2022 26th International Conference on Pattern Recognition (ICPR)
August 21-25, 2022, Montréal, Québec, Canada

Discriminative Mutual Learning for Multi-target Domain Adaptation

Jie Wang*, Chaoliang Zhong*, Cheng Feng*, Ying Zhang*, Jun Sun* and Yasuto Yokota†
*Fujitsu R&D Center, Co., LTD
†Fujitsu LTD

2022 26th International Conference on Pattern Recognition (ICPR) | 978-1-6654-9062-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICPR56361.2022.9956437

***Abstract*—Unsupervised domain adaptation (UDA) has attracted much attention among those seeking to transfer a model from a labeled source domain to an unlabeled target domain. Many effective algorithms for single target domain adaptation (STDA) have been designed, however, STDA cannot satisfy the scenarios of transferring simultaneously to multiple target domains or transferring to a blending target domain. This paper proposes a novel discriminative mutual learning method for multi-target domain adaptation covering both blending target domain adaptation (BTDA) and multiple target domain adaptation (MTDA). Two key points are considered in the proposed method: one is to learn discriminative features for better prediction, and the other is to self-train the model with pseudo-labeled target data based on distance information. These two aspects are integrated through a mutual learning strategy via two different classifiers. According to extensive experiments on three domain adaptation benchmarks, the proposed method demonstrates the state-of-the-art performance in both BTDA and MTDA settings.**

I. Introduction

Deep learning has achieved much success in computer vision tasks, such as image recognition [1], object detection [2] and segmentation [3]. However, it suffers from poor generalization in real-world applications in the presence of the domain shift [4]. The domain shift arising from data distribution discrepancy reduces the transferability of a well-trained model from the training set to a different testing set. Considering the cost of labeling, unsupervised domain adaptation (UDA) [5] has been proposed to alleviate the negative effect caused by domain shift for a desired target domain by leveraging the unlabeled target data. Diverse UDA algorithms have shown good performance against domain shift, such as moment matching [6], [7], [8], reconstruction [9], adversarial training [10], [11], semantic alignment [12], [13] and self-training [14], [15]. Although these methods improve the transferability of the model remarkably, majority of them focus on single target domain adaptation (STDA) regardless of the more challenging scenarios of multi-target domain adaptation [16], [17].

The multi-target domain adaptation contains two scenarios, one is blending target domain adaptation (BTDA) [16], [18] where multiple target domains are mixed without domain labels, and the other is multiple target domain adaptation (MTDA) [17], [19] where the domain label of a target sample is known. STDA methods [7], [10] are regarded to be sub-optimal for multi-target domain adaptation [16]. For BTDA, current STDA methods cannot handle the heterogeneity of the blending target domain. More compact intra-class features should be learned to overcome the issues regarding the intra-class diversity of the target domain. For MTDA, the direct solution with STDA is to train a specific model for each target domain, however, with the increase in the number of target domains, the number of models needing to be trained also increases, making a unified model a better choice. Therefore, suitable algorithms should be developed for multi-target domain adaptation. Some approaches for multi-target domain adaptation have been developed recently from different perspectives, such as feature disentanglement [20], [17] and knowledge distillation [19].

Different from existing methods, here we propose a novel method named Discriminative Mutual Learning (DML) based on two key points, one is to learn discriminative features and the other is to self-train the model with pseudo-labeled target data. These two aspects are combined by two classifiers to provide mutual promotion to attain good performance for multi-target domain adaptation. In particular, Discriminative Curriculum Mutual Learning (DCML) is further proposed to leverage the known domain label for MTDA. In other words, we sequentially adapt the model from the easy target domain to its harder counterpart.

Our contributions are demonstrated as follows:

- A novel DML method is proposed for multi-target domain adaptation. The proposed method learns discriminative features to support self-training to achieve better transferability in both BTDA and MTDA.
- DCML is further presented to leverage the known domain label for MTDA with curriculum learning. The easy-to-hard transfer promotes smoother, stronger adaptation.
- Our method achieves very competitive performance compared with the state-of-the-art in both BTDA and MTDA settings.

II. Related Work

Unsupervised domain adaptation (UDA). UDA has attracted much attention in recent years [6], [7], [10], [12]. Moment matching was early proposed to align the statistical metrics of feature between source and target domains, such as feature covariances [8] and the maximum mean discrepancy [6], [7]. Adversarial training [10], [21] was developed to establish a widely-used baseline. Instead of matching a handcrafted statistic between two domains, adversarial training extracted domain-invariant features by a min-max player [21] or a gradient reversal layer [10]. The feature alignment via either moment matching [6], [8] or adversarial training [10] constrains the alignment at the domain-level, but ignores the category information within the domain [11]. False alignment or under alignment would occur when different categories are matched but identical categories are not mutually aligned. To solve this problem, multiple [11] and conditional [22] adversarial training approaches were proposed to align features with the same category from two domains. Similarly, semantic alignment [12] was proposed to match the feature centers of each category. These class-level alignment methods [11], [22], [12] leveraged the predictions of the MLP classifier as pseudo-labels for target data to obtain their category information. The pseudo-labels of target data also promoted the application of self-training in UDA [14], [23], [24]. More research has been conducted to improve the accuracy of pseudo-labels [15]. However, all of the aforementioned methods were designed for STDA which are suboptimal when directly applied to multi-target domain adaptation scenarios.

Multi-target domain adaptation. Multi-target domain adaptation is studied in two situations, one is blending-target domain adaptation where target domains are mixed without information of domain labels [16], [18], and the other is multiple target domain adaptation where target domains are separate with information of domain labels [17], [19]. For BTDA, feature disentanglement [20] was used to draw domain-invariant features between source and target domains with no explicit way to deal with the heterogeneity within the target domain. Meta-learning [16] was proposed to separate sub-domains from the blended target domain to conduct adversarial training between each target sub-domain and source domain. However, the adaptation method limits its performance since adversarial training is less effective. Differing from current methods, we propose to learn compact intra-class features to overcome the heterogeneity within the target domains and the proposed mutual learning strategy is a more powerful adaptation method.

For MTDA, information theory [17] inspired a simple but reasonable method with ordinary performance. Knowledge distillation was leveraged to connect STDA and MTDA by transferring knowledge from multiple STDA models for each target domain to one joint model [19]. However, this strategy still faces the problem of training the same number of adaptive STDA models as the number of target domains. Other researchers employed a graph neural network to conduct the semantic propagation among target domains [25] or to model the relationship among intra-class and inter-class samples [26].

Dual classifier. The classification boundary plays an important role in model transferability [27] and samples around the decision boundary are regarded as uncertain ones. Since one classifier is random among many candidates and two classifiers capture different patterns of the training data, two classifiers were used to identify hard target samples [27] or noisy samples [28]. Except for the MLP classifier, a graph neural network was adopted as an additional classifier to provide robust predictions [26]. Different from using a second MLP classifier or graph classifier, here we propose a simple but effective nearest neighbour classifier based on feature distance to couple with the MLP classifier to form a mutual learning strategy.

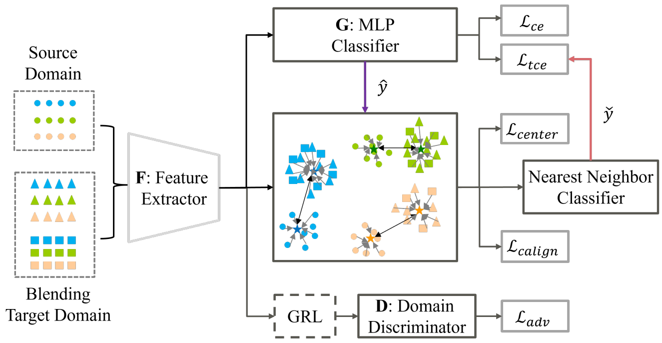


Fig. 1: (Best viewed in color) The proposed discriminative mutual learning for BTDA. The MLP classifier provides pseudo-labels $\hat{y}$ of target data for learning discriminative features, and in a mutual fashion, the nearest neighbour classifier provides pseudo-labels $\check{y}$ of target data for self-training. These two classifiers promote each other to achieve better performance.

III. Methodology

A. Problem Definition

In the blending target domain adaptation (BTDA) setting, given a labeled source domain $\mathcal{S} = \{(x_s^i, y_s^i)\}_{i=1}^{n_s}$ and a mixed unlabeled target domain $\mathcal{T} = \{x_t^j\}_{j=1}^{n_t}$, the aim is to train an adaptive model for the blended target domain. In the multiple target domain adaptation (MTDA), given a labeled source domain $\mathcal{S} = \{(x_s^i, y_s^i)\}_{i=1}^{n_s}$ and M unlabeled target domains $\mathcal{T} = \{\mathcal{T}_m\}_{m=1}^{M}$ where $\mathcal{T}_m = \{x_{t_m}^j\}_{j=1}^{n_{t_m}}$, the aim is to train a single robust model for M target domains. We accept the same assumption as other UDA methods [6], [10] that the data distributions between $\mathcal{S}$ and $\mathcal{T}$ are different but the label space is the same.

B. Discriminative Mutual Learning

Learning discriminative features and self-training with pseudo-labels are two key points to our method, which promote each other by a mutual learning strategy as illustrated in Fig. 1.

1) Baseline method: We employ the adversarial training as our baseline method like [12], [15] and adopt the same neural network design as DANN [10]. The model consists of a feature extractor F, a multi-layer perceptron (MLP) classifier G, and a domain discriminator D. The domain discriminator is designed to classify which domain the input feature originates from and it is connected to the feature extractor by a gradient reversal layer (GRL). D discriminates the features of source domain from the features of target domain, while GRL makes the optimization of F to extract similar cross-domain features to fool D. Therefore, the adversarial training enables F to learn domain-invariant features. Generally, the output of the domain discriminator is a single value $\bar{p}$ denoting the probability of the sample that belongs to the source domain in the setting of STDA and BTDA, but in the setting of MTDA it is an

$M+1$ length vector $\bar{\mathbf{p}}$ denoting the probability of the sample belonging to each domain. In the BTDA setting, the adversarial training loss is defined as the binary cross entropy loss

$$\mathcal{L}_{adv} = -\sum_{i=1}^{n} (\bar{y} \log(\bar{p}_{x_i}) + (1-\bar{y}) \log(1-\bar{p}_{x_i})), \quad (1)$$

where $\bar{y} = 1$ if sample x_i belongs to the source domain, otherwise $\bar{y} = 0$. In the MTDA setting, the adversarial training loss is defined as the cross entropy loss

$$\mathcal{L}_{adv} = -\sum_{i=1}^{n} \bar{\mathbf{y}} \log(\bar{\mathbf{p}}_{x_i}), \quad (2)$$

where $\bar{\mathbf{y}}$ represents the one-hot vector of the domain label with length of $M+1$.

2) Learn discriminative features: Learning discriminative features contributes to the robustness of the model. In the blending target domain, since the heterogeneity arises from the intra-class discrepancy, the key point is the compactness of intra-class features. To aggregate samples with the same label, we employ the center loss [29] which was proposed for face recognition to learn discriminative features. The idea is to push samples to their class centers to achieve separable inter-class features. The class centers are updated within each mini-batch as the mean feature of each class. In BTDA, center loss for both source domain and target domain is performed separately. The predictions $\hat{y}$ of unlabeled target data from the MLP classifier are used as their pseudo-labels in the center loss for the blending target domain, while the ground-truth labels of source data are applied in the center loss for the source domain. Specifically, the center loss is formulated by

$$\begin{aligned} \mathcal{L}_{center} &= \mathcal{L}_{center}^{s} + \mathcal{L}_{center}^{t} \\ &= \sum_{i=1}^{n_s} ||F(x_s^i) - \boldsymbol{c}_{y_i}||^2 + \sum_{j=1}^{n_t} ||F(x_t^j) - \boldsymbol{c}_{\hat{y}_j}||^2, \end{aligned} \quad (3)$$

where $\boldsymbol{c}_{y_i}$ represents the y_ith source class center of the deep feature, and y_i represents the ground-truth label of source sample x_s^i. Similarly, $\boldsymbol{c}_{\hat{y}_j}$ denotes the $\hat{y}_j$th target class center of the deep feature, and $\hat{y}_j$ denotes the pseudo-label of target sample x_t^j according to the MLP classifier. The class centers are defined as the mean feature of each class and updated based on mini-batch with a small learning rate α during the training process. Specifically, the update of kth class center $\boldsymbol{c}_k$ is given by

$$\Delta \boldsymbol{c}_k = \frac{\sum_{i=1}^{\mathcal{B}} \delta(y_i = k) \cdot (\boldsymbol{c}_k - F(x_i))}{1 + \sum_{i=1}^{\mathcal{B}} \delta(y_i = k)}, \quad (4)$$

$$\boldsymbol{c}_k^{\mathbf{t}} \leftarrow \boldsymbol{c}_k^{\mathbf{t}-1} + \alpha \Delta \boldsymbol{c}_k, \quad (5)$$

where $\mathcal{B}$ represents the batch size of mini-batch, $\mathbf{t}$ denotes the training iteration, and $\delta(y_i = k) = 1$ if x_i belongs to the kth class, otherwise it should be 0. α is fixed as 0.7 in our experiments.

We not only compact intra-class features for source domain and target domain separately, but also align the same class center across two domains to intensify the cross-domain feature alignment. Namely,

$$\mathcal{L}_{calign} = \sum_{k=1}^{K} ||\boldsymbol{c}_k^s - \boldsymbol{c}_k^t||^2, \quad (6)$$

where K is the number of classes. Therefore, the discriminative learning loss consists of center loss and center-alignment loss weighted by λ_{center} and λ_{calign} separately:

$$\mathcal{L}_{dis} = \lambda_{center}\mathcal{L}_{center} + \lambda_{calign}\mathcal{L}_{calign}. \quad (7)$$

The discriminative learning can be employed in MTDA by conducting center loss for each target domain and aligning centers for each pair of source-target domains. λ_{center} and λ_{calign} are set as 0.001 and 0.1.

3) Self-train: Discriminative learning loss would improve the compactness of intra-class features, thus making different classes are separable from each other in the deep feature space. To leverage the discriminative deep feature, a nearest neighbour based classifier is proposed to provide pseudo-labels of unlabeled target data to self-train the feature extractor and the MLP classifier. Self-training is to train the model again with its own predictions especially for unlabeled data. Since some of the model predictions are correct, the self-training can improve the performance but the improvement is also limited by the part of false predictions.

Here we do not use the predictions from the MLP classifier to self-train the model, but adopt the predictions from the nearest neighbour classifier to provide pseudo-labels for self-training. Two possible advantages are considered, one is to leverage the discriminative deep feature to provide more robust pseudo-labels compared to the MLP classifier, and the other is to provide complementary information from the feature distance based predictions to optimize the decision boundary instead of using the predictions of MLP classifier to train it again. With the pseudo-labels $\check{\mathbf{y}}$ from the nearest neighbour classifier, the self-training loss for target data is defined as the cross-entropy loss

$$\mathcal{L}_{tce} = -\frac{1}{n_t} \sum_{j=1}^{n_t} \check{\mathbf{y}}_j \log(\mathbf{p}_{x_t^j}), \quad (8)$$

where the pseudo-label $\check{\mathbf{y}}_j$ represents the one-hot vector, and $\mathbf{p}_{x_t^j}$ represents the softmax probability from MLP classifier for target sample x_t^j.

As described above, the MLP classifier provides predictions of unlabeled target data as their pseudo-labels to conduct discriminative learning, and in turn based on the learned discriminative features, the nearest neighbour classifier provides its predictions as the pseudo-labels for self-training. These two strategies, learning discriminative features and self-training, support each other to constitute the mutual learning.

4) Total loss: In addition, we add Tsallis entropy minimization loss proposed by [30] into the proposed DML as a regularization term to minize the uncertainty of target pseudo-labels:

$$\mathcal{L}_{ent} = \frac{1}{\beta - 1}(1 - \sum_{j=1}^{n_t} \mathbf{p}_{x_t^j}^{\beta}), \tag{9}$$

where β is set as 1.9 in our experiments.

Overall, the loss consists of discriminative loss $\mathcal{L}_{dis}$, self-training loss $\mathcal{L}_{tce}$, Tsallis entropy minimization loss $\mathcal{L}_{ent}$ and the baseline losses including the standard cross entropy loss $\mathcal{L}_{ce}$ for the supervised classification training with source data and the adversarial training loss $\mathcal{L}_{adv}$:

$$\mathcal{L} = \mathcal{L}_{dis} + \lambda_{tce}\mathcal{L}_{tce} + L_{ce} + L_{adv} + \lambda_{ent}\mathcal{L}_{ent}, \tag{10}$$

where λ_{tce} is set as 0.01 in both BTDA and MTDA, λ_{ent} is set as 0.001 in BTDA and 0.02 in MTDA.

C. Discriminative Curriculum Mutual Learning

Compared with the BTDA, additional information about domain label is included in MTDA. Although the proposed DML is also suitable for MTDA by small modification on discriminative learning as described before, we further design the DCML specific for MTDA to transfer the model to the target domains one by one.

Among the different sequences of transferring, easy-to-hard is recommended by curriculum learning [31], [18], [26]. Specifically, the training process of discriminative curriculum mutual learning (DCML) consists of three steps: pre-train the model with a labeled source domain, adapt the model to the target domains from easy to difficult, and fine-tune the model. The whole process of DCML is illustrated in Fig. 2 taking two target domains for example.

At the second adaptation step, the target domain with the lowest entropy is selected as the easy domain on which to conduct the one-to-one adaptation with DML. Note that before conducting DML, the parameters of the whole network are randomly re-initialized. After single target domain adaptation, the samples with high confidence over threshold τ in the adapted target domain are selected into the source domain labeled by the predictions from MLP classifier. τ is set as 0.7 in our experiments.

In summary, the curriculum adaptation contains three stages: target domain selection, adaptation and sample selection. These three stages are repeated until one-to-one adaptation is finished for all target domains. Finally, the model is fine-tuned until convergence with the pseudo-source domain with both labeled source data and pseudo-labeled target data from all target domains.

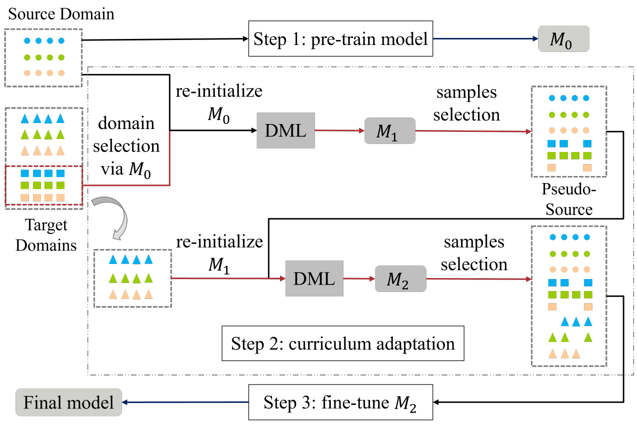


Fig. 2: (Best viewed in color) The discriminative curriculum mutual learning for MTDA (take two target domains for example). DCML contains three steps: pre-training, curriculum adaptation and fine-tuning where the curriculum adaptation includes three stages: target domain selection, adaptation via DML and target sample selection.

IV. Experiments

In this section, extensive experiments and analysis are carried out to validate the effectiveness of the proposed DML and DCML. DML is employed for both BTDA and MTDA, while DCML is only for MTDA. We first introduce three widely used domain adaptation benchmarks, next clarify the implementation details of our experimental settings and present compared state-of-the-art methods for multi-target domain adaptation, then explain our results on both BTDA and MTDA, and finally analyse our method and results from several perspectives.

A. Datasets

Office-31 [35] is a popular dataset for visual domain adaptation containing 4110 images under 31 categories in the office environment collected from three distinct domains: Amazon (A, images downloaded from online merchants), Webcam (W, low-resolution images captured using a web camera), and DSLR (D, high-resolution images captured using a digital SLR camera). **Office-Home** [36] is a bigger and more complicated dataset released recently containing 15588 images under 65 categories that are common in daily life. This dataset is collected from four different domains: Artistic images (Ar), Clipart (Cl), Product images (Pr) and Real-World images (Rw). **DomainNet** [37] is a big benchmark for domain adaptation containing approximately 0.6 million images under 345 categories from six domains: Clipart (C), Infograph (I), Painting (P), Quickdraw (Q), Real (R, photos and real world images) and Sketch (S).

B. Implementation Details

We follow the same evaluation protocols as AMEAN [16] or MT-MTDA [19] where each domain is as the labeled source domain and the rest constitute the unlabeled blending target domain for BTDA or multiple target domains for MTDA. The adaptation task is recorded as the name of the *source* domain in the result tables. Both labeled source data and unlabeled target data are used for training a robust model, and the accuracies on multiple target domains are averaged to evaluate the model performance. Every adaptation task is repeated three times, and the average accuracy is reported.

We implement our experiments on the Pytorch [38] platform and adopt ResNet [33] pre-trained by ImageNet [1] as the

TABLE I: Comparison with state-of-the-art methods on Office-31 and Office-Home. BTDA: blending target domain adaptation, MTDA: multiple target domains adaptation. DML: discriminative mutual learning, DCML: discriminative curriculum mutual learning. The best results are highlighted in bold.

Setting	Model	Office-31				Office-Home				
		Amazon	Dslr	Webcam	Avg	Art	Clipart	Product	Real-World	Avg
w/o target	source only	68.6	70.0	66.5	68.4	47.6	42.6	44.2	51.3	46.4
BTDA	DAN [6]	78.0	64.4	66.7	69.7	55.6	56.6	48.5	56.7	54.4
	RTN [32]	84.3	67.5	64.8	72.2	53.9	56.7	47.3	51.6	52.4
	JAN [7]	84.2	74.4	72.0	76.9	58.3	60.5	52.2	57.5	57.1
	DANN [10]	78.2	72.2	69.8	73.4	58.4	58.1	52.9	62.1	57.9
	CDAN [22]	93.6	80.5	81.3	85.1	59.5	61.0	54.7	62.9	59.5
	AMEAN [16]	90.1	77.0	73.4	80.2	64.3	65.5	59.5	66.7	64.0
	CGCT [26]	93.9	85.1	85.6	88.2	**67.4**	68.1	61.6	68.7	66.5
	DML (ours)	**94.9**	**85.3**	**86.2**	**88.8**	67.0	**70.6**	**62.9**	**69.0**	**67.4**
MTDA	MT-MTDA [19]	87.9	83.7	84.0	85.2	64.6	66.4	59.2	67.1	64.3
	HGAN [25]	88.0	84.4	84.9	85.8	–	–	–	–	–
	CGCT [26]	93.4	86.0	87.1	88.8	70.5	71.6	66.0	71.2	69.8
	DML (ours)	**96.8**	85.3	86.1	**89.4**	67.4	70.2	64.2	70.0	68.0
	DCML (ours)	94.2	**86.5**	**87.4**	**89.4**	**72.1**	**74.2**	**69.3**	**72.2**	**72.0**

TABLE II: Comparison with state-of-the-art methods on DomainNet in the setting of BTDA with ResNet-101 as the backbone. The best results are highlighted in bold.

Model	C	I	P	Q	R	S	Avg
ResNet101 [33]	25.6	16.8	25.8	9.2	20.6	22.3	20.1
SE [34]	21.3	8.5	14.5	**13.8**	16.0	19.7	15.6
MCD [27]	25.1	19.1	27.0	10.4	20.2	22.5	20.7
DADA [20]	26.1	20.0	26.5	12.9	20.7	22.8	21.5
DML (ours)	**32.0**	**25.4**	**29.4**	12.7	**31.5**	**36.4**	**27.9**

TABLE III: Improvement of different baselines with *dml* in BTDA setting on Office-31.

Model	Amazon	Dslr	Webcam	Avg
DANN [10]	88.9	81.7	82.7	84.4
DANN + *dml*	**95.8**	**85.0**	**84.5**	**88.4**
MinEnt [42]	91.2	83.2	83.6	86.0
MinEnt + *dml*	**95.6**	**84.7**	**85.0**	**88.4**
AFN [43]	89.4	83.1	83.5	85.3
AFN + *dml*	**93.6**	**83.8**	**84.3**	**87.2**

network backbone same to other methods [26], [16] on the aforementioned benchmarks. ResNet-50 is used for Office-31 and Office-Home, and ResNet-101 is used for DomainNet. The discriminator consists of three fully connected (FC) layers with dropout [39] and ReLu activation [40] and the MLP classifier consists of two FC layers with batch normalization [41], ReLu activation and dropout. The discriminator and MLP classifier are trained from scratch with a learning rate 10 times that of the lower layers through back-propagation. Mini-batch SGD with momentum of 0.9 is adopted as the optimizer and the learning rate is initialized as 0.001. The total number of training iterations is set as 50000 to ensure the convergence of the model.

C. Comparison

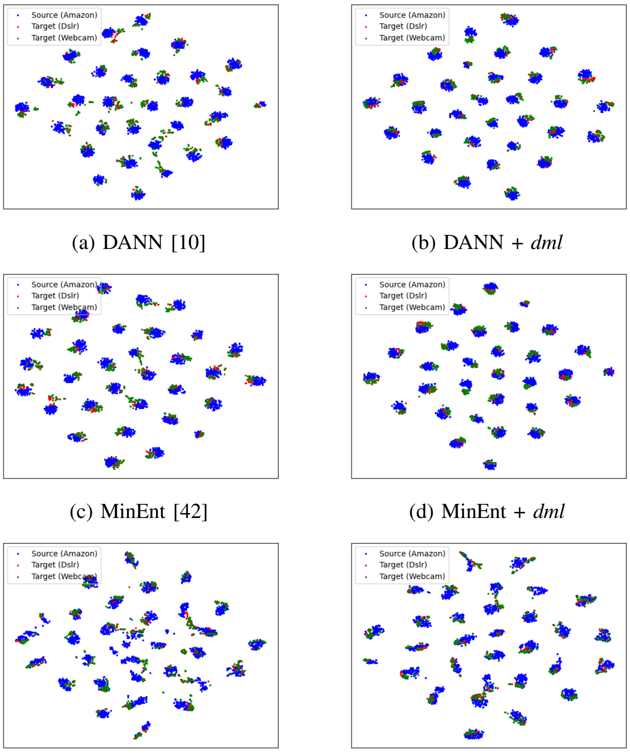
For BTDA, we compare our method with state-of-the-art domain adaptation methods, including: Deep Adaptation Network (DAN) [6], Residual Transfer Network (RTN) [32], Joint Adaptation Network (JAN) [7], Domain Adversarial Neural Network (DANN) [10], Conditional Domain Adversarial Network (CDAN) [22], Deep Adversarial Disentangled Autoencoder (DADA) [20], Adversarial MEta-Adaptation Network (AMEAN) [16] and Curriculum Graph Co-Teaching (CGCT) [26]. For MTDA, there are also competitive researches: Multi-Teacher (MT-MTDA) [19], Heterogeneous Graph Attention Network (HGAN) [25] and Curriculum Graph Co-Teaching (CGCT) [26].

D. Results

Experimental results compared with state-of-the-art methods for both BTDA and MTDA on Office-31 and Office-Home are presented in Table I. For BTDA, results show that the proposed DML outperforms all compared methods on both Office-31 and Office-Home with an improvement of 0.6% and 0.9% on average accuracy, respectively. Similarly for MTDA, DML shows superior performance with respect to average accuracy on Office-31 compared with other methods, and outperforms the competitive method MT-MTDA [19] published recently on Office-Home. The further designed DCML achieves the best performance on Office-Home in the setting of MTDA, surpassing the second-best method [26] by 2.2%.

In addition, the effectiveness of our method is also verified on a harder and bigger benchmark DomainNet which is so far the biggest benchmark for domain adaptation. The results of DML in the setting of BTDA are shown in Table II. The proposed DML outperforms other compared methods on average accuracy with a large margin, demonstrating an improvement of 6.4% on average accuracy.

The proposed discriminative learning ($\mathcal{L}_{dis}$) and mutual self-training ($\mathcal{L}_{tce}$) which are denoted as *dml* are flexible modules to be compatible with many baselines. Here we identify the effectiveness of *dml* on three baselines, including adversarial training method (DANN) [10], Minimal Entropy (MinEnt) [42] and Adaptive Feature Norm (AFN) [43]. All the



(a) DANN [10] (b) DANN + *dml*

(c) MinEnt [42] (d) MinEnt + *dml*

(e) AFN [43] (f) AFN + *dml*

Fig. 3: (Best viewed in color) The t-SNE visualization of deep features on Amazon $\rightarrow$ rest of Office-31 in the setting of BTDA. The proposed *dml* (right) improves the feature compactness and cross-domain alignment (blue: Source (Amazon), red: Target (Dslr), green: Target (Webcam)) compared with three baselines (left).

baselines are implemented in our experimental environment. The results averaged by three repeated experiments are shown in Table III in the BTDA setting on Office-31. The proposed *dml* improves the performance by 4.0%, 2.4% and 1.9% for DANN, MinEnt and AFN on average accuracy, respectively. These obvious and stable improvements validate the effectiveness of the proposed *dml* from another perspective.

E. Analysis

Feature visualization. The deep features from the trained BTDA model are visualized via t-SNE [44] in Fig. 3 with Amazon as the source domain and the remaining domains (Dslr and Webcam) mixed as the blending target domain on Office-31. Visualization illustrates the feature compactness and cross-domain alignment. On the basis of three baselines DANN [10], MinEnt [42] and AFN [43], the proposed *dml* improves the cross-domain feature alignment and renders more discriminative features.

Ablation study. We conduct ablation study on Office-31 to analyse the contribution of each module in the proposed DML. The experimental results are reported in Table IV to show the effects of discriminative learning $\mathcal{L}_{dis}$, self-training $\mathcal{L}_{tce}$, mutual learning (ML), adversarial training $\mathcal{L}_{adv}$, and the entropy minimization $\mathcal{L}_{ent}$. Each component contributes to the improvement of the performance where adding the self-training $\mathcal{L}_{tce}$ brings the most increase of 5.3% compared with discriminative learning $\mathcal{L}_{dis}$ only. Mutual learning also improves the performance by 1.3% ($\mathcal{L}_{dis}$ + $\mathcal{L}_{tce}$ + ML vs. $\mathcal{L}_{dis}$ + $\mathcal{L}_{tce}$) where the latter uses the predictions from the MLP classifier as the pseudo-labels for self-training.

TABLE IV: Ablation study of the proposed DML on Office-31 in the BTDA setting. ML: mutual learning.

Model	$\mathcal{L}_{dis}$	$\mathcal{L}_{tce}$	ML	$\mathcal{L}_{adv}$	$\mathcal{L}_{ent}$	Avg
source only	×	×	×	×	×	68.4
DML (ours)	✓	×	×	×	×	81.7
	✓	✓	×	×	×	87.0
	✓	✓	✓	×	×	88.3
	✓	✓	✓	✓	×	88.4
	✓	✓	✓	✓	✓	88.8

TABLE V: Comparison among one-to-one (o-o), one-to-blending (o-b) and one-to-multiple (o-m) domain adaptation under the proposed DML on Office-31.

	A→D	A→W	D→A	D→W	W→A	W→D	Avg
o-o	94.4	93.1	**73.6**	98.6	**75.6**	**100.0**	89.2
o-b	95.5	94.9	72.8	97.8	72.4	99.9	88.8
o-m	**96.9**	**96.8**	71.9	**98.7**	72.2	**100.0**	**89.4**

One-to-one vs. one-to-many. We have claimed that multi-target domain adaptation enables one model generalized to multiple target domains, thus saving the cost of training. Here we compare the performance of multiple one-to-one (single source single target) adaptation models and one unified model for one-to-many including a blending target domain (one-to-blending) and multiple target domains (one-to-multiple) in Table V. Results show that one-to-one (six models) behaves better than one-to-blending (three models), but one-to-multiple (three models) outperforms one-to-one which indicates the positive effect of more data. Considering the small difference in performance and the big saving in training models especially for a large number of target domains, the multi-target domain adaptation is more suitable for a complex and general applications.

V. Conclusion

In this paper, we propose a discriminative mutual learning (DML) for multi-target domain adaptation consisting of learning discriminative features and self-training with pseudo-labels from each other. The proposed DML improves the compactness of intra-class features which is important for adapting a model to a heterogeneous or blended target domain. With domain labels, curriculum learning is further introduced into DML to transfer from an easy domain to a difficult domain sequentially for multiple target domains adaptation. Extensive experiments demonstrate the superiority of the proposed DML in the settings of both BTDA and MTDA.

References

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *IJCV*, 2015.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *arXiv preprint arXiv:1506.02640*, 2015.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *CVPR*, 2014.

[4] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *CVPR*, 2011.

[5] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Transactions on Neural Networks*, vol. 22, no. 2, pp. 199–210, 2011.

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

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

[8] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *ECCV Workshops*, 2016.

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

[10] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. S. Lempitsky, "Domain-adversarial training of neural networks," *JMLR*, vol. 17, p. 2096–2030, 2016.

[11] Z. Pei, Z. Cao, M. Long, and J. Wang, "Multi-adversarial domain adaptation," in *AAAI*, 2018.

[12] S. Xie, Z. Zheng, L. Chen, and C. Chen, "Learning semantic representations for unsupervised domain adaptation," in *ICML*, 2018.

[13] C. Chen, W. Xie, W. Huang, Y. Rong, X. Ding, Y. Huang, T. Xu, and J. Huang, "Progressive feature alignment for unsupervised domain adaptation," in *CVPR*, 2019.

[14] Y. Zou, Z. Yu, B. V. K. V. Kumar, and J. Wang, "Unsupervised domain adaptation for semantic segmentation via class-balanced self-training," in *ECCV*, 2018.

[15] M. Chen, S. Zhao, H. Liu, and D. Cai, "Adversarial-learned loss for domain adaptation," in *AAAI*, 2020.

[16] Z. Chen, J. Zhuang, X. Liang, and L. Lin, "Blending-target domain adaptation by adversarial meta-adaptation networks," in *CVPR*, 2019.

[17] B. Gholami, P. Sahu, O. Rudovic, K. Bousmalis, and V. Pavlovic, "Unsupervised multi-target domain adaptation: An information theoretic approach," *IEEE Transactions on Image Processing*, vol. PP, no. 99, pp. 1–1, 2020.

[18] Z. Liu, Z. Miao, X. Pan, X. Zhan, D. Lin, S. X. Yu, and B. Gong, "Open compound domain adaptation," in *CVPR*, 2020.

[19] N. M. Le, A. Belal, M. Kiran, J. Dolz, L. A. Blais-Morin, and E. Granger, "Unsupervised multi-target domain adaptation through knowledge distillation," in *WACV*, 2021.

[20] X. Peng, Z. Huang, X. Sun, and K. Saenko, "Domain agnostic learning with disentangled representations," in *ICML*, 2019.

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

[22] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *NeurIPS*, 2018, pp. 1645–1655.

[23] Y. Kim and C. Kim, "Semi-supervised domain adaptation via selective pseudo labeling and progressive self-training," in *ICPR*, 2021.

[24] X. Liu, B. Hu, X. Liu, J. Lu, J. You, and L. Kong, "Energy-constrained self-training for unsupervised domain adaptation," in *ICPR*, 2021.

[25] X. Yang, C. Deng, T. Liu, and D. Tao, "Heterogeneous graph attention network for unsupervised multiple-target domain adaptation," *IEEE TPAMI*, 2020.

[26] S. Roy, E. Krivosheev, Z. Zhong, N. Sebe, and E. Ricci, "Curriculum graph co-teaching for multi-target domain adaptation," in *CVPR*, 2021.

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

[28] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," in *NeurIPS*, 2018.

[29] Y. Wen, K. Zhang, Z. Li, and Q. Yu, "A discriminative feature learning approach for deep face recognition," in *ECCV*, 2016.

[30] H. Liu, J. Wang, and M. Long, "Cycle self-training for domain adaptation," in *NeurIPS*, 2021.

[31] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *ICML*, 2009.

[32] M. Long, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *NeurIPS*, 2016.

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

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

[35] K. Saenko and B. Kulis, "Adapting visual category models to new domains," in *ECCV*, 2010.

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

[37] X. Peng, Q. Bai, X. Xia, Z. Huang, and B. Wang, "Moment matching for multi-source domain adaptation," in *ICCV*, 2019.

[38] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. Devito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NeurIPS-Workshops*, 2017.

[39] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *Computer Science*, vol. 3, no. 4, pp. 212–223, 2012.

[40] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," *Journal of Machine Learning Research*, pp. 315–323, 2011.

[41] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015.

[42] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," in *NeurIPS*, 2005.

[43] R. Xu, G. Li, J. Yang, and L. Lin, "Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation," in *ICCV*, 2020.

[44] V. D. M. Laurens and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. 2605, pp. 2579–2605, 2008.