HoMM: Higher-Order Moment Matching for Unsupervised Domain Adaptation

Chao Chen,[1,2*] **Zhihang Fu,**[2] **Zhihong Chen,**[1] **Sheng Jin,**[2]
Zhaowei Cheng,[1] **Xinyu Jin,**[1] **Xian-Sheng Hua**[2†]
[1]Zhejiang University, [2]Alibaba DAMO Academy, Alibaba Group
chench@zju.edu.cn, xiansheng.hxs@alibaba-inc.com

Abstract

Minimizing the discrepancy of feature distributions between different domains is one of the most promising directions in unsupervised domain adaptation. From the perspective of moment matching, most existing discrepancy-based methods are designed to match the second-order or lower moments, which however, have limited expression of statistical characteristic for non-Gaussian distributions. In this work, we propose a Higher-order Moment Matching (HoMM) method, and further extend the HoMM into reproducing kernel Hilbert spaces (RKHS). In particular, our proposed HoMM can perform arbitrary-order moment matching, we show that the first-order HoMM is equivalent to Maximum Mean Discrepancy (MMD) and the second-order HoMM is equivalent to Correlation Alignment (CORAL). Moreover, HoMM (order$\geq$ 3) is expected to perform fine-grained domain alignment as higher-order statistics can approximate more complex, non-Gaussian distributions. Besides, we also exploit the pseudo-labeled target samples to learn discriminative representations in the target domain, which further improves the transfer performance. Extensive experiments are conducted, showing that our proposed HoMM consistently outperforms the existing moment matching methods by a large margin. Codes are available at https://github.com/chenchao666/HoMM-Master

Introduction

Convolutional neural networks (CNNs) have shown promising results on supervised learning tasks. However, the performance of a learned model always degrades severely when dealing with data from the other domains. Considering that constantly annotating massive samples from new domains is expensive and impractical, unsupervised domain adaptation (UDA), therefore, has emerged as a new learning framework to address this problem (Csurka 2017). UDA aims to utilize full-labeled samples in source domain to annotate the completely-unlabeled target domain samples. Thanks to deep CNNs, recent advances in UDA show satisfactory performance in several computer vision tasks (Hoffman et al. 2018). Among them, most methods bridge the source and target domain by learning domain-invariant features. These dominant methods can be further divided into two categories: (1) Learning domain-invariant features by minimizing the discrepancy between different distributions (Sun and Saenko 2016; Long et al. 2017). (2) Encouraging domain confusion by a domain adversarial objectives whereby a discriminator (domain classifier) is trained to distinguish between the source and target representations. (Ganin et al. 2016; Tzeng et al. 2017; Hoffman et al. 2018).

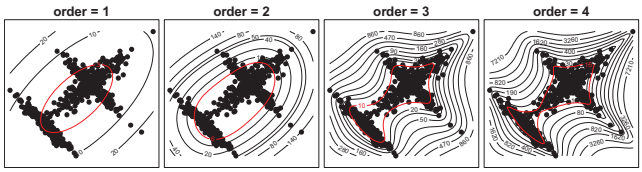


Figure 1: 300 points in $\mathbb{R}^2$ and the level sets of the moment tensor. As observed, higher-order moment tensor captures the shape of the cloud of samples more accurately.

Most existing discrepancy-based methods in UDA are based on Maximum Mean Discrepancy (MMD) (Long et al. 2017) or Correlation Alignment (CORAL) (Sun and Saenko 2016), which are designed to match the first-order (Mean) and second-order (Covariance) statistics of different distributions. However, for the real world applications, the deep features are always a complex, non-Gaussian distribution, which can not be completely characterized by its first-order or second-order statistics (Jia and Darrell 2011; Xu et al. 2016). Therefore, aligning the second-order or lower statistics only guarantees coarse-grained alignment of two distributions. To address this limitation, we propose to perform domain alignment by matching the higher-order (mainly refer to third- and fourth-order) statistics, which contain more discriminative information and can better represent the feature distribution. Inspired by (Pauwels and Lasserre 2016), Fig.1 illustrates the metrics of higher-order moment tensor, where we plot a cloud of points (consists of three different Gaussians) and the level sets of moment tensor with different order. As observed, the higher-order moment tensor characterizes the distribution more accurately.

*This work was done as a research intern in Alibaba Group. This work is supported by the opening foundation of the State Key Laboratory (No. 2014KF06) and CSC Scholarship.

†This is the corresponding author.

Our contribution can be concluded as: (1) We propose a Higher-order Moment Matching (HoMM) method to minimize the domain discrepancy, which is expected to perform fine-grained domain alignment. The HoMM integrates the MMD and CORAL into a unified framework and generalizes the first-order and second-order moment matching to higher-order moment tensor matching. Without bells and whistles, the third- and fourth-order moment matching outperform all existing discrepancy-based methods by a large margin. (2) Due to lack of labels in the target domain, we propose to learn discriminative clusters in the target domain by assigning the pseudo-labels for the reliable target samples, which also improves the transfer performance.

Related Work

Learning Domain-Invariant Features To minimize the domain discrepancy and learn domain-invariant features, various distribution discrepancy metrics have been introduced. The representative ones include Maximal Mean Discrepancy (MMD) (Gretton et al. 2012; Long et al. 2017), KL-divergence, Correlation Alignment (Sun and Saenko 2016; Chen et al. 2019) and Wasserstein distance (Lee et al. 2019). MMD was first introduced for the two-sample tests problem (Gretton et al. 2012), and is currently the most widely used metric to measure the distance between two feature distributions. Specifically, Long et al. proposed DAN (Long et al. 2015) and JAN (Long et al. 2017) which perform domain matching via multi-kernel MMD or a joint MMD criteria in multiple domain-specific layers across domains. Sun et al. proposed the correlation alignment (CORAL) (Sun, Feng, and Saenko 2016; Sun and Saenko 2016) to align the second order statistics of the source and target distributions. Some recent work also extended the CORAL into reproducing kernel Hilbert spaces (RKHS) (Zhang et al. 2018) or deployed alignment along geodesics by considering the log-Euclidean distance (Morerio, Cavazza, and Murino). Interestingly, (Li et al. 2017b) theoretically demonstrated that matching the second order statistics is equivalent to minimizing MMD with the second order polynomial kernel. Besides, the approach most relevant to our proposal is the Central Moment Discrepancy (CMD) (Zellinger et al. 2017), which matches the higher order central moments of probability distributions. Both CMD and our HoMM propose to match the higher-order statistics for domain alignment. The CMD matches the higher-order central moment while our HoMM matches the higher-order cumulant tensor. Another fruitful line of work tries to learn the domain-invariant features through adversarial training (Ganin et al. 2016; Tzeng et al. 2017). These efforts encourage domain confusion by a domain adversarial objective whereby a discriminator (domain classifier) is trained to distinguish between the source and target representations. Also, recent work performing pixel-level adaptation by image-to-image transformation (Hoffman et al. 2018) has achieved satisfactory performance and obtained much attention. In this work, we propose a higher-order moment matching method, which shows great superiority over existing domain matching methods.

Higher-order Statistics The statistics higher than first-order has been successfully used in many classical and deep learning methods (De Lathauwer, Castaing, and Cardoso 2007; Koniusz et al. 2016; Gou, Camps, and Sznaier 2017). Especially in the field of fine-grained image/video recognition, second-order statistics such as Covariance and Gaussian descriptors, have demonstrated better performance than descriptors exploiting zeroth- or first-order statistics (Li et al. 2017a; Wang, Li, and Zhang 2017). However, using second-order or lower statistical information might not be enough when the feature distribution is non-Gaussian (Gou, Camps, and Sznaier 2017). Therefore, the higher-order (mainly refer to third-order and fourth-order) statistics have been explored in many signal processing problems (Mansour and Jutten 1995; Jakubowski et al. 2002; De Lathauwer, Castaing, and Cardoso 2007; Gou, Camps, and Sznaier 2017). In the field of Blind Source Separation (BSS) (De Lathauwer, Castaing, and Cardoso 2007; Mansour and Jutten 1995), for example, the fourth-order statistics are widely used to identify different signals from mixtures. Gou et al. utilizes the third-order statistics for person ReID (Gou, Camps, and Sznaier 2017). Xu et al. exploits the third-order cumulant for blind image quality assessment (Xu et al. 2016). In (Jakubowski et al. 2002; Koniusz et al. 2016), the authors exploit higher-order statistics for image recognition and detection. Matching the second order statistics can not ensure two distributions inseparable, just as using the second order statistics can not identifies different signals from mixtures (De Lathauwer, Castaing, and Cardoso 2007). That's why we explore higher-order moment tensor for domain matching.

Method

In this work, we consider the unsupervised domain adaptation problem. Let $\mathcal{D}_s = \{\boldsymbol{x}_s^i, y_s^i\}_{i=1}^{n_s}$ denotes the source domain with n_s labeled samples and $\mathcal{D}_t = \{\boldsymbol{x}_t^i\}_{i=1}^{n_t}$ denotes the target domain with n_t unlabeled samples. Given $\mathcal{D}_s \cup \mathcal{D}_t$, we aim to train a cross-domain CNN classifier $f_{\boldsymbol{\theta}}(\boldsymbol{x})$ which can minimize the target risks $\epsilon_t = \mathbb{E}_{\boldsymbol{x} \in \mathcal{D}_t}[f_{\boldsymbol{\theta}}(\boldsymbol{x}) \neq \boldsymbol{y}_t]$. Here $f_{\boldsymbol{\theta}}(\boldsymbol{x})$ denotes the outputs of the deep neural networks, $\boldsymbol{\theta}$ denotes the model parameter to be learned. Following (Long et al. 2017; Chen et al. 2019), we adopt the two-stream CNNs architecture for unsupervised deep domain adaptation. As shown in Fig. 2, the two streams share the same parameters (tied weights), operating the source and target domain samples respectively. And we perform the domain alignment in the last full-connected (FC) layer (Sun and Saenko 2016; Chen et al. 2019). According to the theory proposed by Ben-David et al. (Ben-David et al. 2010), a basic domain adaptation model should, at least, involve the source domain loss and the domain discrepancy loss, i.e.,

$$\mathcal{L}(\boldsymbol{\theta}|\mathbf{X}_s, \mathbf{Y}_s, \mathbf{X}_t) = \mathcal{L}_s + \lambda_d \mathcal{L}_d \tag{1}$$

$$\mathcal{L}_s = \frac{1}{n_s} \sum_{i=1}^{n_s} J(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i^s), \boldsymbol{y}_i^s) \tag{2}$$

where $\mathcal{L}_s$ represents the classification loss in the source domain, $J(\cdot, \cdot)$ represents the cross-entropy loss function. $\mathcal{L}_d$ represents the domain discrepancy loss and λ_d is the trade-off parameter. As aforementioned, most of existing

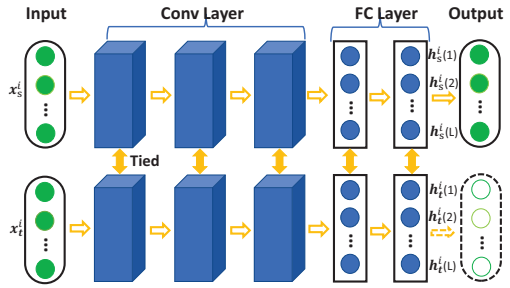


Figure 2: Two-stream CNNs with shared parameters are adopted for unsupervised deep domain adaptation.

discrepancy-based methods are designed to minimize distance of the second-order or lower statistics between different domains. In this work, we propose a higher-order moment matching method, which matches the higher-order statistics of different domains.

Higher-order Moment Matching

To perform fine-grained domain alignment, we propose a higher-order moment matching as

$$\mathcal{L}_d = \frac{1}{L^p}\|\frac{1}{n_s}\sum_{i=1}^{n_s}\phi_{\boldsymbol{\theta}}(\boldsymbol{x}_s^i)^{\otimes p} - \frac{1}{n_t}\sum_{i=1}^{n_t}\phi_{\boldsymbol{\theta}}(\boldsymbol{x}_t^i)^{\otimes p}\|_F^2 \quad (3)$$

where $n_s = n_t = b$ (b is the batch size) during the training process. $\phi_{\boldsymbol{\theta}}(\boldsymbol{x})$ denotes the activation outputs of the adapted layer. As illustrated in Fig. 2, $\boldsymbol{h}^i = \phi_{\boldsymbol{\theta}}(\boldsymbol{x}^i) = [\boldsymbol{h}^i(1), \boldsymbol{h}^i(2), \cdots, \boldsymbol{h}^i(L)] \in \mathbb{R}^L$ denotes the activation outputs of the i-th sample, L is the number of hidden neurons in the adapted layer. Here, $\boldsymbol{u}^{\otimes p}$ denotes the p-level tensor power of the vector $\boldsymbol{u} \in \mathbb{R}^c$. That is

$$\boldsymbol{u}^{\otimes p} = \underbrace{\boldsymbol{u} \otimes \boldsymbol{u} \cdots \otimes \boldsymbol{u}}_{\text{p times}} \in \mathbb{R}^{c^p} \quad (4)$$

where $\otimes$ denotes the outer product (or tensor product). We have $\boldsymbol{u}^{\otimes 0} = 1$, $\boldsymbol{u}^{\otimes 1} = \boldsymbol{u}$ and $\boldsymbol{u}^{\otimes 2} = \boldsymbol{u} \otimes \boldsymbol{u}$. The 2-level tensor product $\boldsymbol{u}^{\otimes 2} \in \mathbb{R}^{c^2}$ defined as

$$\boldsymbol{u}^{\otimes 2} = \boldsymbol{u}^T\boldsymbol{u} = \begin{bmatrix} u_1u_1 & u_1u_2 & \cdots & u_1u_c \\ u_2u_1 & u_2u_2 & \cdots & u_2u_c \\ \vdots & \vdots & \ddots & \vdots \\ u_cu_1 & u_cu_2 & \cdots & u_cu_c \end{bmatrix} \quad (5)$$

when $p \geq 3$, $\mathbf{T} = \boldsymbol{u}^{\otimes p}$ is a p-level tensor with $\mathbf{T}[i, j, \cdots, k] = u_iu_j \cdots u_k$.

Instantiations According to Eq. (3), when $p = 1$, the first-order moment matching is equivalent to the linear MMD (Tzeng et al. 2014), which is expressed as

$$\mathcal{L}_d = \frac{1}{L}\|\frac{1}{b}\sum_{i=1}^{b}\boldsymbol{h}_s^i - \frac{1}{b}\sum_{i=1}^{b}\boldsymbol{h}_t^i\|_F^2 \quad (6)$$

When $p = 2$, the second-order HoMM is formulated as,

$$\begin{aligned}\mathcal{L}_d =& \frac{1}{L^2}\|\frac{1}{b}\sum_{i=1}^{b}\boldsymbol{h}_s^{iT}\boldsymbol{h}_s^i - \frac{1}{b}\sum_{i=1}^{b}\boldsymbol{h}_t^{iT}\boldsymbol{h}_t^i\|_F^2 \\ =& \frac{1}{b^2L^2}\|\boldsymbol{G}(\boldsymbol{h}_s) - \boldsymbol{G}(\boldsymbol{h}_t)\|_F^2\end{aligned} \quad (7)$$

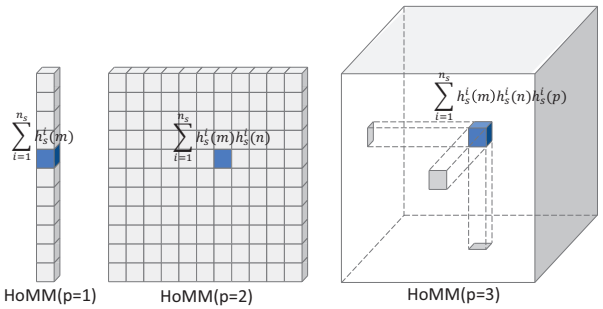


Figure 3: An illustration of first-order, second-order and third-order moments in the source domain. HoMM matches the higher-order ($p \geq 3$) moment across different domains.

where $\boldsymbol{G}(\boldsymbol{h}) = \boldsymbol{H}^T\boldsymbol{H} \in \mathbb{R}^{L\times L}$ is the Gram matrix, $\boldsymbol{H} = [\boldsymbol{h}^1; \boldsymbol{h}^2; \cdots, \boldsymbol{h}^b] \in \mathbb{R}^{b\times L}$, b is the batch size. Therefore, the second-order HoMM is equivalent to the Gram matrix matching, which is also widely used for cross-domain matching in neural style transfer (Gatys, Ecker, and Bethge 2016; Li et al. 2017b) and knowledge distillation (Yim et al. 2017). Li et al. (Li et al. 2017b) theoretically demonstrate that matching the Gram matrix of feature maps is equivalent to minimize the MMD with the second order polynomial kernel. Besides, when the activation outputs are normalized by subtracting the mean value, the centralized Gram matrix turns into the Covariance matrix. In this respect, the second-order HoMM is also equivalent to CORAL, which matches the Covariance matrix for domain matching.

As illustrated in Fig. 3, in addition to the first-order moment matching (e.g. MMD) and the second-order moment matching (e.g. CORAL and Gram matrix matching), our proposed HoMM can also perform higher-order moment matching when $p \geq 3$. Since higher-order statistics can characterize the non-Gaussian distributions better, applying higher-order moment matching is expected to perform fine-grained domain alignment. However, the space complexity of calculating the higher-order tensor $\boldsymbol{u}^{\otimes p}$ ($p \geq 3$) reaches $\mathcal{O}(L^p)$, which makes the higher-order moment matching infeasible in many real-world applications. Adding bottleneck layers to shrink the length of adaptive layer does not even solve the problem. When $L = 128$, for example, the dimension of a third-order tensor still reaches $\mathcal{O}(10^6)$, and the dimension of a fourth-order tensor reaches $\mathcal{O}(10^8)$, which is absolutely computational-unfriendly. To address this problem, we propose two practical techniques to perform the compact tensor matching.

Group Moment Matching. As the space complexity grows exponentially with the number of neurons L, one practical approach is to divide the hidden neurons in the adapted layer into n_g groups, with each group $\lfloor L/n_g \rfloor$ neurons. Then we can calculate and match the high-level tensor in each group respectively. That is,

$$\mathcal{L}_d = \frac{1}{b^2\lfloor L/n_g \rfloor^p}\sum_{k=1}^{n_g}\|\sum_{i=1}^{b}\boldsymbol{h}_{s,k}^{i\otimes p} - \sum_{i=1}^{b}\boldsymbol{h}_{t,k}^{i\otimes p}\|_F^2 \quad (8)$$

where $\boldsymbol{h}_{:,k}^i \in \mathbb{R}^{\lfloor L/n_g \rfloor}$ is the activation outputs of k-th

group. In this way, the space complexity can be reduced from $\mathcal{O}(L^p)$ to $\mathcal{O}(n_g \cdot \lfloor L/n_g \rfloor^p)$. In practice, $\lfloor L/n_g \rfloor \geq 25$ need to be satisfied to ensure satisfactory performance.

Random Sampling Matching. The group moment matching can work well when $p = 3$ and $p = 4$, but it tends to fail when $p \geq 5$. Therefore, we also propose a random sampling matching strategy which is able to perform arbitrary-order moment matching. Instead of calculating and matching two high-dimensional tensors, we randomly select N values in the high-level tensor, and only calculate and align these N values in the source and target domains. In this respect, the p-order moment matching with random sampling strategy can be formulated as,

$$\mathcal{L}_d = \frac{1}{b^2 N} \sum_{k=1}^{N} [\sum_{i=1}^{b} \prod_{j=rnd[k,1]}^{rnd[k,p]} \boldsymbol{h}_s^i(j) - \sum_{i=1}^{b} \prod_{j=rnd[k,1]}^{rnd[k,p]} \boldsymbol{h}_t^i(j)]^2 \tag{9}$$

where $rnd \in \mathbb{R}^{N \times p}$ denotes the randomly generated position index matrix, $rnd[k,j] \in \{1, 2, 3, \cdots, L\}$. Therefore, $\prod_{j=rnd[k,1]}^{rnd[k,p]} \boldsymbol{h}_s^i(j)$ denotes a randomly sampled value in the p-level tensor $\boldsymbol{h}_s^{i \otimes p}$. With the random sampling strategy, we can perform arbitrarily-order moment matching, and the space complexity can be reduced from $\mathcal{O}(L^p)$ to $\mathcal{O}(N)$. In practice, the model can achieve very competitive results even $N = 1000$.

Higher-order Moment Matching in RKHS

Similar to the KMMD (Long et al. 2017), we generalize the higher-order moment matching into reproducing kernel Hilbert spaces (RKHS) as well. That is,

$$\mathcal{L}_d = \frac{1}{L^p} \| \frac{1}{b} \sum_{i=1}^{b} \psi(\boldsymbol{h}_s^{i \otimes p}) - \frac{1}{b} \sum_{i=1}^{b} \psi(\boldsymbol{h}_t^{i \otimes p}) \|_F^2 \tag{10}$$

where $\psi(\boldsymbol{h}_s^{i \otimes p})$ denotes the feature representation of i-th source sample in RKHS. According to the proposed random sampling strategy, $\boldsymbol{h}_s^{i \otimes p}$ and $\boldsymbol{h}_t^{i \otimes p}$ can be approximated by two N-dimensional vectors $\boldsymbol{h}_{sp}^i \in \mathbb{R}^N$ and $\boldsymbol{h}_{tp}^i \in \mathbb{R}^N$, where $\boldsymbol{h}_{sp}^i(k) = \prod_{j=rnd[k,1]}^{rnd[k,p]} \boldsymbol{h}_s^i(j)$, $k = 1, \cdots, N$. In this respect, the domain matching loss can be formulated as

$$\begin{aligned} \mathcal{L}_d = & \frac{1}{b^2} \sum_{i=1}^{b} \sum_{j=1}^{b} k(\boldsymbol{h}_{sp}^i, \boldsymbol{h}_{sp}^j) - \frac{2}{b^2} \sum_{i=1}^{b} \sum_{j=1}^{b} k(\boldsymbol{h}_{sp}^i, \boldsymbol{h}_{tp}^j) \\ & + \frac{1}{b^2} \sum_{i=1}^{b} \sum_{j=1}^{b} k(\boldsymbol{h}_{tp}^i, \boldsymbol{h}_{tp}^j) \end{aligned} \tag{11}$$

where $k(\boldsymbol{x}, \boldsymbol{y}) = \exp(-\gamma \|\boldsymbol{x} - \boldsymbol{y}\|_2)$ is the RBF kernel function. Particularly, when $p = 1$, the kernelized HoMM (KHoMM) is equivalent to the KMMD.

Discriminative Clustering

When the target domain features are well aligned with the source domain features, the unsupervised domain adaptation turns into the semi-supervised classification problem, where the discriminative clustering in the unlabeled data is always encouraged (Grandvalet and Bengio 2005; Xie, Girshick, and Farhadi 2016). There have been a lot of work trying to learn the discriminative clusters in the target domain (Shu et al. 2018; Morerio, Cavazza, and Murino), most of which minimize the conditional entropy to ensure the decision boundaries do not cross high-density data regions,

$$\mathcal{L}_{ent} = \frac{1}{n_t} \sum_{i=1}^{n_t} \sum_{j=1}^{c} -p_j \log p_j \tag{12}$$

where c is the number of classes, p_j is the softmax output of j-th node in the output layer. We find that the entropy regularization works well when the target domain has high test accuracy, but it helps little or even downgrades the accuracy when the test accuracy is unsatisfactory. The reason can be drawn that the classifier may be misled as a result of entropy regularization enforcing over-confident probability on some misclassified samples. Instead of clustering in the output layer by minimizing the conditional entropy, we propose to cluster in the shared feature space. First, we pick up highly confident predicted target samples whose predicted probabilities are greater than a given threshold η, and assign pseudo-labels to these reliable samples. Then, we penalize the distance of each pseudo-labeled sample to its class center. The discriminative clustering loss can be given as

$$\mathcal{L}_{dc} = \frac{1}{n_t} \sum_{i=1}^{n_t} \|\boldsymbol{h}_t^i - \boldsymbol{c}_{\hat{y}_t^i}\|_2^2 \tag{13}$$

where $\hat{y}_t^i$ is the assigned pseudo-label of $\boldsymbol{x}_t^i$, $\boldsymbol{c}_{\hat{y}_t^i} \in \mathbb{R}^L$ denotes its estimated class center. As we perform update based on mini-batch, the centers can not be accurately estimated by a small size of samples. Therefore, we update the class center in each iteration via moving average method. That is,

$$\mathbf{c}_j^{t+1} = \alpha \mathbf{c}_j^t + (1 - \alpha) \Delta \mathbf{c}_j^t \tag{14}$$

$$\Delta \mathbf{c}_j = \frac{\sum_{i=1}^{b} \delta(\hat{y}_i = j) \boldsymbol{h}_t^i}{1 + \sum_{i=1}^{b} \delta(\hat{y}_t^i = j)} \tag{15}$$

where α is the learning rate of the center. $\boldsymbol{c}_j^t$ is the class center of j-th class in t-th iteration. $\delta(\hat{y}_t^i = j) = 1$ if $\boldsymbol{x}_t^i$ belongs to j-th class, otherwise it should be 0.

Full Objective Function

Based on the aforementioned analysis, to enable effective unsupervised domain adaptation, we propose a holistic approach with an integration of (1) source domain loss minimization, (2) domain alignment with the higher-order moment matching and (3) discriminative clustering in the target domain. The full objective function is as follows,

$$\mathcal{L} = \mathcal{L}_s + \lambda_d \mathcal{L}_d + \lambda_{dc} \mathcal{L}_{dc} \tag{16}$$

where $\mathcal{L}_s$ is the classification loss in the source domain, $\mathcal{L}_d$ is the domain discrepancy loss measured by the higher-order moment matching, and $\mathcal{L}_{dc}$ denotes the discriminative clustering loss. Note that in order to obtain reliable pseudo-labels for discriminative clustering, we set $\lambda_{dc} = 0$ during the initial iterations, and enable the clustering loss $\mathcal{L}_{dc}$ after the total loss tends to be stable.

Experiments

Setup

Dataset. We conduct experiments on three public visual adaptation datasets: digits recognition dataset, Office-31 dataset, and Office-Home dataset. The digits recognition dataset includes four widely used benchmarks: MNIST, USPS, Street View House Numbers (SVHN), and SYN (synthetic digits dataset). We evaluate our proposal across three typical transfer tasks, including: **SVHN→MNIST**, **USPS→MNIST** and **SYN→MNIST**. The details of this dataset can be seen in (Chen et al. 2019). Office-31 is another commonly used dataset for real-world domain adaptation scenario, which contains 31 categories acquired from the office environment in three distinct image domains: **A**mazon (product images download from amazon.com), **W**ebcam (low-resolution images taken by a webcam) and **D**slr (high-resolution images taken by a digital SLR camera). The office-31 dataset contains 4110 images in total, with 2817 images in **A** domain, 795 images in **W** domain and 498 images in **D** domain. We evaluate our method on all the six transfer tasks as (Long et al. 2017). The Office-Home dataset (Venkateswara et al. 2017) is a more challenging dataset for domain adaptation, which consists of images from four different domains: Artistic images (**A**), Clip Art images (**C**), Product images (**P**) and Real-world images (**R**). The dataset contains around 15500 images in total from 65 object categories in office and home scenes.

Baseline Methods. We compare our proposal with the following methods, which are most related to our work: Deep Domain Confusion (**DDC**) (Tzeng et al. 2014), Deep Adaptation Network (**DAN**) (Long et al. 2015), Deep Correlation Alignment (**CORAL**) (Sun and Saenko 2016), Domain-adversarial Neural Network (**DANN**) (Ganin et al. 2016), Adversarial Discriminative Domain Adaptation (**ADDA**) (Tzeng et al. 2017), Joint Adaptation Network (**JAN**) (Long et al. 2017), Central Moment Discrepancy (**CMD**) (Zellinger et al. 2017) Cycle-consistent Adversarial Domain Adaptation (**CyCADA**) (Hoffman et al. 2018), Joint Discriminative feature Learning and Domain Adaptation (**JDDA**) (Chen et al. 2019). Specifically, DDC, DAN, JAN, CORAL and CMD are representative moment matching based methods, while DANN, ADDA and CyCADA are representative adversarial training based methods.

Implementation Details. In our experiments on digits recognition dataset, we utilize the modified LeNet whereby a bottleneck layer with 90 hidden neurons is added before the output layer. Since the image size is different across different domains, we resize all the images to 32×32 and convert the RGB images to grayscale. For the experiments on Office-31, we use ResNet-50 pretrained on ImageNet as our backbone networks. And we add a bottleneck layer with 180 hidden nodes before the output layer for domain matching. It is worth noting that the **relu** activation function can not be applied to the adapted layer, as relu activation function will make most of the values in the high-level tensor $\boldsymbol{h}_s^{i\otimes p}$ to be zero, which will make our HoMM fail. Therefore, we adopt **tanh** activation function in the adapted layer. Due to the small samples size of Office-31 and Office-Home datasets, we only update the weights of the full-connected layers (fc) as well as the final block (scale5/block3), and fix other parameters pretrained on ImageNet. Follow the standard protocol of (Long et al. 2017), we use all the labeled source domain samples and all the unlabeled target domain samples for training. All the comparison methods are based on the same CNN architecture for a fair comparison. For DDC, DAN, CORAL and CMD, we embed the official implementation code into our model and carefully select the trade-off parameters to get the best results. When training with ADDA, our adversarial discriminator consists of 3 fully connected layers: two layers with 500 hidden units followed by the final discriminator output. For other compared methods, we report the results in the original paper directly.

Parameters. Our model is trained with Adam Optimizer based on Tensorflow. Regarding the optimal hyper-parameters, they are determined by applying multiple experiments using grid search strategy. The optimal hyper-parameters may be distinct across different transfer tasks. Specifically, the trade-off parameters are selected from $\lambda_d = \{1, 10, 10^2, \cdots, 10^8\}$, $\lambda_{dc} \in \{0.01, 0.03, 0.1, 0.3, 1.0\}$. For the digits recognition tasks, the hyper-parameter λ_d is set to 10^4 for third-order HoMM and set to 10^7 for fourth-order HoMM. For the experiments on Office-31 and Office-Home, λ_d is set to 300 for the third-order HoMM and set to 3000 for the fourth-order HoMM. Besides, the hyper-parameter γ in RBF kernel is set to 1e-4 across the experiments, the learning rate of the centers is set to $\alpha = 0.5$ for digits dataset and set to $\alpha = 0.3$ for Office-31 and Office-Home dataset. The threshold η of the predicted probability is chosen from $\{0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95\}$, and the best results are reported. The parameter sensitivity can be seen in Fig. 5.

Table 1: Test accuracy (%) on digits recognition dataset for unsupervised domain adaptation based on modified LeNet

Method	SN→MT	US→MT	SYN→MT	Avg
Source Only	67.3±0.3	66.4±0.4	89.7±0.2	74.5
DDC	71.9±0.4	75.8±0.3	89.9±0.2	79.2
DAN	79.5±0.3	89.8±0.2	75.2±0.1	81.5
DANN	70.6±0.2	76.6±0.3	90.2±0.2	79.1
CMD	86.5±0.3	86.3±0.4	96.1±0.2	89.6
ADDA	72.3±0.2	92.1±0.2	96.3±0.4	86.9
CORAL	89.5±0.2	96.5±0.3	96.5±0.2	94.2
CyCADA	92.8±0.1	97.4±0.3	97.5±0.1	95.9
JDDA	94.2 ±0.1	96.7±0.1	97.7±0.0	96.2
HoMM(p=3)	96.5±0.2	97.8±0.0	97.6±0.1	97.3
HoMM(p=4)	95.7±0.2	97.6±0.0	97.6±0.0	96.9
KHoMM(p=3)	97.2±0.1	97.9±0.1	98.2±0.1	97.8
Full	**98.8**±0.1	**99.0**±0.1	**99.0**±0.0	**98.9**
KHoMM+$\mathcal{L}_{ent}$	**99.0**±0.0	**99.1**±0.1	**99.2**±0.0	**99.1**

We denote SVHN, MNIST, USPS as SN, MT and US respectively.

Experimental results

Digits Dataset For the experiments on digits recognition dataset, we set the batch size as 128 for each domain and

Table 2: Test accuracy (%) on Office-31 dataset for unsupervised domain adaptation based on ResNet-50

Method	A→W	D→W	W→D	A→D	D→A	W→A	Avg
Source Only	73.1±0.2	93.2±0.2	98.8±0.1	72.6±0.2	55.8±0.1	56.4±0.3	75.0
DDC (Tzeng et al. 2014)	74.4±0.3	94.0±0.1	98.2±0.1	74.6±0.4	56.4±0.1	56.9±0.1	75.8
DAN (Long et al. 2015)	78.3±0.3	95.2±0.2	99.0±0.1	75.2±0.2	58.9±0.2	64.2±0.3	78.5
DANN (Ganin et al. 2016)	73.6±0.3	94.5±0.1	99.5±0.1	74.4±0.5	57.2±0.1	60.8±0.2	76.7
CORAL (Sun and Saenko 2016)	79.3±0.3	94.3±0.2	99.4±0.2	74.8±0.1	56.4±0.2	63.4±0.2	78.0
JAN (Long et al. 2017)	85.4±0.3	97.4±0.2	99.8±0.2	84.7±0.4	68.6±0.3	70.0±0.4	84.3
CMD (Zellinger et al. 2017)	76.9±0.4	94.6±0.3	99.2±0.2	75.4±0.4	56.8±0.1	61.9±0.2	77.5
CyCADA (Hoffman et al. 2018)	82.2±0.3	94.6±0.2	99.7±0.1	78.7±0.1	60.5±0.2	67.8±0.2	80.6
JDDA(Chen et al. 2019)	82.6±0.4	95.2±0.2	99.7±0.0	79.8±0.1	57.4±0.0	66.7±0.2	80.2
HoMM(p=3)	87.6±0.2	96.3±0.1	99.8±0.0	83.9±0.2	66.5±0.1	68.5±0.3	83.7
HoMM(p=4)	89.8±0.3	97.1±0.1	100.0±0.0	86.6±0.1	69.6±0.3	69.7±0.3	85.5
KHoMM(p=4)	90.5±0.2	98.3±0.1	100.0±0.0	87.7±0.2	70.4±0.2	70.3±0.2	86.2
Full	**91.7**±0.3	**98.8**±0.0	**100.0**±0.0	**89.1**±0.3	**71.2**±0.2	**70.6**±0.3	**86.9**
KHoMM+$\mathcal{L}_{ent}$	90.8±0.1	**99.3**±0.1	**100.0**±0.0	87.9±0.2	69.3±0.3	69.5±0.4	86.1

Table 3: Test accuracy (%) on Office-Home dataset for unsupervised domain adaptation based on ResNet-50

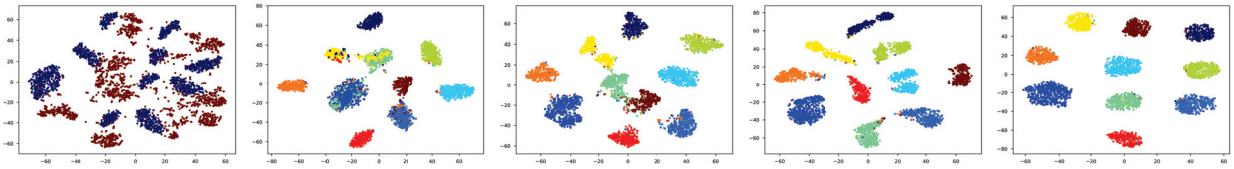
Method	A→P	A→R	C→R	P→R	R→P
Source Only	50.0	58.0	46.2	60.4	59.5
DDC	54.9	61.3	50.5	64.1	65.9
DAN	57.0	67.9	60.4	67.7	74.3
DANN	59.3	70.1	60.9	68.5	76.8
CORAL	58.6	65.4	59.8	68.3	74.7
JAN	61.2	68.9	61.0	70.3	76.8
HoMM(p=3)	60.7	68.3	61.4	69.2	76.7
HoMM(p=4)	63.5	70.2	64.6	72.6	79.3
KHoMM(p=4)	63.9	70.5	65.3	73.3	79.8
Full	**64.7**	**71.8**	**66.1**	**74.5**	**81.2**
KHoMM+$\mathcal{L}_{ent}$	64.2	70.1	65.5	73.2	80.1

set the learning rate as 1e-4 throughout the experiments. Table 1 shows the adaptation performance on three typical transfer tasks based on the modified LeNet. As can be seen, our proposed HoMM yields notable improvement over the comparison methods on all of the transfer tasks. In particular, our method improves the adaption performance significantly in the hard transfer tasks SVHN→MNIST. Without bells and whistles, the proposed third-order KHoMM achieve 97.2% accuracy, improving the second-order moment matching (CORAL) by +8%. Besides, the results also indicate that the third-order HoMM outperforms the fourth-order HoMM and slightly underperforms the KHoMM.

Office-31 Table 2 lists the test accuracies on Office-31 dataset. We set the batchsize as 70 for each domain. The learning rate of the fc layer parameters is set as 3e-4 and the learning rate of the conv layer (scale5/block3) parameters is set as 3e-5. As observed, the fourth-order HoMM outperforms the third-order HoMM and achieves the best results among all the moment-matching based methods. Besides, it is worth noting that the fourth-order HoMM outperforms the second-order statistics matching (CORAL) by more than 10% on several representative transfer tasks A→W, A→D and D→A, which demonstrates the merits of our proposed higher-order moment matching.

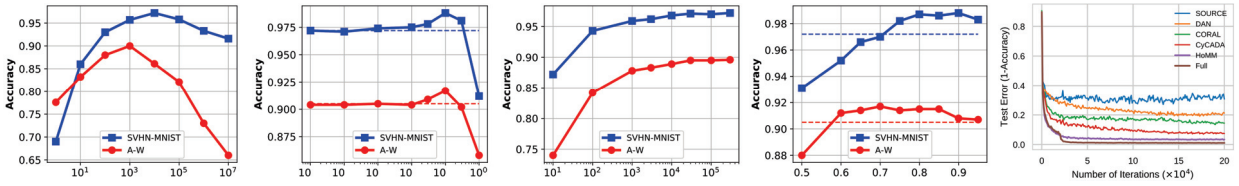
Office-Home Table 3 gives the results on the challenged Office-Home dataset. The parameter settings are the same as in Office-31. We only evaluate our method on 5 out of 12 representative transfer tasks due to the space limitation. As we can see, on all the five transfer tasks, the HoMM outperforms the DAN, CORAL, DANN by a large margin and also outperforms the JAN by 3%-5%. Note that the experimental results of the compared methods are reported from (Wang et al. 2019) directly.

The results in Table 1, Table 2 and Table 3 reveal several interesting observations: (1) All the domain adaptation methods outperform the source only model by a large margin, which demonstrates that minimizing the domain discrepancy contributes to learning more transferable representations. (2) Our proposed HoMM significantly outperforms the discrepancy-based methods (DDC, CORAL, CMD), and the adversarial training based methods (DANN, ADDA and CyCADA), which reveals the advantages of matching the higher-order statistics for domain adaptation. (3) The JAN performs slightly better than the third-order HoMM on several transfer tasks, but it's always not as good as the fourth-order HoMM in spite of aligning the joint distributions of multiple domain-specific layers across domains. The performance of our HoMM will be improved as well if we utilize such a strategy. (4) The kernelized HoMM (KHoMM) consistently outperforms the plain HoMM, but the improvement seems limited. We believe the reason is that, the higher-order statistics are originally the high-dimensional features, which conceals the advantages of embedding the features into RKHS. (5) In all transfer tasks, the performance increases consistently by employing the discriminative clustering in target domain. In contrast, entropy regularization improves the transfer performance when the test accuracy is high, but it helps little or even downgrades the performance



(a) Source Only (b) KMMD (c) CORAL (d) HoMM(p=3) (e) Full Loss

Figure 4: 2D visualization of the deep features generated by different model on SVHN→MNIST. Red and green points in (a) denote the source and target domain samples respectively, while each color in (b)-(e) represents different categories.



(a) λ_d (b) λ_{dc} (c) N (d) η (e) Convergence

Figure 5: Analysis of parameter sensitivity (a)-(d) and convergence analysis (e). The dash line in (b) and (d) indicate the performance of HoMM without the clustering loss $\mathcal{L}_{dc}$

when the test accuracy is not that confident.

Table 4: Test accuracy (%) comparison of different-order moment matching on three transfer tasks

order	1	2	3	4	5	6	10
SN→MT	71.9	89.5	**96.5**	95.7	94.8	91.5	58.6
A→W	74.4	79.3	87.6	**89.8**	86.6	85.3	80.2
A→P	54.9	58.6	60.7	**63.5**	60.9	58.2	57.3

We denote SVHN and MNIST as SN and MT respectively.

Analysis

Feature Visualization We utilize t-SNE to visualize the deep features on the tasks SVHN→MNIST by ResNet-50, KMMD, CORAL, HoMM(p=3) and the Full Loss model. As shown in Fig. 4, the feature distributions of the source only model in (a) suggests that the domain shift between SVNH and MNIST is significant, which demonstrates the necessity of performing domain adaptation. Besides, the global distributions of the source and target samples are well aligned with the KMMD (b) and CORAL (c), but there are still many samples being misclassified. With our proposed HoMM, the source and target samples are aligned better and categories are discriminated better as well.

First/Second-order versus Higher-order We also provide the performance of different order moment matching on three typical transfer tasks. As shown in table 4, the order is chosen from $p \in \{1, 2, 3, 4, 5, 6, 10\}$. The results show that the third-order and fourth-order moment matching significantly outperform the other order moment matching. When $p \leq 3$, the higher the order, the higher the accuracy. When $p \geq 4$, the accuracy will decrease as the order increases. Regarding why the fifth-order and above perform worse than the fourth-order, one reason we believe is that the fifth-order and above moments can't be accurately estimated due to the small sample size problem (Raudys and Jain 1991).

Parameter Sensitivity and Convergence We conduct empirical parameter sensitivity on SVHN→MNIST and A→W in Fig. 5(a)-(d). The evaluated parameters include two trade-off parameters λ_c, λ_{dc}, the number of selected values in Random Sampling Matching N, and the threshold η of the predicted probability. As we can see, our model is quite sensitive to the change of λ_{dc} and the bellshaped curve illustrates the regularization effect of λ_d and λ_{dc}. The convergence performance is provided in Fig. 5(e), which shows that our proposal converges fastest compared with other methods. It is worth noting that, the test error of the Full Loss model has a obvious mutation at the 2.0×10^4 iteration where we enable the clustering loss $\mathcal{L}_{dc}$, which also demonstrates the effectiveness of the proposed discriminative clustering loss.

Conclusion

Minimizing statistic distance between source and target distributions is an important line of work for domain adaptation. Unlike previous methods that utilize the second-order or lower statistics for domain alignment, this paper exploits the higher-order statistics for domain alignment. Specifically, a higher-order moment matching is presented, which integrates the MMD and CORAL into a unified framework and generalizes the existing first- and second-order moment matching to arbitrary-order moment matching. We experimentally demonstrate that the third- and fourth-order moment matching significantly outperform the existing moment matching methods. Besides, we also extend the HoMM into RKHS and learn the discriminative clusters in the target domain, which further improves the adaptation performance. The proposed HoMM can be easily integrated into other domain adaptation model, and it is also expected to benefit the knowledge distillation and image style transfer.

References

Ben-David, S.; Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; and Vaughan. 2010. A theory of learning from different domains. *Machine learning* 79(1-2):151–175.

Chen, C.; Chen, Z.; Jiang, B.; and Jin, X. 2019. Joint domain alignment and discriminative feature learning for unsupervised deep domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 3296–3303.

Csurka, G. 2017. Domain adaptation for visual applications: A comprehensive survey. *arXiv preprint arXiv:1702.05374*.

De Lathauwer, L.; Castaing, J.; and Cardoso, J.-F. 2007. Fourth-order cumulant-based blind identification of underdetermined mixtures. *IEEE Transactions on Signal Processing* 55(6):2965–2973.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research* 17(1):2096–2030.

Gatys, L. A.; Ecker, A. S.; and Bethge, M. 2016. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2414–2423.

Gou, M.; Camps, O.; and Sznaier, M. 2017. mom: Mean of moments feature for person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, 1294–1303.

Grandvalet, Y., and Bengio, Y. 2005. Semi-supervised learning by entropy minimization. In *Advances in neural information processing systems*, 529–536.

Gretton, A.; Borgwardt, K. M.; Rasch, M. J.; Schölkopf, B.; and Smola, A. 2012. A kernel two-sample test. *Journal of Machine Learning Research* 13(Mar):723–773.

Hoffman, J.; Tzeng, E.; Park, T.; Zhu, J.-Y.; Isola, P.; Saenko, K.; Efros, A.; and Darrell, T. 2018. Cycada: Cycle-consistent adversarial domain adaptation. In *International Conference on Machine Learning*, 1994–2003.

Jakubowski, J.; Kwiatos, K.; Chwaleba, A.; and Osowski, S. 2002. Higher order statistics and neural network for tremor recognition. *IEEE Transactions on Biomedical Engineering* 49(2):152–159.

Jia, Y., and Darrell, T. 2011. Heavy-tailed distances for gradient based image descriptors. In *Advances in Neural Information Processing Systems*, 397–405.

Koniusz, P.; Yan, F.; Gosselin, P.-H.; and Mikolajczyk, K. 2016. Higher-order occurrence pooling for bags-of-words: Visual concept detection. *IEEE transactions on pattern analysis and machine intelligence* 39(2):313–326.

Lee, C.-Y.; Batra, T.; Baig, M. H.; and Ulbricht, D. 2019. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 10285–10295.

Li, P.; Xie, J.; Wang, Q.; and Zuo, W. 2017a. Is second-order information helpful for large-scale visual recognition? In *Proceedings of the IEEE International Conference on Computer Vision*, 2070–2078.

Li, Y.; Wang, N.; Liu, J.; and Hou, X. 2017b. Demystifying neural style transfer. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 2230–2236. AAAI Press.

Long, M.; Cao, Y.; Wang, J.; and Jordan, M. 2015. Learning transferable features with deep adaptation networks. In *International Conference on Machine Learning*, 97–105.

Long, M.; Zhu, H.; Wang, J.; and Jordan, M. I. 2017. Deep transfer learning with joint adaptation networks. In *International Conference on Machine Learning*, 2208–2217.

Mansour, A., and Jutten, C. 1995. Fourth-order criteria for blind sources separation. *IEEE transactions on signal processing* 43(8):2022–2025.

Morerio, P.; Cavazza, J.; and Murino, V. Minimal-entropy correlation alignment for unsupervised deep domain adaptation. *international conference on learning representations*.

Pauwels, E., and Lasserre, J. B. 2016. Sorting out typicality with the inverse moment matrix sos polynomial. In *Advances in Neural Information Processing Systems*, 190–198.

Raudys, S. J., and Jain, A. K. 1991. Small sample size effects in statistical pattern recognition: Recommendations for practitioners. *IEEE Transactions on Pattern Analysis & Machine Intelligence* (3):252–264.

Shu, R.; Bui, H. H.; Narui, H.; and Ermon, S. 2018. A dirt-t approach to unsupervised domain adaptation. *arXiv preprint arXiv:1802.08735*.

Sun, B., and Saenko, K. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision*, 443–450. Springer.

Sun, B.; Feng, J.; and Saenko, K. 2016. Return of frustratingly easy domain adaptation. In *AAAI*, volume 6, 8.

Tzeng, E.; Hoffman, J.; Zhang, N.; Saenko, K.; and Darrell, T. 2014. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*.

Tzeng, E.; Hoffman, J.; Saenko, K.; and Darrell, T. 2017. Adversarial discriminative domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, 4.

Venkateswara, H.; Eusebio, J.; Chakraborty, S.; and Panchanathan, S. 2017. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5018–5027.

Wang, X.; Li, L.; Ye, W.; Long, M.; and Wang, J. 2019. Transferable attention for domain adaptation. In *AAAI Conference on Artificial Intelligence (AAAI)*.

Wang, Q.; Li, P.; and Zhang, L. 2017. G2denet: Global gaussian distribution embedding network and its application to visual recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2730–2739.

Xie, J.; Girshick, R.; and Farhadi, A. 2016. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, 478–487.

Xu, J.; Ye, P.; Li, Q.; Du, H.; Liu, Y.; and Doermann, D. 2016. Blind image quality assessment based on high order statistics aggregation. *IEEE Transactions on Image Processing* 25(9):4444–4457.

Yim, J.; Joo, D.; Bae, J.; and Kim, J. 2017. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4133–4141.

Zellinger, W.; Grubinger, T.; Lughofer, E.; Natschläger, T.; and Saminger-Platz, S. 2017. Central moment discrepancy (cmd) for domain-invariant representation learning. *arXiv preprint arXiv:1702.08811*.

Zhang, Z.; Wang, M.; Huang, Y.; and Nehorai, A. 2018. Aligning infinite-dimensional covariance matrices in reproducing kernel hilbert spaces for domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3437–3445.