain shifts. Recently, Joint adaptation network (JAN) [8] considers the semantic relationship in different layers, but this method can not effectively reweight the implemented cross-layer constraint, which results in an insufficient domain adaptation. In this paper, we boost model adaptation performance by exploring the relationship of cross-layer semantic information [10]. Specifically, we introduce an attention-based semantic matching method to automatically reweight the divergence minimization loss of each pair of cross-layer.

2.2. Attention mechanism

In recent years, attention mechanisms [35], especially self-attention [36], is widely adopted in various tasks of computer vision [37,38,12]. The self-attention mechanism learns a represen-

tation of a sequence by reweighting each position according to its corresponding similarity/importance. A generic process of the self-attention consists of the following steps: (1) Obtain semantic feature embeddings i.e., query, key, and value of the original features; (2) Calculate the similarity between query and key, and normalize it to obtain the weights; (3) Use the weights to synthesize the value. For example, [37] proposes to capture rich contextual dependencies for scene segmentation by obtaining attention in both spatial and channel dimensions. It first calculates the attention map of the original semantic features by performing the inner dot product of the features, then exploits the generated weights to synthesize the original semantic features. This emphasizes their important semantic information. Inspired by it, we design a dynamic attention mechanism to automatically capture the cross-layer domain-invariant semantic relationship based on the semantic similarity in both spatial and channel dimensions. The proposed attention mechanism precisely reduces domain divergence and effectively improves model adaptation towards new target domain..

Some attention-based DA works, such as [38] proposes to use attention for region-level and image-level context learning by exploring relationship in original images and the semantic features; [39] introduces a spatial attention pyramid network to capture context information at different scales; [40] puts forward generative attention adversarial classification network to allows a discriminator to discriminate the transferable regions; [41] proposes attention-based multi-source DA framework by considering domain correlations and alleviating effect of the dissimilar domains. However, these methods only consider the semantic features in the final model layer, while our work matches the same level of semantic information across model layers through the elaborate dynamic attention mechanism.

3. Method

In this section, we begin with the problem definition of unsupervised domain adaptation (UDA), and then introduce the details of the proposed method ACDA.

3.1. Problem Definition

Let the labeled source data be denoted as $\mathscr{D}_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$, where x_i^s is the i-th source sample, y_i^s is the corresponding class label, and n_s is the source data size. The unlabeled target data is denoted as $\mathscr{D}_t = \{x_i^t\}_{i=1}^{n_t}$, where x_i^t is the i-th target sample, and n_t is the target data size. In UDA setting, the source and target data are sampled from different distributions $p_s(x,y)$ and $p_t(x,y)$, respectively. The source and target domain shares the same label space $\mathscr{C} = \{1, 2, \cdots, K\}$ and we assume there are K classes in both domains. The goal of UDA is to train a predictive model with the labeled source data $\mathscr{D}_s$ and the unlabeled target data $\mathscr{D}_t$ for improving the performance of the model on the target domain.

3.2. Cross-Layer Semantic Alignment

Since each level of semantic information can be distributed across the layers of a model due to the domain shifts, hence we propose to match the same level semantic information of the source and target domains across the layers of the model for precise domain alignment. The framework and algorithm of the proposed ACDA method is shown in Fig. 2 and Algorithm 1. The feature extractor G extracts different levels of semantic features of the source and target data, and the classifier C uses the high-level information of the features (i.e., top layers of a model) for object classification. Different from directly aligning semantic features within the same layers of the model, we learn the cross-layer semantic relationship and reweight each divergence minimization loss of each cross-layer pair according to the semantic similarity calculated using a dynamic attention mechanism. The detail is given in the following section.

3.3. Model Pretraining

To initialize a discriminative model, we use the labeled source data to train the model F, i.e., $F = C \circ G$, where G and C are the feature extractor and classifier, respectively. The cross-entropy classification loss training objective for model F can be defined as:

$$\mathscr{L}_{ce} = -\mathbb{E}_{(x^s, y^s)\sim\mathscr{D}_s} \sum_{k=1}^{K} 1_{[k=y^s]} \log\left(C_k(G(x^s))\right), \tag{1}$$

where K is the number of classes, and C_k is k-dimensional of the output of the classifier C. We use $\mathscr{L}_{ce}$ to pretrain the model for initializing its discrimination capability.

3.4. Cross-Layer Alignment

After model pretraining, the initialized model can extract semantic features of source and target data. However, each level of semantic information can be distributed across layers because of the semantic dislocation problem. Therefore, we then propose to align cross-layer semantic features of the source and target domains.

Convolution-based projection. Since the original semantic features may have different sizes, we project the feature maps to match the size. Let the original semantic features of m layers extracted by the feature extractor G of the source and target are $q^s = \{q_1^s, q_2^s, \ldots, q_m^s\}$ and $q^t = \{q_1^t, q_2^t, \ldots, q_m^t\}$, respectively. Then we use a convolution-based projection to project all the semantic features to the same size. That is,

$$l_i^s = P_i^s(q_i^s), \quad l_i^t = P_i^t(q_i^t), i = 1, 2, \ldots, m \tag{2}$$

where P_i^s and P_i^t are the convolution mapping function of the i-th to last layer for the source and target domains, respectively (see details in experiment section). After the projection, all the semantic features are in the same size, which allows us to implement cross-layer semantic alignment.

Cross-layer semantic alignment. Since each level of semantic information could distribute across layers, we propose to align each pair of cross-layer semantic features. For simplicity, let the semantic features extracted from the target and source domain in the convolution layers (after projection) be l_i^t and l_j^s, respectively; and the final semantic features extracted from the target and source domains using fully-connected layer are f^t and f^s, respectively. For each cross-layer semantic feature pair (i,j) of target and source domains, we minimize the domain divergence with the distance matrix:

$$\begin{aligned}\mathrm{dist}(l_i^t, l_j^s) = &\frac{1}{b_s^2}\sum_{t=1}^{b_s}\sum_{u=1}^{b_s} k\left(l_{j,t}^s, l_{j,u}^s\right) + \frac{1}{b_t^2}\sum_{t=1}^{b_t}\sum_{u=1}^{b_t} k\left(l_{i,t}^t, l_{i,u}^t\right) \\ &- \frac{2}{b_s b_t}\sum_{t=1}^{b_s}\sum_{u=1}^{b_t} k\left(l_{j,t}^s, l_{i,u}^t\right),\end{aligned} \tag{3}$$

where $k(a_1, a_2) = \langle\phi(a_1), \phi(a_2))\rangle$ is a kernel function, b_s and b_t are batch-sizes of source and target data, respectively, respectively. A characteristic kernel $k(l_j^s, l_i^t) = \left\langle \phi(l_j^s), \phi(l_i^t)\right\rangle$ is defined as a convex combination of o positive semi-definite kernels $\{k_u\}$, i.e., $\mathscr{K} \triangleq \{k = \sum_{u=1}^{o}\beta_u k_u : \sum_{u=1}^{o}\beta_u = 1, \beta_u >= 0, \forall u\}$ [6,8]. By minimizing the cross-layer semantic feature pairs of source and target data, we

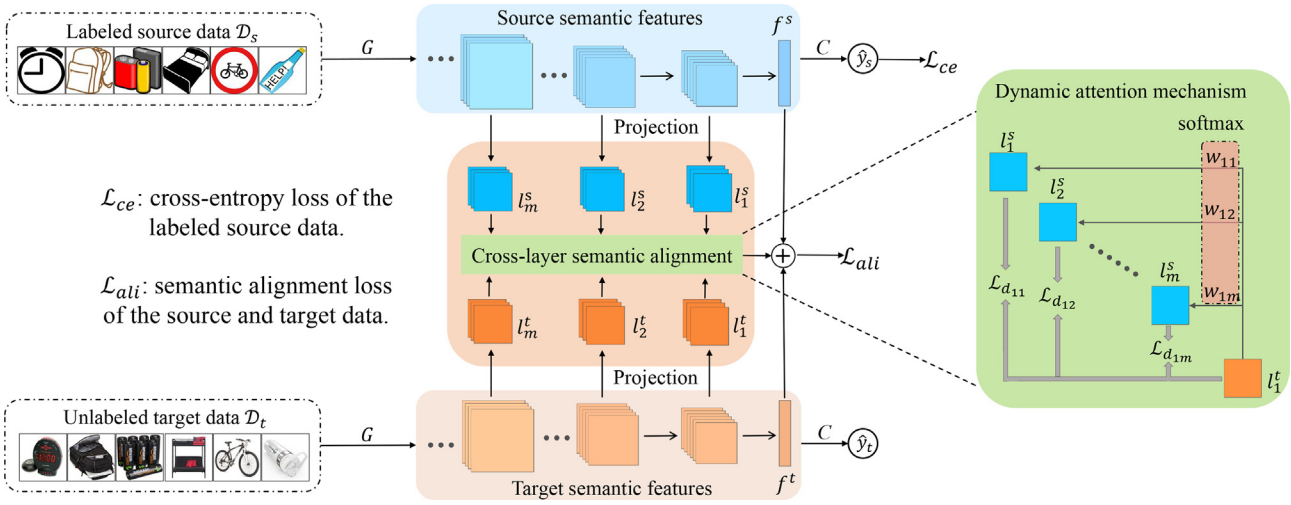

Fig. 2. The proposed Attention-based Cross-layer Domain Alignment (ACDA) framework. The feature extractor G extracts semantic features of the source and target data. After matching the size of feature through projection, we conduct cross-layer semantic alignment with a dynamic attention mechanism to reweight the divergence minimization loss of each cross-layer pair for precise domain alignment.

reduce the domain divergence between the source and target domains.

Attention allocation module. Since the same level of semantic information could be contained in different layers, we design a dynamic attention mechanism to automatically reweight the divergence minimization loss of each cross-layer pair. Specifically, we reweight each pair of semantic features according to their semantic similarity in both spacial and channel dimensions, that is,

$$w_{i,j} = \frac{1}{2}\frac{\exp\left[\text{avg}\left(r\left(l_i^t\right)\cdot r\left(l_j^s\right)^T\right)\right]}{\sum_{u=1}^{m}\exp\left[\text{avg}\left(r\left(l_i^t\right)\cdot r\left(l_u^s\right)^T\right)\right]} + \frac{1}{2} \times \frac{\exp\left[\text{avg}\left(r\left(l_i^t\right)^T\cdot r\left(l_j^s\right)\right)\right]}{\sum_{u=1}^{m}\exp\left[\text{avg}\left(r\left(l_i^t\right)^T\cdot r\left(l_u^s\right)\right)\right]} \tag{4}$$

where $r(\cdot)$ is a reshaping operation that maps semantic features with size $c \times h \times w$ to the size $c \times (h \times w)$. The *avg* operator shows the global average pooling operation. We take the average of each attention similarity matrices generating a real number value, which represents the average semantic similarity between each cross-layer pair. The first term and the second term of Eq. 4 illustrate the spatial and channel semantic relationships, respectively. We normalize the final similarity and obtain the weight $w_{i,j}$ for each cross-layer pair (i,j), which is used to reweight the cross-layer divergence minimization loss:

$$\mathcal{L}_{cross-ali} = \sum_i\sum_j w_{ij}\cdot \text{dist}\left(l_i^t, l_j^s\right), \tag{5}$$

where $\mathcal{L}_{cross-ali}$ is cross-layer semantic alignment loss of features. We also minimize the divergence of logits (semantic features in the final fully-connected layer from G), i.e., f^s and f^t, that is,

$$\mathcal{L}_{same-ali} = \text{dist}(f^s, f^t). \tag{6}$$

Finally, we integrate the cross-layer semantic alignment constraint of features in the convolution layers, i.e., $\mathcal{L}_{cross-ali}$, and the same-layer semantic alignment constraint of logits, i.e., $\mathcal{L}_{same-ali}$ into a unified semantic alignment loss $\mathcal{L}_{ali}$:

$$\mathcal{L}_{ali} = \delta\mathcal{L}_{cross-ali} + (1-\delta)\mathcal{L}_{same-ali}, \tag{7}$$

where δ is a trade-off hyper-parameter. Note that to further facilitate domain divergence minimization, we introduce label-conditioned cross-layer semantic alignment. Inspired by the recent work [42], we obtain pseudo labels of the unlabeled target data using k-means clustering and align the cross-layer semantic feature pairs in the same class.

3.5. Optimization

The optimization of our method consists of two steps as stated in Algorithm 1. The first step is to pretrain the model with the labeled source data by using objective loss function given in Eq. 1. The second step is to implement cross-layer semantic alignment by using the combination of supervised loss and alignment loss, that is,

$$\mathcal{L}_{all} = \mathcal{L}_{ce} + \lambda\cdot\mathcal{L}_{ali}, \tag{8}$$

where λ is a hyper-parameter to keep an appropriate balance between the cross-entropy loss $\mathcal{L}_{ce}$ and the cross-layer semantic alignment loss $\mathcal{L}_{ali}$. We also analyze the sensitivity of the hyper-parameters in the experiments.

4. Experiments

In this section, we evaluate the performance of our ACDA method by comparing it with state-of-the-art UDA methods on four publicly available datasets: Office-31, Office-Home, ImageCLEF-DA, and VisDA. We conduct analyses to provide insights on the design each component of our model.

4.1. Dataset

In this section, first, we introduce the datasets considered for our experiments. These datasets are Office-31, Office-Home, ImageCLEF-DA, and VisDA. Some example images are shown in the Fig. 3.

Office-31 is a popular dataset for domain adaptation, It has 4011 images with 31 classes, these images are collected from three different areas: 1. Amazon 'A': images downloaded from amazon.com, 2. Webcame 'W': images captured using web cameras, 3. DSLR 'D': images captured using Digital SLR cameras. These images contain different photographic settings and viewpoints. We

evaluate the domain adaptation tasks in 6 different settings: $A \rightarrow W, D \rightarrow W, W \rightarrow D, A \rightarrow D, D \rightarrow A$, and $W \rightarrow A$, in each pair of task, the former is used as the labeled source domain and the latter is used as the unlabeled target domain.

Office-Home is a well organized and more challenging dataset, which contains 15,500 images with 65 categories from 4 domains. In detail, Art (Ar) denotes artistic depictions for object images, Clipart (Cl) is the picture collection of clipart, Product (Pr) is object images with a clear background which is similar to Amazon category in Office-31 dataset, and Real-World (Rw) is object images collected with a regular camera. We use all possible combinations of source and target setting. We get 12 such combinations to perform the experiments.

ImageCLEF-DA is a benchmark dataset for ImageCLEF 2014 domain adaptation challenge, created by selecting common categories among the following three public datasets, namely, Caltrch-256 (C), ImageNet ILSVRC 2012 (I), and Pascal VOC 2012 (P). There are 50 images in each category and 600 images in each domain. We adopt six transfer tasks of domain adaptation: I→P, P→I, I→C, C→I, C→P and P→C.

VisDA is also a challenging dataset which has images from two different domains (i.e. simulated images to real images). It contains 152,397 training images and 55,388 validation images in 12 classes. We follow the training and testing protocols of [47,25]. The training of the models has been done using labeled source data and unlabeled target data. The model then test on the target data for unsupervised domain adaptation.

4.2. Baseline Methods

We compare our ACDA methods with the state-of-the-art unsupervised domain adaptation methods, i.e., GAKT [3], DRMEA [4], CDAN + TFLGM [5], DAN [6], JAN [8], GAACN [12], CTSN [13], SAFN + ENT* [14], rRevGrad + CAT [15], SymNets [16], GSDA [20], RWOT [21], GCAN [24], CDAN [25], DANN [31], CBST [43], MCD [44], BSP + DANN [45] and SCA [46]. We show the accuracy of the methods mentioned in their published works.

4.3. Implementation Details

Following previous works [25,44], we use a pretrained ResNet-50 as our backbone for the experiments on Office-31, Office-Home, and ImageCLEF-DA datasets. Whereas we use a pretrained ResNet-101 for the experiments on VisDA dataset. We change the final fully-connected (FC) layer of the original networks with a task-specific FC layer to compose feature extractor G. One more FC layer is attached to it for object classification as classifier C. We use mini-batch stochastic gradient descent (SGD) with momentum of 0.9 to train the network. The semantic features used for cross-layer alignment are the features extracted by the last three blocks of the network. We adopt a three-layer residual convolution networks to match the size of semantic features in different layers of model. We set the learning rate to 0.001. We set the hyper-parameters delta and lambda to 0.7 and 0.3 in the Eq. 7 and Eq. 8 to analyse the effects on performance.

4.4. Main Results

Office-31. Table 1 shows the average classification accuracy for the six different settings on the office-31 dataset. ACDA shows a significant improvement over all other baseline UDA methods, achieving the state-of-the-art performance. It is also worth to mention that our ACDA method achieves the best performance on half the domain adaptation setting, which indicates that our cross-layer semantic alignment strategy is significantly effective in comparison to the other alignment-based methods [6,8].

Office-Home. We report the results of the experiments on the Office-Home dataset in Table 2. The results show that our ACDA method outperforms other baseline methods for most of the dataset settings. Our method improves performance for JAN by around 12.2%, which also considers the relationship between different layer of the model. We can demonstrate it as the effectiveness of attention-based cross-layer semantic alignment strategy, which effectively calibrates each level of semantic information and facilitate precise domain adaptation.

ImageCLEF-DA. The results on the ImageCLEF-DA dataset are in Table 3. For half set of training dataset setting, our method outperforms other state-of-the-art methods and achieves the highest average classification accuracy.

VisDA. VisDA is a huge dataset with 152,397 and 55,388 image samples for training and validation. The experiments on VisDA dataset is reported in Table 4. It is observed that our ACDA method significantly outperforms the other baseline methods on this dataset. The reason is that our ACDA method needs large data to

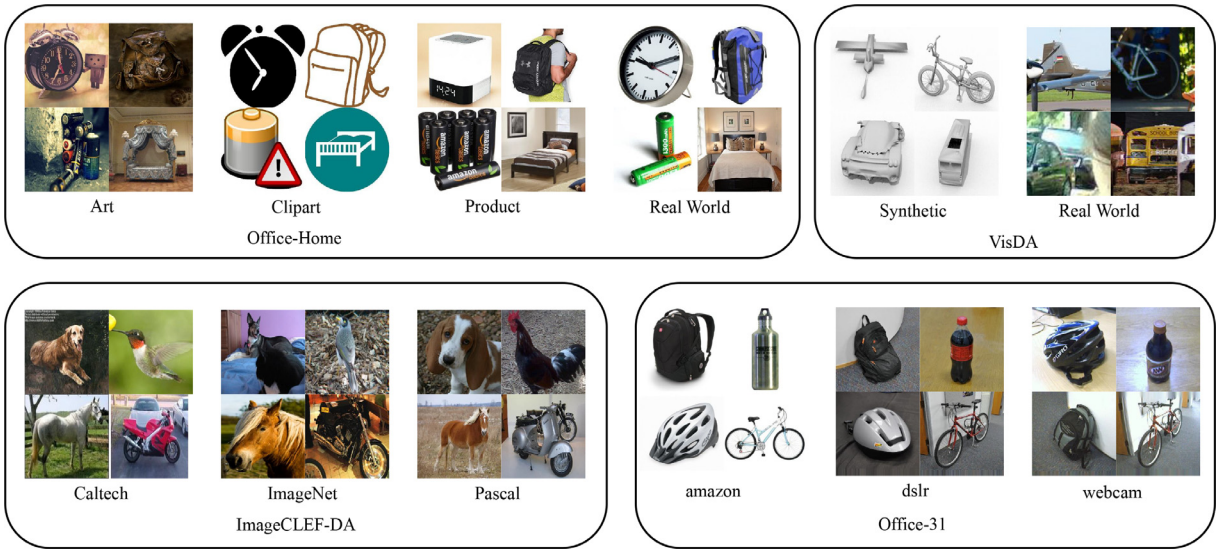

Fig. 3. Image examples of Office-31, Office-Home, ImageCLEF-DA, and VisDA datasets.

Table 1
Classification accuracy (%) for unsupervised domain adaptation on Office-31 dataset (mean ± standard error over 3 runs).

Method	A→W	D→W	W→D	A→D	D→A	W→A	Avg.
ResNet-50 [1]	68.4±0.2	96.7±0.1	99.3±0.1	68.9±0.2	62.5±0.3	60.7±0.3	76.2
DAN [6]	80.5±0.4	97.1±0.2	99.6±0.1	78.6±0.2	63.6±0.3	62.8±0.2	80.4
DANN [31]	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 [8]	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
CBST [43]	87.8±0.8	98.5±0.1	**100.0±0.0**	86.5±1.0	71.2±0.4	70.9±0.7	85.8
MCD [44]	89.6±0.2	98.5±0.1	**100.0±0.0**	91.3±0.2	69.6±0.1	70.8±0.3	86.6
CDAN [25]	93.1±0.2	98.2±0.2	**100.0±0.0**	89.8±0.3	70.1±0.4	68.0±0.4	86.6
BSP + DANN [45]	93.0±0.2	98.0±0.2	**100.0±0.0**	90.0±0.4	71.9±0.3	73.0±0.3	87.7
GCAN [24]	82.7±0.1	97.1±0.1	99.8±0.1	76.4±0.5	64.9±0.1	62.6±0.3	80.6
SAFN + ENT* [14]	90.3	98.7	**100.0**	92.1	73.4	71.2	87.6
rRevGrad + CAT [15]	94.4±0.1	98.0±0.2	**100.0±0.0**	90.8±1.8	72.2±0.6	70.2±0.1	87.6
SymNets [16]	90.8±0.1	98.8±0.3	**100.0±0.0**	93.9±0.5	74.6±0.6	72.5±0.5	88.4
SRDC [23]	**95.7±0.2**	**99.2±0.1**	**100.0±0.0**	95.8±0.2	76.7±0.3	77.1±0.1	90.8
GSDA [20]	**95.7**	99.1	**100.0**	94.8	73.5	74.9	89.7
GAACN [12]	90.2	98.4	**100.0**	90.4	67.4	67.7	85.6
SCA [46]	93.6±0.1	98.0±0.2	**100.0±0.0**	89.5±0.1	72.6±0.3	72.4±0.3	87.7
CDAN + TFLGM [5]	95.3±0.3	99.0±0.1	**100.0±0.0**	94.1±0.2	72.5±0.2	71.5±0.1	88.7
ACDA	94.57±0.3	98.48±0.49	99.93±0.12	**96.44±0.27**	**78.53±0.35**	**77.70±0.91**	**90.94**

capture and align the same level of semantic information among all the different layers of the model during training in domain adaptation setting..

4.5. Discussions

Sensitivity analysis. In Fig. 4(a) and 4(b), we report average accuracy of sensitivity analysis of the hyper-parameters δ and λ on Office-31 dataset. We find that our ACDA method is generally robust to different hyper-parameters, indicating that exhaustive hyper-parameter fine-tuning is not very necessary for ACDA to achieve good performance.

Training curves of accuracy and loss. Fig. 4(a) and 4(b), and Fig. 5(b) show the training accuracy curve and training loss curve of the three training setting on Office-Home dataset. It shows that our ACDA algorithm is more stable and it converges easily for different dataset settings. It indicates that the cross-layer semantic alignment strategy is very general and more robust.

Semantic feature distributions. To further analyze the effectiveness of our adaptation strategy, we visualize the learned semantic feature distributions for the C→P setting on ImageCLEF-DA dataset using t-SNE [48] in Fig. 6. It is observed that our ACDA method learns more domain-invariant semantic feature representations by aligning the source and target data in the semantic feature representation space via cross-layer semantic alignment. It proves that our method ACDA consistently achieves excellent performance for the UDA task on different datasets.

Ablation studies. We also implement ablation studies to investigate the effectiveness of each part of our ACDA method. The results are reported in Table 5. We can find that our proposed ACDA method improves the accuracy by 14.74% compared to the ResNet-50 method with no UDA strategy. By comparing the accuracy of the second and last row in Tabel 5, it shows that our proposed key design improves the accuracy by 3.94%. We find that it is necessary to explore the relationship between cross-layers of source and target domains. Besides, the elaborate dynamic attention mechanism is effective for achieving state-of-the-art model adaptation performance.

Structure of Projection Network We adopt a three-layer residual convolution networks as the learnable projection network, we compare different structure of the projection networks in Table 6.

Table 2
Classification accuracy (%) for unsupervised domain adaptation on Office-Home dataset (mean ± standard error over 3 runs).

Method	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg.
ResNet-50 [1]	42.5	50.0	58.0	37.4	41.9	46.2	38.5	42.4	60.4	53.9	41.2	59.9	47.7
DAN [6]	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 [31]	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 [8]	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
CDAN [25]	49.0	69.3	74.5	54.4	66.0	68.4	55.6	48.3	75.9	68.4	55.4	80.5	63.8
CBST [43]	51.4	74.1	78.9	56.3	72.2	73.4	54.4	41.6	78.8	66.0	48.3	81.0	64.7
BSP + DANN [45]	51.4	68.3	75.9	56.0	67.8	68.8	57.0	49.6	75.8	70.4	57.1	80.6	64.9
GAKT [3]	34.49	43.63	55.28	36.14	52.74	53.16	31.59	40.55	61.43	45.64	44.58	64.92	47.01
SAFN* [14]	**54.4**	73.3	77.9	65.2	71.5	73.2	63.6	52.6	78.2	72.3	58.0	82.1	68.5
SymNets [16]	47.7	72.9	78.5	64.2	71.3	74.2	**64.2**	48.8	**79.5**	**74.5**	52.6	82.7	67.6
GCAN [24]	36.43	47.25	61.08	37.90	58.25	57.00	35.77	42.66	64.47	50.08	49.12	72.53	51.05
GAACN [12]	53.1	71.5	74.6	59.9	64.6	67.0	59.2	53.8	75.1	70.1	59.3	80.9	65.8
SCA [46]	46.7	64.6	71.3	53.1	65.3	65.2	54.6	47.2	71.7	68.2	56.0	80.2	62.1
DRMEA [4]	52.3±0.4	73.0±0.6	77.3±0.3	64.3±0.3	72.0±0.7	71.8±0.5	63.6±0.6	52.7±0.7	78.5±0.2	72.0±0.1	57.7±0.6	81.6±0.2	68.1±0.2
CDAN + TFLGM [5]	51.4	72.0	77.2	61.7	71.9	72.2	60.0	51.7	78.8	72.8	58.9	82.0	67.6
ACDA	53.1±0.9	**74.8±1.2**	**82.6±0.5**	**69.8±0.8**	**75.8±1.1**	**77.4±0.1**	63.6±0.4	**54.7±0.8**	78.6±0.5	71.6±0.3	**60.6±0.9**	**83.2±0.7**	**70.5**

Table 3
Classification accuracy (%) for unsupervised domain adaptation on Image-CLEF dataset (mean ± standard error over 3 runs).

Method	I→P	P→I	I→C	C→I	C→P	P→C	Avg.
ResNet-50 [1]	74.8±0.3	83.9±0.1	91.5±0.3	78.0±0.2	65.5±0.3	91.2±0.3	80.7
DAN [6]	74.5±0.4	82.2±0.2	92.8±0.2	86.3±0.4	69.2±0.4	89.8±0.4	82.5
DANN [31]	75.0±0.6	86.0±0.3	**96.2±0.4**	87.0±0.5	74.3±0.5	91.5±0.6	85.0
JAN [8]	76.8±0.4	88.0±0.2	94.7±0.2	89.5±0.3	74.2±0.3	91.7±0.3	85.8
rRevGrad + CAT [15]	77.2±0.2	**91.0±0.3**	95.5±0.3	**91.3±0.3**	75.3±0.6	93.6±0.5	87.3
GCAN [24]	68.2±0.5	84.1±0.2	92.2±0.1	82.5±0.1	67.2±0.2	91.3±0.1	80.9
GAACN [12]	77.2	90.3	95.7	90.2	77.3	93.3	87.3
ACDA	**77.39±0.1**	89.32±1.0	95.9±0.3	89.41±0.6	**78.56±0.7**	**96.51±0.2**	**87.85**

Table 4
Classification accuracy (%) for unsupervised domain adaptation on VisDA dataset.

Method	ResNet-50 [1]	DAN [6]	MCD [44]	CDAN [25]	DRMEA [4]	GAACN [12]	CTSN [13]	SAFN [14]	GSDA [20]	RWOT [21]	ACDA
Avg.	52.4	61.1	71.9	73.7	79.3	69.8	75.4	76.1	81.5	84.0	**86.4**

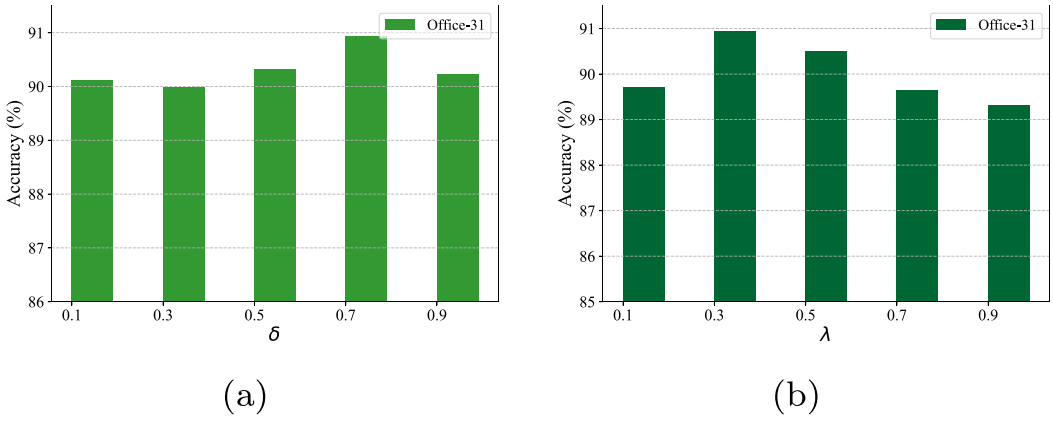

Fig. 4. (a): Sensitivity analysis of hyper-parameter δ of our ACDA method on Office-31 dataset; (b): Sensitivity analysis of hyper-parameter λ of our ACDA method on Office-31 dataset.

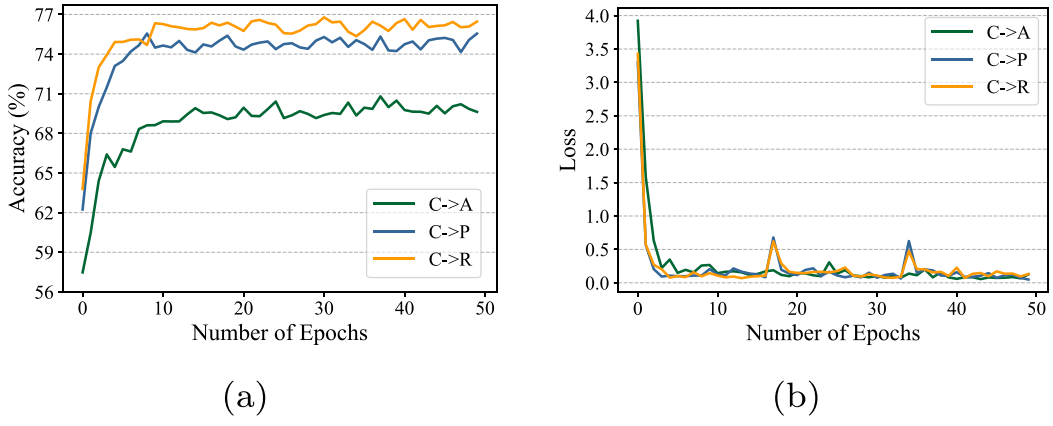

Fig. 5. (a): The accuracy of our ACDA method for the transfer task C→A, C→P, and C→R on Office-Home dataset; (b): The training loss of our ACDA method for the transfer task C→A, C→P, and C→R on Office-Home dataset.

It shows that the different structure of the projection networks have minor impact for the ACDA method to achieve the excellent model adaptation performance, and the adopted three-layer residual convolution networks achieves better results.

Alignment Layers We compare different number of layers for alignment in Table 7. It shows that we may have minor adaptation performance gain by adopting more layers for alignment, but it may increase the calculation cost.

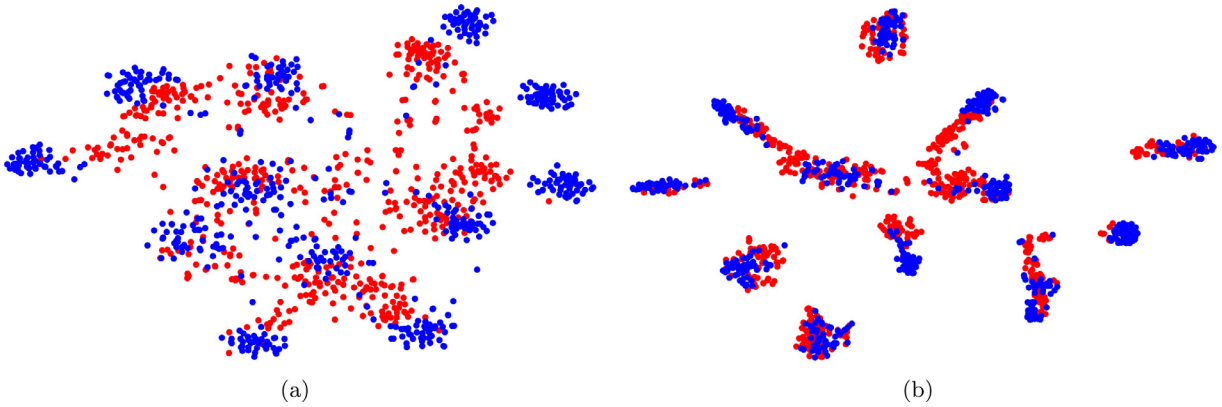

Fig. 6. T-SNE visualization of the semantic feature distributions of our method for the adaptation task C→P on ImageCLEF-DA dataset before adaptation (a) and after adaptation (b). The blue and red points represent samples in domain Caltech(C) and Pascal(P), respectively.

Table 5
Ablation experiments on Office-31 dataset for unsupervised domain adaptation.

Method	Avg.
ResNet-50 [1]	76.20
ACDA w/o label-conditioned & cross-layer alignment	87.00
ACDA w/o cross-layer alignment	89.27
ACDA w/o label-conditioned alignment	88.20
ACDA w/o dynamic attention mechanism	89.85
ACDA	**90.94**

Table 6
Comparison of different structure of convolution-based projection of ACDA method on Office-31 dataset.

Layers	Pooling	Residual block	Avg.
×	✓	×	89.78
1	×	×	90.12
3	×	×	90.29
3	✓	×	90.48
3	×	✓	**90.94**
3	✓	✓	90.82

Table 7
Comparison of different number of layers for alignment on Office-31 dataset.

Number of layers for alignment	Avg.
2	89.87
3	90.94
4	91.02
all	**91.09**

5. Conclusion

In this paper, we first point out that the same level of semantic information can be distributed across the different layers of the model, which can cause of negative transfer gain in previous UDA methods with same-layer alignment. We propose a novel attention-based cross-layer domain alignment method to address this problem by reweighting each cross-layer pair according to the semantic similarity for precise domain alignment. Extensive experiments show the superior performance of our method in comparison to the other state-of-the-art UDA methods. In future, we will extend our framework to other adaptation tasks of computer vision, like semantic segmentation and object detection.

CRediT authorship contribution statement

Xu Ma: Writing - original draft, Software, Conceptualization, Methodology, Validation. **Junkun Yuan:** Software, Writing - original draft. **Yen-wei Chen:** Writing - review & editing. **Ruofeng Tong:** Writing - review & editing. **Lanfen Lin:** Supervision, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Natural Science Foundation of Zhejiang Province (LZ22F020012) and Major Scientific Research Project of Zhejiang Lab (2020ND8AD01).

References

[1] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.
[2] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, J.W. Vaughan, A theory of learning from different domains, Machine learning 79 (2010) 151–175.
[3] Z. Ding, S. Li, M. Shao, Y. Fu, Graph adaptive knowledge transfer for unsupervised domain adaptation, in: Proceedings of the European Conference on Computer Vision (ECCV), pp. 37–52.
[4] Y.-W. Luo, C.-X. Ren, P. Ge, K.-K. Huang, Y.-F. Yu, Unsupervised domain adaptation via discriminative manifold embedding and alignment, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, pp. 5029–5036.
[5] R. Zhu, X. Jiang, J. Lu, S. Li, Transferable feature learning on graphs across visual domains, in: 2021 IEEE International Conference on Multimedia and Expo (ICME), IEEE, pp. 1–6.
[6] M. Long, Y. Cao, J. Wang, M. Jordan, Learning transferable features with deep adaptation networks, in: International conference on machine learning, PMLR, pp. 97–105.
[7] M. Long, H. Zhu, J. Wang, M.I. Jordan, Unsupervised domain adaptation with residual transfer networks, arXiv preprint arXiv:1602.04433 (2016).
[8] M. Long, H. Zhu, J. Wang, M.I. Jordan, Deep transfer learning with joint adaptation networks, in: International conference on machine learning, PMLR, pp. 2208–2217.
[9] H. Venkateswara, J. Eusebio, S. Chakraborty, S. Panchanathan, Deep hashing network for unsupervised domain adaptation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5018–5027.
[10] D. Chen, J.-P. Mei, Y. Zhang, C. Wang, Z. Wang, Y. Feng, C. Chen, Cross-layer distillation with semantic calibration, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 35, pp. 7028–7036.
[11] J. Yuan, X. Ma, D. Chen, K. Kuang, F. Wu, L. Lin, Collaborative semantic aggregation and calibration for separated domain generalization, arXiv e-prints (2021) arXiv–2110.

[12] W. Chen, H. Hu, Generative attention adversarial classification network for unsupervised domain adaptation, Pattern Recognition 107 (2020) 107440.
[13] L. Zuo, M. Jing, J. Li, L. Zhu, K. Lu, Y. Yang, Challenging tough samples in unsupervised domain adaptation, Pattern Recognition 110 (2021) 107540.
[14] R. Xu, G. Li, J. Yang, L. Lin, Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1426–1435.
[15] Z. Deng, Y. Luo, J. Zhu, Cluster alignment with a teacher for unsupervised domain adaptation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9944–9953.
[16] Y. Zhang, H. Tang, K. Jia, M. Tan, Domain-symmetric networks for adversarial domain adaptation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5031–5040.
[17] J. Yuan, X. Ma, K. Kuang, R. Xiong, M. Gong, L. Lin, Learning domain-invariant relationship with instrumental variable for domain generalization, arXiv preprint arXiv:2110.01438 (2021).
[18] Y. Pan, T. Yao, Y. Li, Y. Wang, C.-W. Ngo, T. Mei, Transferrable prototypical networks for unsupervised domain adaptation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2239–2247.
[19] G. Kang, L. Jiang, Y. Yang, A.G. Hauptmann, Contrastive adaptation network for unsupervised domain adaptation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4893–4902.
[20] L. Hu, M. Kan, S. Shan, X. Chen, Unsupervised domain adaptation with hierarchical gradient synchronization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4043–4052.
[21] R. Xu, P. Liu, L. Wang, C. Chen, J. Wang, Reliable weighted optimal transport for unsupervised domain adaptation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4394–4403.
[22] J. Yuan, X. Ma, D. Chen, K. Kuang, F. Wu, L. Lin, Domain-specific bias filtering for single labeled domain generalization, arXiv preprint arXiv:2110.00726 (2021).
[23] H. Tang, K. Chen, K. Jia, Unsupervised domain adaptation via structurally regularized deep clustering, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 8725–8735.
[24] X. Ma, T. Zhang, C. Xu, Gcan: Graph convolutional adversarial network for unsupervised domain adaptation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8266–8276.
[25] M. Long, Z. Cao, J. Wang, M.I. Jordan, Conditional adversarial domain adaptation, arXiv preprint arXiv:1705.10667 (2017).
[26] J. Li, E. Chen, Z. Ding, L. Zhu, K. Lu, Z. Huang, Cycle-consistent conditional adversarial transfer networks, in: Proceedings of the 27th ACM International Conference on Multimedia, pp. 747–755.
[27] W. Zhang, W. Ouyang, W. Li, D. Xu, Collaborative and adversarial network for unsupervised domain adaptation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3801–3809.
[28] M. Chen, S. Zhao, H. Liu, D. Cai, Adversarial-learned loss for domain adaptation, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 3521–3528.
[29] S. Cui, S. Wang, J. Zhuo, C. Su, Q. Huang, Q. Tian, Gradually vanishing bridge for adversarial domain adaptation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12455–12464.
[30] E. Tzeng, J. Hoffman, K. Saenko, T. Darrell, Adversarial discriminative domain adaptation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7167–7176.
[31] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, V. Lempitsky, Domain-adversarial training of neural networks, The journal of machine learning research 17 (2016) 2096–2030.
[32] R. Shao, X. Lan, P.C. Yuen, Feature constrained by pixel: Hierarchical adversarial deep domain adaptation, in: Proceedings of the 26th ACM international conference on Multimedia, pp. 220–228.
[33] B. Yang, P.C. Yuen, Cross-domain visual representations via unsupervised graph alignment, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pp. 5613–5620.
[34] L. Mescheder, A. Geiger, S. Nowozin, Which training methods for gans do actually converge?, in: International conference on machine learning, PMLR, pp. 3481–3490.
[35] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, arXiv preprint arXiv:1409.0473 (2014).
[36] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in neural information processing systems, pp. 5998–6008.
[37] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3146–3154.
[38] X. Wang, L. Li, W. Ye, M. Long, J. Wang, Transferable attention for domain adaptation, in: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 5345–5352.
[39] C. Li, D. Du, L. Zhang, L. Wen, T. Luo, Y. Wu, P. Zhu, Spatial attention pyramid network for unsupervised domain adaptation, in: European Conference on Computer Vision, Springer, pp. 481–497.
[40] W. Chen, H. Hu, Generative attention adversarial classification network for unsupervised domain adaptation, Pattern Recognition 107 (2020) 107440.
[41] Y. Zuo, H. Yao, C. Xu, Attention-based multi-source domain adaptation, IEEE Transactions on Image Processing 30 (2021) 3793–3803.
[42] J. Liang, D. Hu, J. Feng, Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation, in: International Conference on Machine Learning, PMLR, pp. 6028–6039.
[43] Y. Zou, Z. Yu, B. Kumar, J. Wang, Unsupervised domain adaptation for semantic segmentation via class-balanced self-training, in: Proceedings of the European conference on computer vision (ECCV), pp. 289–305.
[44] K. Saito, K. Watanabe, Y. Ushiku, T. Harada, Maximum classifier discrepancy for unsupervised domain adaptation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3723–3732.
[45] X. Chen, S. Wang, M. Long, J. Wang, Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation, in: International conference on machine learning, PMLR, pp. 1081–1090.
[46] W. Deng, L. Zheng, Y. Sun, J. Jiao, Rethinking triplet loss for domain adaptation, IEEE Transactions on Circuits and Systems for Video Technology 31 (2020) 29–37.
[47] K. Saito, Y. Ushiku, T. Harada, Asymmetric tri-training for unsupervised domain adaptation, in: International Conference on Machine Learning, PMLR, pp. 2988–2997.
[48] L. Van der Maaten, G. Hinton, Visualizing data using t-sne, Journal of machine learning research 9 (2008).

Xu Ma received his B.S. degree from the College of Computer Science and Technology at Harbin Institute of Technology in 2020. He is currently pursuing Master degree at the College of Computer Science and Technology at Zhejiang University since 2020. His research interests include domain adaptation and domain generalization.

Junkun Yuan received his B.S. degree from the College of Information Engineering at Zhejiang University of Technology in 2019. He is currently pursuing the Ph.D. degree with the College of Computer Science and Technology at Zhejiang University since 2019.

Yen-Wei Chen (Member, IEEE) received the B.E. degree from Kobe University, Kobe, Japan, in 1985, and the M.E. and D.E. degrees from Osaka University, Osaka, Japan, in 1987 and 1990, respectively. From 1991 to 1994, he was a Research Fellow with the Institute for Laser Technology, Osaka. From October 1994 to March 2004, he was an Associate Professor and a Professor with the Department of Electrical and Electronic Engineering, University of the Ryukyus, Okinawa, Japan. He is currently a Professor with the College of Information Science and Engineering, Ritsumeikan University, Kyoto, Japan. He is also a Visiting Professor with the College of Computer Science and Technology, Zhejiang University, China, and the Research Center for Healthcare Data Science, Zhejiang Laboratory, China. His research interests include pattern recognition, image processing, and machine learning. He has published more than 200 research articles in these fields. He is an Associate Editor of the International Journal of Image and Graphics (IJIG) and an Associate Editor of the International Journal of Knowledge-Based and Intelligent Engineering Systems.

Ruofeng Tong received his B.S. degree in mathematics from Fudan University and his Ph.D. degree in applied mathematics in 1996 from Zhejiang University. Currently, he is a professor in the College of Computer Science and Engineering, Zhejiang University, China. His research interests include CAD&CG, medical image reconstruction, and virtual reality.

Lanfen Lin (Member, IEEE) received Ph.D. degrees in Aircraft Manufacture Engineering from Northwestern Polytechnical University in 1995. She held a postdoctoral position with the College of Computer Science and Technology, Zhejiang University, China, from January 1996 to December 1997. Now she is a Full Professor and the Vice Director of the Artificial Intelligence Institute in Zhejiang University. Her research interests include medical image processing, big data analysis, data mining, and so on.