Region Selection based on Evidence Confidence for Localized Content-Based Image Retrieval

Wu-Jun Li, Dit-Yan Yeung

Abstract

Over the past decade, multiple-instance learning (MIL) has been successfully utilized to model the localized content-based image retrieval (CBIR) problem, in which a bag corresponds to an image and an instance corresponds to a region in the image. However, existing feature representation schemes are not effective enough to describe the bags in MIL, which hinders the adaptation of sophisticated single-instance learning (SIL) methods for MIL problems. In this paper, we propose a conceptually simple but very powerful method for localized CBIR. First, a novel measure, called *evidence confidence (EC)*, is proposed to select those regions which are most likely to support the labels of the images (i.e., bags). Then, based on the selected regions, a very effective feature representation scheme, which is also very computationally efficient and robust to labeling noise, is proposed to describe the bags. As a result, the MIL problem is converted into a standard SIL problem and traditional SIL methods, such as support vector machine (SVM), can be easily adapted for localized CBIR. Furthermore, using our feature representation scheme, a semi-supervised learning framework based on manifold regularization is proposed to incorporate unlabeled data into the training process. Experimental results on two challenging data sets show that our supervised learning method, called EC-SVM, outperforms the state-of-the-art methods and our semi-supervised learning framework, called EC-LapSVM, can lead to further performance improvement.

Index Terms

Localized content-based image retrieval, CBIR, multiple-instance learning, object categorization, semi-supervised learning, manifold regularization

1 INTRODUCTION

According to the low-level image features used in the retrieval process, existing *content-based image retrieval* (CBIR) methods can be categorized into two major classes, namely, global methods and localized methods (a.k.a. *localized CBIR* [1], [2]). Global methods exploit features characterizing the global view of an image, such as color histograms, to compute the similarity between images. These methods have been widely used by traditional CBIR systems. Although global features can be extracted easily, in many cases, only a small part or several small parts of the image are useful for characterizing the visual content of the image. If features from the whole image area are used to represent an image, the useful information may be overridden by noisy information from irrelevant regions. For example, in Figure 3, if the interest of the user is in the object "FabricSoftenerBox", the two images with label "FabricSoftenerBox" should have higher similarity than the first two images in the upper row. However, the first two images in the upper row are expected to give higher similarity than the two images in the leftmost column if global methods are used. On the contrary, localized CBIR, which describes the task where the user is only interested in a portion of the image with the rest being irrelevant [1], [2], is more natural and is in line with human perception. For example, in Figure 3, a user may only be interested in the *apple* in the image with label "Apple".

A new machine learning paradigm called *multiple-instance learning* (MIL) [3] was proposed to model those learning problems where the class labels are only associated with sets of examples rather than individual examples. In MIL, an individual example is called an *instance* and a set of instances is called a *bag*. Training labels are associated with bags rather than instances. A bag is labeled positive if at least one of its instances is positive; otherwise, the bag is negative. In this paper, we use the term *single-instance learning* (SIL) to refer to the traditional supervised learning paradigm in which each individual example has a class label.

In the existing localized CBIR work, the region of interest can be either at a fixed location or marked by the user. The first case does not conform to the general image retrieval task and the second case requires too much effort from the user making it unattractive in practice. Hence, the *focus of this paper* is to design a general automatic localized CBIR system that does not

Wu-Jun Li and Dit-Yan Yeung are with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Hong, China. E-mail: liwujun@cse.ust.hk; dyyeung@cse.ust.hk

necessarily require the user to mark the region of interest. Specifically, we require that multiple labeled images be provided for the system to automatically learn the interest of the user. This can be achieved through relevance feedback or by inputting a *query image set* [2] labeled as positive or negative by the user according to whether the images contain the target regions of interest. Under this setting, the underlying learning problem for localized CBIR is essentially an MIL problem where an image corresponds to a bag and each region in the image corresponds to an instance.

1.1 Motivations

Few of the existing MIL methods have designed effective feature representation schemes to describe the bags, making it difficult to adapt some sophisticated SIL methods for MIL problems. DD-SVM [4] is the first MIL method trying to propose a feature representation scheme for the bags in MIL to convert MIL into SIL. However, the features of DD-SVM are very sensitive to noise and incur very high computation cost. MILES [5] (Multiple Instance Learning via Embedded instance Selection) also converts MIL into a standard SIL problem via feature mapping, in which each feature is defined by an instance from the training bags, including both positive and negative bags. Although MILES is much more efficient than DD-SVM, the feature space for representing bags is of very high dimensionality because it contains too many irrelevant features. Hence, appropriate classifiers that can make use of the feature representation scheme in MILES are limited to those that can perform both feature selection and classification simultaneously, such as 1-norm SVM [5]. Therefore, *one motivation* of this work is to design an effective as well as efficient feature representation scheme for representing the bags in MIL.

Furthermore, in CBIR, we cannot expect the user to input a large number of labeled images. On the other hand, it is usually easy to get a large number of unlabeled images from the image repository. Hence, semi-supervised learning methods [6], which can incorporate unlabeled data into the training process, are very meaningful for CBIR. Most existing semi-supervised learning methods for MIL, such as those in [7], [8], are *transductive*[1] in nature, which means that the trained classifier cannot be generalized easily to perform prediction on unseen test data. However,

1. In the machine learning community, *inductive learning* refers to learning a general classification function that can be applied to future unforeseen examples, while the goal of *transductive learning* is to only infer the class labels of the unlabeled examples which are already seen during the training phase.

the main goal of CBIR is to retrieve relevant *unseen* images from the image repository. Hence, *another motivation* of this work is to design an *inductive* semi-supervised learning method to effectively utilize unlabeled images for training.

1.2 Main Contributions

In this paper, we propose a feature representation scheme for the bags in MIL to convert MIL into SIL and adapt several sophisticated SIL techniques to solve MIL problems. The main contributions of this paper are summarized as follows:

- We propose a novel measure, called *evidence confidence (EC)*, to select those regions which are most likely to support the labels of the images (i.e., bags).
- A very effective feature representation scheme, which is also very computationally efficient and robust to labeling noise, is proposed to describe the bags based on the selected regions. As a result, the MIL problem is converted into a standard SIL problem and an SVM is successfully adapted for localized CBIR.
- Using our feature representation scheme, a semi-supervised learning framework based on manifold regularization is proposed to incorporate unlabeled data into the training process.
- We compare our methods extensively with many state-of-the-art methods on two challenging data sets to demonstrate the promising performance of our methods with respect to multiple performance metrics, including accuracy, efficiency and robustness.
- Besides its extraordinarily good performance, our method is also conceptually simple and easily implementable, which makes it a very practical method for localized CBIR.

It should be emphasized that the *focus of this work* is on CBIR rather than image classification. Although the techniques for CBIR are also suitable for image classification, and vice versa, their application scenarios are somewhat different. While for image classification a large number of labeled images can be provided for training, for CBIR it is unreasonable (or impractical) to require the user to input a large number of query images.

2 Related Work

Recently, a lot of work has employed MIL for localized CBIR or image classification. Diverse density (DD) [9], [10] and its extensions are very popular methods for localized CBIR. In [9],

[10], DD was used for CBIR where an image is represented as a bag of feature vectors extracted by the single-blob with neighbors (SBN) method. EM-DD [11], [12], which combines expectation maximization (EM) [13] with the DD formulation, was used for CBIR in which the images are partitioned by a segmentation algorithm. Rahmani et al. [2] formally defined localized CBIR and used Ensemble EM-DD to retrieve images whose features are extracted based on segmentation. Zhang et al. [14] proposed a salient points method to represent the instances in the bags and presented an improved EM-DD method for localized CBIR. Very recently, Rahmani et al. [1] proposed to use generalized EM-DD (GEM-DD), which can be seen as an improved version of Ensemble EM-DD in [2], for localized CBIR, and very promising performance was achieved.

Some other methods try to modify standard SIL methods for MIL by introducing constraints derived from the MIL formulation. In [15], two methods based on SVM were proposed, one (mi-SVM) for instance-level classification and the other (MI-SVM) for bag-level classification. Both methods are formulated as mixed integer quadratic programming problems. Deterministic annealing [16] was applied to the SVM formulation to find better local mimima compared to the heuristic methods in [15]. In [17], a kernel function was proposed directly for bags. With this multi-instance (MI) kernel, in principle any kernel-based classifier can be trained for classifying the bags. Kwok et al. [18] extended this work by proposing a marginalized MI kernel to convert the MIL problem from an incomplete data problem to a complete data problem. Tao et al. [19] proposed a count-based kernel for generalized MIL. Zhou et al. [20] treated the instances in MIL as non-i.i.d. samples and proposed two graph kernels for MIL. Yang et al. [21] formulated the image annotation problem as an MIL problem based on the region representation and proposed an asymmetric SVM algorithm to solve it. Cheung et al. [22] proposed a new regularization framework for MIL and applied the constrained concave-convex procedures to solve the nonlinear optimization problem. Zhou et al. [23] proposed a multi-instance multi-label formulation for scene classification. Cholleti et al. [24] adapted Winnow for MIL and proposed a method called MI-Winnow for image retrieval. MILBoost [25], [26] adapted Boosting for MIL. In [27], an online version of MILBoost was proposed for visual tracking. In [28], an SVM-based method, called sparse MIL (sMIL), was proposed for sparse positive bags by directly enforcing the desired constraint that at least one of the instances in a positive bag is positive. Vijayanarasimhan and Grauman [29] proposed an improved version of sMIL for weakly supervised object categorization. Zhou et al. [30] studied the relationship between

MIL and semi-supervised learning. Qi et al. [31] employed concurrent tensors to model the inter-dependency between the instances for image categorization. Wang et al. [32] proposed an adaptive framework for MIL that adapts to different application domains by learning the domain-specific mechanisms merely from labeled bags. Raykar et al. [33] extended the relevance vector machine (RVM) [34] to MIL for feature selection. Semi-supervised learning [7], [8], [35] and active learning [36], [37], [38] have also been applied to MIL.

Recently, some methods try to convert MIL into a standard single-instance problem and then solve it with conventional methods by representing the bags as feature vectors. In [39], the instances of all the training bags, including positive and negative bags, were firstly clustered into d groups, and then each bag was represented by a binary feature vector of length d, with the ith feature being set to 1 if the corresponding bag contains instances in the ith group and 0 otherwise. Finally, several SVMs trained on these binary feature vectors were ensembled to perform prediction. DD-SVM [4] trained an SVM based on a feature mapping defined on some instance prototypes learned by DD. MILES [5], an extension of [40], proposed a novel instance-based feature mapping and adopted the 1-norm SVM model for simultaneous feature selection and classification. These methods, such as DD-SVM and MILES, have achieved state-of-the-art performance. Furthermore, the conversion of MIL into SIL makes it very convenient to adapt sophisticated SIL methods for MIL. Therefore, in this paper, we will follow the path of DD-SVM and MILES to represent the bags as feature vectors. As stated in section 1.1, however, existing feature representation schemes are not effective enough for localized CBIR. This has motivated the research work reported in this paper.

3 A Feature Representation Scheme for MIL

In this paper, B_i^+ denotes a positive bag and B_i^- denotes a negative bag. When the label of a bag is irrelevant, we simply denote the bag as B_i. B_{ij}^+ denotes an instance in a positive bag B_i^+ and B_{ij}^- is an instance in a negative bag B_i^-. Let $\mathfrak{B} = \{B_1^+, B_2^+, \ldots, B_{n^+}^+, B_1^-, B_2^-, \ldots, B_{n^-}^-\}$ denote the set of all n^+ positive and n^- negative training bags. For each bag B_i, its bag label is $y_i \in \{+1, -1\}$. All the instances are represented as feature vectors of the same dimensionality. Furthermore, in CBIR, a bag refers to an image and an instance corresponds to a region in some image.

3.1 Evidence Confidence based Region Selection

According to the MIL problem formulation, a bag is labeled positive if at least one of its instances is positive; otherwise, the bag is labeled negative. Because whether or not there exist positive instances in a bag provides *evidence* for supporting the bag's label, we call the positive instances *evidence instances*. If a bag refers to an image, evidence instances are also referred to as *evidence regions*.

3.1.1 Region Selection Algorithm

The *evidence confidence* $\mathrm{EC}(B_{gh})$, which is used to represent the confidence (or likelihood) for the instance B_{gh} in bag B_g to be an evidence instance, is defined as follows:

$$\mathrm{EC}(B_{gh}) = \prod_{i=1}^{n^+} \Pr(B_{gh} \,|\, B_i^+) \prod_{i=1}^{n^-} \Pr(B_{gh} \,|\, B_i^-), \tag{1}$$

where $\Pr(B_{gh} \,|\, B_i)$ is estimated based on the noisy-OR model [9]:

$$\Pr(B_{gh} \,|\, B_i^+) = \left\{ 1 - \prod_j \left[1 - \Pr(B_{gh} \,|\, B_{ij}^+) \right] \right\} \tag{2}$$

$$\Pr(B_{gh} \,|\, B_i^-) = \prod_j \left[1 - \Pr(B_{gh} \,|\, B_{ij}^-) \right]. \tag{3}$$

Here, $\Pr(B_{gh} \,|\, B_{ij})$ is estimated as follows:

$$\Pr(B_{gh} \,|\, B_{ij}) = \exp\left\{ - \frac{\sum_k (B_{ijk} - B_{ghk})^2}{\sigma^2} \right\}, \tag{4}$$

where σ is a scaling parameter, k ranges over all the features, and B_{ijk} and B_{ghk} refer to the kth features of the corresponding feature vectors.

The noisy-OR model conforms well to the MIL formulation. From (2), we can see that as long as one instance in B_i^+ is close to B_{gh}, $\Pr(B_{gh} \,|\, B_i^+)$ will be high. From (3), we can see that only if all the instances in B_i^- are far away from B_{gh}, $\Pr(B_{gh} \,|\, B_i^-)$ will be high. Hence, if every positive bag contains at least one instance close to B_{gh} and simultaneously all the instances in the negative bags are far away from B_{gh}, $\mathrm{EC}(B_{gh})$ will be high. Therefore, $\mathrm{EC}(\cdot)$ actually reflects the *confidence* for the instance to be an evidence instance. The larger the EC value of the instance, the more likely this instance will be an evidence instance.

The definition of EC "looks" similar to that of DD [9]. However, except that both EC and DD use the noisy-OR model to compute the corresponding probability, the rationales behind EC and DD are in fact very different. This will be demonstrated in detail in the following subsection.

From the MIL definition, we know that evidence instances only exist in the positive bags and each positive bag contains at least one evidence instance. Hence, we just need to compute the EC values for all instances from the positive bags and then select those instances with the largest EC values from each positive bag. Those selected instances are most likely to be evidence instances.

Another issue about the above instance selection method is how many instances should be selected from each positive bag. This may be determined from prior knowledge. More specifically, for localized CBIR, this parameter can be completely observed from the given training images. For example, for the SIVAL image set [2], [14] used in our experiment, since from the training images we observe that the target object occupies about 15% of the image area for most images and each image (bag) contains 32 instances, it is very reasonable to set this parameter to 5 which is about $15.6\%\,(5/32)$ of the number of all instances in a bag.

Algorithm 1 summarizes the instance selection procedure presented above.

Algorithm 1 Evidence Confidence based Instance Selection for MIL

Input: All training bags $B_1^+, \ldots, B_{n^+}^+, B_1^-, \ldots, B_{n^-}^-$; Parameter m indicating how many instances should be selected from each positive bag.

Initialize: $E^* = \phi$

for $g = 1$ **to** n^+ **do**

  for $h = 1$ **to** $|B_g^+|$ **do**

    Compute $\mathrm{EC}(B_{gh}^+)$ according to (1)

  end for

  Select m instances with the largest EC values from B_g^+, and add the selected instances to E^*

end for

Output: E^*, a set of selected instances which are most likely to be evidence instances.

Remark *1: An alternative strategy for instance selection is to select those instances with the largest EC values from all the instances of the positive training bags. More specifically, we put all the instances from the positive training bags into a set, and then from this set we select N instances with the largest EC values. From our experiments, we find that, with the same number*

of selected instances, i.e., $N = n^+ \times m$, the instance selection strategy in Algorithm 1 can achieve slightly better performance than this alternative strategy. For example, when $n = 8$ in Figure 10, the accuracy for the strategy in Algorithm 1, called EC-SVM in Figure 10, is 81.3, but the accuracy will decrease to 80.1 if this alternative strategy is adopted. The better performance of the strategy in Algorithm 1 might be attributed to the fact that it can introduce diversity (from different bags) into the feature representation. Therefore, in this paper, we adopt Algorithm 1 for instance selection.

3.1.2 Comparison with DD

The DD method [9] tries to find the *target point* [2] by maximizing the following objective function:

$$\arg\max_c \prod_{i=1}^{n^+} \Pr(c \,|\, B_i^+) \prod_{i=1}^{n^-} \Pr(c \,|\, B_i^-), \tag{5}$$

where $c \in C$ and C is the space of all possible instances, including both the observed training instances in $\mathfrak{B}$ and the (possibly infinite number of) unobserved ones.

$\Pr(c \,|\, B_i)$ is also estimated based on the noisy-OR model [9]. However, unlike our EC definition, $\Pr(c \,|\, B_{ij})$ in DD is estimated as follows:

$$\Pr(c \,|\, B_{ij}) \propto \exp\left\{ -\sum_k \left(s_k (B_{ijk} - c_{\cdot k})^2 \right) \right\}, \tag{6}$$

where c corresponds to a feature vector, which might not be an observed instance, in the input instance space, k ranges over all the features, s_k is a scaling coefficient for the kth feature, and B_{ijk} and $c_{\cdot k}$ refer to the kth features of the corresponding feature vectors.

The main difference between EC and DD can be easily seen from the difference between (1) and (5), where the EC value, which is defined only for the *observed* instances in $\mathfrak{B}$, can be directly computed from the training data, while DD tries to *maximize* an objective function, i.e., to search for the target point, over C which is a continuous space with infinitely many members. The flow charts for EC computation and DD are illustrated in Figure 1 and Figure 2, respectively. In Figure 1, the *direct computation* step is based on (1) without the need for any optimization procedure. In Figure 2, however, an *optimization procedure*, such as gradient ascent

2. This target point is not necessarily an observed instance in the training set $\mathfrak{B}$. We must search for it in the whole instance space which may be a continuous space containing infinitely many instances.

in [9], should be firstly applied to find the target point c_t by maximizing the objective function in (5). Then, based on c_t, a value, such as the distance between B_{gh} and c_t in [9], is computed by the *further computation* step for further processing.

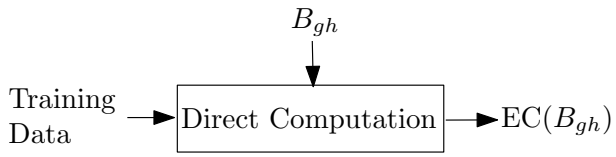

Fig. 1. Flow chart for the evidence confidence (EC) computation.

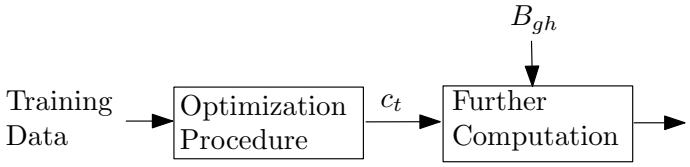

Fig. 2. Flow chart for the diverse density (DD) method.

From (5), it is not difficult to realize that the *optimization procedure* in DD is very sensitive to labeling noise. For example, if we mislabel just a single positive bag $\hat{B}$ as a negative bag, then $\Pr(c_t \mid \hat{B})$ computed based on (3) for the true target point c_t will decrease exponentially. As a result, the objective function value on c_t is likely to be very small. Hence, in this case, the computed target point will be relatively far away from the true target point. This problem has been validated by the experiments in [4], [5]. Moreover, the DD landscape typically contains local maxima. Searching for the true target point by applying gradient-ascent or EM does not guarantee global optimality. With no prior knowledge for a good initialization point, multiple restarts are generally needed and hence high computation cost is incurred.

Another difference between EC and DD comes from the difference between (4) and (6). In (4), all the features (indexed by k) have the same scaling parameter σ. In (6), each feature (k) has its specific scaling parameter s_k. The adoption of (4) for EC computation is motivated by MILES [5]. In MILES, they use the same scaling parameter for all the features, but the performance of MILES is still better than DD-SVM [4] which adopts a scaling vector to weight the features. The computation cost for the EC value based on (4) will be dramatically decreased

because we do not have to search through a huge space of possible scaling coefficient values. Moreover, the meaning of $\Pr(B_{gh} \mid B_{ij})$ in (4) is much more obvious, which is just a kernel density estimate with B_{ij}.

Our EC-based instance selection method *totally avoids* the two disadvantages of DD-based methods, which are high computation cost and high sensitivity to labeling noise. The advantages of our method are summarized as follows:

- The EC value of each *observed instance* is directly computed from the training set. The parameter m in Algorithm 1 can be obtained from prior knowledge or observed directly from the training data for image retrieval. In our experiments, we just set σ to 1 if the data are normalized, and the performance is still very promising. Hence, our method essentially has no parameters to tune, making it several orders of magnitude faster than DD-based methods.
- Because our instance selection method constrains the search scope to be within a bag, it is also very robust towards labeling noise. To illustrate this, let us assume that x^+ is a true positive instance from a positive training bag and x^- is a negative instance (false positive instance) from the same bag. Without labeling noise, most (or even all) terms, $\Pr(x^+|B_i^+)$ and $\Pr(x^+|B_i^-)$, should be expected to be larger than their counterparts, $\Pr(x^-|B_i^+)$ and $\Pr(x^-|B_i^-)$, in (1). Hence, $\mathrm{EC}(x^+)$ should be much larger than $\mathrm{EC}(x^-)$. Even if a portion of the training bags are mislabeled to make *some* terms, $\Pr(x^-|B_i^+)$ and $\Pr(x^-|B_i^-)$, larger than their counterparts, $\Pr(x^+|B_i^+)$ and $\Pr(x^+|B_i^-)$, $\mathrm{EC}(x^+)$ will still be larger than $\mathrm{EC}(x^-)$ as long as the number of these terms is not too large. Even if $\mathrm{EC}(x^+)$ will decrease in this case, x^+ will still be selected. This means that the instance selection result will not be affected because only the *relative EC values*, rather than the absolute EC values, for the instances in a specific bag will affect the result in Algorithm 1.

3.2 Feature Representation Scheme

Based on the selected instances, we propose a feature mapping to map every bag B_i to a point $\psi(B_i)$ in the instance based feature space:

$$\psi(B_i) = \left(d(e_1^*, B_i), d(e_2^*, B_i), \ldots, d(e_{|E^*|}^*, B_i)\right)^T, \tag{7}$$

where $e_k^* \in E^*$, E^* is the set of selected instances in Algorithm 1, and $d(e, B_i)$ is defined as follows:

$$d(e, B_i) = \min_{B_{ij} \in B_i} (\|e - B_{ij}\|), \tag{8}$$

which means that the distance between an instance and a bag is equal to the distance between the instance and the nearest instance in the bag.

This feature mapping is very meaningful because generally the distance between two evidence instances is expected to be smaller than the distance between one evidence instance and a non-evidence instance from the background. Because positive bags contain evidence instances, the distance from one evidence instance to a positive bag is expected to be smaller than the distance from this evidence instance to a negative bag. Because the selected instances in E^* are most likely to be evidence instances, the features in (7) are expected to have strong discrimination ability. Furthermore, for a specific bag, different instances in it will be selected as the nearest instances to compute the distance in (8) for different e_k^*. Hence, the feature vector in (7) actually implicitly contains the inter-dependency between the instances in a bag, the effectiveness of which has been validated by [31].

4 Single Instance Formulation for Localized CBIR

After the feature mapping defined in (7), the MIL problem is converted into a standard SIL problem and hence any conventional classification method can easily be adapted for MIL problems.

4.1 Supervised Formulation

In this paper, we adapt SVM for MIL because it can deliver promising generalization performance via margin maximization. The resulting method is called EC-SVM. Since SVM has become a mature technique which has been widely used in many applications, we do not introduce it in detail here. We refer the readers to the related literature, such as the book in [41] or LIBSVM [42] and its documentation.

4.2 Semi-Supervised Formulation

It is usually easy to get a large number of unlabeled images from the image repository. Hence, semi-supervised learning methods, which can incorporate unlabeled data into the training process,

are very meaningful for CBIR. Here, we propose to adapt *manifold regularization* [43] for semi-supervised localized CBIR. The resulting method is called EC-LapSVM.

The underlying assumption of manifold regularization is that if two points are close with respect to the intrinsic geometry of the marginal distribution, then their predicted values should be similar. Hence, manifold regularization can exploit the geometric structure of the marginal data distribution.

Suppose the training set contains l labeled bags $\{(B_i, y_i)\}_{i=1}^{l}$ and u unlabeled bags $\{B_j\}_{j=l+1}^{l+u}$. The goal of manifold regularization is to find a function f^* that minimizes the following objective function:

$$\min_{f \in \mathcal{H}_K} \frac{1}{l} \sum_{i=1}^{l} V(B_i, y_i, f) + \gamma_A \|f\|_K^2 + \frac{\gamma_I}{(l+u)^2} \mathbf{f}^T L \mathbf{f},$$

where $V(\cdot)$ is some loss function, such as the hinge loss function $\max(0, 1 - y_i f(B_i))$ for SVM, $\mathcal{H}_K$ is the reproducing kernel Hilbert space (RKHS) of function f corresponding to a Mercer kernel K, $\|f\|_K^2$ is the complexity of f in terms of the RKHS norm $\|\cdot\|_K$, $\mathbf{f} = [f(B_1), f(B_2), \ldots, f(B_{l+u})]^T$, γ_A and γ_I are the regularization parameters, and L is the graph Laplacian given by $L = D - W$. Here, W_{ij} is the edge weight between nodes B_i and B_j in the adjacency graph of the data, and D is a diagonal matrix with $D_{ii} = \sum_{j=1}^{l+u} W_{ij}$.

In this paper, we adopt the hinge loss for V and get the following Laplacian SVM (LapSVM) formulation:

$$\min_{f \in \mathcal{H}_K} \frac{1}{l} \sum_{i=1}^{l} (1 - y_i f(B_i))_+ + \gamma_A \|f\|_K^2 + \frac{\gamma_I}{(l+u)^2} \mathbf{f}^T L \mathbf{f}. \tag{9}$$

From the representer theorem [43], the solution for the optimization problem can be expressed in the following form:

$$f^*(B) = \sum_{i=1}^{l+u} \alpha_i K(B_i, B) + b.$$

For all the computation, W_{ij} is computed based on $\psi(B_i)$ and $\psi(B_j)$ defined in (7), and $K(B_i, B_j) = K(\psi(B_i), \psi(B_j))$. The detailed optimization procedure to solve the problem in (9) can be found in [43].

4.3 Relation to Existing Work

From the formulation point of view, EC "looks" similar to DD. However, EC's modification to DD makes EC-SVM ingeniously integrate the advantages of both DD-SVM and MILES,

and simultaneously overcome their shortcomings. From MILES, we can see that using the instances from the training bags to construct the feature representation is sufficient for good performance. Hence, the optimization procedure in DD for finding local optima, which is both time-consuming and noise sensitive, is unnecessary. From DD-SVM, we can see that the most discriminative features might be those constructed based on evidence instances. Hence, the features constructed based on negative instances, which are adopted by MILES, might be useless, or even harmful (cf. Figure 16 and the discussion). EC-SVM constructs only those discriminative features corresponding to those instances most likely to be evidence instances without any time-consuming optimization procedure. Hence, EC's modification to DD makes EC-SVM much more effective than DD-SVM and MILES.

To the best of our knowledge, there exist only three methods for semi-supervised localized CBIR. Two of them, [7], [8], are *transductive* and cannot be generalized easily to perform prediction on unseen test data. As stated in Section 1.1, these two transductive methods will not be suitable for practical applications. The other method is called semi-supervised MIL (SSMIL) [35], which can perform prediction on unseen test data. However, it must use MILES as a preprocessing method to perform feature selection before training the following classifier. Hence, it also possesses the shortcomings of MILES. Our semi-supervised method, EC-LapSVM, is *inductive*, and will achieve better performance than other methods (cf. Table 3).

5 Performance Evaluation

Note that the motivation of our paper is to design an effective feature representation scheme to describe the bags. Hence, among all the proposed MIL methods, DD-SVM and MILES are the most related ones. Moreover, DD-SVM and MILES have achieved performance comparable with other state-of-the-art MIL methods for many applications, including image classification and retrieval. Hence, DD-SVM and MILES are adopted as baselines for our supervised method (EC-SVM). As for our semi-supervised method (EC-LapSVM), SSMIL [35] and MISSL [7] are adopted as baselines.

For SVM, the Gaussian kernel $\kappa(x, y) = \exp^{-r\|x-y\|^2}$ is used for our method in all the experiments. We use LIBSVM [42] to train all the SVM classifiers. A MATLAB implementation

of our methods can be downloaded from our web site.[3]

5.1 Data Sets

We evaluate our methods based on two publicly available image data sets: the SIVAL (Spatially Independent, Variably Area, and Lighting) image set [2], [14] and the COREL image set [5].

5.1.1 SIVAL Data Set

The SIVAL data set contains 1,500 images of 25 categories, with 60 images for each category. These categories are complex objects photographed against 10 different highly diverse backgrounds. Six different images are taken for each object-background pair. For each object category, the same physical object is used in all scenes but the scenes are highly complex and diverse. The objects are photographed at different angles and they can appear at any location in the images. The target object occupies only about 10–15% of the whole image area in most images. Category 1 to category 25 are: "AjaxOrange", "Apple", "Banana", "BlueScrunge", "CandleWithHolder", "CardboardBox", "CheckeredScarf", "CokeCan", "DataMiningBook", "DirtyRunningShoe", "DirtyWorkGloves", "FabricSoftenerBox", "FeltFlowerRug", "GlazedWoodPot", "GoldMedal", "GreenTeaBox", "JuliesPot", "LargeSpoon", "RapBook", "SmileyFaceDoll", "SpriteCan", "StripedNoteBook", "TranslucentBowl", "WD40Can", and "WoodRollingPin". Figure 3 shows some sample images from the SIVAL image set. We use the same preprocessing method as that in [2], [7] to generate the bags. More specifically, we first use the IHS method [7] to segment each image into a specified number of regions. Then each region is described by 30 features. Let r_i denote a region. The first 6 features of r_i are the average color (in YCrCb color space) and texture values in r_i and the next 6 features denote the difference between the average color and texture values of the northern neighbor and those of r_i. Similarly, there are other 3×6 features for the difference between r_i and its other 3 neighbors. In [7] and this paper, we segment each image into 32 regions. Hence, each image is represented as a bag of 32 30-dimensional instances.

5.1.2 COREL Data Set

As in MILES [5], we choose 2,000 images from 20 (category 0 to category 19) COREL Photo CDs. Each CD contains 100 images representing a different category. The images are

3. http://www.cse.ust.hk/~liwujun/code/ecsvm.rar

Fig. 3. Sample images from the SIVAL image set.

in JPEG format with size 384×256 or 256×384. We use the same image segmentation and feature representation methods in MILES to construct the corresponding bags and instances. After segmentation, each region in an image is characterized by a 9-dimensional feature vector representing the color, texture and shape information from the region.

Figure 4 shows one sample image from each of the 20 categories. The categories are ordered in a row-wise manner from the upper-leftmost image (category 0) to the lower-rightmost image (category 19).

5.2 Illustration

In this subsection, we show some examples to demonstrate the effectiveness of our region selection algorithm in an intuitive way. We treat the images from Category 1 ("AjaxOrange") in SIVAL set as positive and the other images as negative. In each "AjaxOrange" image, only the plastic bottle with the text "AJAX" on its body supports the label of the image. The upper-leftmost image in Figure 5 shows such an example from Category 1.

First, we want to show that when the background of the negative images is similar to that of the positive images, evidence instances can be easily identified even if there are only one positive and one negative training images. Figure 5 illustrates such an example. In Figure 5, the first column corresponds to the positive training image, the second column corresponds to the negative training image, and the third column shows the five selected regions with the largest EC values. In the first two columns, the above row shows the original training images, and the

Fig. 4. Sample images from the 20 categories of the COREL image set.

second row shows the segmentation results of the corresponding images in the above row. The segmented regions are shown in their representative colors. The numbers in the third column show the ranks of the EC values for the corresponding regions. More specifically, the region pointed by the number i has the ith largest EC value. We can find that all the five selected regions with the largest EC values are located on the target object.

Second, we want to show that our method will fail when there are only one positive and one negative training images and both of them have totally different background. Figure 6 shows such an example, where the meanings of the images and numbers are the same as those in Figure 5. Because all the regions, including those from the background, in the positive image are far away from the regions in the negative image, we cannot correctly identify the evidence regions. This can also be seen from the five selected regions with the largest EC values. However, it should be noted that under this setting all existing methods, even people, will fail without any further information.

Third, we want to show that our method can successfully select evidence regions with two or more positive training images. Figure 7 shows an example with two positive and two negative

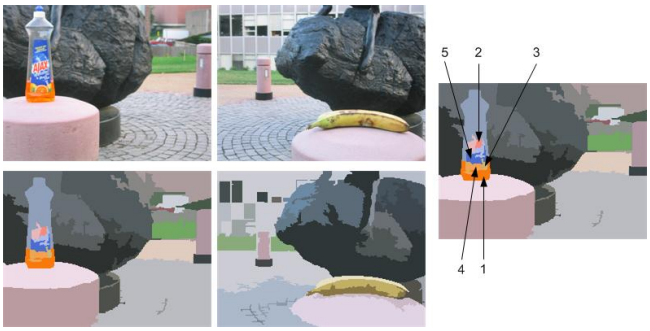

Fig. 5. Selected regions when there are only one positive and one negative training images, both of which have similar background.

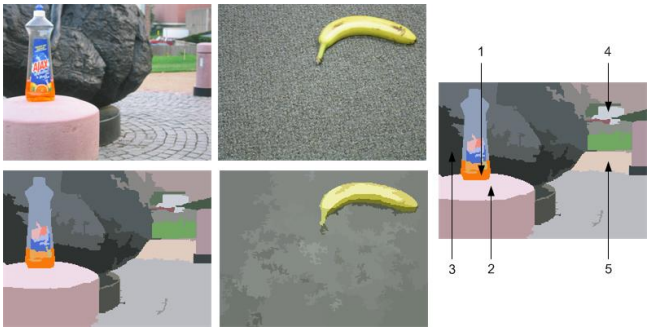

Fig. 6. Selected regions when there are only one positive and one negative training images, both of which have totally different background.

images. When there are more training images, the problem will typically become easier, which will not be discussed here. In Figure 7, the first two rows show the training images, with the first two columns for positive images and the last two columns for negative images. The selected regions are shown in the last row. We can find that even if the background of the positive and negative images is totally different, most of the selected instances with the largest EC values are still located on the target object. One exception is the fifth selected region in the second positive image. Actually, it is easy to understand this result. Because color values are extracted as features for SIVAL set and the color of the fifth selected region in the second positive image is very similar to that of some regions on the target object, this selected region is very close to the positive regions.

From the above results illustrated in Figure 5-7, we can see that the EC value is a very

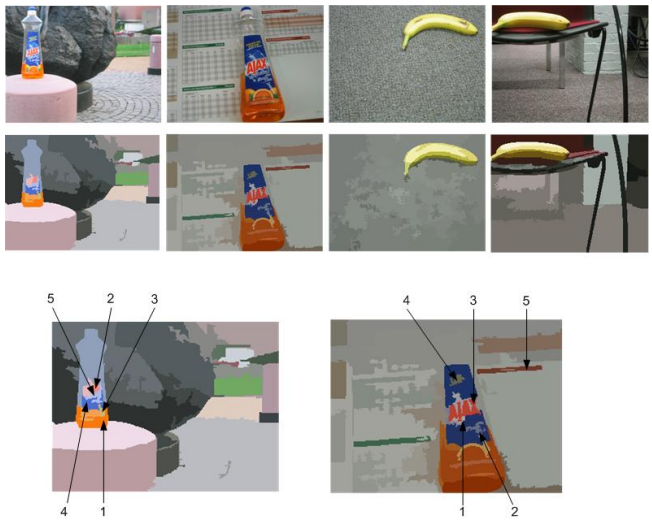

Fig. 7. Selected regions when there are two positive and two negative training images.

informative measure for instance selection. Hence, our instance selection method in Algorithm 1 can indeed select those instances which are most likely to be evidence instances.

5.3 Verification of the Necessity of Region Selection

One natural question to ask is why we must select part of the regions for feature construction in Section 3.2. Here, we will verify that region selection is very important for localized CBIR. We study the necessity of region selection on the SIVAL image set. We treat the 60 images of Category 1 ("AjaxOrange") as positive and the other 1,440 images as negative. We randomly select $n \in \{1, 2, 4, 8\}$ positive and n negative images to form the training set and the remaining $1,500-2n$ images to form the test set. The results are reported based on 30 rounds of independent test. Average AUC (area under the ROC curve) values (in percent) with 95% confidence interval of EC-SVM over 30 rounds of test are shown in Figure 8, where the method *WithoutSelection* means that we do not perform region selection and use all the regions from the positive training bags for feature construction, and the method *WithSelection* means that we use Algorithm 1 to select $m = 5$ regions from each positive training bag and use the selected regions for feature construction. The other settings for both *WithoutSelection* and *WithSelection* are the same. It is obvious that region selection plays a key role for localized CBIR, and our EC-based region selection method is very effective to select those regions with strong discrimination ability.

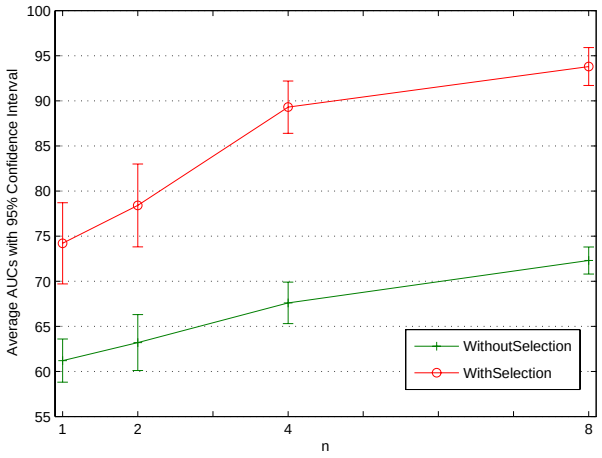

Fig. 8. Comparison between the method with region selection and that without region selection. Average AUC values (in percent) with 95% confidence interval over 30 rounds of test are reported.

Because in *WithoutSelection* all the instances, including evidence and non-evidence instances, are used for feature construction, the instance based feature mapping in (7) actually captures all the information in the images, including information from evidence regions and that from background regions (non-evidence regions). Hence, the method *WithoutSelection* can be seen as a global method. The worse performance of *WithoutSelection* verifies that the global methods are not suitable for localized CBIR.

Furthermore, the poor performance of *WithoutSelection* also implies that the universal bag-of-visual-words (BOV) representation [44] might not be suitable for the SIVAL-like image sets, because BOV based methods randomly select some local regions (either positive or negative) from some images (either positive or negative) to build a vocabulary. For the SIVAL-like image sets, in which the main part of an image is the background, most words in the built vocabulary will be from the background. Because the background can appear in either positive or negative images, the information in the histogram for an image will be dominated by the background information. Our EC-based region selection method might provide a very effective way to construct informative vocabulary for general object categorization [45]. Further study of our method for general object categorization is beyond the scope of this paper, and it will be pursued in our future work. The principle underlying the feature construction method in [39]

is very similar to that of BOV. Hence, the method in [39], which also adopts the non-evidence regions for feature construction, cannot be expected to achieve performance as good as EC-SVM for localized CBIR.

5.4 Sensitivity to the Number of Selected Regions

First, we study the effect of m in Algorithm 1 on the SIVAL image set. From Figure 7, we can find that Category 1 ("AjaxOrange") and Category 3 ("Banana") have different characteristics. More specifically, an image from Category 1 typically contain more positive regions than an image from Category 3. We use these two different cases to demonstrate that our method is robust to m for wide applications. For the first case, we treat the 60 images of Category 1 as positive and the other 1,440 images as negative. For the second case, we treat the 60 images of Category 3 as positive and the other 1,440 images as negative. For both cases, eight positive and eight negative images are randomly selected to form the training set and the remaining 1,484 images are used to form the test set. Average AUC values with 95% confidence interval of EC-SVM over 30 rounds of independent test against m are shown in Figure 9. We can see that there is no significant difference among the performances for different values of m when $2 \leq m \leq 10$, either for the case containing complex object ("AjaxOrange") or for the case containing simple object ("Banana").

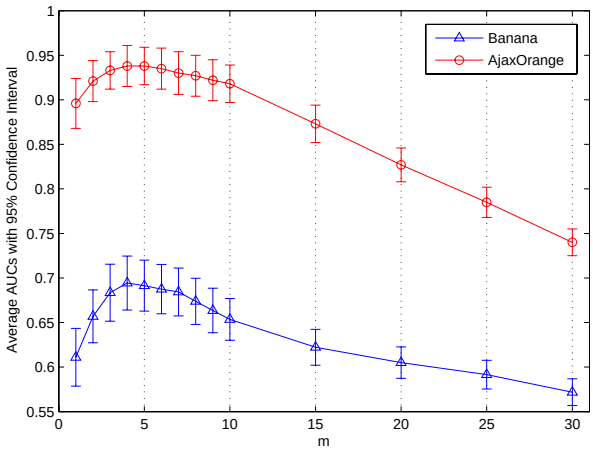

Fig. 9. Average AUC values with 95% confidence interval on the SIVAL data set against the number (m) of instances selected from each positive bag.

Second, we further study the effect of m in Algorithm 1 on the COREL image set. Note that the number of regions per image in COREL set is far less than that in SIVAL set. Most images in COREL set have less than five regions. Similar to the difference between "AjaxOrange" and "Banana" from SIVAL set, Category 1 ("Beach") and Category 14 ("Cars") from COREL set also have different characteristics. Typically, most or even all regions in a beach image are positive, but only one or two regions in a car image are positive. For the first case, we treat the 100 images of Category 1 as positive and the other 1,900 images as negative. For the second case, we treat the 100 images of Category 14 as positive and the other 1,900 images as negative. For both cases, four positive and four negative images are randomly selected to form the training set and the remaining 1,992 images are used to form the test set. Average AUC values with 95% confidence interval of EC-SVM over 50 rounds of independent test against m are shown in Table 1, where "All" means that all the regions from positive images are selected. Note that the number of regions per image in "Beach" is 3.54 and that in "Cars" is 4.93. When the number of regions in an image is less than m, all the regions from that image are selected by Algorithm 1. We can see that there is no significant difference among the performances for different values of m when $2 \leq m \leq 5$, either for the case containing complex object ("Beach") or for the case containing simple object ("Cars"). For the car images, the number of positive regions in one image is typically one or two. From Table 1, we find that even if we select some negative regions in addition to the positive ones, the performance will not be deteriorated as long as the number of selected negative regions is not too large. This also conforms to the results in Figure 9. Of course, the selected negative regions should be those with relatively large EC values. Furthermore, it is also interesting to find that even if all the regions are used for feature construction, the performance on COREL is still satisfactory. This is because in most images from COREL the target objects occupy a large portion of the whole images. Hence, to perform localized CBIR on COREL data set is much easier than that on SIVAL set. Although the performance improvement by using region selection on COREL set is less obvious than that on SIVAL set, we can still find that it is necessary to perform region selection for COREL set. In real applications, we are more likely to handle images like those from SIVAL set.

From the above experiments on both SIVAL and COREL data sets, we find that our method is robust to m. Hence, it is reasonable to set m in Algorithm 1 simply according to the prior knowledge, and it is unnecessary to design a complicated algorithm such as cross-validation to

TABLE 1

Average AUC values (in percent) with 95% confidence interval on the COREL data set against the number (m) of instances selected from each positive bag (the best performance is shown in bold).

m	1	2	3	4	5	6	All
Beach	77.34 ± 2.45	79.39 ± 2.16	80.30 ± 1.96	80.28 ± 1.86	**80.38 ± 1.90**	80.33 ± 1.90	77.75 ± 1.95
Cars	78.74 ± 1.56	**80.54 ± 1.19**	80.53 ± 1.28	80.32 ± 1.35	79.81 ± 1.38	79.49 ± 1.45	78.55 ± 1.63

learn m from the data. For example, for the SIVAL image set, we can simply set the parameter $m = 5 \simeq (32 \times 15\%)$ because the target object occupies about 15% of the image area for most images. The effectiveness of this strategy can also be seen from Figure 9.

From Figure 9, we can see that containing a small portion of non-evidence regions for feature construction will not affect the performance significantly. However, using too many non-evidence regions for feature construction will deteriorate the accuracy, which once again confirms that the BOV methods [44] and the method in [39] are not suitable for localized CBIR. This is in line with the conclusion in Section 5.3.

5.5 Performance of the Supervised Method

In this section, we evaluate EC-SVM on both the SIVAL and COREL data sets.

5.5.1 Evaluation on SIVAL Data Set

We adopt the same experimental settings as those used by related methods, such as [1]. For each category, we use the "one-versus-the-rest" strategy to evaluate the performance. We randomly select eight positive and eight negative images to form the training set and the remaining 1,484 images to form the test set. Unless otherwise mentioned, the results are reported based on 30 rounds of independent test. Because the target object occupies about 15% of the image area for most images, we simply set the parameter m in Algorithm 1 to 5 which is about $15.6\%\,(5/32)$ of the number of all instances in a bag. For EC-SVM, the parameter C and Gaussian kernel parameter r for SVM in LIBSVM [42] are simply set to 1 and 2^{-4} respectively.

Better performance can be expected if a more sophisticated method, such as cross-validation on the training data, is used to set these parameters. For the parameters in MILES [5], we find that $\lambda = 0.2$ and $\sigma^2 = 1$ give the best *test performance* for the SIVAL data set. Hence, we fix $\lambda = 0.2$ and $\sigma^2 = 1$ for MILES in all the following experiments on the SIVAL set. For DD-SVM, we also choose the parameter setting that achieves the best test performance. The average AUC values with 95% confidence interval for the 25 categories are reported in Table 2. We can see that EC-SVM achieves the best performance for most categories.

TABLE 2

Average AUC values (in percent) with 95% confidence interval over 30 rounds of test on the SIVAL image set (the best performance is shown in bold).

Category ID	EC-SVM	MILES	DD-SVM
1	**93.8 ± 2.1**	90.2 ± 2.3	84.1 ± 3.2
2	**68.0 ± 2.6**	64.5 ± 2.5	62.8 ± 2.3
3	**69.1 ± 2.9**	68.1 ± 3.1	62.2 ± 1.6
4	**74.1 ± 2.4**	72.6 ± 2.5	62.1 ± 2.9
5	**88.1 ± 1.1**	84.0 ± 2.3	77.3 ± 2.8
6	**85.6 ± 1.6**	81.2 ± 2.7	73.0 ± 3.0
7	**96.9 ± 0.5**	93.7 ± 1.2	96.2 ± 0.7
8	**94.6 ± 0.8**	92.4 ± 0.8	94.0 ± 0.9
9	**75.0 ± 2.4**	71.1 ± 3.2	68.8 ± 3.7
10	**90.3 ± 1.3**	85.3 ± 1.7	87.3 ± 1.4
11	**83.0 ± 1.3**	77.1 ± 3.1	72.3 ± 2.2
12	**97.9 ± 0.5**	97.1 ± 0.7	95.7 ± 1.8
13	**94.2 ± 0.8**	93.9 ± 0.7	91.4 ± 0.7
14	68.0 ± 2.8	**68.2 ± 3.1**	**68.2 ± 3.4**
15	**87.5 ± 1.4**	80.7 ± 2.9	73.4 ± 4.1
16	86.9 ± 2.2	**91.2 ± 1.7**	86.9 ± 3.1
17	67.3 ± 3.3	**78.7 ± 2.9**	74.3 ± 3.0
18	**61.3 ± 1.8**	58.2 ± 1.6	59.7 ± 1.8
19	**68.6 ± 2.3**	61.7 ± 2.4	66.2 ± 2.0
20	**84.6 ± 1.9**	77.5 ± 2.6	69.3 ± 3.9
21	**85.4±1.2**	80.4 ± 2.0	81.1 ± 2.4
22	**75.6 ± 2.3**	68.7 ± 2.4	67.3 ± 3.0
23	**74.2 ± 3.2**	73.2 ± 3.1	67.3 ± 2.7
24	**94.3 ± 0.6**	88.1 ± 2.2	86.3 ± 2.6
25	**66.9 ± 1.7**	62.1 ± 2.5	64.8 ± 1.4
Average	**81.3**	78.4	75.7

We further test EC-SVM by varying the size of the training set. The average AUC values for all 25 categories over 30 rounds of test, together with the results of DD-SVM and MILES, are shown in Figure 10, in which n denotes the number of training images for each class. For example, the number "1" refers to the case in which one positive image and one negative image are selected for training and all other images for testing. We can see that EC-SVM achieves the best performance for all cases.

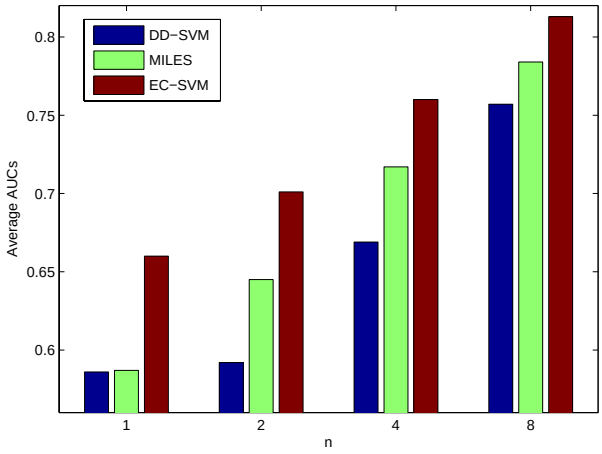

Fig. 10. Average AUC values for all 25 categories over 30 rounds of test on the SIVAL image set.

Accio!, a localized CBIR system introduced in [1], achieves an average AUC value 0.818 for all 25 categories when $n = 8$, which is slightly better than the accuracy of EC-SVM (0.813). However, Accio! is a DD-based method, which is sensitive to noise and highly time-consuming for training. This will be shown in the following experiments. Actually, the accuracy of EC-SVM can be considered to be comparable with the accuracy of Accio!.

5.5.2 Evaluation on COREL Data Set

Because DD-SVM and MILES have achieved better results than many other methods [5], including global methods and local methods, on the COREL data set, we do not report the results of other methods here. As in [5], we choose λ from 0.1 to 0.6 with step size 0.05 and σ^2 from 5 to 15 with step size 1. We find that $\lambda = 0.2$ and $\sigma^2 = 11$ give the best *test performance* for MILES on the COREL data set. Hence, we fix $\lambda = 0.2$ and $\sigma^2 = 11$ for MILES in all the

following experiments on this data set. For DD-SVM, we also choose the parameter setting that achieves the best test performance.

For each category, we use the "one-versus-the-rest" strategy to evaluate the performance. In each round, $n \in \{1, 2, 4\}$ randomly selected positive images and n randomly selected negative images are chosen to form the training set and the remaining $2,000 - 2n$ images to form the test set. The results are reported based on 50 rounds of independent test. Although the target objects in different categories, or the target objects from the same category but in different images, may be partitioned into different number of regions, we just simply set the parameter m in Algorithm 1 to 3. For EC-SVM, the parameter C and Gaussian kernel parameter r for SVM in LIBSVM [42] are set to 1 and 2^{-3} respectively. Figure 11 shows the results of the average AUC values for all 20 categories over 50 rounds of test. Once again, EC-SVM achieves the best performance for all the test cases.

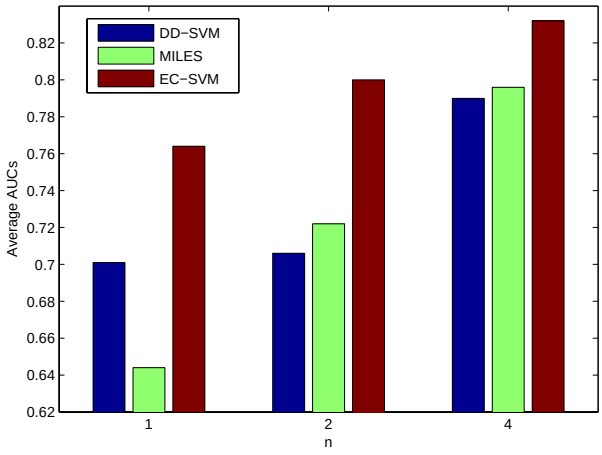

Fig. 11. Average AUC values for all 20 categories over 50 rounds of test on the COREL image set.

Figure 12 and Figure 13 show the average AUC values with 95% confidence interval for each category when $n = 1$ and $n = 2$, respectively. The numbers in the X-axis indicate the category IDs. We can see that EC-SVM achieves the best performance on most categories.

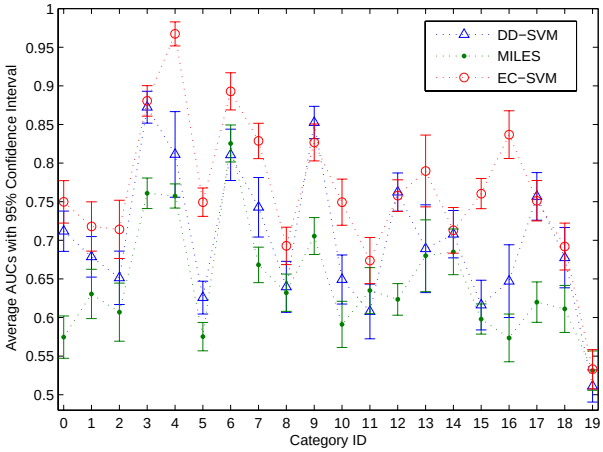

Fig. 12. Comparison on the COREL data set with one positive and one negative examples labeled.

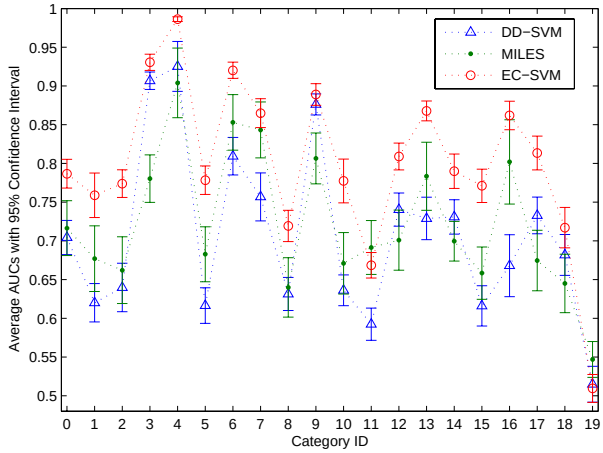

Fig. 13. Comparison on the COREL data set with two positive and two negative examples labeled.

5.6 Performance of Manifold Regularization

MISSL [7] and SSMIL [35] are the most related methods of EC-LapSVM. Hence, we choose them here as the *baselines* for comparison. We use the same data set, SIVAL, and the same experimental settings as those in MISSL and SSMIL to evaluate our method. In MISSL and SSMIL, for every single run, the labeled data contain eight randomly selected positive images

and eight randomly selected negative images, and all the remaining 1,484 images are used as unlabeled data. The test performance is evaluated on the unlabeled data.

Table 3 shows the average AUC values for all 25 categories over 30 rounds of test on the SIVAL data set. We can see that our supervised method, EC-SVM, achieves better results than MISSL and SSMIL even if EC-SVM does not use unlabeled data for training, which indicates the effectiveness of our feature representation scheme. In addition, by incorporating unlabeled data into the training process, our semi-supervised method, EC-LapSVM, can further improve the performance. To the best of our knowledge, the performance of EC-LapSVM (83.1%) is the best result on the SIVAL data set, compared with any learning method with any feature representation scheme.

TABLE 3
Average AUC values (in percent) for all 25 categories over 30 rounds of test on the SIVAL data set.

	EC-SVM	MISSL [7]	SSMIL [35]	**EC-LapSVM**
AUC	81.3	74.8	80.6	**83.1**

Figure 14 shows in detail the average AUC values for each class over 30 rounds of test. The numbers in the X-axis indicate the category IDs. We can see that EC-LapSVM achieves better results than all other methods for most categories.

5.7 Sensitivity to Labeling Noise

We use the same setting as that in MILES [5] to evaluate the noise sensitivity on the COREL data set. We add $d\%$ of noise by changing the labels of $d\%$ of positive bags and $d\%$ of negative bags. We compare EC-SVM with DD-SVM and MILES under different noise levels based on 200 images from Category 2 (Historical buildings) and Category 7 (Horses). The training and test sets are of the same size. The average classification accuracy over five randomly generated test sets is shown in Figure 15. We can see that MILES and EC-SVM are much more robust than DD-SVM, and the robustness of EC-SVM is comparable with MILES.

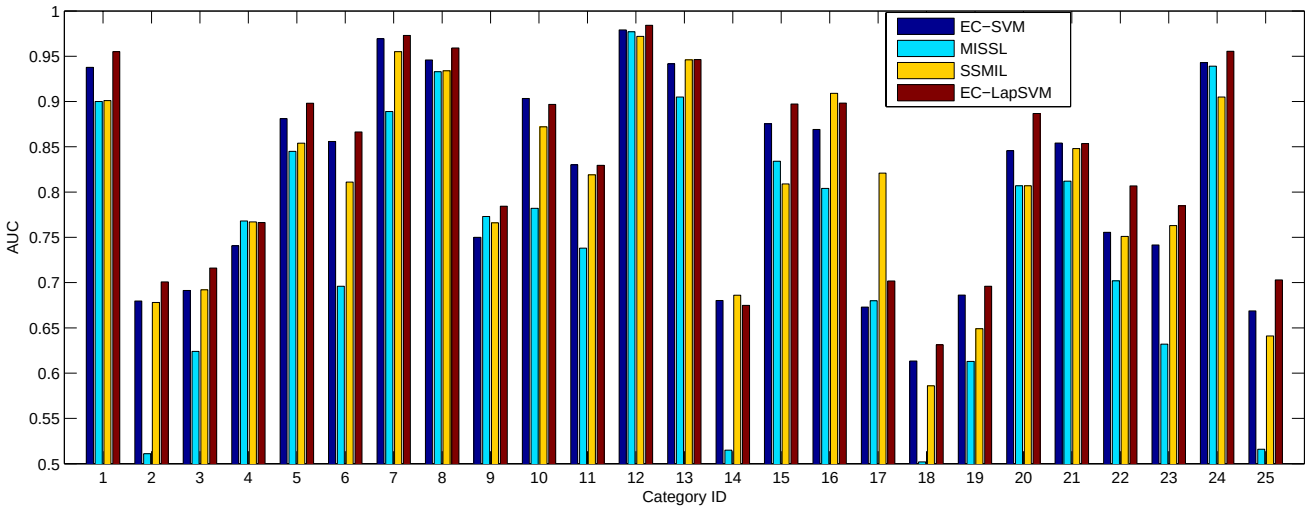

Fig. 14. Comparison of our semi-supervised method, EC-LapSVM, with other methods on the SIVAL image set.

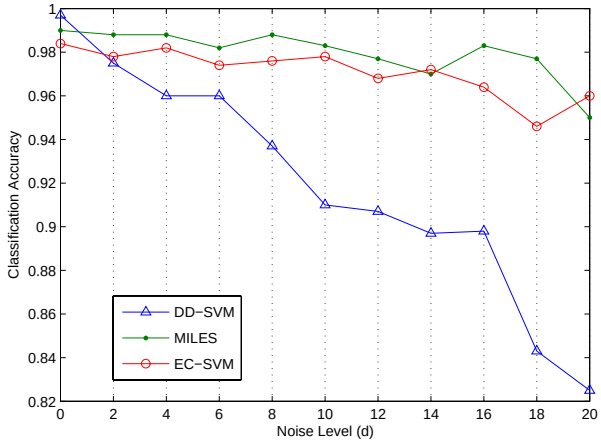

Fig. 15. Comparison of sensitivity to labeling noise on the COREL data set.

We further test the noise sensitivity of EC-SVM on the SIVAL data set. We compare EC-SVM with DD-SVM and MILES under different noise levels $(n/30, n = 1, \ldots, 9)$, by negating the labels of n positive and n negative training images, based on 120 images from Category 7 (CheckeredScarf) and Category 12 (FabricSoftenerBox). The training and test sets are of the same size. The average classification accuracy with 95% confidence interval over 30 randomly

generated test sets is shown in Figure 16. We can see that EC-SVM is much more robust than DD-SVM and MILES on the SIVAL data set.

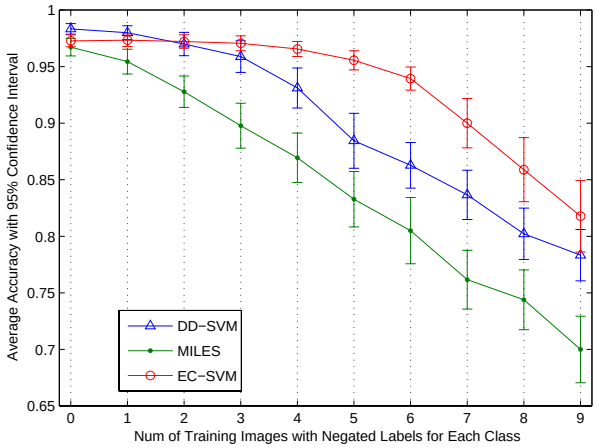

Fig. 16. Comparison of sensitivity to labeling noise on the SIVAL data set.

The SIVAL data set differs from the COREL data set in many aspects. In COREL, the target object occupies a large portion of the whole image, while in SIVAL the main part of an image is the background. Furthermore, the background of some category in COREL is always specific to that category of images. For example, in general, the background in the images of "Historical building" is very different from the background in the images of "Horses". But for SIVAL, the background for one category can appear for another category. MILES uses all the instances, from both positive training bags and negative training bags, as the basis for feature extraction [5]. This will make the effect of instances from the background dominate the effect of the evidence instances on SIVAL. Because the background can appear in either positive or negative bags, the features based on instances from the background actually have very low discrimination ability. Hence, the useful features in MILES are very limited. As a result, MILES will be more easily affected by noise on the SIVAL data set. This might be the cause for the phenomenon that MILES is much more sensitive to noise on the SIVAL data set.

Note that in Figure 16, the number of selected regions m is set to 5. To further test the robustness of EC-SVM, we change the value of m for evaluation. Other settings are the same as those for Figure 16. The results are shown in Figure 17, from which we can see that EC-SVM is robust to labeling noise when m takes different values.

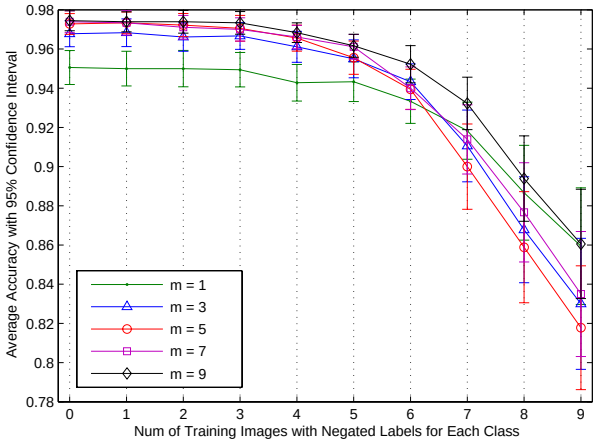

Fig. 17. Comparison of sensitivity to labeling noise when m takes different values in EC-SVM.

5.8 Computation Cost

Table 4 lists the training time (on a 2.4GHz Linux server) required by ACCIO! [1], DD-SVM, MILES, and EC-SVM. "SIVAL" refers to the time for training 25 classifiers for all the 25 categories when four positive and four negative images are used as the training set on the SIVAL data set. "COREL" refers to the time for training 20 classifiers for all the 20 categories when four positive and four negative images are used as the training set on the COREL data set. We can see that MILES is more than one order of magnitude faster than ACCIO! and DD-SVM, and EC-SVM is much more efficient than MILES. Although ACCIO! has achieved accuracy comparable with EC-SVM, the far lower speed of training makes it not as practical as EC-SVM. Actually, EC-SVM only takes about 0.009 seconds to train one classifier for the COREL data set, which is almost real time. Hence, we can say that EC-SVM is a very practical method for localized CBIR.

6 Conclusion

The difference between the formulation of MIL and that of SIL makes MIL very difficult to solve because traditional SIL methods cannot be easily adapted for MIL. In this paper, a novel feature representation scheme for bags is proposed to convert MIL into SIL, which makes it

TABLE 4
Computation time comparison (in minutes).

	SIVAL	COREL
ACCIO!	7.074	1.300
DD-SVM	7.270	1.322
MILES	0.242	0.073
EC-SVM	**0.090**	**0.003**

convenient to adapt sophisticated SIL methods for MIL. Some promising properties of our feature representation method are highlighted as follows:

- It achieves higher accuracy than other state-of-the-art methods.
- It is very computationally efficient and robust against labeling noise.
- It is simple and easily implementable.

Considering the high computation cost and high noise sensitivity of DD-SVM, and the very high dimensionality of the feature vectors used by MILES, the feature representation scheme proposed in this paper is a much more practical one to effectively describe the bags in MIL, which makes localized CBIR practical in real applications.

As we have said in Section 5.3, the feature representation scheme in this paper might provide a more effective way than BOV [44] for general object categorization. This will be pursued in our future work.

ACKNOWLEDGMENT

This research has been supported by General Research Fund 621407 from the Research Grants Council of the Hong Kong Special Administrative Region, China. We thank Dr. Yixin Chen for sharing the code and data for DD-SVM and MILES.

REFERENCES

[1] R. Rahmani, S. A. Goldman, H. Zhang, S. R. Cholleti, and J. E. Fritts, "Localized content-based image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1902–1912, 2008.

[2] R. Rahmani, S. A. Goldman, H. Zhang, J. Krettek, and J. E. Fritts, "Localized content based image retrieval," in *Multimedia Information Retrieval*, 2005, pp. 227–236.

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

[4] Y. Chen and J. Z. Wang, "Image categorization by learning and reasoning with regions." *Journal of Machine Learning Research*, vol. 5, pp. 913–939, 2004.

[5] Y. Chen, J. Bi, and J. Z. Wang, "MILES: Multiple-instance learning via embedded instance selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 1931–1947, 2006.

[6] X. Zhu and A. B. Goldberg, *Introduction to Semi-Supervised Learning*. Morgan & Claypool, 2009.

[7] R. Rahmani and S. A. Goldman, "MISSL: multiple-instance semi-supervised learning," in *Proceedings of the Twenty-Third International Conference Machine Learning*, 2006, pp. 705–712.

[8] J. Tang, X.-S. Hua, G.-J. Qi, and X. Wu, "Typicality ranking via semi-supervised multiple-instance learning," in *ACM Multimedia*, 2007, pp. 297–300.

[9] O. Maron and T. Lozano-Pérez, "A framework for multiple-instance learning." in *Advances in Neural Information Processing Systems*, 1997.

[10] O. Maron and A. L. Ratan, "Multiple-instance learning for natural scene classification." in *Proceedings of the International Conference Machine Learning*, 1998, pp. 341–349.

[11] Q. Zhang and S. A. Goldman, "EM-DD: An improved multiple-instance learning technique." in *Advances in Neural Information Processing Systems*, 2001, pp. 1073–1080.

[12] Q. Zhang, S. A. Goldman, W. Yu, and J. Fritts, "Content-based image retrieval using multiple-instance learning." in *Proceedings of the International Conference Machine Learning*, 2002, pp. 682–689.

[13] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–38, 1977.

[14] H. Zhang, R. Rahmani, S. R. Cholleti, and S. A. Goldman, "Local image representations using pruned salient points with applications to CBIR," in *ACM Multimedia*, 2006, pp. 287–296.

[15] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning." in *Advances in Neural Information Processing Systems*, 2002, pp. 561–568.

[16] P. V. Gehler and O. Chapelle, "Deterministic annealing for multiple-instance learning." in *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics*, 2007.

[17] T. Gärtner, P. A. Flach, A. Kowalczyk, and A. J. Smola, "Multi-instance kernels." in *Proceedings of the International Conference on Machine Learning*, 2002, pp. 179–186.

[18] J. T. Kwok and P.-M. Cheung, "Marginalized multi-instance kernels." in *International Joint Conferences on Artificial Intelligence*, 2007.

[19] 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, 2008.

[20] Z.-H. Zhou, Y.-Y. Sun, and Y.-F. Li, "Multi-instance learning by treating instances as non-i.i.d. samples," in *Proceedings of the International Conference on Machine Learning*, 2009, p. 157.

[21] C. Yang, M. Dong, and J. Hua, "Region-based image annotation using asymmetrical support vector machine-based multiple-instance learning." in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006, pp. 2057–2063.

[22] P.-M. Cheung and J. T. Kwok, "A regularization framework for multiple-instance learning." in *Proceedings of the International Conference on Machine Learning*, 2006, pp. 193–200.

[23] 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*, 2006, pp. 1609–1616.

[24] S. R. Cholleti, S. A. Goldman, and R. Rahmani, "Mi-Winnow: A new multiple-instance learning algorithm," in *18th IEEE International Conference on Tools with Artificial Intelligence*, 2006, pp. 336–346.

[25] P. A. Viola, J. C. Platt, and C. Zhang, "Multiple instance boosting for object detection," in *Advances in Neural Information Processing Systems*, 2005.

[26] C. Zhang and P. A. Viola, "Multiple-instance pruning for learning efficient cascade detectors," in *Advances in Neural Information Processing Systems*, 2007.

[27] B. Babenko, M.-H. Yang, and S. Belongie, "Visual Tracking with Online Multiple Instance Learning," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2009.

[28] R. C. Bunescu and R. J. Mooney, "Multiple instance learning for sparse positive bags." in *Proceedings of the International Conference on Machine Learning*, 2007.

[29] S. Vijayanarasimhan and K. Grauman, "Keywords to visual categories: Multiple-instance learning forweakly supervised object categorization," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008.

[30] Z.-H. Zhou and J.-M. Xu, "On the relation between multi-instance learning and semi-supervised learning." in *Proceedings of the International Conference on Machine Learning*, 2007.

[31] G.-J. Qi, X.-S. Hua, Y. Rui, T. Mei, J. Tang, and H.-J. Zhang, "Concurrent multiple instance learning for image categorization," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007.

[32] H.-Y. Wang, Q. Yang, and H. Zha, "Adaptive p-posterior mixture-model kernels for multiple instance learning," in *Proceedings of the International Conference on Machine Learning*, 2008, pp. 1136–1143.

[33] V. C. Raykar, B. Krishnapuram, J. Bi, M. Dundar, and R. B. Rao, "Bayesian multiple instance learning: automatic feature selection and inductive transfer," in *Proceedings of the International Conference on Machine Learning*, 2008, pp. 808–815.

[34] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

[35] D. Zhang, Z. Shi, Y. Song, and C. Zhang, "Localized content-based image retrieval using semi-supervised multiple instance learning," in *Asian Conference on Computer Vision (1)*, 2007, pp. 180–188.

[36] B. Settles, M. Craven, and S. Ray, "Multiple-instance active learning," in *Advances in Neural Information Processing Systems*, 2007.

[37] S. Vijayanarasimhan and K. Grauman, "Multi-level active prediction of useful image annotations for recognition," in *Advances in Neural Information Processing Systems*, 2008, pp. 1705–1712.

[38] D. Zhang, F. Wang, Z. shi, and C. Zhang, "Interactive localized content based image retrieval with multiple-instance active learning," *Pattern Recognition*, 2009.

[39] Z.-H. Zhou and M.-L. Zhang, "Solving multi-instance problems with classifier ensemble based on constructive clustering," *Knowl. Inf. Syst.*, vol. 11, no. 2, pp. 155–170, 2007.

[40] J. Bi, Y. Chen, and J. Z. Wang, "A sparse support vector machine approach to region-based image categorization." in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005, pp. 1121–1128.

[41] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000.

[42] C.-C. Chang and C.-J. Lin, *LIBSVM: a Library for Support Vector Machines*, 2001, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[43] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of Machine Learning Research*, vol. 7, pp. 2399–2434, 2006.

[44] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *International Conference on Computer Vision*, 2003, pp. 1470–1477.

[45] S. J. Dickinson, A. Leonardis, B. Schiele, and M. J. Tarr, *Object Categorization: Computer and Human Vision Perspectives*. Cambridge University Press, 2009.