Adaptive Mutual Learning for Unsupervised Domain Adaptation

Lihua Zhou, Siying Xiao, Mao Ye*, *Member, IEEE,* Xiatian Zhu* and Shuaifeng Li

***Abstract*—Unsupervised domain adaptation aims to transfer knowledge from labeled source domain to unlabeled target domain. The semi-supervised method based on mean-teacher framework is one of the main stream approaches. By enforcing consistency constraints, it is hopeful that the teacher network will distill useful source domain knowledge to the student network. However, in practice negative transfer often emerges because the performance of the teacher network is not guaranteed to be always better than the student network. To address this limitation, a novel *Adaptive Mutual Learning* (AML) strategy is proposed in this paper. Specifically, given a target sample, the network with worse prediction will be optimized by pushing its prediction close to the better prediction. This is in the spirit of traditional knowledge distillation. On the other hand, the network with better prediction is further refined by requiring its prediction to stay away from the worse prediction. This can be regarded conceptually as *reverse knowledge distillation*. In this way, two networks learn from each other according to their respective performance. At inference phase, the averaged output of these two networks can be taken as the final prediction. Experimental results demonstrate that our AML achieves competitive results.**

***Index Terms*—Unsupervised Domain Adaptation, Consistency Constraints, Adaptive Mutual Learning.**

I. Introduction

In recent years, deep learning has achieved great success in the field of computer vision [1]–[4], which mainly can be attributed to a large amount of labeled data. However, in practical applications, it is complicated to label a large number of data. Therefore, it has been shown crucial for real world to train a model using existing labeled data (source domain) to performs well on unlabeled data (target domain), which emerges the research of Unsupervised Domain Adaptation (UDA) [5]–[8].

The existing UDA methods can be roughly divided into two categories. One is based on *distribution alignment*, which realizes knowledge transfer by aligning the distribution between the source domain and the target domain [9]–[12]. Another is based on *semi-supervised learning*, which achieves the generalization of the model by optimizing some auxiliary tasks [13], [14]. In this line, the mean-teacher based approach is popular [15]. This kind of approach trains student network by supervised learning on source domain and distilling knowledge from the teacher network through consistency constraint on target domain; while the teacher network is updated by an Exponential Moving Average (EMA) strategy based on the parameters of student network [16]–[19].

Although the mean-teacher based approach has achieved good results, it still has the following two problems in domain adaptation scenario. First, since the traditional mean teacher-based method is essentially performing knowledge distillation in terms of consistency loss, however, due to the existence of domain shift, there is no guarantee that the teacher network always performs better than the student network on all target domain samples, which is proved in our experimental section. As shown in Fig. 1(a), the correct predicted sample (red dot) by student network will be misclassified after knowledge distillation, which makes negative transfer. To solve this problem, the existing methods [17]usually selects high-confidence target domain samples predicted by the teacher network to train the student network, so that the reliability of knowledge distillation can be ensured. However, it causes low sample utilization, and even for the selected high-confidence target samples, student network may still perform better than the teacher network, such negative transfer still occurs. Second, the teacher network is updated by the EMA strategy, so the teacher network is essentially a combination of a series of student network. As a result, the teacher network and the student network are highly coupled, rendering the teacher network gradually helpless for the student network [20].

Based on the above analysis, we proposed an Adaptive Mutual Learning (**AML**) method. To solve the first problem, we design a role selection strategy to dynamically set teacher network and student network. It judges which network can perform better for each sample to set the teacher network and student network, thus avoiding the occurrence of negative transfer. Specifically, target domain data are divided into two sets according to whether the prediction of `Net 1` is more discriminative than that of `Net 2`. For the first set where `Net 1` performs better, `Net 1` is set as teacher; while for another set, `Net 2` is set as teacher. Then the prediction of the student network is asked to be close to the teacher network as traditional knowledge distillation. For the second problem, to overcome the problem of high coupling between teacher network and student network, we propose a reverse knowledge distillation strategy instead of using the EMA strategy, which requires the teacher network to stay away from the prediction of the student network and makes teacher network learn more discriminative features. Through the backpropagation of reverse knowledge distillation, instead of the calculation of

Lihua Zhou, Siying Xiao, Mao Ye and Shuaifeng Li are with School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, PR China. E-mail: lihua.zhou@std.uestc.edu.cn, singxsy@126.com, maoye@uestc.edu.cn, hotwindlsf@gmail.com

Xiatian Zhu is with Surrey Institute for People-Centred Artificial Intelligence, CVSSP, University of Surrey, Guildford, UK. E-mail: xiatian.zhu@surrey.ac.uk

* corresponding author.

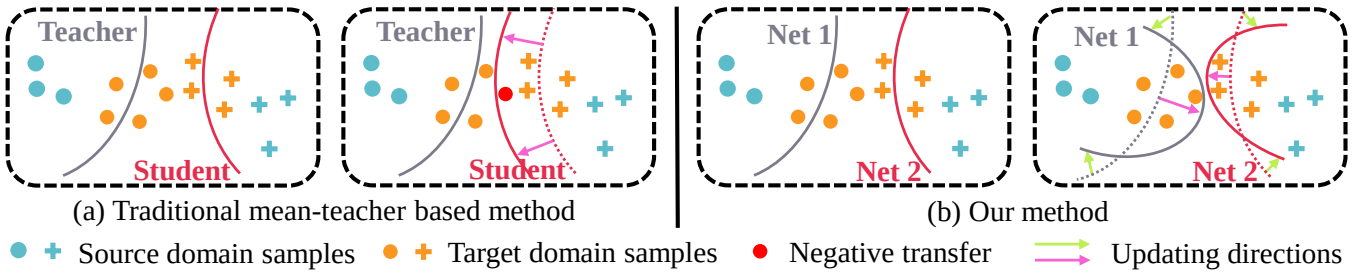


Fig. 1. Illustration of (a) the traditional mean-teacher based UDA methods [17] *vs.* (b) our proposed adaptive mutual learning (AML) method. (a) Traditional method forces student network close to teacher network, which may cause negative transfer when teacher network predicts wrong. (b) Our AML method enables two networks to learn from each other's strengths.

the EMA strategy, the two networks are no longer coupled. As shown in Fig. 1(b), by using the traditional knowledge distillation strategy to train the student network and the reverse knowledge distillation strategy to train the teacher network, negative transfer is inhibited and hard target samples can be correctly classified.

Our contributions can be summarized as follows: 1) An adaptive role selection strategy is proposed to decide which one is the teacher for different target samples, which makes the mutual learning possible. 2) We designed a reverse knowledge distillation strategy. It requires the predictions of teacher network to be away from the predictions of student network. In this way, two networks are no longer highly coupled. 3) We proposed an adaptive mutual learning framework for UDA problem. Our method has achieved competitive results on three public datasets. The ablation experiment also proves the reliability of the above two strategies. Compared with the traditional mean-teacher based approach, an adaptive two-way knowledge distillation is performed and all target samples are utilized in domain adaptation.

II. Related Work

Unsupervised Domain adaptation(UDA). Domain adaptation seeks to transfer information from labeled source domain to unlabeled o target domain [2], [21]–[24]. The existing UDA methods can be roughly divided into two categories, one is based on *distribution alignment*, and another is based on *semi-supervised learning*.

The methods based on *distribution alignment* explicitly align the distribution between the source domain and the target domain to solve the domain shift problem. These methods can be further divided into the following three groups. The first group is *statistic moment matching*, which optimizes the model by defining a distribution discrepancy between the source domain and the target domain as a loss function, such as MMD [11], CORAL [25], CDD [26], etc. The second group is *adversarial learning*, which plays a minimax game between feature extractor and domain discriminator so that feature extractor can learn domain invariant features. The represented methods include DANN [9], CDAN [27], ADDA [28], CDGC [29], ToAlign [30] and so on. CDGC [29] exploits and aligns both sample- and class-level structure information by designing a graph-based feature propagation module. ToAlign [30] decomposes source features into task-related features and task-irrelevant features to make the domain alignment task proactively serve the classification task. The third group is *adversarial generation*, which generates the fake data and aligns the distribution at pix-level. The methods with high attention are CoGAN [10], SimGAN [31], CycleGAN [32] and PAT [33]. PAT [33] generates adversarial samples from pairs of samples across the source and target domains and further exploits these samples to augment training data.

The *semi-supervised learning* approach improves the generalization ability of the model by some auxiliary tasks. For example, CoVi [34] alleviates the inter-domain discrepancy and intra-domain categorical confusion by consistency training. ATDOC [35] generates unbiased accurate pseudo labels for unlabeled target data by memory bank or neighborhood aggregation. CAT [36] achieves the objectives of discriminative learning and class-conditional alignment via a discriminative clustering loss and a cluster-based alignment loss. FixBi [37] constructs intermediate domains by mixup and further proposes bidirectional matching, self-penalization, and consistency regularization for efficient use of intermediate space. SRDC [38] proposes a source-regularized, deep discriminative clustering method to uncover the intrinsic discrimination among target data. SHOT++ [39] further explores the source-data absent problem and proposes a new labeling transfer strategy to splits target samples based on the confidence of its predictions.

Recently, some methods based on mean-teacher were introduced into domain adaptation. SEDA [17] introduces mean-teacher framework to domain adaptation, which further applies class balance strategy and confidence threshold to improve consistency constraint. MTOR [16] proposes consistency constraint on region-level, inter-graph and intra-graph in the target domain. By employing teacher-student framework, MLC-Net [18] exploits point-level, instance-level and neural statistics-level consistency to achieve unsupervised domain adaptation in 3D detection. A segmentation model has been proposed which refers itself as a memory module, and minimizes the discrepancy of primary classifier and auxiliary classifier to enhance the prediction consistency [19]. Whilst achieving good performance, these methods are fundamentally limited to an assumption that a fixed teacher network always performs better than the fixed student network. On the

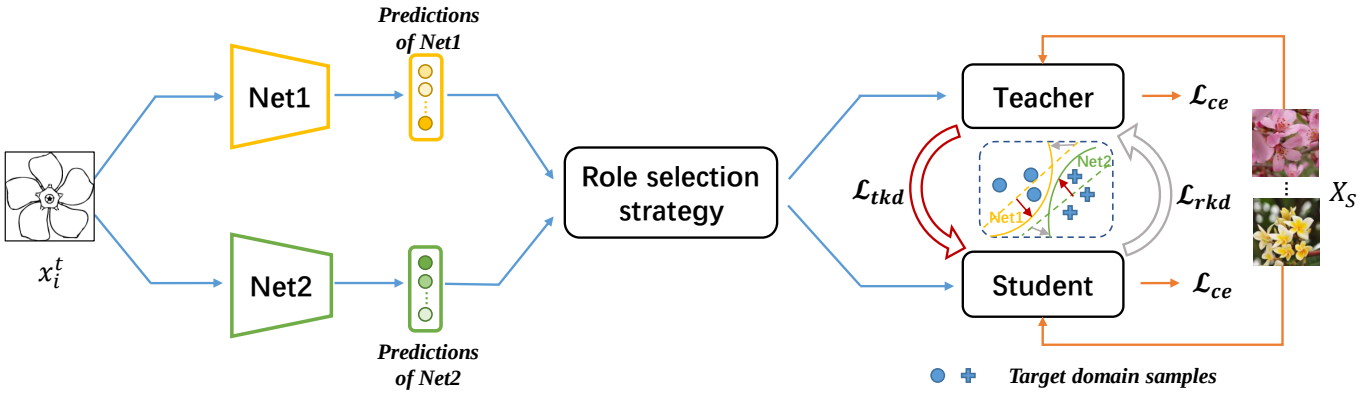


Fig. 2. Overview of the proposed AML (Best viewed in color). The role selection module divides the target domain data into two parts based on the confidence comparison between the predictions of `Net 1` and `Net 2`, and decides which `Net` is the Teacher or Student for each part. Then mutual learning is performed on these two networks. That is, traditional ($\mathcal{L}_{tkd}$) and reverse ($\mathcal{L}_{rkd}$) knowledge distillation strategies are conducted to update the Student and Teacher networks. Cross entropy loss is applied to both networks on source domain data.

contrary, we are convinced that in both networks, the well-behaved network should be the teacher and it also needs to be optimized. Therefore, role selection and reverse knowledge distillation strategies are naturally emerged.

Mean-teacher in UDA. We briefly revisit the processes of mean-teacher in UDA [17], which has two steps to update the student network and teacher network respectively.

In the first step, the unlabeled target domain samples are sent to the teacher network and the student network, and the labeled source domain samples are only sent to the student network. Then for the student network, we need to calculate the supervised loss of the source domain and the consistency loss of the target domain between two networks, as shown below:

$$\mathcal{L}_{cls}(D_s) = \mathbb{E}_{\boldsymbol{x}_i^s \in D_s} \mathcal{L}^{ce}(\boldsymbol{p}_{i,stu}^s, \boldsymbol{y}_i^s),$$
$$\mathcal{L}_{con}(D_t) = \mathbb{E}_{\boldsymbol{x}_i^t \in D_t} \|\boldsymbol{p}_{i,tea}^t - \boldsymbol{p}_{i,stu}^t\|^2,$$

where $\mathcal{L}^{ce}(\cdot,\cdot)$ means cross entropy, $\boldsymbol{p}_{i,stu}^s$ means the prediction of the i-th source samples of student network, $\boldsymbol{p}_{i,stu}^t$ and $\boldsymbol{p}_{i,tea}^t$ represent the prediction of the i-th target samples of student network and teacher network respectively. Therefore, the student network is updated as follows:

$$\min_{\theta_{stu}} \mathcal{L}_{cls}(D_s) + \alpha \mathcal{L}_{con}(D_t) + \beta \mathcal{L}_{bal}(D_t),$$

where θ_{stu} means the parameters of student network, α and β are trade-off hyperparameters. $\mathcal{L}_{bal}(D_t)$ is class balance loss widely used by the mean-teacher method [17]. In this work, the mutual information objective is applied [40], that is, $\mathcal{L}_{bal}(D_t) = \mathbb{E}_{x_i^t \in D_t} En(\boldsymbol{p}_{i,stu}^t) - En(\mathbb{E}_{x_i^t \in D_t} \boldsymbol{p}_{i,stu}^t)$, where $En(\cdot)$ means entropy.

In the second step, the EMA strategy is applied to update the parameters of teacher network, which is shown as follows:

$$\theta_{tea} = (1-\gamma)\theta_{tea} + \gamma\theta_{stu},$$

where θ_{tea} means the parameters of teacher network, γ is the trade-off hyperparameter.

III. Method

At UDA problem setting, there is a source domain $D_s = \{(\boldsymbol{x}_i^s, \boldsymbol{y}_i^s)\}_{i=1}^{n_s}$ consists of n_s labeled samples, and a target domain $D_t = \{(\boldsymbol{x}_i^t)\}_{i=1}^{n_t}$ consists of n_t unlabeled samples. These two domains have same label space $\{1, 2, \ldots, K\}$, but their data distributions are different. The goal of UDA is to train a reliable model for the unlabeled target domain by using both of the labeled source domain data and unlabeled target domain data.

Overview. As shown in Fig. 2, the target domain data are sent into two networks, namely `Net 1` and `Net 2`. To ensure the reliability of knowledge distillation, for each target domain sample, we first perform role selection based on the predictions of these two networks to determine which network acts as the teacher network and another acts as the student network. Then traditional knowledge distillation ($\mathcal{L}_{tkd}$) and reverse knowledge distillation ($\mathcal{L}_{rkd}$) are performed between these two networks. Specifically, the student network is updated by approaching the predictions of the teacher network, as traditional knowledge distillation does. While for the teacher network, its parameters are adjusted by requiring its prediction away from the prediction of the student network. The teacher network can produce more discriminative predictions in this process, which we term reverse knowledge distillation. In addition, compared with the EMA which is uesd by previous mean-teacher approaches, the reverse knowledge distillation will make two networks no longer coupled during teacher network training. For labeled source domain data, the cross entropy loss is used to make these two networks fit the distribution of source domain ($\mathcal{L}_{ce}$).

A. Role selection strategy

As we mentioned before, due to domain shift exists in the UDA scenario, the traditional knowledge distillation strategy may result in negative transfer, as the teacher network cannot always outperform the student network on all target domain samples. To solve this problem, we propose a role selection strategy to dynamically set teacher network and student network for each target domain samples. Specifically, since

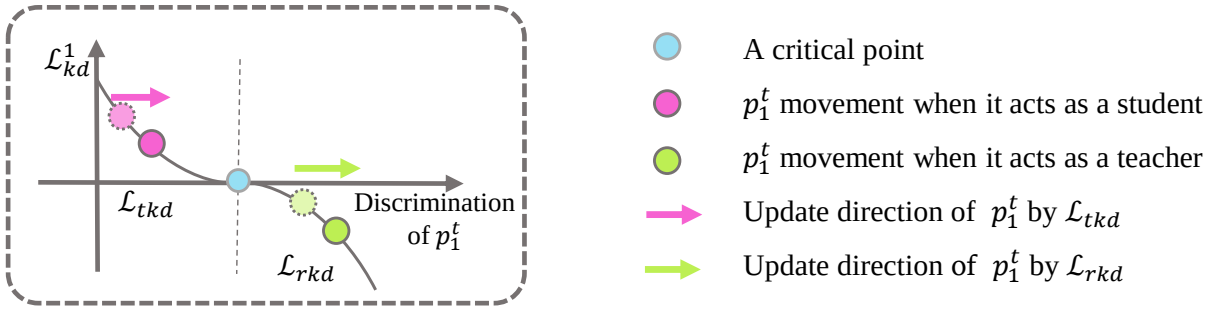


Fig. 3. The relationship between $\mathcal{L}_{kd}^1$ and the discrimination ability of Net 1. For any sample x^t, the blue dot represents the discrimination of Net 1 is equal to that of Net 2. When the discrimination of the prediction $\boldsymbol{p}_1^t$ of Net 1 is not as good as $\boldsymbol{p}_2^t$, the discrimination of Net 1 can be improved by requiring the $\boldsymbol{p}_1^t$ to be close to $\boldsymbol{p}_2^t$ through the traditional knowledge distillation strategy. When $\boldsymbol{p}_1^t$ are more discriminative than $\boldsymbol{p}_2^t$, reverse knowledge distillation can be performed which enables Net 1 to learn more discriminative predictions.

the label information of the target domain is not available, we approximately estimate which network performs better by introducing an evaluation criterion, such as entropy [41], KL divergence [42] and so on, and determine which one is the teacher network or the student network according to the results of the evaluation. Recently, the work in [27] found that the prediction with smaller entropy is more reliable. Therefore, in this work, the entropy is used as the criterion to determine which network is regraded as the teacher between Net 1 and Net 2. So for a target sample, by comparing the entropy of prediction from Net 1 and Net 2, we choose the smaller one as the teacher. Suppose the target sample is $\boldsymbol{x}^t$, the entropy is calculated as $En(\boldsymbol{p}) = -\frac{1}{K}\sum_{k=1}^{K}\boldsymbol{p}_k \log \boldsymbol{p}_k$, where $\boldsymbol{p}_k$ is k-th component of the prediction $\boldsymbol{p}$.

Specifically, a batch of data B^t are sampled from target domain in each iteration. According to our role selection criterion, the target data is divided into two sets B_1^t and B_2^t as follows,

$$\begin{aligned} B_1^t &= \{x_i^t | En(\boldsymbol{p}_{i,1}^t) < En(\boldsymbol{p}_{i,2}^t)\}, \\ B_2^t &= \{x_i^t | En(\boldsymbol{p}_{i,1}^t) > En(\boldsymbol{p}_{i,2}^t)\} \end{aligned} \tag{1}$$

where $\boldsymbol{p}_{i,1}^t = f(\boldsymbol{x}^t; \theta_{Net1})$ and $\boldsymbol{p}_{i,2}^t = f(\boldsymbol{x}^t; \theta_{Net2})$ are the preditions of Net 1 and Net 2 for the i-th target sample x_i^t respectively, and $f(\cdot)$ means the forward process of the network. According to our hypothesis, for the set B_1^t, Net 1 is selected as teacher network while Net 2 is set as student network; for the set B_2^t, the situation is exactly the opposite.

Remark. To demonstrate effectiveness of role selection strategy, we further introduce KL divergence as the evaluation criterion in the experimental section.

B. Mutual knowledge distillation strategy

For Net 1 and Net 2, after the teacher and student roles are decided, mutual knowledge distillation strategy is employed. First, to facilitate the student network learning useful knowledge from the teacher network, the tradition knowledge distillation technical route is used. On the other hand, the output of teacher is required to be far away from student, so that the teacher can be optimized in a more discriminative direction. Furthermore, compared with the EMA, the reverse knowledge distillation does not make the two networks highly coupled.

Specifically, for the B_1^t data set, Net 1 is set as the teacher and Net 2 is set as the student. Therefore, we need to distill the knowledge from Net 1 to Net 2, that is traditional knowledge distillation. To improve the performance of Net 2, the loss is defined as

$$\min_{\theta_{Net2}} \mathcal{L}_{tkd}(B_1^t) = \mathbb{E}_{\boldsymbol{x}_i^t \in B_1^t} \|\boldsymbol{p}_{i,1}^t - \boldsymbol{p}_{i,2}^t\|^2. \tag{2}$$

At the same time, due to the discrimination of Net 2 is not as good as that of Net 1, according to the entropy, we propose a reverse knowledge distillation. By asking the prediction of Net 1 to be far away from Net 2, the discrimination of Net 1 can be further improved, which is defined as follows:

$$\min_{\theta_{Net1}} \mathcal{L}_{rkd}(B_1^t) = -\mathbb{E}_{\boldsymbol{x}_i^t \in B_1^t} \|\boldsymbol{p}_{i,1}^t - \boldsymbol{p}_{i,2}^t\|^2. \tag{3}$$

The above loss requires that $p_{i,1}^t$ leaves away from $p_{i,2}^t$ for the B_1^t. In this way, Net 1 is trained to get more discriminative output. At the same time, compared with EMA, which results in the teacher network being a combination of the parameters of the student network, the reverse knowledge distillation strategy optimized by gradient descent makes the two networks no longer highly coupled and maintains their respective properties.

Similarly, for the B_2^t data set, Net 1 is the student, whose loss is defined as the following,

$$\min_{\theta_{Net1}} \mathcal{L}_{tkd}(B_2^t) = \mathbb{E}_{\boldsymbol{x}_i^t \in B_2^t} \|\boldsymbol{p}_{i,1}^t - \boldsymbol{p}_{i,2}^t\|^2. \tag{4}$$

And vise versa for Net 2, the reverse knowledge distillation is

$$\min_{\theta_{Net2}} \mathcal{L}_{rkd}(B_2^t) = -\mathbb{E}_{\boldsymbol{x}_i^t \in B_2^t} \|\boldsymbol{p}_{i,1}^t - \boldsymbol{p}_{i,2}^t\|^2. \tag{5}$$

All in all, we can combine the above loss functions for Net 1 and Net 2 respectively as follows,

$$\min_{\theta_{Net1}} \mathcal{L}_{kd}^1(B^t) = \mathcal{L}_{tkd}(B_2^t) + \mathcal{L}_{rkd}(B_1^t), \tag{6}$$

$$\min_{\theta_{Net2}} \mathcal{L}_{kd}^2(B^t) = \mathcal{L}_{tkd}(B_1^t) + \mathcal{L}_{rkd}(B_2^t). \tag{7}$$

To further illustrate our mutual learning strategy, we show this process in Fig. 3, which describes the relationship between the value of $\mathcal{L}_{kd}^1$ and the discrimination ability of Net 1. For any target domain sample x^t, its predictions by Net 1 and Net 2 are $\boldsymbol{p}_1^t$ and $\boldsymbol{p}_2^t$ respectively. As mentioned

in Section III-A, the prediction entropy is used to measure the discrimination ability. As shown in Fig. 3, the blue dot represents a critical point where the two networks possess the same discriminative ability on x^t, that is, $En(\boldsymbol{p}_1^t) = En(\boldsymbol{p}_2^t)$, so this sample belongs neither to B_1^t nor B_2^t. When $En(\boldsymbol{p}_1^t) > En(\boldsymbol{p}_2^t)$, it means that this sample $x^t \in B_2^t$ and the discrimination of $\boldsymbol{p}_1^t$ is not as good as $\boldsymbol{p}_2^t$ on this sample, which is shown as the dotted pink dot in Fig.3. Traditional knowledge distillation is performed on `Net 1` such that the discrimination ability increases. When $En(\boldsymbol{p}_1^t) < En(\boldsymbol{p}_2^t)$, it indicates that this sample $x^t \in B_1^t$ and the discrimination of $\boldsymbol{p}_1^t$ outperforms $\boldsymbol{p}_2^t$, which is shown as the dotted green dot in Fig.3. Reverse knowledge distillation is performed on `Net 1` such that the prediction leaves away from `Net 2`, which also increases the discrimination ability of `Net 1`. Therefore, regardless of whether the sample x^t belongs to B_1^t or B_2^t, when the loss $\mathcal{L}_{kd}^1$ is minimized to update `Net 1`, it can make the $\boldsymbol{p}_1^t$ more discriminative.

Remark. In our mutual learning process, two networks learn from each other's strengths to improve their performances. Instead of traditional knowledge distillation route, the role of teacher or student is dynamically changed according to different target samples and all target samples are useful in domain adaptation.

C. Overall loss

Training based on source domain data. Similar to the traditional UDA methods, in each iteration, a batch of source domain data B^s are sampled. We directly use the cross entropy loss to train `Net 1` and `Net 2`, so that these two networks can fit the source domain data, which is defined as follows,

$$\min_{\theta_{Net\ j}} \mathcal{L}_{cls}^j(B^s) = \mathbb{E}_{\boldsymbol{x}_i^s \in B^s} \mathcal{L}_{ce}(\boldsymbol{p}_{i,j}^s, \boldsymbol{y}_i^s), \tag{8}$$

where $\mathcal{L}_{ce}(\cdot,\cdot)$ denotes the cross entropy function, $\boldsymbol{p}_{i,j}^s$ represents the prediction of the i-th source domain data in the j-th network for $j \in \{1,2\}$.

The mutual information loss in [40] is also applied, which is commonly used by other mean-teacher based methods to balance the class distribution in the target domain [17]. That is,

$$\mathcal{L}_{bal}^j(B^t) = \mathbb{E}_{x_i^t \in B^t} En(\boldsymbol{p}_{i,j}^t) - En(\mathbb{E}_{x_i^t \in B^t} \boldsymbol{p}_{i,j}^t), \tag{9}$$

where $\boldsymbol{p}_{i,j}^t$ represents the prediction of the i-th target domain data in the j-th network for $j \in \{1,2\}$.

Combining the loss functions above, **the overall losses** of `Net 1` and `Net 2` are the following,

$$\min_{\theta_{net1}} \mathcal{L}_{cls}^1(B^s) + \alpha \mathcal{L}_{kd}^1(B^t) + \beta \mathcal{L}_{bal}^1(B^t), \tag{10}$$

and

$$\min_{\theta_{net2}} \mathcal{L}_{cls}^2(B^s) + \alpha \mathcal{L}_{kd}^2(B^t) + \beta \mathcal{L}_{bal}^2(B^t), \tag{11}$$

where α and β are two balance parameters.

Our method is summarized in Algorithm 1. For training these two networks, the losses in Eq.(10) and Eq.(11) are repeatedly minimized until convergence. For inference, the average output of `Net 1` and `Net 2` is used as the final prediction.

D. Analysis

In this part, we analyze what happens to the discriminative ability of the two networks after each optimization in the traditional mean teacher framework and our method AML. In order to explain the principle of the proposed method more easily, we consider a two-class classification task, i.e., the label space $\mathcal{Y} = \{1,2\}$.

Lemma 1. Traditional mean teacher methods can improve the discriminative ability of one network but inhibit the discriminative ability of another network after optimization.

In the traditional mean-teacher framework, the student network is updated by calculating the gradient of loss, while the teacher network is derived from the EMA strategy by the student network. Given a target sample $\boldsymbol{x}$, the predictions of the teacher network and the student network can be denoted as $\boldsymbol{P}_{tea} = [\boldsymbol{p}_{tea,1}, \boldsymbol{p}_{tea,2}]$ and $\boldsymbol{P}_{stu} = [\boldsymbol{p}_{stu,1}, \boldsymbol{p}_{stu,2}]$ respectively. Let us assume that the pseudo label $\hat{\boldsymbol{y}}$ of $\boldsymbol{x}$ is [1,0] i.e., of class 1, which has $\boldsymbol{p}_{tea,1} > \boldsymbol{p}_{tea,2}$ and $\boldsymbol{p}_{stu,1} > \boldsymbol{p}_{stu,2}$. Without loss of generality, in **Case 1**, we suppose the teacher network performs better than the student network, which has more discriminative prediction, i.e., $En(\boldsymbol{P}_{tea}) < En(\boldsymbol{P}_{stu})$, which will satisfy the $\boldsymbol{p}_{tea,1} > \boldsymbol{p}_{stu,1}$ and $\boldsymbol{p}_{tea,2} < \boldsymbol{p}_{stu,2}$. Then, the consistency loss, which achieves knowledge distillation from teacher network to student network, is used for target sample $\boldsymbol{x}$ to train the student network, which is shown as follows:

$$\min_{\theta_{stu}} \mathcal{L}_{con} = ||\boldsymbol{P}_{stu} - \boldsymbol{P}_{tea}||^2, \tag{12}$$

Therefore, the gradient of $\boldsymbol{P}_{stu}$ is shown as follows:

$$\frac{\partial \mathcal{L}_{con}}{\partial \boldsymbol{P}_{stu}} = 2(\boldsymbol{P}_{stu} - \boldsymbol{P}_{tea}), \tag{13}$$

Then the updating formula of P_{stu} is expressed as follows according to backpropagation:

$$\hat{\boldsymbol{P}}_{stu} = \boldsymbol{P}_{stu} - \eta \frac{\partial \mathcal{L}_{con}}{\partial \boldsymbol{P}_{stu}} = \boldsymbol{P}_{stu} - 2\eta(\boldsymbol{P}_{stu} - \boldsymbol{P}_{tea}), \tag{14}$$

where $\eta > 0$ represents the learning rate. For Eq.(14), it can be further written as follows:

$$\begin{aligned} \hat{\boldsymbol{p}}_{stu,1} &= \boldsymbol{p}_{stu,1} + 2\eta(\boldsymbol{p}_{tea,1} - \boldsymbol{p}_{stu,1}), \\ \hat{\boldsymbol{p}}_{stu,2} &= \boldsymbol{p}_{stu,2} + 2\eta(\boldsymbol{p}_{tea,2} - \boldsymbol{p}_{stu,2}). \end{aligned} \tag{15}$$

Based on our assumptions, which satisfies $\boldsymbol{p}_{tea,1} > \boldsymbol{p}_{stu,1}$ and $\boldsymbol{p}_{tea,2} < \boldsymbol{p}_{stu,2}$, we can get $\hat{\boldsymbol{p}}_{stu,1} > \boldsymbol{p}_{stu,1}$ and $\hat{\boldsymbol{p}}_{stu,2} < \boldsymbol{p}_{stu,2}$ and then $En(\hat{\boldsymbol{P}}_{stu}) < En(\boldsymbol{P}_{stu})$, which means the student network is more reliable and dicriminative after optimization. On the other hand, the teacher network is updated by the EMA strategy, written as follows:

$$\hat{\theta}_{tea} = (1-\gamma)\theta_{tea} + \gamma\hat{\theta}_{stu}, \tag{16}$$

where $\hat{\theta}_{tea/stu}$ means the parameters of teacher or student network after optimization, γ is the trade-off hyperparameter using for EMA strategy. Obviously, we have $\boldsymbol{P}_{tea} = f(\boldsymbol{x}; \theta_{tea})$ and $\hat{\boldsymbol{P}}_{stu} = f(\boldsymbol{x}; \hat{\theta}_{stu})$. So the updated prediction of teacher network can be written as:

$$\begin{aligned} \hat{\boldsymbol{P}}_{tea} = f(x; \hat{\theta}_{tea}) &\approx (1-\gamma)\boldsymbol{P}_{tea} + \gamma\hat{\boldsymbol{P}}_{stu} \\ &= \boldsymbol{P}_{tea} + \gamma(1-2\eta)(\boldsymbol{P}_{stu} - \boldsymbol{P}_{tea}). \end{aligned} \tag{17}$$

If $f(\cdot)$ is a linear function, we can accurately compute the updated teacher network prediction $\hat{\boldsymbol{P}}_{tea}$ in Eq.(17). However, in general, $f(\cdot)$ is a nonlinear function, so our analysis here can only approximate the predictions of the updated teacher network. Since the learning rate η is usually a very small value, such as 0.001, we can be sure that (1-2η) is greater than 0. To further analyze the discriminativeness of the predictions of the updated teacher network, we further rewrite $\hat{\boldsymbol{P}}_{tea}$ as follows

$$\begin{aligned}\hat{\boldsymbol{p}}_{tea,1} &= \boldsymbol{p}_{tea,1} + \gamma(1-2\eta)(\boldsymbol{p}_{stu,1} - \boldsymbol{p}_{tea,1}) < \boldsymbol{p}_{tea,1},\\ \hat{\boldsymbol{p}}_{tea,2} &= \boldsymbol{p}_{tea,2} + \gamma(1-2\eta)(\boldsymbol{p}_{stu,2} - \boldsymbol{p}_{tea,2}) > \boldsymbol{p}_{tea,2}.\end{aligned} \tag{18}$$

Obviously, the updated teacher prediction $\hat{\boldsymbol{P}}_{tea}$ has a larger entropy value, which means that the teacher network will degenerate in this process.

All in all, in Case 1, the student network becomes more discriminative, while the teacher network suffers performance degradation.

In **Case 2**, suppose that the student network performs better than the teacher network, i.e., $En(\boldsymbol{P}_{tea}) > En(\boldsymbol{P}_{stu})$, which has the $\boldsymbol{p}_{tea,1} < \boldsymbol{p}_{stu,1}$ and $\boldsymbol{p}_{tea,2} > \boldsymbol{p}_{stu,2}$. Similarly to Case1, we can obtain the following equation:

$$\begin{aligned}\hat{\boldsymbol{p}}_{stu,1} &= \boldsymbol{p}_{stu,1} + 2\eta(\boldsymbol{p}_{tea,1} - \boldsymbol{p}_{stu,1}) < \boldsymbol{p}_{stu,1},\\ \hat{\boldsymbol{p}}_{stu,2} &= \boldsymbol{p}_{stu,2} + 2\eta(\boldsymbol{p}_{tea,2} - \boldsymbol{p}_{stu,2}) > \boldsymbol{p}_{stu,2}.\end{aligned} \tag{19}$$

So the prediction of student network on this target sample will get worse. And for the teacher network, we can obtain:

$$\begin{aligned}\hat{\boldsymbol{p}}_{tea,1} &= \boldsymbol{p}_{tea,1} + \gamma(1-2\eta)(\boldsymbol{p}_{stu,1} - \boldsymbol{p}_{tea,1}) > \boldsymbol{p}_{tea,1},\\ \hat{\boldsymbol{p}}_{tea,2} &= \boldsymbol{p}_{tea,2} + \gamma(1-2\eta)(\boldsymbol{p}_{stu,2} - \boldsymbol{p}_{tea,2}) < \boldsymbol{p}_{tea,2}.\end{aligned} \tag{20}$$

In this case, the performance of the student network will become worse, and the teacher model will become more accurate to predict the target samples.

In Case 1 and Case 2, we can find that although one of the teacher network and the student network will improve discriminative ability, the other network will have performance degradation after optimization, which is not conducive to our subsequent operations.

Lemma 2. AML can improve the discriminative ability of two networks after optimization.

For AML, there are two networks with the same structure, for simplicity, here we discuss the case where `Net 1` performs better than `Net 2`, that is, `Net 1` is the teacher and `Net 2` is the student. The opposite case can be derived in the same way. In this situation, we have $En(\boldsymbol{P}_{Net1}) < En(\boldsymbol{P}_{Net2})$, which makes $\boldsymbol{p}_{Net1,1} > \boldsymbol{p}_{Net2,1}$ and $\boldsymbol{p}_{Net1,2} < \boldsymbol{p}_{Net2,2}$. For update network 1, we perform reverse knowledge distillation:

$$\min_{\theta_{Net1}} \mathcal{L}_{rkd} = -\|\boldsymbol{P}_{Net1} - \boldsymbol{P}_{Net2}\|^2. \tag{21}$$

After calculating the gradient and updating the model, the result is as follows:

$$\hat{\boldsymbol{P}}_{Net1} = \boldsymbol{P}_{Net1} - \eta\frac{\partial \mathcal{L}_{rkd}}{\partial \boldsymbol{P}_{Net1}} = \boldsymbol{P}_{Net1} + 2\eta(\boldsymbol{P}_{Net1} - \boldsymbol{P}_{Net2}), \tag{22}$$

And it can be further deduced as the following:

$$\begin{aligned}\hat{\boldsymbol{p}}_{Net1,1} &= \boldsymbol{p}_{Net1,1} + 2\eta(\boldsymbol{p}_{Net1,1} - \boldsymbol{p}_{Net2,1}) > \boldsymbol{p}_{Net1,1},\\ \hat{\boldsymbol{p}}_{Net1,2} &= \boldsymbol{p}_{Net1,2} + 2\eta(\boldsymbol{p}_{Net1,2} - \boldsymbol{p}_{Net2,2}) < \boldsymbol{p}_{Net1,2}.\end{aligned} \tag{23}$$

Algorithm 1 Our method AML

Input: Source domain $D_s = \{(\boldsymbol{x}_i^s, \boldsymbol{y}_i^s)\}_{i=1}^{n_s}$, target domain $D_t = \{(\boldsymbol{x}_i^t)\}_{i=1}^{n_t}$, the epoch number T, the mini-batch number M.

Output: An adapted model with two networks.

Procedure:

```
for t = 1:T do
    for m = 1:M do
        Sample a batch of data B^s and B^t from the source domain D_s and the target domain D_t respectively;
        Forward a mini-batch source domain data B^s and target domain data B^t through Net 1 and Net 2;
        Compute cross entropy on B^s according to Eq. (8);
        Divide B^t into two parts B_1^t and B_2^t according to Eq. (1);
        Compute mutual knowledge distillation on B^t according to Eq. (6) and Eq. (7);
        Optimize the two networks according to Eq. (10) and Eq. (11);
    end for
end for
return Adapted model.
```

It can be seen that we have $En(\hat{\boldsymbol{P}}_{Net1}) < En(\boldsymbol{P}_{Net1})$ after optimization and the `Net 1` will more discriminative.

And traditional knowledge distillation is used to train the `Net 2`. From the previous derivation in Case 1, which has a better teacher network to train the student network, it is easy to obtain the following formula:

$$\begin{aligned}\hat{\boldsymbol{p}}_{Net2,1} &= \boldsymbol{p}_{Net2,1} + 2\eta(\boldsymbol{p}_{Net1,1} - \boldsymbol{p}_{Net2,1}) > \boldsymbol{p}_{Net2,1},\\ \hat{\boldsymbol{p}}_{Net2,2} &= \boldsymbol{p}_{Net2,2} + 2\eta(\boldsymbol{p}_{Net1,2} - \boldsymbol{p}_{Net2,2}) < \boldsymbol{p}_{Net2,2}.\end{aligned} \tag{24}$$

As can be seen from Eq.(24), $En(\hat{\boldsymbol{P}}_{Net2}) < En(\boldsymbol{P}_{Net2})$, `Net 2` is also more discriminative after optimization. And we also find both `Net 1` and `Net 2` improve their performance after optimization.

In conclusion, in the traditional mean-teacher framework, when one of the teacher and student networks gets better, the other network is always optimized in a bad direction. However, in our proposed AML, the two networks continuously improve the discrimination ability of each other through traditional and reverse knowledge distillation, so as to gradually carry out more accurate classification.

IV. Experiments

A. Settings

Datasets. There are three standard UDA datasets used in our experiments. ***Office-31*** [58] is a popular dataset which consists of 4110 images in 31 classes from 3 domains: Amazon(A), Webcam(W), Dslr(D). ***Office-Home*** [59] is a more challenging dataset which consists of 15588 images in 65 classes from 4 domains: Artistic images(A), Clipart images(C), Product images and Realworld images(R). ***ImageCLEF*** [60] is a balanced dataset with 3 domains: Caltech-256(C), ImageNet ILSVRC2012(I) and PASCALVOC2012(P). Each domain has

TABLE I
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON *Office-31* DATASET. METRIC: CLASSIFICATION ACCURACY (%); BACKBONE: RESNET-50.

Method	A→D	A→W	D→A	D→W	W→A	W→D	*avg*
ResNet-50 [1]	68.9	68.4	62.5	96.7	60.7	99.3	76.1
DANN [9]	79.7	82.0	68.2	96.9	67.4	99.1	82.2
CDAN [27]	92.9	94.1	71.0	98.6	69.3	**100.0**	87.7
TSA [43]	92.6	**94.8**	74.9	99.1	74.4	**100.0**	89.3
ATDOC [35]	**95.2**	91.6	74.6	99.1	74.7	**100.0**	89.2
GVB+MetaAlign [44]	94.5	93.0	75.0	98.6	73.6	**100.0**	89.2
DWL [45]	91.2	89.2	73.1	**99.2**	69.8	**100.0**	87.1
SCDA [46]	95.2	94.2	**75.7**	98.7	**76.2**	99.8	85.3
SEDA [17]	87.1	90.8	70.7	97.6	72.2	99.8	86.4
SUDA [47]	91.2	90.8	72.2	98.7	71.4	**100.0**	87.4
CaCo [47]	91.7	89.7	73.1	98.4	72.8	**100.0**	87.6
DMAL [48]	89.1	88.4	71.8	99.2	70.8	**100.0**	86.7
AEO [49]	95.1	94.8	73.0	98.9	71.6	**100.0**	88.9
Our method (AML)	92.4±0.2	94.2±0.3	75.5±0.3	98.9±0.0	75.9±0.0	**100.0±0.0**	**89.5**

TABLE II
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON *Office-Home* DATASET. METRIC: CLASSIFICATION ACCURACY (%); BACKBONE: RESNET-50.

Method	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	*avg*
ResNet-50 [1]	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DANN [9]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
CDAN [27]	50.7	70.6	76.0	57.6	70.0	70.0	57.4	50.9	77.3	70.9	56.7	81.6	65.8
TSA [43]	53.6	75.1	78.3	64.4	73.7	72.5	62.3	49.4	77.5	72.2	58.8	82.1	68.3
ATDOC [35]	54.4	**77.6**	80.8	66.5	75.6	75.8	65.9	51.9	81.1	72.7	57.0	83.5	70.2
CKB+MMD [50]	54.2	74.1	77.5	64.6	72.2	71.0	64.5	53.4	78.7	72.6	58.4	82.8	68.7
GVB+MetaAlign [44]	**59.3**	76.0	80.2	65.7	74.7	75.1	65.7	56.5	81.6	74.1	61.1	**85.2**	71.3
SCDA [46]	57.5	76.9	80.3	65.7	74.9	74.5	65.5	53.6	79.8	74.5	59.6	83.7	70.5
TCM [51]	58.6	74.4	79.6	64.5	74.0	75.1	64.6	56.2	80.9	74.6	60.7	84.7	70.7
SEDA [17]	57.6	76.3	80.4	66.8	75.0	77.0	65.3	54.4	81.1	74.1	60.3	83.4	71.0
DMAL [49]	48.1	70.6	76.6	60.8	67.7	68.8	62.5	51.2	78.1	73.3	54.0	81.0	66.1
AEO [48]	52.2	73.6	76.9	59.7	72.1	73.2	61.3	52.1	78.9	72.4	58.1	82.6	67.8
H-SRDC [52]	50.0	75.3	79.9	63.7	71.9	74.4	62.6	49.6	80.1	71.3	53.6	83.1	68.0
Our method (AML)	58.9 ±0.2	77.2 ±0.1	**81.7** ±0.0	**69.6** ±0.0	**77.9** ±0.0	**78.6** ±0.2	**66.6** ±0.2	**57.9** ±0.3	**82.3** ±0.1	**74.7** ±0.0	**62.5** ±0.3	84.5 ±0.1	**72.7**

600 images collected from 12 categories. ***Visda-17*** [61] is a widely used benchmark for domain adaptation with focus on a 12-class synthesis-to-real object classification task. The source domain contains 152,397 synthetic images and the target domain has 55,388 real object images. ***DomainNet*** [62] is one of the most challenging datasets in domain adaptation. It contains about 600 thousand images in 345 categories from 6 domains: Clipart (C), Infograph (I), Painting (P), Quickdraw (Q), Real (R) and Sketch (S).

Implementation details. Our experiment is performed on Pytorch platform. To ensure the robustness of the results, each task is performed five times. For fair comparison, we use the same backbone with other methods. Specifically, Resnet-50 is used as the feature extraction network on all datasets. In addition, `Net 1` and `Net 2` use the same neural network structure in our method. For data augmentation, `Net 1` uses standard way, which includes Resize, RandomCrop and Normalize, and `Net 2` further adds ColorJitter. For the hyperparameter α, it sets as 0.01 in both Office-31 dataset, Office-Home dataset, Visda dataset and Domainnet dataset and 0.1 in imageCLEF dataset. For the hyperparameter β, it sets as 0.1 in both Office-31 dataset, Office-Home dataset, Visda dataset and Domainnet dataset and 1.0 imageCLEF. The batch sizes are set as 32 for source and target domain respectively. And the learning rate is set as 0.001 and CosineAnnealingLR [63] is used to update the learning rate during the training.

Competitors. To prove the effectiveness of our method, we compare our method with the following state-of-the-art methods: DANN [9], CDAN [27], TSA [43], ATDOC [35], CKB+MDD [50], GVB+MetaAlign [44], DWL [45], SCDA [46], TCM [51], MCD [53], SWD [54], ETD [56], A^{2}LP [55], CGDM [57], SUDA [47], CaCo [47], DMAL [48], AEO [49] and H-SRDC [52]. In addition, since our method has a certain relationship with the mean-teacher based method, SEDA [17] is also introduced as a comparison.

B. Result analysis

Results on Office-31. The performance of AML is shown in Table I. The overall average performance of our method is 89.5% which is competitive with the state-of-the-art methods. Our method can achieve 100% accuracy in the task W→D and most other methods also achieve 100% accuracy in this task. In the tasks A→W, D→A, D→W, W→A, our method is not too far behind in comparison with the best results.

Results on Office-Home. The comparison result on Office-31 dataset between our method and other state-of-the-art UDA methods is shown in Table II. The overall average performance of our method is 72.7%. Compared with other methods, our method obtains the state-of-the-art performance. From the experimental results, our method achieved the best

TABLE III
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON *ImageCLEF* DATASET. METRIC: CLASSIFICATION ACCURACY (%); BACKBONE: RESNET-50.

Method	I→P	P→I	I→C	C→I	C→P	P→C	*avg*
ResNet-50 [1]	74.8	83.9	91.5	78.0	65.5	91.2	80.7
DANN [9]	75.0	86.0	96.2	87.0	74.3	91.5	85.0
CDAN+E [27]	77.7	90.7	97.7	91.3	74.2	94.3	87.7
MCD [53]	77.3	89.2	92.7	88.2	71.0	92.3	85.1
SWD [54]	78.3	90.3	93.2	89.7	73.3	93.8	86.4
A^2LP [55]	79.6	92.7	96.7	92.5	78.9	96.0	89.4
DWL [45]	82.3	94.8	98.1	92.8	77.9	97.2	90.5
ETD [56]	**81.0**	91.7	**97.9**	**93.3**	79.5	95.0	89.7
CGDM [57]	78.7	93.3	97.5	92.7	79.2	95.7	89.5
CKB+MMD [50]	80.7	92.2	96.5	92.2	79.9	96.7	89.7
DMAL [48]	80.3	93.2	96.5	90.5	76.3	**96.0**	88.8
AEO [49]	79.9	92.6	98.6	93.2	77.5	96.5	89.7
H-SRDC [52]	79.0	92.3	97.0	92.6	77.0	94.7	88.8
Our method (AML)	80.8±0.3	**93.8±0.1**	97.7±0.1	93.2±0.2	**80.2±0.3**	**98.2 ±0.1**	**90.7**

TABLE IV
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON *Visda-17* DATASET. METRIC: PER-CLASS CLASSIFICATION ACCURACY (%); BACKBONE: RESNET-101.

Method	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	*avg*
ResNet-101 [1]	55.1	53.3	61.9	59.1	80.6	17.9	79.7	31.2	81.0	26.5	73.5	8.5	52.4
DANN [9]	81.9	77.7	82.8	44.3	81.2	29.5	65.1	28.6	51.9	54.6	82.8	7.8	57.4
CDAN [27]	85.2	66.9	83.0	50.8	84.2	74.9	**88.1**	74.5	83.4	76.0	81.9	38.0	73.9
DWL [45]	90.7	80.2	**86.1**	67.6	92.4	81.5	86.8	78.0	90.6	57.1	85.6	28.7	77.1
TSA [43]	-	-	-	-	-	-	-	-	-	-	-	-	78.6
ATDOC [43]	93.7	83.0	76.9	58.7	89.7	95.1	84.4	71.4	89.4	80.0	86.7	55.1	80.3
SEDA [17]	95.9	87.4	85.2	58.6	**96.2**	95.7	90.6	80.0	**94.8**	90.8	**88.4**	47.9	84.3
SUDA [47]	88.3	79.3	66.2	64.7	87.4	80.1	85.9	78.3	86.3	87.5	78.8	**74.5**	79.8
CaCo [47]	90.4	80.7	78.8	57.0	88.9	87.0	81.3	79.4	88.7	88.1	86.8	63.9	80.9
DMAL [48]	-	-	-	-	-	-	-	-	-	-	-	-	77.6
Our method (AML)	**96.7**	**88.5**	79.6	**69.0**	95.9	**96.3**	87.3	**83.3**	94.4	**92.9**	87.0	58.7	**85.8**

TABLE V
COMPARISONS WITH THE STATE-OF-THE-ART METHODS ON *DomainNet* DATASET. METRIC: CLASSIFICATION ACCURACY (%); BACKBONE: RESNET-50.

ResNet	clp	inf	pnt	qdr	rel	skt	Avg.	MCD	clp	inf	pnt	qdr	rel	skt	Avg.	BNM	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	14.2	29.6	9.5	43.8	34.3	26.3	clp	-	15.4	25.5	3.3	44.6	31.2	24.0	clp	-	12.1	33.1	6.2	50.8	40.2	28.5
inf	21.8	-	23.2	2.3	40.6	20.8	21.7	inf	24.1	-	24.0	1.6	35.2	19.7	20.9	inf	26.6	-	28.5	2.4	38.5	18.1	22.8
pnt	24.1	15.0	-	4.6	45.0	29.0	23.5	pnt	31.1	14.8	-	1.7	48.1	22.8	23.7	pnt	39.9	12.2	-	3.4	54.5	36.2	29.2
qdr	12.2	1.5	4.9	-	5.6	5.7	6.0	qdr	8.5	2.1	4.6	-	7.9	7.1	6.0	qdr	17.8	1.0	3.6	-	9.2	8.3	8.0
rel	32.1	17.0	36.7	3.6	-	26.2	23.1	rel	39.4	17.8	41.2	1.5	-	25.2	25.0	rel	48.6	13.2	49.7	3.6	-	33.9	29.8
skt	30.4	11.3	27.8	3.4	32.9	-	21.2	skt	37.3	12.6	27.2	4.1	34.5	-	23.1	skt	54.9	12.8	42.3	5.4	51.3	-	33.3
Avg.	24.1	11.8	24.4	4.7	33.6	23.2	20.3	Avg.	28.1	12.5	24.5	2.4	34.1	21.2	20.5	Avg.	37.6	10.3	31.4	4.2	40.9	27.3	25.3
SWD	clp	inf	pnt	qdr	rel	skt	Avg.	CGDM	clp	inf	pnt	qdr	rel	skt	Avg.	AML	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	14.7	31.9	10.1	45.3	36.5	27.7	clp	-	16.9	35.3	10.8	53.5	36.9	30.7	clp	-	18.1	38.2	9.4	53.1	41.9	32.1
inf	22.9	-	24.2	2.5	33.2	21.3	20.0	inf	27.8	-	28.2	4.4	48.2	22.5	26.2	inf	30.7	-	32.0	5.7	50.0	26.7	30.0
pnt	33.6	15.3	-	4.4	46.1	30.7	26.0	pnt	37.7	14.5	-	4.6	59.4	33.5	30.0	pnt	38.4	15.7	-	5.9	56.1	33.7	30.0
qdr	15.5	2.2	6.4	-	11.1	10.2	9.1	qdr	14.9	1.5	6.2	-	10.9	10.2	8.7	qdr	19.1	3.6	7.8	-	10.0	11.9	10.5
rel	41.2	18.1	44.2	4.6	-	31.6	27.9	rel	49.4	20.8	47.2	4.8	-	38.2	32.0	rel	54.2	22.1	50.2	7.1	-	41.7	35.1
skt	44.2	15.2	37.3	10.3	44.7	-	30.3	skt	50.1	16.5	43.7	11.1	55.6	-	35.4	skt	52.1	16.7	42.9	13.2	56.9	-	36.4
Avg.	31.5	13.1	28.8	6.4	36.1	26.1	23.6	Avg.	36.0	14.0	32.1	7.1	45.5	28.3	27.2	Avg.	38.9	15.2	34.2	8.3	45.2	31.2	**28.8**

performance on 9 tasks out of 12 tasks. In the remaining three tasks A→C, R→P and A→P, our method achieves second performance, which only lags behind by 0.4% and 0.7% compared to GVB+MetaAlign [44] and by 0.4% compared to ATDOC [35].

Results on ImageCLEF. The results of our method AML and other state-of-the-art methods are reported in Table III. Our method achieves the highest accuracy in 3 out of 6 tasks and overall performance. Compared with CKB+MMD [50], AML leads the overall performance by 1.0%, especially on task P→C, which performance are improved by 1.5%. On tasks I→P, task I→C and task C→I, our method only lags behind the optimal performance by 0.2%, 0.2% and 0.1%, respectively.

Results on Visda. The comparison result on Visda dataset between our method AML and other state-of-the-art UDA methods is shown in Table IV, and the per-class classification accuracy is reported. From the result, AML achieves the best performance on plane, bcycl, car, knife, person and sktbrd classes in the compared methods. For other classes, AML also yields good results and not far behind the best compared methods. Compared with SEDA, our method improves the

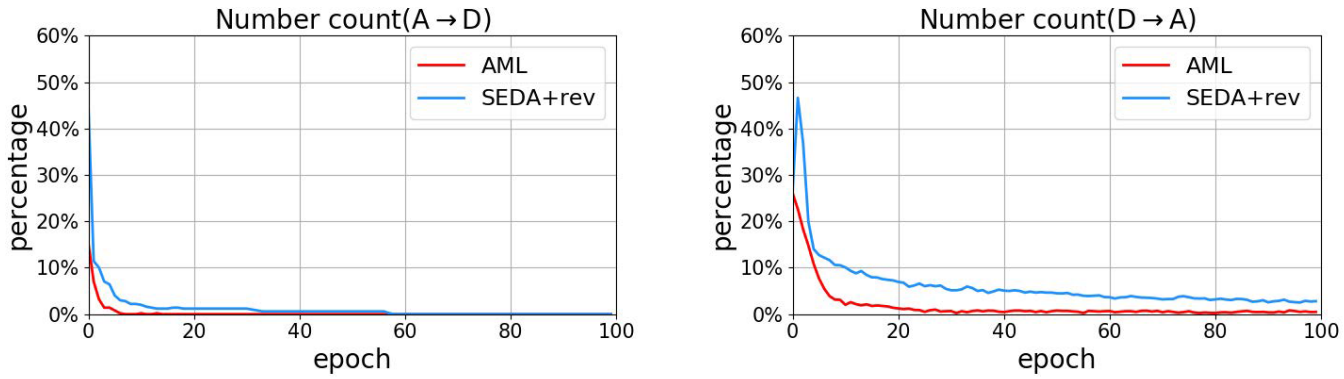


Fig. 4. In the training progress, the percentage of target domain samples that are misclassified by the teacher network but correctly classified by the student network.

TABLE VI
ABLATION STUDY ON *Office-31* DATASET.

Method	A→D	A→W	D→A	D→W	W→A	W→D
SEDA	87.1	90.8	70.7	97.6	72.2	99.8
SEDA+rev	89.8	91.8	72.9	98.6	75.0	99.8
AML	**92.4**	**94.2**	**75.5**	**98.9**	**75.9**	**100.0**

TABLE VII
COMPARISON OF ROLE SELECTION STRATEGY USING DIFFERENT EVALUATION CRITERION.

Method	A→D	A→W	D→A	D→W	W→A	W→D
AML(EN)	92.4	94.2	75.5	98.9	75.9	100.0
AML(KL)	92.6	93.9	75.8	99.0	75.5	100.0
AML(ROM)	87.8	86.2	70.7	96.6	67.5	99.2

TABLE VIII
COMPARISON OF EACH NETWORK ON DIFFERENT DATASETS.

Acc(%)	Office-home	Office-31	Image-CLEF	Visda	Domainnet
Net1	72.5	89.3	90.2	81.2	28.3
Net2	72.4	89.2	90.1	80.9	28.1
AML	72.7	89.5	90.7	81.8	28.8

overall performance by 1.5%.

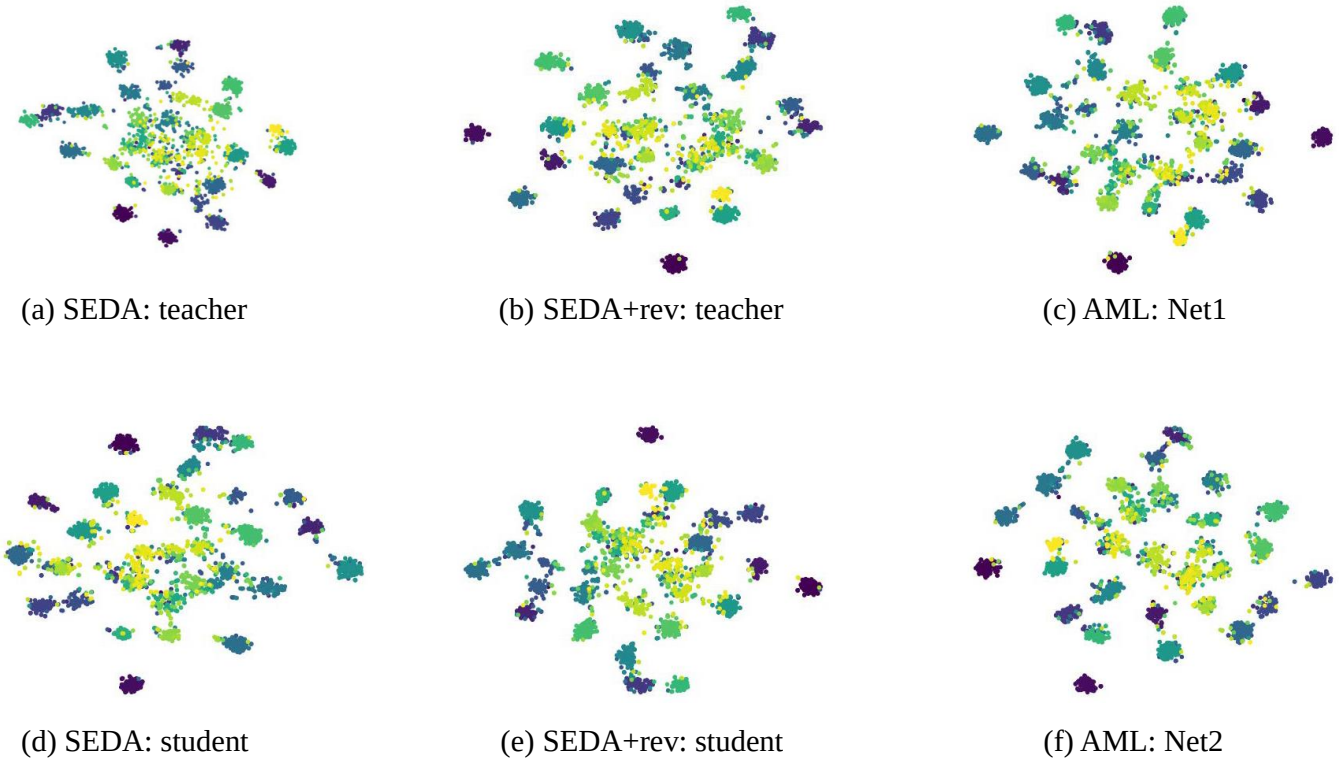
Results on DomainNet. The comparison result on DomainNet dataset between our method AML and other state-of-the-art UDA methods is shown in Table V, and the classification accuracy is reported. For each cross-domain pair, the source domains are specified in the corresponding row fields and the target domains are specified in the corresponding column fields. In this dataset, our method also yields the best performance in the compared methods, which improves performance by 1.6% compared with the CGDM and 3.5% compared with BNM.

From Table I to Table V, we can summarize the following three observations. First, compared with the baseline ResNet [1], which trains a model in the source domain and applies it directly to the target domain, all UDA methods can greatly improve performance. It proves that domain adaptation can help the model generalize to the target domain. Second, compared with SEDA [17], our method can significantly improve the performance, which proves that the adaptive role selection of teacher and student and reverse knowledge distillation strategies works. Finally, our method has improved performance on both datasets, especially on the office-home dataset, which reflects the effectiveness of our method to a certain extent.

C. Model analysis

Ablation study. In this work, our core contribution is to propose a role selection strategy and a reverse knowledge distillation strategy. Therefore, in this experiment, we mainly verify these two points instead of studying the role of each loss function like the previous work. The ablation study is conducted on Office-31 dataset which is shown in Table VI. SEDA [17] means the basic mean-teacher based method, which is used as a baseline. SEDA+rev represents the original mean-teacher framework adding the reverse knowledge distillation strategy. For the update of the student network, it is based on the consistency constraints like the traditional mean-teacher method, whereas for the update of the teacher network, it uses reverse knowledge distillation to require its predictions away from the student network. Comparing SEDA+rev with SEDA, we can find that reverse knowledge distillation strategy can indeed improve performance. AML is our complete algorithm, which further adds the role selection strategy based on SEDA+rev. From the experimental results, the effectiveness of role selection strategy is also verified.

Misclassified sample statistics and the effectiveness of role selection. In this part, we aim to verify the hypothesis that the teacher network not always outperforms the student network on all target domain samples due to the domain shift, and further verify the effectiveness of role selection strategy. The SEDA+rev and the AML are performed, the difference between them is whether or not a role selection strategy is used. Specifically, we count the percentage of target domain samples that incorrectly predicted by the teacher network but correctly predicted by the student network. Considering whether the discrepancy of domain shift is severe, we perform two tasks of Office-31, A→D and D→A, respectively. As shown in Fig. 4, the red line reprents the result of SEDA+rev



(a) SEDA: teacher (b) SEDA+rev: teacher (c) AML: Net1

(d) SEDA: student (e) SEDA+rev: student (f) AML: Net2

Fig. 5. Visualization with t-SNE for different methods. **Left**: SEDA. **Center**: SEDA + Reverse Knowledge Distillation. **Right**: AML. The results are on Office-31 task D→A.

TABLE IX
COMPARISON WITH THE STATE-OF-THE-ART METHODS ON *Office-Home* DATASET FOR PARTIAL-SET UDA (PDA) SETTING. METRIC: CLASSIFICATION ACCURACY (%); BACKBONE: RESNET-50.

Method	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	*avg*
ResNet-50 [1]	43.5	67.8	78.9	57.5	56.2	62.2	58.1	40.7	74.9	68.1	46.1	76.3	60.9
Pseudo-labeling	51.9	70.7	77.5	61.7	62.4	67.8	62.9	54.1	73.8	70.4	56.7	75.0	65.4
MinEnt [64]	45.7	73.3	81.6	64.6	66.2	73.0	66.0	52.4	78.7	74.8	56.7	80.8	67.8
MCC [65]	54.1	75.3	79.5	63.9	66.3	71.8	63.3	55.1	78.0	70.4	55.7	76.7	67.5
BNM [66]	54.6	77.2	81.1	64.9	67.9	72.8	62.6	55.7	79.4	70.5	54.7	77.6	68.2
ATDOC [35]	59.5	**80.3**	83.8	71.8	71.6	79.7	70.6	59.4	82.2	**78.4**	**61.1**	81.5	73.3
ETN [67]	59.2	77.0	79.5	62.9	65.7	75.0	68.3	55.4	84.4	75.7	57.7	84.5	70.5
SAFN [68]	58.9	76.3	81.4	70.4	**73.0**	77.8	72.4	55.3	80.4	75.8	60.4	79.9	71.8
RTNet [69]	**63.2**	80.1	80.7	66.7	69.3	77.2	71.6	53.9	**84.6**	77.4	57.9	**85.5**	72.3
Our method (AML)	60.9	77.7	**84.5**	**73.1**	71.3	**81.4**	**74.0**	**62.3**	84.2	77.8	60.3	81.2	**74.1**

and the blue line reprents the result of AML.

From the results, we can get following conclusion: First, teacher network do not necessarily perform better than student network, especially when the domain discrepancy is large. Second, regardless of whether the domain discrepancy is huge, the method using the role selection strategy can minimize the number of samples that the teacher network predicts incorrectly but the student network predicts correctly. Third, the method using the role selection strategy can converge the model faster.

Role selection strategy based on KL divergence. To further analyze role selection strategy, we introduce KL divergence as an evaluation criterion for experiments. Specifically, before the start of each epoch, we average the features extracted by the two networks, and then calculate the source cluster center for each category of source domain. After that, K-means clustering algorithm [70] is performed on the target domain features to obtain the pseudo-label $\{\boldsymbol{y}_i^t\}_{i=1}^{n_t}$ of each target domain sample, where the cluster centers of the source domain are used as the initial cluster centers of K-means clustering algorithm. Finally, in each training process, we calculate the KL divergence between the predictions of the two networks and the pseudo-label, and divide the target domain by comparing the KL divergence, which is as follows:

$$B_1^t = \{x_i^t | KL(\boldsymbol{p}_{i,1}^t, \boldsymbol{y}_i^t) < KL(\boldsymbol{p}_{i,2}^t, \boldsymbol{y}_i^t)\},$$
$$B_2^t = \{x_i^t | KL(\boldsymbol{p}_{i,1}^t, \boldsymbol{y}_i^t) > KL(\boldsymbol{p}_{i,2}^t, \boldsymbol{y}_i^t)\},$$

where $KL(A, B)$ means the KL divergence between A and B. Then subsequent operations proceed as mentioned in our method section. The experimental results are shown in Table VII, where AML(EN) uses entropy as the evaluation criterion and AML(KL) uses the KL divergence as the evaluation

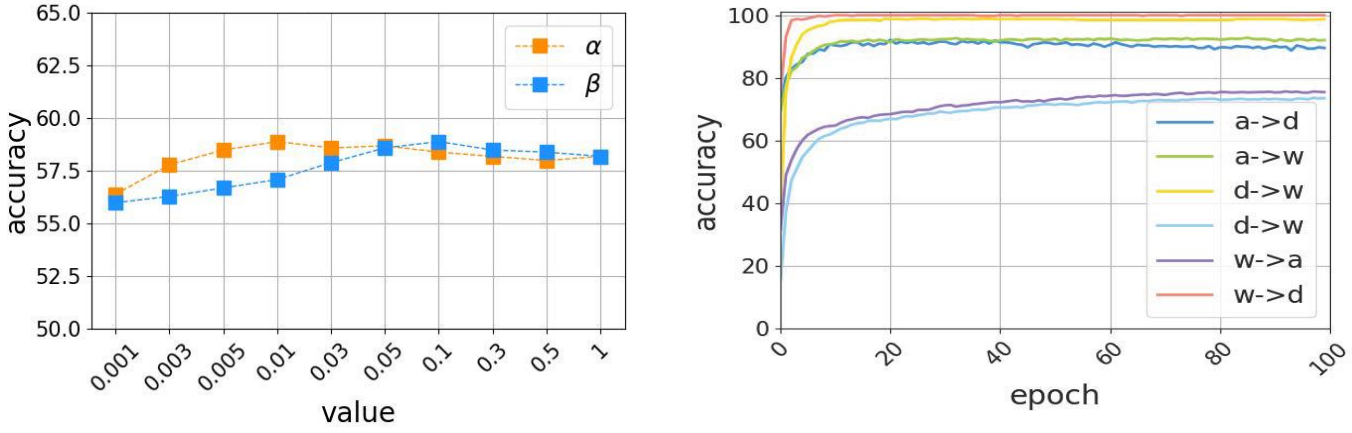


Fig. 6. (a) Parameter sensitivity analysis of AML on Office-Home dataset task A→C. (b) Convergence curves of AML on Office-31 dataset.

criterion. From the experimental results, the results of different evaluation criteria do not vary greatly. So in this work, the entropy is used as the evaluation criteria, which is relatively simple. In addition, we further evaluated the performance of our method when setting the teacher-student network randomly, which is shown in AML(ROM).

Visual analysis. To give an intuitive understanding of our method, the target features learned by both networks of transfer task D→A on Office-31 dataset are visualized by t-SNE [71] in Fig.5. Fig.5(a) shows the visualization of the features learned by the teacher network in the mean-teacher based framework [17]; Fig.5(d) represents the visualization of the features learned by the student network in the mean-teacher based framework. Fig.5(b) and Fig.5(e) are the visualizations of the features learned by the teacher network and the student network respectively, which is added with the reverse knowledge distillation strategy. Fig.5(c) and Fig.5(f) are the results of feature visualization of `Net 1` and `Net 2` in our algorithm, respectively.

From the results, there are two observations. First, the comparison between Fig.5(a) and Fig.5(d) and the comparison between Fig.5(b) and Fig.5(e) cannot show that the teacher network is definitely better than the student network. Second, from left to right, the features in Fig.5 are more and more compact, which proves the effectiveness of role selection strategy and mutual learning strategy.

Parameter analysis. We perform sensitivity analysis on two hyperparameters α and β in our algorithm. The task A→C on Office-Home dataset is performed and the result is shown in Fig. 6(a). The α is ranged from 0.001 to 1 when β is fixed as 0.1, which is shown in orange line. And β is ranged from 0.001 to 1 when α is fixed as 0.01, which is shown in blue line. From the experimental results, the performance of the model will increase first and then decrease with the increase of α and β. The range of model performance changes is not large, which prove that our model is robust respect to these two parameters.

Convergence analysis. The accuracy curves of target samples on Office-31 dataset during training process are depicted in Fig. 6(b). It shows that as the number of epochs increases, the accuracy is improved and finally reaches a plateau, demonstrating that the training process is smooth and convergent.

Results of `Net 1` and `Net 2`. In this task, we present the overall performance of `Net 1` and `Net 2` on each dataset. We also list the performance of AML, which is the ensemble of the two networks, as shown in the Table VIII. From this table we can see that the result of AML can achieve better performance compared to a single network. And our single-network is also competitive compared to existing alternatives.

Results of Partial-set UDA. In this experiments, we adopt the standard partial-set UDA setting as [67] on office-home dataset, where target domain is consists of data from the first 25 categories, and the result is shown in Table IX. Compared with other methods, AML still still produces a competitive result, which also proves that our method can be easily transferred to other scenarios.

V. Conclusion

Traditional mean-teacher based UDA methods always distill knowledge from the teacher to the student. The roles of teacher and student are fixed and not all target samples are used in domain adaptation. Due to existence of domain shift, this one-way knowledge distillation will bring negative transfer. In this paper, we proposed a novel adaptive mutual learning method to address these limitations. For different target samples, the role of teacher or student is adaptively selected based on the entropy of network predictions. Then traditional knowledge distillation can be employed from teacher network to student network; and reverse knowledge distillation is proposed to further render teacher network more discriminative. The experimental results on the public datasets validate the efficacy of our method.

Acknowledgments

This work was supported in part by the National Key R&D Program of China (2018YFE0203900), National Natural Science Foundation of China (62276048), Sichuan Science and Technology Program (2020YFG0476).

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. of CVPR*, 2016.

[2] Y. Zuo, H. Yao, L. Zhuang, and C. Xu, "Margin-based adversarial joint alignment domain adaptation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 4, pp. 2057–2067, 2022.

[3] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, 2015.

[4] B. Zhao, J. Feng, X. Wu, and S. Yan, "A survey on deep learning-based fine-grained object classification and semantic segmentation," *International Journal of Automation and Computing*, 2017.

[5] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, 2009.

[6] M. Meng, Z. Wu, T. Liang, J. Yu, and J. Wu, "Exploring fine-grained cluster structure knowledge for unsupervised domain adaptation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 8, pp. 5481–5494, 2022.

[7] X. Xu, H. He, H. Zhang, Y. Xu, and S. He, "Unsupervised domain adaptation via importance sampling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 12, pp. 4688–4699, 2020.

[8] L. Zhou, M. Ye, X. Li, C. Zhu, Y. Liu, and X. Li, "Disentanglement then reconstruction: Learning compact features for unsupervised domain adaptation," *arXiv preprint arXiv:2005.13947*, 2020.

[9] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *The journal of machine learning research*, 2016.

[10] M.-Y. Liu and O. Tuzel, "Coupled generative adversarial networks," *Proc. of NIPS*, 2016.

[11] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. of ICML*, 2015.

[12] L. Zhang, P. Wang, W. Wei, H. Lu, C. Shen, A. van den Hengel, and Y. Zhang, "Unsupervised domain adaptation using robust class-wise matching," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 5, pp. 1339–1349, 2019.

[13] M. Ghifary, W. B. Kleijn, M. Zhang, and D. Balduzzi, "Domain generalization for object recognition with multi-task autoencoders," in *Proc. of ICCV*, 2015.

[14] M. Ghifary, W. B. Kleijn, M. Zhang, D. Balduzzi, and W. Li, "Deep reconstruction classification networks for unsupervised domain adaptation," in *Proc. of ECCV*, 2016.

[15] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. of ICONIP*, 2017.

[16] Q. Cai, Y. Pan, C.-W. Ngo, X. Tian, L. Duan, and T. Yao, "Exploring object relation in mean teacher for cross-domain detection," in *Proc. of CVPR*, 2019.

[17] G. French, M. Mackiewicz, and M. Fisher, "Self-ensembling for visual domain adaptation," in *Proc. of ICLR*, 2018.

[18] Z. Luo, Z. Cai, C. Zhou, G. Zhang, H. Zhao, S. Yi, S. Lu, H. Li, S. Zhang, and Z. Liu, "Unsupervised domain adaptive 3d detection with multi-level consistency," in *Proc. of ICCV*, 2021.

[19] Z. Zheng and Y. Yang, "Unsupervised scene adaptation with memory regularization in vivo," in *Proc. of IJCAI*, 2020.

[20] Z. Ke, D. Wang, Q. Yan, J. Ren, and R. W. Lau, "Dual student: Breaking the limits of the teacher in semi-supervised learning," in *Proc. of ICCV*, 2019, pp. 6728–6736.

[21] Q. Tian, Y. Zhu, H. Sun, S. Chen, and H. Yin, "Unsupervised domain adaptation through dynamically aligning both the feature and label spaces," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2022.

[22] J. Tian, J. Zhang, W. Li, and D. Xu, "Vdm-da: Virtual domain modeling for source data-free domain adaptation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 6, pp. 3749–3760, 2022.

[23] L. Zhou, M. Ye, D. Zhang, C. Zhu, and L. Ji, "Prototype-based multisource domain adaptation," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[24] L. Zhou, M. Ye, and S. Xiao, "Domain adaptation based on source category prototypes," *Neural Computing and Applications*, pp. 1–13, 2022.

[25] J. Zhuo, S. Wang, W. Zhang, and Q. Huang, "Deep unsupervised convolutional domain adaptation," in *Proc. of ACMM*, 2017.

[26] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *Proc. of CVPR*, 2019.

[27] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Proc. of ICONIP*, 2018.

[28] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. of CVPR*, 2017.

[29] R. Zhu, X. Jiang, J. Lu, and S. Li, "Cross-domain graph convolutions for adversarial unsupervised domain adaptation," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[30] G. Wei, C. Lan, W. Zeng, Z. Zhang, and Z. Chen, "Toalign: task-oriented alignment for unsupervised domain adaptation," *Advances in Neural Information Processing Systems*, vol. 34, pp. 13 834–13 846, 2021.

[31] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proc. of CVPR*, 2017.

[32] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. of ICCV*, 2017.

[33] W. Shi, R. Zhu, and S. Li, "Pairwise adversarial training for unsupervised class-imbalanced domain adaptation," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 1598–1606.

[34] J. Na, D. Han, H. J. Chang, and W. Hwang, "Contrastive vicinal space for unsupervised domain adaptation," in *17th European Conference on Computer Vision (ECCV 2022)*, 2022.

[35] J. Liang, D. Hu, and J. Feng, "Domain adaptation with auxiliary target domain-oriented classifier," in *Proc. of CVPR*, 2021.

[36] Z. Deng, Y. Luo, and J. Zhu, "Cluster alignment with a teacher for unsupervised domain adaptation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9944–9953.

[37] J. Na, H. Jung, H. J. Chang, and W. Hwang, "Fixbi: Bridging domain spaces for unsupervised domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1094–1103.

[38] H. Tang, K. Chen, and K. Jia, "Unsupervised domain adaptation via structurally regularized deep clustering," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8725–8735.

[39] J. Liang, D. Hu, Y. Wang, R. He, and J. Feng, "Source data-absent unsupervised domain adaptation through hypothesis transfer and labeling transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 8602–8617, 2021.

[40] R. Gomes, A. Krause, and P. Perona, "Discriminative clustering by regularized information maximization," in *Proc. of ICONIP*, 2010.

[41] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, 2001.

[42] J. R. Hershey and P. A. Olsen, "Approximating the kullback leibler divergence between gaussian mixture models," in *Proc. of ICASSP*. IEEE, 2007.

[43] S. Li, M. Xie, K. Gong, C. H. Liu, Y. Wang, and W. Li, "Transferable semantic augmentation for domain adaptation," in *Proc. of CVPR*, 2021.

[44] G. Wei, C. Lan, W. Zeng, and Z. Chen, "Metaalign: Coordinating domain alignment and classification for unsupervised domain adaptation," in *Proc. of CVPR*, 2021.

[45] N. Xiao and L. Zhang, "Dynamic weight- ed learning for unsupervised domain adaptation," in *Proc. of CVPR*, 2021.

[46] S. Li, M. Xie, F. Lv, C. H. Liu, J. Liang, C. Qin, and W. Li, "Semantic concentration for domain adaptation," in *Proc. of ICCV*, 2021.

[47] J. Zhang, J. Huang, Z. Tian, and S. Lu, "Spectral unsupervised domain adaptation for visual recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9829–9840.

[48] J. Huang, N. Xiao, and L. Zhang, "Balancing transferability and discriminability for unsupervised domain adaptation," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[49] A. Ma, J. Li, K. Lu, L. Zhu, and H. T. Shen, "Adversarial entropy optimization for unsupervised domain adaptation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 11, pp. 6263–6274, 2022.

[50] Y.-W. Luo and C.-X. Ren, "Conditional bures metric for domain adaptation," in *Proc. of CVPR*, 2021.

[51] Z. Yue, Q. Sun, X.-S. Hua, and H. Zhang, "Transporting causal mechanisms for unsupervised domain adaptation," in *Proc. of ICCV*, 2021.

[52] H. Tang, X. Zhu, K. Chen, K. Jia, and C. L. P. Chen, "Towards uncovering the intrinsic data structures for unsupervised domain adaptation using structurally regularized deep clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 6517–6533, 2022.

[53] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proc. of CVPR*, 2018.

[54] C.-Y. Lee, T. Batra, M. H. Baig, and D. Ulbricht, "Sliced wasserstein discrepancy for unsupervised domain adaptation," in *Proc. of CVPR*, 2019.

[55] Y. Zhang, B. Deng, K. Jia, and L. Zhang, "Label propagation with augmented anchors: A simple semi-supervised learning baseline for unsupervised domain adaptation," in *Proc. of ECCV*. Springer, 2020.

[56] M. Li, Y.-M. Zhai, Y.-W. Luo, P.-F. Ge, and C.-X. Ren, "Enhanced transport distance for unsupervised domain adaptation," in *Proc. of CVPR*, 2020.

[57] Z. Du, J. Li, H. Su, L. Zhu, and K. Lu, "Cross-domain gradient discrepancy minimization for unsupervised domain adaptation," in *Proc. of CVPR*, 2021.

[58] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Proc. of ECCV*. Springer, 2010.

[59] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proc. of CVPR*, 2017.

[60] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proc. of ICML*. PMLR, 2017.

[61] X. Peng, B. Usman, N. Kaushik, J. Hoffman, D. Wang, and K. Saenko, "Visda: The visual domain adaptation challenge," *arXiv preprint arXiv:1710.visda06924*, 2017.

[62] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 1406–1415.

[63] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," *arXiv preprint arXiv:1608.03983*, 2016.

[64] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," in *Proceedings of the 17th International Conference on Neural Information Processing Systems*, 2004, pp. 529–536.

[65] Y. Jin, X. Wang, M. Long, and J. Wang, "Minimum class confusion for versatile domain adaptation," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*. Springer, 2020, pp. 464–480.

[66] S. Cui, S. Wang, J. Zhuo, L. Li, Q. Huang, and Q. Tian, "Towards discriminability and diversity: Batch nuclear-norm maximization under label insufficient situations," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 3941–3950.

[67] Z. Cao, K. You, M. Long, J. Wang, and Q. Yang, "Learning to transfer examples for partial domain adaptation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2985–2994.

[68] R. Xu, G. Li, J. Yang, and L. Lin, "Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1426–1435.

[69] Z. Chen, C. Chen, Z. Cheng, B. Jiang, K. Fang, and X. Jin, "Selective transfer with reinforced transfer network for partial domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 706–12 714.

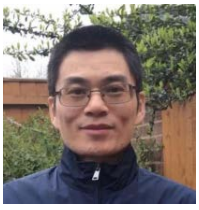
[70] K. Krishna and M. N. Murty, "Genetic k-means algorithm," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 1999.

[71] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, 2008.

Lihua Zhou received the B.S. degree in Internet of Things engineering from the Hefei University of Technology, Xuancheng, China, in 2019. He is currently pursuing the Ph.D. degree with the University of Electronic Science and Technology of China, Chengdu, China. His current research interests include machine learning, computer vision, and transfer learning.

Siying Xiao received the B.S. degree in Computer Science and Technology from the University of Electronic Science and Technology of China, Chengdu, China in 2022. She is currently pursuing the M.S. degree with the University of Electronic Science and Technology of China, Chengdu, China. Her current research interests include machine learning, computer vision, and transfer learning.

Mao Ye received the B.S. degree from Sichuan Normal University, Chengdu, China, in 1995, and the M.S degree from University of Electronic Science and Technology of China, Chengdu, China, in 1998 and Ph.D. degree from Chinese University of Hong Kong, China, in 2002, all in mathematics. He has been a short-time visiting scholar at University of Queensland, and University of Pennsylvania. He is currently a professor and director of CVLab with University of Electronic Science and Technology of China, Chengdu, China. His research interests include machine learning and computer vision. In these areas, he has published over 90 papers in leading international journals or conference proceedings. He has served on the editorial board of ENGINEERING APPLICATIONS OF ARTIFICIAL INTELLIGENCE. He was a co-recipient of the Best Student Paper Award at the IEEE ICME 2017.



Xiatian Zhu is a Senior Lecturer with Surrey Institute for People-Centred Artificial Intelligence, and Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, Guildford, UK. He received his Ph.D. degree from the Queen Mary University of London. He won the Sullivan Doctoral Thesis Prize 2016. He was a research scientist at Samsung AI Centre, Cambridge, UK. His research interests include computer vision, and machine learning.

Shuaifeng Li received the B.S. degree in Computer Science and Technology from the University of Electronic Science and Technology of China, Chengdu, China, in 2020. He is currently pursuing the Ph.D. degree at the University of Electronic Science and Technology of China, Chengdu, China. His current research interests include computer vision and transfer learning.