Keywords to Visual Categories:
Multiple-Instance Learning for Weakly Supervised Object Categorization

Sudheendra Vijayanarasimhan and Kristen Grauman
Department of Computer Sciences
University of Texas at Austin
Austin, TX 78712

Abstract

Conventional supervised methods for image categorization rely on manually annotated (labeled) examples to learn good object models, which means their generality and scalability depends heavily on the amount of human effort available to help train them. We propose an unsupervised approach to construct discriminative models for categories specified simply by their names. We show that multiple-instance learning enables the recovery of robust category models from images returned by keyword-based search engines. By incorporating constraints that reflect the expected sparsity of true positive examples into a large-margin objective function, our approach remains accurate even when the available text annotations are imperfect and ambiguous. In addition, we show how to iteratively improve the learned classifier by automatically refining the representation of the ambiguously labeled examples. We demonstrate our method with benchmark datasets, and show that it performs well relative to both state-of-the-art unsupervised approaches and traditional fully supervised techniques.

1. Introduction

The problem of recognizing generic object categories lies at the heart of computer vision research. It is challenging on a number of levels: objects of the same class may exhibit an incredible variability in appearance, real-world images naturally contain large amounts of irrelevant background "clutter", and subtle context cues can in many cases be crucial to proper perception of objects. Nonetheless, recent advances have shown the feasibility of learning accurate models for a number of well-defined object categories (e.g., [12, 20, 16]).

Unfortunately, the accuracy of most current approaches relies heavily on the availability of labeled training examples for each class of interest, which effectively restricts existing results to relatively few categories of objects. Manually collecting (and possibly further annotating, aligning, cropping, etc.) image examples is an expensive endeavor, and having a human in the loop will inevitably introduce biases in terms of the types of images selected [21]. Arguably, the protocol of learning models from carefully gathered images has proven fruitful, but it is too expensive to perpetuate in the long-term.

The Web is thus an alluring source of image data for vision researchers, given both the scale at which images are freely available as well as the textual cues that surround them. Querying a keyword-based search engine (e.g., Google Image Search) or crawling for meta-tags (e.g., on Flickr) will naturally yield images of varying degrees of relevance: only a portion will contain the intended category at all, others may contain instances of its homonym, and in others the object may barely be visible due to clutter, low resolution, or strong viewpoint variations. Still, dataset creators can use such returns to generate a candidate set of examples, which are then manually pruned to remove irrelevant images and/or those beyond the scope of difficulty desired for the dataset (e.g., [10, 9]).

Though appealing, it is of course more difficult to learn visual category models straight from the automatically collected image data. Recent methods attempt to deal with the images' lack of homogeneity indirectly, either by using clustering techniques to establish a mixture of possible visual themes [25, 11, 17], or by applying models known to work well with correctly labeled data to see how well they stretch to accommodate "noisily" labeled data [13, 24]. Unfortunately, the variable quality of the search returns and the difficulty in automatically estimating the appropriate number of theme modes make such indirect strategies somewhat incompatible with the task.

In this work, we propose a more direct approach to learn discriminative category models from images associated with keywords. We introduce an unsupervised method for *multiple-instance* visual category learning that explicitly acknowledges and accounts for their ambiguity. Given a list of category names, our method gathers groups of potential images of each

category via a number of keyword-based searches on the Web. Because the occurrence of true exemplars of each category may be quite sparse, we treat the returned groups as *positive bags* that contain some unknown amount of positive examples, in addition to some unrelated negative examples. Complementary *negative bags* are obtained by collecting sets of images from unrelated queries, or alternatively from any existing database having categories outside of the input list. We show how optimizing a large-margin objective function with constraints that reflect the expected sparsity of true positive examples yields discriminative models that can accurately predict the presence of the object categories within novel images, and/or provide a good re-ranking of the initial search returns. Further, we develop a means for the learned classifier to iteratively improve itself by continually refining the representation of the ambiguously labeled examples.

Our main contribution is a multiple-instance learning-based approach for weakly supervised[1] category learning from images. Our learning paradigm exploits the wealth of text surrounding natural images that already exists, while properly accounting for their anticipated noise and ambiguity. Experimental results indicate the approach's promise: on benchmark image datasets it competes well with several fully supervised methods, is more accurate than a single-instance learning SVM baseline, and improves on state-of-the-art unsupervised image classification results.

2. Related Work

Given the expense of labeled image data, researchers have explored various ways to reduce supervision requirements. Recent work has provided methods to reduce the number of exemplars required to learn a category [2, 10], novel sources of annotated image data [8, 11, 3, 4, 24], and clustering techniques for grouping unlabeled images with minimal supervision [25, 15, 17, 23].

A number of authors have studied probabilistic clustering methods originally used for text—such as Latent Semantic Analysis (pLSA), Latent Dirichlet Analysis, and Hierarchical Dirichlet Processes—to discover the hidden mixture of visual themes ("topics") in a collection of unorganized [25, 23] or semi-organized [11, 17] image data. A clustering approach based on Normalized Cuts is also proposed in [15]. Clustering methods are most appropriate for mining image data, but not necessarily for learning categories: they may sometimes elicit themes associated with semantic categories, but there is no way to guarantee it. Additionally, these approaches face the difficulty of selecting the appropriate number of clusters; for images collected with Web search this number is bound to be highly variable. Finally, many such methods are themselves not equipped to provide models to classify novel examples. For example, pLSA requires some way to select which topic to use for each class model, and must resort to a "folding-in" heuristic when used for prediction [11, 25]; the Normalized Cuts approach [15] must find prototypes that can serve as good training examples. Our approach streamlines these limitations, allowing categories of interest to be directly specified, and producing a large-margin classifier to recognize novel instances.

Vision researchers have identified innovative ways to take advantage of data sources where text naturally accompanies images, whether in news photograph captions [3], annotated stock photo libraries [8], or generic web pages [27, 4, 24]. Our method also exploits text-based indexing to gather image examples, however thereafter it learns categories from the image content alone.

The multiple-instance learning (MIL) setting (to be defined in detail below) was first identified in [7], who represented ambiguously labeled examples using axis-parallel hyperrectangles and demonstrated applications for drug activity prediction. More recently MIL has received various treatments within the machine learning community [29, 1, 14, 22, 5]. In [5], a large-margin MIL formulation that addresses the possibility of very sparse positive bags is proposed, and it is demonstrated on several machine learning datasets. The ability to learn from sparse positive bags is in fact critical to our application; we show how to integrate their MIL objective for the purpose of unsupervised category learning.

Previous instances of MIL in vision have focused on the task of segmentation, that is, separating foreground regions from background within the same image [19, 26, 28]. While in that setting one image is a positive bag, and only a subset of the component blobs are true positive examples (i.e., correspond to foreground), we consider the problem of learning from an imperfectly labeled collection of images, where only a subset of image examples correspond to the category of interest. We are the first to frame unsupervised category learning as a MIL problem, to provide a direct solution to constructing discriminative category models from keyword-based image search, and to develop an MIL approach to simultaneously refine the classifier and bag representation.

---

[1]Our method is unsupervised in the sense that it does not require human input, but we also refer to it as "weakly supervised" since some partitioning is being done by the search engine.

(a) MIL for visual category learning   (b) Iterative model improvement
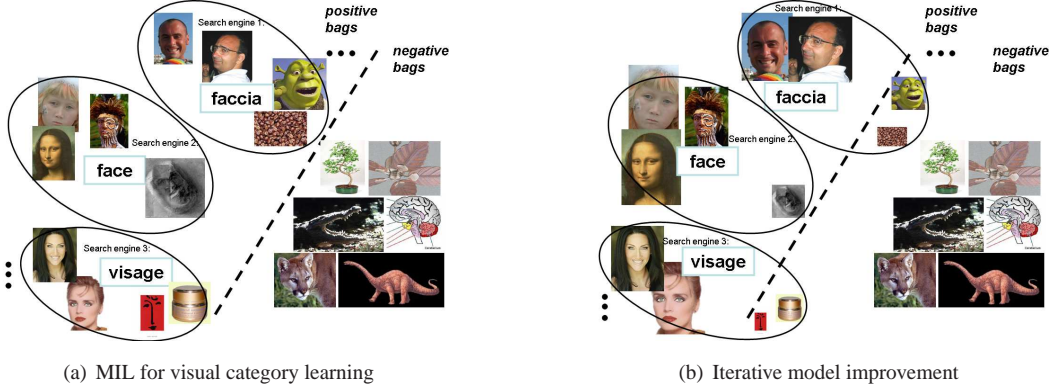
Figure 1. Overview of the proposed approach. (a) Given a category name, our method automatically collects noisy "positive bags" of instances via keyword-based image search on multiple search engines in multiple languages. Negative bags are constructed from images whose labels are known, or from unrelated searches. The sparse MIL classifier can discriminate the true positive instances from the negatives, even when their sparsity in the positive training bags is high. (b) From the initial sparse MIL solution, the classifier improves itself by iteratively updating the representation of the training bags. Stronger positive instances have more impact on the decision boundary, while detected false positives (depicted here with smaller images) have less impact.

3. Approach

The goal of this work is to enable automatic learning of visual categories. Given a list of the names of classes of interest, our method will produce discriminative models to distinguish them. The main idea is to exploit the keyword-based image search functionality of current web search engines to retrieve a collection of images that may have some relationship to the concept of interest, and use them to train classifiers. However, text-based search is an inexpensive but rather imperfect tool for indexing images; it is driven almost entirely by matching the query to keywords that appear within an image file name or surrounding text, both of which need not correspond to actual visual content.

Therefore, rather than simply treat all images returned by a keyword search as positive instances of the class of interest, we formulate a multiple-instance learning problem to explicitly encode this ambiguity. We insert a constraint into the optimization function for a large-margin decision boundary that reflects the fact that as few as one example among those retrieved may be a true positive. Further, from an initial MIL solution, we show how to iteratively improve both the representation of the images and the classifier itself. Having learned classifiers for each category of interest, our method can predict the presence of the learned categories within new images, or re-rank the images from the original searches according to the relevance (see Figure 1).

In the following we overview multiple-instance learning and an MIL approach for sparse positive bags, then describe how our method generates MIL training sets, our iterative technique to boost sparse MIL, and the manner in which novel images are classified.

3.1. Multiple Instance Learning

The traditional (binary) supervised classification problem assumes the learner is provided a collection of N labeled data points $\{(x_i, y_i)\}_{i=1}^{N}$, where each $x_i \in \Re^d$ has a label $y_i \in \{+1, -1\}$, for $i = 1, \dots, N$. The goal is to determine the function $f : \Re^d \to \{+1, -1\}$ that best predicts labels for new input patterns drawn from the same distribution as the training examples, such that the probability of error is minimized. As in [7], one can conceive of more general situations where a learner is provided with *sets* (bags) of patterns rather than individual patterns, and is only told that at least one member of any *positive bag* is truly positive, while every member of any *negative bag* is guaranteed to be negative. The goal of MIL is to induce the function that will accurately label individual instances such as the ones within the training bags. The challenge is that learning must proceed in spite of the label ambiguity: the ratio of negative to positive instances within every positive bag can be arbitrarily high.

One might argue that many MIL settings—including ours—could simply be treated as a standard "single-instance learning" (SIL) setting, just where the labels are noisy. For instance, a support vector machine (SVM) has slack parameters that enable soft-margins, which might deal with some of the false positive training examples. However, a recent study comparing various supervised learners and their MIL counterparts reveals that ignoring the MI setting of a learning problem can be detrimental to performance, depending on the sparsity and distributions of the data [22]. Further, our results comparing our

MIL approach to an SIL baseline corroborate this finding (see Section 4).

3.2. Keyword-based Image Search and MIL

We observe that the mixed success of keyword-based image search leads to a natural MIL scenario. A single search for a keyword of interest yields a collection of images within which (we assume) at least one image depicts that object, thus comprising a positive bag. To generate multiple positive bags of images, we gather the results of multiple keyword-based image queries, by translating the query into multiple languages, and then submitting it to multiple search engines. The negative bags are collected from random samples of images in existing labeled datasets, from only those categories which do not have the same name as the category of interest, or from keyword image search returns for unrelated words (we experiment with both ideas below).

There are several advantages to obtaining the training bags in this manner: doing so requires no supervision (an automated script can gather the requested data), the collection process is efficient since it leverages the power of large-scale text search engines, and the images are typically available in great numbers. Perhaps more interesting, however, is that most of the images will be natural, "real-world" instances illustrating the visual category that was queried. Standard object recognition databases used extensively in the vision community have some inherent biases or simplifications (e.g., limitations to canonical poses, unnaturally consistent backgrounds, etc.), which can in turn limit the scope of the visual categories learned. Our approach will be forced to model a visual category from a much richer assortment of examples, which in some cases could lead to richer category models, or at least may point to a need for more flexible representations.

3.3. Sparse MIL

To recover a discriminative classifier between positive and negative bags of images, we consider the objective function suggested in [5] to determine a large-margin decision boundary while accounting for the fact that positive bags can be arbitrarily sparse. The sparse-MIL (sMIL) optimization adapts a standard SVM formulation to accommodate the multi-instance setting.

We consider a set of training bags of images $\mathcal{X}$, which is itself comprised of a set of positive bags $\mathcal{X}_p \subseteq \mathcal{X}$ and a set of negative bags $\mathcal{X}_n \subseteq \mathcal{X}$. Let X be a bag of images, and $\tilde{\mathcal{X}}_p = \{x | x \in X \in \mathcal{X}_p\}$ and $\tilde{\mathcal{X}}_n = \{x | x \in X \in \mathcal{X}_n\}$ be the set of instances from positive and negative bags, respectively. A particular image instance x is described in a kernel feature space as $\phi(x)$ (and will be defined below). The SVM decision hyperplane weight vector w and bias b are computed as follows:

$$\text{minimize:} \quad \frac{1}{2}||w||^2 + \frac{C}{|\tilde{\mathcal{X}}_n|}\sum_{x \in \tilde{\mathcal{X}}_n}\xi_x + \frac{C}{|\mathcal{X}_p|}\sum_{X \in \mathcal{X}_p}\xi_X \quad (1)$$

$$\text{subject to:} \quad w\,\phi(x) + b \leq -1 + \xi_x, \quad \forall x \in \tilde{\mathcal{X}}_n$$

$$w\frac{\phi(X)}{|X|} + b \geq \frac{2-|X|}{|X|} - \xi_X, \quad \forall X \in \mathcal{X}_p$$

$$\xi_x \geq 0, \xi_X \geq 0,$$

where C is a capacity control parameter, $\phi(X) = \sum_{x \in X}\phi(x)$ is a (possibly implicit) feature space representation of bag X, $|X|$ counts the number of instances it contains, which together yield the normalized sum of a positive bag's features $\frac{\phi(X)}{|X|}$.

This optimization is similar to that used in traditional supervised (single-instance) classification; however the second constraint explicitly enforces that at least one instance $\hat{x}$ from a positive bag should be positive. Ideally we would constrain the labels assigned to the instances to reflect precisely the number of true positive instances: $\sum_{x \in X} w\frac{\phi(x)}{|X|} \geq \sum_{x \in X}\frac{y(x)}{|X|} - \xi_X$, where $y(x) = -1$ for all $x \in X \smallsetminus \{\hat{x}\}$, and $y(\hat{x}) = +1$. The actual number of items in $\{\hat{x}\}$ is unknown; however, there must be at least one, meaning that the sum $\sum_{x \in X} y(x)$ is at least $2 - |X|$. Therefore, instead of tacitly treating all instances as positive (which would yield the constraint $w\frac{\phi(X)}{|X|} + b \geq +1 - \xi_X$ as in [14]), the linear term in the objective requires that the optimal hyperplane treat at least one positive instance in X as positive (modulo the slack variable ξ_X). That the righthand side of this inequality constraint is larger for smaller bags intuitively reflects that small positive bags are more informative than large ones.

This sparse MIL problem is still convex [5], and reduces to supervised SIL when positive bags are of size 1. While alternative MIL techniques would also be applicable [29, 1, 14], sMIL is conceptually most appropriate given that we expect to obtain some fairly low-quality and sparse image retrievals from the keyword search.

3.4. Iterative Improvement of Positive Bags

One limitation inherent to the sparse MIL objective above is that the constraints, while accurately reflecting the ambiguity of the positive instances' labels, also result in a rather coarse representation of each positive bag. Specifically, the second constraint of Eqn. 1 maps each bag to the mean of its component instances' representations, $\phi(X) = \frac{1}{|X|} \sum_{x \in X} \phi(x)$. This "squashing" can be viewed as an unwanted side effect of merging the instance-level constraints to the level of granularity required by the problem. We would prefer that a positive bag be represented as much as possible by the true positives within it. Of course, if we knew which images were true examples, the data would no longer be ambiguous!

To handle this circular problem, we propose an iterative refinement scheme that bootstraps an estimate of the bag sparsity from the image data alone. We first introduce a set of weights $[\omega_1, \ldots, \omega_{|X|}]$ associated with each instance in a bag X, and represent a positive bag as the weighted sum of its member instances: $\phi(X) = \frac{\sum_{i=1}^{|X|} \omega_i^{(t)} \phi(x_i)}{\sum_{i=1}^{|X|} \omega_i^{(t)}}$, where $\omega_i^{(t)}$ is the weight assigned to instance x_i in bag X at iteration t, and $|X|$ denotes the size of the bag. Initially, $\omega_i^{(0)} = \frac{1}{|X|}$, i.e., all instances in a bag are weighted uniformly. (Note that standard sMIL implicitly always uses these initial weights.)

Then, we repeatedly update the amount of weight each positive instance contributes to its bag's representation. After learning an initial classifier from the bags of examples, we use that function to label all training *instances* within the positive bags, by treating each instance as a singleton bag. The weight assigned to every instance x_i in positive bag X is updated according to its relative distance from the current optimal hyperplane. The weight at iteration t is computed as: $\omega_i^{(t)} = \omega_i^{(t-1)} e^{\frac{\|(y_i - y_m)\|}{\sigma^2}}$, where $y_i = w\phi(x_i) + b$, and $y_m = \operatorname{argmax}_{x_i \in X} y_i$. The idea is that at the end of each iteration, the bag representation used to solve for the optimal hyperplane (w and b) is brought closer to the instance that is considered most confidently to be positive. At the subsequent iteration, a new classifier is learned with the re-weighted bag representation, which yields a refined estimate of the decision boundary, and so on.

The number of iterations and the value of σ^2 are parameters of the method. Following [11], we set the number of iterations based on a small cross-validation set obtained in an unsupervised manner from the top hits from a single keyword search return. For each bag we set $\sigma^2 = c(y_m - y_n)$, where y_m and y_n are the bag's maximal and minimal classifier outputs, and c is a constant. This constant is similarly cross-validated, and fixed at $c = 5$ for all experiments.

3.5. Bags of Bags: Features and Classification

In our current implementation, we represent each image as a bag of "visual words" [6], that is, a histogram counting how many times each of a given number of prototypical local features occurs in the image. Given a corpus of unrelated training images, features are extracted within local regions of interest identified by a set of interest operators, and then these regions are described individually in a scale- and rotation-invariant manner (e.g., using the SIFT descriptor of [18]). A random selection of the collected feature vectors are clustered to establish a list of quantized visual words, the k cluster centers. Any new image x is then mapped to a k-dimensional vector that gives the frequency of occurrence of each word: $\phi(x) = [f_1, \ldots, f_k]$, where f_i denotes the frequency of the i-th word in image x.[2]

We have chosen this representation in part due to its success in various recognition algorithms [6, 11, 25], and to enable direct comparisons with existing techniques (see below). In our experiments, we compare the bags of words using a simple Gaussian RBF kernel. However, given that we have a kernel-based method, it can accommodate any representation for which there is a suitable kernel comparison $\langle \phi(x_i), \phi(x_j) \rangle$, including descriptions that might encode local or global spatial relationships between features, or kernels that measure partial matches to handle multiple objects.

After solving Eqn. 1 for a given category name, we iteratively improve the classifier and positive bag representations as outlined above. The classifier can then be used to predict the presence or absence of that object in novel images. Optionally, it can be applied to re-rank the original image search results that formed the positive training bags: the classifier treats each image as a singleton bag, and then ranks them according to their distance from the hyperplane.

4. Results

In this section we present results to demonstrate our method both for learning various common object categories without manual supervision, as well as re-ranking the images returned from a keyword search. We provide comparisons with state-of-the-art methods (both supervised or unsupervised) on benchmark test data, throughout using the same error metrics chosen in previous work. We use the following datasets, which we will later refer to by acronyms:

---

[2]Note the unfortunate double usage of the word *bag*: here the term bag refers to a single image's representation, whereas a positive bag of examples X will contain multiple bags of words $\{\phi(x_1), \ldots, \phi(x_{|X|})\}$.
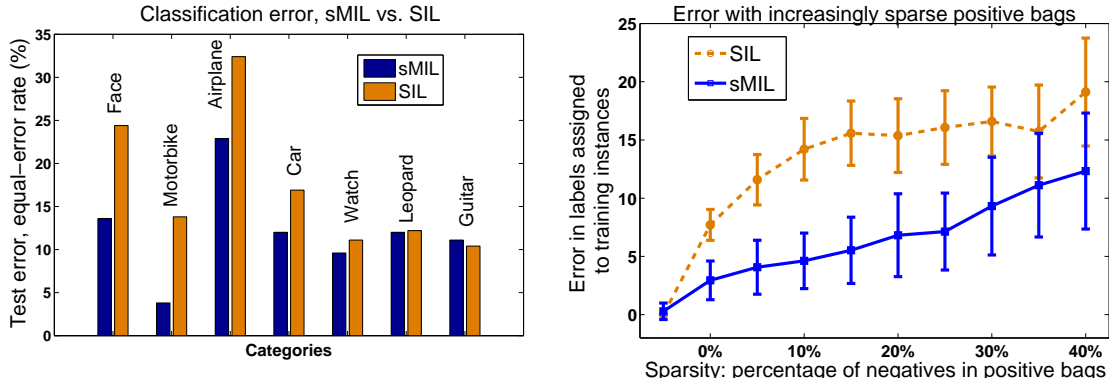
Figure 2. **Left:** Error comparison for our sMIL approach and an SIL baseline when trained with search engine images (CB) and tested on the Caltech-7 (CT). sMIL is noticeably more accurate. **Right:** Errors for the same methods for increasingly sparse positive bags. The two are equivalent for noise-free training examples. However, SIL's error quickly increases once false positives appear in the training set, whereas sMIL is designed to handle sparse bags, and its error grows much more gradually.

Caltech-7 test data (CT): a benchmark dataset containing 2148 images total from seven categories: Wristwatches, Guitars, Cars, Faces, Airplanes, Motorbikes, and Leopards. The dataset also contains 900 "background" images, which contain random objects and scenes unrelated to the seven categories. The test is binary, with the goal of predicting the presence or absence of a given category. Testing with these images allows us to compare with results reported for several existing methods.

Caltech-7 train data (CTT): the training images from the Caltech-7, otherwise the same as CT above.

Google downloads [11] (G): To compare against previous work, we train with the raw Google-downloaded images used in [11] and provided by the authors. This set contains on average 600 examples each for the same seven categories that are in CT. Since the images are from a keyword search, the true number of training examples for each class are sparse: on average 30% contain a "good" view of the class of interest, 20% are of "ok" quality (extensive occlusions, image noise, cartoons, etc.), and 50% are completely unrelated "junk", as judged in [11]. To form positive bags from these images, we must artificially group them into multiple sets. Given the percentage of true positives, random selections of bags of size 25 are almost certain to contain at least one.

Search engine bags for Caltech categories (CB): In order to train our method with naturally occurring bags as intended, we also download our own collection of images from the Web for the seven CT classes. For each class name, we download the top $n=50$ images from each of three search engines (Google, Yahoo, MSN) in five languages (English, French, German, Spanish, Italian), yielding 15 positive bags for each category. The choice of n was arbitrary and meant to take the first few pages of search results; after later trying a few smaller values we found our method's results varied insignificantly. The sparsity in these images appear to be similar to those of G. Negative instances for CB are taken from the CT background images or from the search returns of the other categories (as specified below).

Animal test data [4] (AT): a benchmark test set containing about 10,000 images total from 10 different types of animals. The images originated from a Google Image Search, and are thus quite noisy. The data and ground truth labels are provided from [4].

Search engine bags for Animals categories (AB): This set is just as CB above, except the searches are performed for the 10 classes in AT.

To represent the images from CT and G, we use local features provided by the authors of [11], which were taken from four interest operators; for the CB images we generated a similar bank of local features, and for AT and AB we use Harris-Affine and DoG interest operators for local features, plus global color histograms as described in [4]. Each region surrounding a detected interest point is described with a 72-d SIFT descriptor [18]. To form a visual vocabulary, we cluster a random sample of descriptors from 100 training images using k-means, with $k = 500$. Features from all detectors are merged into a single vocabulary. In all cases we use a Gaussian RBF kernel to compare images. The RBF and SVM parameters (γ and C) are selected in an unsupervised manner via cross-validation using one held-out bag, as described above.

4.1. MIL versus SIL

First we evaluate our MIL approach on the Caltech test set (CT) when it is trained with the CB bags. In some cases SIL—where all instances in positive bags are simply labeled as being positive—can achieve competitive results to MIL [22].

Amt. of manual supervision:	img labels	img labels	img labels +segment.	true img labels	none	none	none	none	none
Source of training data	CTT	CTT	CTT	G	CTT	CTT	G	G	CB
Method / Category	[12]	[20]	[16]	SIL-SVM	[25]	sMIL$'$	[11]	sMIL	sMIL
Airplane	**7.0**	11.1	-	4.9	*3.4*	5.0	15.5	**10.7**	22.9
Car (rear)	9.7	8.9	**6.1**	10.7	21.4	*5.4*	16.0	**11.8**	12.0
Face	*3.6*	6.5	-	21.8	**5.3**	11.5	20.7	23.1	**13.6**
Leopard	*10.0*	-	-	11.1	-	-	13.0	12.4	**12.0**
Motorbike	6.7	7.8	**6.0**	4.0	15.4	*3.8*	6.2	*3.8*	*3.8*
Guitar	-	-	-	*6.9*	-	-	31.8	*8.2*	11.1
Wrist watch	-	-	-	*7.3*	-	-	19.9	**8.9**	9.6
Average error	-	-	-	9.5	-	-	17.59	11.27	12.14

Figure 3. Comparison of the error rates and supervision requirements for the proposed approach and existing techniques (whether supervised or unsupervised) on the Caltech-7 image data. Error rates are measured at the point of equal-error on an ROC curve. Boxes with '-' denote that no result is available for that method and class. The best result for each category under each comparable setting is in bold, and the best result regardless of supervision requirements or training data is in italics (see text). Our approach is overall more accurate than prior unsupervised methods, and can learn good models both with highly noisy Caltech training data (sMIL$'$) and raw images from a Google search (sMIL). Methods learn the categories either from Caltech-7 images (CTT) or from web images (G, CB). All methods are tested with the Caltech-7 test set (CT).

Therefore, we also evaluate an SIL baseline, a binary SVM. Both methods are given one noisy positive bag to cross-validate and select their C parameters, though we found through more exhaustive cross-validation that SIL's accuracy remained similar whether C was chosen with noisy or noise-free examples. Figure 2 (left) compares the error rates on the CT test examples when both methods are trained with CB positives and background negatives. MIL is more accurate than SIL for all but one category, and on average its error is lower by 5 points.

To better understand the gains of our MIL approach in this setting, we next systematically analyze the effect of the sparseness of positive bags on categorization results for both SIL and sMIL. Positive examples from the CT are mixed with background images in different ratios to obtain positive bags of varying degrees of sparsity. The percentage of negative examples in the positive bags is varied from 10% to 90% (in steps of 10). Negative bags are constructed from the background images into bags of size 10. To measure error based on what the classifier has actually learned about the instance space, we consider the labels assigned to the instances composing the training bags when they are treated as singleton bags. Note that the error rate on the training *instances* is different from the training error, as the bags are the training examples. Figure 2 (right) shows the result: by design, sMIL is better equipped to handle sparse positive bags.

Both these results indicate the advantage of encoding the expected label ambiguity into the learning machine. While it is feasible to apply traditional SIL and hope the slack variables can cope, these results serve as evidence that MIL is better suited for discriminative learning from very ambiguous training data.

4.2. Caltech-7 benchmark

The CT dataset is a benchmark that has been used for several years, although primarily in supervised settings. Next we compare the errors of our approach with those reported by other authors using both supervised and unsupervised recognition techniques [12, 20, 16, 25] (see Figure 3). Note that while every single method has been tested on the same data, they vary in terms of their training supply (note the table row named "Source of training data"). In addition, the discriminative methods (sMIL,SIL, and [20]) learn from both positive and negative examples, whereas the others (minus [25]) do not encounter the random "background" images until test time. However, this is an advantage of discriminative methods in general, which focus specifically on distinguishing the classes rather than representing them.

The three rightmost columns correspond to the TSI-pLSA technique of Fergus et al. [11] and our sMIL approach, when trained with either the G or CB downloads. In this comparison our technique improves accuracy for every category. Of all seven categories, the G data for Airplanes happened to yield the sparsest positive bags—over 70% of the Airplane training examples do not contain a plane. For this class, the sMIL error is 4.8 points better than TSI-pLSA, which again illustrates the advantage of specifically accounting for sparsity in the data used to build the category model. Note that results are quite similar whether our method learns from the G or CB images (last two columns).

Presumably, being able to train with images from the same prepared dataset as the test examples is advantageous—whether or not those training examples are labeled—since the test and train distributions will be quite similar. Indeed, when training with the Web images, sMIL falls short of Sivic et al.'s pLSA clustering approach [25] for half of the classes. In order to make a comparison where sMIL also has access to unlabeled Caltech images, we generated an MIL training set as follows:

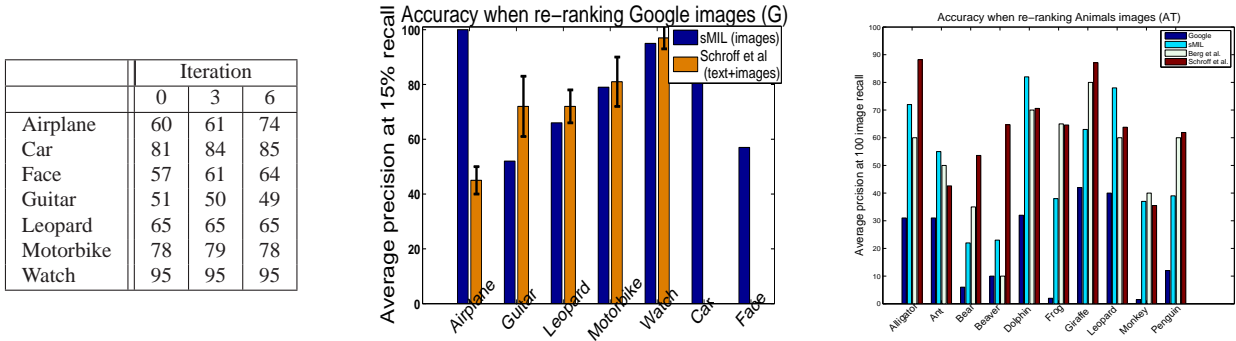|	Iteration		
	0	3	6
Airplane	60	61	74
Car	81	84	85
Face	57	61	64
Guitar	51	50	49
Leopard	65	65	65
Motorbike	78	79	78
Watch	95	95	95

Figure 4. Re-ranking results. **Left:** Refining positive bags: Precision at 15% recall over multiple iterations when re-ranking the Google (G) dataset. **Middle:** Comparison of sMIL and [24] when re-ranking the G images, with accuracy measured by the average precision at 15% recall. Both methods perform fairly similarly, although sMIL re-ranks the images based on image content alone, while the approach in [24] also leverages textual features. (Note, results are not provided for the last two categories in [24]). **Right:** Comparison of sMIL, Google's Image Search, [24], and [4] when re-ranking the AT images. The plot shows the precision at 100-image recall for the 10 animal classes. Our method improves upon Google's precision for all categories and outperforms all methods in three categories. (Best viewed in color.)

starting from pure CTT training sets for each class, we then add background images to each, to form a 50-50 mixture for each category's training set. Each such polluted training set is split into positive bags, and given to our method. We call this variant sMIL' in Figure 3. In this setting, our approach is overall more accurate than [25], by about five points on average. Note however that this pLSA technique is not defined for Web search data, and identifies the categories from one big pool of unlabeled images; our method may have some benefit from receiving the noisy images carved into groups.

Finally, in comparison to the three fully supervised techniques [12, 20, 16], our method does reasonably well. While it does not outperform the very best supervised numbers, it does approach them for several classes. Given that our sMIL approach learns categories with absolutely no manual supervision, it offers a significant complexity advantage, and so we find this to be a very encouraging result.

4.3. Re-ranking Keyword-Search Images

In these experiments, we use our framework to *re-rank* the Web search images by their estimated relevance.

Google Images of the Caltech-7 Categories. First we consider re-ranking the G dataset. Here we can compare our results against the SIL approach developed by Schroff et al. [24]. Their approach uses a supervised classifier to filter out graphics or drawings, followed by a Bayes estimator that uses the surrounding text and meta-data to re-rank the images; the top ranked images passing those filters are then used as noisily-labeled data to train an SVM. Our sMIL model is trained with positive bags sampled from G, while the method of [24] trains from G images and their associated text/tags. Both take negatives from the G images of all other categories.

Figure 4 (middle) compares the results. Overall, sMIL fares fairly comparably to the Schroff et al. approach, in spite of being limited to visual features only and using a completely automated training process. sMIL obtains 100% precision for the Airplane class because a particular airplane image was repeated with small changes in pose across the dataset, and our method ranked this particular set in the top. Our precision for Guitars is relatively low, however; examining sMIL's top ranked images and the positive training bags revealed a number of images of music scores. The unusual regularity of the images suggests that the scores were more visually cohesive than the various images of guitars (and people with guitars, etc.), and thus were learned by our method as the positive class. sMIL is not tuned to distinguish "ok" from "good" images of a class, so this accuracy measure treats the "ok" images as in-class examples, as does [24]. Similar to observations in [24], if we instead treat the "ok" images as negatives, sMIL's accuracy declines from 75.7 to 58.9 average precision. In comparison, Fergus et al. [11] achieve 69.3% average precision if "ok" images are treated as negatives; results are not given for the other setting.

Figure 4 (left) shows the precision at 15% recall for different numbers of iterations. Since sMIL gets 100% precision on Airplanes without refinement, we manually removed the near-duplicate examples for this experiment. As we re-weight the contributions of the positive instances to their bags, we see a notable increase in the precision for Airplanes, Cars, and Faces. For the rest of the classes, there is negligible change (±1 point). Figure 5 shows both the Face images our algorithm automatically down-weighted and subsequently removed from the top ranked positives, and the images that were reclassified as in-class once their weights increased. Examples with other classes are similar, but not included due to space limitations.

Google Images of the Animal Categories Finally, we performed the re-ranking experiment on the AT test images. Here we use both local features and the color histograms suggested in [4]. We simply add the kernel values obtained from both
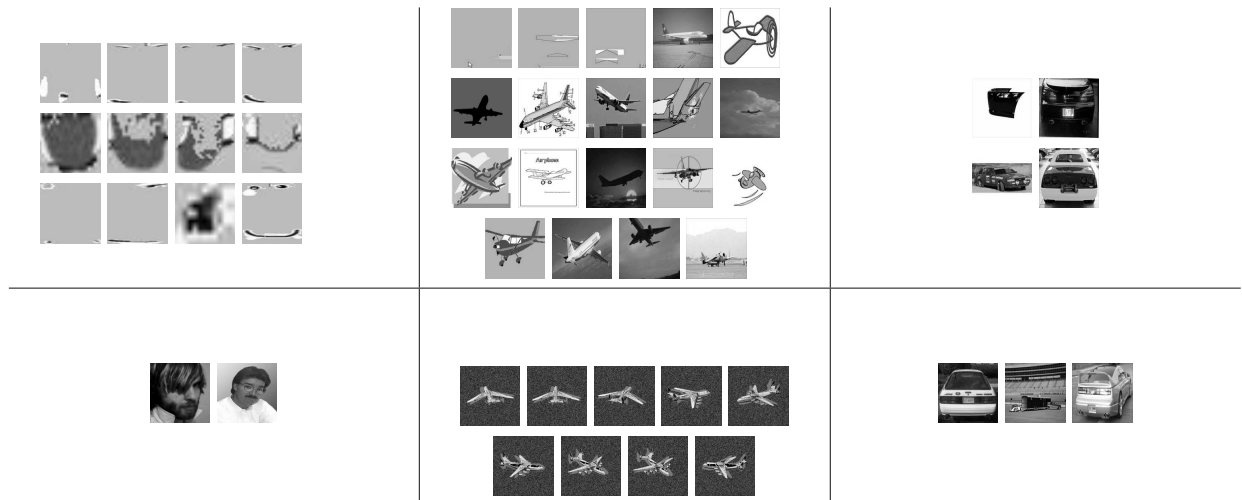
Figure 5. Outlier images (Top) are down-weighted by our refinement algorithm, while weights on better category exemplars increase (Bottom) and thereby improve the classifier. The rows show all images from the G Face, Airplanes and Cars set that move in and out of the 15% recall level before and after refinement, respectively.

feature types in order to combine them into a single kernel. Figure 4 (right) compares the precision at 100-image recall level for our method, the original Google Image Search, and the methods of Berg et al. [4] and Schroff et al. [24]. For all ten categories, sMIL improves significantly over the original Google ranking, with up to a 200% increase in precision (for *dolphin*). Even though [4] and [24] employ both textual and visual features to rank the images, our method performs similarly using image cues alone. In fact, for categories *ant*, *dolphin* and *leopard* our method outperforms both previous approaches by a good margin.

Conclusions. We have developed an MIL technique that leverages text-based image search to learn visual object categories without manual supervision. When learning categories or re-ranking keyword-based searches, our approach performs very well relative to both state-of-the-art unsupervised approaches and traditional fully supervised techniques. In the future we are interested in exploring complementary text features within this framework, and considering how prior knowledge about a category's expected sparsity might be captured in order to boost accuracy.

References

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support Vector Machines for Multiple-Instance Learning. In *NIPS*, 2002. 2, 4

[2] E. Bart and S. Ullman. Cross-Generalization: Learning Novel Classes from a Single Example by Feature Replacement. In *CVPR*, 2005. 2

[3] T. Berg, A. Berg, J. Edwards, and D. Forsyth. Who's in the picture? In *NIPS*, 2004. 2

[4] T. Berg and D. Forsyth. Animals on the Web. In *CVPR*, 2006. 2, 6, 8, 9

[5] R. Bunescu and R. Mooney. Multiple Instance Learning for Sparse Positive Bags. In *ICML*, 2007. 2, 4

[6] G. Csurka, C. Bray, C. Dance, and L. Fan. Visual Categorization with Bags of Keypoints. In *ECCV*, 2004. 5

[7] T. Dietterich, R. Lathrop, and T. Lozano-Perez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1-2):31–71, 1997. 2, 3

[8] P. Duygulu, K. Barnard, N. de Freitas, and D. Forsyth. Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary. In *ECCV*, 2002. 2

[9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html. 1

[10] L. Fei-Fei, R. Fergus, and P. Perona. Learning Generative Visual Models from Few Training Examples: an Incremental Bayesian Approach Tested on 101 Object Cateories. In *Workshop on Generative Model Based Vision*, 2004. 1, 2

[11] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning Object Categories from Google's Image Search. In *ICCV*, 2005. 1, 2, 5, 6, 7, 8

[12] R. Fergus, P. Perona, and A. Zisserman. Object Class Recognition by Unsupervised Scale-Invariant Learning. In *CVPR*, 2003. 1, 7, 8

[13] R. Fergus, P. Perona, and A. Zisserman. A Visual Category Filter for Google Images. In *ECCV*, 2004. 1

[14] T. Gartner, P. Flach, A. Kowalczyk, and A. Smola. Multi-Instance Kernels. In *ICML*, 2002. 2, 4

[15] K. Grauman and T. Darrell. Unsupervised Learning of Categories from Sets of Partially Matching Image Features. In *CVPR*, 2006. 2

[16] B. Leibe, A. Leonardis, and B. Schiele. Combined Object Categorization and Segmentation with an Implicit Shape Model. In *Workshop on Statistical Learning in Computer Vision*, 2004. 1, 7, 8

[17] L. Li, G. Wang, and L. Fei-Fei. Optimol: automatic online picture collection via incremental model learning. In *CVPR*, 2007. 1, 2

[18] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, 60(2), 2004. 5, 6

[19] O. Maron and A. Ratan. Multiple-Instance Learning for Natural Scene Classification. In *ICML*, 1998. 2

[20] A. Opelt, A. Fussenegger, and P. Auer. Weak Hypotheses and Boosting for Generic Object Detection and Recognition. In *ECCV*, 2004. 1, 7, 8

[21] J. Ponce, T. Berg, M. Everingham, D. Forsyth, M. Hebert, S. Lazebnik, M. Marszalek, C. Schmid, B. Russell, A. Torralba, C. Williams, J. Zhang, and A. Zisserman. Dataset Issues in Object Recognition. *Toward Category-Level Object Recognition, Springer-Verlag Lecture Notes in Computer Science*, 2006 (in press). 1

[22] S. Ray and M. Craven. Supervised versus Multiple Instance Learning: An Empirical Comparison. In *ICML*, 2005. 2, 3, 6

[23] B. Russell, A. Efros, J. Sivic, W. Freeman, and A. Zisserman. Using Multiple Segmentations to Discover Objects and their Extent in Image Collections. In *CVPR*, 2006. 2

[24] F. Schroff, A. Criminisi, and A. Zisserman. Harvesting Image Databases from the Web. In *ICCV*, 2007. 1, 2, 8, 9

[25] J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman. Discovering Object Categories in Image Collections. In *ICCV*, Beijing, China, October 2005. 1, 2, 5, 7, 8

[26] C. Yang and T. Lozano-Perez. Image database retrieval with multiple-instance learning techniques. In *ICDE*, 2000. 2

[27] T. Yeh, K. Tollmar, and T. Darrell. Searching the Web with Mobile Images for Location Recognition. In *CVPR*, 2004. 2

[28] C. Zhang, X. Chen, M. Chen, S. Chen, and M. Shyu. A multiple instance learning approach for content based image retrieval using one-class support vector machine. In *ICME*, 2005. 2

[29] Q. Zhang and S. Goldman. EM-DD: An Improved Multiple-Instance Learning Technique. In *NIPS*, 2002. 2, 4