Visual Recognition by Learning From Web Data via Weakly Supervised Domain Generalization

Li Niu, Wen Li, Dong Xu, *Senior Member, IEEE*, and Jianfei Cai, *Senior Member, IEEE*

***Abstract*— In this paper, a weakly supervised domain generalization (WSDG) method is proposed for real-world visual recognition tasks, in which we train classifiers by using Web data (e.g., Web images and Web videos) with noisy labels. In particular, two challenging problems need to be solved when learning robust classifiers, in which the first issue is to cope with the label noise of training Web data from the source domain, while the second issue is to enhance the generalization capability of learned classifiers to an arbitrary target domain. In order to handle the first problem, the training samples within each category are partitioned into clusters, where we use one bag to denote each cluster and instances to denote the samples in each cluster. Then, we identify a proportion of good training samples in each bag and train robust classifiers by using the good training samples, which leads to a multi-instance learning (MIL) problem. In order to handle the second problem, we assume that the training samples possibly form a set of hidden domains, with each hidden domain associated with a distinctive data distribution. Then, for each category and each hidden latent domain, we propose to learn one classifier by extending our MIL formulation, which leads to our WSDG approach. In the testing stage, our approach can obtain better generalization capability by effectively integrating multiple classifiers from different latent domains in each category. Moreover, our WSDG approach is further extended to utilize additional textual descriptions associated with Web data as privileged information (PI), although testing data do not have such PI. Extensive experiments on three benchmark data sets indicate that our newly proposed methods are effective for real-world visual recognition tasks by learning from Web data.**

***Index Terms*— Domain generalization, learning using privileged information (LUPI), multi-instance learning (MIL).**

I. Introduction

THE research interest on utilizing Web images/videos as the training data to recognize new images/videos grows rapidly in recent years. Nevertheless, as mentioned in [1], the data distributions of training and testing samples are most likely to be different, which leads to the data set bias problem [1]. In order to tackle this issue, researchers have proposed abundant domain adaptation approaches for different computer vision tasks [2]–[11].

We follow the terminology in the field of domain adaptation, that is, the training and testing data sets are referred to as the source and target domains, respectively. In the case that the target domain data are unseen in the training stage, the problem is called domain generalization. Compared with domain adaptation, domain generalization targets at learning robust classifiers that have excellent generalization ability to an arbitrary target domain [12]–[15], which is very important in the real-world visual recognition tasks. For instance, different data sets consisting of photos/videos captured by different users with different cameras can be treated as different target domains, which have different visual feature distributions. Due to privacy issues, some users may be reluctant to upload their photos/videos to public websites, and thus, we are lacking of data from some target domains. In such case, it is crucial to develop the effective approaches for domain generalization, which do not require the target domain data during the training stage.

In this paper, the domain generalization problem is explored by utilizing freely available source domain data (i.e., Web images/videos). In particular, a novel method called weakly supervised domain generalization (WSDG) is developed in Section III. Two important issues are considered: 1) Web images/videos are often associated with inaccurate labels, i.e., they are loosely labeled and 2) the data distributions between the source domain and the target domain are usually quite different. Moreover, during the training stage, the target domain data are generally unseen.

To tackle the inaccurate labels of training images/videos, the training samples within each category are first partitioned into clusters. We use one bag to denote each cluster and instances to denote the samples in each cluster. We only have the labels of each training bags, but the instance labels in each training bag are unknown. Inspired by the multi-instance learning (MIL) works, we use a proportion of good samples selected from a bag to represent the bag under the assumption that the training bags from different categories can be well distinguished. We then unify learning robust classifiers and selecting good training samples for each bag in a multiclass multi-instance formulation.

On the other hand, inspired by the recent works [14], [16], [17], we conjecture that the training Web samples possibly form a set of hidden latent domains, each of which has a different data distribution. Thus, we apply the existing technology to discover multiple latent domains and then,

Manuscript received October 27, 2015; revised January 27, 2016 and March 30, 2016; accepted April 3, 2016. Date of publication June 1, 2016; date of current version August 15, 2017.

L. Niu is with the Interdisciplinary Graduate School, Nanyang Technological University, Singapore 639798 (e-mail: lniu002@ntu.edu.sg).

W. Li is with the Computer Vision Laboratory, ETH Zürich, Zürich 8092, Switzerland (e-mail: liwen@vision.ee.ethz.ch).

D. Xu is with the School of Electrical and Information Engineering, The University of Sydney, Sydney, NSW 2006, Australia (e-mail: dong.xu@sydney.edu.au).

J. Cai is with the School of Computer Engineering, Nanyang Technological University, Singapore 639798 (e-mail: asjfcai@ntu.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2016.2557349

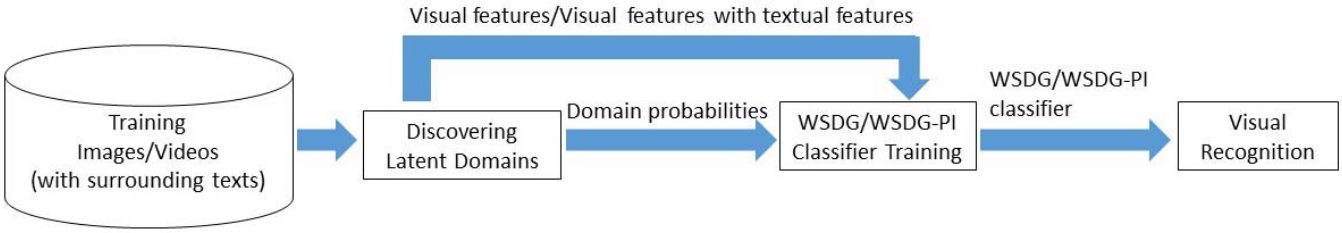


Fig. 1. Flowchart of our visual recognition methods. The flowchart consists of an approach to discover the latent domains, which learns the probabilities that each training sample comes from each latent domain, and a classification method WSDG/WSDG-PI, which learns one classifier for each category and each hidden latent domain. For our WSDG method, only the visual features are required as the input, while for our WSDG-PI method, the visual features together with the textual features are required as the input.

learn one classifier for each category and each latent domain. Since the training samples for learning classifiers for each category and each latent domain are relatively more coherent, the integrated classifier obtained by fusing multiple classifiers from all categories is more robust to various data distributions. As a result, we expect that the integrated classifier will have good generalization ability to an arbitrary target domain. Note that for each training bag, we just use a proportion of training samples to learn the classifiers, and thus, we propose to identify the training instances that have more distinctive data distributions by using a maximum mean discrepancy (MMD)-based regularizer.

In the testing stage, for each testing sample, we select the classifier corresponding to the highest response among all the classifiers from different latent domains in each category, which can be intuitively explained as we select the most matched latent domain for each testing sample. As a result, the data distribution mismatch between training samples and testing samples can be reduced.

In addition, the Web data are usually accompanied by additional textual information (e.g., tags, descriptions, and captions), which can be used as privileged information (PI) [9], [18], though these textual features are not available for the testing data. In Section IV, our WSDG method is extended by utilizing such PI, which is referred to as WSDG-PI. The flowchart of our WSDG and WSDG-PI methods is shown in Fig. 1. In Section V, the extensive experimental results clearly show the effectiveness of our approaches.

Our major contributions can be summarized as follows.

1) To the best of our knowledge, this paper is the first one to explore the domain generalization problem under the weakly supervised learning setting.
2) An effective WSDG approach is developed for domain generalization.
3) Our WSDG approach is further extended to WSDG-PI by utilizing PI (i.e., additional textual descriptions).

We would like to point out that this paper is extended from our preliminary conference paper [19] with the following major differences. First, we provide the detailed formulation and solution for WSDG-PI (see Section IV), in which textual features are utilized as PI. In addition, in Section V-B, we conduct more experiments to analyze why removing outliers in our WSDG method helps learn a better classifier and discover more distinctive latent domains. We also perform the in-depth study for our methods, such as the robustness with respect to parameters, the training time, and the scalability in Sections V-D, V-E, and V-F, respectively. Finally, in order to improve the event recognition performance, we employ the new features and use the aligned space-time pyramid matching (ASTPM) method to better calculate the distances between video clips on the Kodak and Columbia Consumer Video (CCV) data sets.

II. Related Work

MIL is in the sense that we partition the training samples into clusters and use bag (resp., instances) to denote each cluster (resp., the samples in each bag). A set of MIL approaches were developed in [20]–[22]. In multi-instance (mi-SVM) [21], the support vector machine (SVM) classifier is trained at each iteration based on the inferred instance labels from the previous iteration. In key-instance (KI-SVM) [22], the key instances inside each bag are used as the representatives of the bag. Nevertheless, these methods were proposed without taking the data distribution mismatch between two domains into consideration, so that the learned classifiers may not generalize well to the arbitrary target domain.

Domain generalization is another relevant research topic. For domain generalization, a domain invariant feature representation was learned in [13], while an SVM-based approach was proposed in [12]. Xu *et al.* [14] exploited the low-rank structure of source latent domains based on exemplar classifiers. When we have target domain data in the training process, domain adaptation approaches can be used to reduce the domain distribution mismatch. The recently developed domain adaptation approaches can be classified into classifier-based methods [7], [8], [23], instance-reweighting methods [24], and feature-based methods [6], [25]–[28]. Some works [29]–[32] applied low-rank techniques for domain adaptation. In particular, the transformed source domain samples are expected to be linearly constructed by the target domain samples in [30]. In [31], both the source and target domain data are projected to the common subspace, where each target domain sample can be linearly constructed by the source domain samples. Ding *et al.* [32] proposed an iterative approach, in which the transformed source domain is treated as the dictionary to reconstruct the transformed data from both domains at each iteration. Ding *et al.* [29] proposed to recover the missing modality in the target domain under a transfer

learning framework. For more technical details, please refer to a recent survey on domain adaptation [33].

This paper is also related to several recent approaches, which can discover latent domains [14], [16], [17], [34]. In these works, latent domains are discovered based on a clustering approach (i.e., [16]), the MMD criterion (i.e., [17]), or mutual information (i.e., [34]). After discovering the latent domains, these works train an SVM classifier or K-nearest neighbor classifier for each hidden latent domain, and then, all the classifiers learned for different latent domains are integrated to predict the testing samples. Unlike the above works, we jointly learn multiple classifiers for all latent domains and categories, which can be effectively integrated in a unified formulation.

The subcategorization problem [35] is also related to our work, since each category often consists of multiple subcategories. Recently, some works were proposed to integrate MIL with subcategorization [36]–[38]. Nevertheless, the domain distribution mismatch between the training data and the testing data was not considered in these works, which is quite different from the domain generalization problem discussed in this paper.

Finally, learning using PI (LUPI) [18] is also related to our work. In the LUPI paradigm, the training samples are associated with additional features that are not available for the testing data, which are referred to as PI. In some recent works [9], [39]–[41], PI was exploited for different computer vision tasks. In [39], a rank SVM method was proposed to rank Web images based on PI. In [40] and [41], PI was incorporated into distance metric learning. However, these works assume that the training data and the testing data are with the same data distribution, while this assumption does not hold in our setting. In [9], a new method was proposed to simultaneously handle label noise, take advantage of PI, and reduce the domain distribution mismatch. However, the target domain data are required in [9], while they are assumed to be unseen in this paper.

III. Weakly Supervised Domain Generalization

In this section, a novel WSDG approach is proposed, which simultaneously identifies good samples and learns robust classifiers. For consistent presentation, we always use an uppercase/lowercase letter in boldface to represent a matrix/vector, and the superscript $'$ to represent the transpose of a matrix/vector. We denote the elementwise product between two matrices by $\mathbf{A} \circ \mathbf{B}$, and $\mathbf{1}_n$ (resp., $\mathbf{0}_n$) $\in \mathbb{R}^n$ as the n-dimensional column vectors containing all ones (resp., zeros). When the dimensionality is obvious, we use **0/1** instead of $\mathbf{0}_n/\mathbf{1}_n$ for simplicity. The inequality $\mathbf{a} \leq \mathbf{b}$ stands for $a_i \leq b_i$ for $i = 1, \ldots, n$. Moreover, we denote the indicator function as $\delta(a = b)$, in which $\delta(a = b) = 0$ if $a \neq b$, and $\delta(a = b) = 1$, otherwise.

Assuming that there are N training samples from C categories in the source domain, the source domain data are denoted by $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, where $\mathbf{x}_i$ is the i-th training sample, with its corresponding category label $y_i \in \{1, \ldots, C\}$.

Next, we first provide a brief introduction on how to discover latent domains using the existing technology [17]. Then, we develop a multiclass MIL approach without considering the latent domain issues. Last, we integrate the latent domain discovery technique into our multiclass MIL formulation.

A. Discovering Latent Domains

In this paper, the existing latent domain discovering technique in [17] is adopted, which relies on the MMD criterion. We use $\pi_{i,m} \in \{0, 1\}$ to indicate whether each sample belongs to each latent domain. In particular, $\pi_{i,m} = 1$ if $\mathbf{x}_i$ comes from the m-th latent domain, and $\pi_{i,m} = 0$ otherwise. We denote $N_m = \sum_{i=1}^{N} \pi_{i,m}$ as the number of training samples from the m-th hidden latent domain. The approach in [17] aims to maximize the sum of MMDs (SMMDs) between each pair of latent domains and expects the discovered latent domains to be as distinctive as possible, that is

$$\max_{\pi_{i,m}} \sum_{m \neq \tilde{m}} \left\| \frac{1}{N_m} \sum_{i=1}^{N} \pi_{i,m} \phi(\mathbf{x}_i) - \frac{1}{N_{\tilde{m}}} \sum_{i=1}^{N} \pi_{i,\tilde{m}} \phi(\mathbf{x}_i) \right\|^2 \quad (1)$$

where $\phi(\cdot)$ is the feature mapping function, which is induced by a kernel $\mathbf{K} \in \mathbb{R}^{N \times N}$ on the training data (i.e., $\mathbf{K} = [K_{i,j}]$ with $K_{i,j} = \phi(\mathbf{x}_i)' \phi(\mathbf{x}_j)$). Let $\beta_{i,m} = (\pi_{i,m}/N_m)$ and $\boldsymbol{\beta}_m = [\beta_{1,m}, \ldots, \beta_{N,m}]'$, we can relax the above problem according to [17] as

$$\max_{\boldsymbol{\beta}} \sum_{m \neq \tilde{m}} (\boldsymbol{\beta}_m - \boldsymbol{\beta}_{\tilde{m}})' \mathbf{K} (\boldsymbol{\beta}_m - \boldsymbol{\beta}_{\tilde{m}}) \quad (2)$$

$$\text{s.t. } \frac{1}{N} \leq \sum_{m=1}^{M} \beta_{i,m} \leq \frac{1}{C}, \quad \forall i, \quad (3)$$

$$\sum_{i=1}^{N} \delta(y_i = c) \beta_{i,m} = \frac{1}{N} \sum_{i=1}^{N} \delta(y_i = c), \quad \forall c, m \quad (4)$$

$$\sum_{i=1}^{N} \beta_{i,m} = 1, \quad \beta_{i,m} \geq 0, \quad \forall i, m \quad (5)$$

where the first constraint in (3) is to guarantee that at least one training sample is selected in each hidden latent domain per category, the second constraint in (4) is to ensure that the class distribution in the whole source domain is consistent with that in each hidden latent domain, and the third constraint in (5) can be easily obtained based on the definitions of $\beta_{i,m}$ and $\pi_{i,m}$. Interested readers can refer to [17] for more technical details. Note that the above quadratic programming problem is nonconvex, which is not easy to be optimized. However, we can still utilize the existing solver in [42] to achieve satisfactory performance.

After latent domains are discovered by optimizing the objective function in (2), one classifier is learned for each category and each hidden latent domain. Then, a set of classifiers for each category are integrated based on the learned $\beta_{i,m}$'s. Next, we develop a novel multiclass MIL formulation to cope with the label noise, followed by extending our proposed formulation with the learned $\beta_{i,m}$'s to improve the generalization capability of the learnt classifiers to any arbitrary target domain.

B. Formulation

1) Learning With Weakly Supervised Information: In MIL, training samples are partitioned into a set of bags with explicit bag labels, while the accurate labels of training instances in each bag are unknown. Inspired by MIL, the training samples within each category in our case are partitioned into training bags, i.e., $\{(\mathcal{B}_l, Y_l)|l = 1, \ldots, L\}$. As the training samples are obtained by using category names as searching queries, the bag label $Y_l \in \{1, \ldots, C\}$ is the corresponding query name. Similar to [20], each positive bag is assumed to have at least a certain portion of true positive instances. Thus, we use the ratio η to denote the proportion of true positive training instances in each bag. Note that η can be estimated from some prior knowledge, similar to the conventional MIL methods.

To learn robust classifiers, we select good samples from each training bag by removing the outliers with inaccurate class labels. In particular, we use a binary indicator $h_i \in \{0, 1\}$ to indicate whether each training sample $\mathbf{x}_i$ is selected. To be exact, $h_i = 0$ if $\mathbf{x}_i$ is not selected, and $h_i = 1$, otherwise. We define $\mathbf{h} = [h_1, \ldots, h_N]'$ as the indicator vector, and use $\mathcal{H} = \{\mathbf{h}| \sum_{i \in I_l} h_i = \eta|\mathcal{B}_l|, \forall l\}$ to represent the feasible set of $\mathbf{h}$, where I_l represents the set of instance indices in $\mathcal{B}_l$, and $|\mathcal{B}_l|$ denotes the cardinality of $\mathcal{B}_l$.

Based on multiclass SVM [43], we propose our multiclass MIL formulation as follows. In particular, C classifiers $\{f_c(\mathbf{x})|c = 1, \ldots C\}$ are to be learned, where each classifier[1] can be represented as $f_c(\mathbf{x}) = (\mathbf{w}_c)'\phi(\mathbf{x})$. Inspired by the MIL learning method KI-SVM [22] as well as multiclass SVM [43], we propose to jointly learn $\mathbf{h}$ and C classifiers as

$$\min_{\substack{\mathbf{h}\in\mathcal{H}\\ \mathbf{w}_c,\xi_l}} \frac{1}{2}\sum_{c=1}^{C}\|\mathbf{w}_c\|^2 + C_1\sum_{l=1}^{L}\xi_l \tag{6}$$

$$\text{s.t. } \frac{1}{|\mathcal{B}_l|}\sum_{i\in I_l} h_i\left((\mathbf{w}_{Y_l})'\phi(\mathbf{x}_i) - (\mathbf{w}_{\tilde{c}})'\phi(\mathbf{x}_i)\right) \geq \eta - \xi_l, \quad \forall l, \tilde{c} \neq Y_l \tag{7}$$

$$\xi_l \geq 0, \quad \forall l, \tag{8}$$

where C_1 is a tradeoff parameter, and ξ_l's are slack variables. We enforce the total decision value of each bag $\mathcal{B}_l$ obtained based on the classifier corresponding to its category to be larger than those obtained by using the classifiers for the other categories by using the constraint in (7). Intuitively, we expect to identify good instances within each training bag to reduce the bag-level loss.

Note that multiclass SVM [43] is a special case of the problem in (6) with the bag size $|\mathcal{B}_l|$ being 1. Besides, when there are two categories, (6) becomes the MIL learning problem in KI-SVM [22] with slight modifications.

2) Weakly Supervised Domain Generalization: Now, considering that the training samples in the source domain come from M latent domains, we propose to enhance the generalization capability of the learned classifiers by integrating the classifiers from all latent domains for each category.

[1]We omit the bias term here for better representation. Instead, the feature of each training sample is augmented with an extra element of 1.

To be exact, a total of, $C \times M$ classifiers $\{f_{c,m}(\mathbf{x})|c = 1, \ldots C$, and $m = 1, \ldots, M\}$ are to be learned, where $f_{c,m}(\mathbf{x}) = (\mathbf{w}_{c,m})'\phi(\mathbf{x})$ represents the classifier corresponding to the m-th hidden latent domain and the c-th category. Then, we can obtain the decision function on $\mathbf{x}_i$ for each category by integrating the learned classifiers from multiple latent domains as $f_c(\mathbf{x}_i) = \sum_{m=1}^{M}\hat{\beta}_{i,m} f_{c,m}(\mathbf{x}_i)$, where $\hat{\beta}_{i,m}$ is the probability that the i-th training sample comes from the m-th hidden latent domain. $\hat{\beta}_{i,m}$ is defined as $\hat{\beta}_{i,m} = (\beta_{i,m}/\sum_{m=1}^{M}\beta_{i,m})$, where $\beta_{i,m}$'s are precomputed by solving (2). In summary, we expect to learn $C \times M$ classifiers to make the integrated classifier $f_c(\mathbf{x}_i)$'s as discriminative as possible.

Note that the latent domain discovery technique in [17] was proposed for clean training data without label noise. To deal with the training data with noisy labels, while maximizing (2), we need to seek for an optimal $\mathbf{h}$ value to remove outliers. With $\mathbf{B} = [\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_M] \in \mathbb{R}^{N\times M}$, the objective function in (2) can be written as $\rho(\mathbf{B}, \mathbf{K}) = \sum_{m\neq\tilde{m}}(\boldsymbol{\beta}_m - \boldsymbol{\beta}_{\tilde{m}})'\mathbf{K}(\boldsymbol{\beta}_m - \boldsymbol{\beta}_{\tilde{m}})$. In order to learn an optimal $\mathbf{h}$ value, we add a regularizer $\rho(\mathbf{B}, \mathbf{K} \circ (\mathbf{h}\mathbf{h}'))$ and derive the complete objective function of our proposed WSDG approach as

$$\min_{\substack{\mathbf{h}\in\mathcal{H}\\ \mathbf{w}_{c,m},\xi_l}} \frac{1}{2}\sum_{c=1}^{C}\sum_{m=1}^{M}\|\mathbf{w}_{c,m}\|^2 + C_1\sum_{l=1}^{L}\xi_l - C_2\,\rho(\mathbf{B}, \mathbf{K} \circ (\mathbf{h}\mathbf{h}')) \tag{9}$$

$$\text{s.t. } \frac{1}{|\mathcal{B}_l|}\sum_{i\in I_l} h_i\left(\sum_{m=1}^{M}\hat{\beta}_{i,m}(\mathbf{w}_{Y_l,m})'\phi(\mathbf{x}_i) - (\mathbf{w}_{\tilde{c},\tilde{m}})'\phi(\mathbf{x}_i)\right) \geq \eta - \xi_l, \quad \forall l, \tilde{m}, \tilde{c} \neq Y_l \tag{10}$$

$$\xi_l \geq 0, \quad \forall l, \tag{11}$$

where C_2 is a tradeoff parameter. The explanation for the constraint (10) is similar to that for (7) except that we replace $(\mathbf{w}_{Y_l})'\phi(\mathbf{x}_i)$ in (7) with $\sum_{m=1}^{M}\hat{\beta}_{i,m}(\mathbf{w}_{Y_l,m})'\phi(\mathbf{x}_i)$ and $(\mathbf{w}_{\tilde{c}})'\phi(\mathbf{x}_i)$ with $(\mathbf{w}_{\tilde{c},\tilde{m}})'\phi(\mathbf{x}_i)$.

Essentially, we train one classifier for each category and each hidden latent domain. This is mainly because the data distributions of the training samples from one category and one hidden latent domain are generally more similar [17], which is easier for us to learn a discriminant classifier. In the testing stage, given a testing sample $\mathbf{x}$, we predict its label by

$$\arg\max_{c}\left(\max_{m}\mathbf{w}_{c,m}{}'\phi(\mathbf{x})\right). \tag{12}$$

Namely, for each category, we attempt to seek for the most matched hidden latent domain for a given testing sample, whose classifier achieves the largest decision value from all the latent domains. In this way, we conjecture that the integrated classifiers have good generalization ability to the testing data from the arbitrary target domain.

C. Optimization

The nonconvex mixed integer problem in (9) is nontrivial to solve. According to some recent works on MIL [20], [22], the dual form of (9) can be relaxed as a multiple kernel learning (MKL) problem, which shares a similar solution as that in [44]. Next, we introduce how to relax the dual form

of (9) and, then, discuss how to solve the relaxed problem in detail.

1) Reformulation in Dual Form:

Proposition 1: The dual form of (9) is

$$\begin{aligned}\min_{\mathbf{h}\in\mathcal{H}}\max_{\boldsymbol{\alpha}} \quad & -\frac{1}{2}\boldsymbol{\alpha}'\mathbf{Q}^{\mathbf{h}}\boldsymbol{\alpha}+\boldsymbol{\zeta}'\boldsymbol{\alpha}-C_2\,\rho(\mathbf{B},\mathbf{K}\circ(\mathbf{h}\mathbf{h}')) \\ \text{s.t.} \quad & \sum_{c,m}\alpha_{l,c,m}=C_1, \quad \forall l, \\ & \alpha_{l,c,m}\geq 0, \quad \forall l,c,m \end{aligned} \tag{13}$$

where $\boldsymbol{\alpha}\in\mathbb{R}^{\tilde{D}}$ is a vector containing dual variables $\alpha_{l,c,m}$, $\tilde{D}=L\cdot C\cdot M$, $\boldsymbol{\zeta}\in\mathbb{R}^{\tilde{D}}$ is a vector, in which each entry $\zeta_{l,c,m}=0$ if $c=Y_l$ and $\zeta_{l,c,m}=\eta$ otherwise. Each element in the matrix $\mathbf{Q}^{\mathbf{h}}\in\mathbb{R}^{\tilde{D}\times\tilde{D}}$ can be obtained based on $Q^{\mathbf{h}}_{u,v}=(1/|\mathcal{B}_l||\mathcal{B}_{\tilde{l}}|)\ \sum_{i\in\mathcal{I}_l}\sum_{j\in\mathcal{I}_{\tilde{l}}}h_i h_j\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)\gamma(i,j,c,\tilde{c},m,\tilde{m})$, $v=(\tilde{l}-1)\cdot C\cdot M+(\tilde{c}-1)\cdot M+\tilde{m}$ and $u=(l-1)\cdot C\cdot M+(c-1)\cdot M+m$ are the indices, and $\gamma(i,j,c,\tilde{c},m,\tilde{m})=[1-\delta(c=y_i)][1-\delta(\tilde{c}=y_j)][\delta(y_i=y_j)\sum_{q=1}^{M}\hat{\beta}_{i,q}\hat{\beta}_{j,q}+\delta(c=\tilde{c})\delta(m=\tilde{m})]-[1-\delta(c=\tilde{c})]\{[1-\delta(c=y_i)]\delta(c=y_j)\hat{\beta}_{j,m}+[1-\delta(\tilde{c}=y_j)]\delta(\tilde{c}=y_i)\hat{\beta}_{i,\tilde{m}}\}$. The proof of Proposition 1 can be found in the Appendix.

The problem in (13) is a mixed integer programming problem, which is difficult to be solved. Inspired by [20] and [22], we use an alternating optimization approach to find the optimal combination coefficients of $\mathbf{h}_t\mathbf{h}_t'$'s given all feasible $\mathbf{h}_t\in\mathcal{H}$, i.e., $\sum_{\mathbf{h}_t\in\mathcal{H}}d_t\mathbf{h}_t\mathbf{h}_t'$ with d_t being the combination the indicator vector $\mathbf{h}$. For ease of presentation, we denote $T=|\mathcal{H}|$, $\mathbf{d}=[d_1,\ldots,d_T]'$, the feasible set of $\mathbf{d}$ as $\mathcal{D}=\{\mathbf{d}|\mathbf{d}'\mathbf{1}=1,\mathbf{d}\geq 0\}$, and the feasible set of $\boldsymbol{\alpha}$ in (13) as $\mathcal{A}$. Then, we arrive at the following optimization problem:

$$\begin{aligned}\min_{\mathbf{d}\in\mathcal{D}}\max_{\boldsymbol{\alpha}\in\mathcal{A}} \quad & -\frac{1}{2}\sum_{t=1}^{T}d_t\boldsymbol{\alpha}'\mathbf{Q}^{\mathbf{h}_t}\boldsymbol{\alpha}+\boldsymbol{\zeta}'\boldsymbol{\alpha} \\ & -C_2\sum_{t=1}^{T}d_t\rho\big(\mathbf{B},\mathbf{K}\circ\big(\mathbf{h}_t\mathbf{h}_t'\big)\big). \end{aligned} \tag{14}$$

Note that we move the sum operator over d_t outside $\mathbf{Q}^{\mathbf{h}_t}$ and $\rho(\mathbf{B},\mathbf{K}\circ(\mathbf{h}_t\mathbf{h}_t'))$, since both of them are linear terms of $\mathbf{h}_t\mathbf{h}_t'$. The above problem is similar to the MKL dual form when we treat each base kernel as $\mathbf{Q}^{\mathbf{h}_t}$. Therefore, we can solve it based on its following primal form, which is a convex optimization problem:

$$\begin{aligned}\min_{\mathbf{d}\in\mathcal{D},\mathbf{w}_t,\xi_l} \quad & \frac{1}{2}\sum_{t=1}^{T}\frac{\|\mathbf{w}_t\|^2}{d_t}+C_1\sum_{l=1}^{L}\xi_l \\ & -C_2\sum_{t=1}^{T}d_t\rho\big(\mathbf{B},\mathbf{K}\circ\big(\mathbf{h}_t\mathbf{h}_t'\big)\big) \end{aligned} \tag{15}$$

$$\text{s.t.} \quad \sum_{t=1}^{T}\mathbf{w}_t'\psi(\mathbf{h}_t,\mathcal{B}_l,c,m)\geq\zeta_{l,c,m}-\xi_l, \quad \forall l,c,m \tag{16}$$

where $\psi(\mathbf{h}_t,\mathcal{B}_l,c,m)$ is used to denote the feature mapping induced by $\mathbf{Q}^{\mathbf{h}_t}$, i.e., $\psi(\mathbf{h}_t,\mathcal{B}_l,c,m)'\psi(\mathbf{h}_t,\mathcal{B}_{\tilde{l}},\tilde{c},\tilde{m})=Q^{\mathbf{h}_t}_{u,v}$, in which $v=(\tilde{l}-1)\cdot C\cdot M+(\tilde{c}-1)\cdot M+\tilde{m}$, $u=(l-1)\cdot C\cdot M+(c-1)\cdot M+m$. In the following, we prove that the dual form of (15) is (14).

Proof: By introducing a dual variable $\alpha_{l,c,m}$ for each constraint in (16), we can write the Lagrangian form of (15) as

$$\begin{aligned}\mathcal{L}=&\frac{1}{2}\sum_{t=1}^{T}\frac{\|\mathbf{w}_t\|^2}{d_t}+C_1\sum_{l=1}^{L}\xi_l-C_2\sum_{t=1}^{T}d_t\rho\big(\mathbf{B},\mathbf{K}\circ\big(\mathbf{h}_t\mathbf{h}_t'\big)\big) \\ &-\sum_{l,c,m}\alpha_{l,c,m}\left(\sum_{t=1}^{T}\mathbf{w}_t'\psi(\mathbf{h}_t,\mathcal{B}_l,c,m)-\zeta_{l,c,m}+\xi_l\right). \end{aligned} \tag{17}$$

By setting the derivatives of $\mathcal{L}$ with respect to $\mathbf{w}_t$ and ξ_l as zeros, respectively, we obtain

$$\mathbf{w}_t=d_t\sum_{l,c,m}\alpha_{l,c,m}\psi(\mathbf{h}_t,\mathcal{B}_l,c,m), \quad \forall t, \tag{18}$$

$$\sum_{c,m}\alpha_{l,c,m}=C_1, \quad \forall l. \tag{19}$$

Finally, by substituting (18) and (19) back into (15), we reach the objective function in (14), which completes the proof.

2) Solution to (15): We solve the convex problem in (15) by updating $\mathbf{d}$ and $\{\mathbf{w}_t,\xi_l\}$ in an alternative way.

a) Update $\mathbf{d}$: When fixing $\{\mathbf{w}_t,\xi_l\}$, in order to solve $\mathbf{d}$, we introduce a dual variable τ for the constraint $\mathbf{d}'\mathbf{1}=1$ and derive the Lagrangian form of (15) as

$$\begin{aligned}\hat{\mathcal{L}}=&\frac{1}{2}\sum_{t=1}^{T}\frac{\|\mathbf{w}_t\|^2}{d_t}+C_1\sum_{l=1}^{L}\xi_l-C_2\sum_{t=1}^{T}d_t\rho\big(\mathbf{B},\mathbf{K}\circ\big(\mathbf{h}_t\mathbf{h}_t'\big)\big) \\ &-\sum_{l,c,m}\alpha_{l,c,m}\left(\sum_{t=1}^{T}\mathbf{w}_t'\psi(\mathbf{h}_t,\mathcal{B}_l,c,m)-\zeta_{l,c,m}+\xi_l\right) \\ &+\tau\left(\sum_{t=1}^{T}d_t-1\right). \end{aligned} \tag{20}$$

By setting the derivative of (20) with respect to each d_t to zero, we have

$$\tau=\frac{\|\mathbf{w}_t\|^2}{2d_t^2}+C_2\rho\big(\mathbf{B},\mathbf{K}\circ\big(\mathbf{h}_t\mathbf{h}_t'\big)\big), \quad \forall t=1,\ldots,T \tag{21}$$

which can be rewritten as

$$d_t=\frac{\|\mathbf{w}_t\|}{\sqrt{2\tau-2C_2\rho\big(\mathbf{B},\mathbf{K}\circ\big(\mathbf{h}_t\mathbf{h}_t'\big)\big)}}, \quad \forall t=1,\ldots,T. \tag{22}$$

Since the function on the right-hand side of (22) is monotonically decreasing with respect to τ and $\mathbf{d}'\mathbf{1}=1$, we first apply binary search to seek for the value τ, which satisfies the constraint $\sum_{t=1}^{T}d_t=1$, and then recover d_t's based on (22).

b) Update $\mathbf{w}_t$: When $\mathbf{d}$ is fixed, $\boldsymbol{\alpha}$ can be solved in the dual form (14) and $\mathbf{w}_t$ can be recovered by using (18). In particular, we can solve the problem in (14), which is a quadratic programming (QP) problem with respect to $\boldsymbol{\alpha}$, by employing quadratic programming solvers. Nevertheless, it is very time-consuming to use the existing QP solvers, which are not specifically designed for our problem with $L\cdot C\cdot M$

Algorithm 1 WSDG Algorithm

Input: The training data $\{(\mathcal{B}_l, Y_l)|_{l=1}^{L}\}$.

1: Initialize $t = 1$ and[2] $\mathcal{C} = \{\mathbf{h}_1\}$.
2: **repeat**
3: Set $t \leftarrow t + 1$.
4: Based on $\mathcal{H} = \mathcal{C}$, obtain $(\mathbf{d}, \boldsymbol{\alpha})$ by optimizing the MKL subproblem in (14).
5: Solving (26) to find the violated $\mathbf{h}_t$, which is added to the violation set (i.e., $\mathcal{C} \leftarrow \mathcal{C} \bigcup \mathbf{h}_t$).
6: **until** The objective of (14) converges.

Output: The learnt classifier $f(\mathbf{x})$.

variables. Thus, we solve this QP problem by using an efficient sequential minimal optimization (SMO) algorithm, based on [45] and [46].

3) Cutting-Plane Algorithm: When using the above alternating optimization algorithm, the major challenge is that there are too many base kernels. Inspired by the work on infinite kernel learning [47], we begin with a small number of base kernels and then add a new violating base kernel at each iteration iteratively, which is named as the cutting-plane algorithm. The MKL subproblem we need to solve at each iteration only has a small set of $\mathbf{h}$, so it becomes much more efficient to optimize the whole problem. In particular, we replace $\rho(\mathbf{B}, \mathbf{K} \circ (\mathbf{h}_t\mathbf{h}_t'))$ in (20) equivalently by using $\mathbf{h}_t'\mathbf{P}\mathbf{h}_t$ with $\mathbf{P} = \sum_{m \neq \tilde{m}} \mathbf{K} \circ ((\boldsymbol{\beta}_m - \boldsymbol{\beta}_{\tilde{m}})(\boldsymbol{\beta}_m - \boldsymbol{\beta}_{\tilde{m}})')$. By setting the derivatives of (20) with respect to $\{\mathbf{w}_t, \xi_t, d_t\}$ as zeros, we can rewrite (14) as:

$$\max_{\tau, \boldsymbol{\alpha} \in \mathcal{A}} \quad -\tau + \boldsymbol{\zeta}'\boldsymbol{\alpha} \tag{23}$$

$$\text{s.t. } \frac{1}{2}\boldsymbol{\alpha}'\mathbf{Q}^{\mathbf{h}_t}\boldsymbol{\alpha} + C_2\mathbf{h}_t'\mathbf{P}\mathbf{h}_t \leq \tau, \quad \forall t, \tag{24}$$

which has a large number of constraints.

To solve (23), we begin with only one constraint and add a new violated constraint at each iteration. In particular, since each constraint is related to an $\mathbf{h}_t$, the most violated constraint can be obtained by optimizing

$$\max_{\mathbf{h}} \frac{1}{2}\boldsymbol{\alpha}'\mathbf{Q}^{\mathbf{h}}\boldsymbol{\alpha} + C_2\mathbf{h}'\mathbf{P}\mathbf{h}. \tag{25}$$

After simple derivation, (25) can be rewritten as

$$\max_{\mathbf{h}} \mathbf{h}' \left(\frac{1}{2}\hat{\mathbf{Q}} \circ (\hat{\boldsymbol{\alpha}}\hat{\boldsymbol{\alpha}}') + C_2\mathbf{P}\right)\mathbf{h}, \tag{26}$$

where $\hat{\boldsymbol{\alpha}} \in \mathbb{R}^N$ is the shrinked vector of $\boldsymbol{\alpha}$ with its element $\hat{\alpha}_i = 1/|\mathcal{B}_l| \sum_{c,m} \alpha_{l,c,m}$ for each $i \in \mathcal{I}_l$, and $\hat{\mathbf{Q}} \in \mathbb{R}^{N \times N}$ is the shrinked matrix of $\mathbf{Q}$ with its element $\hat{Q}_{i,j} = \sum_{c,\tilde{c},m,\tilde{m}} \gamma(i, j, c, \tilde{c}, m, \tilde{m})\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)$. The problem in (26) can be solved approximately by enumerating the binary indicator vector $\mathbf{h}$ in a bag by bag fashion until there is no change in $\mathbf{h}$.

[2]We initialize $\mathbf{h}_1$ by assigning the entries corresponding to the top $\eta|\mathcal{B}_l|$ instances (i.e., with the highest decision values) in each bag $\mathcal{B}_l$ to 1, and the other entries to 0. In particular, we assign the labels of all training instances as their corresponding bag labels to train SVM classifiers, and then obtain the decision values of all training instances based on the learned SVM classifiers.

The proposed WSDG method is summarized in Algorithm 1.

IV. Weakly Supervised Domain Generalization Using Privileged Information

The Web data are generally accompanied by massive and informative contextual information (e.g., surrounding texts, tags, and captions). Although the contextual information is not available for the testing data, they can still be used as PI to improve the performance of the learned classifiers [9], [18]. Based on the above-mentioned idea, we extend our WSDG approach by further utilizing PI, i.e., the textual features extracted from the textual descriptions of Web images/videos, which leads to our WSDG-PI approach.

Let us denote the textual feature of the i-th training sample as $\mathbf{z}_i$. Inspired by the works in [9] and [18], we define $\tilde{f}_{c,m}(\mathbf{z}_i) = (\tilde{\mathbf{w}}_{c,m})'\tilde{\phi}(\mathbf{z}_i)$ as the slack function, in which $\tilde{\phi}$ is the feature mapping function for $\mathbf{z}_i$. For ease of presentation, we define the left-hand side of (10) as $F(\mathcal{B}_l, \tilde{c}, \tilde{m}) = (1/|\mathcal{B}_l|) \sum_{i \in I_l} h_i (\sum_{m=1}^{M} \hat{\beta}_{i,m} (\mathbf{w}_{Y_l,m})'\phi(\mathbf{x}_i) - (\mathbf{w}_{\tilde{c},\tilde{m}})'\phi(\mathbf{x}_i))$, and also define $\tilde{F}(\mathcal{B}_l, \tilde{c}, \tilde{m}) = (1/|\mathcal{B}_l|) \sum_{i \in I_l} h_i (\sum_{m=1}^{M} \hat{\beta}_{i,m} (\tilde{\mathbf{w}}_{Y_l,m})'\tilde{\phi}(\mathbf{z}_i) - (\tilde{\mathbf{w}}_{\tilde{c},\tilde{m}})'\tilde{\phi}(\mathbf{z}_i))$. Then, we formulate our WSDG-PI approach as

$$\min_{\substack{\mathbf{h} \in \mathcal{H}, \xi_l, \epsilon_l \\ \mathbf{w}_{c,m}, \tilde{\mathbf{w}}_{c,m}}} \frac{1}{2}\sum_{c=1}^{C}\sum_{m=1}^{M}\left(\|\mathbf{w}_{c,m}\|^2 + \lambda\|\tilde{\mathbf{w}}_{c,m}\|^2\right) + C_1\sum_{l=1}^{L}(\xi_l + \epsilon_l) - C_2\rho(B, \mathbf{K} \circ (\mathbf{h}\mathbf{h}')) + C_3\sum_{l=1}^{L}\sum_{\tilde{m}=1}^{M}\sum_{\tilde{c} \neq Y_l}\tilde{F}(\mathcal{B}_l, \tilde{c}, \tilde{m}) \tag{27}$$

$$\text{s.t. } F(\mathcal{B}_l, \tilde{c}, \tilde{m}) \geq \eta - \tilde{F}(\mathcal{B}_l, \tilde{c}, \tilde{m}) - \xi_l, \quad \forall l, \tilde{m}, \tilde{c} \neq Y_l \tag{28}$$

$$\tilde{F}(\mathcal{B}_l, \tilde{c}, \tilde{m}) \geq \eta - \epsilon_l, \quad \forall l, \tilde{m}, \tilde{c} \neq Y_l \tag{29}$$

$$\xi_l \geq 0, \quad \forall l, \tag{30}$$

$$\epsilon_l \geq 0, \quad \forall l, \tag{31}$$

where C_1, C_2, C_3, and λ are the tradeoff parameters, and ϵ_l is the slack variable introduced for the slack function $\tilde{F}(\mathcal{B}_l, \tilde{c}, \tilde{m})$. As discussed in [18], the slack function $\tilde{F}(\mathcal{B}_l, \tilde{c}, \tilde{m})$ plays the role of teacher by providing the explanations to the students, so we expect that $\tilde{F}(\mathcal{B}_l, \tilde{c}, \tilde{m})$ can well adjust the prediction of $F(\mathcal{B}_l, \tilde{c}, \tilde{m})$ for the sample those are difficult to be classified.

To derive the solution to the above problem, we write the dual form of (27) as

$$\begin{aligned}\min_{\mathbf{h} \in \mathcal{H}} \max_{\boldsymbol{\alpha}, \boldsymbol{\varsigma}} \quad & -\frac{1}{2}\boldsymbol{\alpha}'\mathbf{Q}^{\mathbf{h}}\boldsymbol{\alpha} - \frac{1}{2\lambda}(\boldsymbol{\alpha} + \boldsymbol{\varsigma} - C_3\mathbf{1})'\tilde{\mathbf{Q}}^{\mathbf{h}}(\boldsymbol{\alpha} + \boldsymbol{\varsigma} - C_3\mathbf{1}) \\ & + \boldsymbol{\zeta}'(\boldsymbol{\alpha} + \boldsymbol{\varsigma}) - C_2\rho(\mathbf{B}, \mathbf{K} \circ (\mathbf{h}\mathbf{h}')) \\ \text{s.t. } & \sum_{c,m}\alpha_{l,c,m} = C_1, \quad \forall l, \\ & \alpha_{l,c,m} \geq 0, \quad \forall l, c, m \\ & \sum_{c,m}\varsigma_{l,c,m} = C_1, \quad \forall l, \\ & \varsigma_{l,c,m} \geq 0, \quad \forall l, c, m \end{aligned} \tag{32}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^{\tilde{D}}$ is a vector containing the dual variables $\alpha_{l,c,m}$, $\tilde{D} = L \cdot C \cdot M$, $\boldsymbol{\varsigma} \in \mathbb{R}^{\tilde{D}}$ is a vector containing the dual

variables $\varsigma_{l,c,m}$, $\mathbf{Q}^{\mathbf{h}}$ is defined in the paragraph after (13), and $\tilde{\mathbf{Q}}^{\mathbf{h}}$ is defined by replacing $\phi(\mathbf{x})$ in $\mathbf{Q}^{\mathbf{h}}$ with $\tilde{\phi}(\mathbf{z})$. We leave the details of deriving the dual form of (27) in the Appendix.

Similar to solving (13), we optimize over the linear combination coefficients of $\mathbf{h}_t\mathbf{h}_t'$'s given all feasible $\mathbf{h}_t \in \mathcal{H}$, i.e., $\sum_{\mathbf{h}_t\in\mathcal{H}} d_t\mathbf{h}_t\mathbf{h}_t'$, where d_t is the combination coefficient for $\mathbf{h}_t\mathbf{h}_t'$. We denote $T = |\mathcal{H}|$, $\mathbf{d} = [d_1, \ldots, d_T]'$, $\mathcal{D} = \{\mathbf{d}|\mathbf{d}'\mathbf{1} = 1, \mathbf{d} \geq 0\}$ as the feasible set of $\mathbf{d}$, $\mathcal{A}$ as the feasible set of $\boldsymbol{\alpha}$ in (32), and $\mathcal{E}$ as the feasible set of $\boldsymbol{\varsigma}$ in (32). Then, we can arrive at the problem as follows:

$$\begin{aligned}\min_{\mathbf{d}\in\mathcal{D}}\max_{\substack{\boldsymbol{\alpha}\in\mathcal{A}\\ \boldsymbol{\varsigma}\in\mathcal{E}}} & -\frac{1}{2}\sum_{t=1}^{T} d_t\boldsymbol{\alpha}'\mathbf{Q}^{\mathbf{h}_t}\boldsymbol{\alpha} + \boldsymbol{\zeta}'(\boldsymbol{\alpha}+\boldsymbol{\varsigma}) \\ & -\frac{1}{2\lambda}\sum_{t=1}^{T} d_t(\boldsymbol{\alpha}+\boldsymbol{\varsigma}-C_3\mathbf{1})'\tilde{\mathbf{Q}}^{\mathbf{h}_t}(\boldsymbol{\alpha}+\boldsymbol{\varsigma}-C_3\mathbf{1}) \\ & -C_2\sum_{t=1}^{T} d_t\rho\big(\mathbf{B}, \mathbf{K}\circ\big(\mathbf{h}_t\mathbf{h}_t'\big)\big). \end{aligned} \tag{33}$$

We solve (33) based on its primal problem, which is an MKL problem and we can solve it in a similar way as in [44]

$$\begin{aligned}\min_{\substack{\mathbf{d}\in\mathcal{D},\xi_l,\epsilon_l\\ \mathbf{w}_t,\tilde{\mathbf{w}}_t}} & \frac{1}{2}\sum_{t=1}^{T}\frac{\|\mathbf{w}_t\|^2}{d_t} + \frac{\lambda}{2}\sum_{t=1}^{T}\frac{\|\tilde{\mathbf{w}}_t\|^2}{d_t} + C_1\sum_{l=1}^{L}(\xi_l+\epsilon_l) \\ & -C_2\sum_{t=1}^{T} d_t\rho\big(\mathbf{B}, \mathbf{K}\circ\big(\mathbf{h}_t\mathbf{h}_t'\big)\big) \\ & +C_3\sum_{t=1}^{T}\sum_{l,\tilde{m},\tilde{c}\neq Y_l}\tilde{\mathbf{w}}_t'\tilde{\psi}(\mathbf{h}_t, \mathcal{B}_l, \tilde{c}, \tilde{m}) \\ \text{s.t.} & \sum_{t=1}^{T}\mathbf{w}_t'\psi(\mathbf{h}_t, \mathcal{B}_l, \tilde{c}, \tilde{m}) \\ & \geq \zeta_{l,\tilde{c},\tilde{m}} - \sum_{t=1}^{T}\tilde{\mathbf{w}}_t'\tilde{\psi}(\mathbf{h}_t, \mathcal{B}_l, \tilde{c}, \tilde{m}) - \xi_l, \quad \forall l, \tilde{c}, \tilde{m} \\ & \sum_{t=1}^{T}\tilde{\mathbf{w}}_t'\tilde{\psi}(\mathbf{h}_t, \mathcal{B}_l, \tilde{c}, \tilde{m}) \geq \zeta_{l,\tilde{c},\tilde{m}} - \epsilon_l, \quad \forall l, \tilde{c}, \tilde{m} \end{aligned} \tag{34}$$

where $\psi(\mathbf{h}_t, \mathcal{B}_l, c, m)$ is defined after (15) and $\tilde{\psi}(\mathbf{h}_t, \mathcal{B}_l, c, m)$ is the feature mapping induced by $\tilde{\mathbf{Q}}^{\mathbf{h}_t}$, i.e., $\tilde{\psi}(\mathbf{h}_t, \mathcal{B}_l, c, m)'\tilde{\psi}(\mathbf{h}_t, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m}) = \tilde{Q}^{\mathbf{h}_t}_{u,v}$, in which $v = (\tilde{l}-1)\cdot C\cdot M + (\tilde{c}-1)\cdot M + \tilde{m}$ and $u = (l-1)\cdot C\cdot M + (c-1)\cdot M + m$. $\mathbf{w}_t$ and $\tilde{\mathbf{w}}_t$ are defined as

$$\mathbf{w}_t = d_t\sum_{l,c,m}\alpha_{l,c,m}\psi(\mathbf{h}_t, \mathcal{B}_l, c, m), \tag{35}$$

$$\tilde{\mathbf{w}}_t = d_t\sum_{l,c,m}(\alpha_{l,c,m}+\varsigma_{l,c,m}-C_3)\tilde{\psi}(\mathbf{h}_t, \mathcal{B}_l, c, m). \tag{36}$$

Similar as (15), the problem in (34) is also a convex problem, which can be solved by updating $\mathbf{d}$ and $\{\mathbf{w}_t, \tilde{\mathbf{w}}_t, \xi_l, \epsilon_l\}$ alternatively.

A. Update $\mathbf{d}$

When $\{\mathbf{w}_t, \tilde{\mathbf{w}}_t, \xi_l, \epsilon_l\}$ is fixed, we first introduce a dual variable τ for the constraint $\mathbf{d}'\mathbf{1} = 1$ to obtain the Lagrangian form of (34) similarly as (20). Setting the derivative of the Lagrangian form with respect to each d_t as zero, we arrive at

$$\tau = \frac{\|\mathbf{w}_t\|^2}{2d_t^2} + \lambda\frac{\|\tilde{\mathbf{w}}_t\|^2}{2d_t^2} + C_2\rho(\mathbf{B}, \mathbf{K}\circ(\mathbf{h}_t\mathbf{h}_t')), \quad \forall t, \tag{37}$$

which leads to

$$d_t = \sqrt{\frac{\|\mathbf{w}_t\|^2 + \lambda\|\tilde{\mathbf{w}}_t\|^2}{2\tau - 2C_2\rho(\mathbf{B}, \mathbf{K}\circ(\mathbf{h}_t\mathbf{h}_t'))}}, \quad \forall t. \tag{38}$$

Similar to (22), (38) is also monotonically decreasing with respect to τ. So we also use the binary search method to seek for τ, which satisfies $\sum_{t=1}^{T} d_t = 1$, and calculate d_t's by using (38).

Algorithm 2 WSDG-PI Algorithm

Input: The training data $\{(\mathcal{B}_l, Y_l)|_{l=1}^{L}\}$.

1: Initialize $t = 1$ and[3] $\mathcal{C} = \{\mathbf{h}_1\}$.
2: **repeat**
3: Set $t \leftarrow t+1$.
4: Obtain $(\mathbf{d}, \boldsymbol{\alpha}, \boldsymbol{\varsigma})$ by optimizing the MKL problem in (33) based on $\mathcal{H} = \mathcal{C}$.
5: Solve (39) to find the violated $\mathbf{h}_t$, which is added to the violation set (i.e., $\mathcal{C} \leftarrow \mathcal{C}\bigcup\mathbf{h}_t$).
6: **until** The objective of (33) converges.

Output: The learnt classifier $f(\mathbf{x})$.

B. Update $\{\boldsymbol{w}_t, \tilde{\boldsymbol{w}}_t, \xi_l, \epsilon_l\}$

When $\mathbf{d}$ is fixed, we solve $\boldsymbol{\alpha}$ and $\boldsymbol{\varsigma}$ in (33). In particular, we concatenate $\boldsymbol{\alpha}$ and $\boldsymbol{\varsigma}$ into a long vector $\boldsymbol{\vartheta}$, and thus, (33) becomes a QP problem with respect to $\boldsymbol{\vartheta}$. Since there are too many variables in $\boldsymbol{\vartheta}$, it is inefficient to be solved based on the QP solvers. Similar to Section III-C2, we use the SMO algorithm to solve (33).

Again, there are too many $\mathbf{h}_t\mathbf{h}_t'$'s when using the above alternating optimization procedure. Similar to Section III-C3, we employ the cutting-plane algorithm. In each iteration, we seek for the most violating indicator $\mathbf{h}$ by solving the following problem similarly as the one in (25):

$$\begin{aligned}\max_{\mathbf{h}} & \frac{1}{2}\boldsymbol{\alpha}'\mathbf{Q}^{\mathbf{h}}\boldsymbol{\alpha} + \frac{1}{2\lambda}(\boldsymbol{\alpha}+\boldsymbol{\varsigma}-C_3\mathbf{1})'\tilde{\mathbf{Q}}^{\mathbf{h}}(\boldsymbol{\alpha}+\boldsymbol{\varsigma}-C_3\mathbf{1}) \\ & +C_2\rho(\mathbf{B}, \mathbf{K}\circ(\mathbf{h}\mathbf{h}')). \end{aligned} \tag{39}$$

The whole algorithm of WSDG using PI (WSDG-PI) is summarized in Algorithm 2. The testing stage of our WSDG-PI method is similar to that of WSDG, as discussed in Section III-B2. Note that $\tilde{\mathbf{w}}_{c,m}$'s are not used in the testing phase, because the PI (i.e., textual features) is not available for the testing samples.

C. Time Complexity Analysis

Our WSDG-PI method consists of two steps, in which we first discover latent domains by solving the QP problem in (1) and then learn classifiers by solving the problem in (34). In the first step, according to [42], the time complexity for solving

[3]We adopt the same initialization method as in Algorithm 1.

the nonconvex QP problem in (1) is $O((NM)^3)$, in which N (resp., M) is the number of training samples (resp., latent domains).

In the second step, we solve the convex problem in (34) by employing the cutting-plane algorithm, in which we add the most violated label candidate and solve the MKL subproblem at each iteration. Since it is much more time-consuming to solve the MKL subproblems, the time complexity of the problem in (34) can be roughly estimated as $T \cdot O(\text{MKL})$, in which T is the number of iterations and $O(\text{MKL})$ is the time complexity of the MKL subproblem.

Nevertheless, no previous work has studied the time complexity of MKL theoretically. When solving the MKL problem in (33), the most time-consuming step is to solve the convex QP problem with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\varsigma}$ when fixing $\mathbf{d}$, which is solved by using our SMO solver. According to [48], the time complexity of SMO is between $O(LCM)$ and $O((LCM)^{2.3})$, in which M is the number of latent domains, and L and C are the number of bags and categories, respectively. So the time complexity of MKL (i.e., $O(\text{MKL})$) is between $t \cdot O(LCM)$ and $t \cdot O((LCM)^{2.3})$, where t is the number of iterations in MKL. Since our WSDG method also employs the cutting-plane algorithm and solves an MKL subproblem by using our SMO solver at each iteration, its time complexity can be analyzed similarly.

V. Experiments

In this section, the effectiveness of our WSDG approach is demonstrated for image classification and video event recognition by comprehensive experiments on three benchmark data sets. We also analyze why we can learn a better classifier and discover more distinctive latent domains by removing outliers in our WSDG method. Moreover, we extend our WSDG method to WSDG-PI, and the experimental results indicate the benefit of utilizing PI (i.e., additional textual features).

A. Weakly Supervised Domain Generalization

1) Experimental Settings: Our WSDG method is evaluated by utilizing the videos and images crawled from Web to train classifiers for video event recognition and image classification tasks, respectively. In this paper, we use multiclass classification accuracy for performance evaluation, as suggested in [16].

For the video event recognition task, we employ two benchmark data sets Kodak [49] and CCV [50]. The Kodak data set contains 195 consumer videos from 6 event categories. The CCV data set [50] contains 4659 and 4658 videos from 20 categories for training and testing, respectively. Strictly following the experimental setting in [10], only the videos belonging to the related event categories are used and the categories sharing similar semantic meanings are merged, which finally leads to 2440 videos from five event classes.

In order to collect the training set for video event recognition from the Internet, Web videos are crawled from *Flickr.com* by querying based on the six (resp., five) event category names for the Kodak (resp., CCV) testing set. For each query, 100 relevant Web videos are downloaded and partitioned uniformly according to their ranks to construct 20 bags with 5 instances in each bag.

For the visual features used for video event recognition, we first extract improved dense trajectory (IDT) descriptors, which include 100-D trajectory, 96-D histograms of oriented gradients (HOG), 108-D histograms of optical flow (HOF), and 192-D motion boundary histogram by using the source code provided in [51]. Then, following the Fisher vector encoding method in [51], we train 256 Gaussian mixture models by using the IDT descriptors from the videos in the Flickr training data set and generate the 128 000-D Fisher vector for each video on both training and testing data sets. Finally, following [4], we use the ASTPM method to obtain the video clip distances based on Fisher vectors. When employing the ASTPM method, we set the volume size as $1/2^l$ ($l = 1, \ldots, L$) of the original video in height, width, and temporal dimension, in which L is set as 2, as suggested in [4]. Based on the obtained distance matrices, we calculate the average of RBF kernel matrices from different pyramid levels, which are used in the training or testing procedure.

For the image classification task, the BING data set [2] is used as the source domain, while the Caltech-256 data set is used as the testing set. Strictly following the experimental setting in [16], we only utilize the images belonging to the first 30 categories in the BING and Caltech-256 data sets. Following [16], 20 training images and 25 testing images are used per category, which leads to a total of 600 (resp., 750) training (resp., testing) samples. Similar to video event recognition, we uniformly partition the training images based on the given indices to construct training bags with five instances in each bag. We employ the DeCAF features [52] (i.e., the sixth layer outputs) as the visual features, which leads to 4096-D DeCAF_6 features.

As Web data are not associated with explicit domain labels and even the number of latent domains is not given, we follow [16] to assume there are two latent domains for all methods on all data sets. We empirically fix $C_1 = C_2 = 1$ and $\eta = 0.8$ (resp., 0.2) for our WSDG approach for image classification (resp., video event recognition). For fair comparison, the optimal parameters are selected for baseline methods based on their best performances on the testing data set.

2) Baselines: Our WSDG approach is compared with three sets of baselines: the MIL baselines, the domain generalization baselines, and the latent domain discovering baselines. The MIL methods can be categorized into the instance-level methods, including mi-SVM [21] and MIL-constrained positive bags (MIL-CPB) [20] and the bag-level methods, including sparse MIL (sMIL) [53] and KI-SVM [22]. The domain generalization methods contain the low-rank exemplar SVM (LRESVM) method [14] and the domain-invariant component analysis (DICA) method [13]. Note that the approach in [12] cannot be directly applied to our tasks, since the training Web data are not associated with the domain labels. For the two latent domain discovering methods [16], [17], we employ two strategies named Match and Ensemble following the suggestion in [14].

Furthermore, as the max-margin multiple-instance dictionary learning (MMDL) method in [36] and the

TABLE I

ACCURACIES (%) OF BASELINES AND OUR WSDG METHOD, INCLUDING TWO SPECIAL CASES FOR THE IMAGE CLASSIFICATION AND VIDEO EVENT RECOGNITION TASKS. WE DENOTE THE BEST RESULTS IN BOLDFACE

Method	Testing Dataset		
	Kodak	CCV	Caltech-256
SVM [54]	40.00	45.80	70.93
sMIL [53]	46.15	50.52	71.33
mi-SVM [21]	43.59	51.31	71.47
MIL-CPB [20]	46.67	51.76	71.60
KI-SVM [22]	46.15	46.36	71.20
DICA [13]	45.12	50.80	70.80
LRESVM [14]	49.74	54.69	72.93
[16] (Match)	41.03	50.18	71.07
[16] (Ensemble)	42.05	49.96	70.08
[17] (Match)	45.13	50.78	71.47
[17](Ensemble)	46.15	52.20	72.40
Sub-Cate [35]	45.13	53.17	72.27
MMDL [36]	47.69	54.70	72.80
WSDG_sim1	48.21	52.02	71.87
WSDG_sim2	50.26	55.37	74.00
WSDG	**51.28**	**56.83**	**75.20**

discriminative subcategorization method [35] are related to our approach, both methods are also used as the baselines.

To demonstrate the benefits of discovering latent domains and validate our MMD-based regularizer in (9), the performances of two simplified versions of our WSDG approach are additionally reported. We refer to them as WSDG_sim1 and WSDG_sim2, respectively. In particular, in WSDG_sim2, we set $C_2 = 0$ to remove the MMD-based regularizer $\rho(\mathbf{B}, \mathbf{K} \circ (\mathbf{h}\mathbf{h}'))$ in our WSDG approach. Based on WSDG_sim2, the latent domain issues are further ignored by setting the number of latent domains to one (i.e., $M = 1$) and we refer to this case as WSDG_sim1, in which our objective in (9) can be reduced to that in (6).

3) Experimental Results: The experimental results are reported in Table I, from which we can see that the subcategorization baselines MMDL and Sub-Cate, the domain generalization baselines LRESVM and DICA, and the latent domain discovering baselines [16], [17] generally outperform SVM. These results show that exploiting additional information, such as subcategories, low-rank structure, or hidden latent domains in the training samples, is helpful.

Another observation is that the MIL baselines (e.g., mi-SVM, MIL-CPB, sMIL, and KI-SVM) outperform SVM on all three data sets, although various MIL assumptions are used in these methods. We also observe that the MMDL method outperforms both the Sub-Cate method and MIL baselines, possibly because it simultaneously exploits subcategories and utilizes the MIL technique to cope with the label noise in Web data.

The performances of MIL baselines are worse than that of our special case WSDG_sim1, which might because the classifiers for different categories are jointly learned. WSDG_sim1 is worse than WSDG_sim2 on all three data sets, which demonstrates the advantage of integrating multiple classifiers from different latent domains. Moreover, our WSDG approach achieves better performances than WSDG_sim2 on all three data sets, which proves that our MMD-based regularizer in (9) is effective. Another observation is that WSDG and WSDG_sim2 are better than all the MIL baselines [20]–[22], [53] and the domain generalization baselines LRESVM and DICA, which shows the advantage of handling label noise and exploiting latent domains in the Web images/videos at the same time.

Finally, the best results are achieved by our WSDG method on all data sets and the results clearly show that our WSDG method is effective for the image classification and video event recognition tasks by utilizing the Web data.

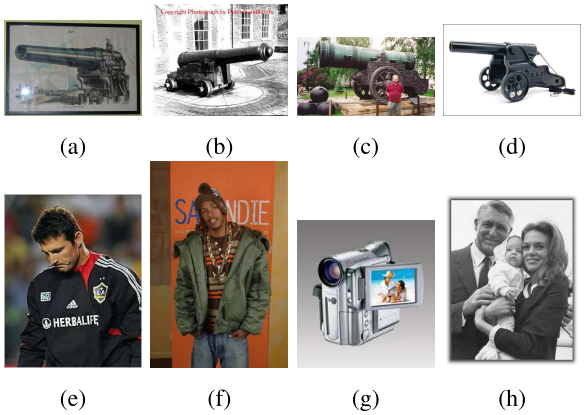


Fig. 2. Top and bottom rows show the most and least confident images for the category cannon on the Bing data set, respectively. (a) $\tilde{h} = 1$. (b) $\tilde{h} = 1$. (c) $\tilde{h} = 1$. (d) $\tilde{h} = 1$. (e) $\tilde{h} = 0$. (f) $\tilde{h} = 0.24$. (g) $\tilde{h} = 0.24$. (h) $\tilde{h} = 0.24$.

B. Experimental Analysis on WSDG

Recall that in our WSDG method, we tend to identify a subset of outliers from the training samples and simultaneously expect the selected samples coming from more distinctive latent domains by using the indicator $\mathbf{h}$ in (9). Let us take the image classification task (i.e., the training and testing sets are the Bing and Caltech-256 data sets, respectively) as an example to show the benefits by introducing $\mathbf{h}$ for removing outliers and discovering more distinctive latent domains.

We first demonstrate the effectiveness of our WSDG method for removing the outliers. Note that the problem for solving a binary indicator $\mathbf{h}$ is relaxed to seeking for a linear combination of feasible $\mathbf{h}_t$'s, so we calculate $\tilde{h} = \sum_{t=1}^{T} d_t \mathbf{h}_t$ as the approximation of $\mathbf{h}$, where d_t and $\mathbf{h}_t$ are learned by solving (15). Intuitively, for each element $\tilde{h}_i$ in the vector $\tilde{h}$, the higher value $\tilde{h}_i$ indicates that it is more confident that the corresponding training image is a true positive instance. We show the most and least confident images from the category cannon in the Bing data set and their corresponding $\tilde{h}_i$'s in Fig. 2. We can observe that the images with the highest $\tilde{h}_i$'s are all true positive instances (see the top row), while the images with the lowest $\tilde{h}_i$'s are the outliers (see the bottom row). This indicates that our WSDG method is able to remove the outliers from the training samples, and thus, we can learn more robust classifiers for the domain generalization problem.

In order to demonstrate the effectiveness of our WSDG approach for constructing more distinctive latent domains, we calculate the SMMDs between each pair of latent domains to measure the distinctiveness of latent domains.

TABLE II

SMMDs BETWEEN EACH PAIR OF LATENT DOMAINS BY USING DIFFERENT METHODS

Method	[16]	[17]	WSDG
SMMDs	24.46	27.08	31.56

We also compare our WSDG method with the latent domain discovering methods in [16] and [17]. For [16], we denote the binary latent domain indictor as $\bar{\pi}_{i,m}$'s, where $\bar{\pi}_{i,m}$ indicates whether the i-th training sample comes from the m-th hidden latent domain, and then, we calculate the SMMDs between each pair of latent domains as $\sum_{m\neq\tilde{m}} \|(1/N_m)\sum_{i=1}^{N}\bar{\pi}_{i,m}\phi(\mathbf{x}_i)-(1/N_{\tilde{m}})\sum_{i=1}^{N}\bar{\pi}_{i,\tilde{m}}\phi(\mathbf{x}_i)\|^2$. For [17], we calculate the SMMDs between each pair of latent domains based on the soft assignment coefficients $\boldsymbol{\beta}_m$'s as $\sum_{m\neq\tilde{m}}(\boldsymbol{\beta}_m-\boldsymbol{\beta}_{\tilde{m}})'\mathbf{K}(\boldsymbol{\beta}_m-\boldsymbol{\beta}_{\tilde{m}})$ [see (2)]. For our method, we first calculate $\bar{\boldsymbol{\beta}}_m=(\tilde{\mathbf{h}}\circ\boldsymbol{\beta}_m/\|\tilde{h}\circ\boldsymbol{\beta}_m\|_1)$, and then obtain the SMMDs between each pair of the latent domains by using $\sum_{m\neq\tilde{m}}(\bar{\boldsymbol{\beta}}_m-\bar{\boldsymbol{\beta}}_{\tilde{m}})'\mathbf{K}(\bar{\boldsymbol{\beta}}_m-\bar{\boldsymbol{\beta}}_{\tilde{m}})$.

In Table II, the image classification task is taken as an example to report the SMMDs between each pair of latent domains from different methods. It can be seen from Table II that the SMMDs of the work in [17] are larger than that of [16], possibly because the work in [17] is specifically designed to maximize the SMMDs between each pair of latent domains. We also observe that SMMDs of our WSDG approach is larger than that of [17], which demonstrates that our WSDG method can construct more distinctive latent domains by removing the outliers. So our WSDG method has better generalization ability than [16] and [17].

C. Weakly Supervised Domain Generalization Using Privileged Information

1) Experimental Settings: Our proposed WSDG-PI approach is evaluated using the Flickr Web video data set (resp., the CCV and Kodak data sets) as the training set (resp., the testing sets). Note that the Bing data set provided in [2] is not associated with textual information, so our WSDG-PI method cannot be evaluated on the Caltech-256 data set. We crawl the surrounding tags of each Flickr video and extract a 2000-D term frequency (TF) feature based on the associated tags for each video. When extracting the TF features, the vocabulary is constructed by using 2000 most frequent words after removing the stop words. These textual features of the training data are considered as PI. All other settings are identical to those in Section V-A. WSDG-PI has two more parameters C_3 and λ, compared with WSDG. We empirically fix C_3 as 0.1 and λ as 10 on both data sets. For the baselines, the optimal parameters are selected based on their best performances on the testing data set.

2) Baselines: Our method is compared with RankTransfer (RT) [39] and SVM+ [18]. Moreover, we additionally include Classeme [55] as well as two multi-view learning methods SVM-2K [56] and kernel canonical correlation analysis (KCCA) [57] as the baselines, because they can also utilize both textual features and visual features of training samples.

TABLE III

ACCURACIES (%) OF THE BASELINES AND OUR METHODS FOR THE VIDEO EVENT RECOGNITION TASK. WE DENOTE THE BEST RESULTS IN BOLDFACE

Method	Testing Dataset	
	Kodak	CCV
SVM [54]	40.00	45.80
SVM-2K [56]	46.15	51.33
KCCA [57]	45.64	51.05
Classeme [55]	44.62	47.67
RT [39]	43.59	49.22
SVM+ [18]	47.69	52.69
sMIL-PI [9]	49.23	54.88
WSDG	51.28	56.83
WSDG-PI	**55.38**	**58.15**

1) *Classeme [55]:* For each word in the 2000-D textual features, we learn a classeme classifier based on the relevant and irrelevant samples. For each sample from both training set and testing set, the visual features are augmented with the 2000 decision values, which are obtained by using 2000 prelearned classeme classifiers. Finally, we use the the augmented features to train the SVM classifiers and predict the testing samples.
2) *SVM-2K [56]:* SVM-2K classifiers are trained by utilizing both visual features and textual features of training data. Then, the classifier based on visual features is used to classify the testing samples.
3) *KCCA [57]:* KCCA is employed on the visual features and textual features of the training data. Then, we use the projected visual features to train SVM classifiers and classify the testing samples.

We also compare our WSDG-PI method with sMIL-PI [9], which can simultaneously cope with label noise and take the advantage of PI (i.e., textual features). We additionally include SVM and WSDG for comparison.

3) Experimental Results: The experimental results are reported in Table III, from which we observe that LUPI methods SVM+ and RT outperform SVM on both data sets, which indicates the advantage of utilizing PI (i.e., additional textual features). Besides, multiview approaches SVM-2K and KCCA also outperform SVM on both data sets after employing both visual features and textual features. We also observe that Classeme outperforms SVM on both data sets. One possible explanation is that it is helpful to augment the visual features with the decision values obtained by using classeme classifiers. Moreover, on both data sets, sMIL and our WSDG-PI method are better than sMIL reported in Table I and WSDG, respectively, gain demonstrates the benefits of utilizing the textual features as PI.

Finally, our method WSDG-PI outperforms all the baselines on both data sets, which indicates the benefits of simultaneously handling label noise, exploiting PI, and learning robust classifiers for better generalization ability.

D. Robustness of Our Approaches With Respect to Parameters

We take the CCV data set as an example to study the performance variation of our WSDG and WSDG-PI methods

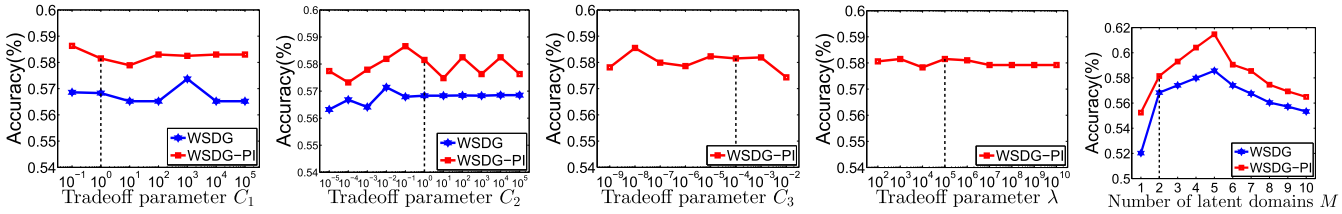


Fig. 3. Accuracies of our WSDG and WSDG-PI methods on the CCV data set when using different tradeoff parameters. Vertical dashed lines: default parameters.

TABLE IV
TRAINING TIME (S) OF THE BASELINES WITHOUT USING PI AND OUR WSDG APPROACH ON THE BING AND CCV DATA SET

Method	KI-SVM	[16]	[17]	DICA	LRESVM	Sub-Cate	MMDL	WSDG
Bing	94.89	213.64	19.54	126.61	2986.57	436.18	195.01	102.15
CCV	20.50	189.91	17.49	83.85	2484.31	77.62	46.43	39.54

TABLE V
TRAINING TIME (S) OF THE BASELINES USING PI AND OUR WSDG-PI APPROACH ON THE CCV DATA SET

Method	SVM-2K	KCCA	Classeme	RT	SVM+	sMIL-PI	WSDG-PI
CCV	31.67	41.32	1526.12	89.90	37.88	29.16	72.40

by varying one parameter when fixing all other parameters as their default values. Note that C_1, C_2, and M (i.e., the number of latent domains) are the common parameters shared by our WSDG and WSDG-PI methods, while C_3 and γ are the additional parameters of WSDG-PI method. From Fig. 3, we observe that our methods are relatively robust when the tradeoff parameters C_1, C_2, C_3, and γ are varied in certain ranges. We also observe that the results of our methods are improved when M increases but less than 5. If M increases over 5, the results of our methods decrease. One possible explanation is that the training set is considerably diverse, so it contains more than two latent domains. On the other hand, the total number of training samples is limited (only 500 or 600 training images/videos on the Bing/Flickr data set), so that the results of our methods will decrease if we use too many latent domains.

E. Comparison of Training Time

We compare the training time of our WSDG and WSDG-PI methods with other baseline methods. All the experiments are conducted on a server machine with 18-GB RAM and Intel Xeon 3.33-GHz CPUs using a single thread. Let us take the Bing and CCV data sets as two examples. In Table IV, we report the training time of our WSDG method and other baselines without using PI. We observe that our WSDG method is more efficient than other baselines except [17] and KI-SVM. WSDG is slower than [17], because we need to solve (9) instead of directly using SVM after employing the latent domain discovery technique in [17]. KI-SVM is also faster than WSDG. One possible explanation is that we need to solve a more complex subproblem in each iteration.

In Table V, we report the training time of our WSDG-PI method and other baseline methods using PI. Note that the images in the Bing data set do not have additional textual information, so we only report the training time on the CCV data set in Table V. The training time of WSDG-PI is longer than that of WSDG reported in Table IV, because we need to solve a larger scale QP problem at each iteration in our WSDG-PI method. Our WSDG-PI method is still reasonably efficient when compared with the other baseline methods.

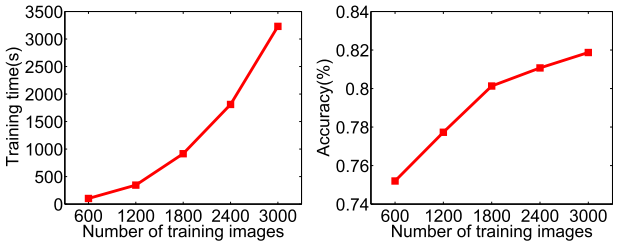


Fig. 4. Training time and accuracies of our WSDG method with respect to the number of training images on the Bing data set.

F. Time Complexity and Scalability of Our Approach

Let us take the image classification task with Bing as the training set and Caltech-256 as the testing set as an example to demonstrate the scalability of our WSDG method. As the Bing data set and its associated training indices with respect to various numbers of training samples per category are provided in [2], we use various numbers of training samples for each category (i.e., [20, 40, 60, 80, 100]) to construct the training set in order to evaluate the performance and the scalability of our algorithms. Since we use 30 categories on the Bing data set and n training samples per category, we have a total of $30n$ training samples. The accuracies and the training time with various numbers of training samples are shown in Fig. 4, from which we observe that both the accuracy and the training time increase as the number of training samples increases.

VI. CONCLUSION

In this paper, a novel WSDG approach has been proposed for visual recognition tasks by utilizing loosely labeled Web images/videos as training data. Our WSDG method is able

to handle the label noise in training Web data and has good generalization ability to the arbitrary target domain. In addition, we have extended our WSDG approach to WSDG-PI by utilizing the textual descriptions of the training Web data as PI. The effectiveness of our WSDG and WSDG-PI methods has also been demonstrated by the comprehensive experiments.

APPENDIX
DERIVATIONS OF (13) AND (32)

The derivations of (13) and (32) are very similar. In fact, the derivation of (32) can be used for deriving (13) by removing the terms related to PI. In the following, we first provide the derivation of (32), and then discuss how to derive (13).

To derive the dual form in (27), we first reformulate it into a simpler form. In particular, an intermediate variable $\theta_{i,c,m,\tilde{m}}$ is introduced as follows:

$$\theta_{i,c,m,\tilde{m}} = \begin{cases} \hat{\beta}_{i,m} & c = y_i \\ \delta(m = \tilde{m}) & c \neq y_i. \end{cases} \tag{40}$$

Then, we have $\sum_{m=1}^{M} \hat{\beta}_{i,m}(\mathbf{w}_{y_i,m})'\phi(\mathbf{x}_i) = \sum_{m=1}^{M} \theta_{i,y_i,m,\tilde{m}}(\mathbf{w}_{y_i,m})'\phi(\mathbf{x}_i)$ and $(\mathbf{w}_{c,\tilde{m}})'\phi(\mathbf{x}_i) = \sum_{m=1}^{M} \theta_{i,c,m,\tilde{m}}(\mathbf{w}_{c,m})'\phi(\mathbf{x}_i)$. Similarly, we can represent $\sum_{m=1}^{M} \hat{\beta}_{i,m}(\tilde{\mathbf{w}}_{y_i,m})'\tilde{\phi}(\mathbf{z}_i)$ as $(\tilde{\mathbf{w}}_{c,\tilde{m}})'\tilde{\phi}(\mathbf{z}_i)$ by using θ.

Let us define a function $G(\mathcal{B}_l, \tilde{c}, \tilde{m}) = (1/|\mathcal{B}_l|)\sum_{i\in\mathcal{I}_l} h_i (\sum_{m=1}^{M} \theta_{i,Y_l,m,\tilde{m}}(\mathbf{w}_{Y_l,m})'\phi(\mathbf{x}_i) - \sum_{m=1}^{M} \theta_{i,\tilde{c},m,\tilde{m}}(\mathbf{w}_{\tilde{c},m})'\phi(\mathbf{x}_i))$. By similarly defining $\tilde{G}(\mathcal{B}_l, \tilde{c}, \tilde{m})$ using $\theta_{i,c,m,\tilde{m}}$'s, the constraints in (28) and (30) can be uniformly written as follows:

$$G(\mathcal{B}_l, \tilde{c}, \tilde{m}) \geq \zeta_{l,\tilde{c},\tilde{m}} - \tilde{G}(\mathcal{B}_l, \tilde{c}, \tilde{m}) - \xi_l, \quad \forall l, \tilde{c}, \tilde{m} \tag{41}$$

in which $\zeta_{l,\tilde{c},\tilde{m}} = 0$ if $\tilde{c} = Y_l$, and $\zeta_{l,\tilde{c},\tilde{m}} = \eta$ otherwise.

Similarly, the constraints in (29) and (31) can be uniformly written as $\tilde{G}(\mathcal{B}_l, \tilde{c}, \tilde{m}) \geq \zeta_{l,\tilde{c},\tilde{m}} - \epsilon_l, \quad \forall l, \tilde{c}, \tilde{m}$.

All $\mathbf{w}_{c,m}$'s are concatenated and we define $\mathbf{w} = [\mathbf{w}'_{1,1}, \ldots, \mathbf{w}'_{1,M}, \mathbf{w}'_{2,1}, \ldots, \mathbf{w}'_{C,M}]'$. Furthermore, a new mapping function is defined for each $\mathcal{B}_l$ as $\varphi(\mathbf{h}, \mathcal{B}_l, c, \tilde{m}) = [(1/|B_l|) \sum_{i\in I_l} h_i \ \theta_{i,1,1,\tilde{m}}\delta(c = 1)\phi(\mathbf{x}_i)', \ldots, (1/|B_l|) \sum_{i\in I_l} h_i \ \theta_{i,C,M,\tilde{m}}\delta(c = C)\phi(\mathbf{x}_i)']'$. Similarly, we concatenate all $\tilde{\mathbf{w}}_{c,m}$'s as $\tilde{\mathbf{w}}$ and define $\tilde{\varphi}(\mathbf{h}, \mathcal{B}_l, c, \tilde{m})$ by replacing $\phi(\mathbf{x}_i)$'s with $\tilde{\phi}(\mathbf{z}_i)$'s. By further denoting $\psi(\mathbf{h}, \mathcal{B}_l, c, \tilde{m}) = \varphi(\mathbf{h}, \mathcal{B}_l, Y_l, \tilde{m}) - \varphi(\mathbf{h}, \mathcal{B}_l, c, \tilde{m})$ and $\tilde{\psi}(\mathbf{h}, \mathcal{B}_l, c, \tilde{m}) = \tilde{\varphi}(\mathbf{h}, \mathcal{B}_l, Y_l, \tilde{m}) - \tilde{\varphi}(\mathbf{h}, \mathcal{B}_l, c, \tilde{m})$, we observe $G(\mathcal{B}_l, \tilde{c}, \tilde{m})$ and $\tilde{G}(\mathcal{B}_l, \tilde{c}, \tilde{m})$ can be represented as $\mathbf{w}'\psi(\mathbf{h}, \mathcal{B}_l, c, m)$ and $\tilde{\mathbf{w}}'\tilde{\psi}(\mathbf{h}, \mathcal{B}_l, c, m)$, respectively, so we can simply the objective function in (27) as follows:

$$\min_{\substack{\mathbf{h}\in\mathcal{H},\mathbf{w},\tilde{\mathbf{w}} \\ \xi_l,\epsilon_l}} \frac{1}{2}(\|\mathbf{w}\|^2 + \lambda\|\tilde{\mathbf{w}}\|^2) + C_1 \sum_{l=1}^{L}(\xi_l + \epsilon_l)$$
$$-C_2\,\rho(\mathbf{B}, \mathbf{K} \circ (\mathbf{h}\mathbf{h}')) + C_3\tilde{\mathbf{w}}'\tilde{\psi}(\mathbf{h}, \mathcal{B}_l, c, m) \tag{42}$$
$$\text{s.t. } \mathbf{w}'\psi(\mathbf{h}, \mathcal{B}_l, c, m) \geq \zeta_{l,c,m} - \tilde{\mathbf{w}}'\tilde{\psi}(\mathbf{h}, \mathcal{B}_l, c, m) - \xi_l, \quad \forall l, c, m \tag{43}$$
$$\tilde{\mathbf{w}}'\tilde{\psi}(\mathbf{h}, \mathcal{B}_l, c, m) \geq \zeta_{l,c,m} - \epsilon_l, \quad \forall l, c, m. \tag{44}$$

We introduce a dual variable $\alpha_{l,c,m}$ and $\varsigma_{l,c,m}$ for each constraint in (43) and (44), respectively. When the derivatives of the Lagrangian form of (42) with respect to ξ_l's and ϵ_l's are set to zeros, respectively, we can obtain $\sum_{c,m} \alpha_{l,c,m} = C_1, \forall l$ and $\sum_{c,m} \varsigma_{l,c,m} = C_1, \forall l$. By, respectively, setting the derivative of the Lagrangian of (42) with respect to $\mathbf{w}$ and $\tilde{\mathbf{w}}$ as zero, we can obtain the following equations:

$$\mathbf{w} = \sum_{l,c,m} \alpha_{l,c,m}\,\psi(\mathbf{h}, \mathcal{B}_l, c, m), \tag{45}$$

$$\tilde{\mathbf{w}} = \frac{1}{\lambda}\sum_{l,c,m} (\alpha_{l,c,m} + \varsigma_{l,c,m} - C_3)\,\tilde{\psi}(\mathbf{h}, \mathcal{B}_l, c, m). \tag{46}$$

By substituting (45) and (46) back into the Lagrangian of (42), we can arrive at the dual form of (42) as follows:

$$\min_{\mathbf{h}\in\mathcal{H}} \max_{\boldsymbol{\alpha},\boldsymbol{\varsigma}} -\frac{1}{2}\boldsymbol{\alpha}'\mathbf{Q}^{\mathbf{h}}\boldsymbol{\alpha} - \frac{1}{2\lambda}(\boldsymbol{\alpha} + \boldsymbol{\varsigma} - C_3\mathbf{1})'\tilde{\mathbf{Q}}^{\mathbf{h}}(\boldsymbol{\alpha} + \boldsymbol{\varsigma} - C_3\mathbf{1}) + \boldsymbol{\zeta}'(\boldsymbol{\alpha} + \boldsymbol{\varsigma}) - C_2\rho(\mathbf{B}, \mathbf{K} \circ (\mathbf{h}\mathbf{h}'))$$
$$\text{s.t. } \sum_{c,m} \alpha_{l,c,m} = C_1, \quad \forall l,$$
$$\alpha_{l,c,m} \geq 0, \quad \forall l, c, m$$
$$\sum_{c,m} \varsigma_{l,c,m} = C_1, \quad \forall l,$$
$$\varsigma_{l,c,m} \geq 0, \quad \forall l, c, m \tag{47}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^{\tilde{D}}$ (resp., $\boldsymbol{\varsigma} \in \mathbb{R}^{\tilde{D}}$) is a vector containing the dual variables $\alpha_{l,c,m}$ (resp., $\varsigma_{l,c,m}$), and $\tilde{D} = L \cdot C \cdot M$, $\boldsymbol{\zeta} \in \mathbb{R}^{\tilde{D}}$ is a vector, in which each entry $\zeta_{l,c,m} = 0$ if $c = Y_l$ and $\zeta_{l,c,m} = \eta$ otherwise. $\mathbf{Q}^{\mathbf{h}} \in \mathbb{R}^{\tilde{D}\times\tilde{D}}$ is a matrix with each entry being $Q^{\mathbf{h}}_{u,v} = \psi(\mathbf{h}, \mathcal{B}_l, c, m)'\psi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m})$, in which u and v are the indices defined in the following paragraph (13), and $\tilde{\mathbf{Q}}^{\mathbf{h}}$ is similarly defined as $\mathbf{Q}^{\mathbf{h}}$ by replacing $\psi(\mathbf{h}, \mathcal{B}_l, c, m)$ with $\tilde{\psi}(\mathbf{h}, \mathcal{B}_l, c, m)$.

In the following, we derive the detailed form of $Q^{\mathbf{h}}_{u,v}$ and $\tilde{Q}^{\mathbf{h}}_{u,v}$. Recall that $\psi(\mathbf{h}, \mathcal{B}_l, c, \tilde{m}) = \varphi(\mathbf{h}, \mathcal{B}_l, Y_l, \tilde{m}) - \varphi(\mathbf{h}, \mathcal{B}_l, c, \tilde{m})$, then we can obtain that

$$\begin{aligned}
&\psi(\mathbf{h}, \mathcal{B}_l, c, m)'\psi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m}) \\
&= (\varphi(\mathbf{h}, \mathcal{B}_l, Y_l, m) - \varphi(\mathbf{h}, \mathcal{B}_l, c, m))' \\
&\quad \times \big(\varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, Y_{\tilde{l}}, \tilde{m}) - \varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m})\big) \\
&= -\varphi(\mathbf{h}, \mathcal{B}_l, Y_l, m)'\varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m}) + \varphi(\mathbf{h}, \mathcal{B}_l, Y_l, m)' \\
&\quad \times \varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, Y_{\tilde{l}}, \tilde{m}) \\
&\quad + \varphi(\mathbf{h}, \mathcal{B}_l, c, m)'\varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m}) - \varphi(\mathbf{h}, \mathcal{B}_l, c, m)' \\
&\quad \times \varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, Y_{\tilde{l}}, \tilde{m}).
\end{aligned} \tag{48}$$

Let us define $S_1 = \varphi(\mathbf{h}, \mathcal{B}_l, Y_l, m)' \ \varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m})$, $S_2 = \varphi(\mathbf{h}, \mathcal{B}_l, Y_l, m)' \ \varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, Y_{\tilde{l}}, \tilde{m})$, $S_3 = \varphi(\mathbf{h}, \mathcal{B}_l, c, m)' \ \varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m})$, and $S_4 = \varphi(\mathbf{h}, \mathcal{B}_l, c, m)' \ \varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, Y_{\tilde{l}}, \tilde{m})$, then $\psi(\mathbf{h}, \mathcal{B}_l, c, m)'\psi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m}) = -S_1 + S_2 + S_3 - S_4$. We derive the detailed form of S_1, S_2, S_3, and S_4 as follows. Recall that $\varphi(\mathbf{h}, \mathcal{B}_l, c, \tilde{m}) = [(1/|B_l|) \sum_{i\in I_l} h_i \ \theta_{i,1,1,\tilde{m}}\delta(c = 1) \ \phi(\mathbf{x}_i)', \ldots, (1/|B_l|) \sum_{i\in I_l} h_i \ \theta_{i,C,M,\tilde{m}}\delta(c = C)\phi(\mathbf{x}_i)']'$.

The first term S_1 can be derived as

$$\begin{aligned} S_1 &= \varphi(\mathbf{h}, \mathcal{B}_l, Y_l, m)'\varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m}) \\ &= \frac{1}{|\mathcal{B}_l||\mathcal{B}_{\tilde{l}}|}\sum_{i\in\mathbf{I}_l}\sum_{j\in\mathbf{I}_{\tilde{l}}} h_i h_j \Bigg[\delta(c=Y_l)\delta(c=Y_{\tilde{l}})\sum_{q=1}^{M}\hat{\beta}_{i,q}\hat{\beta}_{j,q} \\ &\qquad \times \phi(\mathbf{x}_i)'\phi(\mathbf{x}_j) + (1-\delta(c=Y_l)) \\ &\qquad \times \delta(c=Y_{\tilde{l}})\hat{\beta}_{j,m}\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)\Bigg]. \end{aligned} \tag{49}$$

The second term S_2 is derived as

$$\begin{aligned} S_2 &= \varphi(\mathbf{h}, \mathcal{B}_l, Y_l, m)'\varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, Y_{\tilde{l}}, \tilde{m}) \\ &= \frac{\delta(Y_l=Y_{\tilde{l}})}{|\mathcal{B}_l||\mathcal{B}_{\tilde{l}}|}\sum_{i\in\mathbf{I}_l}\sum_{j\in\mathbf{I}_{\tilde{l}}} h_i h_j \sum_{q=1}^{M}\hat{\beta}_{i,q}\hat{\beta}_{j,q}\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j). \end{aligned} \tag{50}$$

Similarly, we can derive the third term S_3 as follows:

$$\begin{aligned} &S_3 \\ &= \varphi(\mathbf{h}, \mathcal{B}_l, c, m)'\varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m}) \\ &= \frac{\delta(c=\tilde{c})}{|\mathcal{B}_l||\mathcal{B}_{\tilde{l}}|}\sum_{i\in\mathbf{I}_l}\sum_{j\in\mathbf{I}_{\tilde{l}}} h_i h_j \Bigg[\delta(c=Y_l)\delta(\tilde{c}=Y_{\tilde{l}})\sum_{q=1}^{M}\hat{\beta}_{i,q}\hat{\beta}_{j,q} \\ &\quad \times \phi(\mathbf{x}_i)'\phi(\mathbf{x}_j) + (1-\delta(c=Y_l))\delta(\tilde{c}=Y_{\tilde{l}})\hat{\beta}_{j,m}\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j) \\ &\quad + \delta(c=Y_l)(1-\delta(\tilde{c}=Y_{\tilde{l}}))\hat{\beta}_{i,\tilde{m}}\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j) \\ &\quad + (1-\delta(c=Y_l))(1-\delta(\tilde{c}=Y_{\tilde{l}}))\delta(m=\tilde{m})\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)\Bigg]. \end{aligned} \tag{51}$$

Finally, the last term S_4 is derived as

$$\begin{aligned} &S_4 \\ &= \varphi(\mathbf{h}, \mathcal{B}_l, c, m)'\varphi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, Y_{\tilde{l}}, \tilde{m}) \\ &= \frac{1}{|\mathcal{B}_l||\mathcal{B}_{\tilde{l}}|}\sum_{i\in\mathbf{I}_l}\sum_{j\in\mathbf{I}_{\tilde{l}}} h_i h_j \Bigg[\delta(\tilde{c}=Y_{\tilde{l}})\delta(\tilde{c}=Y_l)\sum_{q=1}^{M}\hat{\beta}_{j,q}\hat{\beta}_{i,q} \\ &\qquad \times \phi(\mathbf{x}_i)'\phi(\mathbf{x}_j) + (1-\delta(\tilde{c}=Y_{\tilde{l}})) \\ &\qquad \delta(\tilde{c}=Y_l)\hat{\beta}_{i,\tilde{m}}\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)\Bigg]. \end{aligned} \tag{52}$$

By substituting [49]–[52] into (48), and combining similar terms, we arrive at

$$\begin{aligned} &\psi(\mathbf{h}, \mathcal{B}_l, c, m)'\psi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m}) \\ &= -S_1 + S_2 + S_3 - S_4 \\ &= \frac{1}{|\mathcal{B}_l||\mathcal{B}_{\tilde{l}}|}\sum_{i\in\mathbf{I}_l}\sum_{j\in\mathbf{I}_{\tilde{l}}} h_i h_j \\ &\quad \times \Bigg[\delta(Y_l=Y_{\tilde{l}})(1-\delta(c=Y_l)) \\ &\qquad \times (1-\delta(\tilde{c}=Y_{\tilde{l}}))\sum_{q=1}^{M}\hat{\beta}_{j,q}\hat{\beta}_{i,q}\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j) \\ &\qquad - (1-\delta(c=Y_l))(1-\delta(c=\tilde{c}))\delta(c=Y_{\tilde{l}}) \\ &\qquad \times \hat{\beta}_{j,m}\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j) - (1-\delta(\tilde{c}=Y_{\tilde{l}}))(1-\delta(c=\tilde{c})) \\ &\qquad \times \delta(\tilde{c}=Y_l)\hat{\beta}_{i,\tilde{m}}\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j) + \delta(m=\tilde{m})\delta(c=\tilde{c}) \\ &\qquad \times (1-\delta(\tilde{c}=Y_{\tilde{l}}))(1-\delta(c=Y_l))\phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)\Bigg] \\ &= \frac{1}{|\mathcal{B}_l||\mathcal{B}_{\tilde{l}}|}\sum_{i\in\mathbf{I}_l}\sum_{j\in\mathbf{I}_{\tilde{l}}} h_i h_j \phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)\gamma(i,j,c,\tilde{c},m,\tilde{m}), \end{aligned}$$

where $\gamma(i, j, c, \tilde{c}, m, \tilde{m})$ is defined in the paragraph after (13).

Recall that $Q^{\mathbf{h}}_{u,v} = \psi(\mathbf{h}, \mathcal{B}_l, c, m)'\psi(\mathbf{h}, \mathcal{B}_{\tilde{l}}, \tilde{c}, \tilde{m})$, where u and v are the indices defined in the paragraph after (13), so $Q^{\mathbf{h}}_{u,v} = (1/|\mathcal{B}_l||\mathcal{B}_{\tilde{l}}|)\sum_{i\in\mathbf{I}_l}\sum_{j\in\mathbf{I}_{\tilde{l}}} h_i h_j \phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)\gamma(i, j, c, \tilde{c}, m, \tilde{m})$. Note that the detailed form of each entry in $\tilde{\mathbf{Q}}^{\mathbf{h}}$, i.e., $\tilde{Q}^{\mathbf{h}}_{u,v}$, can be similarly derived by replacing $\phi(\mathbf{x}_i)$ with $\tilde{\phi}(\mathbf{z}_i)$. Given the detailed form of each entry in $\mathbf{Q}^{\mathbf{h}}$ and $\tilde{\mathbf{Q}}^{\mathbf{h}}$, the optimization problem in (47) is equivalent to (32), so we complete the derivation of (32) here.

To derive (13), by concatenating all $\mathbf{w}_{c,m}$'s as $\mathbf{w}$ and using the same definition of $\varphi(\mathbf{h}, \mathcal{B}_l, c, \tilde{m})$, we can simplify the problem in (9) as follows:

$$\begin{aligned} \min_{\substack{\mathbf{h}\in\mathcal{H}\\ \mathbf{w},\xi_l}} \ & \frac{1}{2}\|\mathbf{w}\|^2 + C_1\sum_{l=1}^{L}\xi_l - C_2\,\rho(\mathbf{B}, \mathbf{K}\circ(\mathbf{h}\mathbf{h}')) \\ \text{s.t.} \ & \mathbf{w}'\psi(\mathbf{h}, \mathcal{B}_l, c, m) \geq \zeta_{l,c,m} - \xi_l, \quad \forall l, c, m. \end{aligned} \tag{53}$$

After introducing the dual variable $\alpha_{l,c,m}$'s for the constraints in (53), we can similarly obtain the dual form of (53) as (13).

ACKNOWLEDGEMENT

This research was supported by funding from the Faculty of Engineering and Information Technologies, The University of Sydney, under the Faculty Research Cluster Program.

REFERENCES

[1] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *Proc. 24th IEEE Conf. Comput. Vis. Pattern Recognit.*, Colorado Springs, CO, USA, Jun. 2011, pp. 1521–1528.

[2] A. Bergamo and L. Torresani, "Exploiting weakly-labeled Web images to improve object classification: A domain adaptation approach," in *Proc. 24th Annu. Conf. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2010, pp. 181–189.

[3] L. Cheng and S. J. Pan, "Semi-supervised domain adaptation on manifolds," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 12, pp. 2240–2249, Dec. 2014.

[4] L. Duan, D. Xu, I. W.-H. Tsang, and J. Luo, "Visual event recognition in videos by learning from Web data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1667–1680, Sep. 2012.

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

[6] R. Gopalan, R. Li, and R. Chellappa, "Domain adaptation for object recognition: An unsupervised approach," in *Proc. 13th Int. Conf. Comput. Vis.*, Barcelona, Spain, Nov. 2011, pp. 999–1006.

[7] L. Duan, I. W. Tsang, and D. Xu, "Domain transfer multiple kernel learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 465–479, Mar. 2012.

[8] L. Duan, D. Xu, and I. W. Tsang, "Domain adaptation from multiple sources: A domain-dependent regularization approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 3, pp. 504–518, Mar. 2012.

[9] W. Li, L. Niu, and D. Xu, "Exploiting privileged information from Web data for image categorization," in *Proc. 13th Eur. Conf. Comput. Vis.*, Zürich, Switzerland, Sep. 2014, pp. 437–452.

[10] L. Duan, D. Xu, and S.-F. Chang, "Exploiting Web images for event recognition in consumer videos: A multiple source domain adaptation approach," in *Proc. 25th IEEE Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, Jun. 2012, pp. 1338–1345.

[11] L. Chen, L. Duan, and D. Xu, "Event recognition in videos by learning from heterogeneous Web sources," in *Proc. 26th IEEE Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, Jun. 2013, pp. 2666–2673.

[12] A. Khosla, T. Zhou, T. Malisiewicz, A. A. Efros, and A. Torralba, "Undoing the damage of dataset bias," in *Proc. 12th Eur. Conf. Comput. Vis.*, Florence, Italy, Oct. 2012, pp. 158–171.

[13] K. Muandet, D. Balduzzi, and B. Schölkopf, "Domain generalization via invariant feature representation," in *Proc. 30th IEEE Int. Conf. Mach. Learn.*, Atlanta, GA, USA, Jun. 2013, pp. 10–18.

[14] Z. Xu, W. Li, L. Niu, and D. Xu, "Exploiting low-rank structure from latent domains for domain generalization," in *Proc. 13th Eur. Conf. Comput. Vis.*, Zürich, Switzerland, Sep. 2014, pp. 628–643.

[15] L. Niu, W. Li, and D. Xu, "Multi-view domain generalization for visual recognition," in *Proc. 15th Int. Conf. Comput. Vis.*, Santiago, Chile, Dec. 2015, pp. 4193–4201.

[16] J. Hoffman, B. Kulis, T. Darrell, and K. Saenko, "Discovering latent domains for multisource domain adaptation," in *Proc. 12th Eur. Conf. Comput. Vis.*, Florence, Italy, Oct. 2012, pp. 702–715.

[17] B. Gong, K. Grauman, and F. Sha, "Reshaping visual datasets for domain adaptation," in *Proc. 27th Annu. Conf. Neural Inf. Process. Syst.*, Lake Tahoe, NV, USA, Dec. 2013, pp. 1286–1294.

[18] V. Vapnik and A. Vashist, "A new learning paradigm: Learning using privileged information," *Neural Netw.*, vol. 22, nos. 5–6, pp. 544–557, 2009.

[19] L. Niu, W. Li, and D. Xu, "Visual recognition by learning from Web data: A weakly supervised domain generalization approach," in *Proc. 28th IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, Jun. 2015, pp. 2774–2783.

[20] W. Li, L. Duan, D. Xu, and I. W. Tsang, "Text-based image retrieval using progressive multi-instance learning," in *Proc. 13th Int. Conf. Comput. Vis.*, Barcelona, Spain, Nov. 2011, pp. 2049–2055.

[21] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Proc. 25th Annu. Conf. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2002, pp. 561–568.

[22] Y.-F. Li, J. T. Kwok, I. W. Tsang, and Z.-H. Zhou, "A convex method for locating regions of interest with multi-instance learning," in *Proc. Eur. Conf. Mach. Learn. Principles Pract. Knowl. Discovery Databases*, Bled, Slovenia, Sep. 2009, pp. 15–30.

[23] L. Bruzzone and M. Marconcini, "Domain adaptation problems: A DASVM classification technique and a circular validation strategy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 770–787, May 2010.

[24] J. Huang, A. J. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf, "Correcting sample selection bias by unlabeled data," in *Proc. NIPS*, Whistler, BC, Canada, Dec. 2006, pp. 601–608.

[25] B. Kulis, K. Saenko, and T. Darrell, "What you saw is not what you get: Domain adaptation using asymmetric kernel transforms," in *Proc. 24th IEEE Conf. Comput. Vis. Pattern Recognit.*, Colorado Springs, CO, USA, Jun. 2011, pp. 1785–1792.

[26] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Proc. 25th IEEE Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, Jun. 2012, pp. 2066–2073.

[27] M. Baktashmotlagh, M. T. Harandi, B. C. Lovell, and M. Salzmann, "Unsupervised domain adaptation by domain invariant projection," in *Proc. 14th Int. Conf. Comput. Vis.*, Sydney, NSW, Australia, Dec. 2013, pp. 769–776.

[28] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars, "Unsupervised visual domain adaptation using subspace alignment," in *Proc. 14th Int. Conf. Comput. Vis.*, Sydney, NSW, Australia, Dec. 2013, pp. 2960–2967.

[29] Z. Ding, S. Ming, and Y. Fu, "Latent low-rank transfer subspace learning for missing modality recognition," in *Proc. 28th AAAI Conf. Artif. Intell.*, Québec City, QC, Canada, Jul. 2014, pp. 1–7.

[30] I.-H. Jhuo, D. Liu, D. T. Lee, and S.-F. Chang, "Robust visual domain adaptation with low-rank reconstruction," in *Proc. 25th IEEE Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, Jun. 2012, pp. 2168–2175.

[31] M. Shao, D. Kit, and Y. Fu, "Generalized transfer subspace learning through low-rank constraint," *Int. J. Comput. Vis.*, vol. 109, nos. 1–2, pp. 74–93, 2014.

[32] Z. Ding, M. Shao, and Y. Fu, "Deep low-rank coding for transfer learning," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, Buenos Aires, Argentina, Jul. 2015, pp. 3453–3459.

[33] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: A survey of recent advances," *IEEE Signal Process. Mag.*, vol. 32, no. 3, pp. 53–69, May 2014.

[34] C. Xiong, S. McCloskey, S.-H. Hsieh, and J. J. Corso, "Latent domains modeling for visual domain adaptation," in *Proc. 28th AAAI Conf. Artif. Intell.*, Québec City, QC, Canada, Jul. 2014, pp. 2860–2866.

[35] M. Hoai and A. Zisserman, "Discriminative sub-categorization," in *Proc. 26th IEEE Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, Jun. 2013, pp. 1666–1673.

[36] X. Wang, B. Wang, X. Bai, W. Liu, and Z. Tu, "Max-margin multiple-instance dictionary learning," in *Proc. 30th IEEE Int. Conf. Mach. Learn.*, Atlanta, GA, USA, Jun. 2013, pp. 846–854.

[37] D. Zhang, F. Wang, L. Si, and T. Li, "M^3IC: Maximum margin multiple instance clustering," in *Proc. 21st Int. Joint Conf. Artif. Intell.*, vol. 9. Pasadena, CA, USA, Jul. 2009, pp. 1339–1344.

[38] M.-L. Zhang and Z.-H. Zhou, "M^3MIML: A maximum margin method for multi-instance multi-label learning," in *Proc. 8th IEEE Int. Conf. Data Mining*, Pisa, Italy, Dec. 2008, pp. 688–697.

[39] V. Sharmanska, N. Quadrianto, and C. H. Lampert, "Learning to rank using privileged information," in *Proc. 14th Int. Conf. Comput. Vis.*, Sydney, NSW, Australia, Dec. 2013, pp. 825–832.

[40] S. Fouad, P. Tino, S. Raychaudhury, and P. Schneider, "Incorporating privileged information through metric learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 7, pp. 1086–1098, Jul. 2013.

[41] X. Xu, W. Li, and D. Xu, "Distance metric learning using privileged information for face verification and person re-identification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 12, pp. 3150–3162, Dec. 2015.

[42] P.-A. Absil and A. L. Tits, "Newton-KKT interior-point methods for indefinite quadratic programming," *Comput. Optim. Appl.*, vol. 36, no. 1, pp. 5–41, Jan. 2007.

[43] K. Crammer and Y. Singer, "On the algorithmic implementation of multiclass kernel-based vector machines," *J. Mach. Learn. Res.*, vol. 2, pp. 265–292, Mar. 2001.

[44] M. Kloft, U. Brefeld, S. Sonnenburg, and A. Zien, "ℓ_p-norm multiple kernel learning," *J. Mach. Learn. Res.*, vol. 12, pp. 953–997, Mar. 2011.

[45] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 27:1–27:27, 2011.

[46] R.-E. Fan, P.-H. Chen, and C.-J. Lin, "Working set selection using second order information for training support vector machines," *J. Mach. Learn. Res.*, vol. 6, pp. 1889–1918, Dec. 2005.

[47] P. V. Gehler and S. Nowozin, "Infinite kernel learning," in *Proc. 22th Annu. Conf. Neural Inf. Process. Syst. Workshop Kernel Learn.*, Vancouver, BC, Canada, 2008, pp. 1–4.

[48] J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods*, vol. 3. Cambridge, MA, USA: MIT Press, 1999.

[49] A. Loui *et al.*, "Kodak's consumer video benchmark data set: Concept definition and annotation," in *Proc. 9th ACM SIGMM Int. Workshop Multimedia Inf. Retr.*, Augsburg, Germany, Sep. 2007, pp. 245–254.

[50] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui, "Consumer video understanding: A benchmark database and an evaluation of human and machine performance," in *Proc. 1st ACM Int. Conf. Multimedia Retr.*, Trento, Italy, Apr. 2011, p. 29.

[51] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proc. 14th Int. Conf. Comput. Vis.*, Sydney, NSW, Australia, Dec. 2013, pp. 3551–3558.

[52] J. Donahue *et al.*, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. 31st IEEE Int. Conf. Mach. Learn.*, Beijing, China, Jun. 2014, pp. 647–655.

[53] R. C. Bunescu and R. J. Mooney, "Multiple instance learning for sparse positive bags," in *Proc. 24th IEEE Int. Conf. Mach. Learn.*, Corvallis, OR, USA, Jun. 2007, pp. 105–112.

[54] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[55] L. Torresani, M. Szummer, and A. Fitzgibbon, "Efficient object category recognition using classemes," in *Proc. 11th Eur. Conf. Comput. Vis.*, Crete, Greece, Sep. 2010, pp. 776–789.

[56] J. D. R. Farquhar, D. R. Hardoon, H. Meng, J. S. Shawe-Taylor, and S. Szedmak, "Two view learning: SVM-2K, theory and practice," in *Proc. 19th Annu. Conf. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2005, pp. 355–362.

[57] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural Comput.*, vol. 16, no. 12, pp. 2639–2664, 2004.

Li Niu received the B.E. degree from the University of Science and Technology of China, Hefei, China, in 2011. He is currently pursuing the Ph.D. degree with the Interdisciplinary Graduate School, Nanyang Technological University, Singapore.

His current research interests include machine learning and computer vision.

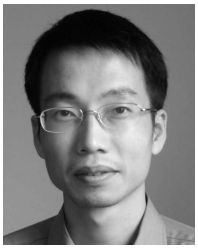
Wen Li received the B.S. and M.Eng. degrees from Beijing Normal University, Beijing, China, in 2007 and 2010, respectively, and the Ph.D. degree from Nanyang Technological University, Singapore, in 2015.

He is currently a Post-Doctoral Researcher with the Computer Visional Laboratory, ETH Zürich, Zürich, Switzerland. His current research interests include transfer learning, multiview learning, multiple kernel learning, and their applications in computer vision.

Dong Xu (M'07–SM'13) received the B.Eng. and Ph.D. degrees from the University of Science and Technology of China, Hefei, China, in 2001 and 2005, respectively.

When pursuing his Ph.D. degree, he was with Microsoft Research Asia, Beijing, China, and The Chinese University of Hong Kong, Hong Kong, for more than two years. He was also a Post-Doctoral Research Scientist with Columbia University, New York, NY, USA, from 2006 to 2007, and a Faculty Member with Nanyang Technological University, Singapore, from 2007 to 2015. He is currently a Professor (Chair in Computer Engineering) with the School of Electrical and Information Engineering, The University of Sydney, Sydney, NSW, Australia. He has authored over 100 papers in the IEEE TRANSACTIONS and top tier conferences.

Dr. Xu's co-authored work on transfer learning for video event recognition received the Best Student Paper Award in the IEEE International Conference on Computer Vision and Pattern Recognition in 2010. The co-authored work also received the IEEE TRANSACTIONS ON MULTIMEDIA Prize Paper Award in 2014.



Jianfei Cai (S'98–M'02–SM'07) received the Ph.D. degree from the University of Missouri–Columbia, Columbia, MO, USA.

He is currently an Associate Professor and the Head of the Visual and Interactive Computing Division, and the Head of the Computer Communication Division with the School of Computer Engineering, Nanyang Technological University, Singapore. He has authored over 170 technical papers in international journals and conferences. His current research interests include computer vision, visual computing, and multimedia networking.

Dr. Cai has been actively participating in program committees of various conferences. He served as the Leading Technical Program Chair of the IEEE International Conference on Multimedia and Expo in 2012 and the Leading General Chair of the Pacific-Rim Conference on Multimedia in 2012. Since 2013, he has been serving as an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING. He also served as an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 2006 to 2013.