INVITED PAPER

Learning Object Categories From Internet Image Searches

This paper shows how the results returned by an image search engine can be used to construct models from Internet images and use them for object recognition.

By Rob Fergus, Li Fei-Fei, Pietro Perona, *Member IEEE*, and Andrew Zisserman

ABSTRACT | In this paper, we describe a simple approach to learning models of visual object categories from images gathered from Internet image search engines. The images for a given keyword are typically highly variable, with a large fraction being unrelated to the query term, and thus pose a challenging environment from which to learn. By training our models directly from Internet images, we remove the need to laboriously compile training data sets, required by most other recognition approaches—this opens up the possibility of learning object category models "on-the-fly." We describe two simple approaches, derived from the probabilistic latent semantic analysis (pLSA) technique for text document analysis, that can be used to automatically learn object models from these data. We show two applications of the learned model: first, to rerank the images returned by the search engine, thus improving the quality of the search engine; and second, to recognize objects in other image data sets.

KEYWORDS | Internet image search engines; learning; object categories; recognition; unsupervised

I. INTRODUCTION

The Internet contains vast quantities of visual data in the form of images or video. Methods for searching and utilizing this rich resource have recently become a topic of interest within computer vision, being collectively known as "Internet vision."

To effectively search the visual content of the Internet, we must tackle a central problem in vision, that of recognizing the object categories present in an image. This task requires learning models of the visual appearance of object categories (e.g., dogs, tea cups), typically using data sets of images which have been manually "labeled" to indicate the objects present.

However, in Internet search, a myriad of queries are possible. This makes it impractical to manually gather the training images needed to build visual search models for every query.

In this paper, we describe two simple techniques that let us build visual search models directly from Internet images, without the need for manually labeled images. We leverage existing image search engines (which mainly use textual cues) to give us a set of images, some of which are visually related to query term. From these, we show how a robust visual model can be learned, which can then be used for a variety of applications. Crucially, our approach is fully automatic, hence we can learn models for *any* object.

One important application, which we explore in this paper, is to use the learned models to rerank the images returned by the search engine thus boosting its performance. The current performance of image search engines is rather variable, a direct consequence of them relying on weak textual cues. By using our visual models to improve their output, a large gain in search quality can be achieved. A second application that we explore is using the learned model to recognize the object in other image data sets, distinct from the original search engine images. A wide range of other applications are possible and we discuss these at the end of the paper.

Manuscript received April 7, 2009; revised September 23, 2009; accepted March 22, 2010. Date of publication June 10, 2010; date of current version July 21, 2010. This work was supported by the Caltech Center for Neuromorphic Systems Engineering (CNSE), the U.K. Engineering and Physical Sciences Research Council (EPSRC), European Union NOE PASCAL, the European Research Council (ERC) under Grant VisRec, and the U.S. Office of Naval Research (ONR) Multidisciplinary University Research Initiative (MURI) under Grants N00014-06-1-0734 and N00014-07-1-0182.
R. Fergus is with the Department of Computer Science, Courant Institute, New York University, New York, NY 10003 USA (e-mail: fergus@cs.nyu.edu).
L. Fei-Fei is with the Department of Computer Science, Stanford University, Stanford, CA 94305 USA.
P. Perona is with the Department of Electrical Engineering, California Institute of Technology (Caltech), Pasadena, CA 91125 USA.
A. Zisserman is with the Department of Engineering Science, University of Oxford, Oxford OX1 3PJ, U.K.
Digital Object Identifier: 10.1109/JPROC.2010.2048990

0018-9219/$26.00 ©2010 IEEE

The background to this work is the following: a wide range of different approaches has been taken to the challenging problem of object category recognition—[2], [11], [16], [25], [30], [32], [40]. All share a common assumption: the data used to train the system is labeled in some way. The level of labeling varies considerably: some require manual segmentation of object instances [26]; others require the centroid of each instance [40]; while some just require that each training image contains an instance of the object class to be learned [16]. The approaches also vary in the number of images required, ranging from dozens [12] up to thousands [25], [43]. However, they all assume the existence of a set of manually gathered collection of training images, each containing an instance of the object. In this paper, we address the problem of learning from *contaminated* data where a substantial portion of the training set images do not contain the object of interest, consisting instead of totally unrelated objects. From a machine learning perspective, the labels of the training data are considered to be noisy—while we expect them to be positive, there is a nonzero probability that they may be negative.

The motivation for investigating such a challenging learning scenario is that gathering and labeling images is time consuming and expensive and a constraint on progress. To obtain large sets of training data for more than a handful of categories, the vision community has been forced to adopt large collaborative efforts [10], [19], [21], [33], [41], [44], or leverage cheap web labor (e.g., Amazon's Mechanical Turk [37]). Such projects make the implicit assumption that the training of a recognition system only needs to be done once, justifying the time and effort expended. While this may be true in general, there are important cases where new models might need to be trained on-the-fly or from contaminated data. For example, home photo collections contain a diverse collection of objects, varying in their specificity [e.g., your girlfriend/boyfriend (whose appearance and clothing will evolve with time); your dog; people's backyards]. To search through your photos, e.g., to find all the photos containing a dog, existing approaches would require you to manually label a number of examples of each object from which a model could be trained. It would be more desirable if the computer could automatically learn these objects without your guidance—simply by typing "dog." Another application would be mobile robotics where a robot could learn frequently occurring objects as it explores its environment (contaminated data).

As will be shown, by removing the need for a consistently labeled set of training data, we are able to train directly from the loose collections of images typically returned by Internet image search engines. These use weak text-based cues (such as the filename or Alt text) to group collections of images, meaning that they frequently contain many different visual groups and a large portion of visually unrelated images. The search engines also track user click through and use this to rerank the images. However, since users rarely go beyond the first couple of pages of images, this reranking only affects a tiny fraction of the total images returned. Nevertheless, we show how we can use this small number of positive examples to help automate the learning procedure.

To train robust models we need many images, and since these search engines return many hundreds or thousands of images, despite the heavy contamination, several hundred positive examples exist. Nonetheless, it makes for a highly challenging learning environment. Fig. 1 shows a sample of images returned from Google's image search using the keyword "airplane," illustrating the diversity of the data. However, if one can succeed in learning from such noisy contaminated data the reward is tremendous: it enables us to automatically learn a classifier for any visual category, given only its name.

In this work, we use Google's image search[1] exclusively but any other image search engine may be used provided that the images can be gathered in an automated manner. Since all of the major Internet search engines use mainly text cues and user click through in their image search, their performance is similar.

A. Related Work

The methods we present are related to three areas of work: 1) discovering object categories within a collection of images; 2) training on the output of image search engines; and 3) learning categories with text-based cues.

The problem of object discovery in a large corpus of image data is that of extracting coherent components in an unsupervised manner, and has many parallels with problems in the field of textual analysis. A leading approach in this field is that of probabilistic latent semantic analysis (pLSA) [20] and its hierarchical Bayesian form, latent Dirichlet allocation (LDA) [4]. Recently, these two generative approaches have been applied to computer vision: Fei-Fei and Perona [13] applied LDA to scene classification and more relevantly, Sivic *et al.* applied pLSA to unsupervised object categorization. In the latter work, the California Institute of Technology (Caltech, Pasadena) data sets used by Fergus *et al.* [16] were combined into one large collection and the different objects extracted automatically using pLSA. The drawback to these schemes is that they incorporate no spatial information, being a simple visual bag-of-words model [8], [36].

A variety of papers have employed and extended these methods. Quelhas *et al.* [31] and Bosch *et al.* [5] perform scene classification. Spatial information has been incorporated loosely in [17], for unsupervised discovery in data sets such as PASCAL [10], and by Lazebnik *et al.* [24] who use spatial pyramid matching in the Caltech data sets [19]. Sudderth *et al.* [38], [39] construct hierarchical scene-object-part models based on LDA.

[1]http://www.google.com/imghp.

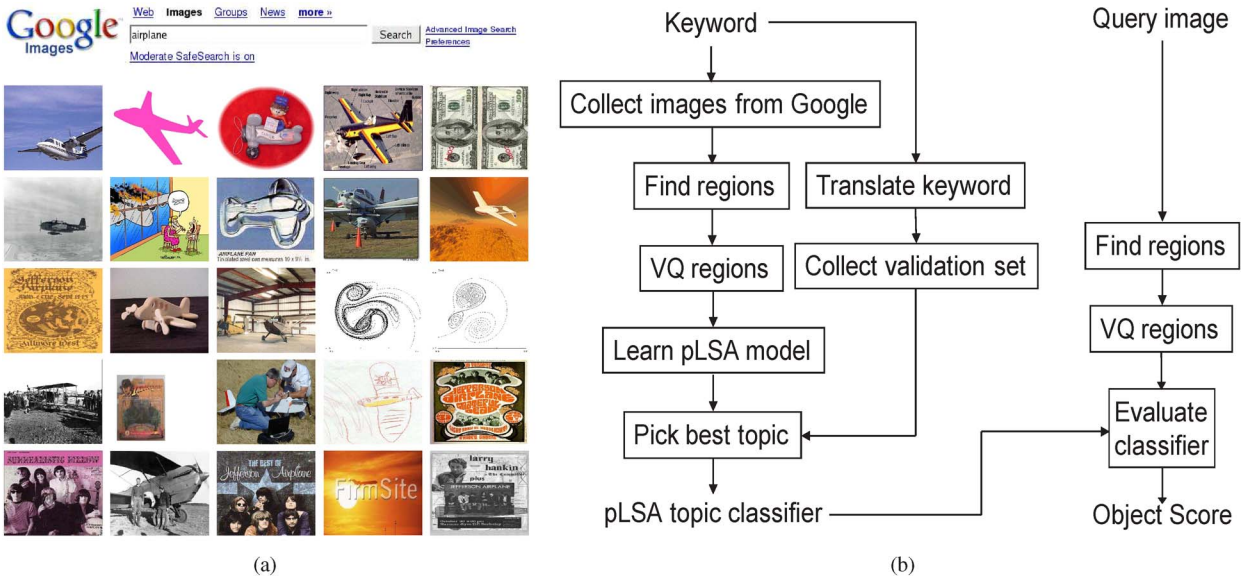


Fig. 1. ***(a) Images returned from Google's image search using the keyword "airplane." This is a representative sample of our training data. Note the large proportion of visually unrelated images and the wide pose variation. (b) An overview of the training and testing procedures for the pLSA-based model introduced in this paper.***

For the most part, existing work has used training data which were cleanly divided into separate groups: each image contained one instance of an object. The Google data are more varied in that there may be one or more coherent visual components (due to polysemes, e.g., "iris" can be iris flower, iris eye, Iris Murdoch) with the remaining images either vaguely related to one of the visual components (e.g., cartoon depiction of the object), or totally unrelated. This makes the learning problems harder and the data are no longer a set of tightly grouped clusters.

In this paper, we will show that the pLSA model can be applied successfully to this type of data, and use the pLSA model to illustrate the idea of learning visual object models from Google image data. Previous work has used a variety of methods to learn from such internet data. Berg and Forsyth [3] overcome the limitations of Google's image search by using text on the original web pages to extract further contextual cues. By using these in conjunction with the image features they demonstrate how large sets of animal images can be gathered from the web, although some manual intervention is required. Schroff *et al.* [34] extend this method to be fully automatic by first ranking the images based on the text and other metadata, and then learning a visual model discriminatively from the highly ranked images by using a support vector machine (SVM) classifier to cope with the noisy image labels. Vijayanarasimhan and Grauman [42] use multiple-instance learning techniques to overcome the labeling noise in Internet images. Li *et al.* [27] show how methods, similar to those introduced in this paper, can be used in incremental fashion to compile a data set of a desired class from the Internet. Collins *et al.* [7] use an active learning approach to rapidly build up a large data set from Internet images, using a human-in-the-loop with the recognition model.

The techniques presented in this paper are closely related to these approaches. Indeed, methods from the papers above could also be applied to the tasks we explore. However, an attraction of the pLSA approach is that it is simple and straightforward to implement, while still addressing many of the important elements of the problem.

Other work has focused on the joint learning of text and images. Barnard *et al.* [1] present a method where models are learned not just from images but also accompanying text labels. Each image is oversegmented using normalized cuts to give large number of regions. The regions are represented by vectors encoding low-level concepts such as color and area. The vectors from each image are modeled jointly with the text labels, establishing a correspondence between the two. Hence, in a recognition scenario, given one the other can be predicted. Carbonetto *et al.* [6] also consider the text and images problem but here use sparse kernel methods to determine sets of features related to each object class.

Both these works assume that text annotations are available for the training set. The text cues given by the Internet search engines are very crude: the same keyword for every image returned, thus these techniques are not directly applicable to our problem.

II. OVERVIEW

In this paper, we investigate the ability of a pLSA-based model, based on Sivic *et al.* [35], to successfully learn from

Google images. We first introduce the standard pLSA model, which does not use any form of location information, and then extend pLSA to include location information in a straightforward manner to address this shortcoming. For this extension, we use a simple spatial model that captures only the absolute location of objects within the image using a discrete grid. At first sight, this would seem to be rather limited: if the object were to shift significantly within the image, then the representation would change. However, a quirk of our training images results in this happening far less than might be expected. For the most part, images containing good examples have been captured by a human photographer who has centered and filled the frame with the object of interest, thus giving a fairly stable position and scale (although the aspect is highly variable). Furthermore, search engines rely on textual cues such as the filename, which often reflects the dominant object in the image. Hence, while the images overall are highly variable, some images will contain approximately centered instances of the object filling the frame. See Fig. 6 for examples of the phenomenon.

We investigate two different scenarios involving images from Google. In the first scenario, we train models directly on Google images and use them to rerank the training images, using the likelihood of each image under the model learned. Hopefully the good images will score highly, while the junk (visually unrelated) images score poorly, resulting in the first few reranked pages containing more good examples than they originally did, thus improving the performance of Google's image search.

The second, more ambitious scenario, is to train models on the Google images and then to test them on other data sets on the web—the Caltech and PASCAL VOC [45] data sets—to see how they compare to existing methods, trained on manually prepared (uncontaminated) data. This open world evaluation requires stronger models than the first scenario since they will be used in a more general setting.

The structure of the paper is as follows. In Section III, we introduce the pLSA framework of Sivic *et al.* and detail our extension to their scheme. Having described the two approaches, in Section IV, we look at the specific issues in applying them to the Google data. We describe the visual features in Section V. In Section VI, we apply the algorithms to learning from the Google data, evaluating them under both scenarios. Finally, in Section VII, we discuss their relative benefits and draw conclusions.

III. PROBABILISTIC LATENT SEMANTIC ANALYSIS (pLSA)

A problem of significant interest in the text analysis community is that of extracting coherent components, such as topics, from a large corpus of documents. Latent semantic analysis (LSA) is an approach whereby each document is represented by a histogram of word counts over a vocabulary of fixed size. The histograms from all documents in the corpus form a large co-occurrence matrix which is then decomposed using singular value decomposition, with the eigenvectors corresponding to different topics within the corpus and the eigenvalues giving their relative weighting.

Hofmann [20] placed LSA in a probabilistic context, calling it pLSA. Each document is modeled as a mixture of Z topics, each topic being a distribution over the vocabulary of words. More formally, we have a set of D documents, each modeled as a histogram of word counts over a vocabulary of size W. The corpus of documents is represented by a co-occurrence matrix of size $W \times D$, with entry $n(w, d)$ listing the number of words w in document d. Document d has N_d words in total. The model has a single latent topic variable z associating the occurrence of word w to document d

$$P(w, d) = \sum_{z=1}^{Z} P(w|z)P(z|d)P(d). \tag{1}$$

Thus, we are decomposing a $W \times D$ matrix into a $W \times Z$ matrix and a $Z \times W$ one. $P(w|z)$ captures the co-occurrence of words within a topic, while $P(z|d)$ gives the weighting of each topic within a document. The graphical model is shown in Fig. 2(a).

The densities of the model $P(w|z)$ and $P(z|d)$ are learned using an alternating optimization scheme known

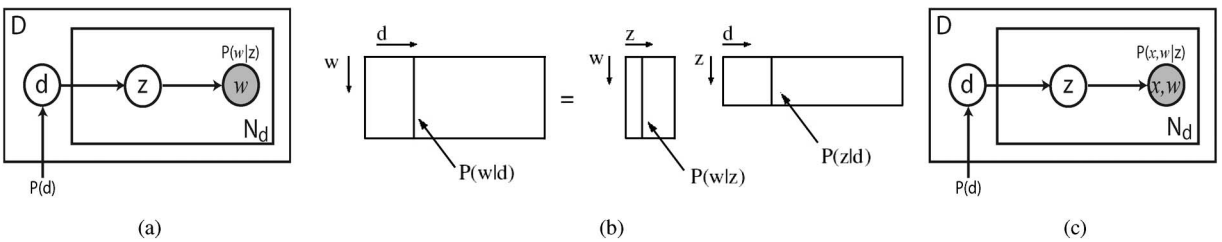


Fig. 2. *(a) Graphical model of pLSA. (b) Learning a pLSA model consists of factoring the matrix $P(w|d)$ into two smaller ones: $P(w|z)$ and $P(z|d)$. See text for details. (c) Graphical model of ABS-pLSA. It is trained in a similar fashion to pLSA.*

as expectation–maximization (EM). The E-step computes the posterior over the topic $P(z|w,d)$ and then the M-step updates the densities

$$\text{E-step:} \quad P(z|w,d) = \frac{P(w|z)P(z|d)}{\sum_{z'} P(w|z')P(z'|d)} \tag{2}$$

$$\text{M-step:} \quad P(w|z) \propto \sum_{d=1}^{D} n(w,d)P(z|w,d) \tag{3}$$

$$P(z|d) \propto \sum_{w=1}^{W} n(w,d)P(z|w,d). \tag{4}$$

This maximizes the likelihood of the model over the data

$$L = \prod_{d=1}^{D}\prod_{w=1}^{W} P(w,d)^{n(w,d)}. \tag{5}$$

A novel document d^* is classified by running EM with $P(w|z)$ fixed, computing $P(z|d^*)$, the mix of topics within the image.

A. Applying pLSA to Visual Data

While pLSA and LDA model text documents, we can use them to model images by using the analogies (as proposed by Sivic *et al.* [35] and Fei-Fei and Perona [13]), which may be summarized as follows: a document d corresponds to an image; a word w corresponds to a visual word; and finally, a topic z corresponds to an object. These terms are self-evident except for *visual words* which we now explain.

1) Visual Words: Both Sivic *et al.* and Fei-Fei and Perona adopt the same procedure whereby a set of regions is extracted from the image using a feature detector that finds salient localized regions. Then, the appearance of each localized region is vector quantized to a prebuilt vocabulary of *visual words* [36]. No location information is taken from the regions, the image being represented solely by a histogram of visual words.

The visual vocabulary is built by running k-means on a large set of regions from an independent set of training images of widely varying content. The size of the vocabulary is a specified parameter of the system, with Sivic *et al.* and Fei-Fei and Perona using W in the range 200–2000. The feature detectors and descriptors used are detailed in Section V. This process is illustrated in Fig. 3.

2) An Example: To consolidate our description, we give a toy example. Fig. 4(a)–(c) shows the results of a two topic model trained on a collection of images of which 50% were airplanes from the Caltech data sets and the other 50% were background scenes from the Caltech data sets. Learning is performed in an unsupervised manner in that image labels (airplane or background) were not provided to the algorithm. The regions are colored according to the most likely topic of their visual word [using $P(w|z)$]: red for the first topic (which happens to pick out the airplane images) and green for the second (which picks out background images). $P(z|d)$ is shown above each image.

B. Adding Location Into the pLSA Model

Having described how pLSA may be applied to visual data, we now introduce location information into the model. We choose to apply these changes to pLSA rather than LDA since the former is simpler in nature. The benefits of LDA are marginal since, for the most part, we do not intend to train the model from a small number of images, making the priors in LDA irrelevant. However, the proposed changes could easily be made to LDA also.

A straightforward way to incorporate location is to quantize the location within the image into one of X bins and then to have a joint density on the appearance and location of each region. Thus, $P(w|z)$ in pLSA becomes $P(w,x|z)$, a discrete density of size $(W \times X) \times Z$

$$P(w,x,d) = \sum_{z=1}^{Z} P(w,x|z)P(z|d). \tag{6}$$

We denote this model ABS-pLSA. The graphical model is shown in Fig. 2(b).

The same pLSA update equations outlined above can be easily applied to this model in learning and recognition. The problem with this representation is that it is not translation or scale invariant at all, since x is an absolute coordinate frame.

C. Recognition Using the pLSA and ABS-pLSA Models

To classify a test image, interest points are found and vector quantized in the same manner as in training. Then, the EM equations (4) are iterated, holding $p(w|z)$ fixed [$p(w,x|z)$ in the case of ABS-pLSA]. Hence, only the weighting of the different topics within the image $p(z|d)$ is inferred. Given a particular topic z^* (for details on how this can be done automatically, see Section IV-B), the classification confidence is thus given by $p(z^*|d)$.

Although we restrict ourselves to classification in this paper, if the test images contain objects whose scale and location vary considerably, both models can be employed in a sliding-window fashion, whereby a small subwindow of the image is cropped out and classified independently.

IV. APPLICATION TO GOOGLE DATA

The pLSA-based approaches outlined above have previously been used in weakly supervised settings where the

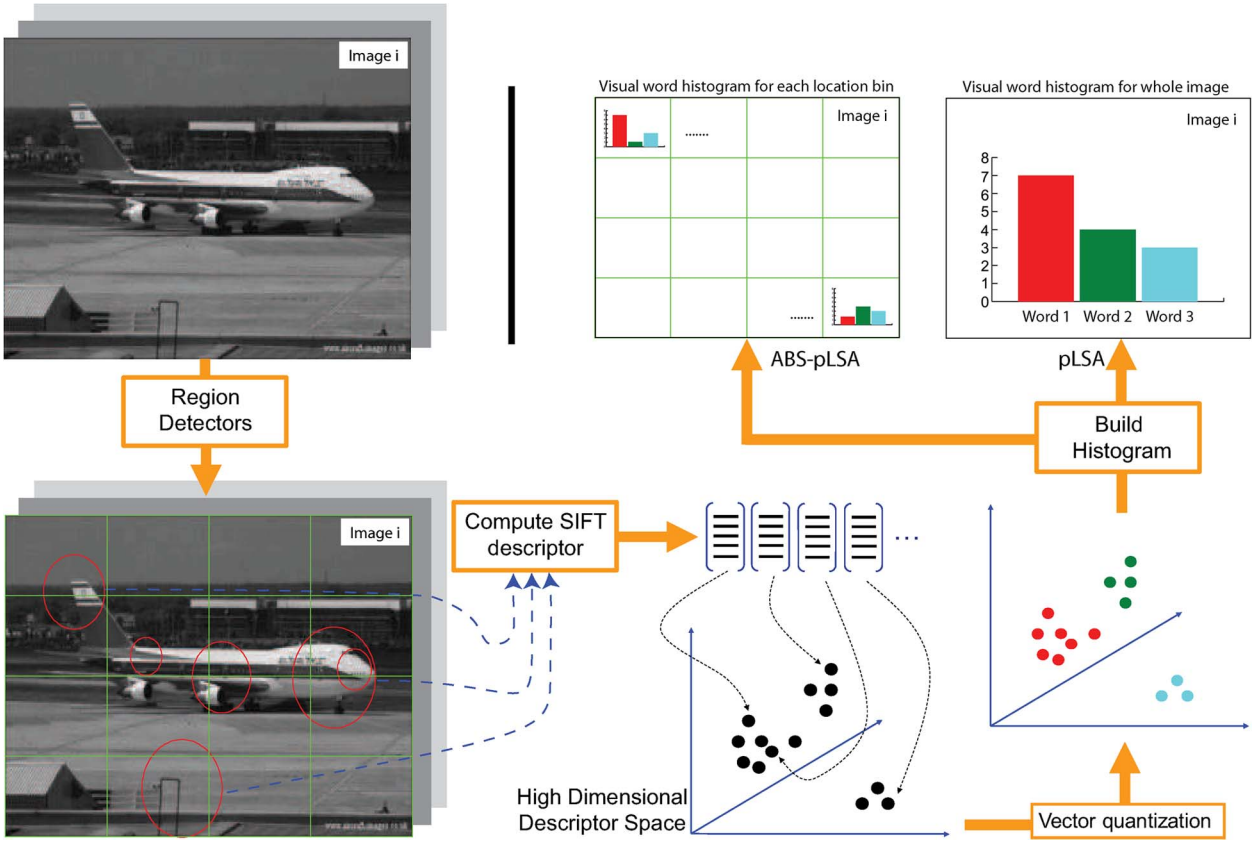


Fig. 3. *Our image representation based on visual word histograms. Counterclockwise from top left: Region detectors are applied to a large set of images, where salient regions are localized (in reality, many hundreds of regions are output per image). Each region is represented by a SIFT descriptor which corresponds to a point in the high-dimensional descriptor space. All descriptors from all images are then vector quantized using a precomputed set of cluster centers ($W = 350$ in our case). Each cluster center corresponds to a visual word. Then, a histogram is built over the visual words. In pLSA, location is ignored and the histogram is formed using all regions in the image. In ABS-pLSA, we use a discrete location grid and build a separate histogram within each grid cell, using only the regions that occur within it.*

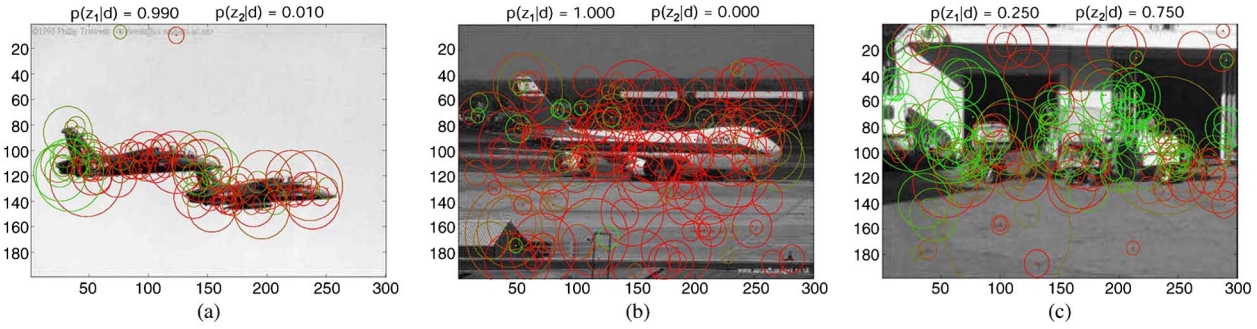


Fig. 4. *(a) and (b) Two airplane and (c) one background image, with regions superimposed, colored according to topic of a learned pLSA model. Red corresponds to topic 1; green to topic 2. The color of each circle is given by the probability of belonging to each topic, based on the regions' visual word (e.g., a brown circle means the region is equally likely to belong to both topics). As is evident, topic 1 corresponds to airplanes and topic 2 to the background. Only a subset of regions are shown for clarity.*

images were known to contain an instance of the object to be learned. In applying the methods to search engine images, issues arise due to the unsupervised nature of the problem and the labeling noise of the data. In particular, the methods require a validation set to make important model design decisions but it is not clear how such a set

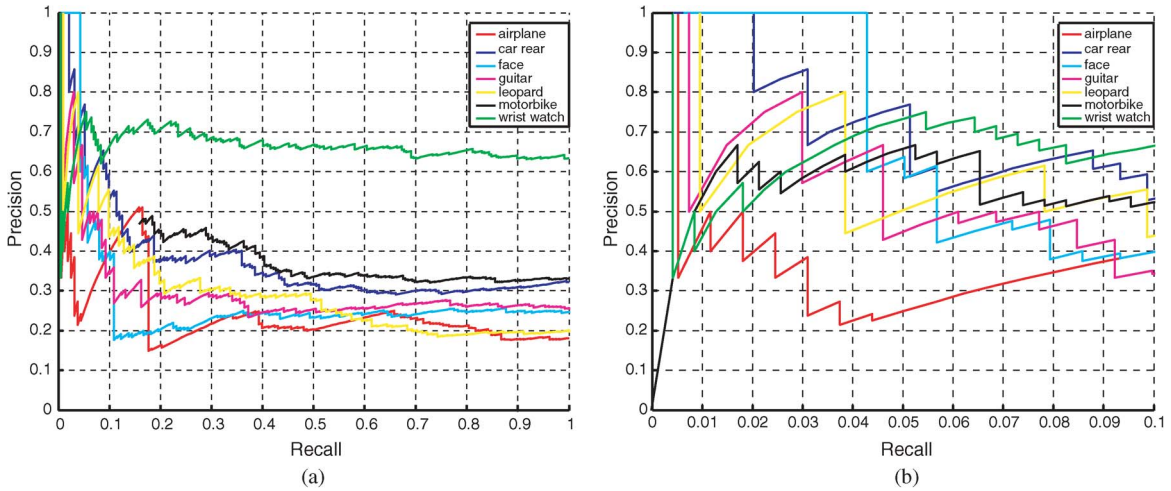


Fig. 5. *Recall precision curves of the raw output of Google's image search for seven keywords. Good labels count as positive examples while intermediate and junk labels are negative examples (the three classes of label are defined in the text). (a) Shows recall from 0 to 1, while (b) zooms in, showing the recall from 0 to 0.1. As each data set is roughly 500 images in size, with the fraction of good images around 25% meaning that 0.1 recall corresponds to around 12 good images. Note that the precision drops rapidly as the recall increases, leveling out at 20%–30% for most categories.*

may be obtained, given the lack of labeled data. We now describe a novel way to obtain a small set of images which may be used as a noisy validation set and then detail the application of both approaches to the Google data.

A. Automatic Gathering of Noisy Validation Set

We make the empirical observation (as seen in Fig. 5) that the first few pages returned by Google tend to contain more good images than those returned later on.[2] The idea is that we assume that the images from these first pages are positive examples, and hence may be used as a validation set to make model selection choices in our experiments. The catch is that the dropoff in quality of Google's search is so steep that only the first few images of the first page are likely to be good examples.

Using Google's automatic translation tool [18], we obtain the translations of the user's keyword in the following languages: German, French, Spanish, Italian, Portuguese, and Chinese. Since each translation returns a different set of images, albeit with the same dropoff in quality, we automatically download the first few images from each different language, and combine them to give a validation set of a reasonable size without a degradation in quality. Although automatic translation tools are far from perfect, the fact that the keyword is a noun usually means that there is a unique translation, hence making the process more reliable. The idea of using linguistic translation of query terms to assist with visual search has also been explored by Etzioni *et al.* [9].

[2]The images were downloaded from Google in June 2005. All data sets used in this paper may be downloaded from http://cs.nyu.edu/~fergus/.

Using seven different languages (including English), taking the first five images, we can obtain a validation set of up to 35 images (since languages may share the same word for a category and we reject duplicate images). Note that this scheme does not require any supervision. Fig. 6 shows the validation sets for airplane and motorbike.

B. Picking the Number of Topics

The example in Section III-A2 only required two topics, but in general, more topics are required to model the variety of the data. The choice of the number of topics brings up two additional issues when training our models on images from Google: 1) the optimal number of topics Z; and 2) which subset of these topics should be used to form a classifier for use in testing. A larger number of topics will result in more homogeneous topics at the expense of their ability to generalize. Given the varied nature of images obtained from Google, a large number of topics might seem appropriate, but this raises the issue of how to pick the topics corresponding to the good images, while ignoring topics which model the junk images within the data set.

For a given keyword, the validation set used to pick topics from a model. In view of the small size and imperfect quality of the validation set, we limit ourselves to picking a single topic from the larger model. While this is a suboptimal strategy, it is all that can be done reliably given the lack of labeled data.

In our experiments, we used $Z = 8$ topics, this being the largest number from which we can reliably pick the single topic that corresponds to the good images. For more details on these design choices, see [14].

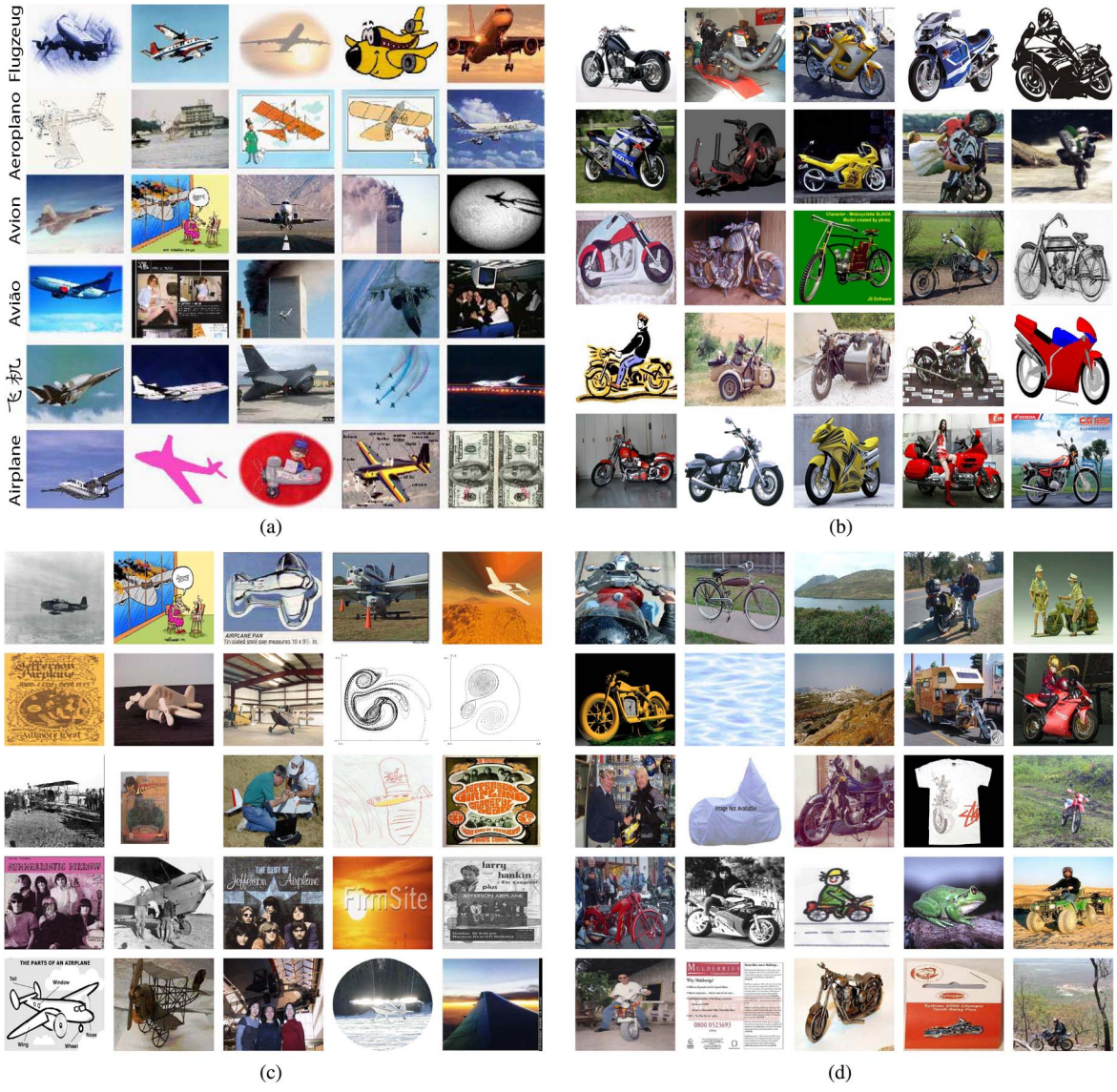


Fig. 6. *(a) The entire validation set for "airplane" obtained automatically using Google's translation tool and Google's image search. The text by each row shows the translated keyword used to gather that particular row. The quality of the images is noticeably higher than in the random sample of airplane images downloaded from Google, shown in (c). (b) The entire validation set for motorbikes. Again, the portion of good images is considerably higher than a random page of images, shown in (d).*

V. FEATURE DETECTORS AND REPRESENTATION

Since we wish to apply our methods to a wide variety of object categories, we use a variety of different object detectors to ensure a good coverage of the image, regardless of its content.

We use four different types of circular region detector: 1) Kadir & Brady saliency operator [23]; 2) multiscale Harris detector [29]; 3) difference of Gaussians, as used by Lowe [28]. The fourth type is an edge-based operator, designed for certain categories where edge information is important since the other types of detectors do not adequately capture this information. The details of this operator can be found in [14]. In total, all four types of detector produce around 700 regions/image.

Following the work of Sivic *et al.* [35] and others, we describe each region using the SIFT descriptor [28], using 72 dimensions rather than the usual 128, resulting in larger histogram bins which are more appropriate for object categorization.

The descriptors are then vector quantized using a fixed codebooks of visual words, precomputed using k-means from a large set of images drawn from the training sets of a large number of different categories. A separate codebook was formed for each feature type and then combined to give a total of $W = 350$ visual words.

VI. EXPERIMENTS

A. Data Sets

For seven categories, a set of images was automatically downloaded from Google's image search using the keywords: airplane, cars rear, face, guitar, motorbike, leopard and wrist watch. Although in this paper Google's image search was used exclusively, any other image search engine may be used provided that the images can be gathered in an automated manner, using the category name. Duplicates images and very small images (< 100 pixels in width) were discarded and Google's SafeSearch filter was left on, to reduce the proportion of unrelated images returned. For assessment purposes, the images returned by Google were divided into three distinct groups.

1) Good images: these are good examples of the keyword category, lacking major occlusion, although there may be a variety of viewpoints, scalings, and orientations.
2) Intermediate images: these are in some way related to the keyword category, but are of lower quality than the good images. They may have extensive occlusion; substantial image noise; be a caricature or cartoon of the category; or the object is rather insignificant in the image, or some other fault.
3) Junk images: these are totally unrelated to the keyword category.

The labeling was performed by an individual who was not connected with the experiments in anyway, possessing no knowledge of our algorithms. Table 1 shows the relative portions of labels for each of the seven categories. Additionally, a background data set was collected from Google, using the keyword "things." This gave a large collection of highly variable images.

We also use two open-world data sets to explore the performance of the models trained on Google images. The first are seven classes from the Caltech data set [15], matching the classes collected from Google and listed above (see examples in Figs. 9 and 10). These are relatively straightforward in that the objects tend to fairly centered and large within the image. The second, more challenging data set, consists of two classes from the PASCAL VOC 2005 [11]: motorbikes and cars. Note that we use the more challenging test set 2 in both cases, where the object pose is highly variable. For both these open-world data sets, we restrict ourselves to a classification task (i.e., object present/absent within image), making no attempt to localize the object.

B. Improving Google's Image Search

To improve Google's image search with the pLSA methods, we first train a model on Google data, then choose a single topic with the validation set and use it to rerank the entire Google data set. An overview of the whole training procedure is given in Fig. 1.

In Figs. 7 and 8, we can see what each topic in pLSA and ABS-pLSA has learned from the motorbike, and cars rear Google data sets. Note that each topic clusters visually consistent images: some correspond to the object; others to images of text or other junk. Some of the ABS-pLSA models seem to be clustering object instances by aspect (e.g., motorbikes). Qualitatively, the addition of location information seems to improve the consistency of each topic. The performance improvement is quantified in Table 1, where the precision at 15% recall (corresponding to a couple of web pages) is recorded for the two pLSA methods and the raw Google ranking.

Table 1 shows pLSA and ABS-pLSA improving on the raw Google precision, in many cases significantly. The method using location information, ABS-pLSA, generally outperforms pLSA.

C. Open World Experiments

The models used to improve Google's image search are now tested on other data sets, enabling us to see what performance penalty is incurred when training on contaminated data.

For such experiments, we use the Caltech data sets matching the keywords entered into Google's image search. In Figs. 9 and 10, we show ABS-pLSA models trained on

Table 1 Column 2: Number of Images Collected From Google's Image Search, Column 3: Size of Validation Set. Columns 4, 5, and 6: Breakdown of Google Images for Each Class. Note the Low Proportion of Good Examples Present in the Majority of Categories. Remaining Columns: Precision at 15% Recall for Different pLSA-Based Methods of Reranking Images Returned by Google's Images Search. Good Images Are Taken as Positive Examples, While Intermediate and Junk Images as Negative Examples. The Raw Google Ranking Is Compared to the Three pLSA-Based Methods. 15% Recall Corresponds to a Couple of Web Pages Worth of Images

Dataset	Train ims.	Valid. ims.	% Junk	% Inter.	% Good	Raw Google	pLSA	ABS-pLSA
Airplane	874	30	73.3	8.6	18.1	0.50	1.0	1.0
Cars Rear	596	30	54.9	12.9	32.2	0.41	0.88	0.94
Face	564	30	54.4	21.3	24.3	0.19	0.53	0.64
Guitar	511	25	44.2	30.5	25.3	0.31	0.39	0.40
Leopard	516	15	52.9	27.5	19.6	0.42	0.34	0.38
Motorbike	688	30	36.8	29.8	33.4	0.46	0.46	0.55
Wrist watch	342	35	22.8	13.8	63.4	0.70	0.83	0.97

Fig. 7. *(a) pLSA applied to images collected from Google using the "motorbike" keyword. Each column shows the top ten images for a given topic. The colored dots indicate the ground-truth label (only used for assessment purposes): green—good; yellow—intermediate; red—junk. Note the visual consistency of most topics: some consist mainly of motorbikes while others cluster figures and diagrams with a white background. The validation set automatically selects topic 7, containing many good motorbike images, as the best topic. (b) As for (a) but using ABS-pLSA instead of pLSA. The addition of location information increases visual consistency, with different aspects of motorbikes now being separated into two different topics.*

Fig. 8. *(a) pLSA and (b) ABS-pLSA applied to images collected from Google using the "car rear" keywords. Topics 1 and 4, respectively, were chosen by the validation set. Both these topics contain a high proportion of good images (indicated by the green dots). In (b), the addition of location information means that the different viewpoints of the car are clustered by the different topics.*

Google data being tested on Caltech images. These are the same models used in Section VI-B. The location densities in the figures give some idea of the tightness of the model, which appears to be surprisingly good—better in some cases than the models trained directly on Caltech data.

Table 2 shows the performance on a classification task (object versus background class of Google "things" images) of the pLSA-based methods when tested on the Caltech data sets (having been trained on Google data). For the most part, a significant drop in performance is

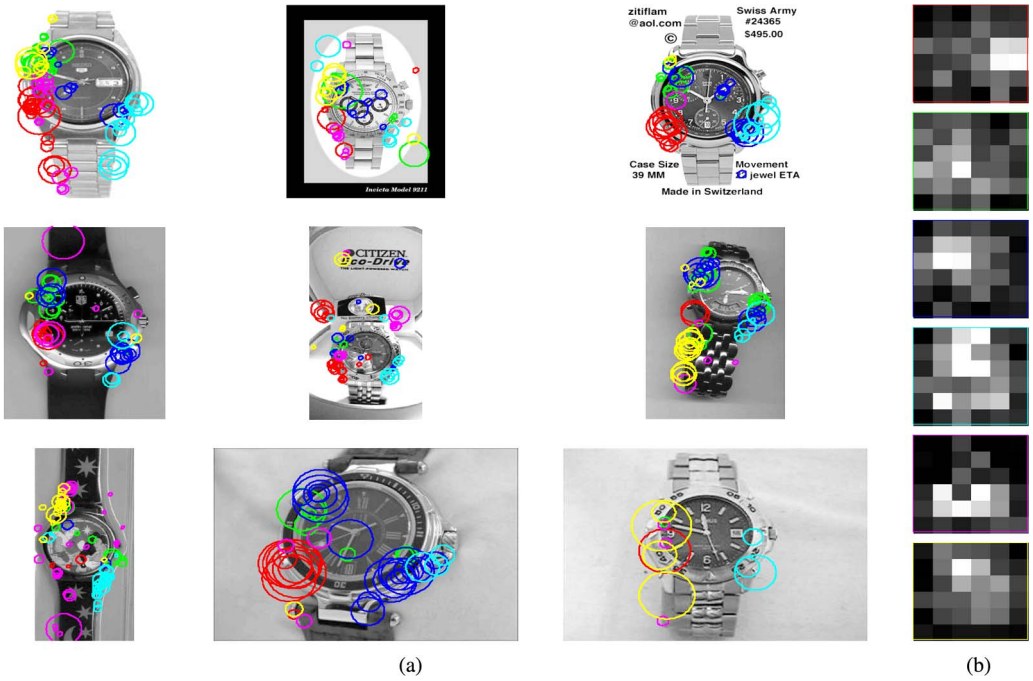


Fig. 9. *An insight into what the model has learned. Watches from the Caltech data set, with regions superimposed that belong to the six most common visual words (irrespective of location) from the automatically chosen topic of the Google-trained ABS-pLSA watch model. Each color shows regions quantized to a different visual word. The circular bezel of the watch face is picked out. Due to the rotation sensitivity of our region representation, different parts of the bezel are quantized to different words. (b) The location densities in the subwindow of the six most common words shown in (a), ordered from red to yellow. White corresponds to a high probability, black to a low one.*

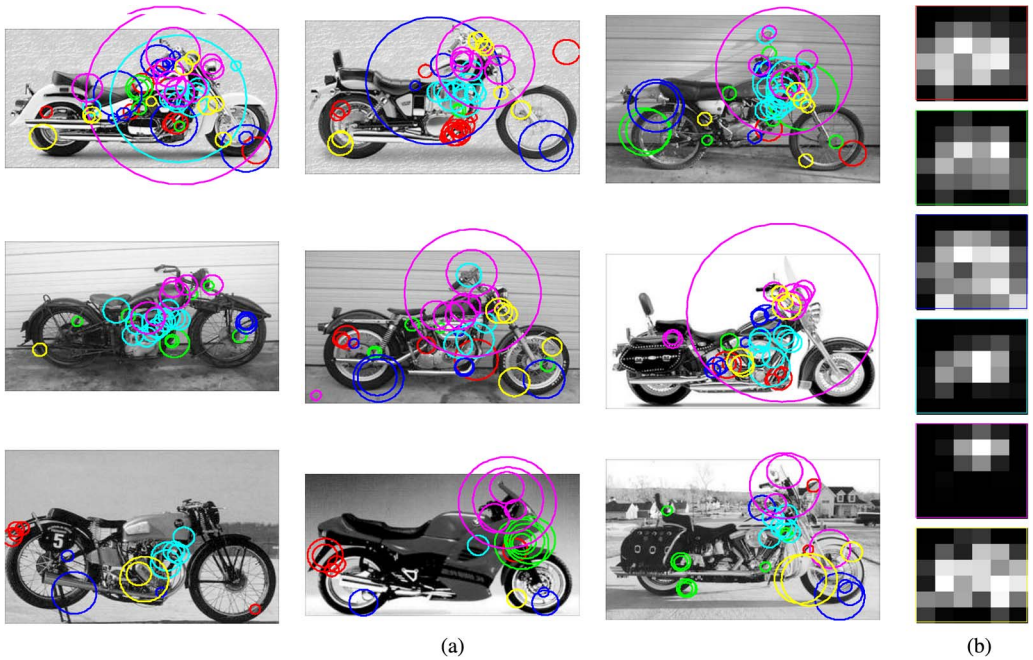


Fig. 10. *Similar to Fig. 9, examples from the Motorbike class are shown. Regions are superimposed that belong to the six most common visual words from the automatically chosen topic of the Google-trained ABS-pLSA motorbike model. Each color shows regions quantized to a different visual word. The common words correspond to curved sections of the wheels, the handlebars, and the engine block area.*

Table 2 Comparison of Different Methods Trained on Caltech and Raw Google Data. All Methods Were Tested on Caltech Data. The Task Is Classification, With the Figures Being the Error Rate at Point of Equal Error on a Receiver Operating Characteristic Curve. The Error Margins Are Roughly +/−2%, Using Different Random Initializations

	pLSA	pLSA	ABS-pLSA	ABS-pLSA
Category/Training method	Caltech	Google	Caltech	Google
(A)irplane	17.7	24.7	13.2	17.2
(C)ars Rear	2.0	21.0	0.2	13.2
(F)ace	22.1	20.3	11.5	36.4
(G)uitar	9.3	17.6	10.0	62.0
(L)eopard	12.0	15.0	12.0	16.0
(M)otorbike	19.0	15.2	6.0	18.5
(W)rist watch	21.6	21.0	7.7	20.5

Table 3 pLSA-Based Models Applied to Two Classes From the PASCAL VOC 2005 Challenge. Values Are Classification Error (in Percent). Column 2: Best Classification Performance Obtained by All Methods in Competition. Columns 3, 4, and 5: Performance of Google-Trained Models, Evaluated on PASCAL Test Data. Columns: 6, 7, and 8: Performance of Models Trained on PASCAL Training Data, Evaluated on PASCAL Test Data

	PASCAL	pLSA	ABS-pLSA	pLSA	ABS-pLSA
Category	Best	Google	Google	PASCAL	PASCAL
Motorbikes	20.2	37.6	36.7	35.1	30.7
Cars	28.0	29.8	25.1	32.0	29.1

observed when training from Google images, compared to training on Caltech images. For around half the categories, the use of location information reduces the error significantly, although only in the case of motorbikes and airplanes is ABS-pLSA better than either of the other two approaches.

ABS-pLSA performs notably poorly on the guitar data set. This may be explained by the fact that all the prepared data have the guitar in a vertical position while guitars appear at a seemingly random orientation in the Google training data. Since neither of the models using location can handle rotation, they perform badly, in contrast to pLSA which still performs respectably.

We also applied our pLSA-based methods to two classes from the PASCAL 2005 VOC Challenge [11]: cars and motorbikes. Table 3 shows the performance of the three approaches in a classification task on the test set of each class. We used two different training approaches: 1) training an eight-topic model on Google images and selecting the single best topic using the PASCAL validation set; 2) training a five-topic model on the PASCAL training set, picking the single best topic using the PASCAL validation set. Table 3 shows that the PASCAL-trained models outperform the Google-trained ones, in the case of cars being competitive with the best competition entry.

VII. SUMMARY

Visual Internet search is a vital tool needed to enable users to access the vast amount of image and video data on the Internet. A number of different approaches to this problem have been proposed, for example, the approach of Jing and Baluja [22] who adapt the PageRank algorithm to the visual domain. By contrast, our approach directly addresses the core object recognition problem, constructing object class models that can be used for recognition in a domain beyond that raw search engine data. A key aspect of our approach is the ability to construct such models with a minimum of supervision, enabling their use without needed human labeled images (which are typically very scarce in real tasks).

The Google data sets present an extremely challenging environment within which to learn. It is therefore pleasing to see that our methods are able to find some consistency within the data, with the addition of location into the model giving a convincing performance gain. Since users typically examine only the first few web pages, the visual model only needs to select a few images to generate big improvement in search quality. There are many possible variants on our testing paradigm, such as the inclusion of user feedback, for example, which would further improve the performance.

The application of models trained on Google data in more general settings showed a difference in performance between the two methods. The learned models gave a respectable performance on the PASCAL and Caltech data sets, with the exception of a few classes. Determining the number of topics to use remains an open issue. If a larger validation set were available, then it might be possible to combine several of them rather than picking a single one. This would be particularly advantageous when different aspects (views) of the object are split into separate topics.

Also, given the easy availability of negative examples, discriminative training techniques for learning the topics would seem to be a potential way to improve performance.

While we have focused on the Internet image search application, our techniques can easily be applied to related applications such as search images on company intranets or grouping portions of video footage. Another intriguing application would be to use the visual models to refine the text query models, for example, cluster of watch images found by the visual model. ■

Acknowledgment

The authors would like to thank David Forsyth.

REFERENCES

[1] K. Barnard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, and M. Jordan, "Matching words and pictures," *J. Mach. Learn. Res.*, vol. 3, pp. 1107–1135, Feb. 2003.

[2] A. Berg, T. Berg, and J. Malik, "Shape matching and object recognition using low distortion correspondence," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, San Diego, CA, Jun. 2005, vol. 1, pp. 26–33.

[3] T. Berg and D. Forsyth, "Animals on the web," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2006, pp. 1463–1470.

[4] D. Blei, A. Ng, and M. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Jan. 2003.

[5] A. Bosch, A. Zisserman, and X. Munoz, "Scene classification via pLSA," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 517–530.

[6] P. Carbonetto, N. de Freitas, and K. Barnard, "A statistical model for general contextual object recognition," in *Computer Vision—ECCV 2004*, vol. 3021. Berlin, Germany: Springer-Verlag, ser. Lecture Notes in Computer Science, pp. 350–362.

[7] B. Collins, J. Deng, K. Li, and L. Fei-Fei, "Towards scalable dataset construction: An active learning approach," in *Computer Vision—ECCV 2008*, vol. 5302. Berlin, Germany: Springer-Verlag, 2008, ser. Lecture Notes in Computer Science, pp. 86–98.

[8] G. Csurka, C. Bray, C. Dance, and L. Fan, "Visual categorization with bags of keypoints," in *Proc. Workshop Stat. Learn. Comput. Vis.*, 2004, pp. 1–22.

[9] O. Etzioni, K. Reiter, S. Soderland, and M. Sammer, "Lexical translation with application to image search on the web," in *Proc. Mach. Transl. Summit XI*, 2007.

[10] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2008 (VOC2008) results." [Online]. Available: http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html

[11] M. Everingham, L. Van Gool, C. Williams, and A. Zisserman, "PASCAL visual object challenge datasets," 2005. [Online]. Available: http://www.pascal-network.org/challenges/VOC/voc/index.html

[12] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 594–611, Apr. 2006.

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

[14] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman, "Learning object categories from Google's image search," in *Proc. 10th Int. Conf. Comput. Vis.*, Beijing, China, Oct. 2005, vol. 2, pp. 1816–1823.

[15] R. Fergus and P. Perona, "Caltech object category datasets," 2003. [Online]. Available: http://www.vision.caltech.edu/html-files/archive.html

[16] R. Fergus, P. Perona, and P. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2003, vol. 2, pp. 264–271.

[17] M. Fritz and B. Schiele, "Decomposition, discovery and detection of visual categories using topic models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, DOI: 10.1109/CVPR.2008.4587803.

[18] "Google Translation Tool," 2005. [Online]. Available: http://translate.google.com/translate_t

[19] G. Griffin, A. Holub, and P. Perona, Caltech-256 object category dataset, California Inst. Technol., Pasadena, Tech. Rep. 7694, 2007.

[20] T. Hofmann, "Probabilistic latent semantic indexing," in *Proc. 22nd Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, Berkeley, CA, 1999, pp. 50–57.

[21] Yahoo Inc., *Flickr*, 2006. [Online]. Available: http://www.flickr.com

[22] Y. Jing and S. Baluja, "Visualrank: Applying PageRank to large-scale image search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1877–1890, Nov. 2008.

[23] T. Kadir and M. Brady, "Scale, saliency and image description," *Int. J. Comput. Vis.*, vol. 45, no. 2, pp. 83–105, 2001.

[24] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, New York, 2006, pp. 2169–2178.

[25] Y. LeCun, F. Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2004, vol. 2, pp. 97–104.

[26] B. Leibe, A. Leonardis, and B. Schiele, "Combined object categorization and segmentation with an implicit shape model," in *Proc. Workshop Stat. Learn. Comput. Vis.*, 2004, pp. 17–32.

[27] J. Li, G. Wang, and L. Fei-Fei, "Optimol: Automatic object picture collection via incremental model learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, DOI: 10.1109/CVPR.2007.383048.

[28] D. Lowe, "Object recognition from local scale-invariant features," in *Proc. 7th Int. Conf. Comput. Vis.*, Kerkyra, Greece, Sep. 1999, pp. 1150–1157.

[29] K. Mikolajczyk and C. Schmid, "Indexing based on scale invariant interest points," in *Proc. 8th Int. Conf. Comput. Vis.*, Vancouver, BC, Canada, 2001, pp. 525–531.

[30] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer, "Weak hypotheses and boosting for generic object detection and recognition," in *Proc. 8th Eur. Conf. Comput. Vis.*, Prague, Czech Republic, 2004, vol. 2, pp. 71–84.

[31] P. Quelhas, F. Monay, J. M. Odobez, D. Gatica-Perez, T. Tuytelaars, and L. Van-Gool, "Modeling scenes with local descriptors and latent aspects," in *Proc. 10th Int. Conf. Comput. Vis.*, Beijing, China, 2005, pp. 883–890.

[32] H. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 1, pp. 23–38, Jan. 1998.

[33] B. Russell and A. Torralba, "Labelme: The open annotation tool," 2006. [Online]. Available: http://people.csail.mit.edu/brussell/research/LabelMe/intro.html

[34] F. Schroff, A. Criminisi, and A. Zisserman, "Harvesting image databases from the web," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Rio de Janeiro, Brazil, 2007, DOI: 10.1109/ICCV.2007.4409099.

[35] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman, "Discovering object categories in image collections," in *Proc. Int. Conf. Comput. Vis.*, 2005.

[36] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. Int. Conf. Comput. Vis.*, Oct. 2003, pp. 1470–1477.

[37] A. Sorokin and D. Forsyth, "Utility data annotation with Amazon Mechanical Turk," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, DOI: 10.1109/CVPRW.2008.4562953.

[38] E. Sudderth, A. Torralba, W. Freeman, and A. Willsky, "Describing visual scenes using transformed Dirichlet processes," in *Advances in Neural Information Processing Systems 18*. Cambridge, MA: MIT Press, 2005, pp. 1299–1306.

[39] E. Sudderth, A. Torralba, W. Freeman, and A. Willsky, "Learning hierarchical models of scenes, objects, and parts," in *Proc. 10th Int. Conf. Comput. Vis.*, Beijing, China, 2005, vol. 2, pp. 1331–1338.

[40] A. Torralba, K. P. Murphy, and W. T. Freeman, "Sharing features: Efficient boosting procedures for multiclass object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Washington, DC, 2004, pp. 762–769.

[41] L. van Ahn, *The ESP Game*, 2006. [Online]. Available: http://www.espgame.org/gwap/

[42] S. Vijayanarasimhan and K. Grauman, "Keywords to visual categories: Multiple-instance learning for weakly supervised object categorization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, DOI: 10.1109/CVPR.2008.4587632.

[43] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. Comput. Vis. Pattern Recognit.*, 2001, vol. 1, pp. 511–518.

[44] B. Yao, X. Yang, and S. C. Zhu, "Introduction to a large-scale general purpose ground truth dataset: Methodology, annotation tool, and benchmarks," in *Proc. Energy Minimization Methods Comput. Vis. Pattern Recognit.*, 2007, pp. 169–183.

[45] M. Everingham, A. Zisserman, C. K. I. Williams, L. Van Gool, M. Allan, C. M. Bishop, O. Chapelle, N. Dalal, T. Deselaers, G. Dorkó, S. Duffner, J. Eichhorn, J. D. R. Farquhar, M. Fritz, C. Garcia, T. Griffiths, F. Jurie, D. Keysers, M. Koskela, J. Laaksonen, D. Larlus, B. Leibe, H. Meng, H. Ney, B. Schiele, C. Schmid, E. Seemann, J. Shawe-taylor, A. Storkey, O. Szedmak, B. Triggs, I. Ulusoy, V. Viitaniemi, and J. Zhang, "The 2005 PASCAL visual object classes challenge," 2006.

ABOUT THE AUTHORS

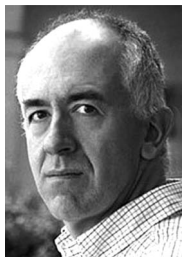
Rob Fergus received the B.S. degree in electrical engineering from the University of Cambridge, Cambridge, U.K., the M.S. degree in electrical engineering with Prof. Pietro Perona from the California Institute of Technology (Caltech), Pasadena, in 2002 and the Ph.D. degree with Prof. Andrew Zisserman from the University of Oxford, Oxford, U.K., in 2005.

He is currently an Assistant Professor of Computer Science at the Courant Institute of Mathematical Sciences, New York University (NYU), New York. Before coming to NYU, he spent two years as a Postdoctoral Researcher at the Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology (MIT), Cambridge, working with Prof. William Freeman. In 2005, his Ph.D. dissertation won the prize for the best Computer Science dissertation in the U.K.

Li Fei-Fei received the B.Sc. degree in physics from Princeton University, Princeton, NJ, in 1999 and the Ph.D. degree in electrical engineering from the California Institute of Technology, Pasadena, in 2005.

From 2005 to August 2009, she was an Assistant Professor at the Electrical and Computer Engineering Department, University of Illinois Urbana-Champaign, Urbana, and Computer Science Department, Princeton University, respectively. She is currently an Assistant Professor at the Computer Science Department, Stanford University, Stanford, CA. Her main research interest is in vision, particularly high-level visual recognition. In computer vision, her interests span from object and natural scene categorization to human activity categorizations in both videos and still images. In human vision, she has studied the interaction of attention and natural scene and object recognition, and decoding the human brain fMRI activities involved in natural scene categorization by using pattern recognition algorithms.

Dr. Fei-Fei is a recipient of a Microsoft Research New Faculty award, a Google research award and a National Science Foundation (NSF) CAREER award.



Pietro Perona (Member, IEEE) received a degree in electrical engineering from the Universita di Padova, Padova, Italy, in 1985 and the Ph.D. degree from the University of California at Berkeley in 1990.

He was a Postdoctoral Fellow at the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology (MIT), Cambridge, in 1990-1991 and became an Assistant Professor of Electrical Engineering at the California Institute of Technology (Caltech), Pasadena, in 1991. In 1996, he became Professor of Electrical Engineering and of Computation and Neural Systems. Since 1999, he has been the Director of the National Science Foundation Engineering Research Center in Neuromorphic Systems Engineering at Caltech. He is interested in the computational aspects of vision; his current research focus is visual recognition. He has worked on PDEs for image analysis and segmentation (anisotropic diffusion); multiresolution–multiorientation filtering for early vision; human texture perception and segmentation; dynamic vision, grouping, detection, and analysis of human motion; human perception of 3-D shape from shading, learning, and recognition of object categories; human categorization of scenes; interaction of attention; and recognition.

Dr. Perona has served on the editorial board of the *International Journal of Machine Vision*, the *Journal of Machine Learning Research*, *Vision Research*, and as Co-General Chair of the 2003 IEEE Conference of Computer Vision and Pattern Recognition (CVPR).

Andrew Zisserman, photograph and biography not available at the time of publication.