Multi-Instance Multi-Label Learning Combining Hierarchical Context and its Application to Image Annotation

Xinmiao Ding, Bing Li, Weihua Xiong, Wen Guo, Weiming Hu, and Bo Wang

***Abstract*—In image annotation, one image is often modeled as a bag of regions ("instances") associated with multiple labels, which is a typical application of multi-instance multi-label learning (MIML). Although lots of research has shown that the interplay embedded among instances and labels can largely boost the image annotation accuracy, most existing MIML methods consider none or partial context cues. In this paper, we propose a novel context-aware MIML model to integrate the instance context and label context into a general framework. Specially, the instance context is constructed with multiple graphs, while the label context is built up through a linear combination of several common latent conceptions that link low level features and high level semantic labels. Comparison with other leading methods on several benchmark datasets in terms of image annotation shows that our proposed method can get better performance than the state-of-the-art approaches.**

***Index Terms*—Image annotation, instance context, label context, multi-instance, multi-label.**

I. Introduction

CONVENTIONAL supervised learning often assumes that an object is represented by a single instance and associated with one class label [1]–[5]. Although this formulation is prevailing and successful, it does not fit many real-world problems very well, for example, an image can include different contents belonging to different classes (e.g. Sky/Grass/Vehicle). To address this issue, another more general framework named multi-instance multi-label learning (MIML) [6], [8] emerged recently, in which an object is allowed to be represented by a bag of instances and associated with multiple class labels simultaneously. Given a training dataset consisting of a collection of bags of instances and each bag is associated with multiple labels, MIML is to learn a classifier that can predict all labels for an unseen bag.

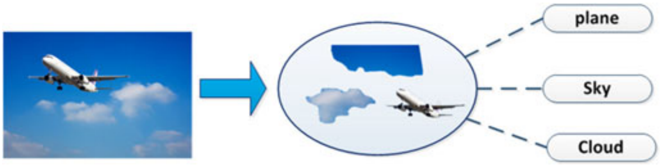


Fig. 1. Example of image annotation with semantic words.

A variety of different applications have been formulated as MIML problems, such as video annotation [13], gene pattern annotation [14], relation extraction in natural language processing [15], sensitive multimedia detection [63]–[65], etc. An important emerging one comes from image annotation, in which an image is usually viewed as a bag of instances (regions) associated with multiple semantic labels [6], [11], [22]. Fig. 1 gives an example of image annotation with semantic words "sky", "plane" and "cloud".

A. Related Work

Unlike traditional multi-class image classifications where the annotation classes are mutually exclusive, each image is often associated with more than one semantic label in annotation task, which poses so-called multi-label learning (MLL). In recent years, a dozen of effective methods have been proposed to solve MLL including unsupervised, supervised or semi-supervised learning [49], [50]–[57]. Barnard *et al.* [49] presented an unsupervised scheme of probabilistic latent variable model to infer relations between visual features of images and associated texts. Wang *et al.* [50] presented a novel supervised topic model to predict class labels and annotation terms simultaneously. Tao *et al.* [53] improved image annotation based on semi-supervised learning algorithm with Hessian regularization, which drives the learned function varying linearly along the data manifold. After that, Liu *et al.* [54], [55] expand the Hessian regularization with multiview features and get excellent performance in image annotation.

All of these MIL approaches always regard an image as one indiscrete entity and neglect a fact that any individual label from an image is more related to some segmented regions in

Manuscript received November 19, 2015; revised March 15, 2016 and May 07, 2016; accepted May 09, 2016. Date of publication May 24, 2016; date of current version July 15, 2016. This work was supported in part by the 973 basic research program of China under Grant 2014CB349303, in part by the Natural Science Foundation of China under Grant 61472421, Grant 61370038, Grant 61303086, Grant 61572296, Grant 61503219, and Grant 61472227, in part by the Strategic Priority Research Program of the CAS under Grant XDB02070003, and in part by the Natural Science Foundation of Shandong Province under Grant ZR2015FL020. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jing-Ming Guo. *(Corresponding author: Bing Li.)*

X. Ding and W. Guo are with the Shandong Technology and Business University, Yantai 264005, China (e-mail: dingxinmiao@126.com; wguo@nlpr.ia.ac.cn).

B. Li, W. Xiong, and B. Wang are with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: bli@nlpr.ia.ac.cn; wallace.xiong@gmail.com; bo.wang2014@nlpr.ia.ac.cn).

W. Hu is with CAS Center for Excellence in Brain Science and Intelligence Technology, National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: wmhu@nlpr.ia.ac.cn).

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/TMM.2016.2572000

it, rather than the entire image. In other words, the multiple semantic meanings (labels) of an image arise from different components (regions) in it. As illustrated in Fig. 1, the three labels "sky", "plane" and "cloud" are characterized by three different regions respectively instead of the entire image. Therefore, multi-instance multi-label learning (MIML) is more suitable for image annotation.

Recent decades have witnessed great progress in MIML algorithms [6]–[11]. They can be roughly classified into two categories, discriminative methods and generative methods. The methods belonging to discriminative category try to model the classification model only on the training data. The initial work can be dated back to MIML-BOOST and MIML-SVM made by Zhou *et al.* [6]. MIML-BOOST transforms training samples into individual set of instances, each of which corresponds to a single label, and apply a typical single label single instance solution, MI-BOOSTING [16], on them. MIML-SVM transforms MIML task into MLL problem. It first generates K medoids in the defined instance feature space and maps each MIML sample into a K-dimensional feature vector based on its Hausdoff distance to these K medoids. After that, the MLL problem was addressed by adopting MLSVM [17] that decomposes the MLL task into a set of single label classification problems. Following these two methods, many other MIML algorithms have been proposed and related applications have also been reported [8]–[12]. Zhang *et al.* [18] proposed a Nearest neighbor approach for MIML; Briggs *et al.* [19] proposed a Rank-Loss Support Instance Machine for MIML instance annotation, and it optimizes a regularized rank-loss object that can be instantiated with different aggregation models connecting instance-level predictions with bag-level predictions; Jin *et al.* [20] proposed an iterative metric learning algorithm for MIML; Huang *et al.* [21] proposed a fast MIML approach which constructed a low-dimensional subspace shared by all labels and trained specific linear models via the efficient stochastic gradient descent.

Generative methods aim at learning a model that can generate the data behind the scenes by estimating the assumptions and distributions of the model. Nguyen *et al.* [22] fused the MIL's feature-word distributions model and topic model to fulfill a MIML framework and applied it into image annotation. Inspired from Latent Dirichlet Allocation, Nguyen *et al.* [23] further extended the standard MIML framework to a multi-modal setting, and presented an advanced model named multi-modal multi-instance multi-label latent Dirichlet allocation (M3LDA). Zha *et al.* [7] proposed a hidden conditional random field model for MIML image annotation. Yang *et al.* [24] proposed a MIML algorithm based on Dirichlet–Bernoulli alignment.

Although many existing MIML approaches have achieved decent performance and validated their superiority in image annotation applications, most of them still ignore two important contextual cues:

1) *Context Among Instances:* Nearly all the existing MIML algorithms treat the instances from a bag as independently and identically distributed (i.i.d.). But Zhou *et al.* [25] have pointed out that the instances in a bag are rarely independent in real tasks, especially in image understanding, better performance can be expected if the relations among instances are considered. Although miGraph proposed in [25] models instance relations as an ε-graph for MIL, it has not yet been extended to MIML. In addition, it is difficult to represent various instance contexts using a single graph structure with fixed ε, as the setting in [25]. Taking Fig. 2 as an example, image in Fig. 2(a) will be viewed as a bag of objects ("instances") and we would like to annotate its conception as "ship" using MIL. Fig. 2(b)–(d) show that the context of an identical image bag [e.g. Fig. 2(a)] may be various under different semantic environment. If the image is considered to be a "travelling ship", then the instance of "ship" will have strong correlation with those of "sea" and "sky" as shown in Fig. 2(b). If it is considered to be "manufacturing ship", the "ship" will consequently have strong correlation with "ground" as shown in Fig. 2(c). If it is considered to be a "harbor", "ship" will have strong correlation with "ground", "building" and "car" as shown in Fig. 2(d). Therefore there is vagueness in terms of which kind of context is intuitively justifiable so that a single graph structure cannot model these various contexts very well.

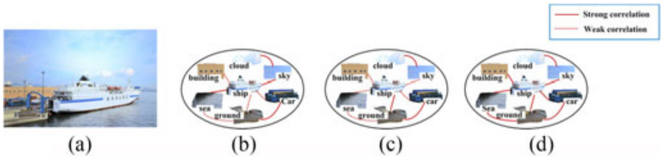


Fig. 2. Example of different instance contexts for identical image. Red solid line represents that the connected nodes have "strong correlation," while red dotted line represents "weak correlation." (a) Example image about "ship." (b) Context of image (a) in which instance "ship" has strong correlation with "sea" and "sky." (c) Context of image (a) in which instance "ship" has strong correlation with "ground." (d) Context of image (a) in which instance "ship" has strong correlation with "building," "ground," and "car."

2) *Context Among Labels:* Most existing MIML algorithms learn an independent classifier for each label without taking the correlation among labels into account. However, much research work on MLL [26]–[29] have showed that semantic terms (i.e. class labels) of each object are not mutually exclusive and such label correlations can largely boost the image annotation accuracy.

B. Our Work

To circumvent these two limitations embedded in the existing MIML methods, we propose a novel context-aware MIML (CMIML) algorithm for image annotation that considers both instance context and label context simultaneously. To model the complex instance contexts, we construct multiple graph structures for each bag to represent the varied relations among instances in it; To express the context among labels, we assume that there exist a set of common latent conceptions between low level features and high level semantic labels, and the high level label is a linear combination of these latent conceptions. The contributions of this paper can be summarized as follows.

1) It introduces multiple ε-graphs to model the complex inner relations among instances in a bag and fuses these graph

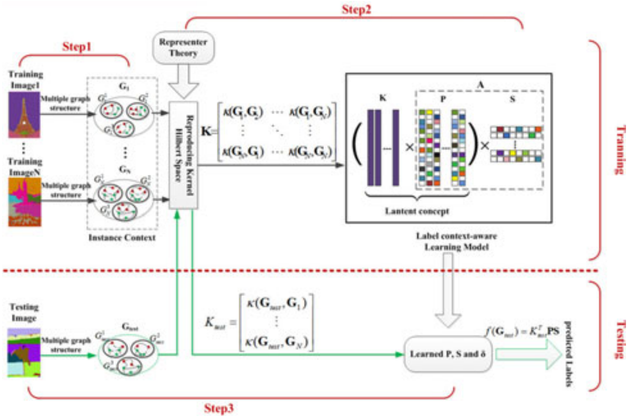

Step1
Step2
Step3
Representer Theory
Training Image1
Multiple graph structure
Training ImageN
Testing Image
G_1
G_N
G_{test}
Instance Context
Reproducing Kernel Hilbert Space
$\mathbf{K}=\begin{bmatrix}\kappa(G_1,G_1) & \cdots & \kappa(G_1,G_N)\\ \vdots & \ddots & \vdots\\ \kappa(G_N,G_1) & \cdots & \kappa(G_N,G_N)\end{bmatrix}$
K
P
S
A
Lantent concept
Label context-aware Learning Model
$K_{test}=\begin{bmatrix}\kappa(G_{test},G_1)\\ \vdots\\ \kappa(G_{test},G_N)\end{bmatrix}$
Learned P, S and δ
$f(G_{test})=K_{test}^T PS$
predicted Labels
Training
Testing

Fig. 3. Framework of the proposed method. In Step 1, multiple graphs are constructed to represent the context of instances. In Step 2, through mapping graphs to RKHS, a multi-label classifier $f(\mathbf{G}_i) = K_i^{\mathrm{T}}\mathbf{A}$ with kernel matrix $\mathbf{K} = [K_1, \ldots, K_N]$ can be deduced based on representer theory. Here the entries of $\mathbf{K}$ are computed by kernel function κ. Then, a latent context-aware learning model is constructed by decomposing the regression coefficient A. After leaning parameters P and S, a test image can be annotated in the following Step 3. In the figure, the black line represents the process of training and the green line represents the process of testing.

structures through a multi-kernel learning. The proposed multiple ε-graph model enables the MIML classifier to consider the diverse contexts among instances.

2) It uses latent conceptions to include the interplays among the class labels. During the training procedure, the proposed CMIML method can effectively learn the latent conceptions and automatically select combination weights for each label to infer the class memberships for each bag.

The remainder of this paper is organized as follows. We briefly introduce the overview of the proposed method in Section II. Section III gives out the construction of instance context. The detail of the proposed context-aware of MIML is presented in Section IV. Section V presents the process of label prediction. The experimental results and analysis are reported in Section VI. Section VII concludes this paper.

II. Overview of CMIML

Before giving an overview of the proposed CMIML, we briefly review the formal definition of the MIML. Let $\partial \subseteq \mathbb{R}^d$ represents the input space of instances and $\mathcal{Y} = \{\pm 1\}^{\mathrm{E}}$ the set of E class labels. We are given a training set $\mathcal{D} = \{\mathbf{B}_i, Y_i\}_{i=1}^N$ with N examples, where $\mathbf{B}_i = \{x_{i,j}\}_{j=1}^{m_i} \subseteq \partial$ is a bag containing m_i instances (suppose that each $x_{i,j}$ is normalized to have unit ℓ^2 norm), and $Y_i = [y_{i,1}, \ldots, y_{i,E}] \in \mathcal{Y}$ is the label vector of bag $\mathbf{B}_i$. $y_{i,e} = +1,\ (e = 1, \ldots, E)$ if bag $\mathbf{B}_i$ is annotated with label e, otherwise $y_{i,e} = -1$. The goal of MIML is to predict all proper labels for any unseen bag, noticing that the number of proper labels is unknown. Mathematically, its task is to learn a decision function $F_{\mathrm{MIML}} : 2^{\partial} \rightarrow 2^{\mathcal{Y}}$ based on $\mathcal{D}$.

After taking into account the context cues among instances and labels, the proposed CMIML model includes three main stages: instance context construction, CMIML based on latent conception, and label prediction. Fig. 3 gives an overview of the framework.

Step 1: Instance Context Construction. We first introduce the ε-graph [31] to construct the contextual relationships among instances in each bag. We define $G_i = \{\mathbf{B}_i, \mathbf{M}_i\}$ as an undirected graph structure for bag $\mathbf{B}_i$, where the instances are represented as the vertices of the graph and $\mathbf{M}_i \in \mathbb{R}^{m_i \times m_i}$ is an adjacency matrix. If the instances $x_{i,a}$ and $x_{i,b}$ are adjacent, there is an edge between them and $\mathbf{M}_i(a,b) = \mathbf{M}_i(b,a) = 1$; otherwise $\mathbf{M}_i(a,b) = \mathbf{M}_i(b,a) = 0$. Furthermore, to model the various relations among instances, we construct multiple graphs for each bag. To this end, we generate Θ graphs to compose a graph set $\mathbf{G}_i = \{G_i^{\theta}\}_{\theta=1}^{\Theta}$ with different values of ε for the bag $\mathbf{B}_i$.

Step 2: CMIML Based on Latent Conception. After instance context construction, the MIML can be re-defined as: given a training set $\Gamma = \{\mathbf{G}_i, Y_i\}_{i=1}^N$, the goal of MIML is to learn a function to map the graph set to its label vector. Since the ε-graph structure cannot be used in classifier learning directly, we map the graphs to the Reproducing Kernel Hilbert Space (RKHS) [32] and further define classifier functions $Y_i = [q_1(\mathbf{G}_i), \ldots q_E(\mathbf{G}_i)]$ on it. However, it is difficult to define the classifier functions $q_e(\cdot), e = 1, \ldots, E$ explicitly. Fortunately, based on the representer theory [30], the function $q_e(\mathbf{G}_i), e = 1, \ldots, E$ is equivalent to $f(\mathbf{G}_i) = K_i{}^{\mathrm{T}}\mathbf{A}$ where $\mathbf{A}$ is a regression coefficient matrix and $K_i \in \mathbb{R}^N$ is the kernel vector that can be computed via a graph kernel function between $\mathbf{G}_i$ and N graph set $\mathbf{G}_1, \mathbf{G}_2, \ldots, \mathbf{G}_N$ in the training set. All the kernel vectors are then stacked to compose a kernel matrix $\mathbf{K} = [K_1, K_2, \ldots, K_N] \in \mathbb{R}^{N \times N}$ of the training set. The details about the computation of the kernel will be discussed in Section IV.

To consider the interplay among labels, we assume that all class labels share a common set of latent conceptions, and each label is a linear combination of these latent conceptions. Consequently, we divide the coefficient matrix $\mathbf{A}$ into two part $\mathbf{P}$ and $\mathbf{S}$, where $K_i{}^{\mathrm{T}}\mathbf{P}$ denotes the latent conceptions and $\mathbf{S}$ denotes the linear combination weights, as shown in Fig. 3. Through such kind of definition, both the correlation among class labels and context cues among instances can be considered simultaneously in the proposed CMIML.

Step 3: Label Prediction. After learning the parameters of the CMIML, given a new unknown bag $\mathbf{B}_{\mathrm{test}}$, we can firstly construct its graph set $\mathbf{G}_{\mathrm{test}}$, and then calculate the kernel vector $K_{\mathrm{test}} \in \mathbb{R}^N$ between the test sample and all the N training samples. Finally, we get its label values through $f(\mathbf{G}_{\mathrm{test}}) = K_{\mathrm{test}}^{\mathrm{T}}\mathbf{PS}$.

III. Instance Context Construction

As Zhou *et al.* [25] indicated, the relations among the instances convey important structure information, e.g. instance context, for many MIL applications. Treating the regions in an image as inter-correlated samples is evidently more meaningful than treating them as independent ones. The case is also true for MIML. In this section, we propose to model the instance context in MIML using multiple graphs and discuss how to construct the graph set $\mathbf{G}_i = \{G_i^{\theta}\}_{\theta=1}^{\Theta}$ for each bag.

In [25], the ε-graph has been successfully used to represent the instance context in MIL. In the ε-graph, every instance in a bag $\mathbf{B}_i$ is regarded as a node. Then, the distance of every pair of nodes, denoted by $x_{i,o}$ and $x_{i,l}$, is computed. If the distance between $x_{i,o}$ and $x_{i,l}$ is smaller than a pre-set threshold ε, an edge is established between them, and the weight value $\mathbf{M}_i(o,l)$ in adjacency matrix $\mathbf{M}_i$ is set as 1, otherwise 0. Finally, a bag of feature vectors of $\mathbf{B}_i$ are reconstructed as an ε-graph G_i which implied the context of instances in each bag.

In the ε-graph, the parameter ε is an important parameter which determines the structure of instance context. In [25], ε is fixed as the average distance in the bag. However, as discussed in Fig. 2, a single ε-graph with any ε value has its own limitations, and cannot represent the complex contexts well in different bags. Therefore we use a graph set $\mathbf{G}_i = \{G_i^\theta\}_{\theta=1}^\Theta$ to represent different contexts among instances in the bag $\mathbf{B}_i$, and learn to fuse them through linear weighting. To obtain the graph set $\mathbf{G}_i = \{G_i^\theta\}_{\theta=1}^\Theta$, we generate different graphs with different ε values. Thus three typical values of ε are selected to generate three kinds of typical graphs, as follows.

1) Set $\varepsilon = 0$, meaning that there's not any edge between every two instances and all instances are treated to be independent, shown in Fig 2(a). This is can be viewed as a special context.
2) Set ε value as the average distance in the bag as[25]. The graph with this adaptive parameter setting takes the local manifold in each bag into account.
3) Set ε to be a selected value through the cross validation on each training set. The parameter selection using this strategy can achieve better overall performance on the training set. The graph structure with the selected ε value is suitable for most samples, and is an optimal value on the training set.

For convenience, we depict the ε value of above three cases as $\varepsilon = 0, \varepsilon = avg$, and $\varepsilon = opt$ respectively. Denote the ε-graphs for the bag $\mathbf{B}_i$ corresponds to the three parameter selections as G_i^1, G_i^2, G_i^3 and a graph set $\mathbf{G}_i$ is defined as $\mathbf{G}_i = \{G_i^1, G_i^2, G_i^3\}$.

IV. CMIML Based on Latent Conception

In this section, we design a CMIML framework that can consider the instance context and label context simultaneously.

A. ε-Graph Embedded MIML Based on Representer Theory

1) Multi-Label Classifier on the RKHS: After graph construction for each bag, the bags can be rewritten as $\{\mathbf{G}_i, Y_i\}_{i=1}^N$ in which $\mathbf{G}_i$ is the multiple ε-graphs containing the instance context cue of a bag. Since the graph structure cannot be directly used for learning a classifier, we introduce the RKHS [32] to solve this issue in the kernel form.

Let $\wp$ be the bag graph space. We define a map from $\wp$ into a RKHS $\mathcal{H}$ which is a space of functions mapping $\wp$ into $\mathbb{R}$, denoted as $\mathbb{R}^\wp$, via $\varphi : \wp \to \mathbb{R}^\wp$[58]. Suppose there exist E classification functions $Q = \{q_e\}_{e=1}^E$ on $\mathcal{H}$ for E labels, where $q_e : \wp \to \mathbb{R}$ corresponds to the classification function for the label e, we can get the predicted annotation Y_i^* of bag $\{\mathbf{B}_i, \mathbf{G}_i\}$ as

$$Y_i^* = [q_1(\mathbf{G}_i), \dots q_E(\mathbf{G}_i)]. \tag{1}$$

To learn the functions $Q = \{q_e\}_{e=1}^E$, the objective function based on E independent label classifiers can be written as

$$\min_Q \frac{1}{E} \sum_{e=1}^E \left\{ \frac{1}{N} \sum_{i=1}^N (q_e(\mathbf{G}_i) - y_{i,e})^2 + \gamma \|q_e\|_{\mathcal{H}}^2 \right\} \tag{2}$$

where $\|\cdot\|_{\mathcal{H}}$ is a norm in the RKHS $\mathcal{H}$ and $\gamma\|q_e\|_{\mathcal{H}}^2$ is an regularization term to confirm robustness.

2) Learning Model Deducing Based on Representer Theorem: Since (2) cannot be solved through numerical method due to the infinite-dimensional property of RKHS, we use the representer theory [30] to reduce the optimization problem from a possibly infinite-dimensional space to a finite-dimensional space. In this section we firstly construct the requirements for representer theorem, then an equivalent optimization solution for (2) is given out.

i) Valid kernel definition for representer theorem

Given an arbitrary bag $\mathbf{B}_i$ and its instance context $\mathbf{G}_i$ which is composed of multiple ε-graphs $\{G_i^1, G_i^2, G_i^3\}$, we define a mapping function $\varphi_\theta : \wp \to \mathbb{R}^\wp$, $\theta = 1, 2, 3$, to map each ε-graph G_i^θ to RKHS $\mathcal{H}$. Then we can get a kernel $\kappa_{\text{graph}}^\theta$ on the θ^{th} ε-graph between any two bags $\mathbf{B}_i$ and $\mathbf{B}_j$ [25]

$$\begin{aligned} \kappa_{\text{graph}}^\theta(G_i^\theta, G_j^\theta) &= < \varphi_\theta(G_i^\theta), \varphi_\theta(G_j^\theta) > \\ &= \frac{\sum\limits_{a=1}^{m_i} \sum\limits_{b=1}^{m_j} \omega_{i,a}^\theta \omega_{j,b}^\theta \kappa_{\text{ins}}(x_{i,a}, x_{j,b})}{\sum\limits_{a=1}^{m_i} \omega_{i,a}^\theta \sum\limits_{b=1}^{m_j} \omega_{j,b}^\theta} \end{aligned} \tag{3}$$

where $\omega_{i,a}^\theta = 1/\sum_{u=1}^{m_i} \mathbf{M}_i^\theta(a,u)$, $\omega_{j,b}^\theta = 1/\sum_{u=1}^{m_j} \mathbf{M}_j^\theta(b,u)$, $\mathbf{M}_i^\theta$ and $\mathbf{M}_j^\theta$ are the adjacency weights matrixes for bag $\mathbf{B}_i$ and $\mathbf{B}_j$ with the θ^{th} graph structure discussed in Section III. In addition, $\kappa_{\text{ins}}(x_{i,a}, x_{j,b})$ is defined using Gaussian radial basis function kernel: $\kappa_{\text{ins}}(x_{i,a}, x_{j,b}) = \exp(-\gamma\|x_{i,a} - x_{j,b}\|^2)$.

To fuse these three instance context structures $\{G_i^1, G_i^2, G_i^3\}$, we introduce a parameter $\delta = [\delta^1, \delta^1, \delta^3]^{\mathrm{T}}$ and define a new multiple graph kernel between bag $\mathbf{B}_i$ and $\mathbf{B}_j$ based on multi-kernel learning [59]

$$\begin{aligned} &\kappa_{\text{graph}}(\mathbf{G}_i, \mathbf{G}_j) \\ &= \delta^1 < \varphi_1(G_i^1), \varphi_1(G_j^1) > + \delta^2 < \varphi_2(G_i^2), \varphi_2(G_j^2) > \\ &\quad + \delta^3 < \varphi_3(G_i^3), \varphi_3(G_j^3) > \\ &= \delta^1 \kappa_{\text{graph}}^1(G_i^1, G_j^1) + \delta^2 \kappa_{\text{graph}}^2(G_i^2, G_j^2) + \delta^3 \kappa_{\text{graph}}^3(G_i^3, G_j^3) \\ &\qquad \text{s.t.} \sum_{\theta=1}^3 \delta^\theta = 1, \delta^\theta \geq 0, \theta = 1, 2, 3. \end{aligned} \tag{4}$$

Definition 1 (Kernel matrix) [30]: Given a kernel κ and patterns $x_1, \dots, x_m \in \aleph$, the $m \times m$ matrix

$$\mathsf{K} := (\kappa(x_i, x_j))_{i,j=1}^m \tag{5}$$

is the Kernel matrix of κ with respect to $x_1, \dots, x_m$. We can get two corresponding Lemmas:

Lemma 1 (The proof is in Appendix A): If κ is a valid real-valued kernel, then its corresponding kernel matrix is symmetric, positive semidefinite.

Lemma 2 (The proof is in Appendix B): κ_{graph} is a valid positive definite real-valued kernel on $\wp \times \wp$.

ii) Representer theorem

Lemma 3 (Nonparametric Representer Theorem) [30]: Suppose we are given a nonempty set χ, a positive definite real-valued kernel k on $\chi \times \chi$, a training sample $(x_1, y_1), \dots, (x_m, y_m) \in \chi \times \mathbb{R}$, a strictly monotonically increasing real-valued function g on $[0, \infty]$, an arbitrary cost function L: $(\chi \times \mathbb{R}^2)^m \to \mathbb{R} \cup \{\infty\}$, and a class of functions

$$F = \{f \in \mathbb{R}^{\chi} | f(\cdot) = \sum_{i=1}^{\infty} \beta_i k(\cdot, z_i), \beta_i \in \mathbb{R}, z_i \in \chi, \|f\| < \infty\}. \quad (6)$$

Here, $\|\cdot\|$ is the norm in the RKHS $\mathcal{H}$ associated with k, i.e. for any $z_i \in \chi,\ \beta_i \in \mathbb{R}\ (i \in N)$

$$\left\| \sum_{i=1}^{\infty} \beta_i k(\cdot, z_i) \right\|^2 = \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \beta_i \beta_j k(z_i, z_j). \quad (7)$$

$f \in \mathsf{F}$ aims at minimizing the regularized risk functional

$$\mathsf{L}((x_1, y_1, f(x_1)), \dots, (x_m, y_m, f(x_m))) + g(\|f\|) \quad (8)$$

with the form

$$f(\cdot) = \sum_{i=1}^{m} \alpha_i k(\cdot, x_i). \quad (9)$$

iii) Deducing of the equivalent form of (2)

According to the *Lemmas 2* and 3, the function $q_e(\mathbf{G}_i)$ in (2) has the representation

$$q_e(\cdot) = \sum_{i=1}^{N} \alpha_{e,i} \kappa_{\text{graph}}(\cdot, \mathbf{G}_i),\ e = 1, \dots, E. \quad (10)$$

Let $\alpha_e = [\alpha_{e,1}, \dots, \alpha_{e,N}]^T \in \mathbb{R}^N$, $K_i = [\kappa_{\text{graph}}(\mathbf{G}_i, \mathbf{G}_1), \dots, \kappa_{\text{graph}}(\mathbf{G}_i, \mathbf{G}_N)]^T \in \mathbb{R}^N$, then a kernel matrix $\mathbf{K} = [K_1, \dots, K_N]$ will include all mappings of training bags. We substitute (10) into (2) and obtain the following objective function:

$$\min_{\alpha_e} \frac{1}{E} \sum_{e=1}^{E} \left\{ \frac{1}{N} \sum_{i=1}^{N} \left(K_i^T \alpha_e - y_{i,e}\right)^2 + \gamma \left\|\alpha_e\right\|_{\mathcal{H}}^2 \right\} \quad (11)$$

where $\gamma \|\alpha_e\|_{\mathcal{H}}^2$ is the regularization term, $K_i = \sum_{\theta=1}^{3} \delta^\theta K_i^\theta$, K_i^θ is the ith column of the θth kernel matrix $\mathbf{K}^\theta$. Based on Lemma 3, $\|\alpha_e\|_{\mathcal{H}}^2$ can be denoted as $\|\alpha_e\|_{\mathcal{H}}^2 = \alpha_e^T \mathbf{K} \alpha_e$. With slack of triangle inequality, we can get the equivalent form of (11) as following:

$$\min_{\alpha_e} \frac{1}{E} \sum_{e=1}^{E} \left\{ \frac{1}{N} \sum_{i=1}^{N} \left(K_i^T \alpha_e - y_{i,e}\right)^2 + \gamma \left\|\alpha_e\right\|_2^2 \right\}. \quad (12)$$

If we set $\mathbf{A} = [\alpha_1, \dots, \alpha_e, \dots, \alpha_E]$ and $\mathbf{Y} = [Y^1, \dots, Y^e, \dots, Y^E]$ $(Y^e = [y_{1,e}, \dots, y_{N,e}]^{\mathrm{T}})$ as the label matrix containing labels of all bags, the (12) can be rewritten as

$$\min_{\mathbf{A}} \frac{1}{EN} \left\|\mathbf{K}^T \mathbf{A} - \mathbf{Y}\right\|_{\mathrm{F}}^2 + \gamma' \left\|\mathbf{A}\right\|_{\mathrm{F}}^2 \quad (13)$$

where $\gamma' = \gamma / N$ and $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm of matrix avoiding the overfitting. Now the optimization of (13) is equivalent to (2).

Since $\mathbf{K} = \sum_{\theta=1}^{3} \delta^\theta \mathbf{K}^\theta$, we can learn optimal kernel weights $\delta = [\delta^1, \delta^2, \delta^3]^{\mathrm{T}}$, from training set by adding a regularization term $\|\delta\|_2^2$ to avoid overfitting

$$\min_{\mathbf{A}, \delta} \frac{1}{EN} \left\|\mathbf{K}^T \mathbf{A} - \mathbf{Y}\right\|_{\mathrm{F}}^2 + \gamma' \left\|\mathbf{A}\right\|_{\mathrm{F}}^2 + \zeta \left\|\delta\right\|_2^2$$
$$\text{s.t.} \sum_{\theta=1}^{3} \delta^\theta = 1, \delta^\theta \geq 0, \theta = 1, 2, 3. \quad (14)$$

B. CMIML Based on Latent Conceptions

Although the objective function in (14) can now learn the multiple labels, it ignores their relations that will degrade the performance of label learning [21], [28], [33]. Thus we propose a set of latent conceptions to solve it [66], [67].

Assuming that there are c latent conceptions and each observed label can be represented as linear combination of a subset of these latent conceptions, we can obtain the weight matrix $\mathbf{A}$ as

$$\mathbf{A} = \mathbf{PS} \quad (15)$$

where $\mathbf{P}$ is a matrix of size $N \times c$ with each column p_i representing the learning parameter vector of a latent conception $\mathbf{K}^{\mathrm{T}} p_i, i = 1, \dots, c$. $\mathbf{S} = [s_1, \dots, s_E]$ is a matrix of size $c \times E$ containing the weights of linear combination for each conception. Now, we will learn the latent conception learning matrix $\mathbf{P}$ and the combination weight matrix $\mathbf{S}$, rather than weight matrix $\mathbf{A}$. Such decomposition enables different labels to share similar visual patterns which are represented by latent conception, and related labels are expected to help each other.

However, considering that not all labels are actually related to each other, we further enforce latent conception to be selectively shared by different observed labels. Formally, we apply the ℓ_1 norm regularization on combination weight s_e of each label $e = 1, \dots, E$. As a result, each classification model is reconstructed by a small number of latent conceptions, which equivalently forces latent conceptions to be shared only among those related labels. The CMIML classification model is now formulated as follows:

$$\min_{\mathbf{P}, \mathbf{S}, \delta} \frac{1}{E} \sum_{e=1}^{E} \left\{ \frac{1}{N} \left\|\mathbf{K}^{\mathrm{T}} \mathbf{P} s_e - Y^e\right\|_2^2 + \mu \|s_e\|_1 \right\} + \lambda \left\|\mathbf{P}\right\|_{\mathrm{F}}^2 + \zeta \left\|\delta\right\|_2^2$$
$$\text{s.t.} \sum_{\theta=1}^{3} \delta^\theta = 1, \delta^\theta \geq 0, \theta = 1, 2, 3 \quad (16)$$

where $\|\mathbf{P}\|_{\mathrm{F}}^2 = \mathrm{trace}(\mathbf{P}\mathbf{P}^{\mathrm{T}})$ is Frobenius norm that targets at avoiding the overfitting and $\|s_e\|_1$ enables the model to learn a sparse linear combination of latent conception for each observed label. $\mu,\ \lambda$ and ζ are regularization parameters.

C. Model Learning

In the proposed model, three parameters need to be determined: δ, $\mathbf{S}$ and $\mathbf{P}$. We notice that, although the cost function in (16) is not jointly convex in these three parameters, it is convex in one parameter if the other two are fixed. Hence we adopt alternating optimization strategy that converges to a local minimum.

Our optimization procedure can be outlined as three steps:

Step 1: For fixed $\mathbf{P}$ and δ, we learn the combination weight matrix $\mathbf{S}$ by solving the following optimization problem that will be decomposed into individual problems for s_e

$$s_e = \frac{1}{N} \arg\min_{s} \left\| \mathbf{K}^{\mathrm{T}} \mathbf{P} s - Y^e \right\|_2^2 + \mu \|s\|_1. \tag{17}$$

Since this optimization problem is non-smooth due to the ℓ_1 norm regularization of s, we use two-metric projection method that has superlinear convergence [34], [35].

Step 2: For fixed $\mathbf{S}$ and δ, we obtain the optimal $\mathbf{P}$ from (16) by solving the following optimization problem:

$$\min_{\mathbf{P}} \frac{1}{E} \sum_{e=1}^{E} \left\{ \frac{1}{N} \left\| \mathbf{K}^{\mathrm{T}} \mathbf{P} s_e - Y^e \right\|_2^2 \right\} + \lambda \left\| \mathbf{P} \right\|_{\mathrm{F}}^2 . \tag{18}$$

Since this problem is convex in $\mathbf{P}$ and has a closed form solution for squared loss function, we can easily solve it through derivation.

Step 3: For fixed $\mathbf{P}$ and $\mathbf{S}$, (16) is actually equivalent to

$$\begin{aligned} &\min_{\delta} \frac{1}{EN} \left\| \sum_{\theta=1}^{3} \delta^{\theta} \left(\mathbf{K}^{\theta}\right)^{\mathrm{T}} \mathbf{PS} - \mathbf{Y} \right\|_{\mathrm{F}}^{2} + \zeta \left\| \delta \right\|_2^2 \\ &\text{s.t.} \sum_{\theta=1}^{3} \delta^{\theta} = 1, \delta^{\theta} \geq 0, \theta = 1, 2, 3. \end{aligned} \tag{19}$$

(19) is a non-linear programming (NLP) problem.

The alternating optimization procedure is terminated when there is little change in $\mathbf{P}$, $\mathbf{S}$ or δ between two consecutive iterations. Algorithm 1 outlines several major steps as well as initialization procedure. And we will detail the optimization procedure in the following.

Model Initialization: The first step of our optimization algorithm is to initialize δ and the latent conception matrix $\mathbf{P}$. δ can be initialized with $[1/3, 1/3, 1/3]$. Since $\mathbf{P}$ corresponds to the latent conception in which the relations among the column vectors from it are required to be as far as possible, we learn all independent classification models on the training data separately, pack all trained weight vector column by column into a single one $\mathbf{A}_0$ and set the selected top-c left singular vectors of $\mathbf{A}_0$ as $\mathbf{P}$.

Optimizing $\mathbf{S}$ *With Fixed* $\mathbf{P}$ *and* δ*:* For a fixed $\mathbf{P}$ and δ, we need the gradient and Hessian of the squared loss function $f(s) = \frac{1}{N} \|\mathbf{K}^{\mathrm{T}} \mathbf{P} s - Y^e\|_2^2$ to optimize s_e using two-metric projection method

$$\nabla_{s_e} f(s) = \frac{2}{N} \mathbf{P}^{\mathrm{T}} \mathbf{K} (\mathbf{K}^{\mathrm{T}} \mathbf{P} s_e - Y^e) \tag{20}$$

$$\nabla_{s_e}^2 f(s) = \frac{2}{N} \mathbf{P}^{\mathrm{T}} \mathbf{K} \mathbf{K}^{\mathrm{T}} \mathbf{P}. \tag{21}$$

Algorithm1: Context-aware MIML.

```
Input:
$\mathbf{K}$: Kernel matrix of training data
$c$: Number of latent conception
$\mu$, $\lambda$, $\zeta$: Regularization parameters
Output: Observed label predictor matrix $\mathbf{A}$, $\mathbf{P}$ and $\mathbf{S}$.
1: Learn individual predictors for each observed label
   without any label sharing.
2: Let $\mathbf{A}_0$ be the matrix that contains these initial predictors
   as columns.
3: Compute top-c singular vectors: $\mathbf{A}_0 = \mathbf{U\Sigma V}^{\mathrm{T}}$
4: Initialize $\mathbf{P}$ to first $c$ columns of $\mathbf{U}$.
5: Initialize $\boldsymbol{\delta} = [1/3, 1/3, 1/3]$.
while not converged do
  for $e = 1$ to $E$ do
    6: Solve (17) to obtain $s_e$.
  end for
  7: Construct matrix $\mathbf{S} = [s_1, \cdots, s_E]$.
  8: Fix $\mathbf{S}$ and $\delta$ solve (18) to obtain $\mathbf{P}$.
  9: Fix $\mathbf{S}$ and $\mathbf{P}$ to learning $\delta$ through Reduced Gradient.
end while
10: Return outputs: $\mathbf{P}$, $\mathbf{S}$ and $\mathbf{A} = \mathbf{PS}$.
```

Optimizing $\mathbf{P}$ *With Fixed* $\mathbf{S}$ *and* δ*:* For a fixed $\mathbf{S}$ and δ, equating the gradient of (18) to zero can produce

$$\frac{1}{EN} \sum_{e=1}^{E} \mathbf{K} Y^e s_e^{\mathrm{T}} = \frac{1}{EN} \sum_{e=1}^{E} \mathbf{K}\mathbf{K}^{\mathrm{T}} \mathbf{P} s_e s_e^{\mathrm{T}} + \lambda \mathbf{P}. \tag{22}$$

Now we can simply apply a vectorization operator on both sides to solve the linear equation in which all columns of a matrix are stacked one by one to form a long vector

$$\begin{aligned} \frac{1}{EN} \sum_{e=1}^{E} \mathrm{vec}(\mathbf{K} Y^e s_e^{\mathrm{T}}) &= \mathrm{vec} \left(\frac{1}{EN} \sum_{e=1}^{E} \mathbf{K}\mathbf{K}^{\mathrm{T}} \mathbf{P} s_e s_e^{\mathrm{T}} + \lambda \mathbf{P} \right), \\ &= \left[\frac{1}{EN} \sum_{e=1}^{E} (s_e s_e^{\mathrm{T}}) \otimes (\mathbf{K}\mathbf{K}^{\mathrm{T}}) + \lambda \mathbf{I} \right] \mathrm{vec}(\mathbf{P}). \end{aligned} \tag{23}$$

Here we use $\mathrm{vec}(\cdot)$ to represent the vectorization operator and can get $\mathrm{vec}(\mathbf{ODF}) = (\mathbf{F}^{\mathrm{T}} \otimes \mathbf{O})\mathrm{vec}(\mathbf{D})$ after Kronecker product. (23) is obviously a standard form of system of linear equations that is full rank and has a unique solution. It can be easily solved using LU decomposition [36] or by iterative methods, both of which are much faster and numerically more stable than matrix inversion operation.

Optimizing δ *With Fixed* $\mathbf{S}$ *and* $\mathbf{P}$*:* If we write down matrix $\mathbf{\Pi} = diag((\mathbf{K}^1)^{\mathrm{T}} \mathbf{PS}, (\mathbf{K}^2)^{\mathrm{T}} \mathbf{PS}, (\mathbf{K}^3)^{\mathrm{T}} \mathbf{PS})$, (19) can be simplified as

$$\begin{aligned} &\min_{\delta} \frac{1}{EN} \left\| \delta^T \mathbf{\Pi} - \mathbf{Y} \right\|_{\mathrm{F}}^2 + \zeta \left\| \delta \right\|_2^2 , \\ &\text{s.t.} \ [1, 1, 1] \begin{bmatrix} \delta^1 \\ \delta^2 \\ \delta^3 \end{bmatrix} = 1, \begin{bmatrix} \delta^1 \\ \delta^2 \\ \delta^3 \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \end{aligned} \tag{24}$$

Obviously, (24) has the standard form of NLP with linear constrain as following:

$$\min_{\delta} f(\delta)$$
$$\text{s.t. } \mathbf{A}\delta = b \quad \delta \geq 0 \tag{25}$$

and (25) can be solved by Reduced Gradient[61].

V. Label Prediction

After learning the latent conception learning matrix $\mathbf{P}$ and the combination weight matrix $\mathbf{S}$, we can obtain a linear regression function for all labels as following:

$$f(\mathbf{G}_i) = K_i^{\mathrm{T}} \mathbf{PS} \tag{26}$$

where $\mathbf{G}_i$ is the multiple graph structure of bag $\mathbf{B}_i$ and $K_i = \sum_{\theta=1}^{3} \delta^{\theta} K_i^{\theta}$ is the multi-graph kernel of $\mathbf{G}_i$ with all training bag graphs.

For a new test sample $\mathbf{B}_{\text{test}}$, we firstly construct the multiple bag graph $\mathbf{G}_{\text{test}}$ and compute its kernel with all training bag graphs K_{test}, then we can calculate regression values for all labels using (26). If a label has positive regression value, it is considered to be proper for bag $\mathbf{B}_{\text{test}}$, otherwise the label is abandoned.

VI. Experiments

In this section, we compare our method with several state-of-the-art MIML methods on the image annotation task, including DBA [24], KISAR [37], MIMLBoost [6], MIMLkNN [18], MIMLSVM [6], RankLoss-SIM [19] and Fast MIML [21]. All experiments were conducted on five benchmark image sets (Scenes [6], [8], MSRCv2 [41], Corel5k [42], MSRA_MM [44] and IAPS [45]).

A. Evaluation Criteria

We evaluate the performance of all the compared MIML approaches using five commonly used criteria: *hamming loss, one-error, coverage, ranking loss and average precision* [38], [39].

1) The *hamming loss* evaluates how many times an object-label pair is misclassified, i.e., a proper label is missed or a wrong label is predicted.
2) The *one-error* evaluates how many times the top-ranked label is not a proper label of the object.
3) The *coverage* evaluates how far it is needed, on the average, to go down the list of labels in order to cover all the proper labels of the object. It is loosely related to precision at the level of perfect recall.
4) The *ranking loss* evaluates the average fraction of label pairs that are mis-ordered for the object.
5) The *average precision* evaluates the average fraction of proper labels ranked above a particular label.

For the first four metrics, smaller value means better performance; on the contrary, the larger the value of *average precision* indicates the better performance of the technique.

B. Data Set Descriptions and Preprocessing

Five image sets, Scenes, MSRC v2, Corel5k, MSRA-MM and IAPS, are used in the following experiments.

1) *Scenes:* This dataset consists of 2000 natural scene images belonging to several classes: desert, mountains, sea, sunset and trees. Over 22% of these images belong to multiple classes simultaneously. Each image has already been represented as a bag of nine instances generated by the SBN method [40], the method uses a Gaussian filter to smooth the image and then subsamples the image to an 8×8 matrix of color blobs in which each blob is a 2×2 set of pixels. An instance corresponding to the combination of a single blob with its four neighboring blobs (up, down, left, right) is described with 15 features.
2) *MSRC v2:* This dataset, named 'v2' [41], is a subset of the Microsoft Research Cambridge (MSRC) image dataset. It contains 591 images and 23 classes. Around 80% images are associated with more than one label and there are around three labels per image on average. These labels often arise from respective regions in the images. MSRC data set also provides pixel level ground truth, where each pixel is labeled as one of 23 classes or "void". "horse" and "mountain" are also treated as "void" since they have few positive samples. Thus there are 21 labels in total. Each image is treated as a bag and each contiguous region in the ground-truth segmentation as an instance [19]. Each instance is described by a 16-dimensional histogram of gradients, and a 32-dimensional histogram of colors.
3) *Corel5k:* This dataset has become the benchmark for image annotation recently [42]. The dataset contains 5000 images collected from the larger Corel CD set. The whole set consists of 50 groups. There are 100 similar images in each group, such as beach, aircraft and tiger. The set is annotated from a dictionary of 260 keywords(labels), with each image having been annotated by an average of 3.5 keywords(labels). In our experiments, each image is segmented by Normalized Cuts [62] and each region in the image is regarded as an instance described by nine features [43]. There are typically 5–10 regions for each image.
4) *MSRA-MM:* This dataset [44] is collected by Microsoft Research Asia. There are around 1 million web images acquired by Live Image Search using different predefined queries. The queries are manually classified into eight categories, i.e., "Animal", "Cartoon", "Event", "Object", "Scene", "PeopleRelated", "NamedPerson", and "Misc". Among these 1 million images, there are 50 000 images that are manually annotated with ground-truth labels. Each image from the MSRA-MM database is labeled as positive or negative with respect to each concept. There are 100 concepts in total. Detailed information of this database can be found in [44]. Each image is segmented and represented with the same method as that for Corel5K. 2000 annotated images are randomly selected from this data set for comparison in our experiment.

TABLE I
ANNOTATION RESULTS ON FIVE MIML DATA SETS

	Compared Algorithms							
Criteria	CMIML	MIMLfast	KISAR	DBA	MIMLBoost	MIMLkNN	MIMLSVM	RankL.SIM
				Scene				
h.l.↓	**.009 ± .008**	.188 ± .009	.194 ± .005	.269 ± .009	N/A	.196 ± .007	.200 ± .008	.204 ± .007
o.e.↓	**.175 ± .019**	.351 ± .023	.351 ± .020	.386 ± .025	N/A	.370 ± .018	.380 ± .021	.392 ± .019
Co.↓	**.143 ± .010**	.207 ± .012	.204 ± .008	.334 ± .011	N/A	.222 ± .009	.225 ± .010	.237 ± .010
r.l.↓	**.100 ± 0.14**	.189 ± .014	.185 ± .010	.348 ± .012	N/A	.207 ± .011	.212 ± .011	.222 ± .010
a.p.↑	**.841 ± 0.13**	.770 ± .015	.772 ± .012	.600 ± .013	N/A	.757 ± .011	.750 ± .012	.738 ± .011
				MSRCv2				
h.l.↓	**.071 ± .011**	.100 ± .007	.086 ± .004	.140 ± .006	N/A	.131 ± .007	.084 ± .003	.110 ± .004
o.e.↓	**.251 ± 0.28**	.295 ± .025	.341 ± .031	.415 ± .026	N/A	.440 ± .031	.320 ± .029	.302 ± .028
Co.↓	**.188 ± .013**	.238 ± .014	.254 ± .015	.837 ± .018	N/A	.312 ± .020	.256 ± .018	.239 ± .013
r.l.↓	**.071 ± 0.14**	.108 ± .009	.131 ±.010	.675 ± .017	N/A	.165 ± .013	.125 ± .011	.107 ± .007
a.p.↑	**.750 ± .014**	.688 ± .017	.666 ±.018	.326 ± .016	N/A	.591 ± .018	.685 ± .018	.687 ± .013
				Corel5k				
h.l.↓	**.009 ± .021**	.015 ± .020	.017 ± .025	.024 ± .020	N/A	.015 ± .018	.020 ± .019	N/A
o.e.↓	**.319 ± .008**	.624 ± .006	.657 ± .009	.801 ± .005	N/A	.656 ± .007	.802 ± .007	N/A
Co.↓	**.268 ± .011**	.312 ± .012	.781 ± .010	.991 ± .011	N/A	.551 ± .009	.501 ± .011	N/A
r.l.↓	**.109 ± .013**	.183 ± .014	.512 ± .008	.911 ± .012	N/A	.281 ± .011	.280 ± .010	N/A
a.p.↑	**.531 ± .012**	.352 ± .015	.198 ± .014	.051 ± .011	N/A	.301 ± .012	.199 ± .011	N/A
				MSRA-MM				
h.l.↓	.031 ± 0.21	**.029 ± .018**	.031 ± .007	.032 ± .006	N/A	.034 ± .017	.031 ± .015	N/A
o.e.↓	**.519 ± .014**	.576 ± .145	.611 ± .027	.945 ± .016	N/A	.601 ± .006	.645 ± .084	N/A
Co.↓	.228 ± .020	**.221 ± .027**	.301 ± .115	.998 ± .031	N/A	.344 ± .024	.256 ± .011	N/A
r.l.↓	.135 ± .013	.121 ± .017	.159 ± .018	.954 ± .034	N/A	.185 ± .081	.137 ± .014	N/A
a.p.↑	**.452 ± .017**	.441 ± .035	.398 ± .012	.042 ± .016	N/A	.391 ± .014	.401 ± .006	N/A
				IAPS				
h.l.↓	**.172 ± 0.08**	.370 ± .035	.222 ± .006	—	.204 ± .002	.303 ± .007	.258 ± .006	—
o.e.↓	**.371 ± .021**	.770 ± .027	.859 ± .027	—	.653 ± .040	.719 ± .017	.733 ± .024	—
Co.↓	**.252 ± .008**	.449 ± .144	.489 ± .115	—	.492 ± .094	.555 ± .084	.472 ± .081	—
r.l.↓	**.203 ± .010**	.449 ± .019	.502 ± .018	—	.418 ± .015	.447 ± .011	.375 ± .014	—
a.p.↑	**.698 ± .011**	.528 ± .017	.472 ± .012	—	.470 ± .024	.461 ± .010	.473 ± .015	—

(‘↓’ indicates ‘the smaller the better’; ‘↑’ indicates ‘the larger the better’; ‘—’ indicates that the results cannot be got for no opening code and published results; h.l.: hamming loss, o.e.: one-error, Co.: coverage, r.l.: ranking loss and a.p.: average precision). N/A indicates that no result was obtained in 24 hours.

5) *IAPS:* The IAPS set [45] is a common stimulus set frequently used in emotion research. It consists of 716 natural colored pictures taken by professional photographers. They depict complex scenes containing objects, people, and landscapes. All pictures are categorized in emotional valence (positive, negative, no emotion) [45]. A subset of 396 IAPS images are used in our experiment, each of which is labeled either as one specific emotion or as a mixture of several emotions: anger, awe, disgust, fear, sadness, excitement, contentment, and amusement [46]. Note that any single picture can belong to different emotion. The images are segmented using waterfall segmentation and represented with composition features including color, texture and others [47].

C. Experimental Results

In this section, we compare our method CMIML with the state-of-the-art MIML methods on the Scene, MSRCv2, MSRM_MM, Corel5k and IAPS datasets.

In the experiments, for each data set, 2/3 of the data are randomly sampled to compose the training set, while the remaining examples are used as the test set. We repeat the procedure 30 times and report the average results. The parameters are selected by 3-fold cross validation on the training data with regard to the average precision. Table I shows the performance of our method and other 7 MIML methods. From Table I, the following points were revealed.

1) Our method based on hierarchical context can achieve the best performance on almost overall evaluation criteria, about max 7%, 6% and 17% improvement of the average precision on four datasets and max 0.4 decrease on other evaluations. It indicates that the fusion of two context cues is effective in improving the performance of MIML on the image annotation.
2) Our method, as well as MIMLfast, outperforms other MIML methods since these two methods model relationship among labels. It shows the effectiveness of the label context, while the best performance of CMIML shows the further promotion caused by instance context.
3) The experiment results on Corel5k show that the performance of our method is also satisfactory with the larger number of labels. Although this dataset is challenging for

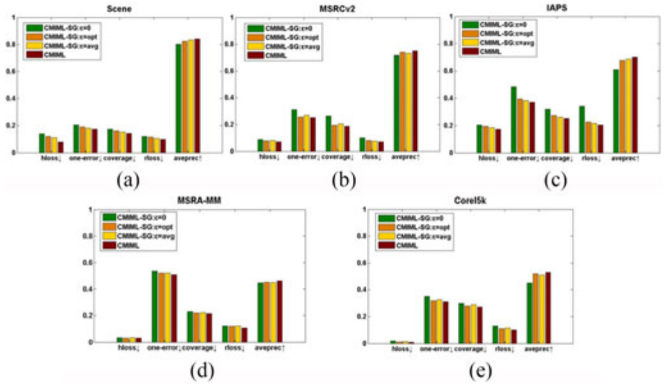


Fig. 4. Compare results on all datasets between different graph structures ("↓" indicates "the smaller the better;" "↑" indicates "the larger the better"). (a) Compare results of dataset Scene. (b) Compare results of dataset MSRC v2. (c) Compare results of dataset IAPS. (d) Compare results of dataset MSRA_MM. (e) Compare results of dataset Corel5k.

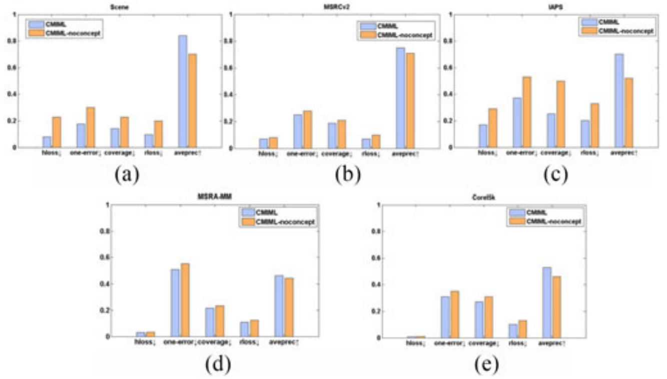


Fig. 5. Compare results on all datasets between CMIML and CMIML-noconcept ("↓" indicates "the smaller the better;" "↑" indicates "the larger the better"). (a) Compare result of dataset Scene. (b) Compare result of data set MSRC v2. (c) Compare result of dataset IAPS. (d) Compare result of dataset MSRA-MM. (e) Compare result of dataset Corel5k.

most MIML methods, some of which only obtain less than 20% on average precision, our proposed CMIML can still achieve more than 50% on average precision and max 0.3 decreases on other evaluations.

4) Our method outperforms all other methods on IAPS data set. It shows that affection is a type of high level semantics, the instance context and latent conception in CMIML can provide two middle semantic layers to link the lower level representation with high level affection semantics.

D. Further Analysis and Discussion

1) Evaluating Instance Context: To show the effect of instance context based on multiple ε-graphs, we compare our method with those methods taking single ε-graph as instance context in case of $\varepsilon = 0$, $\varepsilon = avg$ or $\varepsilon = opt$ on five criterions. Fig. 4 (a)–(e) show the results of five MIML datasets between CMIML with multiple ε-graphs and CMIML with single ε-graph, which are represented as CMIML and CMIML-SG respectively. The performance of CMIML, CMIML-SG: $\varepsilon = avg$ and CMIML-SG: $\varepsilon = opt$ are all better than that of CMIML-SG: $\varepsilon = 0$. This is due to the fact that $\varepsilon = 0$ is the extreme case in the graph structure and cannot bring in obvious performance improvement. If we compare CMIML-SG: $\varepsilon = avg$ and CMIML-SG: $\varepsilon = opt$, we find that no one is always better than the other one on all database in that each of them has its own advantage and limitation. Therefore the better way is to fuse multiple graphs which can produce best performance shown in Fig 4.

2) Evaluating Latent Conception Learning: If latent conception is not included, we learn the decision function through (14) by treating the labels in an independent manner and solve it through iteration optimization with $\mathbf{A}$ and δ. The process is the same as (22) and (24). The results of all datasets are shown in Fig. 5 (a)–(e) with legend named "CMIML-noconcept". The Fig. 5 shows the better performance of CMIML, indicating that label context is an important cue for MIML. The CMIML-noconcept learns a classifier for each label independently and neglects their correlation. However, in practical applications, the labels are impossibly independent. For example, the label "ship" nearly always associates with the label "water". The latent conception embedded in the proposed CMIML can model such kind of concurrence context well so as to improve the performance of annotation.

VII. Conclusion

Context cues existing in instances and labels have been shown to play an important role in MIML, especially in the application of image annotation. But most state-of-art approaches often ignore them. In this paper, we proposed a novel CMIML model that considers these two cues simultaneously. To construct the instance context cue, we build multiple ε-graphs in each bag so as to discover the underlying manifold structure of instances; to construct the label context cue, we assume that the labels can be combined linearly by a set of latent conceptions. Based on representer theory, these two context cues are integrated into a united framework. Through applying the CMIML to image annotation, experiments on benchmark data sets show the superiority of the proposed method over other MIML methods.

Appendix A
Proof of Lemma 1

Proof: Given a kernel κ and patterns $x_1, \ldots, x_m \in \aleph$, if κ is real-valued, then Definition 1 confirm that its kernel matrix K is symmetric. Given a mapping function $\phi : \aleph \rightarrow \mathbb{R}^{\aleph}$, for any arbitrary vector $z \in \mathbb{R}^m$, the following is always established:

$$\begin{aligned} z^{\mathrm{T}}\mathsf{K}z = & \sum_{i=1}^{m}\sum_{j=1}^{m} z_i K_{ij} z_j = \sum_{i=1}^{m}\sum_{j=1}^{m} z_i < \phi(x_i), \phi(x_j) > z_j \\ = & \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{l=1}^{\infty} z_i (\phi(x_i))_l (\phi(x_j))_l z_j \end{aligned}$$

$$= \sum_{l=1}^{\infty}\sum_{i=1}^{m} z_i(\phi(x_i))_l \sum_{j=1}^{m} z_j(\phi(x_i))_l$$

$$= \sum_{l=1}^{\infty}\left(\sum_{i=1}^{m} z_i(\phi(x_i))_l\right)^2 \geq 0.$$

So, K is positive semidefinite.

Appendix B
Proof of Lemma 2

Proof: $\kappa^{\theta}_{\text{graph}}, \theta = 1,2,3$ is a valid positive definite kernel [25]. *Lemma 1* tells us that its corresponding kernel matrix $\mathbf{K}^{\theta}$ is symmetric, positive semidefinite. Denoting $\mathbf{K}$ as the kernel matrix of κ_{graph}, then for any arbitrary vector $z \in \mathbb{R}^m$, $\mathbf{K} = \delta^1\mathbf{K}^1 + \delta^2\mathbf{K}^2 + \delta^3\mathbf{K}^3$ satisfies

$$z^{\mathrm{T}}\mathbf{K}z = z^{\mathrm{T}}\left(\sum_{\theta=1}^{3}\delta^{\theta}\mathbf{K}^{\theta}\right)z = \sum_{\theta=1}^{3}\delta^{\theta}z^{\mathrm{T}}\mathbf{K}^{\theta}z \geq 0$$

$$\text{s.t.} \sum_{\theta=1}^{3}\delta^{\theta} = 1, \delta^{\theta} \geq 0, \theta = 1,2,3.$$

So, $\mathbf{K}$ is positive semidefinite. Based on Mercer theorem [60]: κ is valid kernel if and only if its kernel matrix is positive semidefinite, we can get that κ_{graph} is a valid kernel. Furthermore, κ_{graph} is a positive kernel due to its positive semidefinite kernel matrix which can be derived from definition 1 to definition 3 in [30]. In addition, (4) and (3) confirms κ_{graph} is a real value. In conclusion, κ_{graph} is a valid positive definite real-valued kernel on $\wp \times \wp$.

Acknowledgment

The authors would like to thank the anonymous reviewers for their very valuable comments and suggestions, and also Dr. H. Zou and Dr. S. Feng for their help on the revision and experiments.

References

[1] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *Assoc. Comput. Mach. Comput. Surv.*, vol. 40, no. 5, pp. 1–60, 2008.

[2] U. L. Altintakan and A. Yazici "Towards effective image classification using class-specific codebooks and distinctive local features," *IEEE Trans. Multimedia*, vol. 17, no. 3, pp. 323–332, Mar. 2015.

[3] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2009, pp. 1794–1801.

[4] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2005, vol. 2, pp. 524–531.

[5] U. L. Altintakan and A. Yazici, "Towards effective image classification using class-specific codebooks and distinctive local features," *IEEE Trans. Multimedia*, vol. 17, no. 3, pp. 323–332, Mar. 2015.

[6] Z.-H. Zhou and M.-L. Zhang, "Multi-instance multi-label learning with application to scene classification," in *Advances in Neural Information Processing Systems 19*. Cambridge, MA, USA: MIT Press, 2007, pp. 1609–1616.

[7] Z. Zha *et al.*, "Joint multi-label multi-instance learning for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2008, pp. 1–8.

[8] Z.-H. Zhou, M.-L. Zhang, S.-J. Huang, and Y.-F. Li, "Multi-instance multi-label learning," *Artif. Intell.*, vol. 176, no. 1, pp. 2291–2320, 2012.

[9] J. Luo and F. Orabona, "Learning from candidate labeling sets," in *Advances in Neural Information Processing Systems 23*. Cambridge, MA, USA: MIT Press, 2010.

[10] N. Nguyen, "A new SVM approach to multi-instance multi-label learning," in *Proc. IEEE 10th Int. Conf. Data Mining*, Dec. 2010, pp. 384–392.

[11] C.-T. Nguyen, D.-C. Zhan, and Z.-H. Zhou, "Multi-modal image annotation with multi-instance multi-label LDA," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, 2013, pp. 1558–1564.

[12] Z. Gu, Tao Mei, X.-S. Hua, J. Tang, and X. Wu, *et al.*, "Multi-layer multi-instance learning for video concept detection," *IEEE Trans. Multimedia*, vol. 10, no. 8, pp. 1605–1616, Dec. 2008.

[13] X. Xu, X. Xue, and Z. Zhou, "Ensemble multi-instance multi-label learning approach for video annotation task," in *Proc. 19th ACM Int. Conf. Multimedia*, 2011, pp. 1153–1156.

[14] Y.-X. Li, S. Ji, S. Kumar, J. Ye, and Z.-H. Zhou, "Drosophila gene expression pattern annotation through multi-instance multi-label learning," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 9, no. 1, pp. 98–112, Jan./Feb. 2012.

[15] M. Surdeanu, J. Tibshirani, R. Nallapati, and C. D. Manning, "Multi-instance multi-label learning for relation extraction," in *Proc. Joint Conf. Empirical Methods Natural Language Process. Comput. Natural Language Learn.*, 2012, pp. 455–465.

[16] X. Xu and E. Frank, "Logistic regression and boosting for labeled bags of instances," in *Advacnces in Knowledge Discovery and Data Mining*, ser. Lecture Notes in Comput. Sci. 3056, Berlin, Germany: Springer, 2004, pp. 272–281.

[17] M. Boutell, J. Luo, X. Shen, and C. Brown, "Learning multi-label scene classification," *Pattern Recog.*, vol. 37, no. 9, pp. 1757–1771, Sep. 2004.

[18] M.-L. Zhang, "A k-nearest neighbor based multi-instance multi-label learning algorithm," in *Proc. IEEE 22nd Int. Conf. Tools Artif. Intell.*, Oct. 2010, vol. 2, pp. 207–212.

[19] F. Briggs, X. Fern, and R. Raich, "Rank-loss support instance machines for miml instance annotation," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 534–542.

[20] R. Jin, S. Wang, and Z.-H. Zhou, "Learning a distance metric from multi-instance multi-label data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2009, pp. 896–902.

[21] S.-J. Huang, W. Gao, and Z.-H. Zhou, "Fast multi-instance multi-label learning," in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 1868–1874.

[22] C.-T. Nguyen, N. Kaothanthong, X.-H. Phan, and T. Tokuyama, "A feature-word-topic model for image annotation," in *Proc. 19th ACM Int. Conf. Inf. Knowl. Manage.*, 2010, pp. 1481–1484.

[23] C.-T. Nguyen, D.-C. Zhan, and Z.-H. Zhou, "Multi-modal image annotation with multi-instance multi-label LDA," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, 2013, pp. 1558–1564.

[24] S. Yang, H. Zha, and B. Hu, "Dirichlet-Bernoulli alignment: A generative model for multi-class multi-label multi-instance corpora," in *Advances in Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 2009, pp. 2143–2150.

[25] Z. Zhou, Y. Sun, and Y. Li, "Multi-instance learning by treating instances as non-i.i.d. samples," in *Proc. Int. Conf. Mach. Learn.*, 2009, pp. 1249–1256.

[26] S. Ji, L. Tang, S. Yu, and J. Ye, "A shared-subspace learning framework for multi-label classification," *ACM Trans. Knowl. Discovery Data*, vol. 4, no. 2, pp. 8:1–8:29, 2010.

[27] H. Wang, H. Huang, and C. Ding, "Image annotation using bi-relational graph of imges and semantic labels," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 793–800.

[28] H. Wang, H. Huang, and C. Ding, "Image annotation using multi-label correlated Green's function," presented at the IEEE 12th Conf. Comput. Vis., Kyoto, Japan, 2009.

[29] J. Xu, V. Jagadeesh, and B. S. Manjunath, "Multi-label learning with fused multimodal bi-relational graph," *IEEE Trans. Multimedia*, vol. 16, no. 2, pp. 403–412, Feb. 2014.

[30] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," in *Computational Learning Theory*, ser. Lecture Notes in Comput. Sci. 2111, Berlin, Germany: Springer, 2001, pp. 416–426.

[31] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, pp. 2319–2323, 2000.

[32] G. Wahba, "Support vector machines, reproducing kernel Hilbert spaces and the randomized GACV," in *Advances in Kernel Methods-*

Support Vector Learning. Cambridge, MA, USA: MIT Press, 1999, pp. 69–88.

[33] Y. Huang, W. Wang, and L. Wang, “Unconstrained multimodal multi-label learning,” *IEEE Trans. Multimedia*, vol. 17, no. 11, pp. 1923–1935, Nov. 2015.

[34] M. Schmidt, G. Fung, and R. Rosales, “Fast optimization methods for l1 regularization: A comparative study and two new approaches,” in *Proc. 18th Eur. Conf. Mach. Learn.*, Sep. 2007, pp. 286–297.

[35] E. Gafni and D. Bertsekas, “Two-metric projection methods for constrained optimization,” *SIAM J. Control Optim.*, vol. 22, pp. 936–964, 1984.

[36] A. M. Turing, “Rounding-off errors in matrix processes,” *Quart. J. Mech. Appl. Math.*, vol. 1, no. 1, pp. 287–308, 1948.

[37] Y.-F. Li, J.-H. Hu, Y. Jiang, and Z.-H. Zhou, “Towards discovering what patterns trigger what labels,” in *Proc. 26th AAAI Conf. Artif. Intell.*, 2012, pp. 1012–1018.

[38] R. E. Schapire and Y. Singer, “BoosTexter: A boosting-based system for text categorization,” *Mach. Learn.*, vol. 39, nos. 2/3, pp. 135–168, 2000.

[39] Z.-H. Zhou and M.-L. Zhang, “Multi-instance multi-label learning with application to scene classification,” in *Advances in Neural Information Processing Systems 19*. Cambridge, MA, USA: MIT Press, 2007, pp. 1609–1616.

[40] O. Maron and A. L. Ratan, “Multiple-instance learning for natural scene classification,” in *Proc. 15th Int. Conf. Mach. Learn.*, 1998, pp. 341–349.

[41] J. Winn, A. Criminisi, and T. Minka, “Object categorization by learned universal visual dictionary,” in *IEEE 10th Int. Conf. Comput. Vis.*, Oct. 2005, vol. 2, pp. 1800–1807.

[42] P. Duygulu, K. Barnard, J. Freitas, and D. Forsyth, “Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary,” in *Proc. 7th Eur. Conf. Comput. Vis.*, 2002, pp. 97–112.

[43] Y. Chen and J. Z. Wang, “Image categorization by learning and reasoning with regions,” *J. Mach. Learn. Res.*, vol. 5, pp. 913–939, 2004.

[44] H. Li, M. Wang, and X.-S. Hua, “MSRA-MM 2.0: A large-scale web multimedia dataset,” in *Proc. IEEE Int. Conf. Data Mining Workshops*, Dec. 2009, pp. 164–169.

[45] P. J. Lang, M. M. Bradley, and B. N. Cuthbert, “International affective picture system (IAPS): Technical manual and affective ratings,” Centre Res. Psychophysiol., Gainesville, FL, USA, Tech. Rep., 1999. [Online]. Available: http://www2.unifesp.br/dpsicobio/Nova_versao_pagina_psicobio/adap/instructions.pdf

[46] J. A. Mikels *et al.*, “Emotional category data on images from the international affective picture system,” *Behavior Res. Methods*, vol. 37, pp. 626–630, 2005.

[47] J. Machajdik and A. Hanbury, “Affective image classification using features inspired by psychology and art theory,” in *Proc. Int. Conf. Multimedia*, 2010, pp. 83–92.

[48] T. Gartner, P. A. Flach, A. Kowalczyk, and A. J. Smola, “Multi-instance kernels,” in *Proc. Int. Conf. Mach. Learn.*, 2002, pp. 179–186.

[49] K. Barnard *et al.*, “Matching words and pictures,” *J. Mach. Learn. Res.*, vol. 3, pp. 1107–1135, 2003.

[50] C. Wang, D. M. Blei, and F.-F. Li, “Simultaneous image classification and annotation,” in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, Jun. 2009, pp. 1903–1910,.

[51] W. Liu, H. Liu, D. Tao, Y. Wang, and K. Lu, *et al.*, “Manifold regularized kernel logistic regression for web image annotation,” *Neurocomputing*, vol. 172, no. C, pp. 3–8, 2016.

[52] M. Boutell, J. Luo, X. Shen, and C. Brown, “Learning multi-label scene classification,” *Pattern Recog.*, vol. 37, no. 9, pp. 1757–1771, 2004.

[53] D. Tao, L. Jin, W. Liu, and X. Li, “Hessian regularized support vector machines for mobile image annotation on the cloud,” *IEEE Trans. Multimedia*, vol. 15, no. 4, pp. 833–844, Jun. 2013.

[54] W. Liu and D. Tao, “Multiview Hessian regularization for image annotation,” *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2676–2687, Jul. 2013.

[55] W. Liu, D. Tao, J. Cheng, and Y. Tang, “Multiview Hessian discriminative sparse coding for image annotation,” *Comput. Vis. Image Understanding*, vol. 118, pp. 50–60, 2014.

[56] H. Wang, H. Huang, and C. Ding, “Image annotation using bi-relational graph of imges and semantic labels,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 793–800.

[57] H. Wang, H. Huang, and C. Ding, “Image annotation using multi-label correlated green’s function,” in *Proc. IEEE Conf. Comput. Vis.*, Sep.-Oct. 2009, pp. 1–6.

[58] C. Berg, J. P. R. Christensen, and P. Ressel, *Harmonic Analysis on Semigroups*. New York, NY, USA: Springer-Verlag, 1984.

[59] M. Gönen and E. Alpaydın, “Multiple kernel learning algorithms,” *J. Mach. Learn. Res.*, vol. 12, pp. 2211–2268, 2011.

[60] M. Hazewinkel, Ed., “Mercer theorem,” in *Encyclopedia of Mathematics*. Berlin, Germany: Springer, 2001.

[61] P. Wolfe, “Methods of nonlinear programming,” in *Nonlinear Programming*, J. Abadie, Ed. Amsterdam, The Netherlands: North Holland, 1967, pp. 97–131.

[62] J. Shi and J. Malik, “Normalised cuts and image segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 1997, pp. 731–737.

[63] B. Li, W. Xiong, O. Wu, and W. Hu, “Horror image recognition based on context-aware multi-instance learning,” *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5193–5025, Dec. 2015.

[64] W. Hu *et al.*, “Multi-perspective cost-sensitive context-aware multi-instance sparse coding and its application to sensitive video recognition,” *IEEE Trans. Multimedia*, vol. 18, no. 1, pp. 76–89, Jan. 2016.

[65] X. Ding, B. Li, X. Xiong, and W. Hu, “Horror video scene recognition based on multi-view multi-instance learning,” in *Proc. 11th Asian Conf. Comput. Vis.*, 2012, pp. 599–610.

[66] A. Kumar and Hal Daum´e, III, “Learning task grouping and overlap in multi-task learning,” in *Proc. 29th Int. Conf. Mach. Learn.*, 2012, pp. 1383–1390.

[67] B. Li, W. Xiong, W. Hu, and B. Funt, “Multi-cue illumination estimation via a tree-structured group joint sparse representation,” *Int. J. Comput. Vis.*, vol. 117, no. 1, pp. 21–47, 2016.

Xinmiao Ding received the Ph.D. degree in mechanical, electronic, and information engineering from the China University of Mining and Technology, Beijing, China, in 2013.

From March 2015 to March 2016, she was a Visiting Scholar with the Institute of Automation, Chinese Academy of Sciences, Beijing, China. Her research interests include image and video analysis and understanding, machine learning, and internet security.

Bing Li received the Ph.D. degree in computer science and engineering from Beijing Jiaotong University, Beijing, China, in 2009.

He is currently an Associate Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include color constancy, visual saliency, and web content mining.

Weihua Xiong received the Ph.D. degree in computer science from Simon Fraser University, Burnaby, BC, Canada, in 2007.

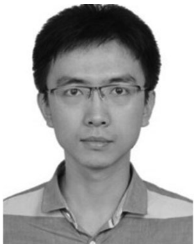
His research interests include color science, computer vision, color image processing, and stereo vision.

Wen Guo received the B.E. degree from Central South University, Changsha, China, in 2001, the M.S. degree from Shandong University, Jinan, China, in 2007, and the Ph.D. degree from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2012.

He is currently an Associate Professor with the Shandong Institute of Business and Technology, Yantai, China. His research interests include computer vision, multimedia, machine learning, and pattern recognition.

Weiming Hu received the Ph.D. degree in computer science and engineering from Zhejiang University, Hangzhou, China, in 1998.

From April 1998 to March 2000, he was a Postdoctoral Research Fellow with the Institute of Computer Science and Technology, Peking University, Beijing, China. He is currently a Professor with the Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include visual motion analysis, recognition of web objectionable information, and network intrusion detection.



Bo Wang received the M.S. degree from the College of Computer and Information Technology, Northeast Petroleum University, Daqing, China, in 2013.

He is currently an Intermediate Engineer with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include image retrieval and face recognition.