Learning Object Categories from Google's Image Search

R. Fergus[1] L. Fei-Fei[2] P. Perona[2] A. Zisserman[1]

[1]Dept. of Engineering Science
University of Oxford
Parks Road, Oxford
OX1 3PJ, U.K.
{fergus,az}@robots.ox.ac.uk

[2]Dept. of Electrical Engineering
California Institute of Technology
MC 136–93, Pasadena
CA 91125, U.S.A.
{feifeili,perona}@vision.caltech.edu

Abstract

Current approaches to object category recognition require datasets of training images to be manually prepared, with varying degrees of supervision. We present an approach that can learn an object category from just its name, by utilizing the raw output of image search engines available on the Internet. We develop a new model, TSI-pLSA, which extends pLSA (as applied to visual words) to include spatial information in a translation and scale invariant manner. Our approach can handle the high intra-class variability and large proportion of unrelated images returned by search engines. We evaluate the models on standard test sets, showing performance competitive with existing methods trained on hand prepared datasets.

1. Introduction

The recognition of object categories is a challenging problem within computer vision. The current paradigm [1, 2, 5, 10, 14, 15, 21, 22, 24] consists of manually collecting a large training set of good exemplars of the desired object category; training a classifier on them and then evaluating it on novel images, possibly of a more challenging nature. The assumption is that training is a hard task that only needs to be performed once, hence the allocation of human resources to collecting a training set is justifiable. However, a constraint to current progress is the effort in obtaining large enough training sets of all the objects we wish to recognize. This effort varies with the size of the training set required, and the level of supervision required for each image. Examples range from 50 images (with segmentation) [15], through hundreds (with no segmentation) [10], to thousands of images [14, 23].

In this paper we propose a different perspective on the problem. There is a plentiful supply of images available at the typing of a single word using Internet image search engines such as Google, and we propose to learn visual models directly from this source. However, as can be seen in Fig. 1, this is not a source of pure training images: as many as 85% of the returned images may be visually unrelated to the intended category, perhaps arising from polysemes (e.g. "iris" can be iris-flower, iris-eye, Iris-Murdoch). Even the 15% subset which do correspond to the category are substantially more demanding than images in typical training sets [9] – the number of objects in each image is unknown and variable, and the pose (visual aspect) and scale are uncontrolled. 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 whatever visual category we wish. In our previous work we have considered this source of images for training [11], but only for the purpose of re-ranking the images returned by the Google search (so that the category of interest has a higher rank than the noise) since the classifier models learnt were too weak to be used in a more general setting, away from the dataset collected for a given keyword.

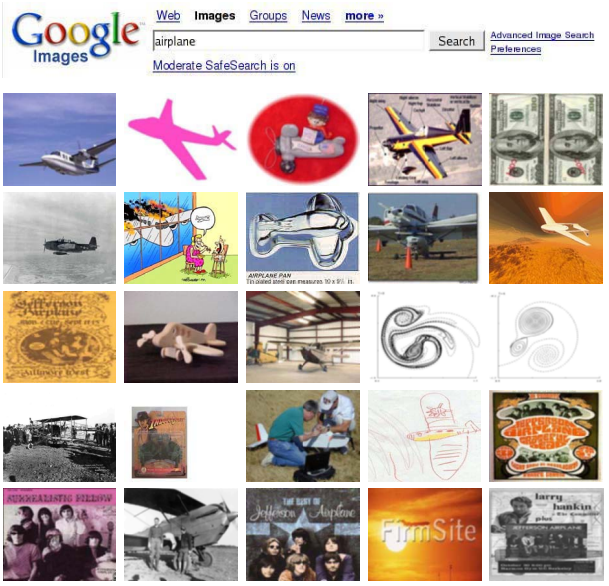


Figure 1: Images returned from Google's image search using the keyword "airplane". This is a representative sample of our training data. Note the large proportion of visually unrelated images and the wide pose variation.

Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05)
1550-5499/05 $20.00 © 2005 **IEEE**

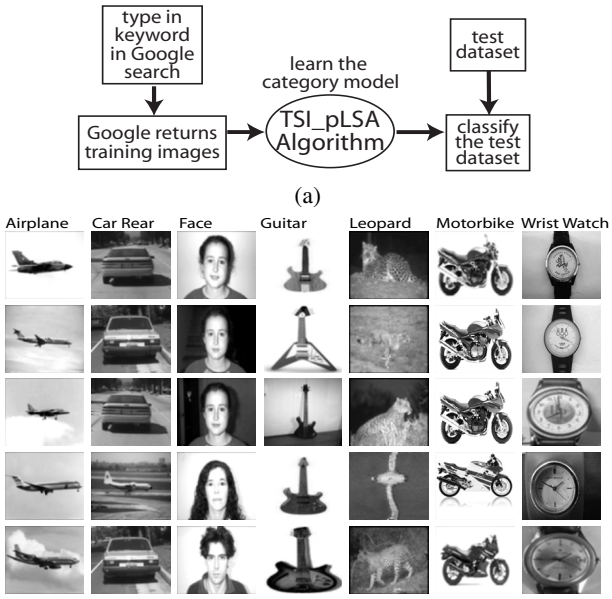


Figure 2: **(a)** A summary of our approach. Given the keywords: airplane, car rear, face, guitar, leopard, motorbike, wrist watch we train models from Google's image search with no supervision. We test them on a collection of 2148 images from the Caltech datasets and others, showing the top 5 images returned for each keyword in **(b)**.

The problem of extracting coherent components from a large corpus of data in an unsupervised manner 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) [12] and its hierarchical Bayesian form, Latent Dirichlet Allocation (LDA) [4]. Recently, these two approaches have been applied to the computer vision: Fei-Fei and Perona [8] applied LDA to scene classification and Sivic *et al.* applied pLSA to unsupervised object categorisation. In the latter work, the Caltech datasets used by Fergus *et al.* [10] were combined into one large collection and the different objects extracted automatically using pLSA.

In this paper, we adopt and extend pLSA methods to incorporate spatial information in a translation and scale-invariant manner and apply them to the more challenging problem of learning from search engine images. To enable comparison with existing object recognition approaches, we test the learnt models on standard datasets.

2. Approach

Before outlining our approaches, we first review pLSA and its adaption to visual data, following Sivic *et al.*

We describe the model using the terminology of the text literature, while giving the equivalence in our application. We have a set of D documents (images), each containing regions found by interest operator(s) whose appearance has been vector quantized into W *visual words* [20]. 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 regions in total. The model has a single latent *topic* variable, z, associating the occurrence of word w to document d. More formally:

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

Thus we are decomposing a $W \times D$ matrix into a $W \times Z$ matrix and a $Z \times W$ one. Each image is modeled as a mixture of topics, with $P(w|z)$ capturing the co-occurrence of words within a topic. There is no concept of spatial location within the model. The densities of the model, $P(w|z)$ and $P(z|d)$, are learnt using EM. The E-step computes the posterior over the topic, $P(z|w, d)$ and then the M-step updates the densities. This maximizes the log-likelihood of the model over the data:

$$L = \prod_{d=1}^{D} \prod_{w=1}^{W} P(w, d)^{n(w,d)} \tag{2}$$

In recognition, we lock $P(w|z)$ and iterate with EM, to estimate the $P(z|d)$ for the query images. 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 datasets and the other 50% were background scenes from the Caltech datasets. The regions are coloured 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 image) and green for the second (which picks out background images). $P(z|d)$ is shown above each image.

2.1. Absolute position pLSA (ABS-pLSA)

Previous work with pLSA applied to images did not use location information and we now extend the pLSA model to incorporate it. A straightforward way to do this 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 $(W \times X) \times Z$:

$$P(w, x, d) = \sum_{z=1}^{Z} P(w, x|z)P(z|d)P(d) \tag{3}$$

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. However, it will provide a useful comparison with our next approach.

2.2. Translation and Scale invariant pLSA (TSI-pLSA)

The shortcomings of the above model are addressed by introducing a second latent variable, c, which represents the position of the centroid of the object within the image, as well as its x-scale and y-scale, making it a 4-vector specifying a bounding box. As illustrated in Fig. 3(c), location x is

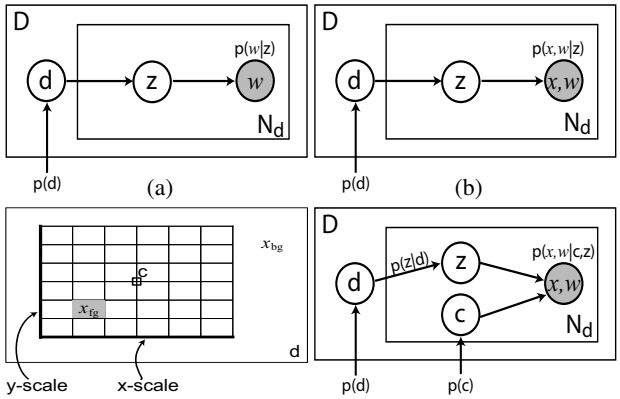


Figure 3: **(a)** Graphical model of pLSA. **(b)** Graphical model of ABS-pLSA. **(c)** The sub-window plus background location model. **(d)** Graphical model for translation and scale invariant pLSA (TSI-pLSA).

now modeled relative to the centroid c, over a sub-window of the image. Within the sub-window, there are X_{fg} location bins and one large background bin, x_{bg}, giving a total of $X = X_{fg} + 1$ locations a word can occur in. The word and location variables are then modeled jointly, as in section 2.1. This approach means that we confine our modeling of location to only the object itself where dependencies are likely to be present and not the background, where such correlations are unlikely. The graphical model of this approach is shown in Fig. 3(d).

We do not model an explicit $P(w, x|c, z)$, since that would require establishing correspondence between images as c remains in an absolute coordinate frame. Rather, we marginalize out over c, meaning that we only model $P(w, x|z)$:

$$P(w, x|z) = \sum_c P(w, x, c|z) = \sum_c P(w, x|c, z)P(c) \tag{4}$$

$P(c)$ here is a multinomial density over possible locations and scales, making for straightforward adaptations of the standard pLSA learning equations: $P(w, x|z)$ in (3) is substituted with the expression in (4). In learning we aggregate the results of moving the sub-window over the locations c.

Due to the high dimensionality of the space of c, it is not possible to marginalize exhaustively over scale and location within the image. Instead we use a small set of c, proposed in a bottom up manner for each topic.

2.2.1 Proposing object centroids within an image

We first run a standard pLSA model on the corpus and then fit a mixture of Gaussians with $k = \{1, 2, \ldots, K\}$ components to the location of the regions, weighted by $P(w|z)$ for the given topic. The idea is to find clumps of regions that belong strongly to a particular topic, since these may be the object we are trying to model. The mean of the component gives the centroid location while its axis-aligned variance gives the scale of the sub-window in the x and y directions. We try different number of components, since there may be clumps of regions in the background separate from the object, requiring more than one component to fit. This process gives us a small set (of size $C = K(K + 1)/2$) of values of c to sum over for each topic in each frame. We use a flat density for $P(c)$ since we have no more confidence in any one of the c being the actual object than any other. Fig. 4(a)-(c) shows the pLSA model using to propose centroids for the TSI-pLSA model, which are shown as dashed lines in Fig. 4(d)-(f). In the example, $K = 2$ and $Z = 2$.

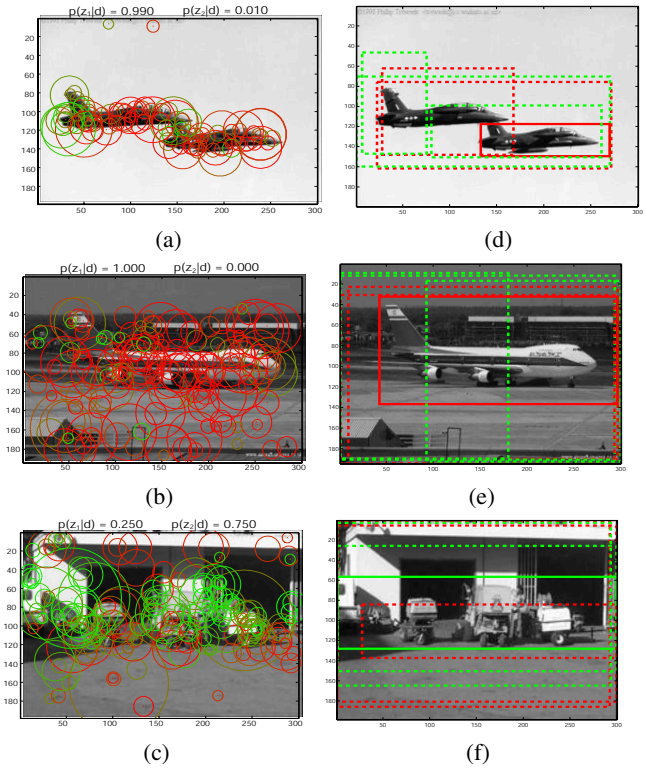


Figure 4: **(a)-(c)** Two airplane and one background image, with regions superimposed, coloured according to topic of a learnt pLSA model. Only a subset of regions are shown for clarity. **(d)-(f)** The same images as in **(a)-(c)** but showing the bounding boxes proposed by the pLSA model with dashed lines. The solid rectangle shows the centroid with highest likelihood under a TSI-pLSA model, with the colour indicating topic (the red topic appears to select airplanes). **(d)** shows multiple instances being handled correctly. **(e)** shows the object being localized correctly in the presence of background clutter.

In recognition, there is no need to learn a standard pLSA model first to propose different values of c. Instead, the average word density over the sub-window ($\hat{P}(w|z) = \sum_{x_{\text{fg}}} P(w, x|z)$) can be used to weight each region and then compute putative centroids in the manner above. Having obtained a set of centroids using $\hat{P}(w|z)$, recognition proceeds by locking $P(w, x|z)$ and iterating to find $P(z|d)$ for the novel images. In estimating $P(z|d)$, all states of c are summed over, thus once convergence is reached, we find c^*, the value of c within a frame which has the highest likelihood (shown in Fig. 4(d)-(f) as a solid box).

2.2.2 Observations about TSI-pLSA

- Multiple object instances in a frame can be captured with $k > 1$, with their information being combined by the marginalisation process. See Fig. 4(d) for an example.
- The model is entirely discrete, consisting of $WXZ + DZ$ parameters, thus is able to cope with multi-modal non-Gaussian distributions. This enables the model to handle multiple aspects of the object since the different word-locations densities for each aspect will appear as different modes within the $P(w, x|z)$ density.
- Since all three approaches use histograms, unless the object occupies a reasonably large proportion of the image, it will not have a sufficient number of detections to compete with regions on the background, meaning that the image is misclassified as background. While the sub-window approach of TSI-pLSA will help, it cannot overcome this effect entirely, so the object must still occupy a reasonable proportion of the image (1/4 to 1/5 of image area).

3. Implementation details

Having outlined the three approaches that we will investigate (pLSA; ABS-pLSA and TSI-pLSA), we now give specific details. All images are first converted to grayscale and resized to a moderate width (300 pixels in our experiments). No further normalization of any kind was carried out.

In view of the large number of parameters in our models, it is vital to have a large number of data points in each frame. We therefore use four different types of circular region detector to give a complete coverage of the image: (i) Kadir & Brady saliency operator [13]; (ii) Multi-scale Harris detector [17]; (iii) Difference of Gaussians, as used by Lowe [16] and (iv) Edge based operator, detailed below.

For certain categories, edge information is important and is not adequately captured by the first three region detectors. Inspired by the approach of Berg *et al.* [3], we first find edgels in the image and then locate a region at points drawn at random from the edgel set. The scale of the region is chosen by drawing from a uniform distribution over a sensible scale range (a radius range of 5-30 pixels). The total number of regions sampled is capped to give a number similar to the other three types of detector. On average, around $N = 700$ regions per image were found, with Kadir & Brady and the difference of Gaussians giving around 100 per image; the edge based detector 175, and multi-scale Harris 350.

Having found a large set of regions, we represent them by a SIFT descriptor, using 72 dimensions rather than the usual 128, resulting in larger histogram bins which are more appropriate for object categorization. The regions did not have their orientation normalised before histogramming, making them orientation variant. The descriptors are then vector quantized using a fixed codebooks of visual words, pre-computed 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 W visual words in total. In our experiments, we used $W = 350$. Regions could be quantized to any word, e.g. we did not restrict edge regions to only be allocated to the sub-section of the codebook formed from edge regions alone.

The two approaches with spatial densities used a grid of moderate coarseness, mindful of the need to keep the number of parameters to a reasonable level. The sub-window used in the experiments had a 6×6 grid, giving $X = 37$. Training a TSI-pLSA model with $Z = 8$, $D \sim 500$ and the aforementioned parameters takes roughly 30 minutes using a Matlab implementation. ABS-pLSA takes approximately the same time. pLSA takes around half a minute. 100 iterations of EM were used. Assuming $X = 37$, $W = 350$, $D = 500$, $N = 700$, $Z = 8$, we have $109,200$ parameters in the model which are estimated from $350,000$ data points, giving a data/parameter ratio of just over 3, the minimum sensible level.

4. Datasets

The experiments used 7 different object categories in 9 datesets. 5 of these were the Caltech datasets [9]: Airplane; Car (Rear); Leopard; Face and Motorbike. Additionally, more challenging datasets for the car and motorbike classes were taken from PASCAL [6], using the `test2` set of foreground/background training and test images. Finally, Guitar and Wrist watch were the two remaining categories. For each category four subsets of data were compiled: two hand gathered sets, where each image contains at least one instance of the object and two automatically gathered sets with may be contaminated with images unrelated to the category.

1. Prepared training set (PT): Manually gathered frames. In the case of the Caltech datasets, the training frames from [10] were used. The pose of the object is quite constrained within these frames. The PASCAL datasets contained large viewpoint and pose variation.

2. Prepared test set (P): Manually gathered frames, disjoint although statistically similar to (PT). For the Caltech datasets, the test frames from [10] were used. Again, the

pose is fairly constrained. In contrast, the PASCAL datasets contained large viewpoint and pose variation.

3. Raw Google set (G): A set of images automatically downloaded from Google's Image Search[1], using the category name. See Fig. 1 for typical images downloaded using "airplane". Duplicates images 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 3 distinct groups:

i **Good images:** these are good examples of the keyword category, lacking major occlusion, although there may be a variety of viewpoints, scalings and orientations.

ii **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.

iii **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. Fig. 5 shows the recall-precision curves of the raw Google sets for each category.

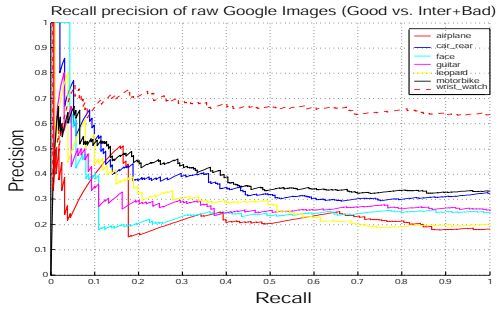


Figure 5: Recall precision curves of the raw output of Google's image search for the 7 keywords. Good labels count as positive examples while Intermediate and Junk labels are negative examples. Note the precision drops rapidly as the recall increases, leveling out at 20–30% for most categories.

4. Google validation set (V): An empirical observation (as seen in Fig. 5) is that the first few pages returned by Google tend to contain more good images than those returned later on. The idea is that we assume 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 drop off 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[2] we obtain the translations of the users keyword in the following languages: German, French, Spanish, Italian, Portugese and Chinese. Since each translation returns a different set of images, albeit with the same drop off in quality, we automatically download the first few images from each different language, and combine to give a validation set of a reasonable size without a degradation in quality.

Using 7 different languages (including English), taking the first 5 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 set for "airplane". All datasets used are summarized in Table 1.

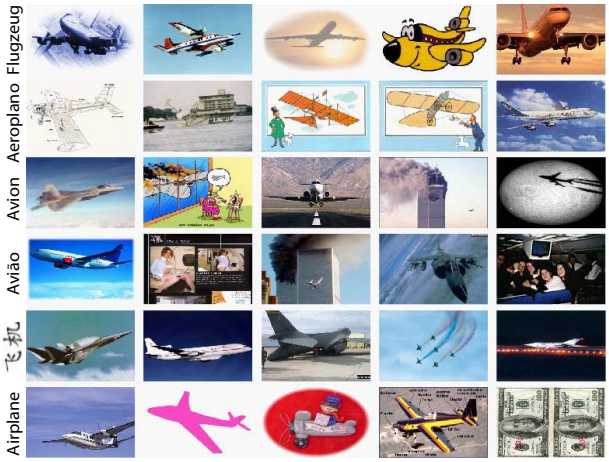


Figure 6: The entire validation set for "airplane" obtained automatically using Google's translation tool and Google's image search. The text by each row shows the translated keyword used to gather that particular row. The quality of the images is noticeably higher than those in Fig. 1.

	Size of Dataset				Distrib. of Google Images (%)		
Category	PT	P	V	G	Good	Inter.	Junk
Airplane	400	400	30	874	18.1	8.6	73.3
Cars Rear	400	400	30	596	32.2	12.9	54.9
Face	217	217	30	564	24.3	21.3	54.4
Guitar	450	450	25	511	25.3	30.5	44.2
Leopard	100	100	15	516	19.6	27.5	52.9
Motorbike	400	400	30	688	33.4	29.8	36.8
Wrist watch	180	181	35	342	63.4	13.8	22.8
PASCAL Cars	272	275	-	-	-	-	-
PASCAL Cars Bg.	412	412	-	-	-	-	-
PASCAL Motorbike	214	202	-	-	-	-	-
PASCAL Motorbike Bg.	570	754	-	-	-	-	-
Caltech Bg.	400	400	-	-	-	-	-
Cars Rear Bg.	400	400	-	-	-	-	-

Table 1: Statistics of the datasets used in experiments. Columns 2 & 3: Size of the hand prepared training (PT) and test (P) datasets. Column 4: The number of validation (V) images automatically obtained. Column 5: The number of images automatically downloaded from Google's image search (G). The last 3 columns show the breakdown (for evaluation purposes) of the raw Google images for each category. Note the low proportion of good examples present in the majority of categories.

5. Selection of the final classifier

There are two issues to consider when training our models: (i) the optimal number of topics, Z ; (ii) which subset of these topics should be used to form a classifier for use

[1]While in this paper Google's image search was used exclusively (`http://www.google.com/imghp`), any other image search engine may be used provided that the images can be gathered in an automated manner

[2]`http://translate.google.com/translate_t`

Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05)
1550-5499/05 $20.00 © 2005 **IEEE**

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 dataset.

The number of topics to use in experiments was determined empirically: the performance of the face and airplane categories was recorded as the number of topics was varied when training from Google and a stable peak picked at $Z = 8$ (see Fig. 8(b)). This value was then used for all experiments involving Google data. Having trained an 8 topic model, each topic is run across the validation set and single topic that performed best is picked to be the classifier used in testing.

6. Experiments

Several sets of experiments were performed:

A **Caltech experiments.** Training on a 50-50 mix of prepared data (PT) from the Caltech datasets (including watch and guitar) and data from the Caltech background dataset. Testing, in classification setting, on prepared data (P) and test data from the Caltech background. In the case of Cars Rear, the Caltech background was substituted for the Cars Rear Background for a more realistic experiment. All 3 methods (pLSA, ABS-pLSA and TSI-pLSA) were run with 2 topics (reflecting the true number of components in the training and test data).

B **PASCAL experiments.** Training on prepared data (PT) of the two PASCAL datasets (cars, motorbikes) and their background images. Testing on prepared data (P) of PASCAL. Training was unsupervised, in the manner of [19], with the foreground and background data combined into one training set. All 3 methods (pLSA, ABS-pLSA and TSI-pLSA) were run with 6 topics and the best topic or equally weighted pair of topics chosen based on their performance on (PT). These experiments are designed to investigate the difference between ABS-pLSA and TSI-pLSA and measure localisation as well as detection performance.

C **Google experiments.** Training on raw Google data (G); the best topic is then picked using the validation set (V), which is then tested on prepared data (P), measuring classification performance. All 3 methods were evaluated with 8 topics. The ability of our algorithm to train directly from Google data is evaluated.

D **Search engine improvement experiments.** In the manner of [11]. Training on raw Google data (G); picking the best topic using (V) and using it to re-rank the Google images (G). The idea is that the recall-precision curve of good images should be improved by the models learnt.

	pLSA	ABS	TSI	pLSA	ABS	TSI
Category	Prep.	Prep.	Prep.	Google	Google	Google
(A)irplane	17.7	13.2	4.7	24.7	17.2	15.5
(C)ars Rear	2.0	0.2	0.7	21.0	13.2	16.0
(F)ace	22.1	11.5	17.0	20.3	36.4	20.7
(G)uitar	9.3	10.0	14.4	17.6	62.0	31.8
(L)eopard	12.0	12.0	11.0	15.0	16.0	13.0
(M)otorbike	19.0	6.0	7.0	15.2	18.5	6.2
(W)rist watch	21.6	7.7	15.5	21.0	20.5	19.9
PASCAL Car	31.7	33.0	25.8	-	-	-
PASCAL Motorbike	33.7	30.2	25.7	-	-	-

Table 2: Comparison of different methods trained on: prepared data (first three columns) and raw Google data (rightmost three columns). All methods were tested on prepared data. The task is classification, with the figures being the error rate at point of equal-error on an ROC curve. The error margins are roughly $+/-2\%$.

6.1. Caltech and PASCAL experiments

The results of experiments A, B in a classification setting are given in Table 2, columns 2–4. The results on the Caltech datasets show that (except for the leopard and guitar categories), the incorporation of location information gives a significant reduction in error rate. However, due to the constrained pose of instances within the images, the ABS-pLSA model often does as well if not better than the TSI-pLSA model (e.g. wrist watch and guitar). By contrast, when testing on the PASCAL datasets which contain far greater pose variability, the TSI-pLSA model shows a clear improvement over ABS-pLSA. See Fig. 7 for some examples of the TSI-pLSA model correctly detecting and localising cars in PASCAL test images. See Table 3 for a comparison between TSI-pLSA and other current approaches on the PASCAL datasets.

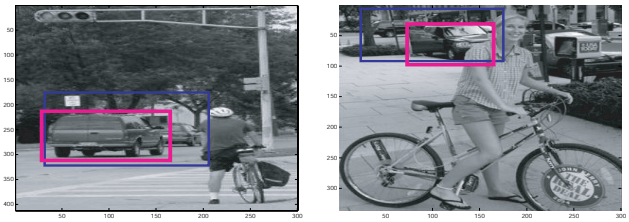


Figure 7: Examples of the TSI-pLSA model, trained on the prepared PASCAL Cars data, correctly localising test instances. The ground truth bounding box is shown in magenta, while the proposed bounding box, c^*, is shown in blue.

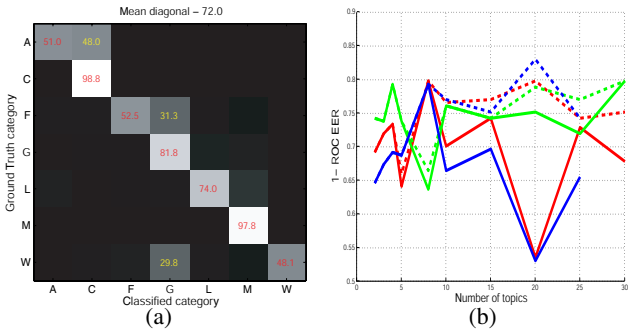
6.2. Google experiments

The results of experiment C in a classification setting are given in the last 3 columns of Table 2. As expected, training directly on Google data gives higher error rates than training on prepared data. For around half the categories, the use of location information reduces the error significantly, although only in the case of motorbikes and airplanes is TSI-pLSA better their either of the other two approaches.

Both ABS-pLSA and TSI-pLSA perform notably poorly on the guitar dataset. This may be explained by the fact that all the prepared data has 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. An example of a TSI-pLSA model learnt from Google data is shown in Fig. 9. In the case of Motorbikes, the common words correspond to parts of the wheels of the bike and the exhaust/tail structure. In the case of Leopards, the textured fur of the animal is captured by the most common regions. However, their location densities are spread out, reflecting the diffuse spatial representation of the animal.

The confusion table of the seven classes is shown in Fig. 8(a). For the majority of classes the performance is respectable. Notable confusions include: airplanes being classified as cars rear (both have lots of horizontal edges); the guitar model misclassifying faces and wrist watches (due to the weak guitar model). See also Fig. 2 for the TSI-pLSA models used in a retrieval application.



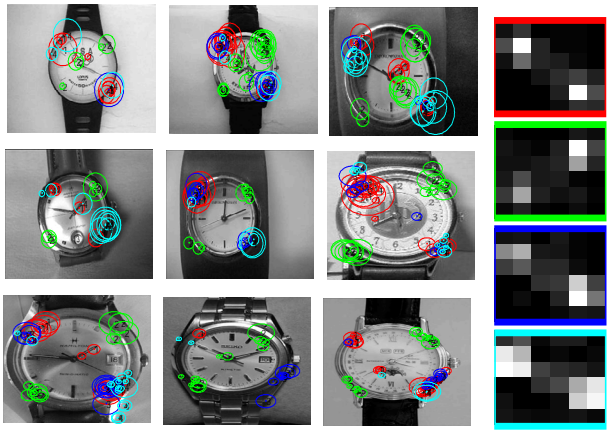
(a) (b)

Figure 8: **(a)** Confusion table for the 7 classes. The row is the ground truth label; the column indicates the classification. **(b)** "Face" keyword. Performance of models trained on Google data, tested on prepared data, with a varying number of topics. Red - pLSA; Green - ABS-pLSA; Blue - TSI-pLSA. Solid lines indicate performance of automatically chosen topic within model. Dashed lines indicate performance of best topic within model.

In Table 3 we compare our performance to existing approaches to object recognition for experiments B and C, noting their degree of supervision.

Dataset	TSI-pLSA	[10]	[18]	[15]
Expt. B	Img. labels	Img. labels	Img. labels	Segmented
PASCAL Car	25.8 / 0.062	-	-	34.2 / 0.181
PASCAL Motorbike	25.7 / 0.249	-	-	31.7 / 0.341
Expt. C	None	Img. labels	Img. labels	Segmented
Airplane	15.5	7.0	11.1	-
Cars Rear	16.0	9.7	8.9	6.1
Face	20.7	3.6	6.5	-
Leopard	13.0	10.0	-	-
Motorbike	6.2	6.7	7.8	6.0

Table 3: Comparison of performance and supervision with other weakly supervised training approaches for experiments B and C. The first value is the ROC EER classification rate; the second (where given) is the average precision [6] in localisation. In PASCAL experiments (B), the classification performance is better than [15], but is less good at localisation. In Google experiments (C), the results for Leopard and Motorbike are comparable to other approaches. Airplane and Cars Rear are around 10% worse. However the supervision requirements of the other methods are greater.



(a) (b)

Figure 9: **(a)** Watches from the prepared dataset, with regions superimposed that belong to the 4 most common visual words (irrespective of location) from the automatically chosen topic of the Google-trained TSI-pLSA watch model. Each colour 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 presentation, different parts of the bezel are quantized to different words. **(b)** The location densities of the 4 most common words shown in **(a)**. White corresponds to a high probability, black to a low one. Note their tightly constrained, multi-modal, nature.

6.3. Investigation of number of topics

In Fig. 8(b) we vary the number of topics in a face model trained on Google data and evaluate: (a) the automatically chosen topic, and (b) the actual best topic on the prepared test set. The performance of all three methods does not seem to increase too much beyond 10 topics. This is due to the selection of a single topic – picking a combination of topics is likely to yield superior results. The difficulty is in deciding which ones to pick: the validation set picks the best topic (or close to it) reliably up to 8 topics or so. Beyond this its performance drops off significantly. For small numbers of topics, the models are unreliable, while it is difficult to pick the correct topic from very large models. The point of compromise seems to be in region of 5-10 topics (the curves are very similar for different categories), hence the use of $Z = 8$.

6.4. Improving Google's image search

As in Fergus *et al.* [11], the models learnt from Google data may be directly employed to improve the quality of the image search by re-ranking the images using the topic chosen from the validation set. As can be seen in Fig. 5, the native performance of Google's search is quite poor. Fig. 10 shows the improvement in precision achieved by using the best topic chosen from an 8 topic model trained on the raw data. Figs. 11 and 12 show the top ranked images for each topic for the pLSA and TSI-pLSA approaches respectively, using the "motorbike" keyword.

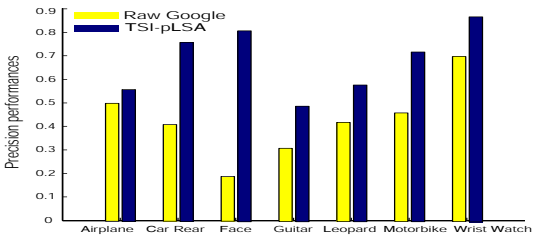


Figure 10: Improvement in the precision at 15% recall obtained with an 8 topic TSI-pLSA model (blue) over the raw Google ranking (yellow). This level of recall corresponds to a couple of web pages worth of images.

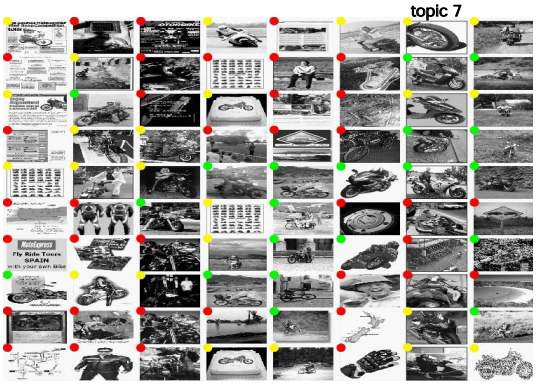


Figure 11: Top ranked images for each topic of an 8 topic pLSA model trained on Google data (G), using the keyword "motorbike". Topic selected by validation set (V) was topic 7. The coloured dots in the top-left corner of each image show the ground truth labels (Green = Good; Yellow = Intermediate and Red = Junk).

7. Summary and Conclusions

We have proposed the idea of training using just the objects name by bootstrapping with an image search engine. The training sets are extremely noisy yet, for the most part, the results are competitive (or close to) existing methods requiring hand gathered collections of images. This was achieved by improving state-of-the-art pLSA models with spatial information. It would be interesting to compare our methods to [7], trained from the Google Validation set. However there are many open issues: the choice of features; better centroid proposals; the use of fixed background densities to assist learning; how to pick the most informative topics; the number of topics to use; the introduction of more sophisticated LDA models using priors.

Acknowledgements

Financial support was provided by: EC Project CogViSys; UK EPSRC; Caltech CNSE and the NSF. This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views. Thanks to Rebecca Hoath and Veronica Robles for image labelling. We are indebted to Josef Sivic for his considerable help with many aspects of the paper.

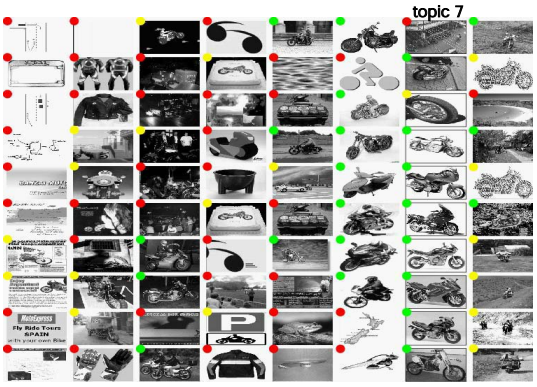


Figure 12: As per Fig. 11 but for an 8 topic TSI-pLSA model. Topic 7 was again the automatically selected topic. Note the increased consistency of each topic compared to pLSA.

References

[1] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *IEEE PAMI*, 20(11):1475–1490, 2004.

[2] K. Barnard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, and M. Jordan. Matching words and pictures. *JMLR*, 3:1107–1135, Feb 2003.

[3] A. Berg, T. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondence. In *Proc. CVPR*, June 2005.

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

[5] G. Csurka, C. Bray, C. Dance, and L. Fan. Visual categorization with bags of keypoints. In *Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22, 2004.

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

[7] L. Fei-Fei, R. Fergus, and P. Perona. A Bayesian approach to unsupervised one-shot learning of object categories. In *Proc. ICCV*, pages 1134–1141, Oct 2003.

[8] L. Fei-Fei and P. Perona. A Bayesian hierarchical model for learning natural scene categories. In *Proc. CVPR*, June 2005.

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

[10] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc. CVPR*, Jun 2003.

[11] R. Fergus, P. Perona, and A. Zisserman. A visual category filter for Google images. In *Proc. ECCV*. Springer-Verlag, May 2004.

[12] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, 1999.

[13] T. Kadir and M. Brady. Scale, saliency and image description. *IJCV*, 45(2):83–105, 2001.

[14] Y. LeCun, F. Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *Proc. CVPR*. IEEE Press, 2004.

[15] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *Workshop on Statistical Learning in Computer Vision, ECCV*, 2004.

[16] D. Lowe. Object recognition from local scale-invariant features. In *Proc. ICCV*, pages 1150–1157, Sep 1999.

[17] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In *Proc. ICCV*, 2001.

[18] A. Opelt, A. Fussenegger, and P. Auer. Weak hypotheses and boosting for generic object detection and recognition. In *Proc. ECCV*, 2004.

[19] J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman. Discovering object categories in image collections. Technical Report A. I. Memo 2005-005, Massachusetts Institute of Technology, 2005.

[20] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proc. ICCV*, Oct 2003.

[21] A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *Proc. CVPR*, pages 762–769, 2004.

[22] N. Vidal-Naquet and S. Ullman. Object recognition with informative features and linear classification. In *Proc. ICCV*, 2003.

[23] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. CVPR*, pages 511–518, 2001.

[24] M. Weber, M. Welling, and P. Perona. Unsupervised learning of models for recognition. In *Proc. ECCV*, pages 18–32, 2000.