Image Set Classification using Multi-Layer Multiple Instance Learning with Application to Cannabis Website Classification

Nianhua Xie*,†, Haibin Ling† and Weiming Hu*
*Institute of Automation, Chinese Academy of Sciences, Beijing, China
†Dept. of Computer and Information Science, Temple University, Philadelphia, USA
Email: {nhxie,wmhu}@nlpr.ia.ac.cn, {hbling}@temple.edu

***Abstract*—We propose using multi-layer multiple instance learning (MMIL) for image set classification and applying it to the task of cannabis website classification. We treat each image as an instance in an image set, then each image is further viewed as containing instances of local image patches. This representation naturally extends traditional multiple instance learning (MIL) to multi-layers. We then show that, when using the *set kernels* for all layers, an MMIL problem can be flattened to a simple one-layer MIL. This flattening, when combined with quantized local image patch representation, drastically improves the computational efficiency by two orders. The flattened set kernel is further improved by weighted codewords and an exponential kernel. The proposed approach is applied to a cannabis website classification task, in which we collected a dataset containing more than 220,000 images from 600 websites. In the experiments our approach compares favorably with several state-of-the-art methods.**

***Index Terms*—Cannabis Website Classification; Image Set Classification; Bag-of-Words**

I. Introduction

Image-based object and category classification have been attracting a large amount of research effort in computer vision community. In comparison, there is little study of image set classification, which has many potential applications especially considering the fast growing of visual information available around the world. For example, we can use image set comparison to classify websites and webpages. Such a comparison can either work independently and therefore achieve language independence, or be combined with text based solutions. Another example is for video classification using on key-frame sets. One more application of image set classification is personal photo collection analysis that is useful in human computer interaction, especially for designing intelligent image browsing techniques. One advantage of image set classification is that only one label is required to label a whole image set; this largely reduces the tedious annotation work involved in many image-based tasks.

Early image classification and retrieval systems often represent an image with global features such as color and/or texture statistics [17], [2]. Then the task is solved by traditional supervised learning approaches since each image has a class label (Figure 1(a)). Recently, it becomes popular to represent images as groups of local patches [18], [3], [29], [24]. If we think of each local patch as an instance in the image

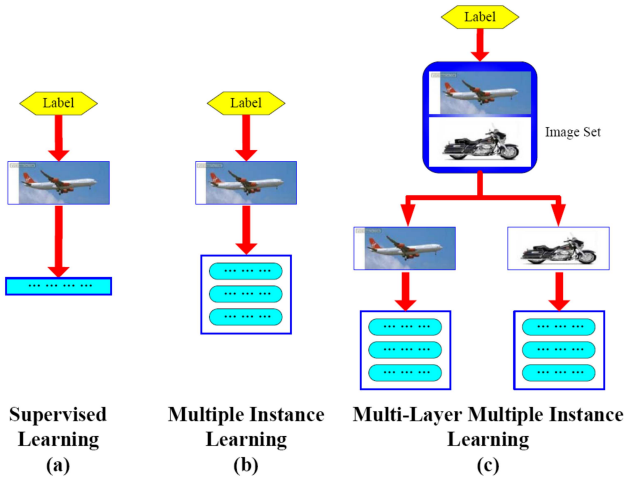

Fig. 1. Comparison of two paradigms of image classification and one model of image set classification.

(Figure 1(b)), image classification can then be viewed as a *multiple instance learning* (MIL) problem [4], which has recently been studied by many researchers from both machine learning and computer vision communities (see Section II). In image set classification tasks, MIL is not directly applicable, since labels are only available for whole image sets but not for individual images (Figure 1(c)). However, by treating images as instances of an image set, the task can be naturally addressed by *multi-layer multiple instance learning* (MMIL), which will be studied in this paper. A natural solution to MMIL for image set classification is to extend the bag-of-words to two layers, which requires building one codebook for image patches and another for images. However, the accumulated inaccuracy during the two quantization processes can be very large and therefore hurt the performance.

To solve this problem, we propose to extend the *set kernel* [15], [16] to multi-level. In such a *multi-layer set kernel*, a kernel at a higher level is computed recursively as a sum of kernels between member instances, which are sets by themselves. We show that this multi-layer set kernel can be flattened to a one-layer set kernel. Furthermore, when combined with

quantized local image patches, this kernel can be reduced to a linear kernel between two visual word frequencies, which represent the two input image sets. As a result, the flattening largely improves the computational efficiency of MMIL. In fact, it speeds up of the kernel computation by an order of two. This acceleration makes MMIL applicable to large scale datasets that are frequently involved in image set classification. Then we further improve the proposed kernel in two aspects: First, it takes into account the discriminability of different visual codes by attaching each of them a weight derived from the classification errors. Second, the improved kernel handles non-linear samples by using an exponential form. We apply the proposed approach to a cannabis website classification task. In the task we collected a dataset of 600 websites containing more than 220,000 images. In the experiment, we compared the proposed approach with several state-of-the-art methods including the standard bag-of-words approach, the multi-layer bag-of-words approach, and MILES [11]. Our approach demonstrates promising results and compares favorably with its competitors.

The rest of paper is organized as follows. Related work is discussed in Section II. In Section III, we introduce the multi-layer set kernel we used and simplify it. Section IV improves the multi-layer set kernel by using weighting codewords and an exponential kernel. Experimental results are reported in Section V. We conclude the paper in Section VI.

II. Related Work

Multiple-instance learning (MIL) is first introduced by Dietterich et al. [4] in the field of drug activity prediction. the key assumption in MIL is that: a bag is positive if at least one of its instances is a positive example; otherwise, the bag is negative.

Based on this key assumption, many algorithms have been proposed. Maron and Lozano-Perez [7] present a general framework, called Diverse Density(DD), for solving MIL problems. Diverse Density tries to locate the true concept in the feature space by maximizing the DD function. Following the idea of DD, Zhang and Goldman [9] propose an algorithm EM-DD, which combines the expectation-maximization (EM) with DD to find the true concept. Andrews et al. [12] treat MIL as a maximum margin problem and use SVM to solve it. Bunescu and Mooney [13] propose a MIL method to tackle the situation that positive bags are sparse in positive instances. Zhou et al. [14] transform each email into a bag of multiple segments, and apply multiple instance logistic regression on the bags. They show an improved result with a multiple instance learning method.

Due to the strict assumption of MIL, it may fail in many other areas. Weidmann et al. [20] investigate a generalized view of the MIL problem. They assume that a set of underlying concepts contribute to the classification, while the only one concept used in conventional MIL. They define three generalizations: Presence-Based, Threshold-Based, and Count-Based MIL, which are distinguished by the number of instances of each concept appeared in a bag. Scott et al. [8] also introduce a similar generalization of the conventional MIL. Chen et al. propose DD-SVM [10] and MILES [11] for generalized MIL. Through different feature mapping, both DD-SVM and MILES convert MIL to a standard supervised learning problem. Haussler et al. [16] propose convolution kernel for tackling general discrete data structure by kernel methods. Gartner et al. [15] study several multiple-instance kernels for multiple instance data. The similarity between two bags can be directly calculated by kernel functions. This type of kernel methods indeed converts MIL to supervised learning problem, which can be easily solved by support vector machine or other classifiers.

Although these MIL approaches have been proved effective, they have all been applied to single set-to-instance structures, while we are interested in multi-layer structures. The most relevant work to ours appears in Gu et al. [5], where a similar idea of MMIL is applied to video concept detection. They also use the set kernel to all layers and construct an MMIL kernel which measures the similarities between the instances in the same and different layers. Our work is different from [5] in two aspects: First, we show that MMIL can be flattened to MIL when using the set kernel over all layers. We use this technique to design a very efficient (and effective as shown in our experiments) kernel, that is much faster than the one in [5]. This efficiency improvement allows us to tackle large scale data tasks, such as the cannabis website classification. Second, we further improve the flattened linear set kernel by weighted codewords and an exponential kernel.

Our work also relates to several second order vector similarity measures, especially the quadratic form (QF) distance [26], [27], [28] and the Mahalanobis kernel [1]. Details of the relations are give in Sec. III.

III. Multi-layer Multiple Instance Learning

A. Image Set Classification

Let us denote an image set as $S = \{I_i\}_{i=1}^n$ that contains n images $I_1, \ldots, I_n$ and n varies from set to set. The task of image set classification is to find a classification function $f : \mathbb{S} \to \{-1, 1\}$, where $\mathbb{S}$ is the set of all image sets, and $f(S) = 1$ predicts that $S \in \mathbb{S}$ belongs to the class of interest (e.g., image set from a cannabis website). For a given image I, we represent it with m local patches as $I = \{\mathbf{p}_j\}_{j=1}^m$, where m varies from image to image, and $\mathbf{p}_j \in \mathbb{R}^d$ is a d-dimensional local features (in our experiment, the 128 dimensional SIFT features [6] is used). Note that, for the notation convenience, we abuse the notation I to denote the image itself as well as its patch representation.

The set-to-image and image-to-patch structures naturally motivate us to use the multi-layer multiple instance learning (MMIL) to find the classification function f. We adopt the support vector machine (SVM) framework for this task. Consequently, our problem reduces to find an effective and efficient kernel for image sets. Note that efficiency is an important requirement here, due to the frequently involved large number of images in practical applications.

B. Multi-layer Set Kernel

One popular kernel used in multiple instance learning (MIL) is the *set kernel* [15], [16]. Given two sets X and X', the set kernel K between them is defined as:

$$K(X, X') := \sum_{x \in X} \sum_{x' \in X'} k(x, x') \ , \tag{1}$$

where $k(.,.)$ is used to denote a kernel on the instance space. When the instances x and x' are sets by themselves, we can define $k(.,.)$ recursively, which results in a *multi-layer set kernel*.

This definition of set kernel in (1) sometimes is biased to large sets since their cardinalities can dominate the solution of the estimation problem. To alleviate this problem, normalization is often needed to generate a new kernel as

$$\tilde{K}(X, X^{'}) := \frac{K(X, X')}{f_{norm}(X) f_{norm}(X')} \tag{2}$$

where $f_{norm}(X)$ is a normalization function of X. Several normalization strategies have been proposed in previous work [15], in this paper we choose the *feature-space normalization* defined as $f_{norm}(X) := \sqrt{K(X, X)}$.

Note that we conduct this normalization on the whole multi-layer set kernel instead of normalizing the set kernel of each layer. In other words, this normalization does not affect the following flattening of multi-layer set kernel.

C. Flattening Multi-layer Set Kernel

Given two image sets S_1 and S_2, we apply the multi-layer set kernel to them,

$$K(S_1, S_2) := \sum_{I_1 \in S_1} \sum_{I_2 \in S_2} K(I_1, I_2) \ . \tag{3}$$

Denoting the kernel between patches $\mathbf{p}_1$ and $\mathbf{p}_2$ as $k(\mathbf{p}_1, \mathbf{p}_2)$ (named a *patch kernel*), we can flatten the multi-layer set kernel (3) to a single-layer set kernel by the following derivation,

$$\begin{aligned} K(S_1, S_2) &= \sum_{I_1 \in S_1} \sum_{I_2 \in S_2} K(I_1, I_2) \\ &= \sum_{I_1 \in S_1} \sum_{I_2 \in S_2} \sum_{\mathbf{p}_1 \in I_1} \sum_{\mathbf{p}_2 \in I_2} k(\mathbf{p}_1, \mathbf{p}_2) \\ &= \sum_{I_1 \in S_1} \sum_{\mathbf{p}_1 \in I_1} \sum_{I_2 \in S_2} \sum_{\mathbf{p}_2 \in I_2} k(\mathbf{p}_1, \mathbf{p}_2) \\ &= \sum_{\mathbf{p}_1 \in \widehat{S}_1} \sum_{\mathbf{p}_2 \in \widehat{S}_2} k(\mathbf{p}_1, \mathbf{p}_2) \\ &= K(\widehat{S}_1, \widehat{S}_2) \ , \end{aligned} \tag{4}$$

where we use $\widehat{S} := \bigcup_{I \in S} I$ to represent the union of all patches in an image set S[1].

The above flattening shows that, when the comparison between set members at higher layer depends purely on the comparisons between their members, the hierarchical structure between these two layers may not carry much information as it appears. The relations between the set kernel, the multi-layer set kernel, and the flattened one layer set kernel are shown in Fig. 2. Fig. 2(a) shows how to calculate the set kernel between two images. Similarly, Fig. 2(b) shows how to measure the similarity between image sets.

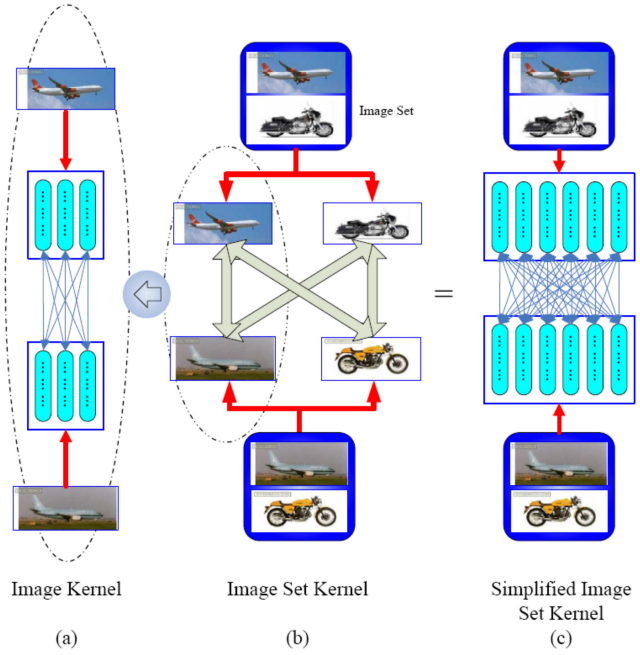

Fig. 2. The proposed multi-layer set kernel used in MMIL model: (a) the kernel between images, (b) the kernel between image sets, and (c) the simplified kernel in the image set – local patch structure.

The flattening in (4) clearly simplifies the notation complexity of the kernel. However, it does not directly reduce the computation, because the number of computation of patch kernel $k(.,.)$ remains unchanged. Fortunately, for quantized patches that are widely used in object and category classification, we show in the following subsection that the computation can be largely accelerated.

D. Efficient MMIL using Visual Vocabulary

Directly working on image patches or its feature representation is very expensive since high dimensional feature analysis is involved. Instead, we apply vector quantization on the feature space to create a visual vocabulary representation, similar to those used in the bag-of-visual-words model [3], [21]. Specifically, we build a visual vocabulary $\mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_{n_v}\}$ of size n_v by clustering in the feature space using k-means. Then a patch $\mathbf{p}$ can be represented by a word in $\mathcal{V}$ by the mapping $\pi(\mathbf{p}) = \arg\min_{1 \le i \le n_v} ||\mathbf{p} - \mathbf{v}_i||$.

With this representation, to define a patch kernel we only need the kernel between two visual words. In this paper we use the RBF-like kernel as

$$k(\mathbf{p}_1, \mathbf{p}_2) := \exp(-\gamma ||\mathbf{v}_{\pi(\mathbf{p}_1)} - \mathbf{v}_{\pi(\mathbf{p}_2)}||) \ , \tag{5}$$

[1]For notation convenience, we allow patches in a set to have same descriptions, i.e., we may have $\mathbf{p}_1 = \mathbf{p}_2$ in a same set (either in I or $\widehat{S}$). Theoretically, this can be done by attaching each patch with a universally unique index.

where γ is a pre-defined parameter. Since there are only limited number (n_v) of visual words, we pre-compute kernel values of all pairs of them. Such pre-computed values are stored in a matrix $\mathbf{G} \in \mathbb{R}^{n_v \times n_v}$, i.e., $\mathbf{G}(i,j) = k(\mathbf{v}_i, \mathbf{v}_j)$.

With the vocabulary representation, an image I can be efficiently represented using its word frequency $h(.;I)$, such that $h(i;I) = |\{\mathbf{p} \in I : \pi(\mathbf{p}) = i\}|$, for $i = 1, \ldots, n_v$, where $|.|$ is the cardinality of a set. Accordingly, the word frequency $H(.;S)$ for an image set S is computed as $H(i;S) = \sum_{I \in S} h(i;I)$, for $i = 1, \ldots, n_v$.

Applying word frequency representation to simplify (4), we have

$$\begin{aligned} K(S_1, S_2) &= \sum_{\mathbf{p}_1 \in \widehat{S}_1} \sum_{\mathbf{p}_2 \in \widehat{S_2}} \mathbf{G}(\pi(\mathbf{p}_1), \pi(\mathbf{p}_2)) \\ &= \sum_{w_1=1}^{n_v} \sum_{w_2=1}^{n_v} H(w_1;S_1) H(w_2;S_2) \mathbf{G}(w_1, w_2) \\ &= H_1^\top \mathbf{G} H_2 \end{aligned} \tag{6}$$

where H_1, H_2 denote the word frequency histograms for S_1, S_2 respectively. Note that the above derivation also applies to the kernel between two images I_1, I_2, i.e.,

$$K(I_1, I_2) = h(.;I_1)^\top \mathbf{G} h(.;I_2) \tag{7}$$

Therefore, if we denote the computation complexity of kernel in (6) as $O(T_k)$ and let $n_1 = |S_1|, n_2 = |S_2|$ (here n_1 and n_2 are the numbers of images in S_1 and S_2 respectively), the computation complexity of $K(S_1, S_2)$ *without* flattening (i.e. Eqn. 3) is $O(n_1 n_2 T_k)$. This shows that the flattening improves the speed by an order of two.

The proposed method has several advantages toward image set classification tasks.

First, the kernel in (6) is a Mercer kernel. It has been proved in [16] that a set kernel is a Mercer kernel if and only if $k(.,.)$ itself is a Mercer kernel. Our patch kernel $k(.,.)$ is an RBF kernel that satisfies the condition.

Second, the proposed kernel takes into account correlation information between visual words via matrix $\mathbf{G}$. Such information is often lost in previous bag-of-words models. In addition, the scheme makes the kernel less sensitive to the generation of vocabulary.

Third, the computation of kernel (6) is very efficient. In addition to being two orders faster than the original MMIL solution, the kernel can actually computed even faster. Note that $\mathbf{G}$ is a symmetric positive definite matrix and therefore has orthogonal decomposition as $\mathbf{G} = \mathbf{Q}^\top \mathbf{Q}$, where $\mathbf{Q}$ is an orthogonal matrix that can be pre-computed when $\mathbf{G}$ is ready. Consequently, we can rewrite the kernel in (6) as

$$K(S_1, S_2) = H_1^\top \mathbf{Q}^\top \mathbf{Q} H_2 = (\mathbf{Q} H_1)^\top (\mathbf{Q} H_2) \ . \tag{8}$$

This indicates that the kernel can be further reduced to a linear kernel after a linear transformation on word frequency histograms.

E. Relation to Other Methods

The proposed methods, after the above derivation, shares similarity with several well-known techniques, which we will summarize below:

Relation to the linear kernel. When $\gamma \to \infty$, we have $G(i,j) \to \delta(i-j)$ and then the kernel $K(S_1, S_2)$ degenerates to linear kernel

$$K(S_1, S_2) = H_1^\top H_2 \ . \tag{9}$$

Relation to the Mahalanobis kernel. Note that our kernel is also relevant to the Mahalanobis kernel [1] where G in (6) is derived from covariance matrix. However, in our case G has totally different meanings. In addition, our kernel is more computationally efficient because there is no exponential computation involved.

Relation to the Quadratic Form distance. The Quadratic Form (QF) distance [26], [27], [28] is a cross-bin distance defined as

$$\begin{aligned} d_{QF}(I_1, I_2) &= \sqrt{(I_1 - I_2)^T A (I_1 - I_2)} \\ &= \sqrt{I_1^T A I_1 + I_2^T A I_2 - 2 I_1^T A I_2} \ , \end{aligned} \tag{10}$$

where $A = (a_{ij}) = (\exp(-\gamma ||c_i - c_j||))$ measures the between bin similarities, which is almost identical to $\mathbf{G}$ in our flattened linear set kernel. As a result, our proposed kernel is nothing but the third term in (10) divided by the first and second terms in (10). This close relationship shows that our kernel is capable of capturing cross-bin information.

IV. Improve the Kernel

The kernel introduced above is efficient and compares favorably with standard bag-of-words kernels. However, we observed two major limitations. First, different visual words have different discriminability, while this fact is not taken into account (explicitly) in the kernel. Second, the linearity of the kernel makes it less effective to handle complicated set samples. Motivated by these observations, we propose improving the kernel from two aspects as described in the following subsections.

A. Weighing Words with Discriminability

One way to measure the discriminability of a visual word $\mathbf{v}_i \in \mathcal{V}$ is to build a "weak" classifier for it and study the classification performance [25]. Inspired by this idea, for $\mathbf{v}_i$, we build a trump classifier $f(.)$ using the i^{th} bin in the histogram of visual words (e.g., H_1, H_2 in (6)),

$$f_i(H) = \text{sign}(H(i) > \tau_i) \ , \tag{11}$$

where τ_i is a threshold that is tuned to achieve an equal error rate (EER). The EER for $f_i(.)$ is denoted as eer_i,

$$eer_i = \frac{\#\{\text{correctly classified positive samples by } f_i\}}{\#\{\text{all positive samples}\}} \ .$$

After getting EERs for all visual words, we define the weight w_i for each word $\mathbf{v}_i$ by first take it as $1 - eer_i$ and then scale all of them to the range $[0, 1]$. Specifically, we have

$$\begin{aligned} w_i &= \frac{(1 - eer_i) - \min_{1\le k\le n_v}\{1 - eer_k\}}{\max_{1\le k\le n_v}\{1 - eer_k\} - \min_{1\le k\le n_v}\{1 - eer_k\}} \\ &= \frac{\max_{1\le k\le n_v} eer_k - eer_i}{\max_{1\le k\le n_v} eer_k - \min_{1\le k\le n_v} eer_k} \ . \end{aligned}$$

The weights $w_1, w_2, \ldots, w_{n_v}$ capture the discriminability of each visual word. We use them to improve the kernel in (6) by weighing each histogram bin with its corresponding weight,

$$\begin{aligned} K_w(S_1, S_2) &= \sum_{i=1}^{n_v}\sum_{j=1}^{n_v} H_1(i) w_i \mathbf{G}_{i,j} H_2(j) w_j \\ &= (\mathbf{W}H_1)^\top \mathbf{G}\ (\mathbf{W}H_2) \\ &= (\mathbf{QW}H_1)^\top (\mathbf{QW}H_2) \end{aligned} \tag{12}$$

where $\mathbf{W} = \mathrm{diag}(w_1, w_2, \ldots, w_{n_v})$ is the diagonal weighting matrix. Equation (12) can also be written as $K_w(S_1, S_2) = H_1^\top(\mathbf{W}^\top \mathbf{G}\mathbf{W})H_2$, which shows how the between-word information (**G**) and the discriminability of each visual code (**W**) is combined into the kernel K_w.

To handle the unbalance in the number of images, we further normalize K_w as,

$$\widehat{K}_w(S_1, S_2) = \frac{K_w(S_1, S_2)}{\sqrt{K_w(S_1, S_1) K_w(S_2, S_2)}} \ . \tag{13}$$

B. Build a Nonlinear Kernel

Handling nonlinearity is very important for the analysis of complicated image sets. To achieve this, we put $\widehat{K}_w$ in the exponential kernel like the following,

$$K_{wn} = \exp\{\kappa \widehat{K}_w\} \ , \tag{14}$$

where parameter κ is a positive constant ($\kappa = 3$ in all our experiments). Such a Gaussian-like kernel has been widely used to generate non-linear kernels, such as the χ^2 kernel and the EMD kernel that have been used in the bag-of-the-word model for image categorization tasks [29]. Because kernel(6) is a Mercer kernel. Consequently, K_{wn} is a Mercer kernel that is ready for the use in kernel based approaches.

V. Experiments

Cannabis website classification is an important task in intelligent internet filtering, especially for preventing drug abusing in young people [19]. Unlike previous work that is based on textual information [22] or image information [23], in this work we treat this problem as an image set classification problem. Specifically, we are interested in cannabis website classification, where a cannabis website is defined as a website where cannabis plants and/or smoking tools are sold. In the task, a website is classified as either a cannabis website or not based on the images it contains. This image set-based approach can also be combined with texture based methods, but this combination is beyond the scope of this paper.

# Images	< 40	40 - 100	> 100
Cannabis Webs	28 (14%)	33 (16.5%)	139 (69.5%)
Normal Webs	69 (17.25%)	78 (19.5%)	253 (63.25%)
All	97 (16.2%)	111 (18.5%)	392 (65.3%)

TABLE I
THE DISTRIBUTION OF NUMBERS OF IMAGES IN COLLECTED WEBSITES.

A. Database and Preprocessing

We have collected 200 cannabis websites containing over 70,000 images and 400 normal websites with about 150,000 images. This large scale data makes a strict request on computational efficiency. Many methods that can be used in small scale data will no longer work in our dataset. For comparison, we further design a small dataset. In this small dataset, each website only contains 20 images which are randomly selected from the corresponding website in the full dataset. This small dataset is more close to real application, because it is time-consuming to collect and process large number of images from websites. In this small dataset, we will compare the proposed approach with state-of-the-art MIL-based methods.

Normal websites (i.e. non-cannabis websites) in our collection contain many different categories, including cigarette, plants, hemp cloth, news, shopping, glass, scientific equipments, etc. The reasons we download these types of websites are different: Some cannabis websites also sell cigarette; plants websites contain various plants, some of which may be similar to cannabis plant; hemp cloth websites may have similar keywords; news websites are representation of normal websites and contain with various information; shopping websites sell various normal products; glass and scientific equipments websites sell glasses which may be similar to glass smoking tools. In summary, we purposely made the normal website collection represent both normal and/or easy-to-confused websites compared with cannabis websites.

Some example image sets from cannabis websites are shown in Fig. 4 and 5; while examples from normal websites are shown in Fig. 6 and 7. The images whose height or width smaller than 100 pixels are discarded, for these images are usually advertisements or thumbnails of no significance. Bilinear interpolation is used to normalize the images whose sizes are larger than 1000×1000 into 1000×1000 pixels. The data will be made publicly accessible.

Table I reports the statistics on number of images in websites in our collection. The table shows that 69.5% cannabis websites have more than 100 images. Since the image number statistics of cannabis and normal Webs are similar, there is no hint for classification through the number of images.

In the preprocessing step, all images are first converted to gray images. Then, we use the standard SIFT features [6] for local patch description.

B. Methods

In addition to the proposed approach, we include several methods in our experiments for comparison.

Bag-of-words. In this method, the "image" concepts are discarded and an image set is represented directly using the histogram of all visual words over its member images. Specifically, after clustering in local feature space using k-means, each website can be represented by a histogram of words. Then we use normalized histograms as features for SVM. RBF kernel is used in SVM to calculate histogram dissimilarity. Five fold cross-validation and grid search in parameter space are used to get the best result.

Multi-layer Bag-of-words. In this method, we first do clustering in local feature space, then each image's feature vector is a histogram of words. After that, we do clustering in image's feature space to form image words. Each image set can be represented by a histogram of image words. Finally, we train SVM classifier using histograms of image words as features of image sets. Similar to baseline method1(bag-of-words), five fold cross-validation and grid search in parameter space are used to get the finally result.

Linear set kernel. This is the kernel in equation (6). We use this linear set kernel in MMIL.

Exponential set kernel. In this method, we only improve the original linear set kernel with the exponential scheme, without using the weights of words.

Weighted set kernel. In this method, we only use the weights of words to improve the original linear set kernel, without using the exponential scheme.

Weighted exponential set kernel. In this method, we use both the weights of words and the exponential scheme to improve the original linear set kernel. This is our final kernel.

MILES: Multiple-Instance Learning via Embedded Instance Selection [11]. This method shows outstanding performance when comparing with other MIL methods. We conduct MILES with the code provided by the authors [11].

C. Evaluation Criteria

We conduct a five-fold cross-validation test on the dataset. Then, three criteria are used for evaluating the different approaches on the image set classification: *true positive rate* (TPR), *false positive rate* (FPR), and *equal error rate* (EER). These rates are defined as: $\text{TPR} = \frac{\#\text{Detected Cannabis Webs}}{\#\text{All Cannabis Webs}}$, $\text{FPR} = 1 - \frac{\#\text{Detected Normals}}{\#\text{All Normals}}$, and $\text{EER} = \text{FPR} = 1 - \text{TPR}$ (when FPR=1-TPR).

By adjusting the SVM separating boundary, we generate ROC (Receiver Operating Characteristic) curves for each methods. These curves and the equal error rates are used for evaluation.

D. Experimental results on the full dataset

The average equal error rates (EERs) and the receiver operating characteristic (ROC) curves over all crosses are used to evaluate the tested methods. EERs are summarized in Table II and ROC curves are shown in Figure 1.

From Table II we can see that the multi-layer bag-of-words method performs much worse than others. The bag-of-words method is similar to the linear set kernel method, for it also discard the image layer and represent image set as a histogram of visual words. The difference between bag-of-words and the linear set kernel method lies in the set kernel we used instead of RBF kernel. Hence, it means the linear set kernel we proposed can actually improve the performance compared with standard bag-of-words model.

Method	EER
Multi-layer Bag-of-words	24.25%
Bag-of-words	13.5%
Linear set kernel	11.75%
Exponential set kernel	11.25%
Weighted set kernel	10.5%
Weighted exponential set kernel	9.75%

TABLE II
AVERAGE EQUAL ERROR RATES (EER) ON FULL DATASET.

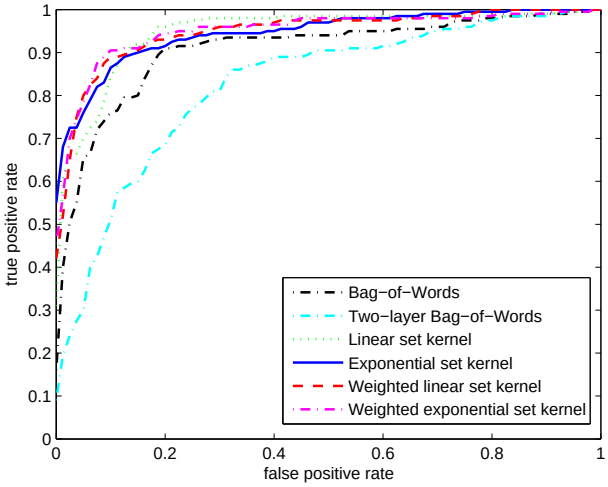

Fig. 3. ROC curves of different approaches for the cannabis website classification task.

In addition, proposed method (weighted exponential set kernel) performs the best EER(9.75%), much better than the standard bag-of-words methods and the linear set kernel in (6). This implies that using our two improvements together can increase the result 2%, while only using weights of words or exponent can increase 1.25% or 0.5% respectively. There are mainly two reasons for these improvements. First, the set kernel only represents a similarity without any training information; it may be fine in traditional supervised learning, but in multi-instance learning, the basic assumption is a bag is positive if at least one of its instance is positive. Under this situation, it is better to consider the sample distribution– which is captured by the weights of words. Second, the exponential mapping lets the new kernel have the ability to handle complicated non-linear situations.

E. Experimental results on the small dataset

Due to the time complexity, it is hard to directly apply state-of-the-art MIL methods (e.g. MILES [11]) on the full dataset. For comparison, we implement MILES on the small

Method	EER	Time cost
MILES [11]	30.25%	464.69s
Weighted exponential set kernel	24.75%	5.24s

TABLE III
AVERAGE EQUAL ERROR RATES (EER) AND COMPUTATION TIME COST ON SMALL DATASET.

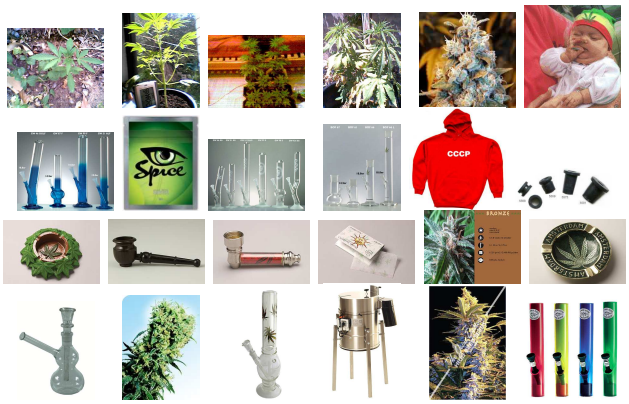

Fig. 4. Example cannabis websites that are correctly classified by our methods. Each row shows 6 randomly selected images from a website (same for the following figures).

dataset using the code provided by Chen et al. [11]. To use MIL methods (e.g. MILES) in multi-layer MIL, we treat each histogram of image as an instance. Then each website is a bag containing some images(instances). We use a vocabulary of 2000 words and 5-fold cross-validation to optimize parameters in MILES and proposed method.

Table III shows the average equal error rates (EER) and computation time cost of MILES and proposed method on small dataset. The computation time cost is the average time-consuming for each training and testing set, which includes feature mapping in MILES and kernel computing in proposed method. For better comparison, we do not include the time cost of SVM classifier. We can see that our method greatly improves the performance and meantime is much faster than MILES. This demonstrates that the method in MIL is not suitable to directly use in Multi-layer MIL, for the simplification from Multi-layer into one layer will greatly impact the MIL method. Because we use the set kernel which can directly flatten the multi-layers, the proposed method is much robust to this kind of simplification.

Fig. 5. Example cannabis websites that are incorrectly classified by our methods.

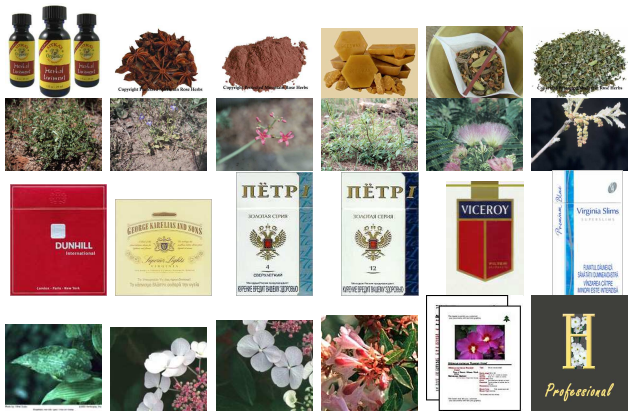

Fig. 6. Example normal websites that are correctly classified by our method.

Fig. 7. Example normal websites that are incorrectly classified by our method.

VI. CONCLUSION AND FUTURE WORK

We have proposed an image set classification algorithm using multi-layer multiple instance learning model. In the approach, an image set is represented by a bag of images, which are further represented by the local image descriptors. This representation naturally links to multi-layer multiple instance learning model. Using a set kernel, we show that such a multi-layer model can be largely simplified by flattening it into one simple layer. To reduce the complexity, a weighted bag-of-words model is used in local feature space. In the application of cannabis website classification, the proposed approach shows promising results in comparison with other methods. In the future we plan to investigate further MMIL (e.g., using other kernel or non-kernel approaches) and its application to other fields, such as video classification, personal photo collection analysis, etc.

Acknowledgement. This work was supported in part by the NSFC (Grant No. 60825204 and 60935002). H. Ling is supported in part by NSF (Grant No. IIS-0916624).

REFERENCES

[1] S. Abe. Training of Support Vector Machines with Mahalanobis Kernels. *ICANN*, pp. 571-576, 2005.
[2] O. Chapelle, P. Haffner and V.N. Vapnik. Support vector machines for histogram-based image classification. *IEEE Trans. on Neural Networks*, pp. 1055-1064, 1999.
[3] G. Csurka, C. Dance, L. Fan, J. Williamowski and C. Bray. Visual categorization with bags of keypoints. In *ECCV Workshop on SLCV*, 2004.

[4] T.G. Dietterich, R.H. Lathrop, and T. Lozano-Perez. Solving the Multiple Instance Problem with Axis-Parallel Rectangles. *Artificial Intelligence*, pp. 31-71, 1997.

[5] Z. Gu, T. Mei, X.S. Hua, J. Tang, X. Wu. Multi-Layer Multi-Instance Learning for Video Concept Detection. *IEEE Trans. on Multimedia*, 2008.

[6] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, vol. 60, pp. 91-110, 2004.

[7] O. Maron and T. Lozano-Perez. A Framework for Multiple-Instance Learning. *Advances in Neural Information Processing Systems*, pp. 570-576, 1998.

[8] S.D. Scott, J. Zhang, and J. Brown. On Generalized Multiple-Instance Learning. *IJCIA*, 2005.

[9] Q. Zhang and S.A. Goldman. EM-DD: An Improved Multiple-Instance Learning Technique. *Advances in Neural Information Processing Systems*, pp. 1073-1080, 2002.

[10] Y. Chen and J.Z. Wang. Image Categorization by Learning and Reasoning with Regions. *J. Machine Learning Research*, vol. 5, pp. 913-939, 2004.

[11] Y. Chen, J. Bi, J.Z. Wang. MILES: Multiple-instance learning via embedded instance selection. *IEEE Trans. on PAMI*, 2006.

[12] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support Vector Machines for Multiple-Instance Learning. *Advances in Neural Information Processing Systems*, 2003.

[13] R.C. Bunescu and R.J. Mooney. Multiple instance learning for sparse positive bags. *ICML*, pp. 105-112, 2007.

[14] Y. Zhou, Z. Jorgensen, M. Inge. Combating Good Word Attacks on Statistical Spam Filters with Multiple Instance Learning. In *IEEE International Conference on Tools with Artificial Intelligence*, vol. 2, pp. 298-305, 2007.

[15] T. Gartner, A. Flach, A. Kowalczyk, and A.J. Smola. Multi-Instance Kernels. *ICML*, pp. 179-186, 2002.

[16] D. Haussler. Convolution kernels on discrete structures. In *Technical Report UCS-CRL-99-10. UC*, 1999.

[17] M.J. Swain and D.H. Ballard. Indexing via color histograms. In *ICCV*, pp. 390-393, 1990.

[18] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning object categories from Google's image search. *ICCV*, 2005.

[19] P.M. Wax. Just a click away: Recreational drug websites on the Internet. *Pediatrics*, 109(6):96, 2002.

[20] N. Weidmann, E. Frank, and B. Pfahringer. A Two-Level Learning Method for Generalized Multi-instance Problems. *ECML*, pp. 468-479, 2003

[21] J. Winn, A. Criminisi and T. Minka. Object categorization by learned universal visual dictionary. *ICCV*, 2005.

[22] B. Grilheres, S. Brunessaux and P. Leray. Combining classifiers for harmful document filtering. In *Recherche d'Information Assistee par Ordinateur*, 2004.

[23] N. Xie, X. Li, X. Zhang, W. Hu and J.Z. Wang. Boosted Cannabis Image Recognition. In *ICPR*, 2008.

[24] N. Xie, H. Ling, W. Hu, and X. Zhang. Bin-Ratio Information for Category and Scene Classification. *CVPR*, 2010.

[25] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. *CVPR*, 2001.

[26] J. Hafner, H. Sawhney, W. Equitz, M. Flickner and W. Niblack. Efficient color histogram indexing for quadratic form distance functions. *IEEE Trans. on PAMI*, pp. 729–736, 1995.

[27] Y. Rubner, J. Puzicha, C. Tomasi and J.M. Buhmann. Empirical evaluation of dissimilarity measures for color and texture. In *CVIU*, 2001.

[28] W. Niblack, R. Barber, W. Equitz, M. D. Flickner, E. H. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, G. Taubin and Y. Heights. Querying images by content, using color, texture, and shape. *In SPIE Conference on Storage and Retrieval for Image and Video Databases*, 1993.

[29] J. Zhang, M. Marszalek, S. Lazebnik and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 2007.