Neurocomputing 461 (2021) 254–265

Contents lists available at ScienceDirect

Neurocomputing

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

A novel class restriction loss for unsupervised domain adaptation

Qi He [a,b], Qi Dai [c], Xiao Wu [a,b,*], Jun-Yan He [a,b]

[a] School of Computing and Artificial Intelligence, Xipu Campus, Southwest Jiaotong University, Chengdu 611756, China
[b] National Engineering Laboratory of Integrated Transportation Big Data Application Technology, China
[c] Microsoft Research Asia, Beijing 100080, China

ARTICLE INFO

Article history:
Received 2 October 2020
Revised 23 June 2021
Accepted 15 July 2021
Available online 20 July 2021
Communicated by Zidong Wang

Keywords:
Unsupervised domain adaptation
Class restriction loss
Deep learning
Convolutional neural network
Image classification

ABSTRACT

Domain adaptation has demonstrated promising performance to learn models in new environments. It aims to transfer the knowledge learned in labeled source domain to a new target domain, avoiding expensive efforts of label annotation. Many popular methods attempt to assign pseudo labels to unlabeled target samples and train the models as if they are true labels. However, the pseudo labels contain inevitable noises while their training strategies would enlarge and accumulate the errors, so that the model easily overfits to noisy data. Instead of focusing on assigning correct pseudo labels, in this paper, an unsupervised domain adaptation approach is proposed by preventing the model from overfitting false samples. Two different types of restrictions are considered on the training data to leverage the intra-class centralization and inter-class normalization. Each training sample and individual category are equally treated, where each point has the same opportunity to contribute to the model learning. A novel class restriction loss is proposed, which is further integrated into a teacher-student architecture, where the outputs from the teacher are treated as the pseudo labels for the student. The whole framework is trained in an end-to-end manner. Extensive experiments conducted on several image classification benchmarks demonstrate that the proposed method can significantly improve the performance, which outperforms the state-of-the-art methods.

1. Introduction

Recently, Convolutional Neural Network (CNN) has successfully demonstrated its great capability of learning visual representations, which is generally workable across a wide range of tasks and domains [1]. Coupled with the well annotated large scale datasets (e.g., ImageNet [2] and Kinetics [3]), a series of CNN architectures have been developed [4–10]. Nevertheless, applying the pre-learned CNN models to a new target dataset still needs to perform the fine-tuning with the labeled target samples, in which expensive manual annotation is required. Meanwhile, unsatisfactory results are usually achieved due to a phenomenon known as "domain shift." Consequently, several Domain Adaptation (DA) techniques have been proposed [11–20], which aim to transfer the rich knowledge in source domain to the target domain. Particularly, it is regarded as Unsupervised Domain Adaptation (UDA), when the target labels are entirely unavailable during the model training. Under this situation, generalized models working well on target data have to be learned.

Domain adaptation or transfer learning methods can be categorized into two directions, *i.e. homogeneous* and *heterogeneous* domain adaptation. In homogeneous domain adaptation, source and target domains have similar feature spaces and with same dimensionality [21]. Most existing methods focus on this task [22,23,18,24], including manifold-based [25,26], integral-probability-metric [27], pseudo-labeling [28], and adversarial-learning methods [12]. Particularly, utilizing the adversarial learning for domain adaptation has been popular in recent studies [12,18,29,14,16,24,30]. Though they have achieved promising results, target domains are usually heterogeneous from source domains in practical scenarios. Therefore, heterogeneous domain adaptation is proposed, where the source and target features are disjointed and with different dimensionalities. Representative methods attempt to design specific distance metrics and feature mapping functions [31,32] for heterogeneous domains. In this paper, we focus on the scenario of homogeneous domain adaptation.

Recent studies [16,28,33,88] have been explored to gradually enlarge the training set with pseudo-labeled target samples. A cru-

* Corresponding author at: School of Computing and Artificial Intelligence, Xipu Campus, Southwest Jiaotong University, Chengdu 611756, China.
E-mail address: wuxiaohk@swjtu.edu.cn (X. Wu).

https://doi.org/10.1016/j.neucom.2021.07.033
0925-2312/

cial problem in these methods is the inevitable noises. The standard classification loss could accumulate the errors due to the sample-selection bias [34], making the model overfitting the samples with false labels. As a result, the classifier would be easily biased to certain dominant classes. In addition, it is difficult to converge for target models. Existing methods always set the weight of target loss as well as the number of iterations to be small, as the noise samples would harm the results.

In this paper, instead of selecting the correct pseudo labels, an unsupervised domain adaptation approach is proposed to prevent the model from overfitting the false samples. The noises will have very little influence on the model, as long as the majority of the samples is correctly labeled and the errors will not be accumulated. The model can thus be fully optimized without the interference of false samples.

To this end, a novel class restriction loss (CRL) is proposed, which contains two types of restrictions on the training data: the intra-class centralization and inter-class normalization. The centralization pulls all samples within the class towards a center. By reducing the scores of easy samples (high scores) and conversely increasing those of hard ones (low scores), the model can evade the overfitting to noisy easy samples and additionally force the hard samples away from the decision boundary, reducing the uncertainty in model optimization. On the other hand, the normalization forces all training points to have the same norm values. In this way, training samples of different classes are mandatorily gathered on a hypersphere, making each class have the opportunity to contribute to the classifier, which prevents the model from being biased to dominant classes. To leverage the proposed loss for pseudo label assignment, it is then integrated into a teacher-student architecture, where the output probabilities are utilized as soft pseudo labels of the student. In essence, the centralization and normalization build the intra-class equality and inter-class balance, respectively. Experiments conducted on several image classification benchmarks demonstrate that the effectiveness of the proposed method, which outperforms the state-of-the-art methods.

The contributions of this paper are summarized as follows:

1. A novel class restriction loss is proposed to alleviate the noisy pseudo label overfitting problem in unsupervised domain adaptation, which consists of two major components, the intra-class centralization and inter-class normalization. In spirit, the approach treats each training sample and category equally, which effectively reduces the influence of noises.
2. The proposed class restriction loss is integrated into a teacher-student architecture, where the generated confidence scores from teacher are utilized as pseudo labels for student. This end-to-end framework additionally boosts the classification performance.
3. Experiments demonstrate that the proposed method effectively alleviates the aforementioned problems. Considerable improvement has been achieved on a set of datasets, including several digits datasets, the standard Office31 and the large-scale VisDA-2017.

The rest of this paper is organized as follows. Section 2 briefly reviews recent related works. Sections 3 and 4 elaborate the proposed approach and experimental comparisons, respectively. Finally, this paper is concluded with a summary.

2. Related work

Recent works related to our approach can be summarized into three aspects: domain adaptation, learning with noisy labels, and centralization/normalization in deep learning.

Deep Unsupervised Domain Adaptation aims to transfer the models learned in a labeled source domain to a target domain in a deep learning framework. Recent research of this topic has proceeded along several main dimensions. Traditional methods match the features of both domains in an embedding space [25]. Neural Embedding Matching (NEM) [26] assumes that the local geometry property of the data can be maintained in the embedding space, which is regularized via metric learning and graph embedding techniques. Two projections are learned using probabilistic class-wise adaptation strategy so that intrinsic information across domains can be preserved with the graph [35]. Maximum Mean Discrepancy (MMD) [36] is utilized as the metric to measure the shift between domains. Deep Domain Confusion (DDC) [37] applies MMD as well as the regular classification loss on source to learn discriminative and domain-invariant representations. Deep Adaptation Network (DAN) [27] extends this idea by embedding all task specific layers in a reproducing kernel Hilbert space. In Deep Correlation Alignment (CORAL) [38], MMD is exploited to match the mean and covariance of two distributions. Joint Adaptation Network (JAN) [39] aligns the joint distributions of features and final outputs across domains. Transferrable Prototypical Network (TPN) [87] utilizes pseudo label of target domain to adapt the class-level discrepancy. Recently, different from MMD, Contrastive Adaptation Network (CAN) [40] is proposed to optimize the contrastive domain discrepancy objective for domain adaptation, which explicitly minimizes the intra-class discrepancy and maximizes the inter-class margin. By counting the class information, it can better align the data from two domains. Cluster Alignment with a Teacher (CAT) [41] also considers the class-conditional information to make a more reasonable domain alignment. Sliced Wasserstein Discrepancy (SWD) [42] improves the discrepancy measure that aims to minimize the cost of moving the marginal distributions between the task-specific classifiers by using Wasserstein metric.

Another category of methods is inspired from adversarial learning [43], which plays a minimax two-player game to guide the representation learning in both domains, so that the difference between source and target representation distributions can be indistinguishable through the domain discriminator. Domain Adversarial Neural Network (DANN) [12] utilizes a gradient reversal method to learn domain-invariant features. Adversarial Discriminative Domain Adaptation (ADDA) [18] unties weight sharing of CNN models to learn domain specific feature embedding. Conditional Domain Adversarial Network (CDAN) [14] extends the Conditional Generative Adversarial Networks (CGANs) [29], in which a conditional domain discriminator is utilized to capture the relationships between features and classifier predictions. Maximum Classifier Discrepancy (MCD) [44] aligns the distribution of a target domain by considering task-specific decision boundaries with adversarial learning. Regularized Conditional Alignment (RCA) [45] imposes a $2C$-way adversarial loss (C is the class number) to make better domain-class alignment. An application for urban village extraction in satellite images is introduced by using adversarial learning [46]. Similar to the aforementioned methods, the adversarial learning is utilized in this paper to reduce the distribution discrepancies between domains.

The most related works are the pseudo label-based methods for domain adaptation. Asymmetric Tri-Training (ATT) [28] trains two networks for assigning pseudo labels to target samples, which are immediately utilized to train the third network. DIRT-T [33] stud-

ies the cluster assumption for domain adaptation, which states that decision boundaries should not cross high-density regions. A teacher model is then adopted to assign pseudo labels in its second stage. Incremental Collaborative and Adversarial Network (iCAN) [16] applies several domain classifiers to the hidden feature representations, and further enlarges the training set with pseudo-labeled target samples. However, these methods can hardly handle the noises in pseudo labels, which is the key focus of this paper.

Learning with Noisy Labels is important for practical applications [47–54]. Most existing methods [47–49] are based on the soft label smoothing or regularization. A knowledge distillation method is proposed in [52] to deal with noisy labeled data. The network parameters and the noisy labels are optimized during training in [53]. Another category of works focuses on the task of domain adaptation under the circumstance of noisy source data [55,56], or namely *weakly-supervised* domain adaptation. A recent observation for deep networks is the memorization effect, which means that the networks attempt to memorize easy patterns (samples) first, and gradually adapt to hard samples [54,56]. Based on this observation, some works treat small-loss data as easy and correct samples, and the data selection or weight adjusting schemes are adopted. Transferable Curriculum Learning (TCL) [55] assigns large training weights to the clean source samples, where an iterative optimization is adopted to compute the weights. According to the memorization effect, Butterfly [56] selects the positive training samples based on their cross-entropy losses. However, only using memorization effect to select clean data may still suffer from the accumulated errors caused by sample-selection bias [34]. Those easy negative samples with small losses would be regarded as correct ones. To address the problem, Butterfly uses a co-teaching paradigm to reduce such samples. Different from aforementioned methods, in this paper the centralization is leveraged to reduce the overconfidence of these easy negative samples, rather than directly selecting them, which avoids the overfitting to them.

Centralization and Normalization have been explored in deep frameworks for robust feature learning [57–63]. Center loss [57] is proposed to learn discriminative features for face recognition. Ring loss [59] constrains the norm to the scaled unit circle, while preserving convexity for learning robust features. Similarly, L_2-softmax [61] also imposes the norm constraint on the deep features to improve the performance of face verification. Adaptive Feature Norm (AFN) [64] finds that larger norm could benefit the domain transfer. Nevertheless, there are limited explorations of centralization and normalization in the domain adaptation task. We innovatively study the equality and balance properties derived from them and prove their effectiveness in this challenging task.

Intra-class compactness and inter-class separability are common techniques in computer vision community, which are also related to our work. It has been widely explored in the representation learning of many tasks, including face recognition [57], person re-identification [65] and domain adaptation [40,66]. In this paper, intra-class equality and inter-class balance are proposed, which are induced from centralization and normalization. The proposed equality and balance modeling aims to achieve an utterly different goal that avoids the model from overfitting, which is proven effective in improving the adaptation performance.

3. Unsupervised domain adaptation with class restriction loss

Unsupervised Domain Adaptation aims to transfer knowledge from a labeled source domain to a target domain, where labeled data are unavailable. Given the data $\mathscr{X}_s = \{\boldsymbol{x}_s^1, \ldots, \boldsymbol{x}_s^n\}$ with labels $\{\boldsymbol{y}_s^1, \ldots, \boldsymbol{y}_s^n\}$ in source and $\mathscr{X}_t = \{\boldsymbol{x}_t^1, \ldots, \boldsymbol{x}_t^m\}$ in target, where both of them are from C classes, we expect to learn a good classifier on the target.

Fig. 1 illustrates the framework of the proposed approach. Given source and target samples, a shared CNN model is first utilized to extract features. A domain discriminator is utilized to guide the

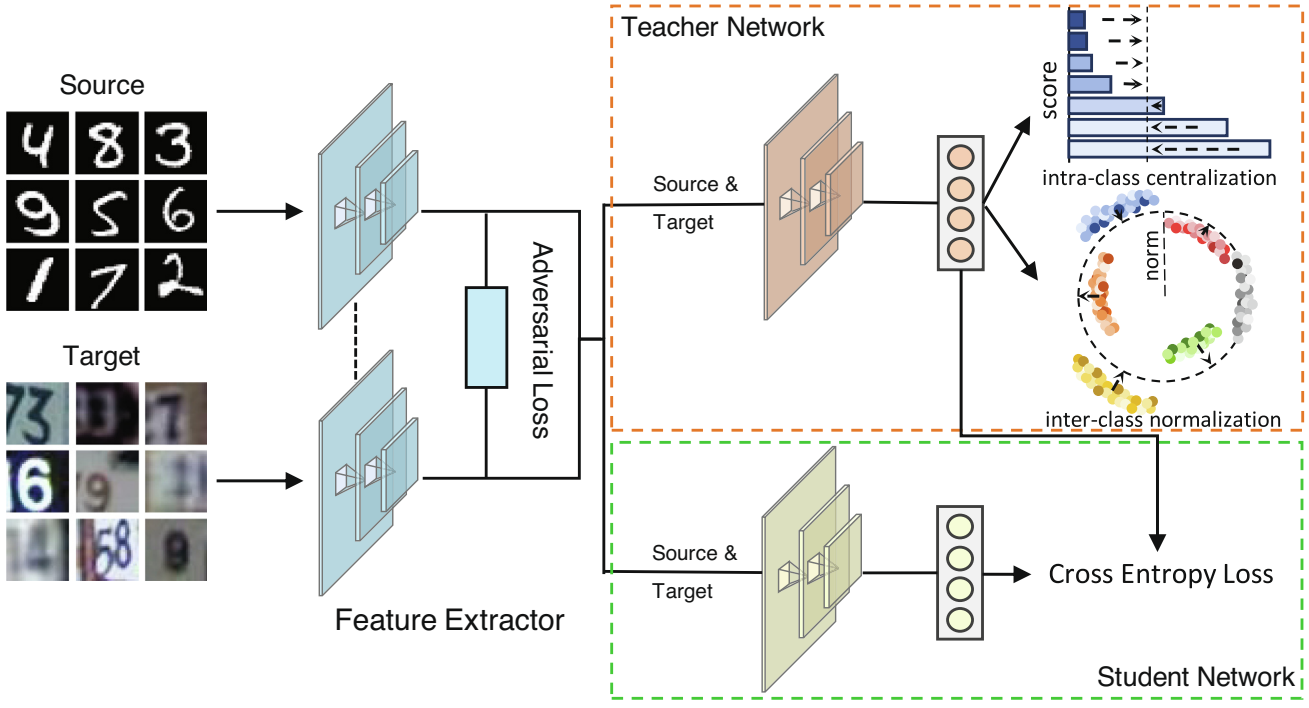


Fig. 1. The framework of the proposed approach. Given source and target samples, a shared CNN model is first utilized to extract features. An adversarial domain discriminator is employed to match the representations of these two domains. A teacher network is trained to produce the soft pseudo labels for student network. The proposed class restriction loss is exploited to train the teacher. The student network shares the feature extraction part with the teacher, and is trained via a Cross Entropy Loss (CEL) with soft pseudo labels. The whole framework is end-to-end trained in a single stage.

representation learning in both domains by leveraging the adversarial learning. A teacher network is trained with the proposed class restriction loss to produce the soft pseudo labels for student network. The student network shares the feature extraction part with the teacher, and the standard Cross Entropy Loss (CEL) optimized with soft pseudo labels is employed to train the classifier. The whole framework is trained in an end-to-end manner. In the following subsections, we will elaborate each part of the proposed method.

3.1. Class restriction loss

The class restriction loss (CRL) has two merits in the model learning. On one hand, by exploiting intra-class centralization, the model introduces a compromising estimation for both easy and hard samples (with high and low confidence scores), which eliminates overestimated (underestimated) scores and evades the overconfidence. On the other hand, the dominance among different categories is reduced with inter-class normalization, which prevents the model from being biased towards a certain dominant class while leaving others vanishing.

3.1.1. Intra-class centralization

The standard cross entropy/conditional entropy loss for labeled/unlabeled training data may cause several issues, when leveraging pseudo labels for training. For example, the conditional entropy loss forces the model to be confident on the unlabeled target data. As a result, the model tends to become overconfident on easy samples, including the false ones. In contrast, the hard samples are very close to the decision boundary, which may increase the uncertainty in model training. Fig. 2 (left) shows an example of data confidence scores from two classes. It can be observed that the misclassified red points could have extremely high scores, resulting in the overfitting to these false samples. The accumulated errors then drive the model to pull true red points towards the boundary. In principle, the samples with both overestimated and underestimated scores should be eliminated. To this end, the centralization is utilized to pull samples together, making the classifier have close confidence for each of them, which guarantees that both easy and hard samples are properly exploited.

We perform the centralization on the logits of samples $lgt(\boldsymbol{x})$, which are the inputs to the final softmax. The logits contain much richer similarity information than the softmax outputs (probabilities), since the latter is constrained to be within the range of 0 and 1, which has little influence on the optimization, when the probabilities are very close to zero or one. Two sets of centers $\{\boldsymbol{c}_s^k\}$ and $\{\boldsymbol{c}_t^k\}$ are kept for source and target domains, respectively, $1 \leqslant k \leqslant C$. The centralization is required to minimize the intra-class variations, thus the loss $\mathscr{L}_c$ can be formulated as follows,

$$\begin{aligned} \mathscr{L}_{c\_s} &= \frac{1}{2}\sum_{i=1}^{b} \| lgt(\boldsymbol{x}_s^i) - \boldsymbol{c}_s^{y_s^i} \|_2^2, \\ \mathscr{L}_{c\_t} &= \frac{1}{2}\sum_{i=1}^{b} \| lgt(\boldsymbol{x}_t^i) - \boldsymbol{c}_t^{\hat{y}_t^i} \|_2^2, \end{aligned} \tag{1}$$

where b is the size of mini-batch, and s, t refer to *source* and *target*, respectively. y_s^i is the ground truth class index of $\boldsymbol{x}_s^i$, indicating that $\boldsymbol{x}_s^i$ belongs to the y_s^ith class. Since the labels for target samples are unavailable, $\hat{y}_t^i$ is the predicted class index of sample $\boldsymbol{x}_t^i$.

The centralization loss $\mathscr{L}_c$ is then utilized to optimize the network by computing its gradient with respect to $lgt(\boldsymbol{x}^i)$ for both domains, which is formulated as

$$\frac{\partial \mathscr{L}_{c\_s}}{\partial lgt(\boldsymbol{x}_s^i)} = lgt(\boldsymbol{x}_s^i) - \boldsymbol{c}_s^{y_s^i}, \frac{\partial \mathscr{L}_{c\_t}}{\partial lgt(\boldsymbol{x}_t^i)} = lgt(\boldsymbol{x}_t^i) - \boldsymbol{c}_t^{\hat{y}_t^i}. \tag{2}$$

After each mini-batch iteration, the centers are updated by the average difference between the center and the samples:

$$\begin{aligned} \Delta\boldsymbol{c}_s^k &= \frac{\sum_{i=1}^{b} \psi(y_s^i = k) \cdot (\boldsymbol{c}_s^k - lgt(\boldsymbol{x}_s^i))}{1 + \sum_{i=1}^{b} \psi(y_s^i = k)}, \\ \Delta\boldsymbol{c}_t^k &= \frac{\sum_{i=1}^{b} \psi(\hat{y}_t^i = k) \cdot (\boldsymbol{c}_t^k - lgt(\boldsymbol{x}_t^i))}{1 + \sum_{i=1}^{b} \psi(\hat{y}_t^i = k)}, \\ \boldsymbol{c}_s^k &= \boldsymbol{c}_s^k - \alpha \cdot \Delta\boldsymbol{c}_s^k, \ \boldsymbol{c}_t^k = \boldsymbol{c}_t^k - \alpha \cdot \Delta\boldsymbol{c}_t^k, \end{aligned} \tag{3}$$

where α is a parameter controlling the learning rate of the centers, and $\psi(\cdot)$ is an indicator that is equal to 1 if $y_s^i = k$ or $\hat{y}_t^i = k$, other-

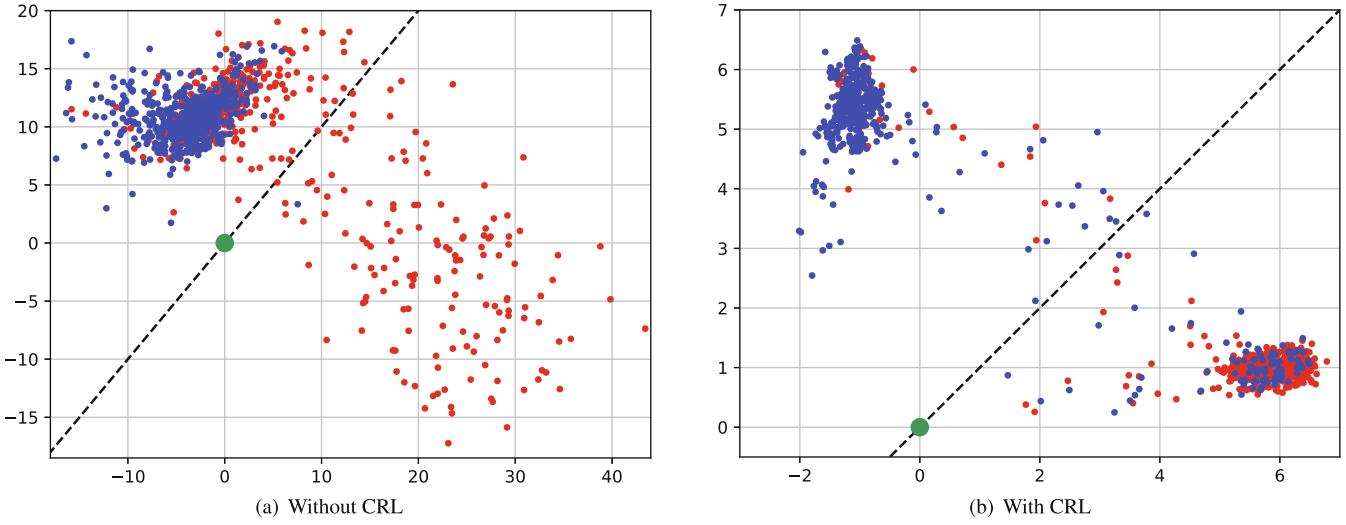


Fig. 2. The illustration of centralization and normalization on MNIST→SVHN. We exploit logits of class "0" (red) and class "1" (blue) to draw the figure, where the logits indicate the classification confidence. The dashed line is the decision boundary. Without CRL (left), red samples are pulled towards the dominant blue class, and the classifier becomes overconfident on samples (the values of logits can reach 20–40). With CRL (right), the samples are gathered, and their logits values are constrained to 4–7.

wise 0. Note that the centers are randomly initialized at the beginning. At last, the overall loss $\mathscr{L}_c$ for two domains is then defined as:

$$\mathscr{L}_c = \rho\mathscr{L}_{c\_s} + \lambda\mathscr{L}_{c\_t}, \tag{4}$$

where ρ and λ are the weights to balance the source and target losses. It is worth noting that the loss is applied to both source and target, since the centralization could also improve the generalization ability for supervised classification task in source domain.

When training, the source and target data have different centers for the same class. The logits contains rich class similarity structure information, which should be similar for source and target. One trivial solution is to directly enforce the same centers for source and target domains (or enforce consistency between them). This constrain imposes a strong assumption that the samples in two domains have identical distributions, which is generally infeasible for source and target. It is expected that the representation distributions between source and target are similar. To this end, the adversarial loss [43] is exploited to learn the indistinguishable logits for source and target domains, so that their centers have similar distributions. The adversarial loss $\mathscr{L}_{adv\_l}$ is formulated as:

$$\begin{aligned}\mathscr{L}_{adv\_l} = &-E_{x_t}[log(D(lgt(\boldsymbol{x}_t)))]\\ &-E_{x_s}[log(1 - D(lgt(\boldsymbol{x}_s)))],\end{aligned} \tag{5}$$

where $D(\cdot)$ is the domain discriminator. The adversarial training is to optimize the following minimax function:

$$\max_{lgt}\min_{D}\mathscr{L}_{adv\_l}. \tag{6}$$

3.1.2. Inter-class normalization

With intra-class centralization, the samples in each category will have similar and low outputted probabilities. Nevertheless, there still exist considerable discrepancies among different categories. Such imbalance would easily bias the model towards dominant classes while leaving others gradually vanishing, as illustrated in Fig. 2. Consequently, the inter-class normalization is devised to balance multiple classes, which is derived from the idea of norm constraints. Given the logits $lgt(\boldsymbol{x}^i)$ of training sample $\boldsymbol{x}^i$, $lgt(\boldsymbol{x}^i)$ is restricted to have the same norm value in each mini-batch, regardless of whether $\boldsymbol{x}^i$ is from source or target. A parameter of targeting norm value P is kept and randomly initialized. The normalization loss function $\mathscr{L}_n$ is formulated as:

$$\mathscr{L}_n = \frac{1}{2}\sum_{i=1}^{b}(\|lgt(\boldsymbol{x}^i)\|_2 - P)^2, \tag{7}$$

where b is the batch size. Note that $\mathscr{L}_n$ is applied to both domains, and the weights ρ, λ are used to balance the source and target. To optimize it, the gradient of $\mathscr{L}_n$ with respect to $lgt(\boldsymbol{x}^i)$ is computed by:

$$\frac{\partial\mathscr{L}_n}{\partial lgt(\boldsymbol{x}^i)} = (1 - \frac{P}{\|lgt(\boldsymbol{x}^i)\|_2}) \cdot lgt(\boldsymbol{x}^i). \tag{8}$$

Similar to the centralization, the targeting norm value P is updated by the average difference between P and the norms of samples, which is given by

$$\Delta P = \frac{1}{1+b} \cdot \sum_{i=1}^{b}(P - \|lgt(\boldsymbol{x}^i)\|_2), P = P - \alpha \cdot \Delta P, \tag{9}$$

where α controls the learning rate of P. Finally, the class restriction loss $\mathscr{L}_{cr}$ is to combine the centralization and normalization as $\mathscr{L}_{cr} = (\mathscr{L}_c + \eta\mathscr{L}_{adv\_l}) + \mathscr{L}_n$, where η is the weight for adversarial loss.

3.2. Soft pseudo label training

The aforementioned CRL loss is further incorporated into a teacher-student framework. The CRL loss is utilized in teacher to facilitate the generation of soft pseudo labels, which is motivated by the knowledge distillation [67]. Compared to the one-hot label, the model could benefit from two aspects. On one hand, the probabilities contain rich information on the correlations among classes, which can boost the model significantly. On the other hand, using soft probabilities will also reduce the confidence of the model, which further prevents it from overfitting false assigned labels.

In particular, the student network shares the basic feature extraction part with the teacher, and employs Cross Entropy Loss (CEL) for optimization. With the shared features, it can enhance the basic representation learning, which in turn helps generate more accurate pseudo labels. The loss function for student network is thus defined as:

$$\mathscr{L}_{stud} = -\sum_{i=1}^{b}\sum_{k=1}^{C}\tilde{p}^{i,k}log(\hat{p}^{i,k}), \tag{10}$$

where $\tilde{p}^{i,k}$ and $\hat{p}^{i,k}$ denote the output probabilities from teacher and student for sample $\boldsymbol{x}^i$ belonging to class k, respectively, b is the batch size, and C is the number of classes.

3.3. Optimization

The proposed network is trained in an end-to-end manner. For the CNN feature extraction part, an adversarial loss $\mathscr{L}_{adv\_f}$ is additionally employed to make the representations of two domains indistinguishable. It is calculated as follows,

$$\begin{aligned}\mathscr{L}_{adv\_f} = &-E_{\boldsymbol{x}_t}[log(D(F(\boldsymbol{x}_t)))]\\ &-E_{\boldsymbol{x}_s}[log(1 - D(F(\boldsymbol{x}_s)))],\end{aligned} \tag{11}$$

and is optimized by following the minimax scheme as

$$\max_{F}\min_{D}\mathscr{L}_{adv\_f}, \tag{12}$$

where F denotes the parameters of the shared CNN model.

For teacher network, in addition to the proposed CRL, the cross entropy loss $\mathscr{L}_{cls}$ and conditional entropy loss $\mathscr{L}_{cond}$ are applied to the source and target data, respectively, which are defined as follows,

$$\begin{aligned}\mathscr{L}_{cls} &= -\sum_{i=1}^{b}\sum_{k=1}^{C}\psi(k = y_s^i) \cdot log(\tilde{p}_s^{i,k}),\\ \mathscr{L}_{cond} &= -\sum_{i=1}^{b}(\tilde{\boldsymbol{p}}_t^i)^\top log(\tilde{\boldsymbol{p}}_t^i),\end{aligned} \tag{13}$$

where $\tilde{p}_s^{i,k}$ is the predicted probability for source sample $\boldsymbol{x}_s^i$ belonging to class k, $\tilde{\boldsymbol{p}}_t^i$ is the predicted probability vector for target sample $\boldsymbol{x}_t^i$, and $\psi(\cdot)$ is an indicator function which equals to 1 if k is the ground truth class label y_s^i of $\boldsymbol{x}_s^i$, otherwise 0. The training loss $\mathscr{L}_{tch}$ for the teacher network is to combine the CRL with the aforementioned two losses:

$$\mathscr{L}_{tch} = \rho\mathscr{L}_{cls} + \lambda\mathscr{L}_{cond} + \mathscr{L}_{cr}, \tag{14}$$

where ρ and λ are the same parameters as in Eqn. (4) to balance the source and target during training.

Finally, the overall training objective $\mathscr{L}$ of the proposed network is formulated as follows,

$$\mathscr{L} = \mathscr{L}_{tch} + \lambda\mathscr{L}_{stud} + \eta\mathscr{L}_{adv\_f}, \tag{15}$$

Fig. 3. Sample images of each dataset. The MNIST, MNIST-M, DIGITS and SVHN contain images of digit numbers from different domains. CIFAR-10 and STL-10 are sets of small images and share 9 common classes. Office-31 has 31 classes in 3 domains: Amazon, DSLR and Webcam. VisDA-2017 consists of images from 12 classes in 2 domains, i.e., the synthetic set and realistic set.

where the student network has the same training weight λ as the target domain in teacher network, and η is the weight for adversarial loss.

It is worth noting that the proposed class restriction loss does not impose any penalization to trivial solutions, i.e. all samples form a single cluster or the normalization loss shrinks to zero. However, such situation would not occur since the training samples in source domain generally provide sufficient supervision for model learning. With these labeled source samples and the optimization of cross entropy loss, the method can first learn good parameters in initial stages, and further avoid the model collapse.

4. Experiments

In this section, extensive evaluations have been conducted on several datasets, including four digits datasets (MNIST [68], MNIST-M [69], SVHN [70], SYN-DIGITS [69]), two small image datasets (CIFAR-10 [71], STL-10 [72]), the standard Office31 [73] and VisDA-2017 [74]. Several sample images of each dataset are illustrated in Fig. 3.

4.1. Datasets

MNIST and MNIST-M datasets are handwritten digits datasets, where the latter is constructed by blending the former with random color patches from BSD500 [75]. SVHN consists of crops of colored street house numbers. SYN-DIGITS contains the synthetic numbers generated from Windows fonts with various text, positioning, orientation, and so on. We follow the standard training sets as in [69,33], and consider adaptation in four directions: MNIST→MNIST-M, SVHN→MNIST, MNIST→SVHN, and SYN DIGITS→SVHN. Among them, the adaptation of MNIST→SVHN is the most challenging one, which requires adapting models learned from black-and-white digits to color digits. Therefore, the evaluations of our method are conducted on this task.

These two small image datasets CIFAR-10 and STL-10 contain 9 overlapping classes. Following [33], the non-overlapping classes ("frog" in CIFAR-10 and "monkey" in STL-10) are removed, and the adaptation is performed on both directions, i.e., CIFAR↔STL.

Office31 consists of 4,110 images from 31 classes in 3 domains: Amazon (A), Webcam (W) and DSLR (D). We follow the common protocol as in [27] to evaluate the methods transductively, where the performances are reported on the target training set. All the adaptation directions are evaluated in the experiments.

VisDA-2017[1] is a well-known Visual Domain Adaptation challenge dataset, which focuses on the difficult simulation-to-real task. It has two parts, 1) the synthetic set which contains 2D renderings of 3D models generated from different angles and with different lighting conditions, and 2) the photo-realistic or real-image validation set. Following [14], the accuracy on validation set is reported.

4.2. Implementation details

For digits and image (CIFAR and STL) datasets, the same CNN architecture used in [33] is utilized, which contains 9 convolutional layers. The teacher and student networks begin from the 7*th* convolutional layer. The adversarial discriminators for both features and logits consist of two fully connected layers: one has 100 hidden units and the other is the discriminator output. The instance normalization [76] is exploited as an image pre-processing step. The Adam optimizer [77] is adopted with the learning rate of 0.001 and the moment of 0.5 for training. The batch size is 64. Hyper-parameter η is set to 0.01. For the domain weights ρ and λ, the target weight λ is gradually increased from 0 to w during the first 20 epochs, and the source weight ρ is decreased from 1 to w during the first 40 epochs, where w is equal to 0.2 for MNIST→SVHN and 0.5 for other adaptations.

For Office31 and VisDA-2017, ResNet-50 [7] pre-trained on ImageNet is utilized as the backbone. The teacher and student networks begin from the *res5a* layer. The adversarial discriminator for features consists of four fully connected layers with the size of 256, 3072, 2048 and 1, respectively. Following [12,14], the SGD optimizer is adopted with the moment of 0.9 and its learning rate is adjusted using an annealing strategy: $l_p = \frac{l_0}{(1+\gamma p)^\beta}$, where p is the training progress linearly changing from 0 to 1, and $l_0 = 0.01, \gamma = 10, \beta = 0.75$. The discriminator weight η is set according to a progressive strategy, which is increased from 0 to 1 by $\frac{2}{1+\exp(-\delta p)} - 1, \delta = 10$. For the domain weights ρ and λ, the settings are the same as MNIST→SVHN adaptation. When compared with the state-of-the-art methods, each adaptation is repeated three times and the average performance is reported for all adaptation experiments.

4.3. The effect of training weight in domain adaptation

The balancing weights for source and target domains (i.e., the hyper-parameters ρ and λ) play an important role in model training. As mentioned in previous sections, existing domain adaptation methods which leverage pseudo labels for training have to set small weights for the target loss due to the noises. The target model is thus trained insufficiently, leading to a sub-optimal solution. An experiment of VADA [33] is conducted on MNIST→SVHN to verify this. For comparison, the fixed small target weight design in VADA ($\lambda = 0.01$) is simply replaced with our weight adjustment strategy, i.e., increasing the target weight λ from 0 to w and

[1] http://ai.bu.edu/visda-2017/.

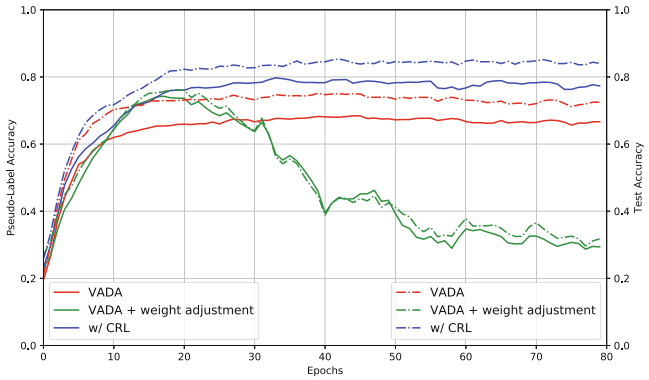


Fig. 4. Pseudo label accuracy (solid lines) and Test accuracy (dash-dot lines) comparison of VADA with/without source and target weight adjustment and with CRL on the adaptation of MNIST→SVHN. The weight adjustment is described in Section 4.3, and without adjustment indicates a fixed small weight for target data. By increasing the weight of target, VADA achieves a higher peak but the model soon collapses due to the noisy pseudo labels.

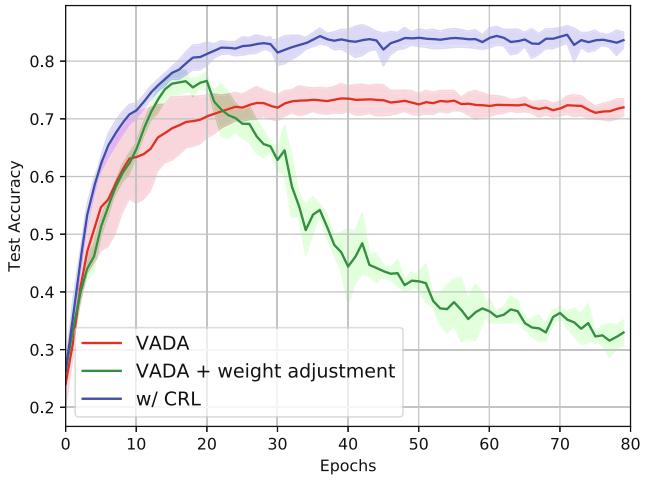


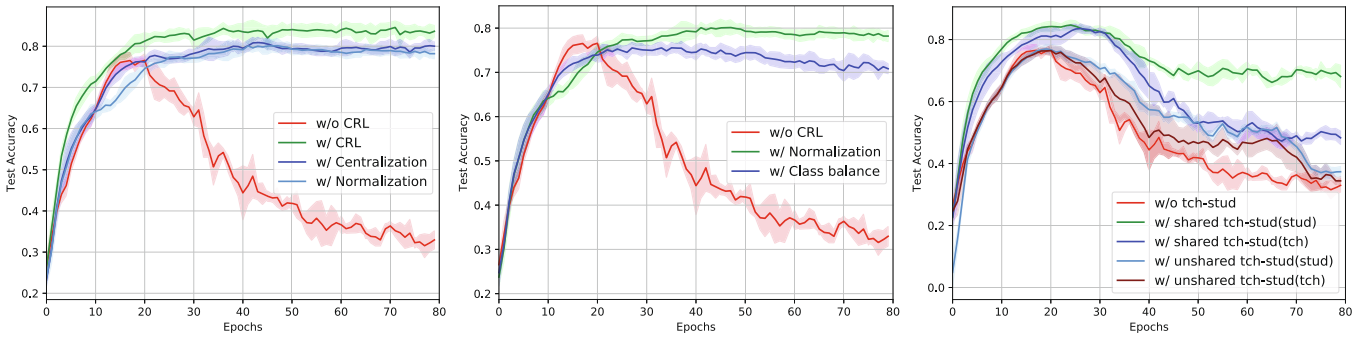
Fig. 5. Test accuracy comparison on the adaptation of MNIST→SVHN. The error bar (std) is attached.

Figs. 4 and 5 demonstrate the pseudo label accuracy and test accuracy during training. The weight adjustment strategy is also adopted by the proposed CRL. When training with small target weights, VADA exhibits a stable accuracy curve against the number of epochs, since the target samples have very little influence on the model training. In contrast, increasing the target weights could achieve a much higher peak accuracy than original VADA. Although the model soon collapses due to memorizing target samples with noisy labels, it clearly reveals the weakness of VADA that small target weights will lead to sub-optimal model. The proposed CRL achieves much higher accuracy and keeps the training stable simultaneously.

4.4. Analysis and evaluation

Class Restriction Loss. Here we evaluate the effect of the centralization and normalization of CRL as well as the whole CRL on adaptation MNIST→SVHN. The weight adjustment strategy is utilized while the teacher-student architecture is removed for fair comparison. The performance curves are demonstrated in Fig. 6 (a). Without CRL, although high peak accuracy is achieved, the model soon collapses and performs very poor on test set. Such method is infeasible for practical scenarios, because it is impossible to know when to stop the model training for achieving high recognition accuracy. When the centralization or normalization is employed, a high level accuracy is obtained without performance degradation. Furthermore, CRL integrates them together, achieving higher and stable results. This indicates that the centralization and normalization are complementary, which make the balance within or between classes, beneficial for the model. Fig. 7 demonstrates the effect of the proposed CRL. Here, the phenomenon of assigning pseudo labels to a dominant class is restricted by utilizing CRL.

Comparison with Class Balance Regularization. The proposed normalization forces samples from different categories gather on a hypersphere. As a result, the classifier will not be biased to any dominant class. To verify the effectiveness of the normalization, we compare it with the class balance regularization loss $\mathscr{L}_p$ [53]. $\mathscr{L}_p$ introduces a prior probability distribution $\boldsymbol{p}$, which is the class distribution of all source training data. Generally speaking, it assumes that the distribution of classes in target domain is similar to the source domain. Therefore, it minimizes the KL-divergence



(a) Evaluations on CRL (b) Evaluations on normalization (c) Evaluations on Teacher-Student architecture

Fig. 6. Method evaluations on the adaptation of MNIST→SVHN. (a) Evaluations on the Class Restriction Loss (CRL). We evaluate the centralization and normalization separately. (b) Performance comparison of the proposed normalization and the class balance regularization loss $\mathscr{L}_p$ [53]. The normalization exhibits much higher result than the balance regularization. (c) Evaluations on the soft pseudo label training (teacher-student architecture). We further evaluate another architecture where the teacher and student networks do not share the feature extraction part. Both performances from teacher (tch) and student (stud) are reported.

decreasing the source weight ρ from 1 to w ($w = 0.2$ for MNIST→SVHN). This modified VADA acts as the baseline in the following evaluations.

from the predicted class distribution $\bar{\boldsymbol{p}}(\mathscr{X}_t)$ to $\boldsymbol{p}$, so that the model will not be biased to certain classes. The loss $\mathscr{L}_p$ is formulated as:

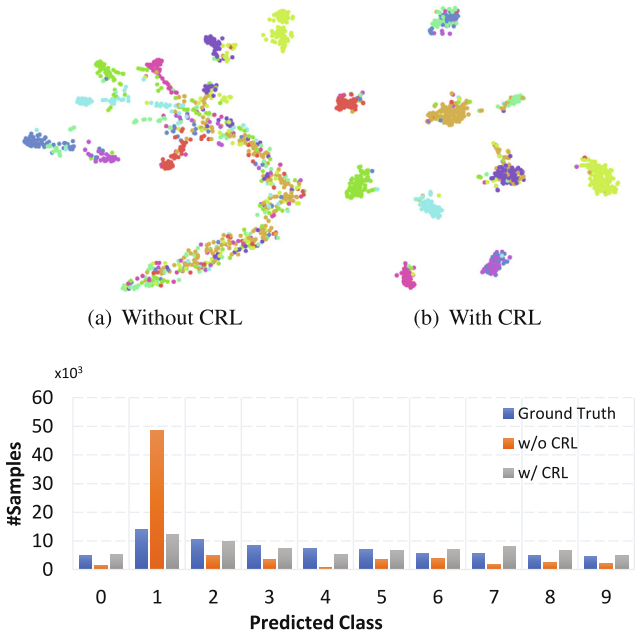


Fig. 7. T-SNE visualization comparisons of logits for target data when performing adaptation MNIST →SVHN. Without Class Restriction Loss (CRL), the model is biased to a dominant class (class "1") due to noisy pseudo labels, as shown in the top left. The proposed method can prevent the model from fitting noisy data by equally treating each sample and each class, as illustrated in the top right. The table lists the number of predicted samples for each class.

$$\mathscr{L}_p = \sum_{k=1}^{C} p^k log \frac{p^k}{\bar{p}^k(\mathscr{X}_t)}, \tag{16}$$

where p^k is the prior probability for class k, and $\bar{p}^k(\mathscr{X}_t)$ is the predicted mean probability for class k in target data, which is approximated in each mini-batch during training. In experiments, the weight adjustment strategy is used and the teacher-student architecture is ignored. The evaluation is carried out on MNIST→SVHN adaption.

Fig. 6(b) illustrates the accuracy comparison of two methods. It clearly demonstrates that both methods can effectively maintain the performance when increasing the weights of target data. Nevertheless, the proposed normalization exhibits much better performance than the balance regularization. It is not surprising since the assumption of same class distribution in $\mathscr{L}_p$ does not hold for the MNIST→SVHN adaptation. In contrast, our normalization is not restricted to this strong assumption, and can work well in various situations.

Soft Pseudo Label Training. The soft pseudo label training (i.e. teacher-student architecture) is evaluated on MNIST→SVHN adaptation. Similarly, the weight adjustment strategy is utilized and the CRL module is removed. In addition to the architecture introduced in Section 3.2, which shares the feature extraction part, another framework is also compared, where student network is identical and independent to the teacher. The accuracies from both student and teacher networks are reported.

The comparison results are illustrated in Fig. 6(c). The performances of all methods are dropped after reaching the peak. The reason is that CRL is removed in this experiment, which prevents the model from overfitting false pseudo labels. Nevertheless, teacher-student architecture improves the performance and significantly slows down the decrease. Another observation is that the shared feature extraction architecture outperforms the unshared one with a large margin. This is because shared features improve the representation learning, which in turn helps produce more accurate pseudo labels. In addition, the student network performs better than the teacher, which has been discussed in knowledge distillation works [67,52] that soft label training can decrease the confidence of the student model.

Evaluation of Domain Weights. Here we evaluate the effect of parameter w. w controls the weights of source and target training losses. Our weight adjustment strategy decreases the source weight ρ from 1 to w, and increases the target weight λ from 0 to w. The results on MNIST→SVHN are reported in Table 1. We can see that both small and large w will lead to the performance drop. The reason is that small w will make the model trained insufficiently, which is similar to VADA, while large w will make the model completely fit to target samples. Therefore, an appropriate w value is important.

4.5. An ablation study

In this section, ablation study is conducted to evaluate the effect of each component in the network. The basic method is to directly train the domain adaptation network with adversarial domain discriminator and cross entropy/conditional entropy loss. Different combinations of four components are evaluated: source/target weight adjustment, centralization loss, normalization loss, and teacher-student architecture. In addition, two results for each adaptation are reported, that is, the accuracy of the epoch with the best performance, and the accuracy of the last epoch. The extra evaluation of the last accuracy indicates the degree of learning with noisy data. The more fitting noisy data, the more model get a decrease on the last accuracy. The experiments are conducted on four digits adaptations and two image adaptations.

The performance comparison with different combinations are listed in Table 2. The weight adjustment strategy improves the accuracy in most cases except the last accuracy in MNIST→SVHN. It is because this adaptation task is challenging and it produces more noisy labels. It also does not work well on CIFAR→STL as the training number of STL is too small (4,500 images). Directly training on labeled STL gives an accuracy of 70.0%, which is much lower than the performance of adaptation (77.6%). Therefore, neither increasing the weight of STL nor assigning pseudo labels to STL can bring the improvement. In addition, both centralization and normalization are effective for domain adaptation. They successfully boost the performance on all datasets, including those datasets with very high baselines (e.g., SVHN→MNIST and DIGITS→SVHN). In most cases, the centralization performs a little better than the normalization. Integrating them further boosts the performance. Similar to the centralization and normalization, the teacher-student framework also contributes the performance improvement except the challenging MNIST→SVHN. By combining the four components together, the method considerably improves the accuracy, indicating that CRL and the teacher-student architecture are complementary to each other.

Table 1
Test accuracies of our method on MNIST→SVHN with different w values.

w	0.01	0.1	0.2	0.3	0.4	0.5
Accuracy	78.3	87.5	93.2	91.2	85.8	84.7

Table 2
Performance contribution of each component. "Weight" denotes the weight adjustment for target/source samples, "CL" and "NL" refer to the centralization and normalization loss, respectively, and "T-S" means the teacher-student architecture. "M-M" is the MNIST-M dataset. Both the best and last performances of the models are reported.

Method				Adaptations									
Weight	CL	NL	T-S	MNIST→SVHN		SVHN→MNIST		DIGITS→SVHN		MNIST→M-M		STL→CIFAR		CIFAR→STL	
				best	*last*	*best*	*last*	*best*	*last*	*best*	*last*	*best*	*last*	*best*	*last*
				75.6	73.0	94.8	94.2	95.0	94.8	96.2	95.8	71.7	71.0	79.5	77.6
✓				80.5	34.1	97.9	97.9	95.6	95.2	98.8	98.7	73.5	73.3	79.7	77.6
✓	✓			84.5	84.0	99.1	99.0	96.0	95.9	98.9	98.7	75.5	75.2	80.1	79.5
✓		✓		82.3	81.1	99.1	99.1	95.9	95.7	98.8	98.7	75.2	74.8	80.3	79.3
✓	✓	✓		85.7	85.7	99.4	99.4	96.2	96.0	99.0	98.9	76.7	76.5	80.6	80.4
✓			✓	86.4	73.2	99.2	99.1	95.5	95.2	98.9	98.8	75.1	74.8	80.7	80.3
✓	✓		✓	91.2	90.6	99.3	99.3	96.1	96.0	99.0	98.9	76.9	76.4	80.7	80.4
✓		✓	✓	90.5	89.3	99.3	99.2	96.0	96.0	99.0	99.0	76.5	76.2	80.6	80.2
✓	✓	✓	✓	93.8	93.2	99.5	99.4	96.2	96.2	99.0	99.0	77.4	77.0	81.0	80.6

Table 3
Test set accuracy comparisons with the state-of-the-art approaches on digits and image adaptations. The proposed method outperforms other approaches on most tasks. Compared with pseudo-label based methods (e.g., ATT and DIRT-T), the proposed method exhibits significant improvement over them.

Source Target	MNIST SVHN	SVHN MNIST	DIGITS SVHN	MNIST MNIST-M	STL CIFAR	CIFAR STL
MMD [36]	–	71.1	88.0	76.9	–	–
DANN [12]	35.7	71.1	90.3	81.5	–	–
DRCN [78]	40.1	82.0	–	–	58.7	66.4
DSN [79]	–	82.7	91.2	83.2	–	–
kNN-Ad [80]	40.3	78.8	–	86.7	–	–
ATT [28]	52.8	86.2	92.9	94.2	–	–
Π-model [81][1]	97.0	99.3	97.1	–	70.0	80.0
CAT [41]	–	98.8	–	–	–	–
GPDA [82]	–	98.2	–	–	–	–
SWD [42]	–	98.9	–	–	–	–
RCA [45]	89.2	99.3	96.2	**99.5**	**77.8**	**81.7**
w/o instance normalization						
DIRT-T [33]	54.5	**99.4**	96.1	98.9	75.3	–
Co-DA [83]	52.0	98.3	96.1	99.0	76.4	81.1
Co-DA$^+$ [83][2]	60.8	**99.4**	**96.5**	99.1	76.6	–
Ours	**78.7**	**99.4**	96.3	99.3	**77.8**	80.7
w/ instance normalization						
DIRT-T [33]	76.5	**99.4**	96.2	98.7	73.3	–
Co-DA [83]	81.7	98.6	96.0	97.5	74.7	80.6
Co-DA$^+$ [83]2	88.0	**99.4**	**96.5**	98.7	74.8	–
Ours	**93.1**	**99.4**	96.2	99.0	77.1	80.6

[1] Π-model exploits additional data augmentation to achieve high performance.
[2] Co-DA$^+$ indicates the combination of Co-DA and DIRT-T.

4.6. Comparisons with state-of-the-art methods

Digits and Image Adaptations. We evaluate the effect of instance normalization. The training is repeated three times and the average accuracy of the last epoch is reported. The results are listed in Table 3. Note that additional data augmentation (e.g., translation, flipping and affine augmentation) is utilized in Π-model [81] to obtain high performance. Co-DA$^+$ means Co-DA [83] is combined with DIRT-T [33], which is a two-stage optimization. Generally, the proposed approach achieves good performance. For example, with instance normalization, it reaches 93.1% on digits adaptation MNIST→SVHN, which outperforms the best competitor RCA by 3.9%. Compared with pseudo-label based methods, ATT [28] and DIRT-T [33], the proposed method significantly improves the performance, because the proposed CRL resists the completely fitting to noises. The performance improvement of CIFAR→STL is minor, where RCA [45] is slightly better than us. This is because STL is too small to provide valuable information with pseudo labels.

Office31 Adaptations. Different from digits and image experiments, we conduct experiments on Office31, i.e., training and testing on the same target set. For fair comparison, all methods adopt the same backbone network (ResNet-50 [7]). The results are presented in Table 4. The proposed method produces competitive performances. For adaptations like W→A and D→A, our method has worse results than the best competitor CAN [40]. The reason is that CAN adopts an alternative pseudo-label update strategy and further considers the class information to constrain the features, which leads to the class-aware alignment between source and target. Such constraint is not included in our method, which is not the focus of this paper. Overall, the proposed network achieves the second highest accuracy 88.3% after CAN.

VisDA-2017 Adaptations. The performance comparison on VisDA-2017 validation set is listed in Table 5. Note that GPDA, SAFN and SWD adopt ResNet-101 as the backbone, while others utilize ResNet-50. Our method outperforms the generative pixel-level adaptation method GTA [85] and the feature-class cross-covariance adaptation method CDAN+E [14], where the former has a very complex design in architecture and the latter requires additional memory cost for the multi-linear map. Among all the methods, SWD [42] achieves promising performance which is better than ours. The reason is that they adopt the task-specific classifiers to detect optimized target samples for distribution alignment, which is similar to the class-aware alignment.

Table 4
Accuracy comparisons with the state-of-the-art approaches on Office31 dataset. We evaluate all the adaptation directions to verify the effectiveness of the proposed method, which produces competitive performance. It achieves the second highest accuracy after CAN.

Method	A→W	W→A	A→D	D→A	W→D	D→W	Avg.
ResNet50 [7]	73.5	59.8	76.5	56.7	99.0	93.6	76.5
DAN [27]	80.5	62.8	78.6	63.6	99.6	97.1	80.4
RTN [84]	84.5	64.8	77.5	66.2	99.4	96.8	81.6
DANN [12]	79.3	63.2	80.7	65.3	99.6	97.3	80.9
ADDA [18]	86.2	68.9	77.8	69.5	98.4	96.2	82.9
JAN [39]	86.0	70.7	85.1	69.2	99.7	96.7	84.6
iCAN [16]	92.5	69.9	90.1	72.1	**100.0**	98.8	87.2
CDAN [14]	93.1	68.0	89.8	70.1	**100.0**	98.2	86.6
CDAN+E [14]	94.1	69.3	92.9	71.0	**100.0**	98.6	87.7
CAN [40]	**94.5**	**77.0**	**95.0**	**78.0**	99.8	**99.1**	**90.6**
SAFN [64]	90.1	70.2	90.7	73.0	99.8	98.6	87.1
CAT [41]	94.4	70.2	90.8	72.2	**100.0**	98.0	87.6
Ours	94.1	70.1	94.9	72.4	**100.0**	98.5	88.3

Table 5
Accuracy comparisons with the state-of-the-art approaches on VisDA-2017 dataset. The performance is reported on the validation set. GPDA [82], SAFN [64] and SWD [42] adopt ResNet-101 while others adopt ResNet-50.

Method	Accuracy
Source-only	40.2
JAN [39]	61.6
GTA [85]	69.5
CDAN [14]	66.8
CDAN+E [14]	70.0
GPDA* [82]	73.3
SAFN* [64]	76.1
SWD* [42]	**76.4**
Ours	72.1

(a) DIRT-T (b) Ours

Fig. 8. T-SNE visualization of last hidden layer for adaptation MNIST→SVHN. Red points are from source dataset (MNIST), and blue points indicate target samples (SVHN).

4.7. Visualization

Fig. 8 depicts the t-SNE [86] visualization of the last hidden layer for MNIST→SVHN adaptation. 1,000 samples are randomly selected from 10 classes (100 samples per class) in each domain (MNIST as source and SVHN as target). We can see that our method can better adapt the model to the target samples than DIRT-T [33], which has a better clustering result on the target.

5. Conclusion

In this paper, a novel class restriction loss is proposed for unsupervised domain adaptation, which leverages noisy pseudo labels for learning reliable target models. The proposed method treats each training sample and each class equally by employing two complementary types of restrictions on the training data: the intra-class centralization and inter-class normalization. The centralization pulls samples within each class together, making easy and hard points contribute equally to the model. The normalization forces the same norm values for all samples, indicating the balance between classes. The proposed CRL prevents the model from overfitting samples assigning false pseudo labels. It is further incorporated into a specially designed teacher-student architecture. Experiments conducted on several standard benchmarks validate the effectiveness of the proposed model.

CRediT authorship contribution statement

Qi He: Conceptualization, Methodology, Formal analysis, Investigation, Visualization, Writing - original draft. **Qi Dai:** Conceptualization, Methodology, Formal analysis, Investigation, Writing - original draft, Writing - review & editing. **Xiao Wu:** Methodology, Writing - review & editing, Supervision. **Jun-Yan He:** Methodology, 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.

Acknowledgment

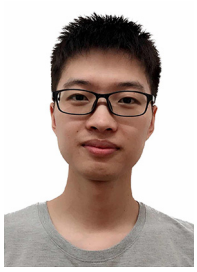
This work was supported in part by the National Natural Science Foundation of China (Grant No: 61772436), Sichuan Science and Technology Program (Grant No. 2020YJ0207), Foundation for Department of Transportation of Henan Province, China (2019J-2-2), and Grant of Institute of Applied Physics and Computational Mathematics, Beijing (Grant No. HXO2020-118).

References

[1] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks?, Proc Adv. Neural Inf. Process. Syst., 2014.
[2] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, L. Fei-Fei, Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (2015) 211–252.
[3] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, A. Zisserman, The kinetics human action video dataset, arXiv preprint arXiv:1705.06950 (2017).
[4] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 1097–1105.
[5] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Proc. Intl. Conf. Learn. Represent., 2015.
[6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2015, pp. 1–9.
[7] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2016, pp. 770–778..

[8] D. Tran, L. Bourdev, R. Fergus, L. Torresani, M. Paluri, Learning spatiotemporal features with 3d convolutional networks, in: Proc. IEEE Intl. Conf. Comp. Vis., 2015, pp. 4489–4497.
[9] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2017, pp. 4700–4708.
[10] J. Carreira, A. Zisserman, Quo vadis, action recognition? a new model and the kinetics dataset, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2017, pp. 6299–6308.
[11] M. Gheisari, M.S. Baghshah, Unsupervised domain adaptation via representation learning and adaptive classifier learning, Neurocomputing 165 (2015) 300–311.
[12] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, V. Lempitsky, Domain-adversarial training of neural networks, J. Mach. Learn. Res. 17 (2016), 2096–2030.
[13] X. Yang, T. Zhang, C. Xu, M.-H. Yang, Boosted multifeature learning for cross-domain transfer, ACM Trans. Multimed. Comput. Commun. Appl. 11 (2015) 1–18.
[14] M. Long, Z. Cao, J. Wang, M.I. Jordan, Conditional adversarial domain adaptation, Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 1640–1650.
[15] L. Zhao, Z. Chen, L.T. Yang, M.J. Deen, Z.J. Wang, Deep semantic mapping for heterogeneous multimedia transfer learning using co-occurrence data, ACM Trans. Multimed. Comput. Commun. Appl. 15 (2019) 1–21.
[16] W. Zhang, W. Ouyang, W. Li, D. Xu, Collaborative and adversarial network for unsupervised domain adaptation, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2018, pp. 3801–3809.
[17] J. Tang, X. Shu, Z. Li, G.-J. Qi, J. Wang, Generalized deep transfer networks for knowledge propagation in heterogeneous domains, ACM Trans. Multimed. Comput. Commun. Appl. 12 (2016) 1–22.
[18] E. Tzeng, J. Hoffman, K. Saenko, T. Darrell, Adversarial discriminative domain adaptation, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2017, pp. 7167–7176.
[19] S. Sun, B. Zhang, L. Xie, Y. Zhang, An unsupervised deep domain adaptation approach for robust speech recognition, Neurocomputing 257 (2017) 79–87.
[20] Y. Li, C. Lin, H. Li, W. Hu, H. Dong, Y. Liu, Unsupervised domain adaptation with self-attention for post-disaster building damage detection, Neurocomputing 415 (2020) 27–39.
[21] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, J.W. Vaughan, A theory of learning from different domains, Mach. Learn. 79 (2010) 151–175.
[22] J. Liang, D. Hu, J. Feng, Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation, Proc. Int. Conf. Mach. Learn., 2020, pp. 6028–6039.
[23] A. Kumar, T. Ma, P. Liang, Understanding self-training for gradual domain adaptation, Proc. Int. Conf. Mach. Learn, 2020, pp. 5468–5479.
[24] X. Ma, T. Zhang, C. Xu, Deep multi-modality adversarial networks for unsupervised domain adaptation, IEEE Trans. Multimed. 21 (2019) 2419–2431.
[25] B. Gong, K. Grauman, F. Sha, Learning kernels for unsupervised domain adaptation with applications to visual object recognition, Int. J. Comput. Vis. 109 (2014) 3–27.
[26] Z. Wang, B. Du, Y. Guo, Domain adaptation with neural embedding matching, IEEE Trans. Neural Netw. Learn. Syst. 31 (2019) 2387–2397.
[27] M. Long, Y. Cao, J. Wang, M.I. Jordan, Learning transferable features with deep adaptation networks, Proc. Intl. Conf. Mach. Learn. 37, 2017, pp. 97–105.
[28] K. Saito, Y. Ushiku, T. Harada, Asymmetric tri-training for unsupervised domain adaptation, Proc. Intl. Conf. Mach. Learn. 70 (2017) 2988–2997.
[29] M. Mirza, S. Osindero, Conditional generative adversarial nets, arXiv preprint arXiv:1411.1784 (2014)..
[30] R. Shen, J. Yao, K. Yan, K. Tian, C. Jiang, K. Zhou, Unsupervised domain adaptation with adversarial learning for mass detection in mammogram, Neurocomputing 393 (2020) 27–37.
[31] F. Liu, G. Zhang, J. Lu, Heterogeneous domain adaptation: An unsupervised approach, IEEE Trans. Neural Netw. Learn. Syst. 31 (2020) 5588–5602.
[32] F. Liu, G. Zhang, J. Lu, Multi-source heterogeneous unsupervised domain adaptation via fuzzy-relation neural networks, IEEE Trans. Fuzzy Syst. (2020), 1–1.
[33] R. Shu, H.H. Bui, H. Narui, S. Ermon, A dirt-t approach to unsupervised domain adaptation, Proc. Int. Conf. Learn. Represent. (2018).
[34] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, L. Fei-Fei, Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels, in: Proc. Intl. Conf. Mach. Learn., 2018, pp. 2304–2313.
[35] Z. Ding, S. Li, M. Shao, Y. Fu, Graph adaptive knowledge transfer for unsupervised domain adaptation, Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 37–52.
[36] A. Gretton, K.M. Borgwardt, M.J. Rasch, B. Schölkopf, A. Smola, A kernel two-sample test, J. Mach. Learn. Res. 13 (2012) 723–773.
[37] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, T. Darrell, Deep domain confusion: Maximizing for domain invariance, arXiv preprint arXiv:1412.3474 (2014).
[38] B. Sun, K. Saenko, Deep coral: Correlation alignment for deep domain adaptation, in: Proc. Eur. Conf. Comput. Vision, 2016, pp. 443–450.
[39] M. Long, H. Zhu, J. Wang, M.I. Jordan, Deep transfer learning with joint adaptation networks, Proc. Intl. Conf. Mach. Learn. 70, 2017, pp. 2208–2217.
[40] G. Kang, L. Jiang, Y. Yang, A.G. Hauptmann, Contrastive adaptation network for unsupervised domain adaptation, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2019, pp. 4893–4902.
[41] Z. Deng, Y. Luo, J. Zhu, Cluster alignment with a teacher for unsupervised domain adaptation, in: Proc. IEEE Intl. Conf. Comp. Vis., 2019, pp. 9944–9953.
[42] C.-Y. Lee, T. Batra, M.H. Baig, D. Ulbricht, Sliced wasserstein discrepancy for unsupervised domain adaptation, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2019, pp. 10285–10295.
[43] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, Proc. Adv. Neural Inf. Process. Syst., 2014, pp. 2672–2680.
[44] K. Saito, K. Watanabe, Y. Ushiku, T. Harada, Maximum classifier discrepancy for unsupervised domain adaptation, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2018, pp. 3723–3732.
[45] S. Cicek, S. Soatto, Unsupervised domain adaptation via regularized conditional alignment, in: Proc. IEEE Intl. Conf. Comp. Vis., 2019, pp. 1416–1425.
[46] Q. Shi, M. Liu, X. Liu, P. Liu, P. Zhang, J. Yang, X. Li, Domain adaption for fine-grained urban village extraction from satellite images, IEEE Geosci. Remote Sens. Lett. 17 (2019) 1430–1434.
[47] G. Patrini, A. Rozza, A.K. Menon, R. Nock, L. Qu, Making deep neural networks robust to label noise: A loss correction approach, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2017, pp. 1944–1952.
[48] A. Ghosh, H. Kumar, P. Sastry, Robust loss functions under label noise for deep neural networks., in: Proc. of the AAAI Conference on Artificial Intelligence, 31, Association for the Advancement of Artificial Intelligence, 2017, pp. 1919–1925.
[49] J. Goldberger, E. Ben-Reuven, Training deep neural-networks using a noise adaptation layer, in: Proc. Intl. Conf. Learn. Represent., 2017.
[50] J. Yin, B. Chen, Y. Li, Highly accurate image reconstruction for multimodal noise suppression using semisupervised learning on big data, IEEE Trans. Multimed. 20 (2018) 3045–3056.
[51] K. Yadati, M. Larson, C.C.S. Liem, A. Hanjalic, Detecting socially significant music events using temporally noisy labels, IEEE Trans. Multimed. 20 (2018) 2526–2540.
[52] Y. Li, J. Yang, Y. Song, L. Cao, J. Luo, L.-J. Li, Learning from noisy labels with distillation, in: Proc. IEEE Intl. Conf. Comp. Vis., 2017, pp. 1910–1918.
[53] D.T.D.I.T. Yamasaki, K. Aizawa, Joint optimization framework for learning with noisy labels, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2018, pp. 5552–5560..
[54] D. Arpit, S. Jastrzębski, N. Ballas, D. Krueger, E. Bengio, M.S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio, et al., A closer look at memorization in deep networks, Proc. Intl. Conf. Mach. Learn. 70, 2017, pp. 233–242.
[55] Y. Shu, Z. Cao, M. Long, J. Wang, Transferable curriculum for weakly-supervised domain adaptation, in: Proc. of the AAAI Conference on Artificial Intelligence, volume 33, 2019, pp. 4951–4958.
[56] F. Liu, J. Lu, B. Han, G. Niu, G. Zhang, M.Butterfly Sugiyama, A panacea for all difficulties in wildly unsupervised domain adaptation, Proc. Adv. Neural Inf. Process. Syst. Workshop (2019).
[57] Y. Wen, K. Zhang, Z. Li, Y. Qiao, A discriminative feature learning approach for deep face recognition, in: Proc. Eur. Conf. Comput. Vision, 2016, pp. 499–515.
[58] X. Zhang, Z. Fang, Y. Wen, Z. Li, Y. Qiao, Range loss for deep face recognition with long-tailed training data, in: Proc. IEEE Intl. Conf. Comp. Vis., 2017, pp. 5409–5418..
[59] Y. Zheng, D.K. Pal, M. Savvides, Ring loss: Convex feature normalization for face recognition, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2018, pp. 5089–5097..
[60] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: Proc. Intl. Conf. Mach. Learn., volume 37, 2015, pp. 448–456..
[61] R. Ranjan, C.D. Castillo, R. Chellappa, L2-constrained softmax loss for discriminative face verification, arXiv:1703.09507 (2017)..
[62] M. Ye, C. Liang, Y. Yu, Z. Wang, Q. Leng, C. Xiao, J. Chen, R. Hu, Person reidentification via ranking aggregation of similarity pulling and dissimilarity pushing, IEEE Trans. Multimed. 18 (2016) 2553–2566.
[63] S. Zhou, J. Wang, R. Shi, Q. Hou, Y. Gong, N. Zheng, Large margin learning in set-to-set similarity comparison for person reidentification, IEEE Trans. Multimed. 20 (2018) 593–604.
[64] R. Xu, G. Li, J. Yang, L. Lin, Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation, in: Proc. IEEE Intl. Conf. Comp. Vis., 2019, pp. 1426–1435.
[65] D. Cheng, Y. Gong, S. Zhou, J. Wang, N. Zheng, Person re-identification by multi-channel parts-based cnn with improved triplet loss function, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2016, pp. 1335–1344.
[66] C. Chen, Z. Chen, B. Jiang, X. Jin, Joint domain alignment and discriminative feature learning for unsupervised deep domain adaptation, in: Proc. AAAI Conf. Artif. intell., 2019, pp. 3296–3303.
[67] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, arXiv preprint arXiv:1503.02531 (2015).
[68] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (1998) 2278–2324.
[69] Y. Ganin, V. Lempitsky, Unsupervised domain adaptation by backpropagation, in: Proc. Intl. Conf. Mach. Learn., volume 37, 2015, pp. 1180–1189.
[70] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A.Y. Ng, Reading digits in natural images with unsupervised feature learning, in: Proc. Adv. Neural Inf. Process. Syst. Workshop on Deep Learning and Unsupervised Feature Learning, 2011.
[71] A. Krizhevsky, G. Hinton, Learning multiple layers of features from tiny images, Master's thesis, Department of Computer Science, University of Toronto, 2009.

[72] A. Coates, A. Ng, H. Lee, An analysis of single-layer networks in unsupervised feature learning, in: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings, 2011, pp. 215–223..
[73] K. Saenko, B. Kulis, M. Fritz, T. Darrell, Adapting visual category models to new domains, in: Proc. Eur. Conf. Comput. Vision, 2010, pp. 213–226..
[74] X. Peng, B. Usman, N. Kaushik, J. Hoffman, D. Wang, K. Saenko, Visda: The visual domain adaptation challenge, arXiv preprint arXiv:1710.06924 (2017)..
[75] P. Arbelaez, M. Maire, C. Fowlkes, J. Malik, Contour detection and hierarchical image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2011) 898–916.
[76] D. Ulyanov, A. Vedaldi, V. Lempitsky, Instance normalization: the missing ingredient for fast stylization. corr abs/1607.0 (2016)..
[77] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: Proc. Intl. Conf. Learn. Represent., 2015..
[78] M. Ghifary, W.B. Kleijn, M. Zhang, D. Balduzzi, W. Li, Deep reconstruction-classification networks for unsupervised domain adaptation, in: Proc. Eur. Conf. Comput. Vision, 2016, pp. 597–613.
[79] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, D. Erhan, Domain separation networks, in: Proc. Adv. Neural Inf. Process. Syst., 2016, pp. 343–351..
[80] O. Sener, H.O. Song, A. Saxena, S. Savarese, Learning transferrable representations for unsupervised domain adaptation, in: Proc. Adv. Neural Inf. Process. Syst., 2016, pp. 2110–2118..
[81] G. French, M. Mackiewicz, M. Fisher, Self-ensembling for visual domain adaptation, in: Proc. Intl. Conf. Learn. Represent., 2018..
[82] M. Kim, P. Sahu, B. Gholami, V. Pavlovic, Unsupervised visual domain adaptation: A deep max-margin gaussian process approach, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2019, pp. 4380–4390..
[83] A. Kumar, P. Sattigeri, K. Wadhawan, L. Karlinsky, R. Feris, W.T. Freeman, G. Wornell, Co-regularized alignment for unsupervised domain adaptation, in: Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 9345–9356.
[84] M. Long, H. Zhu, J. Wang, M.I. Jordan, Unsupervised domain adaptation with residual transfer networks, in: Proc. Adv. Neural Inf. Process. Syst., 2016, pp. 136–144.
[85] S. Sankaranarayanan, Y. Balaji, C.D. Castillo, R. Chellappa, Generate to adapt: Aligning domains using generative adversarial networks, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2018, pp. 8503–8512.
[86] L.v.d. Maaten, G. Hinton, Visualizing data using t-sne, J. Mach. Learn. Res. 9 (2008) 2579–2605.
[87] Y. Pan, T. Yao, Y. Li, Y. Wang, C.-W. Ngo, T. Mei, Transferrable prototypical networks for unsupervised domain adaptation, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2019, pp. 2239-2247.
[88] Y. Pan, T. Yao, Y. Li, C.-W. Ngo, T. Mei, Exploring Category-Agnostic Clusters for Open-Set Domain Adaptation, in: Proc. IEEE Conf. Comp. Vis. Pattern Recog., 2020, pp. 13867-13875.



Qi He is pursuing his Ph.D. degree from School of Computing and Artificial Intelligence, Southwest Jiaotong University, Chengdu, China. He was interned at Microsoft Research Asia in 2018. His research interests include computer vision and artificial intelligence.

Qi Dai received his Ph.D. degree in computer science from Fudan University, Shanghai, in 2017. Currently, he is a Senior Researcher with Microsoft Research Asia. His research interests include multimedia retrieval and computer vision.

Xiao Wu received the B.Eng. and M.S. degrees in computer science from Yunnan University, Yunnan, China, in 1999 and 2002, respectively, and the Ph.D. degree in Computer Science from City University of Hong Kong, Hong Kong in 2008. Currently, he is a Professor and the Assistant Dean of School of Computing and Artificial Intelligence, Southwest Jiaotong University, Chengdu, China. He was with the Institute of Software, Chinese Academy of Sciences, Beijing, China, from 2001 to 2002. He was a Research Assistant and a Senior Research Associate at the City University of Hong Kong, Hong Kong, from 2003 to 2004, and 2007 to 2009, respectively. He was with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA, and School of Information and Computer Science, University of California, Irvine, CA, USA as a Visiting Scholar during 2006 to 2007 and 2015 to 2016, respectively. He has authored or co-authored more than 80 research papers in well-respected journals, such as TIP, TMM, TMI and prestigious proceedings like CVPR, ICCV and ACM MM. He received the Second Prize of Natural Science Award of the Ministry of Education, China in 2016 and the Second Prize of Science and Technology Progress Award of Henan Province, China in 2017. His research interests include artificial intelligence, computer vision, and multimedia information retrieval.

Jun-Yan He received his Ph.D. and B.Sc. degrees from Southwest Jiaotong University, Chengdu, China in 2021 and 2013, respectively. Currently, he is a Researcher at Alibaba DAMO Academy and was interned at it in 2019. His research interests include computer vision, artificial intelligence and intelligent transportation systems.