Improving Web Image Search by Bag-Based Reranking

Lixin Duan, Wen Li, Ivor Wai-Hung Tsang, and Dong Xu, *Member, IEEE*

***Abstract*—Given a textual query in traditional text-based image retrieval (TBIR), relevant images are to be reranked using visual features after the initial text-based search. In this paper, we propose a new bag-based reranking framework for large-scale TBIR. Specifically, we first cluster relevant images using both textual and visual features. By treating each cluster as a "bag" and the images in the bag as "instances," we formulate this problem as a multi-instance (MI) learning problem. MI learning methods such as mi-SVM can be readily incorporated into our bag-based reranking framework. Observing that at least a certain portion of a positive bag is of positive instances while a negative bag might also contain positive instances, we further use a more suitable generalized MI (GMI) setting for this application. To address the ambiguities on the instance labels in the positive and negative bags under this GMI setting, we develop a new method referred to as GMI-SVM to enhance retrieval performance by propagating the labels from the bag level to the instance level. To acquire bag annotations for (G)MI learning, we propose a bag ranking method to rank all the bags according to the defined bag ranking score. The top ranked bags are used as pseudopositive training bags, while pseudonegative training bags can be obtained by randomly sampling a few irrelevant images that are not associated with the textual query. Comprehensive experiments on the challenging real-world data set NUS-WIDE demonstrate our framework with automatic bag annotation can achieve the best performances compared with existing image reranking methods. Our experiments also demonstrate that GMI-SVM can achieve better performances when using the manually labeled training bags obtained from relevance feedback.**

***Index Terms*—Bag-based image reranking, generalized multi-instance (GMI) learning, text-based image retrieval (TBIR).**

I. Introduction

WITH THE ever-growing number of images on the *Internet* (such as in the online photo sharing Website `Flickr.com`, the online photo forum `photoSIG.com`, and so on), retrieving relevant images from a large collection of database images has become an important research topic. Over the past decades, many image retrieval systems have been developed, such as text-based image retrieval (TBIR) [3], [12], [19], [38], [42] and content-based image retrieval [23], [33], [39].

Manuscript received July 09, 2010; revised December 22, 2010, February 16, 2011, and April 06, 2011; accepted May 02, 2011 Date of publication June 09, 2011; date of current version October 19, 2011. This work was supported in part by the Singapore National Research Foundation and Interactive Digital Media R&D Program Office, MDA, under Research Grant NRF2008IDM-IDM004-018 and by Microsoft Research Asia. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Sharath Pankanti.

The authors are with the School of Computer Engineering, Nanyang Technological University, Singapore 639798.

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/TIP.2011.2159227

southwest plane airplane plane bwi marylandone nikographer nikographerjon

city holland building maastricht light town hall glow cityhall nederland townhall stadhuis

Fig. 1. Web images with noisy tags.

As shown in Fig. 1, Web images (e.g., images downloaded from `Flickr.com`) are usually associated with rich semantic textual descriptions (also called surrounding texts or tags). By exploiting such rich semantic textual descriptions of Web images, the TBIR has been widely used in popular image search engines (e.g., `Google`, `Bing`, and `Yahoo!`). Specifically, a user is required to input a keyword as a textual query to the retrieval system. Then, the system returns the ranked relevant images whose surrounding texts contain the query keyword, and the ranking score is obtained according to some similarity measurements (such as cosine distance) between the query keyword and the textual features of relevant images. However, the retrieval performance can be very poor, particularly when the textual features of the Web images are sparse and noisy in a high-dimensional space.

To solve this problem, many image reranking methods have been developed [5], [12]–[14], [31], [32], [36], [42] to rerank the initially retrieved images using visual features. Zhou and Dai [42] proposed a method called Web search exploiting image contents (WEBSEIC), which uses kernel density estimation (KDE) based on visual features to rerank the retrieved relevant images. After that, an image-based ranking of Web pages is generated, and the final search result is obtained by combining with the original text-based search result. Hsu *et al.* [12] presented a reranking method via the information bottleneck principle based on mutual information. In their work, they first clustered the initially retrieved images together with some irrelevant images by using a so-called sequential information bottleneck clustering method [26]. Then, a cluster probability is obtained for cluster ranking. Finally, KDE based on visual features is used to rerank the relevant images within each cluster. Several graph-based reranking methods [13], [14], [32], [36] have been also developed. The basic idea is to construct a graph representing the local similarity of visual features of images for reranking. However, the similarity of low-level visual features among the unconstrained Web images may not reflect the high-level semantic concepts of Web images due to the semantic gap. Moreover, this reranking paradigm does not consider label information and can only achieve limited improvements. To address this issue,

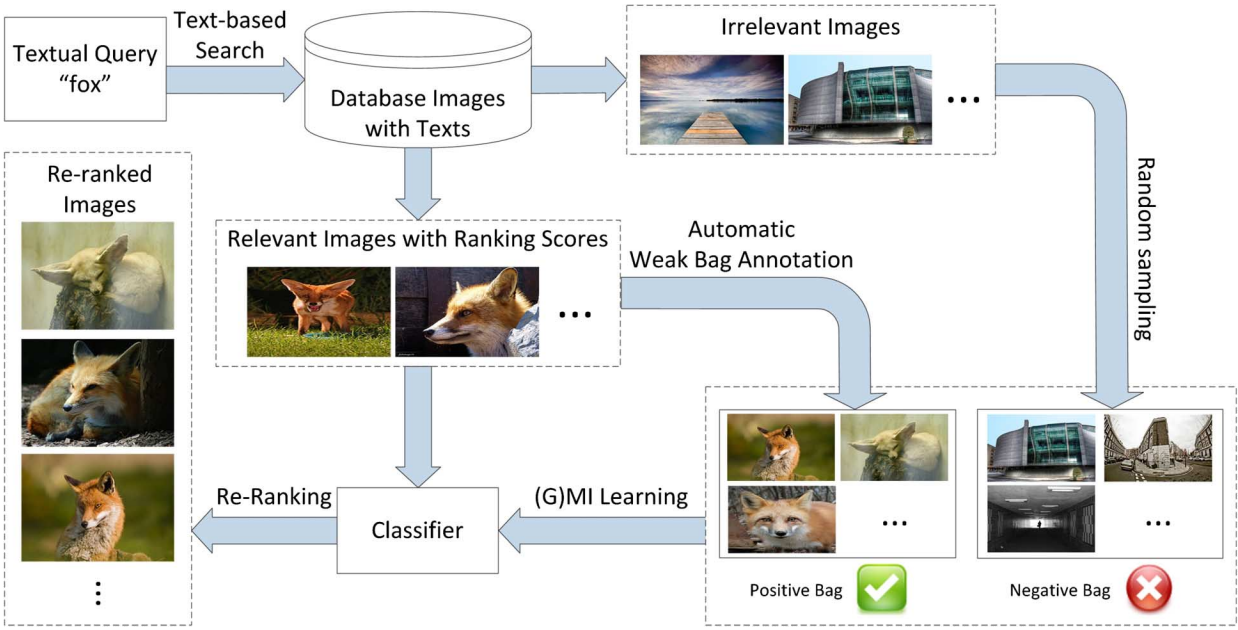


Fig. 2. Bag-based image reranking framework for large-scale TBIR.

relevance feedback (RF) methods [5], [31] have been proposed to acquire the search intentions of the user for further improving the retrieval performance. Aside from these, Zhang *et al.* [38] investigated a so-called user term feedback method to refine the retrieved images. However, they mentioned that the term feedback was not effective in the TBIR. For more comprehensive reviews of image retrieval, interested readers can refer to two surveys in [6] and [27].

To improve the retrieval performance, in this paper, we introduce a new framework, referred to as the bag-based image reranking framework, for large-scale TBIR. We first partition the relevant images into clusters by using visual and textual features. Inspired by multi-instance (MI) learning methods [1], [7], [20], [25], [40], we treat each cluster of images as a "bag" and the images inside the cluster as "instances." Then, existing MI learning methods (e.g., mi-SVM [1]) can be readily adopted in our framework.

In traditional MI learning methods, if a bag contains at least one relevant instance, this bag is labeled as positive; if the instances in a bag are all irrelevant, this bag is labeled as negative. In our image retrieval application, we observe that it is very likely that multiple relevant images are clustered in a positive bag while a few relevant images may be clustered with irrelevant images in a negative bag. Different from traditional MI learning, we propose a generalized MI (GMI) setting for this application in which at least a certain portion of a positive bag is of positive instances, while a negative bag might contain at most a few positive instances. In this case, the traditional MI methods may not be effective to address the ambiguities on the instance labels in both positive and negative bags. Therefore, we propose a new GMI learning algorithm using SVM, referred to GMI-SVM, which uses the recently proposed "Label Generation" strategy [18] and maximum margin criterion to effectively rerank the relevant images by propagating the labels from the bag level to the instance level.

To facilitate (G)MI learning in our framework, we conduct a so-called *weak bag annotation* process to automatically find positive and negative bags for training classifiers. First, we introduce an *instance ranking score* defined by the similarity between the textual query and each relevant image. Then, we obtain a *bag ranking score* for each bag by averaging the instance ranking scores of the instances in this bag. Finally, we rank all bags with the bag ranking score. In our automatic bag annotation method, the top ranked bags are used as the pseudopositive bags, and pseudonegative bags are obtained by randomly sampling a few irrelevant images that are not associated with the textual query. After that, these bags are used to train a classifier that is then used to rerank the database images. Fig. 2 shows the overall flowchart of our proposed bag-based framework for the TBIR. We will show in the experiments that our framework with the automatic bag annotation method performs much better than the existing image reranking methods [12], [42]. Moreover, users are also allowed to manually annotate positive/negative bags during the RF process, and our experiments show that the retrieval performance of GMI-SVM can be further improved by using the manually labeled training bags.

We summarize the main contributions of this paper.

- We present a novel bag-based framework that enables us to formulate the image reranking problem as an MI learning problem and improve TBIR performance by using MI learning methods.
- We further reformulate our problem as a GMI learning problem that relaxes the constraints in the traditional MI learning problem. To address the ambiguities on the instance labels in both positive and negative bags, we propose GMI-SVM, which outperforms other traditional MI learning methods for image retrieval.
- We develop an automatic weak bag annotation method to effectively find positive and negative bags for (G)MI learning methods.

II. Related Work on MI Learning

MI learning methods have been proposed to solve learning problems with ambiguity on training samples. In the traditional supervised learning problems, there is clear knowledge on the labels of training samples. In contrast, in MI learning problems, a label only accompanies each training "bag," which consists of several instances (i.e., training samples). Specifically, in the traditional setting of MI learning problems, each positive bag has at least one positive instance, while a negative bag has no positive instances. MI learning methods [1], [7], [20], [35], [40] learn models from the training data with such ambiguous label information and predict the label of test bags or instances.

Diverse density (DD) [20] finds the concept point that is near at least one instance in the positive bags and far from all instances in the negative bags. EM-DD [40], i.e., the EM variation of the DD, iteratively guesses the positive instances in each positive bag and refines the hypothesis of the concept. Citation kNN [35] predicts the label of a bag based on its nearest neighboring bags (referred to as "references") and the bags that count it as one of nearest neighbors (referred to as "citers"). However, all these methods have a high computational cost, which makes them unsuitable for large-scale systems.

Andrews *et al.* [1] proposed two variants of SVM, i.e., mi-SVM and MI-SVM, to solve MI learning problems. The mi-SVM maximizes the instance margin jointly over possible label assignments, as well as hyperplanes, while MI-SVM maximizes the bag margin. Although these two methods are implemented with mixed integer programming, their speeds are much faster than the previous methods.

MI learning methods have also been used in region-based image retrieval [21], [29], [30], [41] and locating image regions of interest [17]. In these applications, images are considered as bags, whereas regions in the images are considered as instances. Since images from the same concept usually have similar regions, these regions can be considered as the positive instances in the positive bags, and thus this problem can be formulated as an MI learning problem. However, these region-based image retrieval methods are too computationally expensive for large-scale image databases, such as the NUS-WIDE database used in this paper. Note that the work in [34] used a sparse MI learning method called sMIL [2] and its variant called weighted sMIL for bag-based learning. However, they assume that the bags are constructed by using image search engines in multiple languages, which restricts its applicability and cannot be directly used in our setting.

In this paper, we propose a new bag-based reranking framework for large-scale TBIR by treating one image cluster as one "bag" and the images in a bag as "instances." In our setting, each bag (cluster) can have a rough estimate of the proportion of positive instances (images). For example, the positive bags consist of *at least* $\mu = 10\%$ positive instances, whereas the negative bags have *at most* $\gamma = 2\%$ positive instances. Note that our new assumption is different from the conventional MI assumption in two aspects: 1) it removes the strict assertion of the negative bags and 2) it provides more information for positive bags. To address the ambiguities on the instance labels in both positive and negative bags, we then generalize the MI learning problem under the new setting and develop a GMI-SVM algorithm for label prediction on instances (images) to enhance the retrieval performance.

III. Bag-Based Web Image Reranking Framework

Here, we present our proposed bag-based reranking framework for large-scale TBIR. Our goal is to improve the Web image retrieval in Internet image databases, such as `Flickr`. These Web images are usually accompanied by textual descriptions. For the ith Web image, the low-level visual feature $\mathbf{v}_i$ (e.g., color, texture, and shape) and the textual feature $\mathbf{t}_i$ (e.g., term frequency) can be extracted. We further aggregate them into a single feature vector $\mathbf{x}_i$ for subsequent operations, namely, $\mathbf{x}_i = [\lambda \mathbf{v}_i', \mathbf{t}_i']'$, where λ is a weight parameter.

A. Initial Ranking

After the user provides a textual query q (e.g., "fox"), our system exploits the inverted-file method [19] to automatically find relevant Web images whose surrounding text contains the textual query tag q, as well as irrelevant Web images whose surrounding text do not contain q. For each retrieved relevant image $\mathbf{x}$, an instance ranking score can be defined as follows [3]:

$$r(\mathbf{x}) = -\tau + \frac{1}{\delta} \tag{1}$$

where δ is the total number of tags in image $\mathbf{x}$ and τ is the rank position of the query tag q in the tag list of image $\mathbf{x}$. If $\tau_i < \tau_j$ and $i \neq j$, then we have $r(\mathbf{x}_i) > r(\mathbf{x}_j)$. In other words, when one relevant image contains the textual query q at the top position in its tag list, this image will be assigned a higher ranking score. When the positions of the query tag q are the same for the two images (i.e., $\tau_i = \tau_j$), the ranking score is decided by δ_i and δ_j, namely, the image that has fewer tags is preferred.

B. Weak Bag Annotation Process

In our framework, each image is considered as an "instance." To construct "bags," we partition the relevant images into clusters using the k-means clustering method based on visual and textual features. After that, each cluster is considered as a "bag." To facilitate (G)MI learning methods in our framework, we have to annotate positive and negative bags to train classifiers. Note that only the bags are to be annotated, while the labels of instances in each bag are still ambiguous. Therefore, we refer to the annotation of a bag as *weak bag annotation*.

Specifically, for each bag $\mathcal{B}_I$, its bag ranking score $S(\mathcal{B}_I)$ is defined as the average instance ranking score, i.e.,

$$S(\mathcal{B}_I) = \frac{\sum_{\mathbf{x} \in \mathcal{B}_I} r(\mathbf{x})}{|\mathcal{B}_I|} \tag{2}$$

where $|\mathcal{B}_I|$ stands for the cardinality of bag $\mathcal{B}_I$.

In our automatic bag annotation method, the top-ranked bags with higher bag ranking scores are used as pseudopositive bags, and the same number of pseudonegative bags is obtained by randomly sampling a few irrelevant images. We will show in the experiments that our GMI learning method GMI-SVM with this

simple bag annotation method can achieve better retrieval performances when compared with those in [12] and [42]. Note that our proposed automatic weak bag annotation method is similar to the pseudo-RF algorithm proposed in [37], which can annotate instances, whereas our approach can annotate high-confident bags, as demonstrated in Section IV-B.

C. GMI Learning

We denote the transpose of a vector/matrix by superscript $'$. We also define $\mathbf{I}$ as the identity matrix and $\mathbf{0}$ and $\mathbf{1} \in \mathbb{R}^n$ as the zero vector and the vector of all 1's, respectively. Moreover, the element-wise product between matrices $\mathbf{P}$ and $\mathbf{Q}$ is represented as $\mathbf{P} \odot \mathbf{Q}$. Inequality $\mathbf{u} = [u_1, u_2, \ldots, u_n]' \geq \mathbf{0}$ means that $u_i \geq 0$ for $i = 1, \ldots, n$. A positive or negative bag $\mathcal{B}_I$ is associated with a bag label $Y_I \in \{\pm 1\}$. We also denote the unobserved instance label of $\mathbf{x}_i$ as $y_i \in \{\pm 1\}$. With this definition of bags, we can define the GMI constraint on the instance labels of positive and negative bags, respectively, as

$$\begin{aligned} \sum_{i:\mathbf{x}_i \in \mathcal{B}_I} \frac{y_i + 1}{2} &\geq \mu |\mathcal{B}_I|, \qquad \text{for } Y_I = 1 \\ \sum_{i:\mathbf{x}_i \in \mathcal{B}_I} \frac{y_i + 1}{2} &\leq \gamma |\mathcal{B}_I|, \qquad \text{for } Y_I = -1. \end{aligned} \tag{3}$$

In other words, positive instances take up at least portion μ of a positive bag, whereas positive instances occupy at most portion γ of a negative bag. Note that traditional MI learning [1], [40] is actually a special case of GMI learning with $\mu = 1/|\mathcal{B}_I|$ and $\gamma = 0$. In contrast to the restrictive MI assumption in [1] and [40], the GMI constraint in (3) is more suitable to this application.

We further denote $\mathbf{y} = [y_1, \ldots, y_n]'$ as the vector of instance labels and $\mathcal{Y} = \{\mathbf{y} | y_i \in \{\pm 1\}, \text{and } \mathbf{y} \text{ satisfies (3)}\}$ as the domain of $\mathbf{y}$. Then, the decision function f of the GMI learning can be learned by minimizing the following structural risk functional:

$$\min_{\mathbf{y} \in \mathcal{Y}, f} \Omega(\|f\|) + C \sum_{i=1}^{n} \ell(-y_i f(\mathbf{x}_i)) \tag{4}$$

where $\Omega(\|f\|)$ is the regularization term, $\ell(\cdot)$ is a loss function for each instance, and C is the parameter that trades off the complexity and the fitness of the decision function f. Note that the constraints in (3) are integer constraints; thus, the corresponding GMI problem (4) is usually formulated as a mixed integer programming problem.

Discussion: We note that Scott *et al.* addressed "GMI learning" in [25], as well as in their subsequent work [28]–[30]. However, their algorithms, named "GMIL-1" and "GMIL-2," are intrinsically different from ours. In their MI assumption, the label of a bag is represented by a threshold function rather than as a binary label (i.e., $y \in \{\pm 1\}$), which is used in conventional MI learning methods and this paper. Moreover, their work can only predict the label of a bag rather than that of an instance. Although their methods can achieve the state-of-the-art performance in bag prediction, how to predict the labels of image instances is unclear. Therefore, their work is unsuitable for reranking the relevant images in our image retrieval application.

D. GMI-SVMs

In this paper, we assume the decision function is in form of $f(\mathbf{x}) = \mathbf{w}'\varphi(\mathbf{x}) + b$ and the regularization term is $(1/2)\|\mathbf{w}\|^2$. We adopt the formulation of the Lagrangian SVM, in which the square bias penalty b^2 and the square hinge loss for each instance are used in the objective function. The GMI optimization problem can be written as the following constrained optimization problem:

$$\begin{aligned} \min_{\mathbf{y} \in \mathcal{Y}, \mathbf{w}, b, \rho, \xi_i} \quad & \frac{1}{2}\left(\|\mathbf{w}\|^2 + b^2 + C\sum_{i=1}^{n} \xi_i^2\right) - \rho \\ \text{s.t.} \quad & y_i\left(\mathbf{w}'\phi(\mathbf{x}_i) + b\right) \geq \rho - \xi_i, i = 1, \ldots, n. \end{aligned} \tag{5}$$

where ξ_i values are slack variables and $\rho/\|\mathbf{w}\|$ defines the margin separation. By introducing a dual variable α_i for each inequality constraint in (5) and the kernel trick (i.e., $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)$), we arrive at the following minimax saddle-point problem:

$$\min_{\mathbf{y} \in \mathcal{Y}} \max_{\boldsymbol{\alpha} \in \mathcal{A}} -\frac{1}{2}\boldsymbol{\alpha}'\left(\widetilde{\mathbf{K}} \odot \mathbf{y}\mathbf{y}' + \frac{1}{C}\mathbf{I}\right)\boldsymbol{\alpha} \tag{6}$$

where $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]'$ is the vector of the dual variables and $\mathcal{A} = \{\boldsymbol{\alpha} | \boldsymbol{\alpha} \geq \mathbf{0},\ \boldsymbol{\alpha}'\mathbf{1} = 1\}$ is the domain of $\boldsymbol{\alpha}$. We also define $\mathbf{K} = [k(\mathbf{x}_i,\ \mathbf{x}_j)]$ as an $n \times n$ kernel matrix and $\widetilde{\mathbf{K}} = \mathbf{K} + \mathbf{1}\mathbf{1}'$ as an $n \times n$ transformed kernel matrix for the augmented feature mapping $\widetilde{\phi}(\mathbf{x}) = [\phi(\mathbf{x})',\ 1]'$ of kernel $\widetilde{k}(\mathbf{x}_i,\ \mathbf{x}_j) = \widetilde{\phi}(\mathbf{x}_i)'\widetilde{\phi}(\mathbf{x}_j)$. Note that the instance labels y_i in (6) are also integer variables, and thus, (6) is a mixed integer programming problem, which is computationally intractable in general.

Recently, Li *et al.* [18] proposed an efficient convex optimization method to solve the mixed integer programming problem for maximum margin clustering. In this paper, we extend their algorithm [18] to solve the mixed integer programming problem in (6). Our proposed method is then referred to as the GMI-SVM.

1) Convex Relaxation: First, let us consider interchanging the order of $\min_{\mathbf{y} \in \mathcal{Y}}$ and $\max_{\boldsymbol{\alpha} \in \mathcal{A}}$ in (6). Then, we have

$$\max_{\boldsymbol{\alpha} \in \mathcal{A}} \min_{\mathbf{y} \in \mathcal{Y}} -\frac{1}{2}\boldsymbol{\alpha}'\left(\widetilde{\mathbf{K}} \odot \mathbf{y}\mathbf{y}' + \frac{1}{C}\mathbf{I}\right)\boldsymbol{\alpha}. \tag{7}$$

According to the minimax theorem [16], the optimal objective of (6) is an upper bound of that of (7). By introducing θ, we can further rewrite (7) as follows:

$$\max_{\boldsymbol{\alpha} \in \mathcal{A}} \left\{ \max_{\theta} -\theta : \theta \geq \frac{1}{2}\boldsymbol{\alpha}'\left(\widetilde{\mathbf{K}} \odot \mathbf{y}^t\mathbf{y}^{t'} + \frac{1}{C}\mathbf{I}\right)\boldsymbol{\alpha},\ \forall \mathbf{y}^t \in \mathcal{Y} \right\} \tag{8}$$

where $\mathbf{y}^t$ is any feasible solution in $\mathcal{Y}$. For the inner optimization subproblem of (8), we can obtain its Lagrangian L as follows by introducing a dual variable $d_t \geq 0$ for each constraint:

$$L = -\theta + \sum_{t:\mathbf{y}^t \in \mathcal{Y}} d_t \left(\theta - \frac{1}{2}\boldsymbol{\alpha}'\left(\widetilde{\mathbf{K}} \odot \mathbf{y}^t\mathbf{y}^{t'} + \frac{1}{C}\mathbf{I}\right)\boldsymbol{\alpha}\right). \tag{9}$$

Setting the derivative of Lagrangian (9) with respect to θ to zero, we have $\sum_{t:\mathbf{y}^t\in\mathcal{Y}} d_t = 1$. Denote $\mathbf{d}$ as a vector of d_t values and $\mathcal{M} = \{\mathbf{d}|\mathbf{d} \geq \mathbf{0}, \mathbf{d}'\mathbf{1} = 1\}$ as the domain of $\mathbf{d}$. We can then arrive at its dual form as follows:

$$\min_{\mathbf{d}\in\mathcal{M}} -\frac{1}{2}\boldsymbol{\alpha}' \left(\sum_{t:\mathbf{y}^t\in\mathcal{Y}} d_t \widetilde{\mathbf{K}} \odot \mathbf{y}^t\mathbf{y}^{t'} + \frac{1}{C}\mathbf{I} \right) \boldsymbol{\alpha}. \quad (10)$$

Replacing the inner maximization subproblem in (8) with its dual (10), we have the following optimization problem:

$$\max_{\boldsymbol{\alpha}\in\mathcal{A}} \min_{\mathbf{d}\in\mathcal{M}} -\frac{1}{2}\boldsymbol{\alpha}' \left(\sum_{t:\mathbf{y}^t\in\mathcal{Y}} d_t \widetilde{\mathbf{K}} \odot \mathbf{y}^t\mathbf{y}^{t'} + \frac{1}{C}\mathbf{I} \right) \boldsymbol{\alpha}$$
$$= \min_{\mathbf{d}\in\mathcal{M}} \max_{\boldsymbol{\alpha}\in\mathcal{A}} -\frac{1}{2}\boldsymbol{\alpha}' \left(\sum_{t:\mathbf{y}^t\in\mathcal{Y}} d_t \widetilde{\mathbf{K}} \odot \mathbf{y}^t\mathbf{y}^{t'} + \frac{1}{C}\mathbf{I} \right) \boldsymbol{\alpha}. \quad (11)$$

The equality holds as the objective function is concave in $\boldsymbol{\alpha}$ and linear in $\mathbf{d}$, and thus, we can interchange the order of $\max_{\boldsymbol{\alpha}\in\mathcal{A}}$ and $\min_{\mathbf{d}\in\mathcal{M}}$ in (11). Observe that (11) is analogous to the multiple kernel learning (MKL) problem [22], except that a label-kernel matrix, which is a convex combination of the base label-kernel matrices $\widetilde{\mathbf{K}} \odot \mathbf{y}^t\mathbf{y}^{t'}$, is to be learned. Hence, (11) can be viewed as a multiple label-kernel learning (MLKL) problem.

2) Cutting-Plane Algorithm for GMI-SVM: Although $\mathcal{Y}$ is finite and the MLKL problem (11) is a special case of MKL, there are $O(2^n)$ candidates of the label vector $\mathbf{y}^t$, and thus, the number of base label-kernel matrices $\widetilde{\mathbf{K}} \odot \mathbf{y}^t\mathbf{y}^{t'}$ is exponential in size. Thus, it is not possible to directly apply recently proposed MKL techniques such as SimpleMKL [22] to our proposed GMI-SVM.

Algorithm 1: Cutting-plane algorithm for GMI-SVM.

1:Initialize $y_i = Y_I$ for $i \in \mathcal{B}_I$ as $\mathbf{y}^1$, and set $\mathcal{C} = \{\mathbf{y}^1\}$;

2:Compute MKL to solve $\boldsymbol{\alpha}$ and $\mathbf{d}$ in (11) based on $\mathcal{C}$;

3:Use $\boldsymbol{\alpha}$ to select the most violated $\mathbf{y}^t$ and set $\mathcal{C} = \mathbf{y}^t \cup \mathcal{C}$;

4:Repeat lines 2 and 3 until convergence.

Fortunately, not all quadratic inequality constraints in (8) are necessarily active at optimality, and only subset $\mathcal{C} \subset \mathcal{Y}$ of these constraints can usually lead to a very good approximation of the original optimization problem. Therefore, we can apply the cutting-plane method [15] to handle this exponential number of constraints. Moreover, the same strategy has been also applied in the recently proposed infinite kernel learning (IKL) [9], [10], in which the kernel is learned from an infinite set of general kernel parameters, and thus, MLKL (with kernel $\sum_{t:\mathbf{y}^t\in\mathcal{Y}} d_t \widetilde{\mathbf{K}} \odot \mathbf{y}^t\mathbf{y}^{t'}$) can be deemed as a variant of IKL. As a result, our GMI-SVM enjoys the same convergence of IKL [9]. The whole algorithm is summarized in Algorithm 1. First, we set subset $\mathcal{C} = \{\mathbf{y}^1\}$, where the instance label vector $\mathbf{y}^1$ is initialized according to the bag labels. Since $\mathcal{C}$ is no longer exponential in size, one can apply MKL to learn the label kernel to obtain both $\boldsymbol{\alpha}$ and $\mathbf{d}$. With a fixed $\boldsymbol{\alpha}$, the label vector $\mathbf{y}^t$ with a quadratic inequality constraint in (8), which is the most violated one by the current solution, is then added to $\mathcal{C}$. The process is repeated until the convergence criterion (i.e., the relative change of the objective values of (11) between two successive iterations is less than 0.01) is met. After solving the MLKL problem, the decision function can be obtained by

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i \tilde{y}_i \tilde{k}(\mathbf{x}, \mathbf{x}_i)$$

where $\tilde{y}_i = \sum_{t:\mathbf{y}^t\in\mathcal{C}} d_t y_i^t$ and $\tilde{k}(\mathbf{x}, \mathbf{x}_i) = k(\mathbf{x}, \mathbf{x}_i) + 1$.

Algorithm 2: Finding the approximation of the most violated $\mathbf{y}^t$.

1:Initialize $y_i = 1$ for all $\mathbf{x}_i$ in positive bags $\mathcal{B}_P$ and $y_i = -1$ for all $\mathbf{x}_i$ in negative bags $\mathcal{B}_N$;

2:**for** each positive bag $\mathcal{B}_P$ **do**

3:Fix the labeling of instances in all the other bags, and find the optimal $\mathbf{y}_P$ that maximizes the objective of (12) by enumerating the candidates of y_i in $\mathcal{B}_P$;

4:**end for**

5:**for** each negative bag $\mathcal{B}_N$ **do**

6: Fix the labeling of instances in all the other bags, and find the optimal $\mathbf{y}_N$ that maximizes the objective of (12) by enumerating the candidates of y_i in $\mathcal{B}_N$;

7:**end for**

8:Repeat lines 2–7 until convergence.

3) Finding the Approximation of the Most Violated $\mathbf{y}^t$: Similar to IKL, finding the most violated constraint (indexed by $\mathbf{y}^t$) in MLKL is problem specific and is the most challenging part in cutting-plane algorithms. Here, we discuss how to search for the most violated constraint to satisfy the GMI constraints in (3).

Referring to (8), to find the most violated $\mathbf{y}^t$, we have to solve the following problem:

$$\max_{\mathbf{y}\in\mathcal{Y}} \boldsymbol{\alpha}'(\widetilde{\mathbf{K}} \odot \mathbf{y}\mathbf{y}')\boldsymbol{\alpha}. \quad (12)$$

Note that finding the most violated $\mathbf{y}^t$ that maximizes (12) is a computationally expensive problem when the bag size is large. To accelerate our framework, we propose to use the instance ranking score defined in (1) to enforce the total number of instances in each positive bag to be 15 (see Section IV-A for more details). Moreover, we can beforehand exclude a large number of candidates of $\mathbf{y}^t$ by checking our proposed GMI constraint in (3). In order to further speed up the process, we develop a simple but effective method. The basic idea is to enumerate the candidates of y_i satisfying (3) for each bag $\mathcal{B}_I$ by fixing the labeling of other bags. Then, we iteratively choose the best $\mathbf{y}_I$ for $\mathcal{B}_I$, which maximizes (12), where $\mathbf{y}_I$ is the vector of instance labels in $\mathcal{B}_I$. The procedure will be terminated when the relative change of the objective values of (12) between two successive iterations is less than 0.001. The detailed procedure is listed in Algorithm 2.

IV. EXPERIMENTS

In our experiments, for any given textual query (e.g., "fox"), the relevant Web images that are associated with the word "fox" are firstly ranked using (1). We refer to this initial Web image search method as *Init_Ranking*. We compare our bag-based reranking framework and two existing methods, i.e., WEBSEIC [42] and information bottleneck (IB) reranking (IBRR) [12], for image reranking. It is worth noting that existing MI learning algorithms can be readily adopted in our reranking framework. Observing that the axis-parallel rectangle [7] and the EM-DD [40] are inefficient for this large-scale image retrieval task and citation kNN [35] and MI-SVM [1] are more suitable for predicting the labels of bags rather than instances, we only employ mi-SVM [1] and single-instance learning SVM (SIL-SVM) [2] in this paper. Note that all the instances in a negative bag are treated as negative instances in mi-SVM [1] and SIL-SVM is a special MI learning algorithm, in which all the instances in positive bags (negative bags) are assumed to be positive (negative). When the automatic weak bag annotation process is performed, SIL-SVM is similar to the pseudorelevance-feedback-based method in [37]. In contrast, the assumption in our newly proposed GMI-SVM is that positive instances comprise at least a certain portion of a positive bag, while a negative bag may contain at most a few positive instances.

For WEBSEIC, the top-ranked 400 relevant images are used for KDE [24], as suggested in [42]. Since we do not have the Web page assumption in our application, the 400 images are directly reranked according to the responses from the density function. For IBRR [12], we also choose the top-ranked 400 relevant images, as well as randomly sampled 400 irrelevant images for IB clustering. To fairly compare different reranking methods, we only rerank the top-400 relevant images from the initial text-based search. We do not compare GMI-SVM using the pseudopositive and pseudonegative training bags with other image reranking methods such as those in [5] and [31] because they employ additional manual annotation, which is not required in our GMI-SVM. We also do not compare our work with graph-based methods such as those in [13], [14], and [32] because the recent work [31] shows that these unsupervised reranking methods can only achieve limited performance improvements. In practice, the manifold assumption may not hold well for relevant Web images with diverse appearance variations. Moreover, some graph-based methods [14], [36] using SIFT features generally require high computational costs, whereas our framework can achieve reasonable efficiency by using unoptimized MATLAB code.

A. Experimental Setup

We use the challenging real-world NUS-WIDE data set [4] for experiments. To the best of our knowledge, it is one of the largest annotated Web image data sets publicly available to researchers today. It contains 269 648 images downloaded from the photo sharing Website `Flickr` and their ground-truth annotations for 81 concepts. Each image is also associated with tags given by `Flickr` users. All the 269 648 images are employed as the database images, and all the 81 concept names are used as textual queries to perform the TBIR.

For performance evaluation, we use top-m retrieval precision, which is defined as the percentage of the correctly retrieved images in the top-m retrieved images. Since online users are usually interested in the top-ranked images only, we set m as 20, 40, 60, 80, and 100. We also use average precision (AP) as another evaluation metric. It corresponds to the multipoint AP value of a precision-recall curve and incorporates the effect of recall when AP is computed over the entire classification result set. The mean AP (MAP) (resp. mean top-m precision) is the mean of the AP (resp. the top-m precision) over all the 81 concepts. For all SVM-based methods, we set the regularization parameter $C = 1$ and use the Gaussian kernel with the bandwidth parameter set as the variance of the instances in the training bags.

Similar to [4], we employ three types of global features. For the grid color moment, we extract the first three moments of three channels in the LAB color space from each of the 5×5 fixed grid partitions and aggregate the features into a single 225-D feature vector. The edge direction histogram feature includes 73 dimensions with 72 bins corresponding to edge directions quantized to five angular bins and one bin for nonedge pixels. We also extract a 128-D wavelet texture feature by performing the pyramid-structured wavelet transform and the tree-structured wavelet transform. We further concatenate all three types of visual features into lengthy feature vectors and normalize each feature dimension to zero mean and unit standard deviation. To improve the speed and reduce the memory cost, principal component analysis is then applied for dimension reduction. We observe that the first 119 principal components are sufficient to preserve 90% of the energy. All the images are then projected into the 119-D visual feature space.

For each image, we also extract the textual features from the associated tags. We first remove high-frequency and misspelled words that are not meaningful (e.g., "a," "the," "srk," "xt," and "de") and convert all the remaining words into their prototypes. We then choose the top-200 words with the highest frequency as the vocabulary. For each image, the corresponding 200-D term-frequency feature is then extracted as the textual feature. For the ith image, we further concatenate the visual feature $\mathbf{v}_i$ and the textual feature $\mathbf{t}_i$ together to form the lengthy feature vector $\mathbf{x}_i$, namely, $\mathbf{x}_i = [\lambda \mathbf{v}_i', \mathbf{t}_i']'$, where the weight parameter λ is empirically fixed as 0.1 in the experiments. The database images are grouped into n_B bags by using the k-means clustering method with the distance metric defined as follows:

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\lambda^2 \|\mathbf{v}_i - \mathbf{v}_j\|^2 + \|\mathbf{t}_i - \mathbf{t}_j\|^2} \tag{13}$$

where $\mathbf{v}_i$, $\mathbf{t}_i$ and $\mathbf{v}_j$, $\mathbf{t}_j$ are the visual and textual features of the ith and jth images, respectively.

Recall that, in our GMI-SVM, we enumerate all possible $\mathbf{y} \in \mathcal{Y}$ to find the most violated $\mathbf{y}^t$ (see Section III-D-3). We observe that it is computationally expensive to exploit the enumeration method for GMI-SVM if the number of instances in one bag is larger than 15. We therefore empirically set $k = \lfloor (T/15) \rfloor$ in the k-means clustering method, where T is the total number of relevant images. We throw away the clusters that have instances fewer than 15. For the remaining clusters, we only keep the top-ranked 15 instances with the highest instance ranking scores

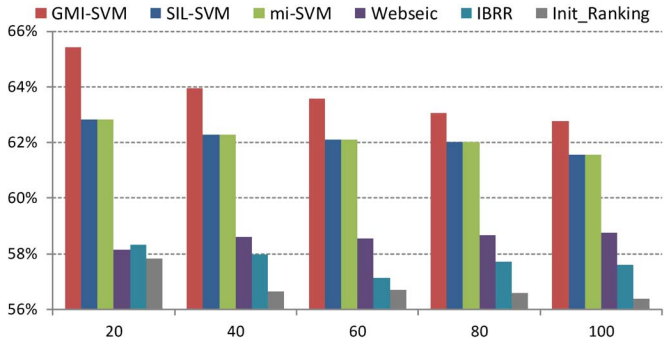


Fig. 3. Mean top-m precisions over 81 concepts of all methods. One positive bag and one negative bag are used for GMI-SVM, SIL-SVM, and mi-SVM.

TABLE I
MAPs OVER 81 CONCEPTS OF ALL METHODS. ONE POSITIVE BAG AND ONE NEGATIVE BAG ARE USED FOR GMI-SVM, SIL-SVM, AND MI-SVM

	GMI-SVM	SIL-SVM	mi-SVM	WEBSEIC	IBRR	Init_Ranking
MAP	**62.4%**	61.6%	61.6%	59.5%	57.8%	57.5%

to form one bag, and the remaining instances are discarded. The bags are then ranked according to the average ranking score of the 15 instances in the bags. In the automatic bag annotation scheme, the top-ranked n_B bags are used as the positive bags, and we also randomly sample $15n_B$ irrelevant images to construct n_B negative bags. The n_B positive and negative bags are then used as the training data for GMI-SVM, mi-SVM, and SIL-SVM.

B. Results of Retrieval Performances

Fig. 3 and Table I show the mean top-m precisions and MAPs of all methods. For GMI-SVM, SIL-SVM, and mi-SVM, one positive bag and one negative bag are used for training. For GMI-SVM, we set proportion $\mu = 0.5$ for positive bags and proportion $\gamma = 0$ for negative bags to fairly compare our GMI-SVM and the other MI learning methods mi-SVM and SIL-SVM. We will discuss the performance variations using different parameters of μ and γ in Section IV-D. We observe that all reranking methods outperform the baseline method Init_Ranking, which demonstrates the effectiveness and the importance of reranking for the TBIR. Moreover, SIL-SVM, mi-SVM, and GMI-SVM achieve significant performance improvements over the other two traditional reranking methods, i.e., WEBSEIC and IBRR, which demonstrates the effectiveness of our proposed bag-based reranking framework. We also observe that the performances of SIL-SVM and mi-SVM are the same. A possible explanation is that, for mi-SVM, the instances in a positive bag are all initialized as positive, and this initialization for positive bags inherently satisfies the convergence criteria in mi-SVM. Thus, after the iterative updating process, all instances in a positive bag are labeled as positive (see [1] for more details about mi-SVM), which is exactly the same as that in SIL-SVM. Our proposed GMI-SVM outperforms SIL-SVM and mi-SVM in all cases. It can be explained from two aspects. On one hand, SIL-SVM and mi-SVM consider all instances in positive bags as positive and all instances in negative bags as negative. Since positive bags may contain some negative instances, the classification performance can be degraded if those negative instances are enforced to be positive. On the other hand, GMI-SVM based on the convex relaxation in [18] can obtain a better optimal solution than other MI learning algorithms for the bag-based reranking framework. The top-ten retrieved images of GMI-SVM, SIL-SVM, mi-SVM, WEBSEIC, IBRR, and Init_Ranking for the textual query "fox" are illustrated in Fig. 4. Again, we observe that GMI-SVM achieves the best performance.

C. Results Using Different Numbers of Training Bags

Based on our bag-based framework, we also compare the performances of SIL-SVM, mi-SVM, and our proposed method GMI-SVM using different numbers of positive/negative training bags. In this experiment, we set $n_B = 1, 3, 5, 7$, and 10. The results of SIL-SVM, mi-SVM, and GMI-SVM are shown in Fig. 5 and Table II. From the results, we observe that GMI-SVM generally outperforms the other two methods in terms of both the mean top-m precisions and the MAPs when using different n_B. When setting $n_B = 1, 3$, and 5, SIL-SVM and mi-SVM achieve similar performances. Nevertheless, when using a larger n_B value (i.e., $n_B = 7$ and 10), SIL-SVM outperforms mi-SVM. An explanation is that, with a large number of training bags, it is generally infeasible to find the optimal solution to the MI learning problem by using the specifically designed heuristics in mi-SVM, which gives mi-SVM worse performances. For GMI-SVM, we also observe that the MAP when setting $n_B = 10$ is worse than that when setting $n_B = 7$ (see Table II). A possible explanation is that the lower ranked training bags are less reliable (i.e., it is more likely that, for the lower ranked positive training bags, the GMI constraint that positive instances take up at least portion μ cannot be satisfied). Therefore, robust classifiers cannot be learned by using these lower ranked bags.

D. GMI-SVM Using Different Positive Proportions for Bags

Recall that, in the proposed GMI assumption, positive instances take up at least proportion μ in a positive bag and at most proportion γ in a negative bag. To evaluate the performance variations using different μ and γ, we set $\mu \in \{0.3, 0.5, 0.7, 0.9\}$ and $\gamma \in \{0, 0.3, 0.5\}$ in this experiment. Given the specific μ and γ, we use n_B positive and n_B negative bags to train the GMI-SVM classifier, where n_B is set as 1, 3, 5, 7, and 10. In Table III, we report the best result of GMI-SVM among the results obtained from $n_B \in \{1, 3, 5, 7, 10\}$.

We observe that, for a fixed γ, the retrieval performance will be degraded if μ becomes too large (i.e., $\mu = 0.7$ and 0.9). An explanation is that some top-ranked positive bags may not contain a large proportion of truly positive instances for every textual query. Specifically, we use the ground-truth labels of the images to analyze the average proportion of truly positive instances in each positive bag of the top-ranked ten bags, and we observe that the average proportion in each positive bag over all 81 concepts is 56.0% when setting $n_B = 10$. As a result, the performance of GMI-SVM using $\mu = 0.7$ or 0.9 will be degraded when the constraints on positive bags are not satisfied. It also explains why GMI-SVM outperforms SIL-SVM, because SIL-SVM is a special case of GMI-SVM when setting $\mu = 1$ and $\gamma = 0$.

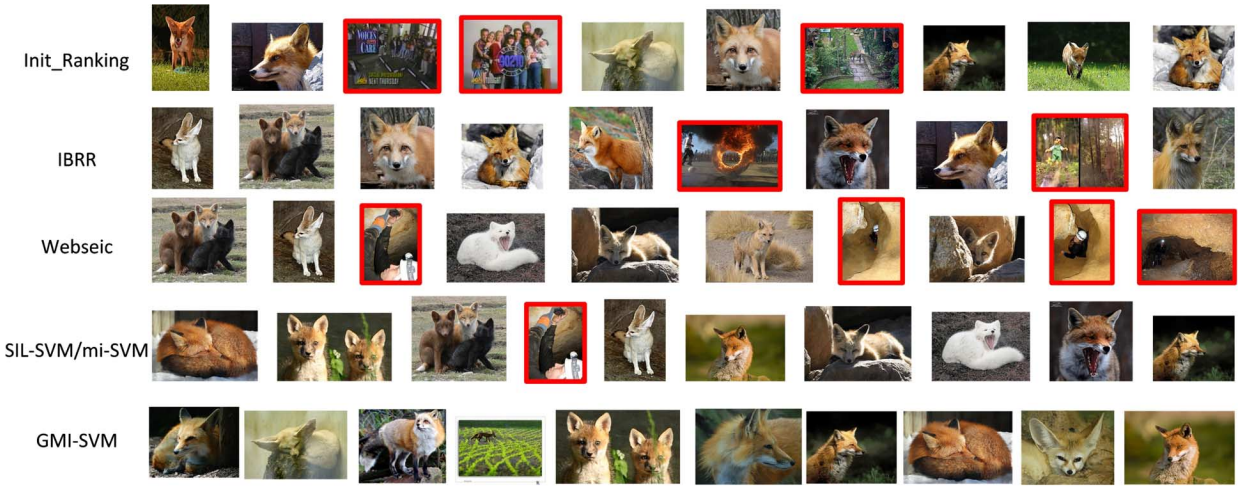


Fig. 4. Top-ten retrieved images of all methods for the textual query "fox." (Red boxes) Incorrect results.

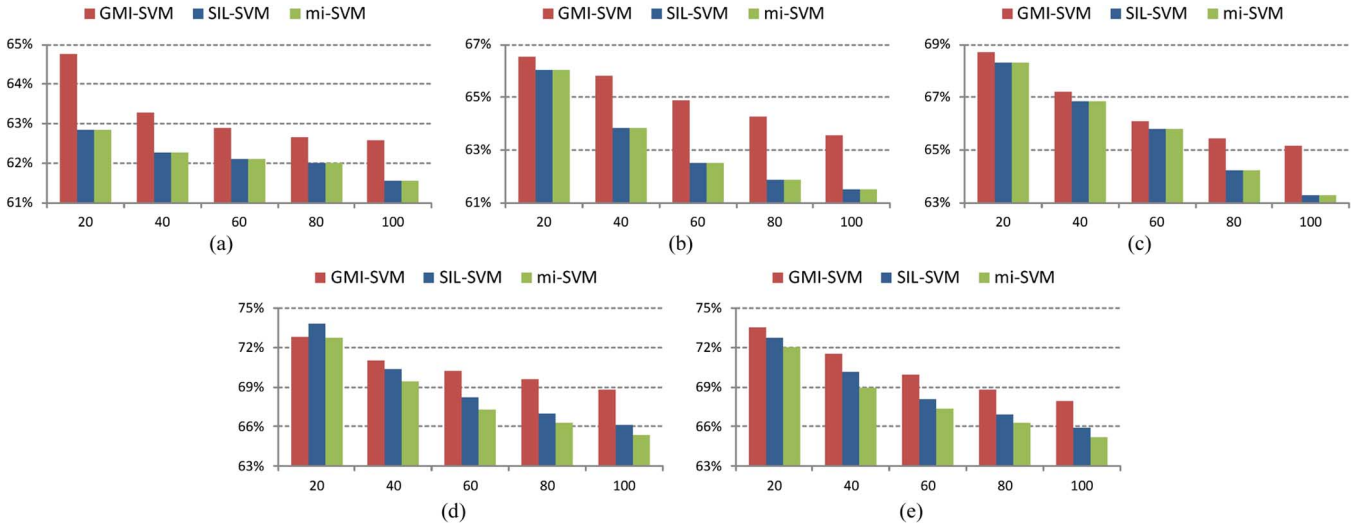


Fig. 5. Mean top-m precisions over 81 concepts of GMI-SVM, SIL-SVM, and mi-SVM using n_B positive and n_B negative bags, where $n_B = 1, 3, 5, 7$, and 10. (a) $n_B = 1$. (b) $n_B = 3$. (c) $n_B = 5$. (d) $n_B = 7$. (e) $n_B = 10$.

TABLE II
MAPs OVER 81 CONCEPTS OF GMI-SVM, SIL-SVM, AND MI-SVM USING DIFFERENT NUMBERS OF POSITIVE AND NEGATIVE TRAINING BAGS

	GMI-SVM	SIL-SVM	mi-SVM
$n_B = 1$	**62.4%**	61.6%	61.6%
$n_B = 3$	**64.4%**	63.2%	63.2%
$n_B = 5$	**66.3%**	64.9%	64.8%
$n_B = 7$	**66.8%**	65.4%	64.7%
$n_B = 10$	**66.6%**	65.3%	64.9%

TABLE III
MAPs OVER 81 CONCEPTS OF GMI-SVM USING DIFFERENT POSITIVE PROPORTIONS (I.E., μ AND γ) FOR POSITIVE AND NEGATIVE BAGS. EACH RESULT IN THE TABLE IS THE BEST AMONG THE RESULTS OBTAINED BY USING DIFFERENT NUMBERS OF POSITIVE AND NEGATIVE TRAINING BAGS (I.E., $n_B = 1, 3, 5, 7$, AND 10)

	$\mu = 0.3$	$\mu = 0.5$	$\mu = 0.7$	$\mu = 0.9$
$\gamma = 0$	65.6%	**66.8%**	66.3%	66.2%
$\gamma = 0.3$	62.2%	66.7%	66.4%	65.9%
$\gamma = 0.5$	60.0%	66.4%	65.3%	65.7%

From Table III, we also observe that, for a fixed μ, GMI-SVM using $\gamma = 0$ generally achieves better performances compared with the results when setting $\gamma = 0.3$ and 0.5, which is consistent with our observation that the negative bags generally do not contain positive instances. Specifically, we also analyze the ground-truth labels of the instances in the negative bags, and we observe that the average proportion of truly positive instances in a negative bag over all 81 concepts is only 1.15% when setting $n_B = 10$. Considering that we fix the number of instances in each positive/negative bag as 15 in the experiments, the number of truly positive instances in a negative bag is approximately zero on the average. From the experiments, we also observe that the per-concept APs of some concepts (such as "person," "lake," "house," and "plant") can be improved by setting $\gamma = 0.3$ rather than $\gamma = 0$. This observation demonstrates that, for those concepts having more positive instances in the negative bags, GMI-SVM can successfully cope with the ambiguities on the instances in the negative bags and thus improve the retrieval performance. Considering that the MAP of GMI-SVM is the best when setting $\gamma = 0$ and $\mu = 0.5$ (see Table III), we fix $\gamma = 0$ and $\mu = 0.5$ in Sections IV-B, IV-C, IV-E, and IV-F.

E. GMI-SVM With Manually Annotated Training Bags

In Sections IV-B–IV-D, the training bags are automatically selected based on the bag ranking score in (2) for GMI-SVM. However, such an automatic weak bag annotation method cannot always guarantee that the GMI constraints in (3) for positive and negative bags are satisfied. RF is allowed in our bag-based framework, in which users are required to manually annotate training bags to further improve the performance of GMI-SVM. For better presentation, we refer to GMI-SVM using the pseudotraining bags obtained from the automatic weak bag annotation method as GMI-SVM and to GMI-SVM using the manually annotated training bags obtained from the RF as GMI-SVM_RF.

In this experiment, we simulate the manual bag annotation process by using the ground-truth labels of the images. For each concept, we use one positive bag and one negative bag as the training data. Since the negative bags do not contain any positive instances in most cases, only the positive bags need to be manually annotated. Specifically, if a positive bag contains at least proportion $\mu = 0.5$ of truly positive instances, it is annotated as a *truly positive* bag. Based on the initial bag ranking results according to the bag ranking score in (2), we observe that only 72 concepts have at least one truly positive bag after checking with the ground-truth labels. Therefore, we report MAPs over the 72 concepts only in this experiment.

Although the image retrieval performance can be improved after conducting RF [11], [19], [39], it is generally time consuming to manually check a considerable number of bags to obtain one truly positive bag in our bag-based image reranking application. After ranking the bags according to the bag ranking score, the users conduct RF to annotate the top-ranked bags in order to obtain the truly positive bags for training. We refer to GMI-SVM using the RF method as GMI-SVM_RF$_{\mathrm{Init}}$ if the bags are ranked according to the initial bag ranking score in (2). To facilitate the annotation process, we also propose another new bag ranking score $\tilde{S}$ in (14) and refer to GMI-SVM using the RF method as GMI-SVM_RF$_{\mathrm{New}}$ if the bags are ranked according to $\tilde{S}$. Specifically, we first learn an initial GMI-SVM classifier by using the automatic weak bag annotation process (see Section III-B). Then, the decision values of the instances from the learned GMI-SVM classifier can be used to calculate the new bag ranking score $\tilde{S}$ as follows:

$$\tilde{S}(\mathcal{B}_I) = \frac{\sum_{\mathbf{x}\in\mathcal{B}_I} f(\mathbf{x})}{|\mathcal{B}_I|} \tag{14}$$

where $f(\mathbf{x})$ is the decision value of the training instance $\mathbf{x}$ from the initial GMI-SVM classifier. Recall that the images are grouped into clusters using k-means clustering; thus, we can also use $f(\mathbf{x})$ to rank the instances in each cluster, in which we still keep the top-ranked 15 instances to construct one pseudopositive bag and discard the remaining instances. It is noteworthy that the top-ranked 15 instances based on $f(\mathbf{x})$ may be different from the top-ranked 15 instances based on the initial instance ranking score in (1). After that, we rank the bags according to the new bag ranking score in (14) and employ the ground-truth labels to find the top-ranked truly positive bags, in which the corresponding GMI constraints are satisfied.

TABLE IV
MAPs OVER 72 CONCEPTS OF GMI-SVM USING PSEUDOTRAINING BAGS OBTAINED FROM THE AUTOMATIC WEAK BAG ANNOTATION METHOD AND GMI-SVM_RF USING MANUALLY ANNOTATED TRAINING BAGS OBTAINED FROM THE RF. NOTE, FOR GMI-SVM_RF, THE TOP-RANKED TRULY POSITIVE BAGS CAN BE OBTAINED BY USING THE INITIAL BAG RANKING SCORE IN (2) [RESP. THE NEW BAG RANKING SCORE IN (14)], WHICH IS REFERRED TO AS GMI-SVM_RF$_{\mathrm{Init}}$ (RESP. GMI- SVM_RF$_{\mathrm{New}}$). ONE POSITIVE AND ONE NEGATIVE BAG ARE USED FOR ALL METHODS

	GMI-SVM	GMI-SVM_RF$_{\mathrm{Init}}$	GMI-SVM_RF$_{\mathrm{New}}$
MAP	66.5%	69.9%	70.8%

TABLE V
AVERAGE CPU TIME (IN SECONDS) PER TEXTUAL QUERY FOR ALL METHODS

	GMI-SVM	SIL-SVM	mi-SVM	WEBSEIC	IBRR	Init_Ranking
CPU	1.120	0.025	0.026	0.027	105.354	0.0005

We report the MAPs over 72 concepts of GMI-SVM, GMI-SVM_RF$_{\mathrm{Init}}$, and GMI-SVM_RF$_{\mathrm{New}}$ in Table IV. We observe that both GMI-SVM_RF$_{\mathrm{Init}}$ and GMI-SVM_RF$_{\mathrm{New}}$ outperform GMI-SVM in terms of the MAP over 72 concepts, which demonstrates that the retrieval performance of GMI-SVM can be further improved by using the manually annotated training bags. GMI-SVM_RF$_{\mathrm{New}}$ performs slightly better than GMI-SVM_RF$_{\mathrm{Init}}$. A possible explanation is that the top-ranked truly positive training bags based on the new bag ranking score in (14) are more reliable and can be thus used to learn a more robust classifier.

In GMI-SVM_RF$_{\mathrm{New}}$ (resp. GMI-SVM_RF$_{\mathrm{Init}}$), on the average, 1.46 (resp. 2.03) bags need to be examined by users before obtaining one truly positive bag for each concept. Thus, the annotation efforts from the users can be greatly alleviated by using the new bag ranking score $\tilde{S}$ in (14).

It is worth mentioning that the users are generally reluctant to conduct manual annotations. Thus, we just treat GMI-SVM_RF as an additional extension, and it is therefore not the main focus of this paper. More details (e.g., how to develop a novel and effective annotation user interface to facilitate the bag annotation process in the real applications and how to fairly compare our approach with conventional RF methods) will be investigated in the future.

F. CPU Time for Image Retrieval and Convergence Analysis

We report the average central processing unit (CPU) time of the TBIR for different methods. For GMI-SVM, SIL-SVM, and mi-SVM, we still use one positive bag and one negative bag obtained by using the automatic weak bag annotation process. All the experiments are implemented with unoptimized MATLAB codes and performed on a workstation (3.33-GHz CPU with 32-GB random access memory). The average CPU-time overall textual queries (81 concepts) are shown in Table V. Init_Ranking is very fast because of the utilization of the inverted-file technique. Moreover, WEBSEIC performs very fast since only 400 relevant images are reranked by using a KDE-based method. SIL-SVM and mi-SVM have comparable training time because mi-SVM converges within a few iterations in most cases. IBRR requires a lot of time for the IB clustering process. Moreover, our proposed method GMI-SVM achieves reasonable efficiency for TBIR using unoptimized MATLAB codes. For GMI-SVM, on

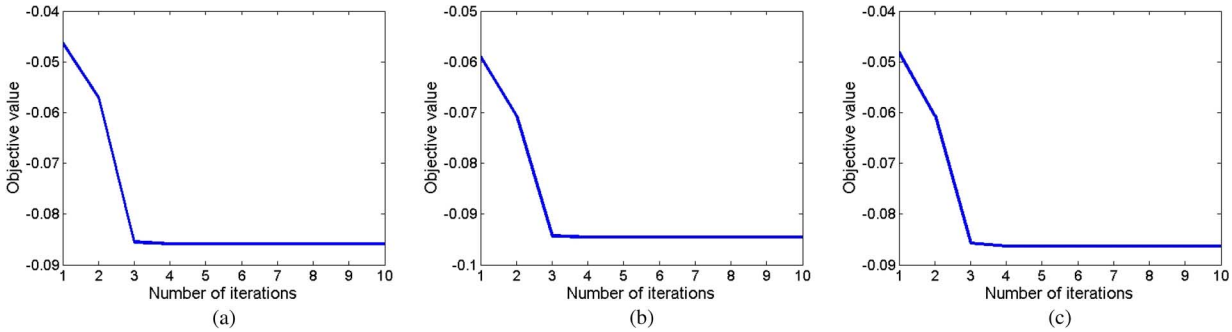


Fig. 6. Illustration of the convergence of GMI-SVM. (a) "Bear." (b) "Bird." (c) "Bridge."

the average, the iterative optimization algorithm (i.e., the cutting-plane algorithm introduced in Section III-D-2) takes about six iterations to converge for each concept. In Fig. 6, we take three concepts (i.e., "bear," "bird," and "bridge") as examples to illustrate the convergence of GMI-SVM, in which the vertical axis indicates the objective value of (11) and the horizontal axis gives the number of iterations. We have similar observations for other concepts.

V. Conclusion and Future Work

In this paper, we have proposed a bag-based framework for large-scale TBIR. `Flickr` images with textual descriptions (i.e., tags) have been used for this real-world application. Given a textual query, relevant images are to be reranked after the initial text-based search. Instead of directly reranking the relevant images by using traditional image reranking methods, we have partitioned the relevant images into clusters. By treating each cluster as a "bag" and the images in a bag as "instances," we have formulated this problem as a MI learning problem. MI learning methods such as mi-SVM can be readily adopted in our bag-based framework. To address the ambiguities on the instance labels in both positive and negative bags, we have developed GMI-SVM to further enhance retrieval performance, in which the labels from the bag level have been propagated to the instance level. To facilitate (G)MI learning in our framework, we have propose an automatic bag annotation method to automatically find positive and negative bags for training classifiers. Our framework using the automatic bag annotation method can achieve the best performance, as compared with other traditional image reranking methods on the NUS-WIDE data set. Moreover, we have shown that the performance of GMI-SVM can be further improved, by using the truly positive training bags from user annotation in a RF process.

Currently, we use the k-means clustering method based on visual and textual features to partition the relevant images into bags/clusters in our weak bag annotation process. In the future, we will investigate more effective clustering methods to further improve the performance of our framework. Inspired by [8], we also plan to extend this paper for video event recognition.

References

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2003, pp. 561–568.

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

[3] L. Chen, D. Xu, I. W. Tsang, and J. Luo, "Tag-based web photo retrieval improved by batch mode re-tagging," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recogn.*, 2010, pp. 3440–3446.

[4] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng, "NUS-WIDE: A real-world web image database from national University of Singapore," in *Proc. ACM Int. Conf. Image Video Retrieval*, 2009, pp. 1–9.

[5] J. Cui, F. Wen, and X. Tang, "Real time google and live image search re-ranking," in *Proc. 16th ACM Int. Conf. Multimedia*, 2008, pp. 729–732.

[6] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Comput. Surv.*, vol. 40, no. 2, pp. 1–60, Apr. 2008.

[7] T. Dietterich, R. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artif. Intell.*, vol. 89, no. 1/2, pp. 31–71, Jan. 1997.

[8] L. Duan, D. Xu, I. W. Tsang, and J. Luo, "Visual event recognition in videos by learning from web data," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recogn.*, 2010, pp. 1959–1966.

[9] P. Gehler and S. Nowozin, "Infinite kernel learning" Max Planck Inst. Biol. Cybern., Tuebingen, Germany, Tech. Rep. TR-178, 2008.

[10] P. Gehler and S. Nowozin, "Let the kernel figure it out; principled learning of pre-processing for kernel classifiers," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 2836–2843.

[11] J. He, M. Li, H. Zhang, H. Tong, and C. Zhang, "Manifold-ranking based image retrieval," in *Proc. ACM Multimedia*, 2004, pp. 9–16.

[12] W. H. Hsu, L. S. Kennedy, and S.-F. Chang, "Video search reranking via information bottleneck principle," in *Proc. 14th ACM Int. Conf. Multimedia*, 2006, pp. 35–44.

[13] W. H. Hsu, L. S. Kennedy, and S.-F. Chang, "Video search reranking through random walk over document-level context graph," in *Proc. 15th ACM Int. Conf. Multimedia*, 2007, pp. 971–980.

[14] Y. Jing and S. Baluja, "Pagerank for product image search," in *Proc. 17th Int. Conf. World Wide Web*, 2008, pp. 307–316.

[15] J. E. Kelley, "The cutting plane method for solving convex programs," *SIAM J. Appl. Math.*, vol. 8, no. 4, pp. 703–712, Dec. 1960.

[16] S.-J. Kim and S. Boyd, "A minimax theorem with applications to machine learning, signal processing, and finance," *SIAM J. Optim.*, vol. 19, no. 3, pp. 1344–1367, 2008.

[17] 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*, 2009, pp. 15–30.

[18] Y.-F. Li, I. W. Tsang, J. T. Kwok, and Z.-H. Zhou, "Tighter and convex maximum margin clustering," in *Proc. 22nd Int. Conf. Artif. Intell. Stat.*, 2009, pp. 344–351.

[19] Y. Liu, D. Xu, I. W. Tsang, and J. Luo, "Textual query of personal photos facilitated by large-scale web data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 1022–1036, May 2011.

[20] O. Maron and T. Lonzano-Pérez, "A framework for multiple-instance learning," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 1998, pp. 570–576.

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

[22] A. Rakotomamonjy, F. R. Bach, and Y. Grandvalet, "SimpleMKL," *J. Mach. Learn. Res.*, vol. 9, pp. 2491–2521, 2008.

[23] Y. Rui, T. S. Huang, and S. Mehrotra, "Content-based image retrieval with relevance feedback in MARS," in *Proc. IEEE Int. Conf. Image Process.*, 1997, pp. 815–818.

[24] D. W. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization*. New York: Wiley-Interscience, 1992.

[25] S. Scott, J. Zhang, and J. Brown, "On generalized multiple-instance learning," *Int. J. Comput. Intell. Appl.*, vol. 5, pp. 21–35, 2005.

[26] N. Slonim, N. Friedman, and N. Tishby, "Unsupervised document classification using sequential information maximization," in *Proc. 25th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2002, pp. 129–136.

[27] A. W. M. Smeulders, M. Worring, S. Santini, and A. Gupta, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1349–1380, Dec. 2000.

[28] Q. Tao and S. Scott, "A faster algorithm for generalized multiple-instance learning," in *Proc. 17th Int. Florida Artif. Intell. Res. Soc. Conf.*, 2004, pp. 550–555.

[29] Q. Tao, S. D. Scott, N. V. Vinodchandran, and T. T. Osugi, "SVM-based generalized multiple-instance learning via approximate box counting," in *Proc. 21st Int. Conf. Mach. Learn.*, 2004, pp. 779–806.

[30] Q. Tao, S. D. Scott, N. V. Vinodchandran, T. T. Osugi, and B. Mueller, "Kernels for generalized multiple-instance learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 12, pp. 2084–2098, Dec. 2008.

[31] X. Tian, D. Tao, X.-S. Hua, and X. Wu, "Active reranking for web image search," *IEEE Trans. Image Process.*, vol. 19, no. 3, pp. 805–820, Mar. 2010.

[32] X. Tian, L. Yang, J. Wang, Y. Yang, X. Wu, and X.-S. Hua, "Bayesian video search reranking," in *Proc. 16th ACM Int. Conf. Multimedia*, 2008, pp. 131–140.

[33] S. Tong and E. Chang, "Support vector machine active learning for image retrieval," in *Proc. 9th ACM Int. Conf. Multimedia*, 2001, pp. 107–118.

[34] S. Vijayanarasimhan and K. Grauman, "Keywords to visual categories: Multiple-instance learning for weakly supervised object categorization," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.*, 2008, pp. 1–8.

[35] J. Wang and J.-D. Zucker, "Solving the multiple-instance problem: A lazy learning approach," in *Proc. 17th Int. Conf. Mach. Learn.*, 2000, pp. 1119–1125.

[36] S. Wang, Q. Huang, S. Jiang, L. Qin, and Q. Tian, "Visual contextrank for web image re-ranking," in *Proc. 1st ACM Workshop Large-Scale Multimedia Retrieval Mining*, 2009, pp. 121–128.

[37] R. Yan, A. G. Hauptmann, and R. Jin, "Multimedia search with pseudo-relevance feedback," in *Proc. ACM Int. Conf. Image Video Retrieval*, 2003, pp. 238–247.

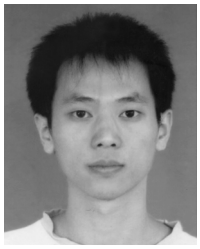
[38] C. Zhang, J. Y. Chai, and R. Jin, "User term feedback in interactive text-based image retrieval," in *Proc. 28th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2005, pp. 51–58.

[39] L. Zhang, F. Lin, and B. Zhang, "Support vector machine learning for image retrieval," in *Proc. IEEE Int. Conf. Image Process.*, 2001, pp. 721–724.

[40] Q. Zhang and S. A. Goldman, "EM-DD: An improved multiple-instance learning technique," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2002, pp. 1073–1080.

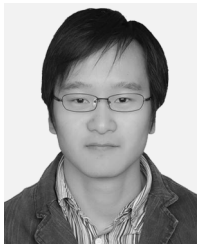
[41] Q. Zhang, S. A. Goldman, W. Yu, and J. E. Fritts, "Content-based image retrieval using multiple-instance learning," in *Proc. 19th Int. Conf. Mach. Learn.*, 2002, pp. 682–689.

[42] Z.-H. Zhou and H.-B. Dai, "Exploiting image contents in web search," in *Proc. 20th Int. Joint Conf. Artif. Intell.*, 2007, pp. 2928–2933.



Lixin Duan received the B.E. degree from the University of Science and Technology of China, Hefei, China, in 2008. He is currently working toward the Ph.D. degree at the School of Computer Engineering, Nanyang Technological University, Singapore.

Mr. Duan was a recipient of the Microsoft Research Asia Fellowship in 2009 and the Best Student Paper Award in the IEEE Conference on Computer Vision and Pattern Recognition 2010.



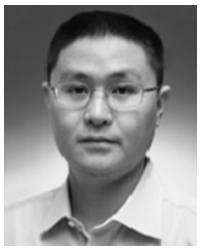
Wen Li received the B.S. and M.Eng. degrees from Beijing Normal University, Beijing, China, in 2007 and 2010, respectively. He is currently working toward the Ph.D. degree in the School of Computer Engineering, Nanyang Technological University, Singapore.

His main interests include multiple instance learning, image understanding, and learning from Web data.

Ivor Wai-Hung Tsang received the Ph.D. degree in computer science from Hong Kong University of Science and Technology, Kowloon, Hong Kong, in 2007.

He is currently an Assistant Professor with the School of Computer Engineering, Nanyang Technological University, Singapore, where he is also the Deputy Director of the Center for Computational Intelligence.

Dr. Tsang was a recipient of the IEEE TRANSACTIONS ON NEURAL NETWORKS Outstanding 2004 Paper Award in 2006, the second class prize of the National Natural Science Award 2008, China, in 2009, the Microsoft Fellowship in 2005, the Best Paper Award from the IEEE Hong Kong Chapter of Signal Processing Postgraduate Forum in 2006, and the Best Student Paper Prize at the IEEE Conference on Computer Vision and Pattern Recognition 2010.



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

While working toward the Ph.D. degree, he was with the Microsoft Research Asia, Beijing, China, and the Chinese University of Hong Kong, Shatin, Hong Kong, for more than two years. He was a Postdoctoral Research Scientist with Columbia University, New York, NY, for one year. He is currently an Assistant Professor with Nanyang Technological University, Singapore. His current research interests include computer vision, statistical learning, and multimedia content analysis.

Dr. Xu was the coauthor of a paper that won the Best Student Paper Award in the prestigious IEEE International Conference on Computer Vision and Pattern Recognition in 2010.