Modeling Hierarchical Structural Distance for Unsupervised Domain Adaptation

Yingxue Xu*, Guihua Wen*, Yang Hu, Member, IEEE, and Pei Yang*, Member, IEEE

Abstract—Unsupervised domain adaptation (UDA) aims to estimate a transferable model for unlabeled target domains by exploiting labeled source data. Optimal Transport (OT) based methods have recently been proven to be a promising solution for UDA with a solid theoretical foundation and competitive performance. However, most of these methods solely focus on domain-level OT alignment by leveraging the geometricity of domains for domain-invariant features based on the global embeddings of images. However, global representations of images may destroy image structure, leading to the loss of local details that offer category-discriminative information. This study proposes an end-to-end Deep Hierarchical Optimal Transport method (DeepHOT), which aims to learn both domain-invariant and category-discriminative representations by mining hierarchical structural relations among domains. The main idea is to incorporate a domain-level OT and image-level OT into a unified OT framework, hierarchical optimal transport, to model the underlying geometry in both domain space and image space. In DeepHOT framework, an image-level OT serves as the ground distance metric for the domain-level OT, leading to the hierarchical structural distance. Compared with the ground distance of the conventional domain-level OT, the image-level OT captures structural associations among local regions of images that are beneficial to classification. In this way, DeepHOT, a unified OT framework, not only aligns domains by domain-level OT, but also enhances the discriminative power through image-level OT. Moreover, to overcome the limitation of high computational complexity, we propose a robust and efficient implementation of DeepHOT by approximating origin OT with sliced Wasserstein distance in image-level OT and accomplishing the mini-batch unbalanced domain-level OT. Extensive experiments show the superiority of DeepHOT in several benchmark datasets. The code is available on GitHub (https://github.com/Innse/DeepHOT).

Index Terms—Unsupervised domain adaptation, optimal transport, hierarchical optimal transport, deep learning.

I. INTRODUCTION

In the context of traditional machine learning, one generally assumes that the training and test data should be independent and identically distributed. However, in many practical vision problems, various factors, such as different acquisition conditions, changing lighting conditions, presence of background, can break this assumption and cause discrepancies in data distribution (a.k.a. domain shift). Domain adaptation methods aim to mitigate this issue by finding a shared representation among domains where a classifier can be learned and used for target domain [1]. In this work, we focus on unsupervised domain adaptation (UDA), where no label is available in the target domain.

Fruitful works [2], [3], [4], [5], [6] have been done for unsupervised domain adaptation, where the core idea boils down to aligning distributions across domains. Among others, optimal-transport-based methods have shown superiority in various unsupervised domain adaptation tasks by leveraging the underlying geometric structure of domains in domain-level OT alignment [1], [7]. They aim to achieve the overall minimum discrepancy between domains by finding optimal matching flow from source and target domains based on the ground distance of pairwise global representation of images [7], [8], [9], [10], [11]. However, these global representations of images destroy image structures, resulting in the loss of local features. Local features offer essential and transferable information that contributes to the discriminative power [12]. As a result, despite the decreased domain discrepancies, the discriminative power of image features across categories is degraded as well.

In this work, to learn domain-invariant yet category-discriminative representations, we propose a Deep Hierarchical Optimal Transport framework (DeepHOT) for unsupervised domain adaptation by incorporating the domain-level OT and the image-level OT into a unified OT framework to mine the hierarchical structural correlations at both domain-level and image-level. Compared with the conventional domain-level OT approaches for UDA, the ground distance of domain-level OT in the proposed DeepHOT is also an OT at image-level, instead of the distance between global representation of images, resulting in a nested OT framework. The benefit of image-level OT is that it is capable of capturing correspondences between local regions of two images for enhancing discriminative features.
Specifically, an image would be divided into a set of local patches, which then allows the patch-to-patch correlations to be encoded into the distance of pairwise images by image-level OT. Subsequently, by employing this distance as the ground distance of domain-level OT, DeepHOT improves the domain-level OT’s ability to discriminate between different categories. As a result, in a unified manner, not only domains can be aligned by domain-level OT, but the discriminative power is enhanced by image-level OT.

However, despite the performance superiority, optimal-transport-based methods have the drawback of high computational complexity. To this end, we propose a robust and efficient implementation of the DeepHOT framework. The key approaches to address this issue are three-fold. First, for image-level OT, sliced Wasserstein distance (SWD) [13] is applied to compute the discrepancy between pairwise images, which has been proved to be more efficient than Earth Mover Distance in time complexity. Second, for domain-level OT, we compute the domain distance over mini-batches and use the averaged result as a proxy for the original problem. And then, time complexity would be considerably reduced if using a small mini-batch. Third, it is difficult for mini-batch OT methods to converge to the true transport plan when the batch size decreases. Therefore, we exploit mini-batch unbalanced optimal transport (UOT) [14] for domain-level OT, which can provide a more robust solution to the mini-batch sampling.

The contributions of this work are summarized as follows: (1) We propose to learn domain-invariant yet category-discriminative representations for UDA in a unified OT-based and end-to-end manner by a unified deep hierarchical optimal transport method. To the best of our knowledge, it is the first deep hierarchical optimal transport method for UDA. (2) We propose a robust and efficient implementation of the DeepHOT framework, making it possible to apply it for a large-scale dataset in practice. (3) Extensive experiments on four benchmark domain adaptation datasets show consistent improvement over state-of-the-art methods.

II. RELATED WORK

A. Unsupervised Domain Adaptation

Unsupervised domain adaptation aims to learn a model on the labeled source domain that can adapt the target domain in which labels are unavailable. Most existing works can be broadly classified into two categories: discrepancy-based domain alignment and adversarial learning.

1) Discrepancy-Based Domain Alignment: This kind of methods [15], [16] tend to explicitly minimize a divergence that measures the gap between the source and target domains. A typical discrepancy metric, Maximum Mean Discrepancy (MMD) [15], has been extensively applied to domain adaptation tasks and [17] further proposed multi-kernels MMD. Moreover, Weighted-MMD (WDAN) [16] integrated class prior with original MMD by class-specific auxiliary weights. Another measure that can be utilized for domain alignment is covariance, leading to a method Deep Correlation Alignment (Deep CORAL) [18]. These methods only consider aligning the marginal distribution, and however, Deep Transfer Network (DTN) [19] proposed a conditional alignment method using pseudo labels of target domain. Reference [20] learned a joint distribution by dynamically quantifying the relative importance of marginal and class-conditional distribution. Similarly, [21] dynamically aligned both the feature and label spaces. Reference [22] employed a teacher-student framework for source and target domains and then aligned their prediction. Along with jointly adapting the marginal and conditional distributions, CKET [23] further attempted to transfer discriminative information by enforcing the structure consistency between the original feature space and the latent feature space. The same team further underscored the significance of fine-grained cluster structures in the source and target domains [24] to enhance discriminative power. In recent developments, they have proposed applying structural risk minimization not only at the domain-level but also at the class-level to enhance discrimination [25]. Another branch is based on optimal transport which is illustrated in next subsection.

2) Adversarial Learning: Another critical type of methods rely on adversarial training, where the core idea is to maximally confuse domain classifier so as to learn domain-invariant representation [26]. Based on this idea, Domain Adversarial Neural Network (DANN) adversarially [27] learned a feature extractor and a domain discriminator, and [28] had a similar solution for UDA by simply fooling a GAN’s discriminator. In addition, there are some adversarial image generation methods [29], [30] that impose a class-conditioning or cycle consistency constraint to capture discriminative embeddings.

Note that our method DeepHOT belongs to one of discrepancy-based domain alignment methods that minimize the matching cost by optimal transportation to align source and target domains.

B. Optimal Transport for Domain Adaptation

Optimal Transport (OT) theory has been theoretically proven to be competitive in the context of domain adaptation [31], [32], and [1] first introduced OT into unsupervised domain adaptation. Then OT-reglab [33] and JDOT [7] further took label information into account where they assumed joint feature/label distribution of source and target domains could be estimated by a non-linear transformation. DEEPJDOT [11] extended JDOT into a deep learning-based version, and JUMBOT [14] alleviated the sampling effect of mini-batch optimal transport which allowed optimal transport for large-scale datasets in an end-to-end way. Recently, TIDOT [34] has jointly mitigated both data shift and label shift by teacher-student framework with optimal transport, and LAMDA [10] has explicitly defined the label shift and minimized it by optimal transportation on the latent space. COOK [35] further incorporated the label information into the transformation of OT from source to target domains, leading to the transportation between class-conditional source distribution and a distribution over all the target samples. COT [36] established an OT mapping between learnable clustering centers in the source and target domains in an end-to-end manner. MOT [37] extends OT to a more real-world scenario that the label space typically undergoes changes when the distribution shifts by incorporating a masking strategy for OT.
between domains. The main difference between the proposed DeepHOT method and the related work is that DeepHOT is a generalized hierarchical optimal transport formulation consisting of domain-level OT and image-level OT for domain adaptation, while the related work such as JUMBOT [14] involves domain-level OT only.

Another member of optimal transport family is hierarchical optimal transport that is a generalization of OT. It has been recently applied to different tasks such as multimodal distribution alignment [38], document representation [39], semi-supervised learning [40] and multi-view learning [41]. As far as we know, only one work HOT-DA [42] has used hierarchical optimal transport for UDA. The differences between our method and HOT-DA are two-fold. First, HOT-DA operated under the assumption that clusters in domains inherently correspond to categories, and thus utilized the knowledge of domain-cluster-relationships for UDA. However, the granularity of ‘cluster’ still relies on the global representation of images, resulting in the loss of fine-grained details in an individual image. As a consequence, hard samples, such as the ones with fine-grained discriminative characteristics, become more susceptible to misclassification. In our method, the structural information of images can enhance the discriminative features directly over samples during the adaptation process by leveraging domain-image-level relations in hierarchical optimal transport. Second, HOT-DA suffers from another limitation that it cannot be trained on large-scale datasets in an end-to-end manner since it needs to group all samples into clusters offline at first, leading to suboptimal learning of the model. To the best of our knowledge, we propose the first deep learning-based HOT model that can be trained over a mini-batch in an end-to-end way for large-scale UDA tasks.

III. THE PROPOSED DEEPHOT METHOD

A. Overview

Let \( X_s = \{ x_{s}^i \}_{i=1}^{N_s} \) be a set of data drawn from a distribution \( \mu_s \) on the source domain, associated with a set of class labels \( Y_s = \{ y_{s}^i \}_{i=1}^{N_s} \) and \( X_t = \{ x_{t}^i \}_{i=1}^{N_t} \) a data set drawn from a distribution \( \mu_t \) on the target domain. The labels for the target domain are not available during training.

In the unsupervised DA problem, Optimal Transport allows to bring source and target distributions closer by finding optimal transport plan between pairwise samples from source and target domains. Formally, OT is commonly defined by the Kantorovitch formulation [43] to search a probabilistic coupling \( \gamma \in \Pi(\mu_s, \mu_t) \) between the distribution \( \mu_s \) of source domain and \( \mu_t \) of target domain:

\[
W_d(\mu_s, \mu_t, \gamma) = \inf_{\gamma \in \Pi(\mu_s, \mu_t)} \int_{\Omega_s \times \Omega_t} c(\mathbf{x}, \mathbf{x}') \, d\gamma(\mathbf{x}, \mathbf{x}')
\]  

where \( c(\cdot) \) is the ground metric that measures the discrepancy between pairwise samples \( \mathbf{x} \) and \( \mathbf{x}' \). Here, \( \Omega_s \) (resp. \( \Omega_t \)) is a compact input measurable space related to source (resp. target) domain, and \( \Pi(\mu_s, \mu_t) \) can be understood as the joint probability measure with marginals \( \mu_s \) and \( \mu_t \).

Here we use discrete optimal transport theory to formulate both domain-level and image-level distances. The empirical distribution \( \mu_s \) for source domain is formulated as \( \mu_s = \sum_{i=1}^{N_s} p_i^s \delta_{x_i^s} \) where \( \delta_{x_i^s} \) is the Dirac function located at \( x_i^s \), and \( p_i^s \) is probability mass associated with \( x_i^s \) and belongs to the probability simplex, i.e. \( \sum_{i=1}^{N_s} p_i^s = 1 \). The empirical distribution \( \mu_t \) for target domain can be formulated in the similar way. In this case, we can adapt Kantorovitch OT formulation [43] to the discrete case.

In a discrete setting, optimal transport between the empirical distributions is formulated as:

\[
W_d(\mu_s, \mu_t, \gamma) = \min_{\gamma \in \mathcal{B}} \langle \gamma, \mathbf{C} \rangle
\]

where \( \mathcal{B} = \{ \gamma \in (\mathbb{R}^+)^{N_s \times N_t} | \gamma_1 \mathbf{1}_{N_t} = \mu_s, \gamma^\top \mathbf{1}_{N_t} = \mu_t \} \), and \( \mathbf{C} \) is a \( d \)-dimensional vector of ones. In this case, \( \mathcal{B} \) refers to the set of probabilistic couplings between the two empirical distributions. \( \langle \cdot, \cdot \rangle \) is the Frobenius dot product and \( \mathbf{C} \geq 0 \) is the cost matrix, whose term \( C(i, j) = c(x_i^s, x_j^t) \) represents the dissimilarity between pairwise samples from source and target domains. In the general case, this cost function \( c(\cdot) \) is chosen as Euclidean or cosine distance.

As previously mentioned, local features of image can help to enhance discriminative power when aligning the marginal distribution across domains. Here, when decomposing an image into a set of local regions, we can use optimal transport \( W_{\text{img}} \) to measure the distance between samples instead of Euclidean distance, which can be used as the cost function \( c(\cdot) \) of problem (2). As such, the problem (2) becomes a nested OT formulation, i.e. a general Hierarchical Optimal Transport (HOT) formulation:

\[
H(\mu_s, \mu_t) = W_d(\mu_s, \mu_t, W_{\text{img}})
\]

where correspondences are resolved at two levels: the outer level resolves domain-correspondences while the inner level resolves point-wise correspondences between pairwise samples from source and target domains. Our key insight is depicted in Figure 1, where image-level OT permits to encode correlations between local image regions to enhance discriminative power for domain alignment that is achieved by domain-level OT.

Note that domain-level (outer level) OT and image-level (inner level) OT can be any optimal transport algorithm [13], [44], [45], [46]. Next we will propose a robust and efficient solution for the generalized DeepHOT framework. In the remainder, we detail the OT formulation for both domain-level (outer level) and image-level (inner level) in Section III-B and III-C. Finally optimization process and computational complexity of HOT formulation are presented in Section III-D.

B. Domain-Level OT

The domain-level OT aims to learn the optimal mappings between the samples from source and target domains. To reduce computational complexity, the domain-level OT is trained over mini-batch using deep learning in an end-to-end way. The averaged result among mini-batches is used as a proxy for the original problem.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
To build mini-batch, we select \( n \) samples from \( X_s \) and \( X_t \). Then optimal transport over mini-batch becomes

\[
W_d^n(\mu_s^n, \mu_t^n, W_{img}) = \min_{\gamma^n \in \mathcal{B}^n} < \gamma^n, C^n >_F
\]  

(4)

where \( C^n \) is a square matrix of size \( n \) and \( \mathcal{B}^n = \{\gamma \in (\mathbb{R}^+)^{n \times n} | \gamma_{1n} = \mu_s^n, \gamma^{T}1_t = \mu_t^n \} \). The term of cost matrix \( C^n \) can be computed by optimal transport on image-level \( W_{img} \), which will be illustrated in next subsection.

In practice, however, it was proven that the solution to optimal transport over mini-batch tends to be less sparse, and causes undesired pairings between samples that shouldn’t have been matched in true OT [14]. A variant [14] of Unbalanced Optimal Transport (UOT) recently has been theoretically validated to deal with this issue by relaxing marginal constraints. Therefore, we propose to use unbalanced mini-batch optimal transport for the domain-level OT to alleviate the side effect of mini-batch sampling. Then the problem (4) becomes

\[
W_d^n(\mu_s^n, \mu_t^n, W_{img}) = \min_{\gamma^n \in \mathcal{B}^n} < \gamma^n, C^n >_F
\]

\[
+ \varepsilon KL \left( \gamma^n / \mu_s^n \otimes \mu_t^n \right) + \tau \left( D_{\phi}(\gamma_s^n \parallel \mu_s^n) + D_{\phi}(\gamma_t^n \parallel \mu_t^n) \right)
\]  

(5)

where \( \gamma^n \) is the transport plan over mini-batch, \( \gamma_s^n \) and \( \gamma_t^n \) are plan’s marginals. Here, \( D_{\phi} \) is Csiszár \( r \) divergences, \( \tau \) is the marginal penalization and \( \varepsilon \geq 0 \) is the regularization coefficient. KL is the Kullback-Leibler divergence.

This OT formulation (5) is used to replace \( W_d \) in Eq. (3), which naturally proposes a deep learning-based version of HOT method \( \mathcal{H}^h(\cdot) \), i.e. Deep Hierarchical Optimal Transport (DeepHOT):

\[
\mathcal{H}^h(\mu_s^n, \mu_t^n) = W_d^n(\mu_s^n, \mu_t^n, W_{img})
\]  

(6)

C. Image-Level OT

The image-level OT aims to align the image pairs by capturing the correlations among the local regions of images. To approximate OT in a deep learning-based manner, we first formulate an embedding function (e.g. CNN) \( g : x \rightarrow z \) that maps the input into the latent space \( Z \), and the classifier \( f : z \rightarrow y \) which maps the latent space into the label space \( Y \).

Here latent space can be modelled by any feature layer. In the unsupervised DA task, optimal transport on image-level aims to utilize local information to learn discriminative features. As such, we use the last convolution layer of a CNN to interpret this latent space that preserves the structural information of feature maps \( z \in \mathbb{R}^{H \times W \times C} \) (\( H \) and \( W \) denote the spatial size of feature maps and \( C \) refers to the channel dimension). And then \( z \) is viewed as a collection of patch features \( \left[ z^1, z^2, \ldots, z^{HW} \right] \) where each element \( z^k \in \mathbb{R}^{C \times \times (1 \leq k \leq HW)} \) is a vector. The strength of viewing the feature map as a collection of local patch feature vectors is that it allows us to use the optimal transport to mine the structural relationships among local regions of two images, in order to learn the discriminative and transferable information across categories.
In this case, an image $x_i$ can be represented with a collection of local patch features $\{x_{ij}^1, x_{ij}^2, \ldots, x_{ij}^{HW}\}$. Optimal transport can be used to capture the local correspondences between two images which are important for classification. Then we can formulate optimal transport on image-level as
\[
\nu_{\text{img}}^{i,j}(x_i^j, x_j^i) = \min_{y_{\text{img}} \in \Pi(g(x_i^j), g(x_j^i))} <y_{\text{img}}, C_{\text{img}}>
\]
where $c(x_i^j, x_j^i) = ||z_i^m - z_j^m||^2$. However, problem (7) can be solved by linear programming, which is not efficient, especially when combing with domain-level OT. And thus we introduce a learnable Sliced Wasserstein Distance [13], [41] (SWD) to approximate problem (7) for an more efficient OT formulation:
\[
\nu_{\text{img}}^{i,j}(x_i^j, x_j^i, c) = \frac{1}{M} \sum_{m=1}^{M} c(sort(\theta_m^T g(x_i^j)), sort(\theta_m^T g(x_j^i)))
\]
where $\{\theta_m\}_{m=1}^{M}$ contains $M$ linear projectors $\theta_m \in \mathbb{R}^{C \times 1}$ to allow to solve one-dimensional OT problems. In this manner, we simplify a linear programming problem (7) into equation (8) with the help of a sorting algorithm, where the sorting function sorts the elements of the vector in ascending order, such that $0 \leq k \leq HW - 1$, $(\theta_m^T z_i)^{(k)} \leq (\theta_m^T z_i)^{(k+1)}$. Then we calculate the Euclidean distance between the sorted vectors and get the averaged result across projectors.

The above term (SWD) together with Euclidean distance are jointly applied as cost function of domain-level OT (i.e. Problem (5)) to align marginal distribution. Furthermore, following [7], [11], [14], we also use the semantic constraint to handle a conditional distribution shift. Finally, the image-level OT is re-formulated as
\[
\tilde{\nu}_{\text{img}}^{i,j}(x_i^j, x_j^i, c) = \eta_1 \frac{1}{M} \sum_{m=1}^{M} ||sort(\theta_m^T g(x_i^j)) - sort(\theta_m^T g(x_j^i))||^2
\]
\[
+ \eta_2 \|\hat{g}(z_i) - \hat{g}(z_j)||^2 + \eta_3 L_{ce}(y_i, f(\hat{g}(x_i)))
\]
where $\hat{g}()$ refers to global average pooling of $g()$ and $\eta_1$, $\eta_2$, $\eta_3$ are the parameters controlling the tradeoff between these terms. In this way, $\tilde{\nu}_{\text{img}}^{i,j}()$ is used as the cost function of domain-level OT, and thus the term of cost matrix in problem (5) has $C(i, j) = \tilde{\nu}_{\text{img}}^{i,j}(x_i^j, x_j^i, c)$. As a result, DeepHOT model (6) can be interpreted as
\[
\mathcal{H}^\eta(\mu_x^i, \mu_y^i) = \nu_{\text{img}}^\eta(\mu_x^i, \mu_y^i, \tilde{\nu}_{\text{img}})
\]

To perform classification tasks, the final loss function $L$ of DeepHOT over mini-batch is integrated with the cross entropy term $L_{ce}$ on the source data:
\[
L^n = L_{ce}(x^n, y^n) + \mathcal{H}^\eta(\mu_x^n, \mu_y^n)
\]

Algorithm 1 The DeepHOT Algorithm

Input: Source data as $X_s = \{x_i^s\}_{i=1}^{N_s}$, target data as $X_t = \{x_i^t\}_{i=1}^{N_t}$. $T$ is set as the number of training iterations, and $n$ refers to the batch-size for training.

Output: The optimal parameters of embedding function $g$ and classifier $f$.

1: Randomly initialize $\{\theta_m\}_{m=1}^{M}$ of $M$ projections.
2: for $t = 1$ to $T$
   3: Sample mini-batch from source data $\{(x_i^s, y_i^s)\}_{i=1}^{n}$ and target data $\{(x_i^t)\}_{i=1}^{n}$.
   4: Fix $g$, $f$ and $\{\theta_m\}_{m=1}^{M}$.
   5: for $(x_i^s, y_i^s), (x_j^t) do$
      6: // Image-level OT
      7: Calculate $\tilde{\nu}_{\text{img}}^{i,j}(x_i^s, y_i^s, x_j^t, c)$ according to Eq. (9).
      8: $C(i,j) \leftarrow \mathcal{H}^\eta(\mu_x^i, \mu_y^i, \tilde{\nu}_{\text{img}})$
   9: end for
10: $C^n = \{C(i,j)\}_{i,j}^{n,n}$
11: Optimize $\gamma^n$ via Sinkhorn-Knopp matrix scaling algorithm [44], [45]:
12: // Domain-level OT
13: $\gamma^n \leftarrow \arg \min_{\gamma^n \in \mathbb{R}^{K \times N}} \mathcal{H}^\eta(\mu_x^i, \mu_y^i, \tilde{\nu}_{\text{img}}) \parallel Eq. (5)$
14: Fix $\gamma^n$ and calculate the loss of DeepHOT $\mathcal{H}^\eta(\mu_x^i, \mu_y^i)$ following Eq. (10) and (5).
15: Calculate the total loss function $L^n$ according to Eq. (11).
16: Update $g$, $f$ and $\{\theta_m\}_{m=1}^{M}$ by SGD [47].
17: end for

D. Optimization and Computational Complexity

We propose an efficient learning algorithm to solve the HOT problem (10) forUDA, which is a new member of Hierarchical Optimal Transport family, combining deep learning-based Unbalanced Optimal Transport on domain-level with Slice Wasserstein Distance on image-level. The details of optimization procedure are presented in Alg. 1.

In general, OT formulation on each level of HOT model implemented by Wasserstein Distance (or Earth Mover Distance) is solved by linear programming algorithms where the computational complexity is $O(N^3 \log(N))$ and $N = \max(N_x, N_t)$ on domain-level, as well as $O(K^3 \log(K))$ and $K = HW$ on image-level.

Compared with the above regular setting, our DeepHOT implementation has great advantages on computational complexity: (1) The complexity of domain-level OT over mini-batch solved by the generalized Sinkhorn-Knopp matrix scaling algorithm [44], [45] is $O(n^3)$ in which $n \ll N$ is the size of mini-batch. It allows to apply HOT model for large scale datasets. For the whole dataset, complexity is $O(\frac{N_x \times N_t}{n} \times \frac{N_x \times n}{N_t}) = O(N_x \times n)$ which is far smaller than $O(N^3 \log(N))$. (2) On image-level, the complexity of calculating the transport map is $O(K \log(K))$, where $K$ is small as the spatial size of feature maps (e.g. $K = HW$) down-sampled by CNN is limited. The complexity of calculating $M$ projections of a d- dimensional probability density functions is $O(M N^d)$ which dominates the primary complexity. Hence the overall

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
computational complexity of image-level OT formulation is \(O(MN^d)\). In this case, we can trade off the computational complexity of image-level formulation by setting the number \(M\) of projections.

IV. EXPERIMENTS

In this section, we evaluate our method on four benchmark datasets for UDA tasks, and investigate the efficacy of each component in Ablation Study. Training efficiency is also evaluated and we intuitively validate the effectiveness of proposed method by visualization technique. We rely on the POT package [54] and PyTorch tools for all experimental implementation.

A. Datasets

Office-Home [55] contains 15,500 images from four domains: Artistic images (A), ClipArt (C), Product images (P) and Real-World (R), consisting of 65 categories for each domain. We evaluate our method over all 12 adaptation tasks and report results on the test target datasets.

Digits involves three challenging UDA scenarios following the evaluation protocol of [14]: USPS [56] to MNIST [57] (U→M), MNIST to M-MNIST [27] (M→MM) and SVHN [58] to MNIST (S→M). It includes four domains: MNIST with 60,000 images, USPS with 7,291 images, SVHN with 33,212 images and M-MNIST that is colored version of MNIST with 10 classes of digits.

Office-31 [59] consists of 4652 images from 31 categories, collected from three domains including Amazon (2817 images), Webcam (795 images) and DSLR (498 images), respectively, where every pair UDA task is evaluated as in [60].

VisDA [61] is a large-scale dataset for UDA including two domains: Synthetic and Real-World. It contains 152,397 synthetic images and 55,388 real-world images that share 12 classes. Following the setting of [14], we train all methods on Synthetic and test on VisDA validation set.

B. Experiment Setup

1) Networks: For Office-Home, Office-31 and VisDA, we use ResNet-50 [48] as the generator \(g\) and two fully connected layers with dropout as the classifier \(f\), where ResNet-50 is pretrained on ImageNet, as done in [14], [27], and [49].

For Digits datasets, we employ 9 CNN layers as generator and 1 dense layer for classification, which is the same as previous works [11], [14]. Following their settings, we pretrain our model on the source domain for 10 epochs.

2) Training: We present the details of the training setting so as to reproduce the results of the proposed method, including Sampling, Data Augmentation, Optimization and Hyper-parameters.

a) Sampling: For all datasets, all experiments adopt class-balanced sampling on source domain where each class has the same number samples for a mini-batch, and random sampling is used on target domain since labels are unavailable in the target domain. These sampling strategies follow the setting of previous works [11], [14].

b) Data augmentation: For Office-Home, Office-31 and VisDA, we use the same data pre-processing where we first resize an image into the size of \(256 \times 256\) and then randomly crop an area of \(224 \times 224\). Random translation/mirror and normalization are also applied for training. When testing, we adopt the ten-crop technique [14], [49] for robust results. For Digits datasets, the inputs are solely resized and normalized without any other data augmentation. Note that these settings are standard and the same as previous works [14], [49] for fair comparisons.

c) Optimization: Following the settings of [14], for Office-Home, Office-31 and VisDA, we adopt SGD optimizer with 0.9 momentum and \(5e^{-4}\) weight decay for training, and the learning rates are set with strategy [27] by adjusting \(\lambda_p = \frac{\lambda_0}{(1+\mu)\eta}\), where \(q\) linearly changes from 0 to 1, \(\chi_0 = 0.01\), \(\mu = 10\), \(\eta = 0.75\). Note that the learning rate of classifier \(f\) is set to be 10 times that of the generator \(g\) as classifier \(f\) is trained from scratch. The batch sizes are set as 65, 31 and 72 for Office-Home, Office-31 and VisDA, respectively, and all experiments are trained for 10,000 iterations. For digits datasets, we apply Adam optimizer with a learning rate of \(2e^{-4}\). Due to the limitation of computational complexity, the batch size of DeepHOT is set as 50. We train models for 100 epochs which ensures convergence.

d) Hyper-parameters: For the domain-level OT, we follow JUMBOT [14] for the hyper-parameters settings, which include the regularization coefficient \(\varepsilon\) and the marginal penalization \(\tau\) in Eq. (5). For the image-level OT, the trade-off hyper-parameters, \(\eta_1, \eta_2, \eta_3\), are set to scale each loss term to the similar order of magnitude. For the number of projections \(M\) used in Eq. (9), we have conducted parameter sensitivity analysis in the Subsection IV-E2.

The detailed hyper-parameters setting are as follows. The primary hyper-parameters include tradeoff weights \(\eta_1, \eta_2, \eta_3\) of image-level OT in Eq. (9) which are set as follows: \(\eta_1 = 0.001, \eta_2 = 0.001, \eta_3 = 1.0\) for Office-Home, \(\eta_1 = 0.001, \eta_2 = 0.001, \eta_3 = 0.5\) for Office-31, \(\eta_1 = 0.01, \eta_2 = 0.001, \eta_3 = 1.0\) for VisDA and \(\eta_1 = 0.1, \eta_2 = 0.1, \eta_3 = 1.0\) for Digits. Another hyper-parameter \(\varepsilon\) in Eq. (5) is involved in UOT on domain-level, where \(\varepsilon\) is set at 0.01 for Office-Home, Office-31 and VisDA, and 0.1 for digits datasets. Moreover, \(\tau\) in Eq. (5) is set at 0.5 for Office-Home and Office-31, 0.3 for VisDA and 1.0 for digits according to [14]. \(M\) is the number of projections of SWD in Eq.(9). Here we empirically set it at 128 for Office-Home and Office-31, 64 for VisDA and 16 for digits.

C. Results

We compare our method against baseline and some state-of-the-art methods, including Non-OT methods (i.e., traditional UDA): DANN [27], CDAN-E [49], ALDA [50], ADDA [28], DWL [62], CaCo [60], SUDA [63] and SEEBS [51], as well as OT-based methods: DEEPJDOT [11], ROT [52], JUMBOT [14], GLOT-DR [64], COOK [35] and COT [36], that are all based on deep learning with RESNET50 [48] pretrained on ImageNet for all datasets except digits. Additionally, since COT [36] has not released their code, we just demonstrated their reported results available in their papers with the same
settings as ours. Moreover, as COT was trained on the augmented features with MMI [53] based on the model DANN [27], we also demonstrated our results on the same features augmented with MMI for fair comparison. We consider JUMBOT as baseline, as it is the most related to ours. The settings on digits of experimental pipeline follow the public implementations of JUMBOT [14] that is the closest to our method for fair comparisons. We present Accuracy score (in %) reported in JUMBOT or their own papers unless otherwise stated. We conduct each experiment three times and report the average results and error bars. The error bars for the comparison methods are also shown if they are available in their papers. Despite the varying levels of difficulty among different UDA tasks, we still report average results for reference purposes.

We should claim that the reason for not comparing DeepHOT with HOT-DA [42] is that HOT-DA can only be evaluated on a subset of the mentioned large-scale datasets due to its high computational complexity. Additionally, HOT-DA lacks explicit details regarding the sampling subsets, making it challenging to reproduce the results.

1) Results on Office-Home: As shown in Table I, (1) compared with OT-based baseline, JUMBOT, the proposed method gets the better performance on 10 out of 12 UDA tasks, achieving the best averaged performance. Among others, the highest performance gain arises in C→R with 3.1%. (2) Compared with non-OT methods, the proposed approach surpass all of them on 7 out of 12 tasks, which shows the superiority of OT-based methods. (3) Our DeepHOT achieves the best accuracy on 4 of 12 UDA scenarios on the standard ResNet50 features, making it the model with the highest number of best-performing scenarios. Compared with the second best-performing COOK, we get better performance on 8 out of 12 tasks. (4) On the augmented features, the proposed approach surpassed COT on 7 out of 12 tasks, especially on C-R and A-P scenarios with 2.5% and 3.1% performance gains. In essence, both COT and our DeepHOT incorporated more fine-grained level alignment in domain adaptation. As a result, they achieve comparable performance and outperform methods that solely focus on domain-level alignment, which further underscores the importance of fine-grained level alignment. These results confirm the superiority of our
Fig. 2. The t-SNE visualization. Representations of task C→P on Office-Home dataset (65 classes) and task U→M on Digits dataset (10 classes) are visualized for various methods in (a)-(c), where each color denotes a class.

**TABLE V**

**Ablation Study of DeepHOT on Office-Home, Where (D) Denotes Our Method, and (A), (B), (C), (E) Refer to Its Variants**

| Domain-level | Image-level | A→C | A→P | A→R | C→A | C→P | C→R | P→A | P→C | P→R | R→A | R→C | R→P | Avg |
|--------------|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| EMD(UOT)    | i2             | 47.4| 70.0| 75.4| 60.8| 66.6| 68.1| 59.7| 47.4| 76.0| 71.8| 54.9| 78.6| 64.7|
| (a)          | 35.5          | 53.4| 71.7| 77.2| 62.8| 70.2| 71.4| 60.2| 50.2| 71.1| 67.7| 56.3| 80.7| 66.6|
| (b)          | 35.5          | 53.3| 75.3| 80.8| 65.3| 74.4| 74.9| 65.4| 52.7| 79.3| 74.2| 59.9| 83.4| 70.1|
| (c)          | 35.5          | 52.0| 77.3| 81.9| 66.5| 77.4| 78.0| 63.8| 54.8| 81.5| 73.4| 60.0| 84.5| 71.3|
| (d)          | 35.5          | 37.2| 53.2| 65.7| 45.1| 50.9| 51.6| 43.4| 51.3| 63.8| 58.3| 34.9| 68.3| 50.2|
| (e)          | 35.5          | 35.5| 53.3| 65.7| 45.1| 50.9| 51.6| 43.4| 51.3| 63.8| 58.3| 34.9| 68.3| 50.2|

2) Results on Office-31: Note that results of JUMBOT are reproduced using its official code as they don’t report them on their paper. As we can see from Table II, (1) compared with OT-based baseline, JUMBOT, our DeepHOT achieves 4 of 6 UDA scenarios, leading to the best averaged performance. (2) Compared with non-OT methods, the proposed method gets the winner on all UDA tasks, which demonstrates the advantage of OT-based methods again. (3) Our approach gets the best accuracy on 3 of 6 UDA tasks.

3) Results on Digits: Table III lists performance of DeepHOT compared with a series of other state-of-the-art methods. As can be seen, DeepHOT achieves a comparable or higher score to the current state-of-the-art methods. It is considerably important that we see increases of the performance among all scenarios compared with JUMBOT that is the most related to our approach. The improvement on digits dataset is significant although it enhances not too much, since the accuracy on digits dataset itself is relatively high and is difficult to further improve.

4) Results on VisDA: Table IV gathers results on VisDA dataset, where our method achieves the best accuracy on the validation set with a margin of 1%. The reported results are convincing since the dataset is large-scale.

From the results mentioned above, the proposed method can achieve the consistent improvement on 4 standard benchmarks of various sizes and difficulties, which provides sufficient empirical evidences for validating the effectiveness of the proposed method.

**D. Ablation Study**

In ablation study, we first validate the efficacy of each component in DeepHOT for both domain-level and image-level,
and then we compare the training speed among different implements of OT formulation to prove the efficiency of DeepHOT. We follow the setting of hyper-parameters on Office-Home and each experiment in ablation study adopts the same setting.

1) Component Validation: As shown in Table V, I) for domain-level OT, we observe that UOT leads to a better performance by comparing the experiments (b) EMD (i.e., origin OT formulation) and (c). 2) For image-level OT, the effectiveness of SWD can be proven from the experiment (c) and (d) in which the performance with SWD further increases. The results of only using image-level OT are shown in the variant (e), achieving the worst performance. The reason is that it ignores the class structure in domains, causing poor classification performance, as what image-level OT does is to minimize the distance between two samples of arbitrary arbitrary class. This also validates the effectiveness of domain-level OT. Based on these consistent improvements, the effectiveness of the proposed hierarchical OT framework
composed of domain-level OT and image-level OT can be validated.

2) Training Efficiency: To show the efficiency of SWD on image-level OT, we compare the training time between original OT (a.k.a EMD) and SWD that are combined with domain-level OT (UOT), as well as single-level OT. This experiment runs on the same GeForce GTX 1080Ti with 4 GPUs. To ensure a fair comparison, we test the training time on task C → R of Office-Home from 10th iteration to 100th iteration and get the averaged results. Since the model has not stabilized yet before 10th iteration, performance results are prone to be random and thus we compare results after stable convergence. The number of projections in SWD is fixed at 128. The results are shown in Fig. 3. We see that DeepHOT with SWD + UOT (green) achieves a faster speed than that with EMD + UOT (red), which approaches single-level OT method (blue) much more. It suggests our solution is more efficient among Hierarchical OT-based methods.

E. Parameter Sensitivity Analysis

We conduct experiments to investigate the parameter sensitivity regarding the batch size for sampling and the number of projections in Sliced Wasserstein Distance.

1) The Robustness to Batch Size: In this part, we investigate the effect of batch size with/without UOT [14] for domain-level OT of DeepHOT. Here we fix all hyperparameters and replace UOT with original OT (EMD), compared with DeepHOT. We show the relative change of accuracy from batch size 65 to 130 for DeepHOT with/without UOT in Fig. 5.

We can see that, if using original OT (EMD) instead of UOT in DeepHOT, the model is considerably influenced by the batch size, in which the incremental rate of performance varies from 3% to 22% when batch size increases. However, our DeepHOT (with UOT) are robust to the batch size, where all the relative change rates of performances between different batch sizes are less than 5%. It suggests that our solution based on UOT is robust to mini-batch sampling. Therefore, we can use a small mini-batch to reduce computational complexity.

2) The Number of Projections: We conduct sensitivity analysis for the number of projections in Sliced Wasserstein Distance (Eq. (8)) of DeepHOT. Here we vary the number of projections for task C → R on Office-Home and task U → M on digits to inspect its effect on the final performances, as shown in Fig. 6. We can see that they slightly change for a series of quantities, suggesting DeepHOT is robust to the number of
projections. And thus we can choose a small but appropriate number of projections to reduce computational complexity.

F. Visualization Analysis

1) t-SNE Visualization: To present the effectiveness of adaptation process more intuitively, the representations of various methods (no-adaptation, JUMBOT and DeepHOT) are visualized by t-SNE [65] technique, as shown in Fig. 2. Here we only show the C→P scenario on Office-Home dataset and the U→M scenario on digits dataset, and more cases are presented in supplementary materials. Obviously, the source and target are not aligned without adaptation process, as presented in Fig. 2(a). Fig. 2(b) and (c) in C→P scenario, intuitively the sample points of (b) are more scattered than that of (c), while the boundaries of clusters marked with different colors for different categories in (c) are more clear. Furthermore, there are overlaps in the other red boxes of (b), while the counterparts in (c) are separated. Similarly, as shown in Fig. 2(b) of U→M scenario, there are some overlaps on the boundaries between different classes (like red and blue classes) for JUMBOT, while the proposed DeepHOT almost totally distinguishes all the classes. This suggests that our proposal is able to enhance category-discriminative representations when aligning domains.

2) Image-Level Transport Plan: To dive into how image-level OT helps the domain adaptation process, we show the transport plan of image-level OT that captures the correspondence of image pairs between source domain and target domains to enhance discrimination. During unsupervised training, the image pairs involve two cases including ‘from consistent categories’ and ‘from different categories’ of source and target domains, which are shown in Fig. 4 and 7, respectively. In each figure, we show the transport plan for three UDA tasks in which the upper/bottom row is from source/target domain respectively. The first column is original images. The second column is the corresponding image patches of the top 10 indices in descending order of SWD, where we summarize the correlation values of all projectors and get the target patches with maximal correlation value for each source patch (Sometimes there are multiple maximal values, and thus one source patch would correspond to several target patches). The following five columns are the image pairs of the first 5 indices, respectively. More examples are presented in the supplementary materials.

From the image-level transport plan with consistent category in Fig. 4, we can see that SWD is capable of transporting the local counterparts from source to target correspondingly, which verified our motivation that local regions of images contribute to enhancing discriminative features, such as the digit of alarm clocks, the flame of candles and the push-button of calculators. Even if transporting image patches between different categories, SWD is still able to transfer some local concepts to help learn in the target domain. For example, as shown in the P-C scenario of Fig. 7, SWD can capture the texture feature of the fan which is similar to the wheel hubs of the bicycle. In the C-R scenario, it also learns the features of handles between Mug and Kettle. For the knowledge-transferring between Notebook and Oven, the image-level OT learns the outline features.

V. CONCLUSION

In this paper, we tackle two important issues for unsupervised domain adaptation. First, how to learn domain-invariant yet category-discriminative representations? We propose the deep hierarchical optimal transport framework to model hierarchical structural distance among domains, which uses the domain-level OT to learn domain-invariant representations, and adopts the image-level OT to capture category-discriminative features. Second, how to build a robust and efficient DeepHOT model? We exploit unbalanced mini-batch optimal transport and sliced Wasserstein distance in domain-level and image-level OTs, respectively. The effectiveness and efficiency of the proposed DeepHOT method are verified by the extensive experiments on four benchmark datasets. It is worth noting DeepHOT is a generalized framework. We will try different implementations of DeepHOT in the future.

REFERENCES

[1] N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy, “Optimal transport for domain adaptation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 9, pp. 1853–1865, Sep. 2017.
[2] Z. Yu, J. Li, L. Zhu, K. Lu, and H. T. Shen, “Classification certainty maximization for unsupervised domain adaptation,” IEEE Trans. Circuits Syst. Video Technol., vol. 33, no. 8, pp. 4232–4243, Aug. 2023, doi: 10.1109/TCSVT.2022.3242614.
[3] Z. Mei, P. Ye, B. Li, T. Chen, J. Fan, and W. Ouyang, “DeNKD: Decoupled non-target knowledge distillation for complementing transformer-based unsupervised domain adaptation,” IEEE Trans. Circuits Syst. Video Technol., Sep. 2023, doi: 10.1109/TCSVT.2023.3315872.
[4] S. Wang et al., “Cluster alignment with target knowledge mining for unsupervised domain adaptation semantic segmentation,” IEEE Trans. Image Process., vol. 31, pp. 7403–7418, 2022.
[5] W. Wang, Z. Shen, D. Li, P. Zhong, and Y. Chen, “Probability-based graph embedding cross-domain and class discriminative feature learning for domain adaptation,” IEEE Trans. Image Process., vol. 32, pp. 72–87, 2023.
[6] Y. Bai, C. Wang, Y. Lou, J. Liu, and L. Y. Duan, “Hierarchical connectivity-centered clustering for unsupervised domain adaptation on person re-identification,” IEEE Trans. Image Process., vol. 30, pp. 6715–6729, 2021.
[7] N. Courty, R. Flamary, A. Habrard, and A. Rakotomamonjy, “Joint distribution optimal transportation for domain adaptation,” in Proc. 31st Int. Conf. Neural Inf. Process. Syst. (NIPS), Red Hook, NY, USA: Curran Associates Inc., 2017, pp. 3733–3742.
[8] R. Xu, P. Liu, L. Wang, C. Chen, and J. Wang, “Reliable weighted optimal transport for unsupervised domain adaptation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 4394–4403.
[9] M. Li, Y.-M. Zhang, Y.-W. Luo, P.-F. Ge, and C.-X. Ren, “Enhanced transport distance for unsupervised domain adaptation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 13936–13944.
[10] T. Le, T. Nguyen, N. Ho, H. Bui, and D. Phung, “LAMDA: Label matching deep domain adaptation,” in Proc. Int. Conf. Mach. Learn., 2021, pp. 6043–6054.
[11] B. B. Damodaran, B. Kellenberger, R. Flamary, D. Tuia, and N. Courty, “DeepJDOT: Deep joint distribution optimal transport for unsupervised domain adaptation,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 447–463.
[12] C. Zhang, Y. Cai, G. Lin, and C. Shen, “DeepEMD: Few-shot image classification with differentiable earth mover’s distance and structured classifiers,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2020, pp. 12203–12213.
[58] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, “Reading digits in natural images with unsupervised feature learning,” in Proc. NIPS Workshop Deep Learn. Unsupervised Feature Learn., vol. 2011, no. 5, Dec. 2011, p. 7.

[59] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, “Adapting visual category models to new domains,” in Proc. Eur. Conf. Comput. Vis. (ECCV). Cham, Switzerland: Springer, 2010, pp. 213–226.

[60] J. Huang, D. Guan, A. Xiao, S. Lu, and L. Shao, “Category contrast for unsupervised domain adaptation in visual tasks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2022, pp. 1193–1204.

[61] X. Peng, B. Usman, N. Kaushik, J. Hoffman, D. Wang, and K. Saenko, “VisDA: The visual domain adaptation challenge,” 2017, arXiv:1710.06924.

[62] N. Xiao and L. Zhang, “Dynamic weighted learning for unsupervised domain adaptation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 15242–15251.

[63] J. Zhang, J. Huang, Z. Tian, and S. Lu, “Spectral unsupervised domain adaptation for visual recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2022, pp. 9829–9840.

[64] H. Phan, T. Le, T. Phung, A. T. Bui, N. Ho, and D. Phung, “Global-local regularization via distributional robustness,” in Proc. Int. Conf. Artif. Intell. Statist., 2023, pp. 7644–7664.

[65] L. Van der Maaten and G. Hinton, “Visualizing data using t-SNE,” J. Mach. Learn. Res., vol. 9, pp. 2579–2605, 2008.

Yingxue Xu received the B.Eng. and M.Sc. degrees from the South China University of Technology. She is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology. Her main research interests include multimodal learning and medical image analysis.

Guihua Wen received the Ph.D. degree in computer science and engineering from the South China University of Technology. He is currently a Professor and a Ph.D. Supervisor with the School of Computer Science and Technology, South China University of Technology. He is also a Professor in Chief of the Data Mining and Machine Learning Laboratory, South China University of Technology. His research interests include cognitive affective computing, machine learning, and data mining.

Yang Hu (Member, IEEE) received the M.A.Eng. degree from Kunming University of Science and Technology, Kunming, China, in 2016, and the Ph.D. degree from the South China University of Technology, Guangzhou, China, in 2020. He was a Research Assistant with the University of Southampton, U.K. He is currently a Post-Doctoral Research Scientist with the University of Oxford, U.K. His research interests include neural networks, deep learning, computer vision, and biomedical information processing.

Pei Yang (Member, IEEE) received the bachelor’s degree from Shanghai Jiao Tong University and the Ph.D. degree from the South China University of Technology. He was a Research Scientist with Arizona State University. He is currently an Associate Professor with the South China University of Technology. He has published over 30 research papers on refereed journals and top-tier conference proceedings. His research interests include statistical machine learning and data mining, such as multi-task learning, transfer learning, deep learning, heterogeneous learning, meta learning, few-shot learning, and graph representation learning, with applications on healthcare, computer vision, intelligent transportation, and bioinformatics. His paper in ICDM 2016 had been awarded as the “Bests of the Conference.” He has served on the program committee for conferences, such as NeurIPS, KDD, IJCAI, AAAI, and TheWebConf.