Domain Alignment with Triplets

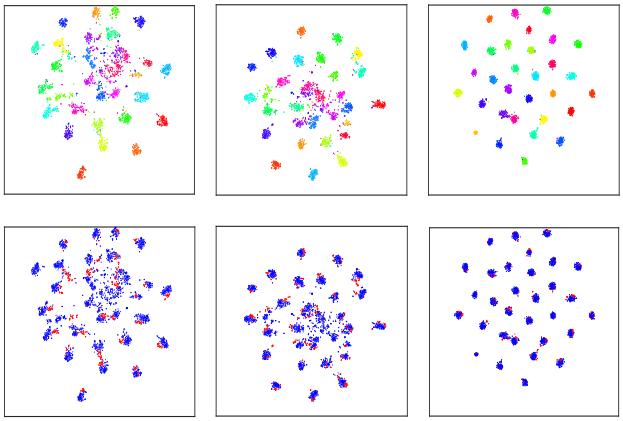
Weijian Deng[†], Liang Zheng[‡], Jianbin Jiao[†*]
[†]University of Chinese Academy of Sciences [‡] Australian National University
dengweijian16@mails.ucas.ac.cn

Abstract

Deep domain adaptation methods can reduce the distribution discrepancy by learning domain-invariant embeddings. However, these methods only focus on aligning the whole data distributions, without considering the class-level relations among source and target images. Thus, a target embeddings of a bird might be aligned to source embeddings of an airplane. This semantic misalignment can directly degrade the classifier performance on the target dataset. To alleviate this problem, we present a similarity constrained alignment (SCA) method for unsupervised domain adaptation. When aligning the distributions in the embedding space, SCA enforces a similarity-preserving constraint to maintain class-level relations among the source and target images, i.e., if a source image and a target image are of the same class label, their corresponding embeddings are supposed to be aligned nearby, and vise versa. In the absence of target labels, we assign pseudo labels for target images. Given labeled source images and pseudo-labeled target images, the similarity-preserving constraint can be implemented by minimizing the triplet loss. With the joint supervision of domain alignment loss and similarity-preserving constraint, we train a network to obtain domain-invariant embeddings with two critical characteristics, intra-class compactness and inter-class separability. Extensive experiments conducted on the two datasets well demonstrate the effectiveness of SCA.

1. Introduction

In many real-world application of visual recognition, the training and testing data distributions are often different due to *dataset bias* [41]. This distribution discrepancy decreases the generalization capability of the learned visual representations. One example is that the model trained on synthetic images fails to generalize well on the real-world images. To eliminate the effect of the dataset bias, a common used strategy is unsupervised domain adaption (UDA). In UDA, we are provided with a labeled source dataset and an unlabeled target dataset, and the goal is to learn a model on the source dataset which minimizes the test error on the target dataset.



(a) ResNet (b) Domain alignment (c) SCA

Figure 1. Visualization of cross-domain embeddings for task **A** → **W** on Office-31 [35]. We present the 2D visualization of t-SNE for embeddings learned by (a) ResNet (trained on source images only), (b) domain alignment (based on JMMD [28]), and (c) SCA (ours). For the **first row**, different colors denote data of different object categories. For the **second row**, red color represents the data of **W**, and blue color represents data of **A**. Under SCA, different classes are well-separated, and the two domains are well-aligned on the class level. Best viewed in color.

In literature, recent UDA methods [10, 28, 9, 42, 43, 25] adopt deep neural networks to learn a shared embedding space where the distribution discrepancy can be reduced. These methods typically involve two objectives: 1) learn embeddings that maintain a low classification error on the source dataset; 2) make embeddings domain-invariant, such that the classifier trained on the source can be directly used on the target dataset. To learn domain-invariant embeddings, recent methods usually minimize some measure of domain variance [43, 28, 25] (such as correlation distance [40]) or adopt the adversarial learning [10, 9, 42]. However, this line of methods have an intrinsic limitation: they only focus on reducing the global distribution discrepancy, without exploiting the class-level relations among the source and

*Corresponding Author

target images. Thus, even with perfect distribution alignment, the images with different labels from different domains might be misaligned nearby in the embedding space. As shown in Fig. 1(b), domain-level alignment (based on JMMD [28]) has the ability to reduce distribution discrepancy. However, these exists the semantic misalignment problem in the aligned embeddings. For examples, some samples from different classes are mapped nearby in the embedding space. This semantic misalignment is detrimental to the classifier performance on the target dataset.

Motivated by this problem, we present a similarity constrained alignment (SCA) method for UDA. The working mechanism of SCA is that it can align the distributions, while preserving the class-level relations among source and target images. Specifically, we add a similarity-preserving constraint for the source and target images during domain alignment. The impact of the similarity-preserving constraint is two-fold. 1) *Class unification*: images with same labels should be pulled together in the embedding space; 2) *class separation*: images with different labels should be pushed apart. In practice, the similarity-preserving constraint can be implemented by minimizing the triplet loss [37]. During training, SCA learns domain-invariant embeddings by optimizing an objective that includes both the domain confusion loss and the triplet loss [37]. First, the domain confusion loss aims at mapping the source and target distributions into a shared feature space. Several existing methods can be directly used to achieve this goal. In this paper, we adopt JMMD [28] to align the data distributions. Second, the triplet loss is to enhance the discriminative ability of the deeply learned embeddings, so that source and target embeddings possess the properties of intra-class compactness and inter-class separability.

Unfortunately, the target dataset is totally unlabeled, so the similarity-preserving constraint cannot be directly imposed for the source and target images. In the absence of target labels, we use a classifier trained on source images to assign pseudo labels for target images. To eliminate the influence of the incorrectly assigned images, we only select images with high predicted scores for training. Given labeled source images and pseudo-labeled target images, we utilize the triplet loss [37] to constrain their similarity in the embedding space. Specifically, if a source image and a target image are with the same class label, their corresponding embeddings are supposed to be aligned nearby, and vise versa. In this manner, the semantic misalignment problem can be alleviated. As shown in Fig. 1(c), we observe that the embeddings learned by our method preserve the two class-level relations: 1) the embeddings that belong to the same class are close (class unification); 2) the embeddings that belong to different classes are separated well (class separation). Based on the domain-invariant embeddings learned by SCA, the classifier can generalize well on the target dataset.

To summarize, this paper is featured in three aspects. First, to our knowledge, this is an early work that explores the class-level relations across domains under the UDA setting. Second, by consolidating the idea of domain-level alignment and metric learning, this paper presents a novel similarity constrained alignment (SCA) method for UDA. SCA attempts to reduce the distribution discrepancy while preserving the underlying difference and commonness among source and target images. Thus, the class-level misalignment problem can be alleviated. Third, extensive experiment results demonstrate that the proposed method improves the generalization ability of the learned classifier. Moreover, the proposed method is capable of producing competitive accuracy to state-of-the-art methods on two UDA benchmarks.

2. Related Work

Many methods are proposed to solve the domain adaptation problem. This section briefly reviews works that are closely related to our paper.

Unsupervised domain adaptation. Unsupervised domain adaptation methods attempt to minimize the shift between source and target data distributions. Some methods focus on learning a mapping function between source and target distributions [20, 13, 8, 39]. In [39], Correlation Alignment is proposed to match the two distributions. In [8]. The source and target domain are aligned in the subspace described by Eigenvectors.

Other methods seek to find a shared feature space for source and target distributions [9, 43, 25, 28]. Long *et al*. [25] and Tzeng *et al*. [43] utilize the maximum mean discrepancy (MMD) metric [14] to learn a share feature representations. Moreover, the joint maximum mean discrepancy (JMMD) [28] is proposed to align the joint distributions of multi-layers across domains. Recent methods [10, 9, 42, 46, 3, 31] adopt adversarial learning [12] to learn representations that are not able to distinguish between domains. The gradient reversal algorithm (RevGrad) [9] is proposed to learn the domain invariant feature. Tzeng *et al*. [42] propose a generalized framework for adversarial domain adaptation. Pei *et al*. [31] propose a multi-domain adversarial network for fine-grained distribution alignment. SimNet [32] proposes to classify an image by computing its similarity to prototype representations of each category. Some methods [17, 2, 23, 7] use the adversarial learning to learn a transformation in the pixel space from one domain to another. CYCADA [17] maps samples across domains at both pixel level and feature level. In this paper, we also attempt to reduce the distribution discrepancy, and we are more concerned with preserving the class-level relations among the source and target datasets.

Self-training. Our method is related to self-training, a

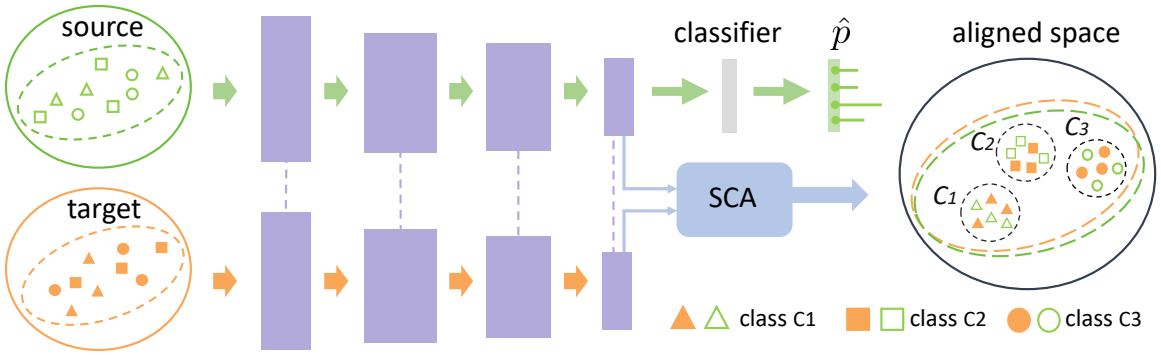


Figure 2. Framework of the similarity constrained alignment (SCA) method. SCA has the ability to align the distribution, while preserving the class-level relations among source and target images. Thus, if a source image and a target image are with the same class label, their corresponding embeddings are supposed to be aligned nearby, and vise versa. Due to the target dataset is unlabeled, we assign pseudo labels for the target images (see Section 3.2.2). In this figure, different colors denote different domain distributions and different shapes represent different classes.

strategy in which the predictions of a classifier on the unlabeled data are used to retrain the classifier [22, 5, 21, 46, 33, 19]. The assumption of self-training is that an image with the high predicted score is more likely to be classified correctly. In unsupervised domain adaptation, some methods [46, 4, 36] use pseudo-labeled images to improve classifier accuracy on the target dataset. Zhang *et al*. [46] propose a progressive way to select pseudo-labeled images for training the classifier. Chen *et al*. [4] use two classifiers to assign labels for target images. Saito [36] adopt three asymmetric classifiers to improve the quality of pseudo labels. Unlike these methods, we leverage the selected images with their pseudo-labels for semantic alignment instead of retraining the classifier. This practice provides a new way to utilize unlabeled data for learning feature representations.

Deep Metric learning. Deep metric learning [6, 11, 44, 38, 37, 18] aims to learn discriminative embeddings such that similar samples are nearer and different samples are further apart from each other. The most widely used loss functions for deep metric learning are the contrastive Loss [6] and triplet loss [37]. The problem settings of these works are different from ours. We aim to reduce the distribution discrepancy and utilize the triplet loss [37] to preserve the class-level relations among images from the two domains. Since the target domain is unlabeled, we assign pseudo labels for the target images.

3. Proposed Method

3.1. Overview

In UDA, we are provided with a set of labeled images from the source dataset and a set of unlabeled images from the target dataset, where the data distributions of the two datasets are different. For the source dataset, we denote it as $\mathcal{D}_s = \{(\mathbf{x}_i^s, \mathbf{y}_i^s)\}_{i=1}^{n_s}$, where $\mathbf{x}_i^s$ is the i-th source image, $\mathbf{y}_i^s$ is its label, and n_s is the total number of images on the source dataset. Similarly, we denote the target dataset as $\mathcal{D}_t = \{\mathbf{x}_j^t\}_{j=1}^{n_t}$, where $\mathbf{x}_i^t$ is the i-th target image and n_t is the total number of images on the target dataset. Our goal is to leverage labeled source images and unlabeled target images to learn a classifier that can generalize well on the target dataset.

It is worth repeating that, for UDA, we not only deal with the whole distribution discrepancy that caused by the dataset bias. We also consider preserving the class-level relations among source and target images. To this end, we present a similarity constrained alignment (SCA) method for UDA. As shown in Fig. 2, the goal of SCA is two-fold, 1) it learns a domain-invariant embedding space to align the whole distributions; 2) it preserve the underlying difference and commonness among source and target images. Based on the learned embeddings, we can train a classifier that can generalize well on the target dataset.

In Section 3.2.1, we briefly describe the domain-level alignment method used in this paper. In Section 3.2.2, we introduce the similarity-preserving scheme. In Section 3.3, we have a discussion about the proposed method.

3.2. Similarity Preserving Alignment

In this paper, we utilize the deep convolution neural network to learn the classifier. For K-way classification with a cross-entropy loss, this is corresponding to,

$$\mathcal{L}_c = \frac{1}{n_s} \sum_{i=1}^{n_s} L\left(f\left(g_\theta(\mathbf{x}_i^s)\right), \mathbf{y}_i^s\right), \tag{1}$$

where $L(\cdot, \cdot)$ is the cross-entropy loss, where $g_\theta(\cdot)$ is the feature extractor, and $f(\cdot)$ is the classifier trained on the source dataset.

In general, the classifier $f(\cdot)$ is a simple fully-connected network followed by a softmax over the classes . Due the dataset bias [41], the classifier trained on the source dataset often fails to generalize well on the target dataset. To alleviate this problem, we present a similarity constrained alignment (SCA) method. SCA can eliminate the distribution discrepancy, while preserving the underlying difference and commonness among source and target images. In practice, SCA learn domain-invariant embeddings by optimizing over an objective that includes both the domain-level alignment loss and the similarity-preserving loss.

3.2.1 Domain-level alignment

Domain-level alignment focuses on reducing the whole distribution discrepancy between the source and target datasets. In the community, recent deep domain adaptation methods utilize a domain confusion loss to align the distributions. These methods usually adopt the discrepancy-based metric [14, 28] or adversarial adaptation [9] to design the domain confusion loss function.

Following the practice in [28], we build the domain-level alignment loss by using the JMMD metric. The JMMD formally reduces the discrepancy in the joint distributions of the activations in domain-specific layers $\mathcal{L}$, i.e. $P(\mathbf{Z}^{s1}, \ldots, \mathbf{Z}^{s|\mathcal{L}|})$ and $Q(\mathbf{Z}^{t1}, \ldots, \mathbf{Z}^{t|\mathcal{L}|})$. Thus, the loss function of domain-level alignment is written as,

$$\begin{aligned}\mathcal{L}_d = &\frac{2}{n}\sum_{i=1}^{n/2}\left(\prod_{\ell\in\mathcal{L}} k^\ell(\mathbf{z}_{2i-1}^{s\ell}, \mathbf{z}_{2i}^{s\ell}) + \prod_{\ell\in\mathcal{L}} k^\ell(\mathbf{z}_{2i-1}^{t\ell}, \mathbf{z}_{2i}^{t\ell})\right) \\ &- \frac{2}{n}\sum_{i=1}^{n/2}\left(\prod_{\ell\in\mathcal{L}} k^\ell(\mathbf{z}_{2i-1}^{s\ell}, \mathbf{z}_{2i}^{t\ell}) + \prod_{\ell\in\mathcal{L}} k^\ell(\mathbf{z}_{2i-1}^{t\ell}, \mathbf{z}_{2i}^{s\ell})\right),\end{aligned} \tag{2}$$

where $n = n_s$, $\mathbf{z}^{t\ell}$ denotes the activations of the target image in the layer ℓ, and $\mathbf{z}^{s\ell}$ denotes the activations of the source image in the layer ℓ. k^ℓ is the kernel function in a reproducing kernel Hilbert space (RKHS).

We adopt the ResNet-50 [15] as the backbone network. We discard its last layer and add two fully connected layers (a bottleneck layer, and a classifier layer) for our task. In practice, we align the joint distributions of the activations in two newly added layers.

3.2.2 Similarity-constrained Scheme

Domain-level alignment only aims at reducing the whole distribution discrepancy, but it can mix up the class-level relations among the source and target images. Consequently, there exists a semantic misalignment problem, *i.e.*, source images of class A might be falsely aligned to target images of class B in the embedding space. This semantic misalignment problem directly degrades accuracy on the target dataset. To mitigate this problem, we should consider the class-level relations of images across two datasets. In this paper, we propose to preserve the underlying difference and commonness among images during the domain alignment.

Class-level relations. A general assumption behind the similarity-preserving alignment is that if a source image and a target image are with the same class label, their corresponding embeddings are supposed to be aligned nearby, and vise versa. On the top of domain-level alignment, we add a similarity-preserving constraint to maintain two class-level relations among source and target images. In this paper, the two class-level relations are defined as follow.

- *Class separation.* Images from different domains and with different labels, should be mapped far apart in the embedding space.
- *Class unification.* Images from different domains but with same labels, should be mapped nearby in the embedding space

Similarity-preserving loss function. To mitigate the semantic misalignment problem, we want images to preserve the above class-level relations during the domain-level alignment. Let $D_{i,j} = \|g_\theta(x_i) - g_\theta(x_j)\|_2^2$ measures the distance between two images in the feature space, where $g_\theta(\cdot)$ is the feature extractor. If x_i and x_j are with the same label, we want $D_{i,j}$ to be small, corresponding to the class unification. If x_i and x_j are with different labels, we want $D_{i,j}$ to be large, corresponding to the class separation.

Based on the above analysis, we utilize the triplet loss [37] to achieve similarity-preserving constraint. Given an *anchor* image x_a, a *positive* image x_p, and a *negative* image x_p, we minimize the loss,

$$\mathcal{L}_s(\theta) = \sum_{\substack{a,p,n \\ y_a=y_p\neq y_n}} [m + D_{a,p} - D_{a,n}]_+ , \tag{3}$$

where x_a and x_p is a positive pair (their labels y_a and y_p are same), x_a and x_p is a negative pair (their labels y_a and y_n are different). m is the margin that is enforced between positive and negative pairs.

This loss encourages the distance between x_a and positive image x_p to be smaller than the distance between x_a and negative x_n by the enforced margin m.

Training data construction. The similarity-preserving loss supervises the embedding learning, so that class-level relations among source and target images can be preserved. When optimizing the similarity-preserving loss, we should pay attention to two crucial things, 1) the target dataset is totally unlabeled; 2) the construction of training triplet samples is non-trivial. For these two things, we propose corresponding techniques.

(i) Label estimation for unlabeled target data. The target dataset is totally unlabeled, so the semantic relations cannot be directly built. In the absence of target labels, we

use a classifer pre-trained on the source images to assign labels for unlabeled target images.

To ensure the accuracy of the pseudo label, we adopt three tactics. (a) *domain-level alignment.* When pre-training the classifier, we also utilize the dataset-level to reduce the harmful influence of dataset bias. This practice improves the performance of the classifier on the target dataset, so that more accurate pseudo labels can be gained. (b) *Threshold T.* Intuitively, the image with the high predicted score is more likely to be classified correctly. Thus, we only select target images with predicted scores above a high threshold T for building the semantic relations. Note that the threshold T is constant during training. (c) *Progressive selection.* With the help of the similarity-preserving alignment, the classifier will improve itself during training. This motivates us to re-assign the label for the target image every several iterations (K). By doing so, the target images can be progressively selected for the class-level alignment.

(ii) Sample triplet images. Given labeled source images and pseudo-labeled target images, we now introduce the way to construct triplet samples. The possible number of triplets is large, and optimizing all triplets is computationally infeasible. To avoid this problem, we follow the sampling strategy in [16]. For the labeled source images, we randomly select C classes and randomly select K images of each class. In this way, we select CK source images. Similarly, we select CK pseudo-labeled target images. Thus, we get a mini-batch of $2CK$ training images and perform triplet sampling in each mini-batch.

3.2.3 Overall objective

We present a similarity constrained alignment (SCA) for UDA. During the training, SCA jointly optimizes an objective that includes both a domain-level alignment loss and a similarity-preserving loss, such that more discriminative domain-invariant embeddings can be gained. On the top of learned embeddings, we can train a classifier that generalizes well on the target dataset. The final objective of SCA is written as,

$$\mathcal{L}_{can} = \mathcal{L}_c + \alpha\mathcal{L}_d + \beta\mathcal{L}_s, \tag{4}$$

where $\mathcal{L}_c$ is the classification loss, $\mathcal{L}_d$ is the domain-level domain alignment, and $\mathcal{L}_s$ is the similarity-preserving loss. The α and the β control the relative importance of domain-level alignment and similarity preservation, respectively.

3.3. Discussion

Collaborative working mechanism. The working mechanism of SCA is that it can align the distributions, while preserving the class-level relations among source and target images. On the one hand, if we only use the domain-level alignment to reduce the distribution discrepancy, the resulting embeddings would exist the semantic misalignment problem. On the other hand, the similarity-preserving constraint can map a source image and a target image nearby, if they are with the same class label. Thus, the similarity-preserving constraint can be viewed as the class-level distribution alignment. With the collaborative supervision of them, we can reduce the distribution at both domain level and class level, *i.e.*, learning domain-invariant embeddings that preserve the class-level relations. In our experiment, we validate this collaborative working mechanism. Moreover, we also study the impact of only adopting the similarity-preserving constraint on the transfer accuracy.

Closely related to our work, Motiian *et al*. [30] also study the class-level alignment. Our work is different from [30] in two aspects, 1) the setting of [30] is supervised domain adaptation, where the labeled target images are available; 2) the authors of [30] do not consider the domain-level alignment, while our work collaboratively aligns the distributions at both domain and class level.

Label estimation. To construct class-level relations among the source and target images, we need to estimate the labels of unlabeled target images. In this paper, we simply adopt a classifier pre-trained on the source images to assign pseudo labels for unlabeled target images. We only select target images with their scores above a certain threshold T. Note that we do not adaptively adjust the threshold T as in [46]. In practice, we set the threshold T a high value (0.9) to guarantee that the selected samples are more likely to be predicted correctly.

During the training, the classifier will gradually improve itself, so we re-assign pseudo labels every several iterations. In this way, more and more target images will be progressively selected for training.

How to use pseudo-labeled target images? Existing methods [22, 5, 21, 46, 33, 19] usually utilize the pseudo-labeled target images for training classifier directly. In this paper, the pseudo-labeled images are **not used** for training the classier, but for building the class-level relations. We argue that there exist a set of wrongly pseudo-labeled images, which can directly bring a bad influence to the classifier. To avoid this problem, we use selected target images for optimizing the similarity-preserving loss function.

Moreover, as analyzed in [24, 45], cross-entropy loss encourage the features of different classes staying apart. Thus, using selected target images for training classifier can be viewed as an indirect way to preserve the class separation relation. However, the cross-entropy loss does not consider the class unification relation. In contrast, we adopt pseudo-labeled target images and source images for constructing both class unification and separation relations.

Figure 3. Visual examples of the Office-31 dataset. From top to bottom: DSLR images (high-resolution), Amazon images (medium-resolution), and Webcam images (low-resolution).

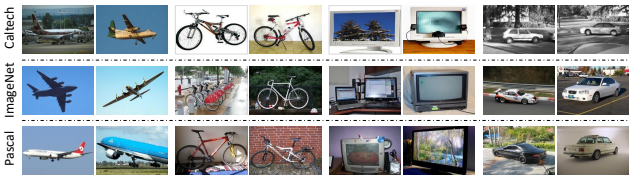


Figure 4. Visual examples of the ImageCLEF-DA dataset. From top to bottom: Caltech-256 images, ImageNet ILSVRC 2012 images, and Pascal VOC 2012 images.

4. Experimental Evaluation

4.1. Datasets

We evaluate the proposed unsupervised domain adaptation method on two datasets, *i.e.*, Office-31 [35] and ImageCLEF-DA[1].

Office-31 is a widely used benchmark for visual domain adaptation. It contains 4,652 images and 31 categories collected from three distinct domains: *Amazon* (**A**), *Webcam* (**W**) and *DSLR* (**D**). The images in DSLR are captured with a digital SLR camera and have high resolution. Amazon consists of images downloaded from online merchants (www.amazon.com). These images are of products at medium resolution. The images in Webcam are collected by a web camera, and they are of low resolution. We evaluate the proposed method across six transfer tasks **A** $\rightarrow$ **W**, **D** $\rightarrow$ **W**, **W** $\rightarrow$ **D**, **A** $\rightarrow$ **D**, **D** $\rightarrow$ **A** and **W** $\rightarrow$ **A**. We report the results following the protocol in [25].

ImageCLEF-DA is a benchmark dataset for ImageCLEF 2014 domain adaptation challenge. It contains three subsets, including *Caltech-256* (**C**), *ImageNet ILSVRC 2012* (**I**), and *Pascal VOC 2012* (**P**), and each subset is considered as a domain. There are 12 categories and each categories contains 50 images. We use all domain combinations and build 6 transfer tasks: **I** $\rightarrow$ **P**, **P** $\rightarrow$ **I**, **I** $\rightarrow$ **C**, **C** $\rightarrow$ **I**, **C** $\rightarrow$ **P**, and **P** $\rightarrow$ **C**. We report the results following the protocol in [28]. Sample images of the Office-31 and ImageCLEF-DA are shown in Fig 3 and Fig 4, respectively.

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

Algorithm 1 Similarity Constrained Alignment (SCA).

inputs
  source images and labels $\{(\mathbf{x}_i^s, \mathbf{y}_i^s)\}_{i=1}^{n_s}$,
  unlabeled target images $\{\mathbf{x}_j^t\}_{j=1}^{n_t}$.
  threshold (T), max number of steps (S), and number of SCA updates per step (K).
stage 1: pre-train a classifier:
train a classifer by minimizing Eq. 1 and Eq. 2.
stage 2: class-level alignment:
for s=1; s $\leq S$; s ++ **do**
  use classifer to assign pseudo labels for target images with predicted score above T.
  for k =1; k $\leq K$; k ++ **do**
    train SCA by minimizing Eq. 4.
  end for
end for

Through these images, we can observe the dataset bias discussed in [41].

4.2. Implementation Details

We implement our method on pytorch framework, and fine-tune from ResNet-50 model [15] pre-trained on the ILSVRC 2012 dataset [34]. All the images are resized to 256×128. We discard its last layer and add two fully connected layers for our task. The first layer has 256 units, and the second goes down to the number of training classes. During training, we adopt random flipping and random cropping as data augmentation methods. We use stochastic gradient descent (SGD) for optimization, and adopt the same INV learning rate strategy as in RevGrad [9]. The learning rate decreases gradually after each iteration from 0.001, the momentum is set to 0.9, and the weight decay is set to 0.0004. We set $\alpha = 1$ and $\beta = 1$ in Eq. 4.

We adopt a two-stage training procedure: we first initial the classifier by minimizing Eq. 1 and Eq. 2, then train the whole network by minimizing Eq. 4. The training procedure is summarized in Algorithm 1. For the stage one training, we train the network for 5000 iterations. For the stage two training, we training the remaining 30000 iterations. We set threshold $T = 0.9$, max number of step $S = 15$, and number of SCA updates per step $K = 2000$.

4.3. Experimental Results

Compared Apporaches. In this section, we mainly compare the proposed method with several state-of-the-art methods, including DAN [25], RTN [27], JAN [28], RevGrad [9], MADA [31], SimNet [32], iCAN [46], and CDAN [26]. These methods are all based on the deep neural network (ResNet-50 [15]) to learn domain-invariant embeddings. For the fair comparison, the results of these methods are directly reported from their original papers.

Method	A→W	A→D	W→A	W→D	D→A	D→W	Avg.
ResNet-50 [15]	72.5	73.6	59.9	99.3	61.0	93.6	76.7
DAN [25]	80.5	78.6	62.8	99.6	63.6	97.1	80.4
RTN [27]	84.5	77.5	64.8	99.4	66.2	96.8	81.6
JAN [28]	85.8	85.0	70.0	99.7	68.9	96.7	84.4
RevGrad [9]	82.0	79.7	67.4	99.1	68.2	96.9	82.2
MADA [31]	90.0	87.8	66.4	99.6	70.3	97.4	85.2
SimNet [32]	88.6	85.3	71.8	99.7	**73.4**	98.2	86.2
iCAN [46]	92.5	90.1	69.9	**100.0**	72.1	**98.8**	87.2
CDAN-RM [26]	93.0	89.2	69.4	**100.0**	70.2	98.4	86.7
CDAN-M [26]	93.1	**93.4**	70.3	**100.0**	71.0	98.6	**87.7**
SCA	**93.5**	89.5	**72.7**	**100.0**	72.4	97.5	87.6

Table 1. Comparison of different methods for unsupervised domain adaptation on the Office-31 dataset. The best results are in **bold**.

Method	I→P	P→I	I→C	C→I	C→P	P→C	Avg.
ResNet-50 [15]	74.8	82.9	91.5	78.0	66.2	87.2	80.1
DAN [25]	74.5	82.2	92.8	86.3	69.2	89.8	82.5
RTN [27]	74.6	85.8	94.3	85.9	71.7	91.2	83.9
RevGrad [9]	75.0	86.0	96.2	87.0	74.3	91.5	85.0
JAN [28]	76.8	88.0	94.7	89.5	74.2	91.7	85.8
MADA [31]	75.0	87.9	96.0	88.8	75.2	92.2	85.8
iCAN [46]	**79.5**	89.7	94.7	89.9	**78.5**	92.0	87.4
CDAN-RM [26]	77.2	88.3	**98.3**	90.7	76.7	**94.0**	87.5
CDAN-M [26]	76.2	**89.5**	96.0	91.2	75.0	93.5	86.9
SCA	78.1	89.2	96.8	**91.3**	78.2	**94.0**	**87.9**

Table 2. Comparision of different methods for unsupervised domain adaptation on the ImageCLEF-DA dataset. The best results are in **bold**.

Comparison on the Office-31 dataset. We compare the proposed method with the recent state-of-the-art methods in Table 1. Our method (SCA) gains 87.6% accuracy, which is the second best performance on the Office-31 dataset. Note that our method is comparable with CDAN-M [26] (87.6% vs. 87.7%). Besides, our method achieves the highest performance on three tasks (A → W, W → A, and W → D). Our method is higher than MADA [31] (87.6% vs. 85.2%). Moreover, our method outperforms SimNet, iCAN, and JAN by 1.4%, 0.4%, and 3.2%, respectively.

Comparison on the ImageCLEF-DA dataset. In Table 2, we compare the proposed method with state-of-the-art methods. SCA obtains 87.9%, which outperforms the other methods. The accuracy of our method is 0.4% higher than the second best method CDAN-RM [26]. Moreover, the proposed method respectively outperforms the MADA [31], iCAN [46], and JAN [28] by 2.1%, 0.5%, and 2.1%. Specifically, our methods achieves the highest performance on two tasks (C → I and P → C).

The comparisons on the Office-31 dataset (Table 1) and the ImageCLEF-DA dataset (Table 2) demonstrate the effectiveness of the proposed method.

Method	A→W	A→ D	W→A	W→D	D→A	D→W	Avg.
B (Basel.)	76.5	78.0	64.0	99.0	65.0	94.8	79.6
B + D	87.2	84.9	69.8	99.2	67.8	96.5	84.2
B + S	85.0	87.0	67.2	99.4	67.5	98.2	84.1
SCA	93.5	89.5	72.7	100.0	72.4	97.5	87.6

Table 3. Ablation experimental results of SCA. The results are on the Office-31 dataset. "B" (Basel.) denotes the baseline trained only the source dataset, " S" represents the similarity-preserving constraint, and "D" denotes the domain-level alignment. SCA is the full system ("B + D + S").

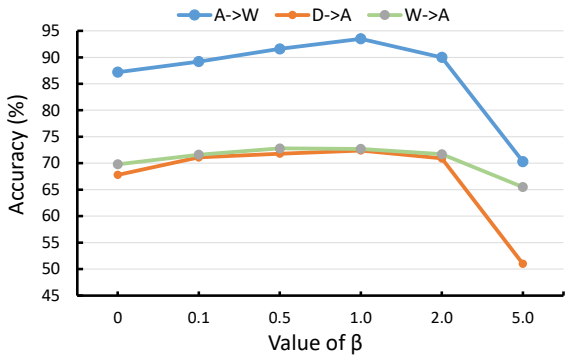


Figure 5. Sensitivity to parameter β (weight of the similarity-preserving constraint) in Eq. 4. A larger β means that the constraint has a greater impact on the distribution alignment.

4.4. Component analysis

In this section, we present step-by-step evaluation to analyze the effectiveness of SCA.

Ablation study. We investigate the impact of different components in SCA. We conduct the experiment on the Office-31 and report the results on Table 3.

The baseline is the network that we modify from ResNet-50, and it does not adopt any domain adaptation technique. In this paper, we adopt JMMD [28] for the domain-level alignment, and the result of "B+ D" is consistent with the experiment in [28]. Compared with "B" (Basel.), "B + D" achieves higher performance, which indicates that it has ability to reduce the distribution discrepancy.

On the top of domain-level alignment, the similarity-preserving constraint further brings +3.4% improvement in average accuracy. This well demonstrates the importance of preserving underlying difference and commonness among source and target images.

As discussed in 3.3, the similarity-preserving constraint can be viewed as a way to align distributions at class level. We further study its impact on the transfer accuracy, and report its results ("B+S") in Table 3. We can observe that only adopting the similarity constraint can also improve the baseline performance it gains +4.5% improvement over the baseline in average accuracy. This indicts that preserving class-level relations benefits the transfer accuracy.

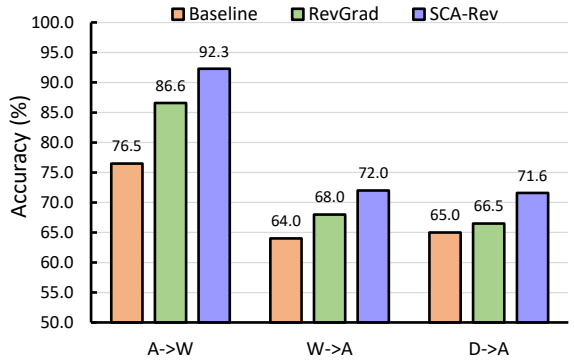


Figure 6. Performance of various methods on three tasks ($\mathbf{A} \rightarrow \mathbf{W}$, $\mathbf{W} \rightarrow \mathbf{A}$, and $\mathbf{D} \rightarrow \mathbf{A}$) of Office-31. Reverse Gradient (RevGrad) [9] is a domain-level alignment method based on adversarial learning. SCA-Rev is the similarity-preserving alignment based on RevGrad.

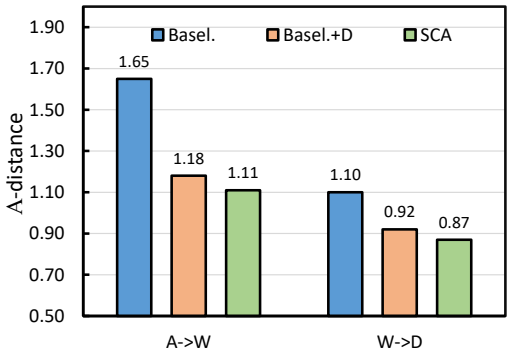


Figure 7. Distribution discrepancy measured by $\mathcal{A}$-distance on tasks $\mathbf{A} \rightarrow \mathbf{W}$ and $\mathbf{W} \rightarrow \mathbf{D}$. Three methods are compared: (a) baseline (Basel.), (b) domain-level alignment (Basel. + D), and (c) SCA.

Weight of the similarity-preserving constraint. The β in Eq. 4 control the importance of similarity-preserving constrain. A larger β means that the constraint has a greater impact on the distribution alignment. In Fig 5, we demonstrate the transfer accuracy of SCA by varying the $\beta \in \{0, 0.1, 0.5, 1, 2, 5\}$ on three tasks, $\mathbf{A} \rightarrow \mathbf{W}$, $\mathbf{W} \rightarrow \mathbf{A}$, and $\mathbf{D} \rightarrow \mathbf{A}$. Note the when β is set to 0, the similarity-preserving constrain has no impact. As shown in Fig. 5, when the β increases from 0 to 1, the performance on three tasks grow and reach the best at $\beta = 1$. However, when the β is too large (β=5), the accuracy will drop by a large margin. Empirically, the best parameter β is between 0.5 to 2 in our method.

Domain-level alignment method. As discussed in Section 3.2.1, we use a discrepancy-based metric JMMD for domain-level alignment. We note that the proposed similarity-preserving constraint can work collaboratively with other domain-level alignment methods. To validate this, we conduct the experiment on three tasks of Office-31: $\mathbf{A} \rightarrow \mathbf{W}$, $\mathbf{W} \rightarrow \mathbf{A}$, and $\mathbf{D} \rightarrow \mathbf{A}$. We adopt an adversarial adaptation method named Reverse Gradient (RevGrad) [9] for domain-level alignment. Based on RevGrad, we construct the similarity constrained alignment network (SCA-Rev), and report the results on the Fig. 6.

As shown in Fig. 6, RevGrad can improve the accuracy of baseline, which indicates it has ability to reduce the distribution discrepancy. Moreover, SCA-Rev further improves the accuracy of RevGrad. SCA-Rev gains +5.7%, +4.0% and 5.1% improvements over RevGrad on $\mathbf{A} \rightarrow \mathbf{W}$, $\mathbf{W} \rightarrow \mathbf{A}$, and $\mathbf{D} \rightarrow \mathbf{A}$, respectively. On the one hand, the results demonstrate that preserving the two class-level relations is crucial for the domain-level alignment. On the other hand, these results indicate that the similarity-preserving constraint can work collaboratively with other domain-level alignment methods.

Distribution discrepancy. The domain adaptation theory [1, 29] introduces $\mathcal{A}$-distance to measure the distribution discrepancy . The $\mathcal{A}$-distance is defined as $d_{\mathcal{A}} = 2\,(1 - 2\epsilon)$, where ϵ is the generalization error of a classifier trained to discriminate source and target. We report the $d_{\mathcal{A}}$ on two tasks ($\mathbf{A} \rightarrow \mathbf{W}$, $\mathbf{W} \rightarrow \mathbf{D}$) of Office-31 with features of baseline, domain-level alignment (basel. + G), and SCA. As shown in Fig. 7, $d_{\mathcal{A}}$ on SCA features is much smaller than $d_{\mathcal{A}}$ on the baseline and domain-level alignment features. This indicates that SCA features can reduce the distribution discrepancy more effectively.

5. Conclusion and Future Work

In this paper, we present the similarity constrained alignment (SCA) method to address the semantic misalignment problem. SCA enforces a similarity-preserving constraint to maintain the underlying difference and commonness among the source and target images. In the absence of target labels, we use a classifier trained on source images to assign pseudo labels to the target images. Given labeled source images and pseudo-labeled target images, the similarity-preserving constraint can be implemented by minimizing the triplet loss. Under the collaborative supervision of the domain alignment loss and the triplet loss, SCA learns domain-invariant embeddings with two important properties, *i.e.*, intra-class compactness and inter-class separability. Thus, the distributions can be aligned at both domain and class level, which alleviates the semantic misalignment problem. The experimental results on two benchmarks demonstrate that the proposed SCA is effective and competitive with the state-of-the-art methods. In the future, we will extend this idea to multiple target domains, where the class-level relations among multi-domains will be explored.

References

[1] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175, 2010. 8

[2] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2

[3] Z. Cao, L. Ma, M. Long, and J. Wang. Partial adversarial domain adaptation. In *ECCV*, 2018. 2

[4] M. Chen, K. Q. Weinberger, and J. Blitzer. Co-training for domain adaptation. In *NIPS*, 2011. 3

[5] X. Chen, A. Shrivastava, and A. Gupta. NEIL: extracting visual knowledge from web data. In *ICCV*, 2013. 3, 5

[6] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005. 3

[7] W. Deng, L. Zheng, Q. Ye, G. Kang, Y. Yang, and J. Jiao. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2

[8] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *ICCV*, 2013. 2

[9] Y. Ganin and V. S. Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, 2015. 1, 2, 4, 6, 7, 8

[10] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. S. Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17:59:1–59:35, 2016. 1, 2

[11] J. Goldberger, S. T. Roweis, G. E. Hinton, and R. Salakhutdinov. Neighbourhood components analysis. In *NIPS*, 2004. 3

[12] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative adversarial networks. In *NIPS*, 2014. 2

[13] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *ICCV*, 2011. 2

[14] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. J. Smola. A kernel method for the two-sample-problem. In *NIPS*, 2006. 2, 4

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4, 6, 7

[16] A. Hermans, L. Beyer, and B. Leibe. In defense of the triplet loss for person re-identification. *CoRR*, abs/1703.07737, 2017. 5

[17] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. A. Efros, and T. Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International Conference on Machine Learning*, 2018. 2

[18] J. Hu, J. Lu, and Y. Tan. Discriminative deep metric learning for face verification in the wild. In *CVPR*, 2014. 3

[19] G. Kang, L. Zheng, Y. Yan, and Y. Yang. Deep adversarial attention alignment for unsupervised domain adaptation: the benefit of target expectation maximization. In *ECCV*, 2018. 3, 5

[20] B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *CVPR*, 2011. 2

[21] S. Laine and T. Aila. Temporal ensembling for semi-supervised learning. *CoRR*, abs/1610.02242, 2016. 3, 5

[22] L. Li and F. Li. OPTIMOL: automatic online picture collection via incremental model learning. *International Journal of Computer Vision*, 2010. 3, 5

[23] M. Liu and O. Tuzel. Coupled generative adversarial networks. In *Advances in Neural Information Processing Systems*, 2016. 2

[24] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*, 2017. 5

[25] M. Long, Y. Cao, J. Wang, and M. I. Jordan. Learning transferable features with deep adaptation networks. In *ICML*, 2015. 1, 2, 6, 7

[26] M. Long, Z. Cao, J. Wang, and M. I. Jordan. Conditional adversarial domain adaptation. In *NIPS*, 2018. 6, 7

[27] M. Long, H. Zhu, J. Wang, and M. I. Jordan. Unsupervised domain adaptation with residual transfer networks. In *NIPS*, 2016. 6, 7

[28] M. Long, H. Zhu, J. Wang, and M. I. Jordan. Deep transfer learning with joint adaptation networks. In *ICML*, 2017. 1, 2, 4, 6, 7

[29] Y. Mansour, M. Mohri, and A. Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *COLT 2009 - The 22nd Conference on Learning Theory, Montreal, Quebec, Canada, June 18-21, 2009*, 2009. 8

[30] S. Motiian, M. Piccirilli, D. A. Adjeroh, and G. Doretto. Unified deep supervised domain adaptation and generalization. In *ICCV*, 2017. 5

[31] Z. Pei, Z. Cao, M. Long, and J. Wang. Multi-adversarial domain adaptation. In *AAAI*, 2018. 2, 6, 7

[32] P. O. Pinheiro. Unsupervised domain adaptation with similarity learning. In *CVPR*, 2018. 2, 6, 7

[33] I. Radosavovic, P. Dollár, R. B. Girshick, G. Gkioxari, and K. He. Data distillation: Towards omni-supervised learning. In *CVPR*, 2018. 3, 5

[34] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 6

[35] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*, 2010. 1, 6

[36] K. Saito, Y. Ushiku, and T. Harada. Asymmetric tri-training for unsupervised domain adaptation. In *ICML*, 2017. 3

[37] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015. 2, 3, 4

[38] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, 2016. 3

[39] B. Sun, J. Feng, and K. Saenko. Return of frustratingly easy domain adaptation. In *Proceedings of AAAI Conference on Artificial Intelligence*, 2016. 2

[40] B. Sun and K. Saenko. Deep CORAL: correlation alignment for deep domain adaptation. In *ECCV Workshops*, 2016. 1

[41] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 1, 4, 6

[42] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *CVPR*, 2017. 1, 2

[43] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell. Deep domain confusion: Maximizing for domain invariance. *CoRR*, abs/1412.3474, 2014. 1, 2

[44] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10, 2009. 3

[45] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, 2016. 5

[46] W. Zhang, W. Ouyang, W. Li, and D. Xu. Collaborative and adversarial network for unsupervised domain adaptation. In *CVPR*, 2018. 2, 3, 5, 6, 7