Entropy Minimization Versus Diversity Maximization for Domain Adaptation

Xiaofu Wu, *Member, IEEE*, Suofei Zhang, Quan Zhou, *Member, IEEE*, Zhen Yang, *Senior Member, IEEE*, Chunming Zhao, *Member, IEEE*, and Longin Jan Latecki

***Abstract*— Entropy minimization has been widely used in unsupervised domain adaptation (UDA). However, existing works reveal that the use of entropy-minimization-only may lead to collapsed trivial solutions for UDA. In this article, we try to seek possible close-to-ideal UDA solutions by focusing on some intuitive properties of the ideal domain adaptation solution. In particular, we propose to introduce diversity maximization for further regulating entropy minimization. In order to achieve the possible minimum target risk for UDA, we show that diversity maximization should be elaborately balanced with entropy minimization, the degree of which can be finely controlled with the use of deep embedded validation in an unsupervised manner. The proposed minimal-entropy diversity maximization (MEDM) can be directly implemented by stochastic gradient descent without the use of adversarial learning. Empirical evidence demonstrates that MEDM outperforms the state-of-the-art methods on four popular domain adaptation datasets.**

***Index Terms*— Domain adaptation, entropy minimization, image classification, transfer learning, Visual Domain Adaptation (VisDA) challenge.**

I. Introduction

THE recent success of deep learning depends heavily on the large-scale fully labeled datasets and the development of easily trainable deep neural architectures under the backpropagation algorithm, such as convolutional neural networks (CNNs) and their variants [1], [2]. In practical applications, a new target task and its dataset (target domain) may be similar to a known source task and its fully labeled dataset (source domain). However, the difference between the source and target domains is often not negligible, which makes the previously trained model not work well for the new task. This is known as domain shift [3]. As the cost of massive labeling is often expensive, it is very attractive for the target task to exploit any existing fully labeled source dataset and adapt the trained model to the target domain [4]–[10].

This domain adaptation approach is aiming to learn a discriminative classifier in the presence of domain shift [11], [12]. It can be achieved by optimizing the feature representation to minimize some measures of domain shift, typically defined as the distance between the source and target domain distributions or its degraded form, such as maximum mean discrepancy (MMD) [13], [14] or correlation distance [15].

With the invention of generative adversarial networks [16], various adversarial methods have been proposed for the purpose of unsupervised domain adaptation (UDA) [12], [17]–[19], where the domain discrepancy distance is minimized through an adversarial objective with respect to a binary domain discriminator. The domain-invariant features could be extracted whenever this binary domain discriminator cannot distinguish between source and target samples [12], [17].

In recent years, there is also a broad class of domain adaptation methods, which employs entropy minimization as a proxy for mitigating the harmful effects of domain shift. The entropy minimization is performed on the target domain, which may take explicit forms [20]–[24] or implicit forms [17], [25]. Without any further regularization, it may produce trivial solutions with insufficient prediction diversity [26], [27], where unlabeled target samples are prone to be pushed into majority categories.

Often, UDA faces the challenging problem of hyperparameter selection, where the best configuration should be determined without resort to labels in the target dataset. Fortunately, deep embedded validation (DEV) [28] tailored to UDA was recently proposed to solve this difficulty, which embeds adapted feature representation in the validation procedure to yield unbiased estimation of the target risk.

In this article, we make contributions toward close-to-ideal domain adaptation with entropy minimization.

1) We propose a minimal-entropy diversity maximization (MEDM) method for UDA. Instead of minimizing the cross-domain discrepancy, MEDM tries to find a close-to-ideal domain adaptation solution[1] by balancing between entropy minimization and prediction-diversity

Manuscript received 2 September 2020; revised 1 February 2021, 11 June 2021, and 17 August 2021; accepted 31 August 2021. Date of publication 14 September 2021; date of current version 2 June 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 61876093 and Grant 62071242. *(Corresponding author: Xiaofu Wu.)*

Xiaofu Wu, Quan Zhou, and Zhen Yang are with the National Engineering Research Center of Communications and Networking, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (e-mail: xfuwu@ieee.org; quan.zhou@njupt.edu.cn; yangz@njupt.edu.cn).

Suofei Zhang is with the School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (e-mail: zhangsuofei@njupt.edu.cn).

Chunming Zhao is with the National Mobile Communication Research Laboratory, Southeast University, Nanjing 210096, China (e-mail: cmzhao@seu.edu.cn).

Longin Jan Latecki is with the Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122 USA (e-mail: latecki@temple.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2021.3110109.

Digital Object Identifier 10.1109/TNNLS.2021.3110109

[1]For the ideal domain adaptation, we mean that it can achieve the minimum value of the sum of two risks, namely, the source risk and the target risk.

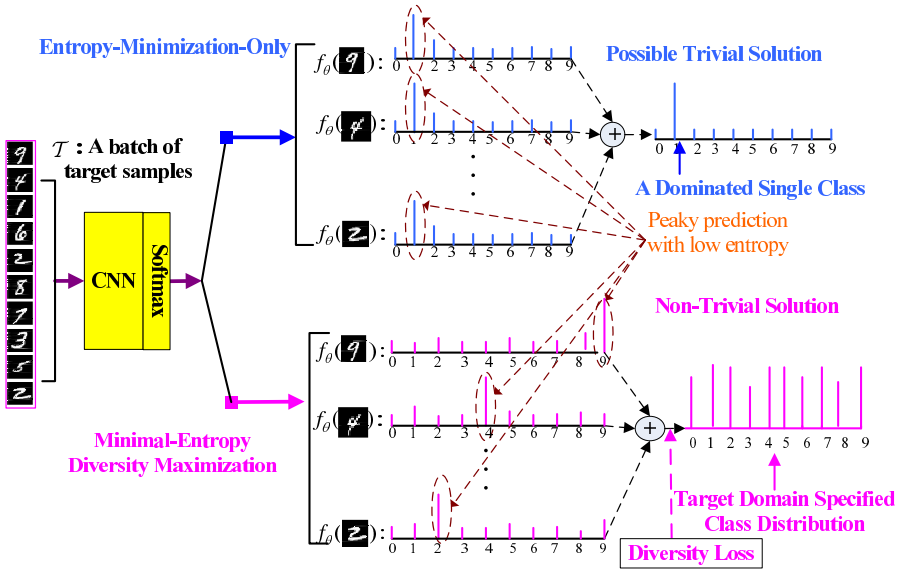


Fig. 1. Comparison of EMO and MEDM: the use of entropy minimization often induces peaky prediction with low entropy, and EMO may result in trivial solutions with a single-type peaky prediction. MEDM tries to maximize the batchwise diversity of the predicted categories, which could push entropy minimization away from trivial solutions.

maximization. The novel use of diversity maximization is shown to be very simple but effective for pushing entropy minimization toward better UDA solutions.

2) We provide both theoretical and experimental results to analyze the domain adaptation solution under the framework of MEDM.
3) Extensive experiments show that MEDM outperforms state-of-the-art methods on four domain adaptation datasets, including Visual Domain Adaption (VisDA)-2017, ImageCLEF, Office-Home, and Office-31. In particular, it boosts a significant accuracy margin on the largest domain adaptation dataset, the VisDA-2017 classification challenge.[2]

The rest of this article is organized as follows. A simple domain adaptation method with entropy-minimization-only (EMO) is introduced, and its insufficiency is discussed in Section II. In Section III, we propose a novel minimal-entropy diversity-maximization method, and its theoretical justification is also given. Simulation results are described in Section IV. Related works are briefly discussed in Section V. Finally, Section VII concludes this article.

II. Insufficiency of Entropy-Minimization-Only

A. Entropy-Minimization-Only

Consider the problem of classifying an image x in a K-classes problem. For UDA, we are given a source domain $\mathcal{D}_s = \{(x_i^s; y_i^s)\}_{i=1}^{n_s}$ of n_s labeled examples and a target domain $\mathcal{D}_t = \{x_j^t\}_{j=1}^{n_t}$ of n_t unlabeled examples. The source domain and the target domain are sampled from joint distributions $P(x^s; y^s)$ and $Q(x^t; y^t)$, respectively, while the independent identically distributed (i.i.d.) assumption is often violated as $P \neq Q$. Hence, the problem is to exploit a bunch of labeled images in $\mathcal{D}_s$ for training a statistical classifier that, during inference, provides probabilities of a given test image $x_t \in \mathcal{D}_t$ belonging to each of the K classes. In this article, we focus on a deep neural-network-based classifier $y = f_\theta(x)$ (in general, the classifier f_θ depends upon a collection of parameters θ), which provides probabilities of x belonging to each class as

$$f_\theta(x) = [\mathbb{P}(y = 1|x), \ldots, \mathbb{P}(y = K|x)]. \tag{1}$$

The goal is to design the classifier $y = f_\theta(x)$ such that the target risk $\epsilon_t(f_\theta) = \mathbb{E}_{(x^t;y^t)\sim Q}[f_\theta(x^t) \neq y^t]$ can be minimized. Since the target risk cannot be computed in the scenario of UDA, the domain adaptation theory [29], [30] suggests to bound the target risk with the sum of the cross-domain discrepancy $D(P; Q)$ and the source risk $\epsilon_s(f_\theta) = \mathbb{E}_{(x^s;y^s)\sim P}[f_\theta(x^s) \neq y^s]$. By jointly minimizing the source risk and the cross-domain discrepancy $D(P; Q)$, various domain adaptation methods were extensively proposed, which differs mainly in the choice of $D(P; Q)$.

For supervised learning on the source domain, the classifier is trained to minimize the standard supervised loss for any given batch of $\mathcal{S} \subset \mathcal{D}_s$

$$\mathcal{L}_s(\theta, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{(x,y)\in\mathcal{S}} \ell(y, f_\theta(x)) \tag{2}$$

with

$$\ell(y, \hat{y}) = \langle y, \hat{y} \rangle = -\sum_{j=1}^{K} y_j \log \hat{y}_j \tag{3}$$

and $|\mathcal{S}|$ denotes the cardinality of the set $\mathcal{S}$.

Recently, label smoothing (LS) was proposed in [31] to refine the cross-entropy loss with noisy labels, which may improve the accuracy of a multiclass neural network. With LS,

[2]Source code is available at https://github.com/AI-NERC-NUPT/MEDM

$\ell(y, \hat{y})$ in (2) is replaced by the smoothed version of

$$\ell^{LS}(y, \hat{y}) = -\sum_{j=1}^{K}\big((1-\alpha)y_j + \alpha/K\big)\log \hat{y}_j \tag{4}$$

where α denotes the smoothing parameter, which is empirically set to 0.1. In [32], the power of LS from the view of optimization was investigated, and it was shown that an appropriate LS can help to speed up the convergence by reducing the variance and improve the generalization accuracy. Furthermore, LS encourages the representations of training examples from the same class to the group in tight clusters [33], which may be more advantageous for the task with a large number of classes, for example, the transferring task over Office-Home.

To adapt to the unlabeled target domain, a large class of domain adaptation methods also minimizes the entropy loss

$$\mathcal{L}_e(\theta, \mathcal{T}) = -\frac{1}{|\mathcal{T}|}\sum_{x_t \in \mathcal{T}}\langle f_\theta(x_t), \log f_\theta(x_t)\rangle \tag{5}$$

for any given batch of $\mathcal{T} \subset \mathcal{D}_t$ as an efficient regularization technique. Therefore, with a batchwise training approach, EMO is to solve the following problem:

$$\min_\theta E_{\mathcal{S},\mathcal{T}}[\mathcal{L}_s(\theta, \mathcal{S}) + \lambda \mathcal{L}_e(\theta, \mathcal{T})] \tag{6}$$

where $E[\cdot]$ denotes the expectation.

The EMO presented in (6) was first proposed in [35] for semisupervised learning, where a decision rule is to be learned from labeled and unlabeled data, and EMO (6) enables to incorporate unlabeled data in the standard supervised learning. *For the scenario of UDA considered in this article, the difference is that unlabeled samples are sampled from the target-domain distribution Q, which may differ considerably from the source-domain distribution P.*

B. Peaky Prediction, Cross-Domain Discrepancy, and Insufficiency of EMO

The rationality of EMO for the purpose of UDA can be illustrated in Fig. 1, where the use of entropy minimization induces peaky predications with low entropy for unlabeled target samples. By enforcing the supervised loss over source samples, peaky predications over both source and target samples can be well expected, which happens to be the case for the ideal UDA classifier f^* (performs well for both domains), namely,

$$f^* = \arg\min_{f \in \mathcal{H}}[\epsilon_s(f) + \epsilon_t(f)] \tag{7}$$

where $\mathcal{H}$ is the space of hypothesis classifiers. Indeed, the fully supervised classifier f_e^* minimizing (7) (assuming labeling is accessible for both source and target domains) always leads to peaky predictions with low entropy for f_e^* for well-defined UDA tasks, as observed in experiments.

One of the potential problems in EMO is that the explicit use of entropy minimization (over target samples) may lead to trivial solutions, where a single class may dominate [26], as demonstrated in the Appendix. In practice, this problem can be partially alleviated by the use of small λ in (6).

TABLE I
ACCURACY (%) ON OFFICE-31 WITH RESNET-50

Method	A → W	W → A
DAN [23]	83.8	62.7
RevGrad [17]	82.0	67.4
MADA [34]	90.0	66.4
EMO ($\lambda = 0.1$) (6)	91.0	68.1

As shown in Table I, EMO ($\lambda = 0.1$) performs not bad over two typical transferring tasks on Office-31, including $A \to W$ and $W \to A$. Note that EMO does not explicitly minimize any cross-domain discrepancy $D(P; Q)$. When the training goes on, the target entropy loss decreases, and the test accuracy increases steadily, as observed in experiments.

We also investigate the effect of the inclusion of MMD loss in EMO. The resulting algorithm, EMO + MMD, seeks to solve the following problem:

$$\min_\theta E_{\mathcal{S},\mathcal{T}}\big[\mathcal{L}_s(\theta, \mathcal{S}) + \lambda \mathcal{L}_e(\theta, \mathcal{T}) + \gamma\, d_k^2(P, Q)\big], \quad \lambda, \gamma > 0 \tag{8}$$

where $d_k^2(P, Q)$ denotes the multiple-kernel MMD loss [13]. With $\lambda = 0.1$ and $\gamma = 1.0$ in (8), the evolution of MMD loss and test accuracy for both EMO and EMO + MMD can be plotted in Figs. 2 and 3, respectively. As shown, the inclusion of MMD loss could lead to improved test accuracy, which may saturate soon as the training iteration goes on. Without an explicit inclusion of the MMD loss, the decrease in the MMD loss can also be observed during training, as indicated in Fig. 2; especially at the beginning of the training, the reduction in its absolute value, however, is not very significant.

For EMO + MMD, experiments show that the MMD loss computed at some intermediate layers could lead to little performance improvement but some unpredicted fluctuations on the test accuracy. *It should be emphasized that the entropy loss focuses on the class predictions, while the MMD loss focuses on the features*. For an end-to-end training approach, we believe that the use of the final class predictions is more beneficial than the intermediate features for the purpose of seeking UDA classifiers [19], [36]. Although the simple inclusion of MMD loss in EMO could lead to improved performance, the performance improvement is still limited compared to the state-of-the-art UDA algorithms. Therefore, it is interesting to explore the entropy minimization further to seek more powerful UDA classifiers.

Note that the minimization of the target risk $\epsilon_t(f_\theta)$ could push the network prediction $f_\theta(x_t)$ toward the true solution $y_t = [y_1, \ldots, y_K]$ with $y_k \in \{0, 1\}$, $\sum_k y_k = 1$, namely, $f_\theta(x_t) \to [0, \ldots, 1, \ldots, 0]$, which results into the minimum value of entropy (zero). This means that entropy minimization is a necessary condition for the minimization of the target risk $\epsilon_t(f_\theta)$. Hence, *entropy minimization may be more direct and simpler for end-to-end training of θ in order to minimize the target risk, compared to the use of more complicated cross-domain discrepancy*. Unfortunately, as a necessary but not sufficient condition for minimization of the target risk

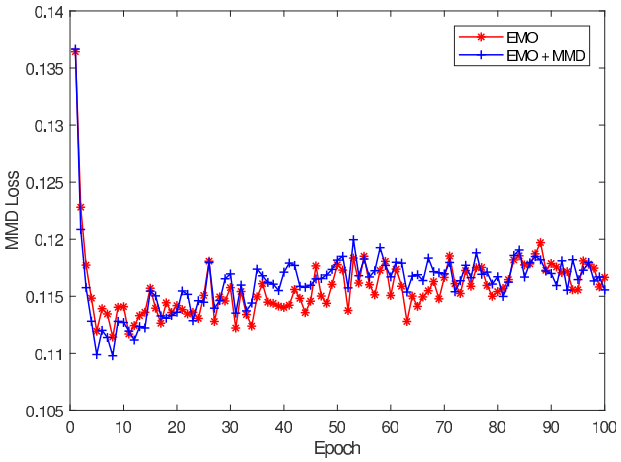


Fig. 2. Evolution of MMD loss for EMO ($\lambda = 0.1$) and EMO + MMD ($\lambda = 0.1$ and $\gamma = 1.0$) during training.

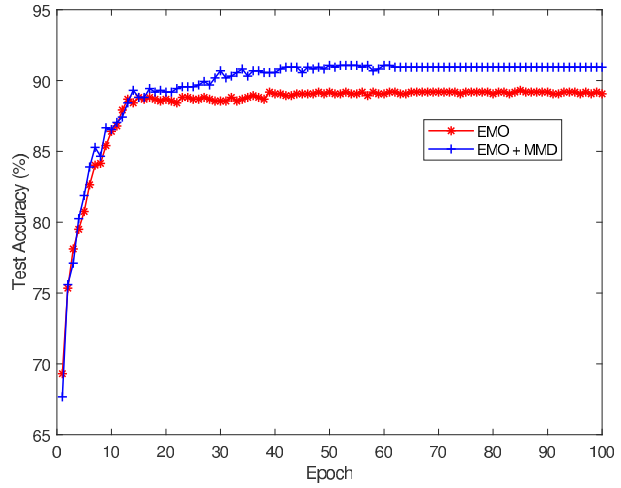


Fig. 3. Evolution of test accuracy for EMO ($\lambda = 0.1$) and EMO + MMD ($\lambda = 0.1$ and $\gamma = 1.0$) during training.

$\epsilon_t(f_\theta)$ [26], this simple technique may result into trivial solutions.

Problem 1: As entropy minimization is necessary but not sufficient for minimization of the target risk, it is natural to ask if we can pose some further regularization to push the optimizer to find the global minima instead of the trivial local minima.

III. Minimal-Entropy Diversity Maximization

The use of EMO may produce trivial solutions, as shown in Fig. 1. By noting that a trivial solution shown in Fig. 1 often has just one category, a nontrivial domain adaptation method may resort to producing sufficient category diversity in its solution.

A. Batchwise Diversity

Consider that two batches of input samples $(\mathcal{S}, \mathcal{T})$ are input to the network during training, where $\mathcal{S} \subset \mathcal{D}_s, \mathcal{T} \subset \mathcal{D}_t$ with the same batch size of $B = |\mathcal{T}| = |\mathcal{S}|$. With each unlabeled image $x_t \in \mathcal{T}$ as input, the output softmax predictions over the network f_θ can be written as $f_\theta(x_t)$. Then, one can compute the predicted category distribution over $\mathcal{T}$ as

$$\hat{\mathbf{q}}(\mathcal{T}) = \frac{1}{B}\sum_{x_t \in \mathcal{T}} f_\theta(x_t) \triangleq [\hat{q}_1, \hat{q}_2, \ldots, \hat{q}_K] \tag{9}$$

where

$$\hat{q}_k = \frac{1}{B}\sum_{x_t \in \mathcal{T}} \mathbb{P}(y_t = k | x_t)$$

and $\sum_{k=1}^{K} \hat{q}_k = 1$. *Note that* $\hat{\mathbf{q}}(\mathcal{T})$ *is computed over* $\mathcal{T}$*, which is dynamically changed during the batch-based training.*

For measuring the category diversity in a given target batch $\mathcal{T}$, we define the batchwise diversity $\mathcal{L}_d(\theta, \mathcal{T})$ as the entropy of $\hat{\mathbf{q}}(\mathcal{T}) = [\hat{q}_1, \hat{q}_2, \ldots, \hat{q}_K]$ (9). Formally, this category diversity over $\mathcal{T}$ can be calculated as

$$\mathcal{L}_d(\theta, \mathcal{T}) \triangleq H(\hat{\mathbf{q}}(\mathcal{T})) = -\sum_{k=1}^{K} \hat{q}_k \log \hat{q}_k. \tag{10}$$

As this diversity metric does not require any prior information about the true category distribution $\mathbf{q}$ over $\mathcal{D}_t$, its computation is easy to implement in practice. Note that random shuffling should be employed in training for approximating the ground-truth value of $H(\mathbf{q})$ over the target domain.

B. Entropy-Minimization Versus Diversity-Maximization

The objective of MEDM is to

$$\min_\theta E_{\mathcal{S},\mathcal{T}}[\mathcal{L}_s(\theta, \mathcal{S}) + \lambda \mathcal{L}_e(\theta, \mathcal{T}) - \beta \mathcal{L}_d(\theta, \mathcal{T})] \tag{11}$$

where $\lambda, \beta \geq 0$ are weighting factors. Given λ and β, this involves the optimization of θ for the minimization of a single total loss (11), which can be directly implemented by stochastic gradient descent without use of adversarial learning.

Note that (11) includes a diversity maximization term, which may prevent the entropy term $\mathcal{L}_e(\theta, \mathcal{T})$ from converging to sufficiently small value. In what follows, we show that this can be well alleviated by the selection of λ and β for finely controlling the tradeoff between entropy-minimization and diversity-maximization.

As shown in (11), our proposed MEDM may encourage to make predictions evenly across the batch, which, however, does not necessarily produce the evenly distributed categories. Let $\mathbf{q} = [q_1, \ldots, q_K]$ be the true category distribution of the target dataset, where q_k denotes the proportion of samples of the kth class among all target samples.

Theorem 1: Consider the EMO method ($\beta = 0$) in (11). If there exists a solution θ^* of (11) with $\mathcal{L}_e(\theta^*, \mathcal{T}) = 0$, we have that

$$E_{\mathcal{T}}\left[\mathcal{L}_d(\theta^*, \mathcal{T})\right] = H(\mathbf{q}^*)$$

where $\mathbf{q}^* = [q_1^*, \ldots, q_K^*]$ is the inferred category distribution of the target dataset when inferring over the network θ^*.

Proof: In the case of $\beta = 0$, diversity maximization is not included in (11). Note that

$$\mathcal{L}_e(\theta^*, \mathcal{T}) = 0.$$

Since the entropy is always nonnegative, we have that

$$-\langle f_\theta(x_t), \log f_\theta(x_t)\rangle = 0 \quad \forall x_t \in \mathcal{T}.$$

Hence, the network prediction $f_\theta(x_t)$ for any $x_t \in \mathcal{T}$ should present a peaky form, namely, $f_\theta(x_t) \to [0, \ldots, 1, \ldots, 0]$.

Given a random batch of samples $\mathcal{T}$ inputting to the network θ^*, the expectation of $\hat{q}_k(\mathcal{T})$ (9) can be computed as

$$E_{\mathcal{T}}\{\hat{q}_k(\mathcal{T})\} = \frac{1}{|\mathcal{T}|} E_{\mathcal{T}}\left\{\sum_{x_t \in \mathcal{T}} f^k_{\theta^*}(x_t)\right\} = \frac{1}{|\mathcal{T}|} \cdot \left(q^*_k|\mathcal{T}|\right) = q^*_k.$$

Therefore,

$$E_{\mathcal{T}}\left\{\mathcal{L}_d\left(\theta^*, \mathcal{T}\right)\right\} = E_{\mathcal{T}}\{H(\hat{\mathbf{q}})\} = H\left(\mathbf{q}^*\right).$$

■

Without the use of diversity maximization, EMO often results in trivial solutions, namely, $\max q^*_k \gg 1 - \max_k q^*_k$, where the predicted single-class samples may dominate among others. With the use of diversity maximization, it may encourage to make prediction evenly across the batch since the maximum value of $\mathcal{L}_d(\theta^*, \mathcal{T})$ could be achieved whenever $\mathbf{q}^* = [1/K, \ldots, 1/K]$. In fact, there exists a tradeoff by adjusting the parameters λ, β as justified in what follows.

Lemma 1: Consider the EMO method ($\beta = 0$) in (11). If there exists a solution θ^* of (11) with $\mathcal{L}_e(\theta^*, \mathcal{T}) = 0$, we have that

$$E_{\mathcal{T}}\left[\mathcal{L}_d\left(\theta^*, \mathcal{T}\right)\right] \leq H(\mathbf{q})$$

where $\mathbf{q} = [q_1, \ldots, q_K]$ is the ground-truth category distribution of the target dataset.

Proof: Without loss of generality, we consider that the network $f_\theta(x)$ can be decomposed into two subnetworks, namely, $f_\theta(x) = C(F(x))$, where F denotes a feature extraction subnetwork and C denotes a classifier over the feature space $\mathcal{F}$. For the purpose of domain adaptation, it is often assumed that $P(y|F(x)) = Q(y|F(x))$.

Let $\mathbf{X}_s = \{x_s\}_{s=1}^{n_s}$ and $\mathbf{X}_t = \{x_t\}_{t=1}^{n_t}$. Then, whenever $x_t \in F^{-1}(F(\mathbf{X}_s) \cap F(\mathbf{X}_t))$, one can expect that the network can give a correct prediction with minimal entropy. While $x_t \in F^{-1}(F(\mathbf{X}_t) - F(\mathbf{X}_s))$ and $\mathcal{L}_e(\theta^*, \mathcal{T}) = 0$, the network ($\theta^*$) encourages to make prediction toward a single class since there are simply no other constraints to be enforced. This means that, when inferring over the network θ^*, the inferred category distribution $\mathbf{q}^* = [q^*_1, \ldots, q^*_K]$ over the target domain has to meet the constraint of $H(\mathbf{q}^*) \leq H(\mathbf{q})$, where $\mathbf{q}$ is the ground-truth category distribution of the target dataset. This completes the proof by using Theorem 1. ■

According to Lemma 1, MEDM works as follows. For a given domain-adaptation task, MEDM starts with $\beta = 0$, namely, EMO. It is straightforward to find the smallest possible λ^* for EMO in achieving nearly minimal entropy of $\mathcal{L}_e(\theta^*, \mathcal{T}) = 0$. Then, we begin to train MEDM by fixing λ^* and increases β from 0. The use of nonzero β in MEDM could encourage the network predications toward diverse categories, while the requisite of minimal entropy makes a peaky prediction, which should not deviate from the source domain due to the use of supervision loss over the source domain.

Let f_{θ^*} be the ideal domain-adaptation classifier, which minimizes the sum of source and target risks [30], namely, $f_{\theta^*} = \arg\min_{f_\theta \in \mathcal{H}} \epsilon_s(f_\theta) + \epsilon_t(f_\theta)$, with $\mathcal{H}$ denoting the space of classifiers. Therefore, when inferring target samples over f_{θ^*}, one can expect peaky predictions or *predictions with low entropies*. Hence, Theorem 1 may still hold in this case. By restricting $\mathcal{H}$ to be the solution space of (11) with $\lambda, \beta \geq 0$, we expect that the same conclusion holds. In fact, *extensive experiments show that an explicit inclusion of entropy minimization in (11) can easily drive the trained CNN model toward small predictive entropy, even coexisting with diversity maximization*. Therefore, we believe that the ideal domain-adaptation classifier under the framework of (11) may output predictions with low entropies, which means that Theorem 1 may still hold, as shown in the following conjecture.

Conjecture 1: Consider the perfect domain-adaptation solution of $f^{(\lambda^*,\beta^*)}_{\theta^*}$ under the framework of (11), namely,

$$f^{(\lambda^*,\beta^*)}_{\theta^*} = \arg\min_{\theta,\lambda,\beta} \epsilon_s\left(f^{(\lambda,\beta)}_\theta\right) + \epsilon_t\left(f^{(\lambda,\beta)}_\theta\right). \tag{12}$$

We have that

$$E_{\mathcal{T}}\left[\mathcal{L}_d\left(\theta^*, \mathcal{T}\right)\right] \approx H\left(\mathbf{q}^*\right)$$

where $\mathbf{q}^* = [q^*_1, \ldots, q^*_K]$ is the inferred category distribution over the target domain with the network θ^*.

Since the target risk for $f^{(\lambda^*,\beta^*)}_{\theta^*}$ is expected to be small, we have that $\mathbf{q}^* \to \mathbf{q}$ and $H(\mathbf{q}^*) \to H(\mathbf{q})$. With *random shuffling* for batch-based training, $\mathcal{L}_d(\theta^*, \mathcal{T}) \to H(\mathbf{q}^*)$ holds with a high probability whenever the training process for (11) under the setting of (λ^*, β^*) converges. This may partially support the reasonability of the use of (11).

Although we do not know the perfect domain-adaptation solution of $f^{(\lambda^*,\beta^*)}_{\theta^*}$, one can search over the space of (λ, β) and further resort to the validation technique [28]. The essential process can be stated as follows. When $\beta = 0$ in (11), the use of explicit entropy minimization under the end-to-end training could lead to a solution of θ^* with $\mathcal{L}_d(\theta^*, \mathcal{T}) \to 0$. When β increases from 0, we would expect that the category diversity (10) increases correspondingly, which can help to find a better solution with $\mathcal{L}_d(\theta^*, \mathcal{T}) \to H(\mathbf{q})$. The problem now is how to determine the best value of λ and β in (11) for finding a close-to-perfect domain adaptation solution. Fortunately, this was recently investigated in [28].

C. Model Selection via Deep Embedded Validation

For the proposed MEDM, the selection of hyperparameters (λ and β) is of great importance for the final performance. For UDA, the model selection should be decided without access to the labels in the target dataset. Fortunately, the recently proposed DEV [28] has been proven very efficient for model selection, which embeds adapted feature representation into the validation procedure to obtain an unbiased estimation of the target risk with bounded variance.

Consider that the feature extractor F is an end-to-end training solution of (11), it is closely connected to the parameters λ and β in (11), i.e., $F \triangleq F_{\lambda,\beta}$. Let $\lambda^L_1 = \{\lambda_i\}_{i=1}^L$ be a finite collection of λ_i, where $\lambda_1 < \lambda_2 < \cdots < \lambda_L$. Let $\beta^V_1 = \{\beta_j\}_{j=1}^V$ be a finite collection of β_j, where $\beta_1 < \beta_2 < \cdots < \beta_V$. Therefore, a model selection procedure for MEDM can be shown in Algorithm 1, which makes a full search of two

Algorithm 1 Model Selection Procedure in MEDM

Require: $\mathcal{D}_s = \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{val}}, \mathcal{D}_t$;
$\lambda_1^L = \{\lambda_i\}_{i=1}^L, \beta_1^V = \{\beta_j\}_{j=1}^V$

Training Initialization: $\beta \leftarrow 0$.
for $\lambda \leftarrow \lambda_1, \ldots, \lambda_L$ **do**
  for $\beta \leftarrow \beta_1, \ldots, \beta_V$ **do**
    Train the network θ_{ij} over $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_t$:

$$\theta_{ij} = \arg\min_{\theta} E[\mathcal{L}_s(\theta, \mathcal{S}) + \lambda \mathcal{L}_e(\theta, \mathcal{T}) - \beta \mathcal{L}_d(\theta, \mathcal{T})]$$

  end for
end for
Deep Embedded Validation [28]:
  1) Get DEV Risks of all models $\mathcal{R} = \{\text{GetRisk}(\theta_{ij})\}$ over $\mathcal{D}_{\text{val}}$
  2) Pick the best $(i^*, j^*) = \arg\min_{1 \le i \le L, 1 \le j \le V} \mathcal{R}_{ij}$

hyperparameters λ and β and picks the best solution in the end.

In many experiments, we also tried a fast model selection procedure, with which MEDM starts by fixing $\beta = 0$ at first and finds the smallest possible λ^* in achieving nearly minimal entropy of $\mathcal{L}_e(\theta^*, \mathcal{T}) = 0$. Then, it increases β from 0 to 1 for searching the best β^* with λ keeping fixed at λ^*. Although many experiments support this fast model selection procedure, there are some cases in which the nearly minimal entropy of $\mathcal{L}_e(\theta^*, \mathcal{T}) = 0$ is not easy to justify, especially for MEDM with LS (4).

D. Discussion

Given two classifiers $f, f^* \in \mathcal{H}$, and define the target disparity between f and f^* as

$$\epsilon_t(f, f^*) = \mathbb{E}_{(x^t; y^t) \sim Q}\left[f(x^t) \neq f^*(x^t)\right]. \tag{13}$$

By using the triangle inequalities, it is straightforward to bound the target risk of f as

$$\epsilon_t(f) \le \epsilon_t(f^*) + \epsilon_t(f, f^*). \tag{14}$$

Suppose now that f^* be the ideal classifier of (7) for minimizing the sum of both source and target risks. The philosophy behind MEDM is to minimize $\epsilon_t(f, f^*)$. Note that the ideal classifier f^* is not available, and its performance is always bounded by that of an empirical fully supervised classifier f_e^*, namely, $\epsilon_t(f^*) \ge \epsilon_t(f_e^*)$. For an empirical fully supervised classifier f_e^*, we mean that it can ideally access labels for both source and target samples. Suppose now that the ideal classifier f^* works well and may perform very close to f_e^*; it could present similar properties as those of f_e^*. Empirically, we observe that a deep-learning-based solution of f_e^* for minimizing (7) has the following two properties.

1) $\mathcal{L}_e(f_e^*, \mathcal{T}) \approx 0$.
2) $E_{\mathcal{T}}[\mathcal{L}_d(f_e^*, \mathcal{T})] = H(\mathbf{q}^*) \le H(\mathbf{q})$.

Here, $\mathbf{q}^* = [q_1^*, \ldots, q_K^*]$ is the inferred category distribution over the target domain when inferring with f_e^*.

The above observation means that an ideal classifier for minimizing the sum of source risk and target risk may lead to both entropy minimization and a nearly constant diversity over any fully randomized target batch. Therefore, MEDM seeks to find such a solution to meet both the above requirements by balancing between entropy minimization and diversity maximization.

TABLE II
ACCURACY (%) OF RESNET-50 MODEL ON VISDA-2017

Method	Synthetic → Real
GTA [38]	69.5
MCD [19]	69.8
CDAN [39]	70.0
MDD [40]	74.6
MEDM	79.6
MEDM-LS	**80.2**

IV. EXPERIMENTS

We evaluate MEDM with state-of-the-art domain adaptation methods for various transferring tasks, which includes VisDA-2017, ImageCLEF-DA, Office-Home, and Office-31 datasets. VisDA-2017 is known as the largest and highly unbalanced DA dataset, and ImageCLEF-DA is a small but balanced dataset, while both Office-Home and Office-31 are slightly unbalanced DA datasets with a large number of classes (65 for Office-Home and 31 for Office-31). In what follows, MEDM always means the use of the standard cross-entropy loss of (3), while MEDM-LS means the use of LS version of (4).

A. Experimental Setting

Throughout the experiments, we employ deep neural network architecture detailed as follows. It has a pretrained ResNet-50/101, followed by two fully connected layers, FC-1 of size 2048×1024 and FC-2 of size $1024 \times K$. Batch-normalization, ReLU activation, and dropout are only employed at the FC-1 layer. The dropout rate is set to 0.5. The Adam optimizer is employed with a learning rate of 0.0001. The batch size is set to 32. The learning rates of the layers trained from scratch are set to be 100 times those of fine-tuned layers from the pretrained ResNet. For model selection, we assume that $\lambda, \beta \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9.1.0\}$. A finer search of λ and β may lead to possibly better performance but with an increased computational burden.

We report the test accuracy results of MEDM, which are compared with state-of-the-art methods: deep adaptation network (DAN) [13], reverse gradient (RevGrad) [17], domain adversarial neural network (DANN) [46], residual transfer network (RTN) [34], multiadversarial domain adaptation (MADA) [45], generate to adapt (GTA) [37], maximum-classifier-discrepancy (MCD) [19], conditional domain adversarial network (CDAN) [38], margin disparity discrepancy (MDD) [39], batch spectral penalization (BSP) + CDAN [40], stepwise adaptive feature norm (SAFN) [42], sliced Wasserstein discrepancy (SWD) [41], Source HypOthesis Transfer (SHOT) [44], batch nuclear-norm maximization (BNM) [27], and MRKLD+LRENT [43]. Note that SANF† denotes our reexperiment with reference to SAFN.

TABLE III
ACCURACY (%) OF RESNET-101 MODEL ON THE VISDA DATASET († DENOTES OUR RE-EXPERIMENT RESULTS)

Method	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	mean
DAN [13]	87.1	63.0	76.5	42.0	90.3	42.9	85.9	53.1	49.7	36.3	85.8	20.7	61.1
RevGrad [17]	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
MCD [19]	87.0	60.9	83.7	64.0	88.9	79.6	84.7	76.9	88.6	40.3	83.0	25.8	71.9
CDAN [39]	85.2	66.9	83.0	50.8	84.2	74.9	88.1	74.5	83.4	76.0	81.9	38.0	73.7
BSP+CDAN [41]	92.4	61.0	81.0	57.5	89.0	80.6	90.1	77.0	84.2	77.9	82.1	38.4	75.9
SWD† [42]	88.0	80.0	79.0	**89.0**	90.0	67.0	84.0	75.0	86.0	63.0	84.0	28.0	76.0
SAFN† [43]	93.6	66.2	84.8	71.6	**94.3**	81.8	91.2	78.1	90.1	55.6	**88.6**	20.8	76.4
MRKLD+LRENT [44]	88.0	79.2	61.0	60.0	87.5	81.4	86.3	78.8	85.6	**86.6**	73.9	**68.8**	78.1
SHOT [45]	92.6	**81.1**	80.1	58.5	89.7	86.1	81.5	77.8	89.5	84.9	84.3	49.3	79.6
MEDM	**93.5**	80.4	**90.8**	70.3	92.8	**87.9**	91.1	79.8	**93.7**	83.6	86.1	38.7	**82.4**
MEDM-LS	93.1	74.2	86.0	68.7	93.9	87.2	**91.8**	**80.4**	92.9	83.1	88.0	49.8	**82.4**

TABLE IV
ACCURACY (%) ON IMAGECLEF-DA DATASET FOR UDA WITH RESNET-50

Method	I → P	P → I	I → C	C → I	C → P	P → C	Avg
DAN [13]	75.0 ± 0.4	86.2 ± 0.2	93.3 ± 0.2	84.1 ± 0.4	69.8 ± 0.4	91.3 ± 0.4	83.3
RTN [34]	75.6 ± 0.3	86.8 ± 0.1	95.3 ± 0.1	86.9 ± 0.3	72.7 ± 0.3	92.2 ± 0.4	84.9
RevGrad [17]	75.0 ± 0.6	86.0 ± 0.3	96.2 ± 0.4	87.0 ± 0.5	74.3 ± 0.5	91.5 ± 0.6	85.0
MADA [46]	75.0 ± 0.3	87.9 ± 0.2	96.0 ± 0.3	88.8 ± 0.3	75.2 ± 0.2	92.2 ± 0.3	85.8
CDAN [39]	77.7 ± 0.3	90.7 ± 0.2	**97.7** ± 0.3	91.3 ± 0.3	74.2 ± 0.2	94.3 ± 0.3	87.7
SAFN† [43]	78.0 ± 0.4	91.7 ± 0.5	96.2 ± 0.1	91.1 ± 0.3	77.0 ± 0.5	94.7 ± 0.3	88.1
SWD† [42]	79.0 ± 0.3	92.0 ± 0.2	93.0 ± 0.3	89.0 ± 0.3	73.0 ± 0.2	94.0 ± 0.3	87.7
MEDM	**78.5** ± 0.5	**93.0** ± 0.5	96.1 ± 0.2	**92.8** ± 0.5	**77.2** ± 0.7	**95.5** ± 0.4	**88.9**
MEDM-LS	78.2	93.3	97.2	93.0	78.3	95.5	**89.3**

TABLE V
ACCURACY (%) ON OFFICE-HOME FOR UDA WITH RESNET-50

Method	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg
DAN [13]	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3	56.3
DANN [47]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
JAN [48]	45.9	61.2	68.9	50.4	59.7	61.0	45.8	43.4	70.3	63.9	52.4	76.8	58.3
CDAN [39]	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
MDD [40]	54.9	73.7	77.8	60.0	71.4	71.8	61.2	53.6	78.1	72.5	60.2	82.3	68.1
SAFN† [43]	44.5	64.6	71.2	56.0	65.3	70.4	58.0	44.6	70.4	65.9	**61.4**	70.9	61.9
SWD† [42]	50.5	70.1	71.2	59.0	70.3	68.4	58.0	50.6	75.4	70.9	**61.4**	73.3	64.9
BNM [27]	52.3	73.9	80.0	63.3	72.9	74.9	61.7	49.5	79.7	70.5	53.6	82.2	67.9
SHOT [45]	56.9	**78.1**	81.0	67.9	78.4	78.1	**67.0**	54.6	81.8	73.4	58.1	84.5	71.6
MEDM	57.1	76.1	80.0	62.0	72.7	76.0	62.3	53.4	81.2	69.9	59.8	83.9	69.5
MEDM-LS	**57.5**	77.5	**83.2**	**69.1**	**78.9**	**80.7**	66.6	**54.9**	**83.4**	**74.9**	59.8	**85.4**	**72.5**

B. VisDA-2017

The VisDA challenge [48] aims to test domain adaptation method's ability to transfer source knowledge and adapt it to novel target domains. As the largest domain-adaptation dataset, the VisDA dataset contains 280k images across 12 categories from the training, validation, and testing domains. The training domain (the source domain) is a set of synthetic 2-D renderings of 3-D models generated from different angles and with different lighting conditions, while the validation domain (the target domain) is a set of realistic photographs. The source domain contains 152 397 synthetic images, and the target domain has 55 388 real images.

Note that the target domain is highly unbalanced, where the number of samples for each category is $[l_1, \ldots, l_{12}] =$ *[3646, 3475, 4690, 10 401, 4691, 2075, 5796, 4000, 4549, 2281, 4236, 5548]. Therefore, the VisDA-2017 also serves to justify the suitability of MEDM for highly unbalanced dataset*. The ground-truth category distribution can be calculated as $\mathbf{q} = 1/(\sum_{i=1}^{12} l_i)[l_1, \ldots, l_{12}]$. Then, the entropy of $\mathbf{q}$ can be directly computed as $H(\mathbf{q}) = 2.3927$.

Table II compares various methods with the pretrained ResNet-50 architecture, while Table III with the pretrained ResNet-101. Our method performs the best in the final mean accuracy among various methods. It surpasses the second best (SHOT) by 2.8% in the final mean accuracy for the scenarios of both ResNet-50 and ResNet-101. With ResNet-101, either MEDM or MEDM-LS achieves the record mean accuracy of 82.4%.

C. ImageCLEF-DA

ImageCLEF-DA is a publicly available dataset for ImageCLEF 2014 domain adaptation challenge. It has 12 common categories shared by the three public datasets: Caltech-256 (C), ImageNet ILSVRC 2012 (I), and Pascal VOC 2012 (P), which are also considered as three different domains. For 12 common categories, they are airplane, bike, bird, boat,

TABLE VI
ACCURACY (%) ON OFFICE-31 DATASET FOR UDA WITH RESNET-50

Method	A → W	D → W	W → D	A → D	D → A	W → A	Avg
DAN [13]	83.8 ± 0.4	96.8 ± 0.2	99.5 ± 0.1	78.4 ± 0.2	66.7 ± 0.3	62.7 ± 0.2	81.3
RevGrad [17]	82.0 ± 0.4	96.9 ± 0.2	99.1 ± 0.1	79.7 ± 0.4	68.2 ± 0.4	67.4 ± 0.5	82.2
MADA [46]	90.0 ± 0.1	97.4 ± 0.1	99.6 ± 0.1	87.8 ± 0.2	70.3 ± 0.3	66.4 ± 0.3	85.2
CDAN [39]	94.1 ± 0.1	98.6 ± 0.1	100.0 ± 0.0	92.9 ± 0.2	71.0 ± 0.3	69.3 ± 0.3	87.7
BSP+CDAN [41]	93.3 ± 0.2	98.2 ± 0.2	100.0 ± 0.0	93.0 ± 0.2	73.6 ± 0.3	72.6 ± 0.3	88.5
MDD [40]	$\mathbf{94.5} \pm 0.3$	98.4 ± 0.1	100.0 ± 0.0	$\mathbf{93.5} \pm 0.2$	$\mathbf{74.6 \pm 0.3}$	72.2 ± 0.1	88.9
SAFN† [43]	90.1 ± 0.6	98.6 ± 0.2	99.8 ± 0.0	90.7 ± 0.5	73.0 ± 0.2	70.2 ± 0.3	87.1
SWD† [42]	90.4 ± 0.3	98.7 ± 0.1	99.9 ± 0.1	94.7 ± 0.2	70.3 ± 0.3	70.5 ± 0.1	87.4
MRKLD+LRENT [44]	89.4 ± 0.7	98.9 ± 0.4	100 ± 0.0	88.7 ± 0.8	72.6 ± 0.7	70.9 ± 0.5	86.8
BNM [27]	90.3	91.5	98.5	100.0	70.9	71.6	87.1
SHOT [45]	90.9	98.8	99.9	93.1	74.5	74.8	88.7
MEDM	93.4 ± 0.6	$\mathbf{98.8} \pm 0.1$	$\mathbf{100.0} \pm 0.0$	93.4 ± 0.5	74.2 ± 0.2	$\mathbf{75.4} \pm 0.4$	**89.2**
MEDM-LS	93.4	**99.2**	99.8	93.2	75.1	75.4	**89.3**

TABLE VII
EFFECT OF β ON DIVERSITY MAXIMIZATION OVER VISDA-2017 (GROUND-TRUTH CATEGORY DIVERSITY $H(\mathbf{q}) = 2.3927$)

β	plane	bcycl	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	**mean**	**diversity**
0.0	89.6	**0.1**	94.7	**2.1**	88.5	99.2	89.0	72.0	58.0	**2.0**	77.1	**0.4**	56.0	**1.7862**
0.1	91.4	62.5	86.2	**1.9**	90.7	93.3	89.0	79.4	89.2	36.5	86.9	**0.7**	67.3	**1.9952**
0.2	91.6	77.6	66.4	**1.6**	90.7	83.4	89.4	78.6	89.7	74.9	89.6	**0.3**	69.5	**2.0619**
0.3	94.4	76.2	87.3	61.1	91.1	79.6	88.0	80.0	92.3	78.9	89.7	35.0	79.4	**2.2446**
0.4	92.7	83.1	82.2	65.8	89.2	89.9	79.8	78.6	91.3	77.7	90.9	33.8	79.6	**2.2493**
0.5	94.2	77.8	80.9	58.4	90.9	**25.9**	81.4	76.1	89.1	67.6	89.7	40.1	72.7	**2.2704**

bottle, bus, car, dog, horse, monitor, motorbike, and people. ImageCLEF-DA is a balanced dataset with 50 images in each category and 600 images in each domain. We consider all domain combinations and build six domain-adaptation tasks: $I \rightarrow P$, $P \rightarrow I$, $I \rightarrow C$, $C \rightarrow I$, $C \rightarrow P$, and $P \rightarrow C$.

Table IV shows the classification accuracy results for various methods on the ImageCLEF-DA dataset with the ResNet50 architecture. The result of the MEDM is obtained by only training 100 epochs, which, however, neatly outperforms the other deep adaptation methods among five adaptation tasks: $I \rightarrow P$, $P \rightarrow I$, $C \rightarrow I$, $C \rightarrow P$, and $P \rightarrow C$. The best average accuracy (88.9%) is achieved by MEDM, which improves SAFN by about 0.8%. As shown, MEDM-LS performs even slightly better than MEDM.

D. Office-Home

Office-Home [49] is a typical dataset with a large number of classes (65 classes), which containing 15 500 images from four visually very different domains: **Ar**tistic images, **Cl**ip Art, **Pr**oduct images, and **R**eal-**w**orld images. We consider all domain combination among these four domains, resulting 12 domain-adaptation tasks.

Table V shows the classification accuracy results on the Office-Home dataset with the ResNet50 architecture. The result of the MEDM (or MEDM-LS) is obtained by only training 100 epochs. As shown, MEDM-LS still outperforms SHOT by about 0.9% mean accuracy, while MEDM performs inferior to SHOT. Since both SHOT and MEDM-LS employed LS, this means that the use of LS may lead to a considerable performance gap for domain adaptation over Office-Home. Note that Office-Home has a large number of classes (65), which may be a potential reason for achieving a better performance advantage with LS.

E. Office-31

Office-31 is a standard benchmark dataset for visual domain adaptation, which has 4652 images and 31 categories collected from three domains: Amazon (A), Webcam (W), and DSLR (D). The Amazon (A) domain contains 2817 images downloaded from amazon.com. We consider all domain combinations, resulting in six domain-adaptation tasks.

For the transferring tasks over Office-31, we employ the same neural network architecture as ImageCLEF-DA. We compare the average classification accuracy of each method on ten random experiments and report the standard error of the classification accuracies by different experiments of the same transfer task. In all experiments, we train each model for 100 epochs, and exceptions include $D \rightarrow A$ and $W \rightarrow A$, where 200 epochs are employed.

We report the classification accuracy results on the Office-31 dataset, as shown in Table VI. Office-31 has three domains of different sizes, which results in unevenly distributed classes in each domain.

Among various domain-adaptation methods, MEDM still performs the best for the mean accuracy. MEDM performs the best for three adaptation tasks, $D \rightarrow W$, $W \rightarrow D$, and $W \rightarrow A$, while MDD [39] performs the best for the three remaining tasks.

F. Ablation Study

1) Effect of β on Diversity Maximization: The superiority of MEDM in the VisDA challenge shows that it is very effective for a highly unbalanced target dataset although the category diversity is expected to achieve its maximum value when the inferred categories are uniformly distributed. We guess that it works well due to the collaboration in meeting both requirements, namely, the minimization of entropy and the

TABLE VIII
EFFECT OF λ ON TRANSFERABILITY FOR $W \rightarrow A$

$\lambda+\beta$	$\mathcal{L}_e$	Acc	$\lambda+\beta$	$\mathcal{L}_e$	Acc
1.0 + 0	0.0	43.1	1.0 + 0.2	0.0	53.3
0.8 + 0	0.0	45.9	0.8 + 0.2	0.0	56.5
0.6 + 0	0.0	49.8	0.6 + 0.2	0.0	65.7
0.4 + 0	0.1	54.0	0.4 + 0.2	0.1	74.3
0.3 + 0	0.2	62.2	0.3 + 0.2	0.2	75.5
0.2 + 0	0.3	63.2	0.2 + 0.2	0.3	74.5
0.1 + 0	0.4	68.1	0.1 + 0.2	0.4	72.3

maximization of category diversity, where the parameter β (11) is used to balance two individual requirements.

To investigate the choice of β on the final performance, we also show the accuracy of MEDM with ResNet-50 by fixing $\lambda = 1.0$ and varying β from $\{0.0, 0.1, 0.2, 0.3, 0.4, 0.5\}$. As shown in Table VII, we observed that $E_{\mathcal{T}}\{\mathcal{L}_d(\mathcal{T})\}$ after training 10 epochs is always less than the entropy of the ground-truth target category distribution $H(\mathbf{p}) = 2.3978$, which means that the maximization of $\mathcal{L}_d(\mathcal{T})$ under the constraint of entropy minimization does not necessarily produce the uniformly distributed categories. When $\beta < 0.3$, it results into poorer performance as some categories (car/truck) simply fail to be identified. With the increase in β, the practical diversity also grows. When β increases to 0.5, it also results into significantly worse performance compared to $\beta = 0.4$. Essentially, individual entropy minimization may automatically tradeoff with diversity maximization *if the values of λ and β are properly validated by the use of DEV [28].*

2) Effect of λ on Transferability: Entropy minimization in MEDM can be adjusted by varying λ. As shown in Algorithm 1, MEDM encourages the use of small λ whenever the target entropy approaches zero as the training iteration goes on. When $\beta = 0$, the transferability is achieved by minimization of both the supervised loss on the source domain and the entropy loss on the target domain. With small λ and keeping the (target) entropy small enough at the same time, it is expected that the end-to-end training of (11) ensures better transferability.

To investigate the choice of λ on the final performance, we also show the accuracy of MEDM on the task $W \rightarrow A$ on the Office-31 dataset when λ takes its value from $\{0.2, 0.3, 0.4, 0.6, 0.8, 1.0\}$ and $\beta \in \{0.0, 0.2\}$. With the smallest possible value of λ for $\mathcal{L}_e(\theta_i, \mathcal{T}) \rightarrow 0$, MEDM can achieve the best performance with a suitable choice of β, as shown in Table VIII. This means that the better transferability could be ensured with smaller possible value of λ if $\mathcal{L}_e(\theta_i, \mathcal{T}) \rightarrow 0$ is satisfied at the end of training.

3) McNemar Test for Comparing EMO and MEDM: We consider to compare EMO ($\lambda = 0.2$ and accuracy $= 63.4\%$) and MEDM ($\lambda = 0.2, \beta = 0.2$, and accuracy $= 74.0\%$) on the task $W \rightarrow A$ for the Office-31. To apply McNemars test [50], we test these classifiers on the target domain. For each example $x \in \mathcal{D}_t$, we record how it was classified and count the following four items, n_{00}, n_{01}, n_{10}, and n_{11}, where n_{00} is the number of examples misclassified by both EMO and MEDM, n_{01} is the number of examples misclassified by EMO but not by MEDM, n_{10} is the number of examples misclassified by MEDM but not by EMO, and n_{11} is the number of examples misclassified by neither EMO nor by MEDM.

Let

$$\zeta = \frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}} \tag{15}$$

be the investigated statistics, which is distributed (approximately) as χ^2 with 1 degree of freedom. In the considered scenario, we finally arrive at $\zeta = 147.4$ with $n_{01} = 401$ and $n_{10} = 102$. If the null hypothesis is correct, the probability that ζ is greater than $\chi^2_{1,0.9999} = 14.4$ is less than 0.0001. Hence, we can reject the null hypothesis in favor of the hypothesis that the two algorithms have very different performance.

V. RELATED WORKS

A. Domain Adaptation

In recent years, domain adaptation has been extensively studied. The main idea is to find domain-invariant feature representations, which enables the learned classifiers from the source domain to generalize well to the target domain. Some recent works on this topic include [51]–[53]. Li *et al.* [51] considered to incorporate domain-invariant feature learning with preserved category-discriminative information. An explicit feature map and feature selection method was proposed in [52]. By learning a neural embedding model to bridge cross-domain distribution divergence, Wang *et al.* [53] proposed to transform both source and target samples into a common feature-embedding space within a regularized risk minimization framework. Instead of using feature alignment, there are some approaches focusing on the direct adaptation of the classifier [4], [27], [39].

Among various domain-adaptation approaches, we shall briefly review the related works on entropy minimization in what follows.

B. Entropy-Minimization Methods

Entropy minimization was first proposed in [35] for semisupervised learning. In many UDA scenarios with very limited domain-shift between source and target domains, it may work well. When the effect of domain-shift increases, the use of entropy regularization is often not enough for achieving sufficient discrimination capability in the target domain [26]. Hence, various ancillary adaptation techniques were invoked, such as covariance alignment [26], batch normalization [22], or learning by association [20].

It was argued in [26] that entropy minimization could be achieved by the optimal alignment of second-order statistics between source and target domains, and therefore, a hyperparameter validation method was proposed for balancing the reduction of the domain shift and the supervised classification on the source domain in an optimal way. In [22], a novel domain alignment layer was introduced for reducing the domain shift by matching source and target distributions to a referenced one, and entropy minimization was also explicitly employed, which was believed to promote classification models with high confidence on unlabeled samples. Long *et al.* [34] used entropy minimization in their approach to directly measure how far samples are from a decision boundary. Satio *et al.* [55, Appendix] proposed an entropy-based adversarial dropout regularization approach, which employed the entropy of the target samples in

implementing min–max adversarial training. In [38], entropy conditioning was employed that controls the uncertainty of the classifier predictions to guarantee the transferability, which can help the proposed CDAN to converge to better solutions.

By noting that not all samples in the target domain are transferable and negative results for entropy minimization may incur if the entropy of these nontransferable samples is forcefully minimized, Wang *et al.* [55] proposed a novel attention-weighted entropy loss, where the weighting attention value of a given target sample is generated according to its transferability. By minimizing the attention-weighted entropy loss, the proposed domain adaptation method can alleviate the effect of negative transfer. Zhang *et al.* [56] proposed to reduce interdomain discrepancy with adversarial learning and diminish intradomain discrepancy using entropy minimization. Note that entropy minimization in [56] is performed not only over the target domain but also over the source domain.

After the submission of this article, the reviewers bring [27], [43], [44], [57] to our attention.

Our method differs with [44] in several aspects. First, the proposed diversity is calculated in a batchwise manner, while Liang *et al.* [44] proposed to compute the diversity over the whole target domain. Second, our method introduces two weighting parameters (β, γ) for balancing entropy minimization and diversity maximization, and the determination of (β, γ) is essential to the superiority of the proposed MEDM algorithm. Note that Liang *et al.* [44] employ the sum of entropy loss and diversity loss in the final loss without any consideration of weighting. Finally, we provide the theoretical justification for balancing entropy minimization and diversity maximization, which constitutes a major difference with SHOT [44]. Compared to our method, Saito *et al.* [57] also employ entropy minimization for feature learning. However, it is a semisupervised domain adaptation method, where the adaptation is achieved by alternately minimizing the target prediction entropy with respect to the feature encoder and maximizing it with respect to the final classifier. Note that the neural network in [57] consists of the feature encoder followed by the classifier. MEDM, however, is to minimize a single loss with respect to the whole network. Therefore, minimizing both entropy and negative diversity (or diversity maximization) is with respect to the whole network for MEDM. We also noticed that entropy maximization with respect to the final classifier in [57] is used to achieve some diversity over the predicted categories, while our method explicitly enforces a well-defined diversity over the predicted categories.

Our method is also similar to [27], and both methods address the problem of prediction discrimination and diversity. However, the way to remedy the problem is very different. Our method adjusts the weighting factors for balancing prediction entropy and diversity, while Cui *et al.* [27] employ BNM for ensuring both high prediction discriminability and diversity. In [43], confidence regularized self-training framework was proposed for achieving balancing between self-training loss and confidence regularizer through the weighting factor α. Note that experiments show that the proposed self-training methods are not sensitive to the weighting factor α, while our method is sensitive to the weighting factors γ and β.

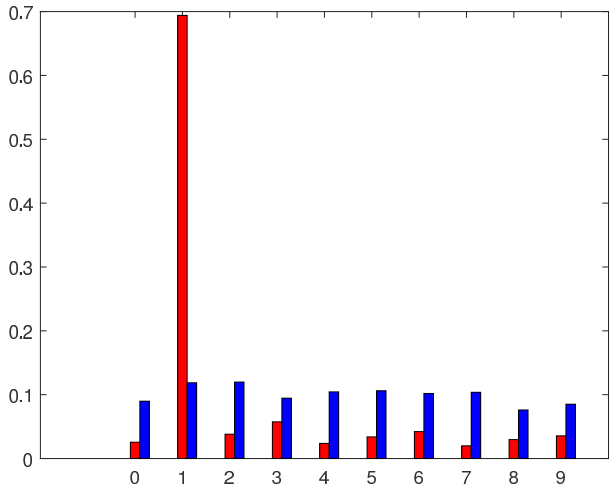


Fig. 4. Predicted category distribution (9) for both MEDM (blue) and EMO (6) (red) for a batch of target samples after a training iterations of 10 000: EMO results into a trivial solution, where digit-1 dominates among others.

VI. Conclusion

Entropy minimization has been shown to be a powerful tool for domain adaptation. However, entropy minimization is insufficient for the minimization of the target risk, and trivial solutions are often observed. In this article, we propose to employ diversity maximization for further regulating the minimal-entropy domain-adaptation methods. We show there exists a tradeoff for entropy minimization and diversity maximization toward the close-to-ideal domain adaptation. With the recently proposed unsupervised model selection method, we show that the proposed MEDM outperforms state-of-the-art methods on several domain adaptation datasets, boosting a large margin, especially on the largest VisDA dataset for cross-domain object classification.

Appendix
Trivial Solution Demonstration for Entropy-Minimization-Only Method

In this appendix, we consider the transfer task of SVHN → MNIST for demonstrating the trivial solutions of EMO.

The MNIST handwritten digits database has a training set of 60 000 examples and a test set of 10 000 examples. The digits have been size-normalized and centered in fixed-size images. SVHN is a real-world image dataset for machine learning and object recognition algorithms with a minimal requirement on data preprocessing and formatting. It has 73 257 digits for training and 26 032 digits for testing. We focus on the task SVHN → MNIST in experiments.

We employed the CNN architecture used in [17]. The number of training iterations is set to 50 000, and the learning rate is set to 0.001. We run ten experiments for computing average accuracy and its deviation.

First, we show that EMO simply results in a trivial solution, as indicated in Fig. 4, where digit-1 dominates among other categories for the model trained with EMO (6) ($\lambda = 1$). By inferring several target batches over the trained model, we observed that digit-1 always dominates for EMO. For MEDM, the predicted category distribution, however, is very close to the true uniform distribution.

TABLE IX
AVERAGE ACCURACY (%) FOR SVHN → MNIST

Method	Acc
RevGrad ([17])	73.9
ADDA ([34])	76.0
DTN ([59])	84.4
TRIPPLE ([23])	86.2
COREL ([15])	90.2
MECA ([60])	95.2
MEDM(Ours)	**98.7** $\pm$ 0.3

Then, we compare our method with six methods in Table IX for UDA, including state-of-the-art methods in visual domain adaptation: RevGrad [17], adversarial discriminative domain adaptation (ADDA) [25], domain transfer network (DTN) [58],TRIPPLE [23], CORrelation ALignment (CORAL) [15], and minimal-entropy correlation alignment (MECA) [26]. MEDM performs the best, and it achieves the average accuracy of 98.7%, which improves 3.5% compared to MECA.

ACKNOWLEDGMENT

The authors wish to thank the Associate Editor and anonymous reviewers for their constructive comments. They also thank Dr. Yueming Yin and Mr. Yuqing Ding for their help in experiments.

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, Jun. 2016, pp. 770–778.
[2] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. CVPR*, Jul. 2017, pp. 4700–4708.
[3] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *Proc. CVPR*, Jun. 2011, pp. 1521–1528.
[4] S. Wang, L. Zhang, W. Zuo, and B. Zhang, "Class-specific reconstruction transfer learning for visual recognition across domains," *IEEE Trans. Image Process.*, vol. 29, pp. 2424–2438, Jan. 2020.
[5] A. Chadha and Y. Andreopoulos, "Improved techniques for adversarial discriminative domain adaptation," *IEEE Trans. Image Process.*, vol. 29, pp. 2622–2637, Jan. 2020.
[6] B. Gholami, P. Sahu, O. Rudovic, K. Bousmalis, and V. Pavlovic, "Unsupervised multi-target domain adaptation: An information theoretic approach," *IEEE Trans. Image Process.*, vol. 29, pp. 3993–4001, Jan. 2020.
[7] W.-G. Chang, T. You, S. Seo, S. Kwak, and B. Han, "Domain-specific batch normalization for unsupervised domain adaptation," in *Proc. CVPR*, Jun. 2019, pp. 7354–7362.
[8] S. Roy, A. Siarohin, E. Sangineto, S. R. Bulo, N. Sebe, and E. Ricci, "Unsupervised domain adaptation using feature-whitening and consensus loss," in *Proc. CVPR*, Jun. 2019, pp. 9471–9480.
[9] C. Chen *et al.*, "Progressive feature alignment for unsupervised domain adaptation," in *Proc. CVPR*, Jun. 2019, pp. 627–636.
[10] M. Kim, P. Sahu, B. Gholami, and V. Pavlovic, "Unsupervised visual domain adaptation: A deep max-margin Gaussian process approach," in *Proc. CVPR*, Jun. 2019, pp. 4380–4390.
[11] M. Chen, K. Q. Weinberger, and J. Blitzer, "Co-training for domain adaptation," in *Proc. NIPS*, 2011, pp. 2456–2464.
[12] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan, "Domain separation networks," in *Proc. NIPS*, 2016, pp. 343–351.
[13] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. ICML*, 2015, pp. 97–105.
[14] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," 2014, *arXiv:1412.3474*. [Online]. Available: http://arxiv.org/abs/1412.3474
[15] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Proc. ICCV Workshop Transferring Adapting Source Knowl. Comput. Vis.*, 2016, pp. 443–450.
[16] G. Ian *et al.*, "Generative adversarial nets," in *Proc. NIPS*, 2014, pp. 1–9.
[17] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proc. ICML*, 2015, pp. 1180–1189.
[18] J. Hoffman *et al.*, "CyCADA: Cycle-consistent adversarial domain adaptation," in *Proc. ICML*, 2018, pp. 1989–1998.
[19] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proc. CVPR*, Jun. 2018, pp. 3723–3732.
[20] P. Haeusser, T. Frerix, A. Mordvintsev, and D. Cremers, "Associative domain adaptation," in *Proc. ICCV*, Oct. 2017, pp. 2765–2773.
[21] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko, "Simultaneous deep transfer across domains and tasks," in *Proc. ICCV*, Dec. 2015, pp. 4068–4076.
[22] F. M. Carlucci, L. Porzi, B. Caputo, E. Ricci, and S. R. Bulo, "AutoDIAL: Automatic domain alignment layers," in *Proc. ICCV*, Oct. 2017, pp. 5077–5085.
[23] K. Saito, Y. Ushiku, and T. Harada, "Asymmetric tri-training for unsupervised domain adaptation," in *Proc. ICML*, 2017, pp. 2988–2997.
[24] R. Shu, H. Bui, and S. Ermon, "A dirt-T approach to unsupervised domain adaptation," in *Proc. ICLR*, 2018, pp. 1–19.
[25] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. CVPR*, Jul. 2017, pp. 7167–7176.
[26] P. Morerio, J. Cavazza, and V. Murino, "Minimal-entropy correlation alignment for unsupervised deep domain adaptation," in *Proc. ICLR*, 2018, pp. 1–14.
[27] 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 *Proc. CVPR*, Jun. 2020, pp. 3941–3950.
[28] K. You, X. Wang, M. Long, and M. Jordan, "Towards accurate model selection in deep unsupervised domain adaptation," in *Proc. ICML*, 2019, pp. 7124–7133.
[29] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," in *Proc. NIPS*, 2007, p. 137.
[30] J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman, "Learning bounds for domain adaptation," in *Proc. NIPS*, 2008, pp. 129–136.
[31] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. CVPR*, Jun. 2016, pp. 2818–2826.
[32] Y. Xu, Y. Xu, Q. Qian, H. Li, and R. Jin, "Towards understanding label smoothing," 2020, *arXiv:2006.11653*. [Online]. Available: http://arxiv.org/abs/2006.11653
[33] R. Müller, S. Kornblith, and G. Hinton, "When does label smoothing help?" in *Proc. NeurIPS*, 2019, pp. 1–10.
[34] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Proc. NIPS*, 2016, pp. 136–144.
[35] Y. Grandvalet and Y. Bengio, "Semi-supervised learning by entropy minimization," in *Proc. NIPS*, 2005, pp. 529–536.
[36] S. Cui, S. Wang, J. Zhuo, C. Su, Q. Huang, and Q. Tian, "Gradually vanishing bridge for adversarial domain adaptation," in *Proc. CVPR*, Jun. 2020, pp. 12455–12464.
[37] S. Sankaranarayanan, Y. Balaji, C. D. Castillo, and R. Chellappa, "Generate to adapt: Aligning domains using generative adversarial networks," in *Proc. CVPR*, Jun. 2018, pp. 8503–8512.
[38] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Proc. NIPS*, 2018, pp. 1647–1657.
[39] Y. Zhang, T. Liu, M. Long, and M. Jordan, "Bridging theory and algorithm for domain adaptation," in *Proc. ICML*, 2019, pp. 7404–7413.
[40] X. Chen, S. Wang, M. Long, and J. Wang, "Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation," in *Proc. ICML*, 2019, pp. 1081–1090.
[41] C.-Y. Lee, T. Batra, M. H. Baig, and D. Ulbricht, "Sliced Wasserstein discrepancy for unsupervised domain adaptation," in *Proc. CVPR*, Jun. 2019, pp. 10285–10295.
[42] R. Xu, G. Li, J. Yang, and L. Lin, "Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation," in *Proc. ICCV*, Oct. 2019, pp. 1426–1435.
[43] Y. Zou, Z. Yu, X. Liu, B. V. K. V. Kumar, and J. Wang, "Confidence regularized self-training," in *Proc. ICCV*, Oct. 2019, pp. 5982–5991.
[44] J. Liang, D. Hu, and J. Feng, "Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation," in *Proc. ICML*, 2020, pp. 6028–6039.
[45] Z. Pei, Z. Cao, M. Long, and J. Wang, "Multi-adversarial domain adaptation," in *Proc. AAAI*, 2018, pp. 1–8.
[46] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2016–2030, 2016.
[47] M. Long, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proc. ICML*, 2017, pp. 2208–2217.

[48] X. Peng, B. Usman, N. Kaushik, D. Wang, J. Hoffman, and K. Saenko, "VisDA: A synthetic-to-real benchmark for visual domain adaptation," in *Proc. CVPR Workshops*, Jun. 2018, pp. 2021–2026.
[49] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proc. CVPR*, Jul. 2017, pp. 5018–5027.
[50] B. S. Everitt, *The Analysis of Contingency Tables*. London, U.K.: Chapman & Hall, 1977.
[51] S. Li *et al.*, "Discriminative transfer feature and label consistency for cross-domain image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 4842–4856, Nov. 2020.
[52] W.-Y. Deng, A. Lendasse, Y.-S. Ong, I. W.-H. Tsang, L. Chen, and Q.-H. Zheng, "Domain adaption via feature selection on explicit feature map," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 1180–1190, Apr. 2019.
[53] Z. Wang, B. Du, and Y. Guo, "Domain adaptation with neural embedding matching," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 7, pp. 2387–2397, Jul. 2020.
[54] K. Saito, Y. Ushiku, T. Harada, and K. Saenko, "Adversarial dropout regularization," in *Proc. ICLR*, 2018, pp. 1–14.
[55] X. Wang, L. Li, W. Ye, M. Long, and J. Wang, "Transferable attention for domain adaptation," in *Proc. AAAI*, 2019, pp. 5345–5352.
[56] Y. Zhang *et al.*, "From whole slide imaging to microscopy: Deep microscopy adaptation network for histopathology cancer image classification," in *Proc. MICCAI*, 2019, pp. 360–368.
[57] K. Saito, D. Kim, S. Sclaroff, T. Darrell, and K. Saenko, "Semi-supervised domain adaptation via minimax entropy," in *Proc. ICCV*, Oct. 2019, pp. 8050–8058.
[58] Y. Taigman, A. Polyak, and L. Wolf, "Unsupervised cross-domain image generation," in *Proc. ICLR*, 2017, pp. 1–14.
[59] J. Manders, T. van Laarhoven, and E. Marchiori, "Adversarial alignment of class prediction uncertainties for domain adaptation," in *Proc. ICPRAM*, 2019, pp. 1–11.

Xiaofu Wu (Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from Nanjing Institute of Communications Engineering, Nanjing, China, in 1996 and 1999, respectively, and the Ph.D. degree in electrical engineering from Peking University, Beijing, China, in 2005.

From 2005 to 2007, he was with a Post-Doctoral Researcher with the National Mobile Communication Research Laboratory, Southeast University, Nanjing. Since 2012, he has been with Nanjing University of Posts and Telecommunications, Nanjing, where he is currently a Full Professor. His research interests are in coding and information theory, information-theoretic security, machine learning, and computer vision.

Suofei Zhang received the B.S. and M.S. degrees from the School of Mechanical Engineering, Jiangsu University, Zhenjiang, China, in 2004 and 2007, respectively, and the Ph.D. degree from the School of Information Science and Engineering, Southeast University, Nanjing, China, in 2013.

He is currently a Lecturer with the School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing. His current research interests include image processing, machine learning, computer vision, and artificial intelligence.

Quan Zhou (Member, IEEE) received the master's and Ph.D. degrees from the Huazhong University of Science and Technology, Wuhan, China, in 2006 and 2013, respectively.

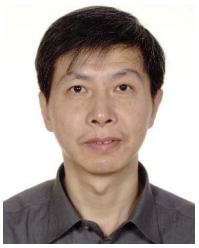
He is currently an Associate Professor of telecommunications and information engineering with Nanjing University of Posts and Telecommunications, Nanjing, China. He has published over 70 research articles in top journals, including IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, and *Pattern Recognition*. His main research interests include image understanding, object detection and recognition in images, machine learning, and digital geometry.

Dr. Zhou is also an Editorial Board Member of journals, such as *Multimedia Tools and Applications*, *ACM/Springer Mobile Networks & Applications*, and *Cognitive Robotics*. He is also member of the International Association for Pattern Recognition (IAPR).

Zhen Yang (Senior Member, IEEE) received the B.Eng. and M.Eng. degrees from Nanjing University of Posts and Telecommunications, Nanjing, China, in 1983 and 1988, respectively, and the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China, in 1999, all in electrical engineering.

He was initially employed as a Lecturer by Nanjing University of Posts and Telecommunications in 1983, where he was promoted to Associate Professor in 1995 and then a Full Professor in 2000. He was a Visiting Scholar with the University of Bremen, Bremen, Germany, from 1992 to 1993, and an Exchange Scholar with the University of Maryland, College Park, MD, USA, in 2003. He has published more than 200 articles in academic journals and conferences. His research interests include various aspects of signal processing and communication, such as communication systems and networks, cognitive radio, spectrum sensing, speech and audio processing, compressive sensing, and wireless communication.

Prof. Yang is currently a fellow of Chinese Institute of Communications and the Vice-Director of the Editorial Board of the *Journal of Communications*. He is also a member of the Editorial Board for several other journals, such as *Chinese Journal of Electronics*, *Data Collection*, and *Processing*. He has served as the Vice-Chairman of Chinese Institute of Communications, the Chairman of Jiangsu Institute of Communications from 2010 to 2015, and the Chair of the Steering Committee of Asian Pacific Communication Conference (APCC) from 2013 to 2014.



Chunming Zhao (Member, IEEE) received the B.S. and M.S. degrees from Nanjing Institute of Posts and Telecommunications, Nanjing, China, in 1982 and 1984, respectively, and the Ph.D. degree from the Department of Electrical and Electronic Engineering, University of Kaiserslautern, Kaiserslautern, Germany, in 1993.

He has been a Post-Doctoral Researcher with the National Mobile Communications Research Laboratory, Southeast University, Nanjing, where he is currently a Professor and the Vice-Director. He has managed several key projects of Chinese Communications High Technology Program. His research interests include communication theory, coding/decoding, mobile communications, and VLSI design.

Dr. Zhao won the First Prize of National Technique Invention of China in 2011. He was awarded Excellent Researcher from the Ministry of Science and Technology, China.

Longin Jan Latecki received the Ph.D. and Habilitation degrees from the University of Hamburg, Hamburg, Germany, in 1992 and 1996, respectively.

He is currently a Professor of computer science with Temple University, Philadelphia, PA, USA. He has published over 225 research articles and books. His main research interests include shape representation and similarity, object detection and recognition in images, robot perception, machine learning, and digital geometry.

Dr. Latecki is also an Editorial Board Member of journals, such as *Pattern Recognition*, *Computer Vision and Image Understanding*, and *International Journal of Mathematical Imaging*. He received the 2010 College of Science and Technology Research Excellence Award and the Annual Pattern Recognition Society Award together with Azriel Rosenfeld for the best article published in the journal *Pattern Recognition* in 1998. He was a recipient of the 2000 Olympus Prize and the Main Annual Award from the German Society for pattern recognition (DAGM).