OPTIMOL: automatic Online Picture collecTion via Incremental MOdel Learning

Li-Jia Li[1], Gang Wang[1] and Li Fei-Fei[2]

[1] Dept. of Electrical and Computer Engineering, University of Illinois Urbana-Champaign, USA

[2] Dept. of Computer Science, Princeton University, USA

jiali3@uiuc.edu, gwang6@uiuc.edu, feifeili@cs.princeton.edu

Abstract

A well-built dataset is a necessary starting point for advanced computer vision research. It plays a crucial role in evaluation and provides a continuous challenge to state-of-the-art algorithms. Dataset collection is, however, a tedious and time-consuming task. This paper presents a novel automatic dataset collecting and model learning approach that uses object recognition techniques in an incremental method. The goal of this work is to use the tremendous resources of the web to learn robust object category models in order to detect and search for objects in real-world cluttered scenes. It mimics the human learning process of iteratively accumulating model knowledge and image examples. We adapt a non-parametric graphical model and propose an incremental learning framework. Our algorithm is capable of automatically collecting much larger object category datasets for 22 randomly selected classes from the Caltech 101 dataset. Furthermore, we offer not only more images in each object category dataset, but also a robust object model and meaningful image annotation. Our experiments show that OPTIMOL is capable of collecting image datasets that are superior to Caltech 101 and LabelMe.

1. Introduction

Type the word "airplane" in your favorite Internet search image engine, say Google Image (or Yahoo!, flickr.com, etc.). What do you get? Of the thousands of images these search engines return, only a small fraction would be considered good airplane images ($\sim 15\%$). It is fair to say that for most of today's average users surfing the web for images of generic objects, the current commercial state-of-the-art results are far from satisfying.

This problem is intimately related to the problem of learning and modeling generic object classes, a topic that has recently captured the attention of search engine developers as well as vision researchers. However, in order to develop effective object categorization algorithms, researchers rely on a critical resource - an accurate object class dataset. A good dataset serves as training data as well as an evaluation benchmark. A handful of large scale datasets exist currently to serve such a purpose, such as Caltech 101 [4], the UIUC car dataset [1], etc. Sec.1.1 will elaborate on the strengths and weaknesses of these datasets. In short, all of them, however, have rather a limited number of images and offer no possibility of expansion other than with extremely costly manual labor.

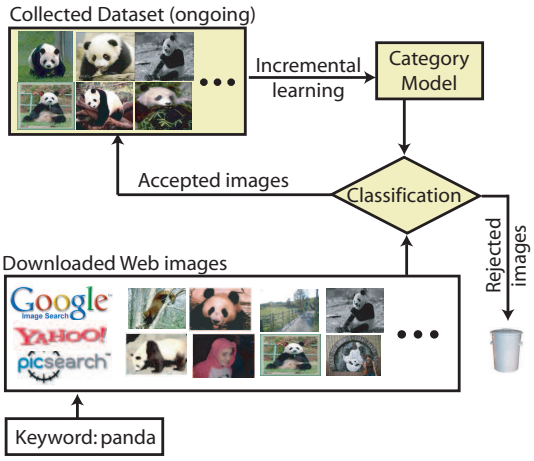

Figure 1. Illustration of the framework of the Online Picture collecTion via Incremental MOdel Learning (OPTIMOL) system. This framework works in an incremental way: Once a model is learned, it can be used to do classification on the images from the web resource. The group of images classified as being in this object category are incorporated into the collected dataset. Otherwise, they are discarded. The model is then updated by the newly accepted images in the current iteration. In this incremental fashion, the category model gets more and more robust. As a consequence, the collected dataset becomes larger and larger.

So far the story is a frustrating one: Users of the web search engines would like better search results when looking for, say, objects; developers of these search engines would like more robust visual models to improve these re-

1-4244-1180-7/07/$25.00 ©2007 IEEE

sults; vision researchers are developing the models for this purpose; but in order to do so, it is critical to have large and diverse object datasets for training and evaluation; this, however, goes back to the same problem that the users face.

In this paper, we provide a framework to simultaneously learn object class models and collect object class datasets. This is achieved by leveraging the vast resource of images available on the Internet. The sketch of our idea is the following. Given a very small number of seed images of an object class (either provided by a human or automatically), our algorithm learns a model that best describes this class. Serving as a classifier, the algorithm can pull from the web those images that belong to the object class. The newly collected images are added to the object dataset, serving as new training data to update and improve the object model. With this new model, the algorithm can then go back to the web and pull more relevant images. This is an iterative process that continuously gathers a highly accurate image dataset while learning a more and more robust object model. We will show in our experiments that our automatic, online algorithm is capable of collecting object class datasets that are far bigger than Caltech 101 or LabelMe [16]. To summarize, we highlight here the main contributions of our work.

- We propose an iterative framework that simultaneously collects object category datasets and learns the object category models. The framework use Bayesian incremental learning as its theoretical base. To the best of our knowledge, ours is among the first papers (if not the first) that deals with these two problems together.
- We have developed an incremental learning scheme that uses only the newly added data points (i.e. images) for training a new model. This memory-less learning scheme is capable of handling an arbitrarily large number of images, a vital property for large image datasets.
- Our experiments show that our algorithm is capable of both learning highly effective object category models and collecting object category datasets far larger than that of Caltech 101 or LabelMe.

1.1. Related works

Image Retrieval from the Web: Content-based image retrieval (CBIR) has been an active field of research for a number of years. However, we do not regard our work as fitting in the conventional framework of CBIR. Instead of learning to annotate images with a list of words and phrases, we instead emphasize collecting the most suitable images possible from the web resources given a single word or phrase. One major difference between our work and traditional CBIR is the emphasis on visual model learning. While collecting images of a particular object category, our algorithm continues to learns a better and better visual model to classify this object.

A few recent systems in this domain are closer to our current framework. H. Feng et al. proposed a method to refine search engine returns by co-training [7]. Their method, however, does not offer an incremental training framework to iteratively improve the collection. Berg and Forsyth developed a system to collect animal pictures from the web [2]. Their system takes advantage of both the text surrounding the web images and the global feature statistics (patches, colors, textures) of the images to collect a large number of animal images. Another method close in spirit to ours is by Yanai and Barnard [21]. Though their method focuses on image annotation, they also utilize the idea of refining web image returns by a probabilistic model. Finally, two papers by Fergus et al. [8, 10] use idea of training a good object class model from web images returned by search engines, hence obtaining an object filter to refine these results. All the techniques above achieve better search results by using either a better visual model or a combination of visual and text models to essentially re-rank the rather noisy images from the web. We show later that by introducing an iterative framework of incremental learning, we are able to embed the processes of image collection and model learning into a mutually reinforcing system.

Object Classification: Given the recent explosion of object categorization research, it is out of the scope of this paper to offer a thorough review of the literature. We would like to emphasize that our proposed framework is not limited to the particular object model used in this paper as an example: any model that can be cast into an incremental learning framework is suitable for our protocol. Of the many possibilities, we have chosen to use a variant of the HDP (Hierarchical Dirichlet Process) [19] model based on "the bag of words" representation of images. A number of systems based on the bag of words model representation have shown to be effective for object and scene classification [8, 17, 6]. The models mentioned above are all developed for a batch learning scenario. A handful of object recognition works have also dealt with the issue of incremental learning explicitly. The most notable ones are [13] and [4]. Our approach, however, is based on a model significantly different from these papers.

Object Datasets: One main goal of our proposed work is to suggest a framework that can replace much of the current human effort in image dataset collection for object categorization. A few popular object datasets exist today as the main training and evaluation resources for the community. *Caltech 101* contains 101 object classes each containing between 40 to 400 images [4]. It was collected by a group of students spending on average three or four hours per 100 images. While it is regarded as one of the most comprehensive object category datasets now available, it is limited in terms of the variation in the images (big, centered objects with few viewpoint changes), numbers of images per

category (at most a few hundred) as well as the number of categories. Recently, LabelMe has offered an alternative way of collecting datasets of objects by having people upload their images and label them [16]. This dataset is much more varied than Caltech 101, potentially serving as a better benchmark for object detection algorithms. But since it relies on people uploading pictures and making uncontrolled annotations, it is difficult to use it as a generic object dataset. In addition, while some classes have many images (such as 8897 for "car"), others have very few (such as 6 for "airplane"). A few other object category datasets such as [1] are also used by researchers. All of the datasets mentioned above require laborious human effort to gather and select the images. In addition, while serving as training and test datasets for researchers, they are not suitable or usable for general search engine users. Our proposed work offers a unified way of automatically gathering data useful both as a research dataset as well as for answering user queries.

2. General Framework of OPTIMOL

Algorithm 1 Incremental learning, classification and data collection

Download from the Web a large reservoir of images obtained by searching for keyword(s)
Initialize the object category dataset with seed images (manually or automatically)
repeat
  Learn object category model with the latest input images to the dataset
  Classify downloaded images using the current object category model
  Augment the dataset with accepted images
until user satisfied or images exhausted

OPTIMOL has two goals to fulfill simultaneously: to automatically collect datasets of object classes from the web and to incrementally learn object category models. We use Fig.1 and Alg.1 to illustrate the overall framework. For every object category we are interested in, say, "panda", we *initialize* our image dataset with a handful of seed images. This can be done either manually or automatically [1]. With this small dataset, we begin the iterative process of model learning and dataset collection. *Learning* is done via an incremental learning procedure we introduce in Sec.3.3. Given the most updated model of the object class, we perform a binary *classification* on a subset of images downloaded from the web (e.g. panda vs. background) [2]. If an image is accepted as a "panda" image based on some statistical criteria (see Sec.3.3), we *augment* our existing panda dataset by appending this new image. We then update our panda model with the subset of the newly accepted images (see Sec.3.4 for details of the "cache set"). Note that the already existing images in the dataset no longer participate in this round of learning. Meanwhile, the background model will also be updated using a constant resource of background images. We *repeat* this process till a sufficient dataset is collected or we have exhausted all downloaded images.

[1]To automatically collect a handful of seed images, we can use the images returned by the first page of Google image search, or any other state-of-the-art commercial search engines.

[2]The background class model is learnt by using a published 'background' image dataset [9, 5]

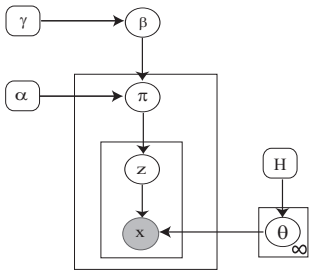

Figure 2. Graphical model of HDP. Each node denotes a random variable.

3. Detailed System of OPTIMOL

3.1. Object category model

We choose the "bag of words" representation for our object category model, but our system is not committed to any particular choice of object category representation or model structure. As long as one could cast the model into an incremental learning framework, it would be theoretically suitable for OPTIMOL. We make this choice, however, based on the recent successes of the "bag of words" models in object and scene recognition [17, 6], particularly the usage of latent topic models for such representation [11, 3, 19]. Similarly to [18, 20], we adapt the Hierarchical Dirichlet process(HDP) [19] for our object category model. Compared to parametric latent topic models such as LDA [3] or plSA [11], HDP offers a way to sample an infinite number of latent topics, or clusters, for each object category model. This property is especially desirable for OPTIMOL because as we grow our dataset, we would like to retain the ability to 'grow' the object class model when new clusters of images arise. We introduce our HDP object category model in more detail in Sec.3.2.

3.2. Hierarchical Dirichlet process

We represent an image as a bag of visual words. Each category consists of a variable number of latent topics corresponding to clusters of images that have similar visual words attributes. We model both object and background classes with HDP [19]. Given γ, α as the concentration parameters and H as a base probability measure, HDP defines a global random probability measure $G_0 \sim DP(\gamma, H)$. Based on G_0, a random measure $G_j \sim DP(\alpha, G_0)$ is independently sampled for each group to explain the internal structure. Here, DP represents the Dirichlet process.

Fig.2 shows the graphical model of HDP. θ corresponds to the distributions of visual words given different latent topics shared among different images. H indicates the prior distribution of θ. Let x_{ji} be the ith patch in jth image. For each patch x_{ji}, there is a hidden variable z_{ji} denoting the latent topic index. If β is the stick-breaking weights and π_j is the mixing proportion of z for the jth image, the hierarchical Dirichlet process can be expressed as:

$$\begin{aligned} \beta|\gamma \sim \text{GEM}(\gamma) \qquad \pi_j|\alpha,\beta \sim \text{DP}(\alpha,\beta) \\ \theta_k|H \sim H \qquad z_{ji}|\pi_j \sim \pi_j \qquad x_{ji}|z_{ji},\theta_k \sim F(\theta_{z_{ji}}) \end{aligned} \quad (1)$$

where GEM represents the stick-breaking process.

For both Fig.2 and Eq.1, we omit the mention of object category to avoid confusion. The distribution of π is class specific, and so is x. In other words, two class-specific θs govern the distribution of x. Given an object class c, a topic z can be generated using a multinomial distribution parameterized by π. Given this topic, a patch is generated using the multinomial distribution of $F(\theta_z^c)$.

3.3. Incremental learning of a latent topic model

Given the object class model, we propose an incremental learning scheme such that OPTIMOL could update the model at every iteration of the dataset collection process. Our goal here is to perform incremental learning by using only new images selected at this given iteration. We will illustrate in Fig.7(b) that this is much more efficient than performing a batch learning with all images in the existing dataset at every iteration. Let Θ denote the model parameters, and I_j denote the jth image represented by a set of patches $x_{j1}, \cdots, x_{jn}$. For each patch x_{ji}, there is a hidden variable z_{ji} denoting the latent topic index. The model parameters and hidden variable are updated iteratively using the current model and the input image I_j in the following fashion:

$$z_j \sim p(z|\Theta^{j-1}, I_j) \qquad \Theta^j \sim p(\Theta|z_j, \Theta^{j-1}, I_j) \quad (2)$$

where Θ^{j-1} represents the model parameters learned from the previous $j-1$ images. Neal & Hinton [15] provides a theoretical ground for incrementally learning mixture models via sufficient statistics updates. We follow this idea by keeping only the sufficient statistics of the parameters associated with the existing images in an object dataset. Learning is then achieved by updating these sufficient statistics with the those provided by the new images. One straightforward method is to use all the new images accepted by the current classification criterion. It turns out that this method will favor too much those images with a similar appearance to the existing ones, hence resulting in more and more specialized object models. In order to take full advantage of the non-parametric HDP model, as well as to avoid this "over specialization", we only use a subset of images to update our model. This subset is called the "cache set". We detail the selection of the "cache set" in Sec.3.4.

3.3.1 Markov Chain Monte Carlo sampling

The goal of learning is to update the parameters in the hierarchical model. In this section, we describe how we learn the parameters by Gibbs sampling [13] of the latent variables. We choose the popular *Chinese restaurant franchise* [19] metaphor to describe this procedure. Imagine multiple Chinese restaurants sharing a set of dishes. At each table of each restaurant, a dish is shared by the customers sitting at that table. Metaphorically, we describe the jth image as the jth restaurant and the image level mixture component for x_{ji} as a table t_{ji}, where x_{ji} is the ith customer in the jth restaurant. Similarly, the global latent topic for the tth table in the jth restaurant is represented as the dish k_{jt} :

$$t_{ji}|t_{j1},\ldots,t_{ji-1},\alpha,G_0 \sim \sum_{t=1}^{T_j} n_{jt}\delta_{t_{ji}=t} + \alpha G_0 \quad (3)$$

$$k_{jt}|k_{11},k_{12},\ldots,k_{21},\ldots,k_{jt-1},\gamma \sim \sum_{k=1}^{K} m_k \delta_{k_{jt}=k} + \gamma H \quad (4)$$

where n_{jt} denotes the number of customers sitting at the tth table in the jth restaurant, T_j is the current number of tables in the jth restaurant, m_k represents the number of tables ordered dish k, and K denotes the current number of dishes. A new table and new dish can also be generated from G_0 and H, respectively, when needed.

Sampling the table. According to Eq.3 and Eq.4, the probability of a new customer x_{ji} assigned to table t is:

$$P(t_{ji}=t|x_{ji},t_{-ji},\mathbf{k}) \propto \begin{cases} a_0 p_{t_{\text{new}}} & \text{for } t = t_{\text{new}} \\ n_{jt} f(x_{ji}|\theta_{k_{ji}}) & \text{for used } t \end{cases} \quad (5)$$

where $p_{t_{\text{new}}}$ is the likelihood for $t_{ji} = t_{new}$:

$$\sum_{k=1}^{K} \frac{m_k}{\sum_{k=1}^{K} m_k + \gamma} f(x_{ji}|\theta_{k_{ji}}) + \frac{\gamma}{\sum_{k=1}^{K} m_k + \gamma} f(x_{ji}|\theta_{k_{new}})$$

$f(x_{ji}|\theta_{k_{ji}})$ is the conditional density of patch x_{ji} given all data items associated with global latent topic k except itself. The probability of assigning a newly generated table t_{new} to a global latent topic is proportional to Eq.6.

$$\begin{cases} m_k f(x_{ji}|\theta_{k_{ji}}) & \text{for used } k \\ \gamma f(x_{ji}|\theta_{k_{new}}) & \text{for new k} \end{cases} \quad (6)$$

Sampling the global latent topic. For the existing tables, the dish can change according to all customers of table. A sample of the global latent topic for the image level mixture component t in image j, k_{jt}, can be obtained from:

$$\begin{cases} m_k f(\mathbf{x}_{jt}|\theta_{k_{ji}}) & \text{for used } k \\ \gamma f(\mathbf{x}_{jt}|\theta_{k_{new}}) & \text{for new k} \end{cases} \quad (7)$$

Similarly, $f(\mathbf{x}_{jt}|\theta_{k_{jt}})$ is the conditional density of a set of patches $\mathbf{x}_{jt}$ given all patches associated with topic k except themselves. A new global latent topic will be sampled from H if k=k_{new} according to Eq.7. n_{jt} and m_k will be updated respectively regarding the table index and global latent topic assigned. Given $z_{ji} = k_{jt_{ji}}$, we in turn update $F(\theta_{z_{ij}}^c)$.

3.4. New Image Classification and Annotation

For every iteration of the dataset collection process, we have a binary classification problem: classify images with foreground object versus background images. Given the current model, we have $p(z|c)$ parameterized by the distribution of global latent topics for each class in the Chinese restaurant franchise and $p(x|z,c)$ parameterized by $F(\theta_z^c)$ learned for each category c by Gibbs sampling. A testing image I is represented as a collection of local patches x_i, where $i = \{1,\ldots,M\}$ and M is the number of patches. The likelihood $p(I|c)$ for each class is calculated by:

$$P(I|c) = \prod_i \sum_j P(x_i|z_j,c)P(z_j|c) \tag{8}$$

Classification is made by choosing the category model that yields the higher probability. From a dataset collection point of view, incorporating an incorrect image into the dataset (false positive) is much worse than missing a correct image (false negative). Hence, a risk function is introduced to penalize false positives more heavily:

$$\begin{aligned} R_i(A|I) &= \lambda_{Ac_f}P(c_f|I) + \lambda_{Ac_b}P(c_b|I) \\ R_i(R|I) &= \lambda_{Rc_f}P(c_f|I) + \lambda_{Rc_b}P(c_b|I) \end{aligned} \tag{9}$$

Here A represents acceptance of an image into our dataset, R rejection. As long as the risk of accepting this image is lower than rejecting it, it gets accepted. Updating the training set is finally decided by the likelihood ratio:

$$\frac{P(I|c_f)}{P(I|c_b)} > \frac{\lambda_{Ac_b} - \lambda_{Bc_b}}{\lambda_{Rc_f} - \lambda_{Ac_f}} \frac{P(c_b)}{P(c_f)} \tag{10}$$

where the c_f is the foreground category while the c_b is the background category. $\frac{\lambda_{Ac_b}-\lambda_{Bc_b}}{\lambda_{Rc_f}-\lambda_{Ac_f}}$ is automatically adjusted by applying the likelihood ratio measurement to a reference dataset[3] at every iteration. New images satisfying Eq.10 are incorporated into the collected dataset.

The goal of OPTIMOL is not only to collect a good image dataset, but also to provide further information about the location and size of the objects contained in the dataset images. Object annotation is carried out by first calculating the likelihood of each patch given the object class c_f:

$$p(x|c_f) = \sum_i p(x|z_i,c_f)p(z_i|c_f) \tag{11}$$

The region with the most concentrated high likelihood patches is then selected as the object region. Sample results are shown in Fig.5.

As we mentioned in Sec.3.3, we use a "cache set" of images to incrementally update our model. The "cache set" is a less "permanent" set of good images compared to the actual image dataset. At every round, if all "good" images are used for model learning, it is highly likely that many of these images will look very similar to the previously collected images, hence reinforcing the model to be even more specialized in picking out such images for the next round. So the usage of the "cache set" is to retain a group of images that tend to be more diverse from the existing images in the dataset. For each new image passing the classification qualification (Eq.10), it is further evaluated by Eq.12 to determine whether it should belong to the "cache set" or the permanent set.

$$H(I) = -\sum_z p(z|I)\ln p(z|I) \tag{12}$$

According to Shannon's definition of entropy, Eq.12 relates to the amount of uncertainty about an event associated with a given probability distribution. Images with high entropy are more uncertain, which indicates possible new topics, or its lack of strong membership to one single topic. Thus, these high likelihood and high entropy images are good for model learning. Meanwhile, images with low entropy are regarded as confident foreground images, which will be incorporated into the permanent dataset.

4. Experiments & Results

We conduct 3 experiments to illustrate the effectiveness of OPTIMOL. Exp.1 demonstrates the superior dataset collection results of OPTIMOL over the existing datasets. Exp.2 shows that OPTIMOL is on par with the state of the art object models [8] for multiple object classification. Exp.3 shows a performance comparison of the batch vs. incremental learning methods. We first introduce the various datasets, then show in Sec.4.2 a walkthrough for how OPTIMOL works for the accordion category.

4.1. Datasets Definitions

We define the following 3 different datasets that we will use in our experiments:

1. Caltech 101-Web & Caltech 101-Human
   2 versions of the Caltech 101 dataset are used in our experiment. Caltech 101-Web is the original raw dataset downloaded from the web with a large portion of contaminated images in each category. The number of images in each category varies from 113 (winsor-chair) to 1701 (watch). Caltech 101-Human is the clean dataset manually selected from Caltech 101-Web. By using this dataset, we show that OPTIMOL can achieve comparable or even better retrieval performance compared to human labeled results.
2. Web-23
   We downloaded 21 object categories from online image search engines with corresponding query words randomly selected from object categories in Caltech 101-Web. In addition, "face" and "penguin" categories are also included in Web-23 for further comparison. The number of images in each category varies from 577 (stop-sign) to 12414 (face).

[3] To achieve a fully automated system, we use the original seed images as the reference dataset. As the training dataset grows larger, the direct effect of the original training images vanishes in terms of the object model. It therefore becomes a good approximation of a validation dataset.

Most of the images in a category are incorrect images (e.g. 352 correct accordions out of 1659 images).

3. Fergus ICCV'05 dataset
   A 7-Category dataset provided by [8]. Object classes are: airplane, car, face, guitar, leopard, motorbike and watch.

4.2. Walkthrough for the accordion category

As an example, we describe how OPTIMOL collects images for the accordion category following Alg.1 and Fig.1. We first download 1659 images by typing the query word "accordion" in image search engines such as Google image, Yahoo image and Picsearch. We use the first 15 images from the web resource as our seed images, assuming that most of them are good quality accordions. We represent each image as a set of local regions. We use the Kadir and Brady [12] salient point detector to find the informative local regions. A 128-dim rotationally invariant SIFT vector is used to represent each region [14]. We build a 500-word codebook by applying K-means clustering to the 89058 SIFT vectors extracted from the 15 seeds images of each of the 23 categories. In Fig.3, we show some detected interest regions as well as some codeword samples. Fig.4 illustrates the first and second iterations of OPTIMOL for accordion category model learning, classification and image collection.

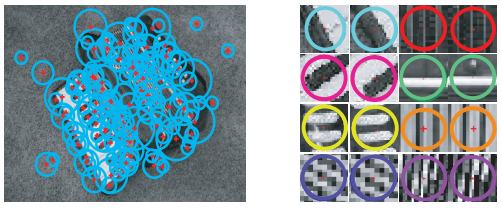

Figure 3. Left: Interest regions found by Kadir&Brady detector. The circles indicate the interest regions. The red crosses are the centers of these regions. Right: Sample codewords. Patches with similar SIFT descriptor are clustered into the same codeword, which are presented in same color.

4.3. Exp.1: Image Collection

21 object categories are selected randomly from Caltech 101-Web for this experiment. The experiment is split into two parts: 1. Retrieval from Caltech 101-Web. The number of collected images in each category is compared with the same numbers in Caltech 101-Human. 2. Retrieval from Web-23 using the same 21 categories as in part 1. Results of these two parts are displayed in Fig.5. We first observe that OPTIMOL is capable of automatically collecting very similar number of images from Caltech 101-Web as the humans have done by hand in Caltech 101-Human. Furthermore, by using images from Web-23, OPTIMOL achieves on average 6 times as many images as Caltech 101-Human (some even $10\times$ higher). In Fig.5, we also compare our results with LabelMe [16] for each of the 22 categories. In addition, a "penguin" category is included so that we can compare our results with the state-of-art dataset collecting approach

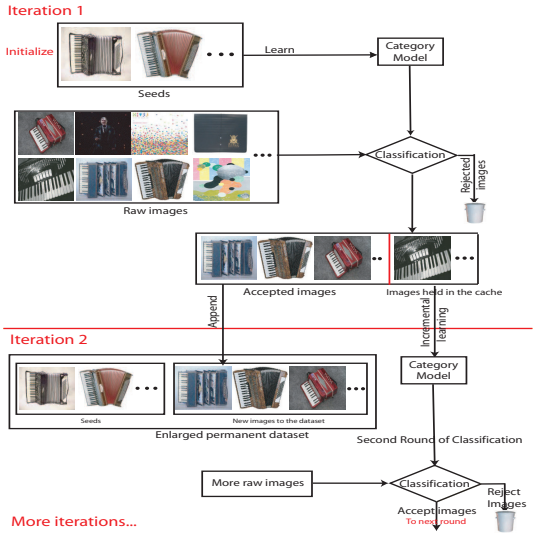

Figure 4. Example of the first and second iterations of OPTIMOL. In the first iteration, a model is learned using the seed images. Classification is done on a subset of raw images collected from the web using the "accordion" query. Images with low likelihood ratios given by Eq.10 will be discarded. For the rest of the images, those with low entropies given by Eq.12 are incorporated into the permanent dataset, while the high entropy ones stay in the "cache set". In the second iteration, we update the model using only the images held in the "cache set" from the previous iteration. Classification is done on the new raw images as well as those in the cache. In a similar way, we append the images with high likelihood ratio and low entropy to the permanent dataset and hold those with high likelihood ratio and high entropy in the "cache set" for the next iteration of model learning. This continues till the system comes to a halt.

[2]. In all cases, OPTIMOL collected more positive images than the Caltech 101-Human, the LabelMe dataset and the approach in [2], with very few mistakes. Note that all of these results are achieved without any human intervention, thus suggesting the viability of OPTIMOL as an alternative to costly human dataset collection.

4.4. Exp.2: Classification

To demonstrate that OPTIMOL not only collects large datasets of images, but also learns good models for object classification, we conduct an experiment using the Fergus ICCV'05 dataset under the settings as in [8]. 7 object category models are learnt from the same training sets used by [8]. Similarly to [8], we use a validation set to train a 7-way SVM classifier to perform object classification. The feature vector of the SVM classifier is a vector of 7 entries, each denoting the image likelihood given each of the 7 class models. The results are shown in Fig.6, where we achieve an average performance of 74.8%. This result is comparable to (slightly better than) the 72.0% achieved by [8]. Our results show that OPTIMOL is capable of learning reliable object models.

4.5. Exp.3: Comparison of incremental learning and batch learning

In this experiment, we compare both the computation time and accuracy of the incremental learning and the batch

airplane	76.0	14.0	0.3	5.3	0.3	0.3	4.8
car	1.0	94.5	0.3	4.5	0.3	0.3	0.3
face	0.5	1.4	82.9	3.7	0.5	0.5	11.5
guita	2.2	4.9	5.6	60.4	13.3	0.2	13.3
leopard	1.0	2.0	1.0	5.0	89.0	1.0	2.0
motorbike	0.3	5.5	0.3	5.5	1.0	67.3	20.5
watch	1.7	5.5	17.7	11.0	5.5	5.0	53.6

Figure 6. Confusion table for Exp.2. We use the same training and testing datasets as in [8]. The average performance of the OPTIMOL-trained classifier is 74.82%, whereas [8] reports 72.0%.

learning (Fig.7). Due to the space limit, all results shown here are collected from the "euphonium" dataset; other datasets yield similar behavior. Fig.7(a) shows that the incremental learning method yields a better dataset than the batch method. Fig.7(b) illustrates that by not having to train with all available images at every iteration, OPTIMOL is more computationally efficient than a batch method. Finally, we show a ROC comparison of OPTIMOL vs. the batch method. In our system, the classifiers change every iteration according to the updates of the models. It is therefore not meaningful to show the ROC curves for the intermediate step classifiers. Thus, an ROC is displayed to compare the classification performance of the model learned by the batch learning and the final model of the incremental learning. Classifier quality is measured by the area under its ROC curve.

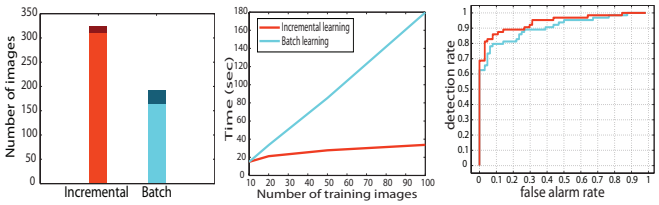

Figure 7. Batch vs. Incremental Learning (case study for the "euphonium" category). Left: the number of images retrieved by incremental learning and batch learning with false alarms represented as a darker hat on the top of each bar. Middle: running time comparison of batch learning and OPTIMOL's incremental learning method as a function of number of training images. The incrementally learned model is initialized by using the batch mode on 10 training images, which takes the same time as batch method does. After initialization, incremental learning is more efficient compared to the batch method. Right: Receiver Operating Characteristic (ROC) Curves of the incrementally learned model (green lines) versus the model learned by using the same seeds images (red line). The area under the ROC curve of OPTIMOL is 0.94, while it is 0.90 for batch learning.

5. Conclusion and future work

We have proposed a new approach (OPTIMOL) for image dataset collection and model learning. Our experiments show that as a fully automated system, OPTIMOL achieves accurate dataset collection result nearly as good as those of humans. In addition, it provides a useful annotation of the objects in the images. Further experiments show that the models learnt by OPTIMOL are competitive with the current state-of-the-art model object classification. Human labor is one of the most costly and valuable resources in research. We provide OPTIMOL as a promising alternative to collect larger image datasets with high accuracy. For future studies, we will further improve the performance of OPTIMOL by refining the model learning step and introducing more descriptive object models.

References

[1] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *PAMI*, 26(11):1475–1490, 2004.

[2] T. Berg and D. Forsyth. Animals on the web. In *Proc. Computer Vision and Pattern Recognition*, 2006.

[3] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[4] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. In *Workshop on Generative-Model Based Vision*, 2004.

[5] L. Fei-Fei, R. Fergus, and P. Perona. One-Shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006.

[6] L. Fei-Fei and P. Perona. A Bayesian hierarchy model for learning natural scene categories. *CVPR*, 2005.

[7] H. Feng and T. Chua. A bootstrapping approach to annotating large image collection. *Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval*, pages 55–62, 2003.

[8] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning Object Categories from Googles Image Search. *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, 2, 2005.

[9] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc. Computer Vision and Pattern Recognition*, pages 264–271, 2003.

[10] R. Fergus, P. Perona, and A. Zisserman. A visual category filter for Google images. In *Proc. 8th European Conf. on Computer Vision*, 2004.

[11] T. Hofmann. Probabilistic latent semantic indexing. *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57, 1999.

[12] T. Kadir and M. Brady. Scale, saliency and image description. *International Journal of Computer Vision*, 45(2):83–105, 2001.

[13] S. Krempp, D. Geman, and Y. Amit. Sequential learning with reusable parts for object detection. Technical report, Johns Hopkins University, 2002.

[14] D. Lowe. Object recognition from local scale-invariant features. In *Proc. International Conference on Computer Vision*, pages 1150–1157, 1999.

[15] R. Neal and G. Hinton. A view of the EM algorithm that justifies incremental, sparse and other variants. In M. Jordan, editor, *Learning in Graphical Models*, pages 355–368. Kluwer academic press, Norwell, 1998.

[16] B. Russell, A. Torralba, K. Murphy, and W. Freeman. Labelme: a database and web-based tool for image annotation. 2005.

[17] J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman. Discovering object categories in image collections. In *Proc. International Conference on Computer Vision*, 2005.

[18] E. Sudderth, A. Torralba, W. Freeman, and A. Willsky. Describing visual scenes using transformed dirichlet processes. *Advances in Neural Information Processing Systems*, 18:1297–1304, 2005.

[19] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical Dirichlet processes. *To appear in Journal of the American Statistical Association*, 2006.

[20] G. Wang, Y. Zhang, and L. Fei-Fei. Using dependent regions for object categorization in a generative framework, 2006.

[21] K. Yanai and K. Barnard. Probabilistic web image gathering. *ACM SIGMM workshop on Multimedia information retrieval*, pages 57–64, 2005.

Figure 5. Image collection and annotation results by OPTIMOL. Due to the space limit, we only show the annotation figures for eighteen categories, and for the remaining five categories, only the image collection results are shown here (Refer to the bar plots shown in the bottom row). The notations of the bars are provided at the bottom. The first 18 blocks are presented in two columns, with each row of the left (right) half page representing a category. Let us use "Laptop" as an example. The left sub-panel gives 4 sample annotation results (bounding box indicates the estimated locations and sizes of the laptop). The right sub-panel shows the comparison of the number of images in "Laptop" category given different datasets. The blue bar indicates the number of "Laptop" images in LabelMe dataset, the yellow bar the number of images in Caltech 101-Human. The OPTIMOL result is displayed using the red and green bars. The red bar represents the number of images retrieved for the "Laptop" category in Caltech 101-Web, the dark part on the top of the red bar is the number of False Positives. The green bar shows the number of clean images retrieved from the "Laptop" category in the Web-23 dataset. Again, the dark part on the top is the number of False Positives. In the last row of the figure, five bar plots for five different object categories are shown. The colors of the bars have the same meaning as the first 18 blocks. Since the pictures in the "face" category of Caltech 101-Human were taken by camera instead of downloading from the web, the raw images of the "face" category are not available. All of our results have been put online at http://vision.cs.princeton.edu/projects/OPTIMOL.htm