Deep Domain Generalization With Structured Low-Rank Constraint

Zhengming Ding, *Student Member, IEEE*, and Yun Fu, *Senior Member, IEEE*

Abstract—Domain adaptation nowadays attracts increasing interests in pattern recognition and computer vision field, since it is an appealing technique in fighting off weakly labeled or even totally unlabeled target data by leveraging knowledge from external well-learned sources. Conventional domain adaptation assumes that target data are still accessible in the training stage. However, we would always confront such cases in reality that the target data are totally blind in the training stage. This is extremely challenging since we have no prior knowledge of the target. In this paper, we develop a deep domain generalization framework with structured low-rank constraint to facilitate the unseen target domain evaluation by capturing consistent knowledge across multiple related source domains. Specifically, multiple domain-specific deep neural networks are built to capture the rich information within multiple sources. Meanwhile, a domain-invariant deep neural network is jointly designed to uncover most consistent and common knowledge across multiple sources so that we can generalize it to unseen target domains in the test stage. Moreover, structured low-rank constraint is exploited to align multiple domain-specific networks and the domain-invariant one in order to better transfer knowledge from multiple sources to boost the learning problem in unseen target domains. Extensive experiments are conducted on several cross-domain benchmarks and the experimental results show the superiority of our algorithm by comparing it with state-of-the-art domain generalization approaches.

Index Terms—Domain generalization, deep learning, low-rank constraint.

I. INTRODUCTION

DOMAIN adaptation [1], [2] has already attracted considerable attentions in pattern recognition and computer vision field, since it well tackles the tasks with no or limited labeled target data. Generally, domain adaptation borrows well-established knowledge from source domains to alleviate the learning problem in the target domain. Conventional domain adaptation methods [3]–[8] consider seeking domain-invariant representation of the data or adapting classifiers, or both of them to mitigate the marginal or conditional

Z. Ding is with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115 USA (e-mail: allanding@ece.neu.edu).

Y. Fu is with the Department of Electrical and Computer Engineering, College of Computer and Information Science, Northeastern University, Boston, MA 02115 USA (e-mail: yunfu@ece.neu.edu).
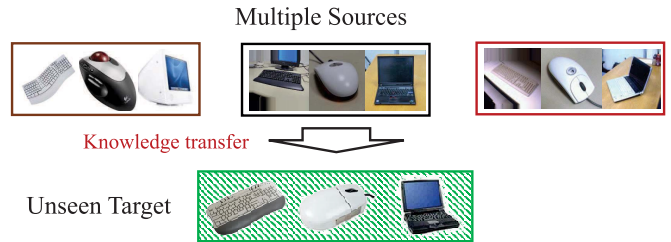
Fig. 1. Illustration of domain generalization problem, where we have multiple sources (3 sources here) while we are not accessible to the target domain during the training stage. Note that multiple sources and unseen target share the same labels. The key is to uncover the consistent information across multiple sources through deep structure to alleviate the unseen target learning.

distribution differences between source and target domains. In traditional domain adaptation scenarios, we still assume we have access to the target data in the training stage. In reality, however, we usually confront such challenges that the target data are totally unavailable in the training stage (Fig. 1). Thus, conventional domain adaptation techniques [3]–[7], [9]–[11] would fail in such cases, since they all need target data to guide knowledge transfer during model training.

Fortunately, we could always find some related source domains to extract valid knowledge to build a classifier for unseen targets. For example, we desire to evaluate image classification on VOC2007, and we can have the images from ImageNet, Caltech-256 and LabelMe databases available ahead of time. That is, we have multiple sources for training but without any knowledge about the target so that it is very difficult to guide the knowledge transfer. Actually, it is different from multi-source domain adaptation [4], [12]–[15], which has access to the target data in the training stage. Since these multiple sources have different distributions with each other as well as the target, it is very important to mitigate the domain shift between multiple sources and target by exploring more consistent information from multiple sources.

Most recently, domain generalization [16]–[23] has been well exploited to fight off the challenge through capturing knowledge from multiple source domains and generalizing to the unseen target domains. Along this line, there are mainly two strategies: one is domain-invariant feature learning, while the other is multiple classifiers adaptation. For example, Xu et al. exploited the low-rank structure from multiple latent source domains by extending an exemplar-SVM in order to capture the likelihoods of all positive samples [19]. However, existing domain generalization research efforts all employ shallow structures, so it is difficult for them to well uncover the rich information within the complex data. Therefore, it is easy
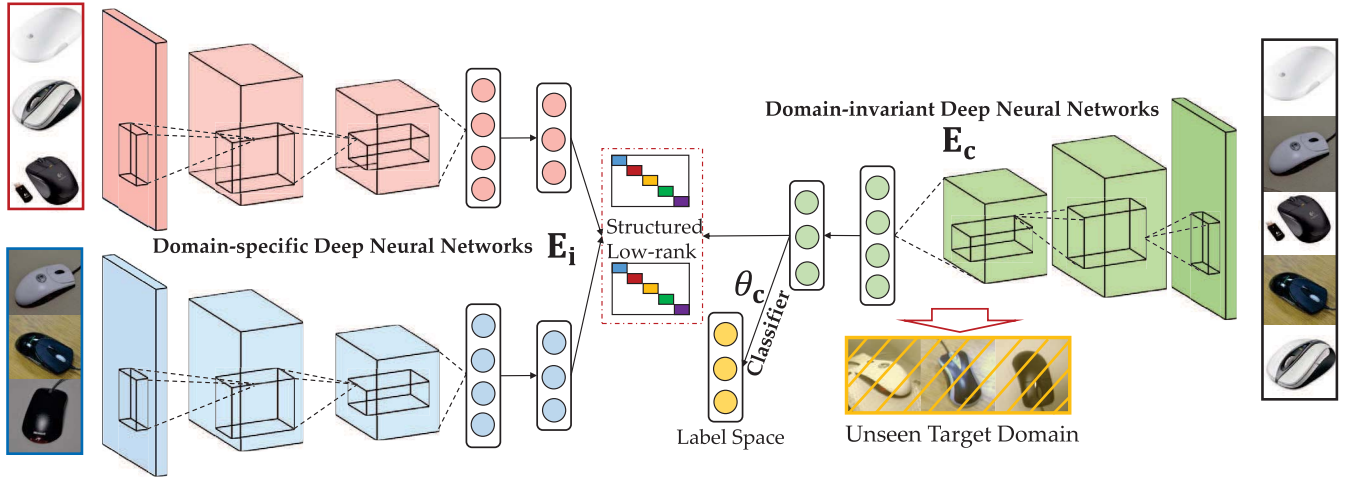
Fig. 2. Framework of our proposed algorithm. (a) multiple domain-specific deep structures $\mathbf{E}_i(\cdot)$ tend to be learned to capture the rich information from each source. (b) A domain-invariant deep structure $\mathbf{E}_c(\cdot)$ is built for all the domains, and further generalize to the unseen domain in the test stage with learned classifier $\theta_c(\cdot)$. (c) To couple the outputs of multiple domain-specific networks and domain-invariant one, low-rank reconstruction is adopted to align two types of networks in structured low-rank fashion.

to ignore the useful knowledge shared by multiple sources and hard to adapt the knowledge to the unseen target domains in the test stage.

Recently, deep learning [24], [25] has been extensively explored, since it could capture more effective knowledge in hierarchical feature extraction structure. That is, more discriminative and powerful knowledge underlying the data would be captured through multi-layer non-linear transformations [26]. Targeting at solving the distribution difference, deep learning has been well explored in domain adaptation, and it presents powerful ability in domain-invariant feature learning [27]–[30]. Specifically, deep domain adaptation technique manages to uncover common information across different domains. With deeper structures, the domain shift can be well mitigated so that the well-learned knowledge in source can be adapted to alleviate the target learning. In this way, deep structure learning is promising to uncover more discriminative features across multiple sources and generalize to the unseen target in the test stage.

In this paper, we present a Deep Domain Generalization framework (DDG) through structured low-rank constraint by leveraging the knowledge between multiple sources and the unseen target domains (Fig. 2). The core idea of DDG is to build a domain-invariant end-to-end deep structure by uncovering shared knowledge across multiple source domains so that the domain-invariant deep structure could well generalize to unseen target domains in the test stage. To our best knowledge, this is the first work for deep learning to fight off domain generalization.

A. Our Contributions

To explore more consistent knowledge across multiple sources, the idea of deep structure learning has been incorporated to build an effective end-to-end architecture to facilitate the unseen target learning. Here we summarize the main contributions of this work in two folds as follows:

- Multiple domain-specific deep structures and one domain-invariant deep structure are jointly built to uncover more useful information from each domain and shared by different domains, respectively. With multi-layer networks, the rich knowledge within sources can be learned to facilitate the unseen target learning. Aiming at coupling two types of networks, we adopt a difference loss to guide the deep learning.
- To better couple multiple domain-specific structures and the domain-invariant one, we deploy a structured low-rank reconstruction scheme to transfer the knowledge between domain-specific and the domain-invariant one. Specifically, the output from domain-specific networks would be only reconstructed by the output from domain-invariant network with the same class label under low-rank constraint. To this end, the learned domain-invariant deep structure can be more effective for the unseen target domains.

The remaining parts of this paper are presented as follows. We show a brief discussion of the related works in Section II. Then we provide our novel deep domain generalization algorithm in Section III, as well as its optimization solution. Experimental evaluations on several different benchmarks are reported in Section IV, which is followed with the conclusion in Section V.

II. RELATED WORK

In this part, we briefly discuss two lines of related work and further highlight the differences between our proposed algorithm and the existing works.

A. Domain Adaptation & Generalization

Domain adaption aims to tackle with the problem that the source distribution is different from the target one [12], [27], [30], [31]. However, traditional domain adaptation assumes the target data is accessible during the training procedure.

In reality, we may confront such cases that we cannot have any evaluation data available in the training stage.

To address the case where target data cannot be accessible for training, domain generalization [16], [18]–[21] has recently attracted increasing attentions, since it is promising to extend the training model from multiple sources to the unseen target. Multi-source adaptation problem [4], [12] is similar to domain generalization, since they both manage to extract the source knowledge and handle the distribution difference across sources. Therefore, their proposed techniques have many similar aspects. However, domain generalization is more challenging as there is no target data achievable in the training procedure for help.

Along this line, there are mainly two strategies: one is domain-invariant feature learning and the other is multiple classifiers adaptation. In the first line, Muandet et al. developed a kernel-based optimization algorithm to build a domain-invariant projection by minimizing the divergence across different domains [17]. Fang et al. designed Unbiased Metric Learning (UML) algorithm through learning to rank framework, which produced a less biased distance metric with better domain generalization performance in weakly-labeled web images [18]. Furthermore, Ghifary et al. developed a multi-task auto-encoder by encoding with a common layer while decoding with domain-specific layers in order to uncover the shared information across multiple domains for the unseen target domains [20]. Along the second line, Khosla et al. designed a multi-task max-marginal classifier on multiple source domains, where the learned weights that are shared to all source domains can be utilized for knowledge generalization [16]. Most recently, Niu et al. extended multiple instance learning and constructed one classifier per class per latent domain, and hence, multiple classifiers could be effectively integrated to obtain promising generalization [21].

However, current work on domain generalization ignored the rich information from the hierarchical structure within the data. Differently, we develop two types of deep neural networks to uncover more effective information across multiple sources to alleviate the unseen target learning. Furthermore, we adopt structured low-rank constraint to build a more effective domain-invariant deep structure for unseen target learning.

B. Deep Learning

In the recent years, deep Learning has been well explored and increased great attention in a lot of real-world applications, due to its promising ability in different learning tasks, e.g., face recognition [32], image super-resolution [25], action classification [33], [34], feature selection [35], image clustering [36] and multi-view learning [37]. In general, deep learning attempts to build hierarchical structures to extract features with the input of raw data. Most deep learning approaches assume that deep architecture is able to extract invariant features which can be transferable across various learning tasks. However, the domain discrepancy, unfortunately, cannot be removed, but only alleviated with the deep structure. Thus, domain mismatch is one of the bottlenecks to the adaptability of deep architecture.

Most recently, a lot of deep domain adaptation methods are proposed to address the domain discrepancy based on the deep structure by making merit of both deep learning and domain adaptation techniques [27], [29], [30], [38], [39]. The core idea is to improve the representation adaptability in the higher layers of the deep structure through explicitly mitigating the domain distribution mismatch. However, current deep domain adaptation algorithms all assume the target data are achievable for model training, that is, none of them are designed for domain generalization.

In this paper, we are the first to explore the deep structures to address the domain generalization challenge. To transfer more effective information from the multiple sources to the domain-invariant network, which is further extended to the unseen target domains, we propose a structured constraint to build a bridge across them under low-rank assumption.

III. THE PROPOSED MODEL

In this part, we will introduce our proposed Deep Domain Generalization learning algorithm for unseen target domains by constructing multiple domain-specific and one domain-invariant deep neural networks from several related sources. Then, we provide the solution to the proposed algorithm.

A. Preliminaries

Domain can be defined as probability distribution $\mathbb{P}_{XY}$ on $\mathcal{X} \times \mathcal{Y}$, in which $\mathcal{X}$ and $\mathcal{Y}$ represent the data and label spaces, respectively. For simplicity, we denote $\mathbb{P}_{XY}$ as $\mathbb{P}$. Assume $\mathbb{D} = \{x_j; y_j\}_{j=1}^{n}$ as an independent and identically distributed (i.i.d.) sample from a domain. In domain generalization, assume $\Delta = \{\mathbb{P}_{s,1}; \cdots; \mathbb{P}_{s,m}\}$ as a combination of m source domains and $\mathbb{P}_t \notin \Delta$ as a target domain. Assume i-th source domain $\mathbb{D}_{s,i} = \{x_i^j; y_i^j\}_{j=1}^{n_i}$ with n_i labeled samples. In the test stage, we aim to evaluate on some unseen domains $\mathbb{D}_t$ with the same categories.

The main difference between multi-source domain adaptation and domain generalization is on the accessibility of the target data during the training stage. Both manage to seek a labeling function $\theta : \mathcal{X} \to \mathcal{Y}$ that can achieve good performance on the target data. Note that domain generalization would definitely become to multi-source domain adaptation, when $\mathbb{P}_t \in \Delta$. Although these two are highly related problems, domain adaptation approaches generally cannot be directly exploited in domain generalization scenario. Thus, it is significantly desirable to propose models efficiently for domain generalization.

B. Motivation

When dealing with a learning problem that the evaluation data are only available in the test stage, domain generalization attracts increasing attentions by exploring more consistent knowledge across multiple related sources domains, which are all drawn from different distributions with the unseen target. Along the line of multi-source domain adaptation [4], [12] and domain generalization [17], [20], most researchers assume there is a latent domain-free space, where sources and target

could align well. Thus, it is very essential to find the latent space so that multi-source knowledge could be transferred to the unseen target to boost the learning problem. However, current domain generalization [16], [18]–[21] works all adopted shallow structures to extract domain-invariant features, which actually cannot uncover enough effective features to facilitate the target learning.

To this end, we desire to explore deep structure learning in domain generalization to uncover more effective knowledge across multiple sources. Considering the distribution divergence among multiple sources, we design multiple domain-specific deep structures, each for one source domain to transform multiple sources into the latent space. Specifically, each domain-specific network can be treated as one task in multi-task scheme, and Bengio et al. mentioned that good representation should have such prior as *shared factors across tasks* [24]. Therefore, we assume different networks should have similar activation values given the same class of objects from different domains, and our domain-invariant network is designed to capture most shared factors across tasks. Such domain-invariant deep structure is treated as an approximation of the specific target domain deep structure when we cannot have access to any target data during the training procedure. Finally, we further learn a classifier based on the domain-invariant deep structure, thus, we can evaluate on the unseen target data in the test stage.

C. Deep Domain Generalization Model

To build a deep domain generalization model, we extend the well-known architectures [38], [40], which are comprised of several convolutional layers and a few fully-connected layers. Generally, each fully-connected (fc) layer l learns a nonlinear transformation $h_i^l = f^l(W^l h_i^{l-1} + b^l)$, in which h_i^l is the l-th layer hidden representation of sample x_i (W^l and b^l are the weights and bias for the l-th layer), while $f^l(\cdot)$ is the activation function, for example, rectifier function $f^l(x) = \max(0, x)$ for hidden layers or Softmax function $f^l(x) = \exp(x) / \sum_{j=1}^{|X|} \exp(x_j)$ for the output layer.

Current deep domain adaptation approaches [27], [29], [38] suggest that high-layer features eventually transition from general to specific along the deep architecture, the feature adaptability gap increase with the domain mismatch particularly enlarged. Here we do not focus on how to calculate the convolutional layers, since we do not enforce distribution-adaptation constraint in those layers, provided that the convolutional layers are able to generate generic features [41]. Define $\mathbf{E}_i = \{W_i^l, b_i^l\}|_{l=1}^L$ as the set of all CNN parameters for each source-specific network, $\mathbf{E}_c = \{W_c^l, b_c^l\}|_{l=1}^L$ as the set of all CNN parameters for the domain-invariant network and θ_c as the parameter for standard Softmax classifier.

For our model, inference is provided with $\bar{y} = \theta_c(\mathbf{E}_c(x))$, in which $\bar{y}$ is the task-specific label prediction. In this paper, we attempt to minimize the following loss function with respect to parameters $\Theta = \{\mathbf{E}_i, \mathbf{E}_c, \theta_c\}$:

$$\mathcal{L} = \mathcal{L}_g + \gamma \mathcal{L}_d + \beta \mathcal{L}_r, \tag{1}$$

which includes three parts, i.e., classification loss $\mathcal{L}_g$, cross-network difference loss $\mathcal{L}_d$ and reconstruction loss $\mathcal{L}_r$.

First of all, the classification loss $\mathcal{L}_g$ tends to guide the model training to assign the output labels. Specifically, we attempt to minimize the negative log-likelihood of the ground truth class, provided every labeled source sample:

$$\mathcal{L}_g = -\sum_{j=0}^{N} y_{s,j} \log(\bar{y}_{s,j}), \tag{2}$$

in which $N = \sum_i n_i$, and $y_{s,j}$ is the class label for source sample $x_{s,j}$, while $\bar{y}_{s,j}$ is the label prediction as $\bar{y}_{s,j} = \theta_c(\mathbf{E}_c(x_{s,j}))$.

The cross-network difference loss $\mathcal{L}_d$ is also exploited to all source domains and encourages the domain-invariant networks to preserve most information across multiple domain-specific networks. Since multiple sources all share the same categories but in different distributions, we assume the domain-specific deep networks share most information in the latent space, where the knowledge can be extended to the unseen target domain. Specifically, each domain-specific network can be treated as one task. It has been suggested in [24] that good representations should have such prior as *Shared Factors Across Tasks*. Therefore, we assume that different networks should have similar activation values given the same class of objects from different domains, and our domain-invariant network is designed to identify shared factors across tasks, which is an approximation for unseen target domain. An intuitive strategy is to couple the weights in each layer between the domain-specific networks and the domain-invariant one. To this end, we minimize the following connections across them as:

$$\mathcal{L}_d = \sum_{i=1}^{m} \sum_{l=1}^{L} \left(\|W_i^l - W_c^l\|_F^2 + \|b_i^l - b_c^l\|_2^2 \right), \tag{3}$$

In this way, the common domain-invariant network can uncover shared factors across multiple domain-specific networks so that it can be better extended to alleviate the unseen target learning in the real test stage. The cross-network constraint (3) aims to uncover more shared knowledge for domain-invariant deep network from multiple sources.

D. Structured Low-rank Reconstruction

The reconstruction loss $\mathcal{L}_r$ is also applied to encourage the shared and private networks to capture more discriminate information. When the domain distribution difference is mitigated with deep architecture, we could observe that the within-class data would gather together although they are from different domains. That is, each sample would be highly similar to other samples in the same class. In this way, the conditional distribution difference could be addressed. Therefore, we explore the structured low-rank constraint could help transfer knowledge from domain-specific networks to domain-invariant networks.

Let $\mathbf{H}_{s,i}$ and $\mathbf{H}_{c,i}$ be matrices whose rows are the domain-specific representation and domain-invariant representations $\mathbf{H}_{s,i} = \mathbf{E}_i(X_{s,i})$ and $\mathbf{H}_{c,i} = \mathbf{E}_c(X_{s,i})$ from samples of i-th source, respectively. To guide two types of networks learning and adapt more effective information to the unseen target data, we adopt low-rank reconstruction to couple the output of
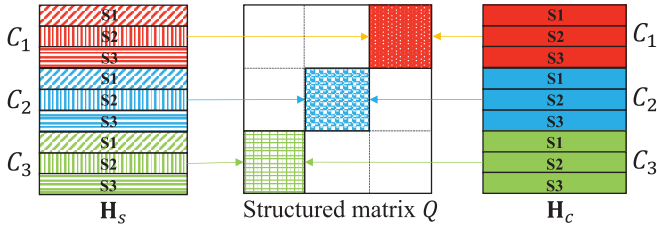
Fig. 3. Illustration of structure matrix Q for three sources (S1, S2, S3) with three categories (C1,C2, C3). Q only has positive values at the positions in which sources share the same labels, otherwise Q is 0.

each domain-specific network and the output of the domain-invariant network in the following:

$$\min_{Z} \text{rank}(Z), \quad \text{s.t. } \mathbf{H}_s = \mathbf{H}_c Z, \quad (4)$$

where $\mathbf{H}_c = [\mathbf{H}_{c,1}, \cdots, \mathbf{H}_{c,m}]$ and $\mathbf{H}_s = [\mathbf{H}_{s,1}, \cdots, \mathbf{H}_{s,m}]$. $Z \in \mathbb{R}^{N \times N}$ is the reconstruction coefficient matrix and $\text{rank}(\cdot)$ is the rank operator [42]. The rank minimization issue can be addressed by the trace norm $\| \cdot \|_*$ as a good surrogate [42].

Actually, low-rank reconstruction matrix Z tends to be block-diagonal in the ideal case, since only the within-class data are correlated across two types of deep networks. Similar ideas were proposed in the literature [31], which adopted multi-view data to reconstruct each view under linear transformations with low-rank constraint. However, such shallow structures with linear transformations would fail to represent complex data. Our algorithm joints deep structure learning and low-rank constraint into a unified framework in order to capture more effective information shared by multiple sources. In this way, the domain shift across multiple sources could be further mitigated to boost the unseen target learning.

However, previous model (4) works in an unsupervised way, which still treats the dataset as whole and ignores the discriminative information during low-rank reconstruction. In other word, it is hard to recover block-diagonal low-rank matrices. Motivated by previous structured low-rank learning [43], [44], we propose to guide the low-rank reconstruction through structured matrix across two types of deep networks as follows:

$$\min_{Z} \|Z\|_* + \alpha \|Z - Q\|_F^2,$$
$$\text{s.t. } \mathbf{H}_s = \mathbf{H}_c Z, \quad (5)$$

where $\alpha > 0$ is the trade-off parameter and Q is the structured low-rank matrix (Figure 3), which is defined as follows:

$$Q[j,k] = \begin{cases} 1, & \text{if } \mathbf{H}_s^j \text{ and } \mathbf{H}_c^k \text{ belong to the same class} \\ 0, & \text{otherwise} \end{cases}$$

Clearly, Q is a block-diagonal structured matrix, because it is constructed based on within-class relationship. Such structured low-rank reconstruction would involve more label information during the deep structures learning so that the conditional distribution difference across two types of deep structures could be reduced. To that end, we develop a novel loss function for deep domain generalization with the

constraint $\mathbf{H}_s = \mathbf{H}_c Z$:

$$\mathcal{L}_r = \|Z\|_* + \alpha \|Z - Q\|_F^2. \quad (6)$$

E. Model Training

Eq. (1) is challenging to be addressed because of the non-linearity and non-convexity. Thus, we present an alternating optimization method to iteratively update two subproblems: 1) optimize the low-rank coefficients; 2) learn two types of deep structures.

Low-Rank Coefficient Optimization: we propose to update low-rank coefficients, then provide the deep structure optimization. When the deep structures are fixed, the objective function in Eq. (1) degenerates to a traditional low-rank modeling problem. Then, we derive the corresponding augmented Lagrangian function of Eq. (5) w.r.t Z:

$$\bar{\mathcal{L}}_r = \|Z\|_* + \alpha \|Z - Q\|_F^2 + \langle R, \mathbf{H}_s - \mathbf{H}_c Z \rangle$$
$$+ \frac{\mu}{2} \|\mathbf{H}_s - \mathbf{H}_c Z\|_F^2, \quad (7)$$

where R is the Lagrange multiplier and $\mu > 0$ is the penalty parameter. $\langle , \rangle$ denotes the inner product operator of two matrices. Specifically, we first define $\bar{\mathcal{L}}_r(Z, Q, R_\tau, \mu_\tau) = \alpha \|Z - Q\|_F^2 + \langle R_\tau, \mathbf{H}_s - \mathbf{H}_c Z \rangle + \frac{\mu_\tau}{2} \|\mathbf{H}_s - \mathbf{H}_c Z\|_F^2$, then have following updating rules of Z variable at time τ:

$$Z = \arg\min_{Z} \|Z\|_* + \bar{\mathcal{L}}_r(Z_\tau, Q, R_\tau, \mu_\tau)$$
$$= \arg\min_{Z} \frac{1}{\xi \mu_\tau} \|Z\|_* + \|Z - Z_\tau + \nabla_{Z_\tau} \bar{\mathcal{L}}_r\|_F^2 \quad (8)$$

where $\nabla_{Z_\tau} \bar{\mathcal{L}}_r = \nabla_{Z_\tau} \bar{\mathcal{L}}_r(Z_\tau, Q, R_\tau, \mu_\tau) = 2\alpha(Z_\tau - Q) - \mathbf{H}_s^\top R_\tau - \mu_\tau \mathbf{H}_c^\top (\mathbf{H}_s - \mathbf{H}_c Z_\tau)$ and $\xi = \|\mathbf{H}_c\|_F^2$. Problem (8) can be effectively addressed with the singular value thresholding (SVT) operator [45]. Suppose $U_Z \Sigma_Z V_Z$ are the SVD of matrix $(Z_\tau - \nabla_{Z_\tau} \bar{\mathcal{L}}_r)$, in which $\Sigma_Z = \text{diag}(\{\sigma\}_{1 \le i \le r})$ with singular value σ. Thus, we could achieve the optimal of Z at time $\tau + 1$ as $Z_{\tau+1} = U_Z \Omega_{(\frac{1}{\mu_\tau})}(\Sigma_Z) V_Z$, in which $\Omega_{(\frac{1}{\mu_\tau})} = \text{diag}(\{\sigma - \frac{1}{\mu_\tau}\}_+)$, and a_+ represents the positive part of a [45].

The multiplier R and the penalty parameter μ are updated as $R_{\tau+1} = R_\tau + \mu_\tau (\mathbf{H}_s - \mathbf{H}_c Z_{\tau+1})$ and $\mu_{\tau+1} = \min(\rho \mu_\tau, \mu_{\max})$, respectively, where $\rho = 1.2$ and $\mu_{\max} = 10^6$. We obtain the low-rank coefficient Z until it converges with the condition of $\|\mathbf{H}_s - \mathbf{H}_c Z\|_\infty < \epsilon$ ($\epsilon = 10^{-6}$).

Learning Deep Weights: after Z updated, we could adopt conventional gradient descent strategy to optimize parameters of neural networks individually. Specially, we first calculate the gradient descant of each variable, w.r.t., $\partial \mathcal{L} / \partial \Theta$. Then, Θ is updated through

$$\Theta = \Theta + \eta \frac{\partial \mathcal{L}}{\partial \Theta}, \quad (9)$$

where η is updated with Adagrad Rule [46].

Large-Scale Data Extension: When dealing with large-scale dataset, our deep domain generalization model with structured low-rank constraints can still adopt mini-batch stochastic gradient descent (SGD) strategy. Specifically, we need to do

some modification on the optimization procedure. For every random sampling, we randomly select a certain number of samples per class from different domains, i.e., n_r samples each class. In this way, we can ensure the low-rank property to be preserved. Thus, in each sampling, we have subset $A_s = [A_{s,1}, \cdots, A_{s,m}]$ from m domains, each $A_{s,i}$ with $n_r \times C$ samples. For each mini-batch, we could iteratively update the above two subproblems. Since our structured low-rank constraint is exploited to transfer more effective knowledge across two types of networks, hence, for each mini-batch, we can still achieve this goal if we sample the data with low-rank property. To this end, we conclude that our model can be quite scalable to large-scale dataset.

F. Model Comparison

In this part, we will discuss three most related works to ours.

The first related work in domain generalization is MTAE [20], which exploits the common encoder to transform multiple sources to the hidden layer, then adopted domain-specific decoders. In this way, the common domain-invariant encoder could be generalized to unseen target domains. Their reconstructions provide a class-level correspondence, which constrains the number of samples in one class to be the same for each domain.

The second related works are DSN [39] and WSDTN [47], which are both designed for domain adaptation. Specifically, DSN is designed for domain adaptation by building two types of networks, i.e., domain-specific and domain-invariant networks. Furthermore, DSN adopts orthogonal constraints to enforce source and target with more shared knowledge. WSDTN proposes two networks for two modalities with weakly-shared constraint, which is the same to our constraint in Eq. (3). However, both are developed for traditional domain adaptation, which assume target data are available during training.

However, we adopt structured low-rank reconstruction to adapt the two types of deep networks for domain generalization, where test data are totally blind in the training stage. Our low-rank reconstruction would make the same class data in two types of networks correlate to mitigate the within-class variance. Moreover, we explore the similar strategy to WSDTN to enforce multiple domains with more shared knowledge.

IV. EXPERIMENTAL RESULTS

In this part, we evaluate our proposed model on several cross-domain benchmarks by comparing state-of-the-art domain generalization algorithms. We also analyze the convergence and each component individually.

A. Datasets & Experimental Setting

In the experiments, we mainly use four cross-domain databases, which are Office-31, Office-10 + Caltech-10 (Fig. 4), VLCS-5 and MUS-10 (Fig. 5). The statistic of four datasets are shown in Table I.

Fig. 4. Example images from the Backpack category in Caltech-256 and Office-31 (i.e., Amazon, DSLR, and Webcam). Specifically, Caltech-256 and Amazon images are mostly from online merchants, while DSLR (high resolution) and Webcam (low resolution) images are from offices. (Best viewed in color).
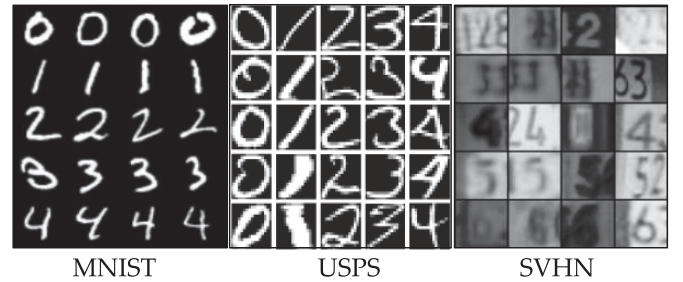

Fig. 5. Example from cross-domain digit dataset of three subsets, i.e., MNIST, USPS and SVHN.

Office-31: is a widely-adopted cross-domain benchmark, which consists of three real-world visual object datasets, i.e., Amazon (images downloaded online), DSLR (high-resolution images collected with a digital SLR camera) and Webcam (low-resolution images captured with a web camera). In total, there are 4,652 images from 31 categories.

Office-10 + Caltech-10: dataset contains Office-31 and Caltech-256, which is a standard benchmark for visual object classification.[1] There are 10 common categories shared with them. Generally, each category has 8 to 151 samples, and in total it includes 2,533 samples. Here we build different evaluation cases by selecting sources and targets.

VLCS-5: is another cross-domain object benchmark, which contains the images from PASCAL VOC2007 (V), LabelMe (L), Caltech-101 (C), and SUN09 (S), with each as a domain. Specifically, V, L, and S are scene-centric datasets, while C is an object-centric dataset. There are five shared object categories, i.e., "car", "bird", "chair", "dog", and "person". We follow the setting in [20] to divide each domain of VLCS into a training set (70%) and a test set (30%) through random selection.

MUS-10: digit dataset includes three subsets, i.e., MNIST (M), USPS (U) and SVHN (S). Each dataset

[1] http://www.scf.usc.edu/ boqinggo/domainadaptation.html

TABLE I

STATISTICS OF THE THREE BENCHMARK DATASETS

Dataset	Type	#Example	#Class	#Subset
Office-31	Object	4,652	31	3
Office-10+Caltech-10	Object	2,533	10	4
VLCS-5	Object	10,729	5	4
MUS-10	Digit	140K	10	3

Fig. 6. The cross-domain recognition accuracy % of 7 algorithms on the MUS-10 dataset.

contains digits belonging to 10 classes (0-9), each captured under different conditions. Specifically, MNIST and USPS are large datasets of handwritten digits captured under constrained conditions. Both these domains are visually very similar and this makes adaptation relatively easy. SVHN has significant variations (See Fig. 5) in many aspects, which makes it quite different from MNIST and USPS.

In this section, we mainly evaluate our proposed approach with the following 8 comparisons:

- **DCNN**: We directly apply labeled sources to only train one deep convolutional neural network (DCNN), then exploit to predict the labels of unseen targets.
- **LRCS** [31]: a multi-view common subspace learning algorithm, which seeks a shared low-rank subspace for view-unknown test data.
- **Undo-Bias** [16]: a multi-task SVM-based model to address the dataset bias.
- **UML** [18]: a structural metric learning-based approach, targeting at building a domain-invariant distance metric for target learning.
- **DICA** [17]: a kernel-based learning approach that seeks an domain-invariant projection by minimizing the discrepancy across domains.
- **LRE-SVM** [19]: a non-linear exemplar-SVM algorithm with a low-rank constraint on the likelihood matrix.
- **MTAE** [20]: multi-task auto-encoder seeks to convert the raw data into analogs in multiple related sources. It thereby extracts new representations robust to domain mismatch.
- **DSN** [39]: a deep domain adaptation model by learning a shared encoding network.
- **SCA** [23]: a linear subspace approach by maximizing the discriminability of classes and minimizing the mismatch between domains.

The first two, DCNN and LRCS, are two baselines while the middle five algorithms are the state-of-the-art domain generalization approaches. DSN is previously designed for unsupervised domain adaptation, since it also attempts to seek a common deep structure shared by different two domains. Applying it to domain generalization, we do some modification that the classifier is learned on all the labeled sources. Specifically, we adopt the MMD as the similarity loss. We evaluate on two-source domain generalization challenges, thus, we can still use the DSN code easily.

Throughout the experiments, we implement CNN topologies based on the deep structures [29], [38], [39]. Detailedly, for **Office-31**, **Office-**10 + **Caltech-10** and **VLCS-5**, we exploit
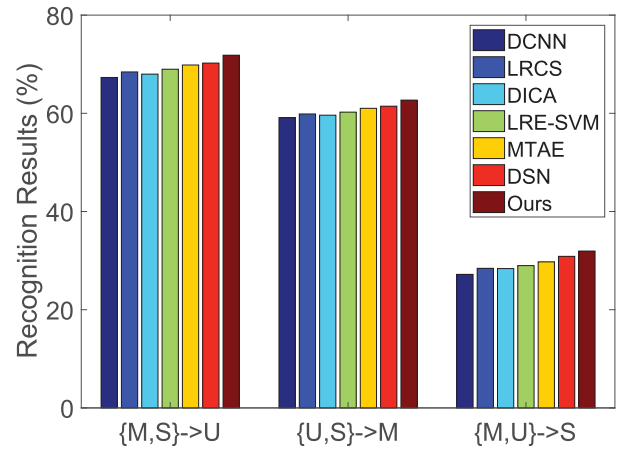
the AlexNet structure [40]. Since these three datasets are rather small-scale and the size of available training data is essential to build a powerful deep model, hence we opted for the fine-tuning of the CNN pre-trained on the ImageNet (AlexNet with the Caffe toolbox [48]) [27], [29], [38]. For cross-domain digit dataset (**MUS-10**), we use the CNN structure [49].

Besides, for shallow learning algorithms, we adopt the DeCAF[6] features extracted from the first three datasets (Office-31, Office-10 + Caltech-10 and VLCS-5) as inputs to the methods. These representations have dimensionality of 4,096 and are public to download. Office-31 is publicly available,[2] Office-10 + Caltech-10 can be downloaded from the website.[3] VLCS-5 features are available on the website.[4] While for the last digit dataset, we extract the features with the structure [49].

In the experiments, we report the model performance in terms of the classification accuracy (%) [19]. Here, we tune the hyper-parameters for each model through 10-fold cross-validation on labeled sources, since we have no labeled target data available for training. Note that "{Webcam, DSLR, Amazon} → Caltech" represents that the sources are Webcam, DSLR and Amazon, while the unseen target is Caltech.

B. Domain Generalization Results

We testify the domain generalization performance of each algorithm, and the results are shown in Tables II, III, IV and Fig. 6.

Generally, we observe that our proposed algorithm could outperform all the other comparisons in all cases, since our proposed algorithm deploys two types of deeper structures and incorporates structured low-rank reconstruction fashion. Therefore, more discriminative knowledge could be transferred to facilitate the learning problem to unseen target domains. Besides, domain generalization algorithms could achieve better results than others in most cases. That shows domain generalization techniques definitely facilitate the unseen target

[2]http://vc.sce.ntu.edu.sg/transfer_learning_domain_adaptation/

[3]https://people.eecs.berkeley.edu/ jhoffman/domainadapt/#datasets_code

[4]http://www.cs.dartmouth.edu/ chenfang/proj_page/FXR_iccv13/index.php

TABLE II

THE CROSS-RECOGNITION ACCURACY % OF 9 ALGORITHMS ON THE OFFICE-31 DATASET. RED COLOR DENOTES THE BEST RECOGNITION RATES. BLUE COLOR DENOTES THE SECOND BEST

Sources → Target	DCNN	LRCS [31]	Undo-Bias [16]	UML [18]	DICA [17]	LRE-SVM [19]	MTAE [20]	DSN [39]	Ours
{A, W} → D	97.59	98.46	98.45	98.76	98.35	98.84	98.97	99.02	99.44
{A, D} → W	89.43	93.24	93.38	93.76	93.56	93.98	94.21	94.45	95.28
{D, W} → A	35.57	43.86	42.43	41.65	42.98	44.14	43.67	43.98	45.36

TABLE III

THE CROSS-RECOGNITION ACCURACY % OF 9 ALGORITHMS ON THE OFFICE-10 + CALTECH-10 DATASET. RED COLOR DENOTES THE BEST RECOGNITION RATES. BLUE COLOR DENOTES THE SECOND BEST

Sources → Target	DCNN	LRCS [31]	Undo-Bias [16]	UML [18]	DICA [17]	LRE-SVM [19]	MTAE [20]	DSN [39]	SCA[23]	Ours
{C,D,W} → A	92.13	92.87	90.98	91.02	90.40	91.87	93.13	-	92.38	94.21
{A,D,W} → C	85.89	86.12	85.95	84.59	84.33	86.38	86.15	-	86.73	87.63
{A,C} → {D,W}	83.41	84.21	80.49	82.29	79.65	84.59	85.35	85.76	85.84	86.32
{D,W} → {A,C}	76.49	79.98	69.98	79.54	69.73	81.17	80.52	81.22	75.54	82.24

TABLE IV

THE CROSS-RECOGNITION ACCURACY % OF 9 ALGORITHMS ON THE VLCS DATASET. RED COLOR DENOTES THE BEST RECOGNITION RATES. BLUE COLOR DENOTES THE SECOND BEST

Sources → Target	DCNN	LRCS [31]	Undo-Bias [16]	UML [18]	DICA [17]	LRE-SVM [19]	MTAE [20]	DSN [39]	SCA [23]	Ours
{L,C,S} → V	59.10	58.36	54.29	56.26	59.62	60.58	63.90	-	64.36	65.25
{V,C,S} → L	58.20	59.22	58.09	58.50	51.82	59.74	60.13	-	59.60	62.34
{V,L,S} → C	86.67	88.68	87.50	91.13	78.30	88.11	89.05	-	88.92	92.76
{V,L,C} → S	57.86	58.54	54.21	58.49	55.33	54.88	61.33	-	59.29	63.54
{C,S} → {V,L}	55.08	58.62	59.28	56.47	50.90	55.04	57.23	58.34	59.50	59.66
{C,L} → {V,S}	52.60	54.86	55.80	54.72	55.47	52.87	53.26	54.28	55.96	56.82
{V,C} → {L,S}	56.62	59.36	62.35	55.49	58.08	58.84	60.84	62.08	60.77	63.12

learning by borrowing the knowledge from related multiple sources.

Secondly, compared with DCNN, our model builds two types of networks, thus our model can have a better flexibility in dealing with unseen target domains. DCNN can only capture the knowledge from sources while cannot well uncover the shared knowledge across sources for unseen targets. LRCS also adopts the low-rank reconstruction to further learn a view-invariant linear projection, however, it is an unsupervised and shallow model so that it cannot well adapt the source knowledge from sources to unseen target. For other domain generalization methods, most of them can achieve better performance based on deep features, compared with DCNN. This indicates that deep structure learning is not enough to handle the unseen domain challenge, and they cannot build a transferable network. However, domain generalization approaches still can improve the performance.

Thirdly, 1) from the classification performance in Tables II, III and IV, we could observe deep learning features, i.e., DeCAF[6] features, could help the unseen target learning. The reason is that all the domains are extracted features through a common deep structure, which is trained on large-scale dataset, so that the domain mismatch across sources and target could be mitigated to some extent. We can also notice that all the algorithms work worse on {D, W} → A than other two cases, since A has a larger distribution difference with W and D. In handcrafted features, domain generalization techniques cannot help a lot for this challenge case. 2) from the results in Table III, A and C have higher similarity while D and W share higher similarity, therefore, we can witness that the first four cases achieve better results, especially {A, C, D} → W. However, domain generalization methods can still improve the performance a little bit. 3) For VLCS-5, the object-centric dataset, Caltech-101, appears to show the best performance, as scene-centric datasets usually tend to have a good generalization over object-centric datasets [50]. 4) For MUS-10, MNIST and USPS have similar distributions, thus, we can achieve better performance when unseen target is MNIST or USPS. However, when SVHN is the unseen target, the performance is worse since SVHN is much different from MNIST and USPS.

C. Empirical Analysis

In this part, we mainly testify some properties of the proposed model.

First of all, we evaluate several variants of our model to to understand our model deeply.

- *DDG_l:* we remove the structured term, i.e., $\|Z - Q\|_{\mathrm{F}}^2$ (set $\alpha = 0$).
- *DDG_p:* we further remove the low-rank reconstruction based on DDG_l by replacing pair-wise reconstruction (set $\beta = 0$). That is, $Z = \mathrm{I}_N$, where I_N is the identity matrix with size of $N \times N$.
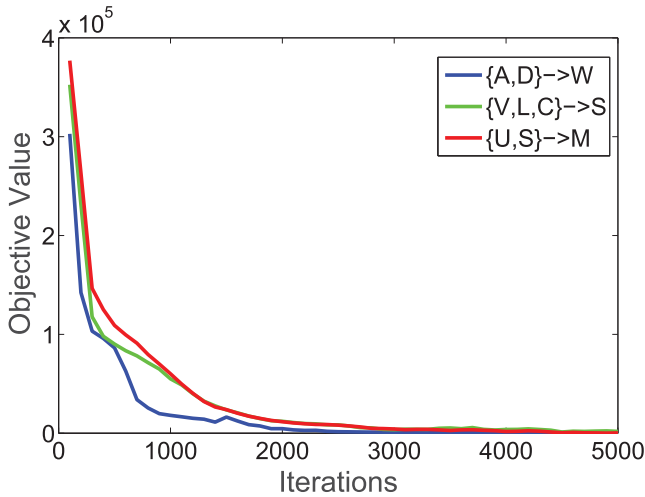
Fig. 7. Convergence curves of our model on three cases, i.e., $\{A, D\} \to W$, $\{V, L, C\} \to S$ and $\{U, S\} \to M$.

TABLE V

RECOGNITION RATE OF 4 ALGORITHMS ON DIFFERENT
EVALUATION CASES

Config\Methods	DDG_l	DDG_p	DDG_s	Ours
$\{D,W\} \to A$	41.98	41.04	39.45	45.36
$\{A,D,W\} \to C$	86.12	85.74	84.56	87.63
$\{V,L,C\} \to S$	60.42	59.68	58.62	63.54
$\{U,S\} \to M$	59.98	58.36	57.74	62.10

- *DDG_s:* we further remove the shared factor constraint, i.e., (3), from DDG_p (set $\gamma = 0$).

From the results, we notice that our model outperforms other variants, which indicates that each component contributes to domain generalization. Since we remove each term step by step, we could have a clear view that each component has an improvement.

Secondly, we testify the convergence of our proposed model. Actually, our model has two subproblems, one is traditional deep convolution neural networks, the other is low-rank modeling. In fact, for each subproblem, researchers has well shown their convergence. However, it is still hard to theoretically prove the convergence of deep learning models, especially for our model with two subproblems. Thus, researchers usually empirically testify the convergence of the deep structure. We found our model converges well for all the cases. Here we adopt the test cases $\{A, D\} \to W$ of Office-31, $\{V, L, C\} \to S$ of VLCS, and $\{U, S\} \to M$ of cross-domain digit dataset, respectively. The convergence curves are shown in Fig. 7, where we could observe that our model converges well.

V. CONCLUSION

In this paper, we developed a deep domain generalization framework with structured low-rank constraint, which aimed to seek most shared discriminative knowledge within multiple sources to facilitate the unseen target learning. Specifically, we built two types of deep structures, domain-specific and

domain-invariant, to capture most common discriminative information shared by multiple sources so that the knowledge could be transferred to the unseen target domains. The structured low-rank reconstructions were adopted to mitigate the gap between domain-specific and domain-invariant structure. Experimental evaluation on several popular cross-domain benchmarks verified the superiority of our proposed model, by comparing with the state-of-the-art domain generalization methods.

REFERENCES

[1] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[2] L. Shao, F. Zhu, and X. Li, "Transfer learning for visual categorization: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 5, pp. 1019–1034, May 2014.
[3] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2066–2073.
[4] I.-H. Jhuo, D. Liu, D. Lee, and S.-F. Chang, "Robust visual domain adaptation with low-rank reconstruction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2168–2175.
[5] M. Long, J. Wang, G. Ding, J. Sun, and P. Yu, "Transfer joint matching for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1410–1417.
[6] O. Sener, H. O. Song, A. Saxena, and S. Savarese, "Learning transferrable representations for unsupervised domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2110–2118.
[7] D. Novotny, D. Larlus, and A. Vedaldi, "I have seen enough: Transferring parts across categories," in *Proc. Brit. Mach. Vis. Conf.*, 2016, p. 1.
[8] Z. Ding, M. Shao, and Y. Fu, "Missing modality transfer learning via latent low-rank constraint," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4322–4334, Nov. 2015.
[9] Z. Ding, S. Ming, and Y. Fu, "Latent low-rank transfer subspace learning for missing modality recognition," in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 1192–1198.
[10] Y. Xu, X. Fang, J. Wu, X. Li, and D. Zhang, "Discriminative transfer subspace learning via low-rank and sparse representation," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 850–863, Feb. 2016.
[11] Z. Ding and Y. Fu, "Robust transfer metric learning for image classification," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 660–670, Feb. 2017.
[12] S. Shekhar, V. M. Patel, H. V. Nguyen, and R. Chellappa, "Generalized domain-adaptive dictionaries," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 361–368.
[13] B. Tan, E. Zhong, E. W. Xiang, and Q. Yang, "Multi-transfer: Transfer learning with multiple views and multiple sources," *Statist. Anal. Data Mining*, vol. 7, no. 4, pp. 282–293, 2014.
[14] F. Wu and Y. Huang, "Collaborative multi-domain sentiment classification," in *Proc. IEEE Int. Conf. Data Mining*, Nov. 2015, pp. 459–468.
[15] K. Zhang, M. Gong, and B. Schölkopf, "Multi-source domain adaptation: A causal view," in *Proc. Assoc. Adv. Artif. Intell.*, 2015, pp. 3150–3157.
[16] A. Khosla, T. Zhou, T. Malisiewicz, A. A. Efros, and A. Torralba, "Undoing the damage of dataset bias," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 158–171.
[17] K. Muandet, D. Balduzzi, and B. Schölkopf, "Domain generalization via invariant feature representation," in *Proc. Int. Conf. Mach. Learn.*, vol. 28. 2013, pp. 10–18.
[18] C. Fang, Y. Xu, and D. N. Rockmore, "Unbiased metric learning: On the utilization of multiple datasets and Web images for softening bias," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1657–1664.
[19] Z. Xu, W. Li, L. Niu, and D. Xu, "Exploiting low-rank structure from latent domains for domain generalization," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 628–643.
[20] M. Ghifary, W. B. Kleijn, M. Zhang, and D. Balduzzi, "Domain generalization for object recognition with multi-task autoencoders," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2551–2559.
[21] L. Niu, W. Li, and D. Xu, "Visual recognition by learning from Web data: A weakly supervised domain generalization approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 2774–2783.

[22] C. Gan, T. Yang, and B. Gong, "Learning attributes equals multi-source domain generalization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 87–97.

[23] M. Ghifary, D. Balduzzi, W. B. Kleijn, and M. Zhang, "Scatter component analysis: A unified framework for domain adaptation and domain generalization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 7, pp. 1414–1430, Jul. 2017.

[24] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[25] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 184–199.

[26] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 392–407.

[27] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko, "Simultaneous deep transfer across domains and tasks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 4068–4076.

[28] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 59, pp. 1–35, 2016.

[29] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 97–105.

[30] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 136–144.

[31] Z. Ding and Y. Fu, "Low-rank common subspace for multi-view learning," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2014, pp. 110–119.

[32] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 815–823.

[33] Y. Kong, Z. Ding, J. Li, and Y. Fu, "Deeply learned view-invariant features for cross-view action recognition," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 3028–3037, Jun. 2017.

[34] H. Rahmani and A. Mian, "Learning a non-linear knowledge transfer model for cross-view action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 2458–2466.

[35] S. Wang, Z. Ding, and Y. Fu, "Feature selection guided auto-encoder," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 2725–2731.

[36] M. Shao, S. Li, Z. Ding, and Y. Fu, "Deep linear coding for fast graph clustering," in *Proc. Int. Joint Conf. Artif. Intell.*, 2015, pp. 3798–3804.

[37] H. Zhao, Z. Ding, and Y. Fu, "Multi-view clustering via deep matrix factorization," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 2921–2927.

[38] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proc. 32nd Int. Conf. Mach. Learn.*, 2015, pp. 1180–1189.

[39] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan, "Domain separation networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 343–351.

[40] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[41] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3320–3328.

[42] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.

[43] Y. Zhang, Z. Jiang, and L. S. Davis, "Learning structured low-rank representations for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 676–683.

[44] K. Tang, R. Liu, Z. Su, and J. Zhang, "Structure-constrained low-rank representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 12, pp. 2167–2179, Dec. 2014.

[45] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optim.*, vol. 20, no. 4, pp. 1956–1982, 2010.

[46] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, Feb. 2011.

[47] X. Shu, G.-J. Qi, J. Tang, and J. Wang, "Weakly-shared deep transfer networks for heterogeneous-domain knowledge propagation," in *Proc. 23rd ACM Int. Conf. Multimedia*, 2015, pp. 35–44.

[48] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[49] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[50] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 1521–1528.
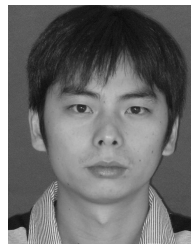
Zhengming Ding (S'14) received the B.Eng. degree in information security and the M.Eng. degree in computer software and theory from the University of Electronic Science and Technology of China, China, in 2010 and 2013, respectively. He is currently pursuing Ph.D. degree with the Department of Electrical and Computer Engineering, Northeastern University, USA. His research interests include machine learning and computer vision. Specifically, he devotes himself to develop scalable algorithms for challenging problems in transfer learning and deep learning scenario. He was the recipient of the Student Travel Grant of IJCAI 16, AAAI 16, ACM MM 14, and ICDM 14. He received the National Institute of Justice Fellowship. He was the recipient of the Best Paper Award. He has served as the reviewers for IEEE journals: IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS and IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE. He is an AAAI student member.
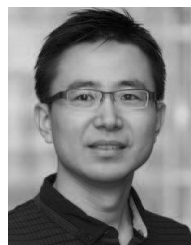
Yun Fu (S'07–M'08–SM'11) received the B.Eng. degree in information engineering and the M.Eng. degree in pattern recognition and intelligence systems from Xi'an Jiaotong University, China, respectively, the M.S. degree in statistics, and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana–Champaign, respectively. Since 2012, he is an Interdisciplinary Faculty Member affiliated with the College of Engineering and with the College of Computer and Information Science, Northeastern University. His research interests are machine learning, computational intelligence, big data mining, computer vision, pattern recognition, and cyber-physical systems. He has extensive publications in leading journals, books/book chapters and international conferences/workshops. He received seven Prestigious Young Investigator Awards from NAE, ONR, ARO, IEEE, INNS, UIUC, Grainger Foundation; seven Best Paper Awards from IEEE, IAPR, SPIE, SIAM; three major Industrial Research Awards from Google, Samsung, and Adobe. He serves as an Associate Editor, Chairs, PC member, and Reviewer of many top journals and international conferences/workshops. He is currently an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEANING SYSTEMS. He is fellow of IAPR, a Lifetime Senior Member of ACM, and SPIE, Lifetime Member of AAAI, OSA, and Institute of Mathematical Statistics, member of ACM Future of Computing Academy, Global Young Academy, INNS, and Beckman Graduate Fellow during 2007–2008.