Scene classification using a hybrid generative/discriminative approach

Anna Bosch, Andrew Zisserman and Xavier Muñoz

Abstract

We investigate whether dimensionality reduction using a latent generative model is beneficial for the task of weakly supervised scene classification. In detail we are given a set of labelled images of scenes (e.g. coast, forest, city, river, etc) and our objective is to classify a new image into one of these categories. Our approach consists of first discovering latent "topics" using probabilistic Latent Semantic Analysis (pLSA), a generative model from the statistical text literature here applied to a bag of visual words representation for each image, and subsequently training a multi-way classifier on the topic distribution vector for each image. We compare this approach to that of representing each image by a bag of visual words vector directly, and training a multi-way classifier on these vectors.

To this end we introduce a novel vocabulary using dense colour SIFT descriptors, and then investigate the classification performance under changes in the size of the visual vocabulary, the number of latent topics learnt, and the type of discriminative classifier used (k-nearest neighbour or SVM). We achieve superior classification performance to recent publications that have used a bag of visual word representation, in all cases using the authors' own datasets and testing protocols. We also investigate the gain in adding spatial information. We show applications to image retrieval with relevance feedback and to scene classification in videos.

Index Terms

Scene Classification, pLSA, Spatial Information.

A. Bosch is with the Computer Vision and Robotics Group, University of Girona, 17003, Girona (e-mail: aboschr@eia.udg.es)
A. Zisserman is with the Robotics Research Group, University of Oxford, Oxford OX1 3PJ (e-mail: az@robots.ox.ac.uk)
X. Muñoz is with the Computer Vision and Robotics Group, University of Girona, 17003, Girona (e-mail: xmunoz@eia.udg.es)

I. INTRODUCTION

Classifying scenes (such as mountains, forests, offices) is not an easy task owing to their variability, ambiguity, and the wide range of illumination and scale conditions that may apply. As was noted in [3], two basic strategies can be found in the literature. The first uses low-level features such as global colour or texture histograms, the power spectrum, etc, and is normally used to classify only a small number of scene categories (indoor versus outdoor, city versus landscape etc...) [29], [30]. The second strategy uses an intermediate representations before classifying scenes [9], [24], [32], and has been applied to cases where there are a larger number of scene categories (up to 15).

In this paper we follow the second strategy and introduce a classification algorithm based on a combination of unsupervised probabilistic Latent Semantic Analysis (pLSA) [16] followed by a discriminative classifier. The pLSA model was originally developed for topic discovery in a text corpus, where each document is represented by its word frequency. Here it is applied to images represented by the frequency of "visual words"[28]. The formation and performance of this "visual vocabulary" is investigated in depth. In particular we compare sparse and dense feature descriptors over a number of modalities (colour, texture, orientation). The approach is inspired in particular by three previous papers: (i) the use of pLSA on sparse features for recognizing compact object categories (such as Caltech cars and faces) in Sivic *et al.* [27]; (ii) the dense SIFT [21] like features developed in Dalal and Triggs [8] for pedestrian detection; and (iii) the semi-supervised application of Latent Dirichlet Analysis (LDA) for scene classification in Fei-Fei and Perona [9]. We have made extensions over all three of these papers both in developing new features and in the classification algorithm. Our work is most closely related to that of Quelhas *et al.* [25] who also use a combination of pLSA and supervised classification. However, their approach differs in using sparse features and is applied to classify images into only three scene types.

We compare our classification performance to that of four previous methods [9], [18], [24], [32] using the authors' own databases. This previous work uses varying levels of supervision in training (compared to the unsupervised topic discovery developed in this paper): Fei-Fei and Perona [9] requires the category of each scene to be specified during learning (in order to discover the *themes* (topics) of each category) – we do not specify the category when discovering

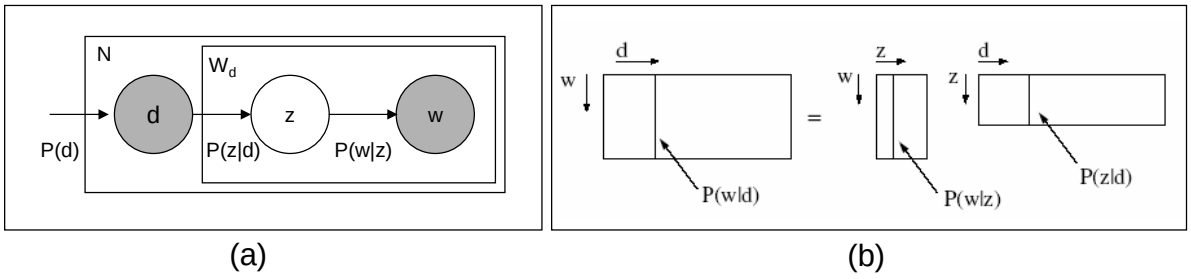


Fig. 1. (a) pLSA graphical model. Nodes inside a given box (plate notation) indicate that they are replicated the number of times indicated in the top left corner (N=number of images;W_d=number of (visual) words per image). Filled circles indicate observed random variables; unfilled are unobserved. (b) The goal is to find the topic specific word distributions $P(w|z)$ and corresponding document specific mixing proportions $P(z|d)$ which make up the observed document specific word distribution $P(w|d)$.

topics; Oliva and Torralba [24] requires a manual ranking of the training images into 6 different properties; and Vogel and Schiele [32] requires a manual classification of 60K local patches from the training images into one of 9 *semantic concepts*. As will be seen, we achieve superior performance in all cases. Lazebnik *et al*. [18] do not use an intermediate topic representation, but improve performance by adding spatial information over the bag of words model. We compare a number of methods that include both latent models and spatial information, and demonstrate improved results over [18].

We briefly give an overview of the pLSA model in Section II. Then in Section III we describe the hybrid classification algorithm based on applying pLSA to images followed by discriminative classification. Section IV describes the features used to form the visual vocabulary and the principal parameters that are investigated. A description of datasets and a detailed description of the experimental procedure is given in Section V. Section VI reports the principal investigation of the paper – first we optimize the performance over changes in the vocabulary and number of latent topics, then we compare the hybrid classifier to a more standard approach of classifying on the bag of words histograms directly. Section VII then introduces three models that include spatial information and compares their performance to the model of Lazebnik *et al*. [18]. In Section IX we demonstrate applications of the hybrid algorithm to relevance feedback, scene classification in videos, and segmentation. In Section X we discuss the ambiguities and difficulties of the scene classification task.

This paper is an expanded version of [4]. The extensions include the comparison of the classifiers (K Nearest Neighbour and SVM), Section VII on spatial information, and evaluations on new datasets (that of [18] and Caltech 101 [20]).

II. pLSA MODEL

Probabilistic Latent Semantic Analysis (pLSA) is a generative model from the statistical text literature [16]. In text analysis this is used to discover topics in a document using the bag of words document representation. Here we have *images* as *documents* and we discover *topics as object categories* (e.g. grass, houses), so that an image containing instances of several objects is modelled as a mixture of topics. The models are applied to images by using a *visual* analogue of a *word*, formed by vector quantizing colour, texture and SIFT feature like region descriptors (as described in Section IV). pLSA is appropriate here because it provides a correct statistical model for clustering in the case of multiple object categories per image. We will explain the model in terms of images, visual words and topics.

Suppose we have a collection of images $D = d_1,...,d_N$ with words from a visual vocabulary $W = w_1,...,w_V$. The data is a $V \times N$ co-occurrence table of counts $N_{ij} = n(w_i, d_j)$, where $n(w_i, d_j,)$ denotes how often the term w_i occurred in an image d_j. A latent variable model associates an unobserved topic variable $z \in Z = z_1,...,z_Z$ with each observation, an observation being the occurrence of a word in a particular image (w_i, d_j). We introduce the following probabilities: $P(d_j)$ denotes the probability of observing a particular image d_j, $P(w_i|z_k)$ denotes the conditional probability of a specific word conditioned on the unobserved topic variable z_k, and finally $P(z_k|d_j)$ denotes an image specific probability distribution over the latent variable space. Using these definitions, the generative model is the following:

- Select an image d_j with probability $P(d_j)$
- Pick a latent topic z_k with probability $P(z_k|d_j)$
- Generate a word w_i with probability $P(w_i|z_k)$.

As a result one obtains an observation pair (w_i, d_j), while the latent topic variable z_k is discarded.

The graphical model representation is shown in Figure 1a corresponding to a joint probability $P(w, d, z) = P(w|z)P(z|d)P(d)$. Marginalizing out the latent variable z gives

$$P(w, d) = \sum_{z \epsilon Z} P(w, d, z) = P(d) \sum_{z \epsilon Z} P(w|z)P(z|d)$$

and thence from $P(w,d) = P(d)P(w|d)$, we obtain $P(w|d)$ as:

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

This amounts to a matrix decomposition as shown in Figure 1b with the constraint that both the topic vectors $P(w|z)$ and mixture coefficients $P(z|d)$ are normalized to make them probability distributions. Essentially, each image is modelled as a mixture of topics, the histogram for a particular document being composed from a mixture of the histograms corresponding to each topic. In particular each image is a convex combination of the Z topic vectors.

Following the likelihood principle, one determines $P(w|z)$, and $P(z|d)$ by maximization of the loglikelihood function:

$$L = \log P(D,W) = \sum_{d \epsilon D} \sum_{w \epsilon W} n(w,d) \log P(w,d) \tag{2}$$

This is equivalent to minimizing the Kullback-Leibler divergence between the measured empirical distribution and the fitted model. The model is fitted using the Expectation Maximization (EM) algorithm as described in [16]. Fitting the model involves determining the topic vectors which are common to all documents and the mixture coefficients which are specific for each document. The goal is to determine the model that gives high probability to the visual words that appear in the corpus.

III. Hybrid Classification

Training proceeds in two stages. First, the topic specific distributions $P(w|z)$ are learnt from the set of training images. Determining both $P(w|z)$ and $P(z|d_{train})$ simply involves fitting the pLSA model to the entire set of training images. In particular it is not necessary to supply the identity of the images (i.e. which category they are in) or any region segmentation. Each training image is then represented by a Z-vector $P(z|d_{train})$, where Z is the number of topics learnt. In the second stage a multi-class discriminative classifier is trained given the vector $P(z|d_{train})$ of each training image and its class label. For the discriminative stage we compare K Nearest Neighbours classifier (KNN) to a Support Vector Machine classifier (SVM). In more detail, the KNN selects the K nearest neighbours of the new image within the training database (using Euclidean distance). Then it classifies the test image according to the category label which is most represented within the K nearest neighbours. For the SVM classifier an exponential kernel

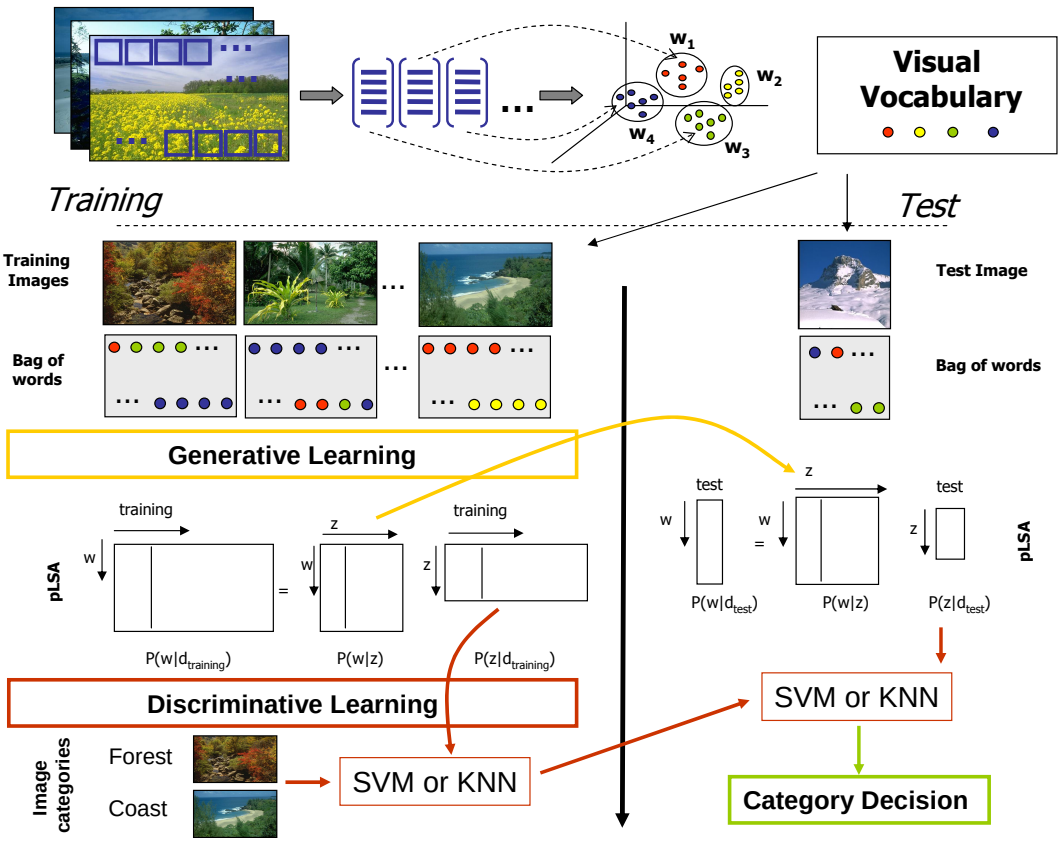


Fig. 2. Overview of visual vocabulary formation, learning and classification stages.

of the form $\exp -\alpha d$ is used, where d is the Euclidean distance between the vectors, and the scalar α is determined as described in [36] (we use the LIBSVM package [5] with the trade-off between training error and margin at $C = 1$). The multi-way classification is done using the one-versus-all rule: a classifier is learned to separate each class from the rest, and a test image is assigned the label of the classifier with the highest response.

Classification of an unseen test image similarly proceeds in two stages. First the document specific mixing coefficients $P(z|d_{test})$ are computed, and these are then used to classify the test images using a discriminative classifier. In more detail document specific mixing coefficients $P(z|d_{test})$ are computed using the fold-in heuristic described in [15]. The unseen image is projected onto the simplex spanned by the $P(w|z)$ learnt during training, i.e. the mixing coefficients $P(z_k|d_{test})$ are sought such that the Kullback-Leibler divergence between the measured distribution and $P(w|d_{test}) = \sum_{z \epsilon Z} P(w|z)P(z|d_{test})$ is minimized. This is achieved by running EM in

a similar manner to that used in learning, but now only the coefficients $P(z_k|d_{test})$ are updated in each M-step with the learnt $P(w|z)$ kept fixed. The result is that the test image is represented by a Z-vector. The test image is then classified by the multi-class discriminative classifier (KNN or SVM) as described above. Figure 2 shows graphically the hybrid generative/discriminative process for both training and testing.

IV. Visual words and visual vocabulary

In the formulation of pLSA, we compute a co-occurrence table, where each image is represented as a collection of visual words, provided from a visual vocabulary. This visual vocabulary is obtained by vector quantizing descriptors computed from the training images using k-means, see the illustration in the first part of Figure 2. Previously both sparse [7], [17], [28] and dense descriptors, e.g. [8], [19], [31], have been used. Here we carry out a thorough comparison over dense descriptors for a number of visual measures (see below) and compare to a sparse descriptor. We vary the size of the patches and degree of overlap, and compare normalized to unnormalized images. We then assess classification performance over four different image datasets described in Section V.

We investigate four dense descriptors, and compare their performance to a previously used sparse descriptor. In the dense case the important parameters are the size of the patches (N) and their spacing (M) which controls the degree of overlap:

Grey patches (dense). As in [31], and using only the grey level information, the descriptor is a N $\times$ N square neighbourhood around a pixel. The pixels are row reordered to form a vector in an N^2 dimensional feature space. The patch size tested are N = 5, 7 and 11. The patches are spaced by M pixels on a regular grid. The patches do not overlap when $M = N$, and do overlap when $M = 3$ (for N = 5, 7) and M = 7 (for N = 11).

Colour patches (dense). As above, but the colour information is used for each pixel. We consider the three colour components HSV and obtain a $N^2 \times 3$ dimensional vector. As in the grey level, we used N = 5, 7, and 11. We use HSV because of its similarities to the way humans tend to perceive colour and because it is less sensitive to shadow and shading.

Grey SIFT (dense). SIFT descriptors [21] are computed at points on a regular grid with spacing M pixels, here M = 5, 10 and 15. At each grid point SIFT descriptors are computed over circular support patches with radii r = 4, 8, 12 and 16 pixels. Consequently each point is represented

by n SIFT descriptors (where n is the number of circular supports), each is 128-dim. Multiple descriptors are computed to allow for scale variation between images. The patches with radii 8, 12 and 16 overlap. Note, the descriptors are rotation invariant.

Colour SIFT (dense). As above, but now SIFT descriptors are computed for each HSV component. This gives a 128×3 dim-SIFT descriptor for each point. Note, this is a novel feature descriptor. It captures the colour gradients (or edges) of the image. Other ways of using colour with SIFT features have been proposed by [12], [34].

Grey SIFT (sparse). Affine co-variant regions are computed for each grey scale image, constructed by elliptical shape adaptation about an interest point [22]. These regions are represented by ellipses. Each ellipse is mapped to a circle by appropriate scaling along its principal axis and a 128-dim SIFT descriptor computed. This is the method used by [7], [17], [27], [28].

A. Implementation details

Dense SIFT descriptors

In most previous applications SIFT like descriptors are used following a sparse feature detection, and so have only been applied at image points where there is sufficient structure (e.g. a strong response from a Harris or Hessian operator). In our case the SIFT descriptors are applied densely, perhaps at every pixel, and this raises two areas of concern.

First, in regions with near constant colour/brightness (like sky, road) that consequently have small image gradients, is the resulting description (the visual words) very sensitive to noise? In practice we find that the assigned word for such patches is often the same and relatively insensitive to patch size. For example if sky patches with $r = 4$ are assigned the word w_1, then sky patches with $r = 8$ are also assigned the word w_1 and so on. Where the small gradients (noise) do result in different random visual word assignments, then the pLSA topic learns this distribution.

Second, is there a problem with noise causing wrap-around in the H colour channel? This could occur with a region consisting of small fluctuations around saturated red, and would result in an alternation of visual word assignment over that region. However, in practice we do not observe this problem in the current databases.

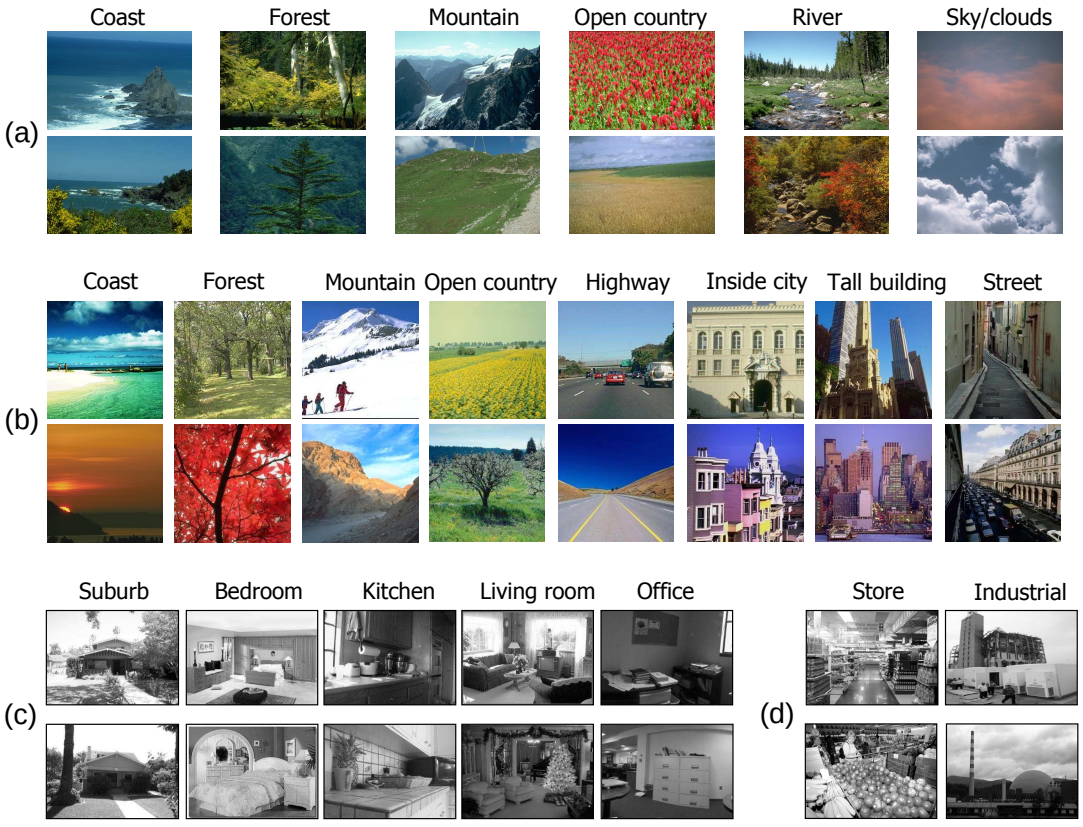


Fig. 3. Example images from the four different datasets used. (a) from dataset OT [24]; (b) from dataset VS [32]; (c) from the dataset FP [9], the remaining images of this dataset are the same as in OT but in greyscale; and (d) from dataset LSP [18], same scenes as in FP plus store and industrial.

Normalization

Grey level images are normalized to have intensities with mean zero and unit standard deviation. Colour images are first normalized as in "Gray World" [6], [11] to have R,G and B components $R*(\mu/\mu_r), G*(\mu/\mu_g), B*(\mu/\mu_b)$ where $\mu = (\mu_r + \mu_g + \mu_b)/3$ and μ_r, μ_g, μ_b are the mean of each component. The HSV is then computed from these normalized values.

V. Datasets and Methodology

A. Datasets

We evaluated our classification algorithm on four different datasets: (i) Oliva and Torralba [24], (ii) Vogel and Schiele [32], (iii) Fei-Fei and Perona [9], and (iv) Lazebnik et *al* [18]. We will

refer to these datasets as OT, VS, FP and LSP respectively. Figure 3 shows example images from each dataset, and the contents are summarized here:

OT: includes 2688 images classified as 8 categories: 360 coasts, 328 forest, 374 mountain, 410 open country, 260 highway, 308 inside of cities, 356 tall buildings, 292 streets. The average size of each image is 250×250 pixels.

VS: includes 702 natural scenes consisting of 6 categories: 144 coasts, 103 forests, 179 mountains, 131 open country, 111 river and 34 sky/clouds. The size of the images is 720×480 (landscape format) or 480×720 (portrait format). Every scene category is characterized by a high degree of diversity and potential ambiguities since it depends strongly on the subjective perception of the viewer.

FP: contains 13 categories and is only available in greyscale. This dataset consists of the 2688 images (8 categories) of the OT dataset plus: 241 suburb residence, 174 bedroom, 151 kitchen, 289 living room and 216 office. The average size of each image is approximately 250×300 pixels.

LSP: contains 15 categories and, as with FP, is only available in greyscale. This dataset consists of the 13 categories of the FP dataset plus: 315 store and 311 industrial. The average size of each image is approximately 250×300 pixels.

B. Methodology

The classification task is to assign each test image to one of a number of categories. The performance is measured using a confusion table, and overall performance rates are measured by the average value of the diagonal entries of the confusion table.

Datasets are split randomly into two separate sets of images, half for training and half for testing. From the training set we randomly select 100 images to form a validation set. This validation set is used to find the optimal parameters, and the rest of the training images are used to compute the vocabulary and pLSA topics. A vocabulary of visual words is learnt from about 30 random training images of each category.

Excluding the preprocessing time of feature detection and visual vocabulary formation, it takes about 20 mins to fit the pLSA model to 1600 images (Matlab implementation on a 1.7GHz computer).

The new classification scheme is compared to two baseline methods. These are included in order to gauge the difficulty of the various classification tasks. The baseline algorithms are:

Global colour model. The algorithm computes global HSV histograms for each training image. The colour values are represented by a histogram with 36 bins for *H*, 32 bins for *S*, and 16 bins for *V*, giving a 84-dimensional vector for each image. A test image is classified using KNN (with $K = 10$).

Global texture model. The algorithm computes the orientation of the gradient at each pixel for each HSV channel at each training image. These orientations are collected into a 72 bin histogram for each colour channel and concatenated to form a histogram of 72×3 bins for each image. The classification of a test image is again carried out using KNN.

VI. Classification results

In this section we carry out a set of experiments to investigate the various choices of vocabularies, parameters and classifiers, and also to assess the benefits or otherwise of using pLSA as an intermediate representation.

The experiments in this section are all on the OT dataset. The results for the other datasets (FP, VS and LSP) are given in Section VIII. For the OT dataset three classification situations are considered: classification into 8 categories, and also classification within the two subsets of natural (4 categories), and man-made (4 categories) images. The latter two are the situations considered in [24].

We start by finding the optimal parameters (V, Z and K) over the validation set for each of the different vocabularies described in Section VI-A. The optimal parameters are then fixed, and subsequent results reported on the test set in Section VI-B.

A. Optimizing the parameters V, Z and K (on the validation set)

We first investigate how classification performance (on the validation set) is affected by the various parameters: the number of visual words (V in the k-means vector quantization), the number of topics (Z in pLSA), and the number of neighbours (K in kNN). Figure 4 shows this performance variation for two types of descriptor – dense colour SIFT with $M = 10$ and four circular supports, and grey patches with $N = 5$ and $M = 3$. Note the mode in the graphs of V, Z and K in both cases. This is quite typical across all types of visual words, though the

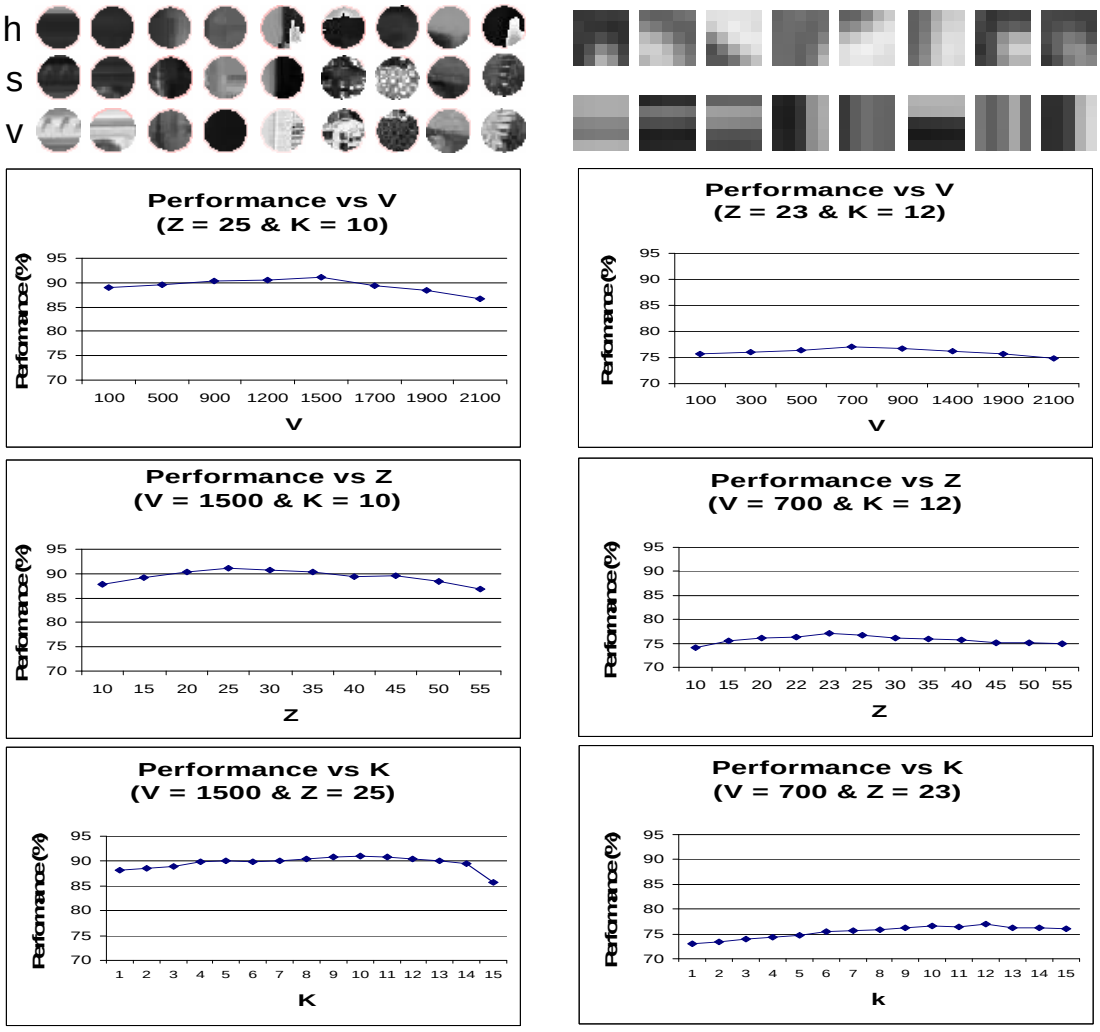


Fig. 4. Validation set performance under variation in various parameters for the 8 category OT classification. Left: example visual words and performance for dense colour SIFT $M = 10$, $r = 4$, 8, 12 and 16 (each column shows the HSV components of the same word). Right example visual words and performance for grey patches with $N = 5$ and $M = 3$. Top graph: varying number of visual words, V, Middle graph: varying number of topics, Z, Bottom graph: varying k (KNN).

position of the modes vary slightly. For example, using colour SIFT the mode is at $V = 1500$ and $Z = 25$, while for grey patches the mode is at $V = 700$ and $Z = 23$. For K the performance increases progressively until K is between 7 and 12, and then drops off slightly.

For colour patches the best performance is obtained when using the 5×5 patch over normalized images, with M = 3, V = 900, Z = 23 and K = 10. The best results overall are obtained with dense colour sift with 4 circular supports, $M = 10$, normalized images, V = 1500, Z = 25 and

K = 10. We will see in next section that this vocabulary is also the one which gives the best results on the test set.

To investigate the statistical variation we repeat the dense colour SIFT experiment (r = 4, 8, 12, 16 and M = 10) 15 times with varying random selection of the training, validation and test sets, and building the visual vocabulary afresh each time. All parameters are fixed with the number of visual words V = 1500, the number of topics Z = 25 and the number of neighbours K = 10. We obtained performance values between 79% and 86% with a mean of 84.7% and standard deviation of 1.9%.

B. Comparison of features and support regions (on the test set)

We next investigate the patch descriptors in more detail. Again, we use the OT dataset with 8 categories and the KNN classifier for this task (the SVM classifier is investigated in Section VI-C). In the following results the optimum choice of parameters determined on the validation set is used for each descriptor type, but here applied to the test set. Figure 5a shows the results when classifying the images of natural scenes with colour-patches. The performance when using normalized images is nearly 1% better than when using unnormalized. When using overlapping patches, the performance increases by almost 6% compared to no overlap. Similar results occur for the man-made and all scene category sets. Comparing results when classifying the images using only grey level information or using colour, it can be seen in figure 5b and table II, that colour brings an increment of around 2%. This is probably because colour is such an important factor in outdoor images, and helps to disambiguate and classify the different objects in the scene.

The performance of SIFT features is shown in Figure 5b. The best results are obtained with dense and not sparse descriptors. This is almost certainly because we have more information on the images: in the sparse case the only information is where a Harris detector fires and, especially for natural images, this is a very impoverished representation. Again colour is a benefit with better results obtained using colour than grey SIFT. The performance using grey SIFT when classifying natural images is 88.5% and increase 2% when using colour SIFT, both with four concentric support regions. The difference when using these vocabularies with man-made images is not as significant. This reiterates that colour in natural images is very important for classification.

Number of support regions. Turning to the performance variation with the number of support

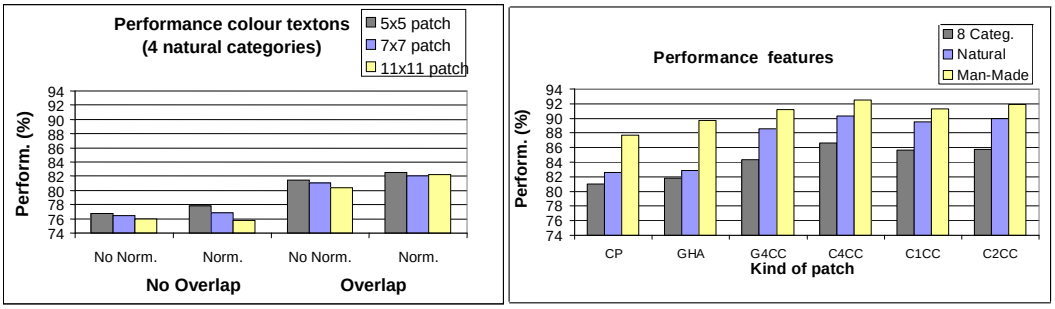


Fig. 5. (a) The OT test set performance when classifying the four natural categories using normalized and unnormalized images and with overlapping and non-overlapping patches. Colour patches are used. (b) Performance when classifying all categories, man-made and natural using different patches and features. Abbreviations for this and subsequent figures: CP (Colour Patches), GHA (Grey Harris Affine – sparse, all the other descriptors are dense), G4CC (Grey SIFT four Concentric Circles), CnCC (Colour SIFT with n Concentric Circles).

regions for dense SIFT. It can be seen from Figure 5b that best results are obtained using four concentric circles. With only one support region to represent each patch, results are around 1% worse. This is probably because of lack of invariance to scale changes: using four support regions to represent each pixel effectively represents the texture at four different scales.

We now investigate how important it is to use four concentric circles to represent each pixel in both training *and* testing. The first row of Table I shows the performance when using four concentric circles with colour to represent each pixel at the training stage, and four, two and one circles also with colour information for the testing data. The second row shows the performances when using the same number of circles to represent the pixels at the training and testing stage. It can be seen that performances in the first row are very similar, so that four concentric circles is enough to represent the training data and fewer patches can be used to represent the pixels in the testing images, i.e. sampling only the training images at multiple scales is sufficient.

Table II summarizes the results for the three OT image sets (all 8 categories, 4 natural and 4 man-made) covering the different dense vocabularies: grey and colour patches, grey and colour SIFT and the two baseline algorithms when using KNN classifier. From these results it can be seen that: (i) The baseline texture algorithm works better than the baseline colour in all three cases. Despite its simplicity the performance of the baseline texture algorithm on man-made images (73.8%) is very high, showing that these images may be easily classified from their edge

TABLE I

TEST SET PERFORMANCE WHEN CHANGING THE NUMBER OF TRAINING AND TEST SUPPORT REGIONS (FOR OT 8 CATEGORIES). FIRST ROW: EACH PIXEL IN THE TRAINING IMAGES ARE REPRESENTED BY FOUR CIRCLES (4CC) AND THE TESTING IMAGES ARE REPRESENTED BY FOUR (4CC), TWO (2CC) AND ONE (1CC) CIRCLE FROM LEFT TO RIGHT. SECOND ROW: PIXELS IN THE TRAINING AND TESTING IMAGES ARE REPRESENTED BY THE SAME NUMBER OF CIRCLES. THE COLOUR SIFT DESCRIPTOR IS USED.

Training Regions	Testing Regions		
	4CC	2CC	1CC
4CC	**86.9**	86.7	86.3
Same as Testing	**86.9**	85.8	85.7

TABLE II

TEST SET PERFORMANCE FOR DIFFERENT FEATURES WHEN USING THE OT DATABASE. GP (GRAY PATCHES), CP (COLOUR PATCHES), G4CC (GREY SIFT FOUR CONCENTRIC CIRCLES), C4CC (COLOUR SIFT WITH FOUR CONCENTRIC CIRCLES). PS (COLOUR PATCHES AND COLOUR SIFT), GLC (GLOBAL COLOUR), GLT (GLOBAL TEXTURE).

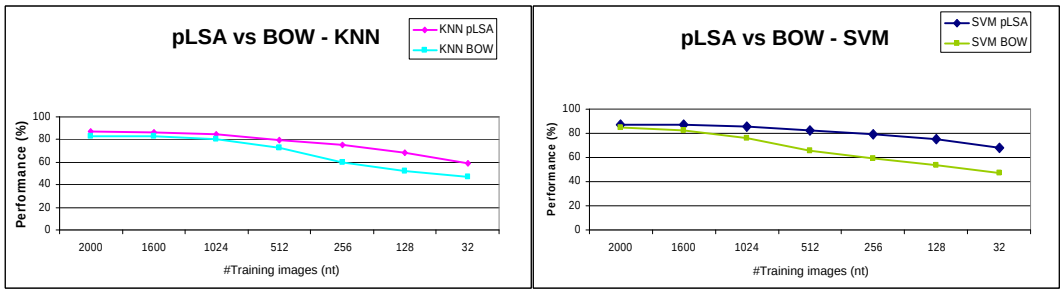
Visual Vocabulary	GP	CP	G4CC	C4CC	PS	GlC	GlT
All categ.	71.5	77.0	84.3	**86.6**	82.6	55.1	64.6
Natural categ.	75.4	82.4	88.5	**90.2**	84.0	59.5	70.1
Man-made categ.	77.4	83.5	91.1	**92.5**	89.3	66.1	73.8

directions. (ii) For the various descriptors there are clear performance conclusions: man-made is always better classified than natural (as expected from the baseline results); SIFT type descriptors are always superior to patches; colour is always superior to grey level. The best performance (86.6% for all 8 categories) is obtained using colour SIFT and four concentric circles. (iii) Somewhat surprizingly, better results are obtained using the SIFT vocabulary alone, rather than when merging both vocabularies (patches and SIFT). This may be because the parameters (V, Z and K) have been optimized for a single vocabulary, not under the conditions of using multiple vocabularies.

TABLE III

PERFORMANCE OBTAINED FOR KNN AND SVM USING PLSA OR BOW VECTORS FOR THE CLASSIFIERS. OT DATABASE (8 CATEGORIES) IS USED. G4CC (GREY SIFT FOUR CONCENTRIC CIRCLES), C4CC (COLOUR SIFT WITH FOUR CONCENTRIC CIRCLES)

	pLSA		BoW	
	KNN	SVM	KNN	SVM
C4CC	86.6	**87.1**	82.5	83.8
G4CC	84.3	84.7	79.7	80.8



(a) (b)

Fig. 6. pLSA and BoW performances when decreasing the number of training images. 8 categories from the OT dataset with four concentric circles and $V = 1500$ words, $Z = 25$ and $K = 10$.

C. KNN vs SVM

All the results above are for $P(z|d)$ with the KNN classifier. Now we investigate classification performance when using a SVM. Table III shows the results for the SIFT support regions for both classifiers KNN and SVM. Optimized parameters for each vocabulary are used. It can be seen that SVM performs around 1% better than KNN.

D. pLSA vs Bag-of-Words (BoW)

The results to this point use pLSA to obtain an intermediate representation, with $P(z|d)$ as the inputs for the classifiers. We now compare to the performance obtained by classifying the BoW representation directly. Again the performance is for the OT dataset with 8 categories, and in all the experiments: $V = 1500$ (unless stated otherwise), $Z = 25$, $K = 10$, and four support regions are used for each point spaced at $M = 10$. For the SVM classifier a χ^2 exponential

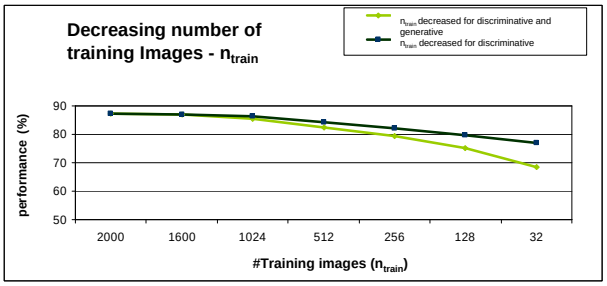


Fig. 7. Performance when decreasing the number of training images in the generative and discriminative parts (blue line) and when deceasing the number of labelled training images only for the discriminative part (yellow line). 8 categories from the OT dataset with four concentric circles and $V = 1500$ words, $Z = 25$ and $K = 10$. A SVM is used as the discriminative classifier.

kernel [36] is used for the BoW, and an Euclidean exponential kernel for pLSA. These kernels were found to give the best performance in each case.

Table III shows pLSA and BoW rates for different support regions and using a SVM and KNN. It can be seen that in all cases the performance using pLSA is around 4% better than that obtained using a BoW.

Number of training Images: We now evaluate the classification performance when less training data is available. The OT dataset is split into 2000 training images and 688 test images. A varying number, n_{train}, of images from the training set are used for both learning the pLSA topics (generative part) and learning the topic distribution of each scene (discriminative part). The classification performance using $P(z|d)$ is compared to that of using BoW vectors. As can be seen in Figure 6, the gap between pLSA and BoW increases as the number of labelled training images decreases, as was demonstrated in [25].

In the previous experiment, we varied the amount of training data for both: the generative and discriminative learning. However, a key advantage of the hybrid approach is that the generative part of the model can be trained on large amounts of unlabelled data (hence discovering the structure of the data), so that relatively few labelled examples are needed for high accuracy. To show this advantage, we repeat the previous experiment training the generative classifier using the 2000 training images and decreasing the number of labelled training images (n_{train}) only for the discriminative classifier. Figure 7 shows the comparison of the previous experiment and the current experiment when using SVM as a discriminative classifier. It can be seen that much better results are obtained when decreasing only the number of labelled training data than when

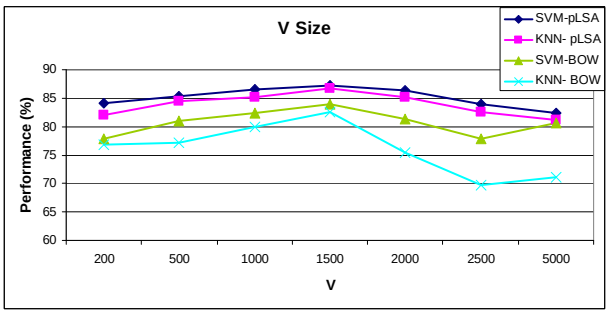
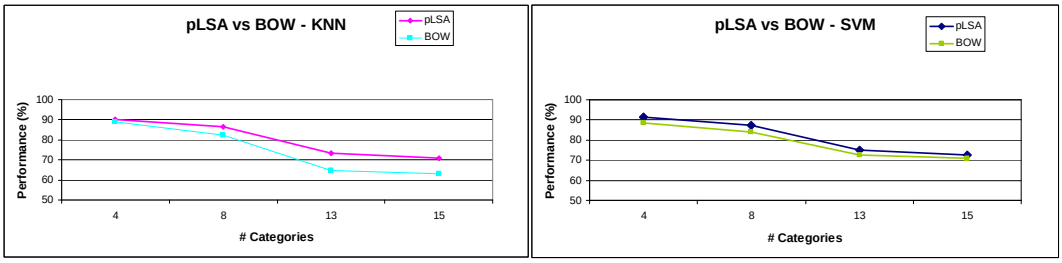


Fig. 8. Changing the vocabulary size for the OT dataset. Parameters are $Z = 25$, $K = 10$, $M = 10$ and four concentric circles.



(a) (b)

Fig. 9. pLSA and BoW performances when classifying different number of categories (from 4 to 15). Parameters used are V = 1500, Z = 25, $M = 10$ and 4 concentric circles as support regions. Top: pLSA vs BoW when using KNN ($K = 10$), Bottom: pLSA vs BoW when using SVM.

reducing the training data in both learning parts. So there is a clear advantage of using a hybrid approach: the system has acceptable performances with less labelled training data.

Vocabulary Size: Figure 8 shows the performance when changing the vocabulary size V (from 200 to 5000 words) for both the discriminative classifiers (KNN and SVM). It can be seen that for both classifiers, pLSA is less affected by the vocabulary size than the BoW.

Number of scene categories: Figure 9 shows the performances when increasing the number of categories to be classified for both KNN (Figure 9a) and SVM (Figure 9b). For the KNN, when classifying the 4 natural images in the OT dataset, the results using the topic distribution is 90.2% and with the BoW directly the classification performance decreases by only around 1.5%, to 88.7%. However for 8 categories, the performance decreases by nearly 4%, from 86.6% to 82.5%. Using the 13 categories from the FP dataset and the 15 LSP dataset, the performance

TABLE IV

CLASSIFICATION RATES FOR PLSA AND BOW WHEN CLASSIFYING CATEGORIES FROM DIFFERENT DATASETS. PARAMETERS USED ARE V = 1500, Z = 25, $M = 10$ AND 4 CONCENTRIC CIRCLES AS SUPPORT REGIONS.

	KNN		SVM	
# Categ.	pLSA	BoW	pLSA	BoW
4 OT dataset	90.2	88.7	**91.5**	88.4
8 OT dataset	86.6	82.5	**87.1**	83.8
13 FP dataset	73.4	64.8	**74.9**	73.6
15 LSP dataset	71.0	63.1	**72.6**	72.5

TABLE V

OPTIMIZED PARAMETERS WHEN USING THE SIFT VOCABULARY FOR THE FOUR DATASETS: $M = 10$ AND $r = 4$, 8, 12 AND 16 PIXELS, AND WHEN USING THE PATCH VOCABULARY: $N = 5$, $M = 3$ PIXELS. A VALIDATION SET IS USED FOR EACH DATASET.

	SIFT			Patch		
Dataset	V	Z	K	V	Z	K
VS	1500	25	7	900	25	9
OT	1500	25	10	900	23	10
FP	1200	35	9	600	33	10
LSP	1200	40	11	700	42	12

falls around 8%, from 73.4% to 64.8% and from 71.0% to 63.1% respectively. Thus there is a clear gain in using pLSA (over the BoW) with KNN when classifying a large number of categories.

If we focus on the SVM, performances with pLSA are better as well. However when classifying a large number of categories (13 or 15) pLSA is 1% better than BoW, thus the gap is not as large as when using the KNN classifier. Table IV summarizes the performances for KNN and SVM over pLSA and BoW.

E. Summary

The best results are obtained using dense descriptors – colour SIFT with four circular support regions. Overlap increases the performance. When using the SIFT vocabulary the values for the

parameters giving the best results are $M = 10$ pixels with concentric circles support regions of $r = 4$, 8, 12 and 16 pixels. For patches the best results are for $N = 5$, $M = 3$. Table V shows the optimized values V, Z and K learnt from a validation set for each dataset. Note that V strongly depends on the size of the feature vector (128×3 dimensionality vector for SIFT and 25×3 dimensionality vector for patches), while Z depends on the number of categories in each dataset. In both (SIFT and patches), colour information increases performance. The result that dense SIFT gives the best performance was also found by [8] in the case of pedestrian detection. It it interesting that the same feature applies both to more distributed categories (like grass, mountains) as well as the compact objects (pedestrians) of their work where essentially only the boundaries are salient.

When comparing the discriminative classifiers KNN and SVM, better performances are obtained with SVM. We also demonstrated that pLSA works better than the BoW representation (pLSA provides a better intermediate representation of the images), and that pLSA is less affected by the vocabulary size and the number of training images. More concretely for the KNN discriminative classifier, when working with a small number of categories the difference between pLSA and BoW is 1.5%. However when the number of categories increases this difference is around 8% showing that pLSA provides a more robust intermediate representation than BoW. Thus there is a clear gain in using pLSA (over the BoW) with KNN when classifying a large number of categories. Moreover a clear advantage of using a generative model (pLSA), over BoW directly, is that the number of *labelled* training images can be reduced considerably without much loss of performance.

VII. Spatial Information

Recently it has been shown [2], [10], [18] that position information can improve scene classification performance (earlier work had shown little benefit [32]). Motivated by this, we add position information into our pLSA framework. We have implemented and compared four methods described below. For these results the colour SIFT vocabulary is used with four concentric circles spaced at $M = 10$, and the SVM is used as the discriminative classifier. The OT dataset with optimized values (see Table V) is used to evaluate performance.

xy-pLSA. The x and y normalized position of each pixel is concatenated to the feature vector. So now the dimension of the feature vector is $N^2 \times 3 + 2$. Each component of the feature vector

(both spatial and SIFT) is in the range $[0, 1]$. However, the SIFT part of the vector is sparse in general.

ABS-pLSA. This is the method proposed in [10]. The pLSA model is extended to incorporate location information by quantizing the location within the image into one of X bins. The joint density on the appearance and location of each region is then represented. Thus $P(w|z)$ in pLSA becomes $P(w, x|z)$, a discrete density of size $(W \times X) \times Z$. The same pLSA update equations outlined in Section II can be easily applied to this model in learning and recognition. The method is evaluated for X = 1, 4 and 16 bins, with the case X = 1 corresponding to standard pLSA with no spatial information.

Spatial Pyramid Matching (SPM). This is the method proposed by Lazebnik et *al*. [18] which is based on spatial pyramid matching [13]. Pyramid matching works by placing a sequence of increasingly coarser grids over the feature space (in this case over the image) and taking a weighted sum of the number of matches that occur at each level of resolution (L). At any fixed resolution, two points are said to match if they fall into the same bin of the grid; matches found at finer resolutions are weighted more highly than matches found at coarser resolutions (α_l represents the weight at level l). The resulting spatial pyramid is an extension of the BoW image representation, it reduces to a standard BoW when $L = 0$, and a level 1 grid is equivalent to X = 4 in the ABS-pLSA model.

Spatial Pyramid – pLSA (SP-pLSA). This method is inspired by both the previous ones, ABS-pLSA and SPM. We incorporate location information in pLSA by using the X bins at each resolution level L, weighting the bins for each level (α_l) as in SPM. Note that in ABS-pLSA only the bins for one resolution level are used and in SP-pLSA we use the weighted bins for L resolutions. So for example when $L = 1$, if using ABS-pLSA we have $X = 4$ bins, and if we use SP-pLSA we have $X = 5$ bins (one bin for $L = 0$ and four bins for $L = 1$). Thus $P(w|z)$ in pLSA becomes $P(w, x, l|z)$. The same pLSA update equations outlined in Section II can be easily applied to this model in learning and recognition.

Table VI shows the values for pLSA without position and the four methods described above. The weights used in these experiments are: $\alpha_0 = 0.25$, $\alpha_1 = 0.25$ and $\alpha_2 = 0.5$ (the same weights are used in [18]). When only the first level of the pyramid is used ($L = 0$) the best result (89.0%) is obtained when using xy-pLSA. In this case SPM works directly over the BoW and has worse results than the methods that use pLSA. When $L = 1$ and $L = 2$ the best results

TABLE VI

PERFORMANCE COMPARISON FOR THE OT DATASET WHEN SPATIAL INFORMATION IS USED. FOUR CONCENTRIC CIRCLES SPACED AT $M = 10$ AND $V = 1500$, $Z = 25$.

Pyramid level	pLSA	xy-pLSA	ABS-pLSA	SPM	SP-pLSA
L = 0	87.1	89.0	$87.1(X = 1)$	83.8	87.1
L = 1	–	–	$87.9(X = 4)$	90.3	90.7
L = 2	–	–	$88.3(X = 16)$	91.0	**91.1**

are obtained for SP-PLSA (90.7% and 91.1%) followed by SPM (90.3% and 91.0%). We only explored up to L=2 which was demonstrated in [18] to be the optimum level.

A. Weight optimization - α_0, α_1 and α_2 on the validation set

Using the validation set (see Section V-B) we optimize the ratio between the weights α_0:α_2 and α_1:α_2 over the range $[0, 1.5]$. Figure 10a shows the test performance when optimizing using SPM on the OT dataset (8 categories). In this case the best performance (92.2% for the test data) is for $\alpha_0 : \alpha_2 = 1$ and $\alpha_1 : \alpha_2 = 0.9$ (weights obtained from the validation set). This performance exceeds that given in Table VI.

Figure 10b shows performances when optimizing the weights for SP-pLSA on the OT dataset (8 categories). Now the performance increases to 92.7% for the test data using the validation set optimized ratios $\alpha_0 : \alpha_2 = 0.7$ and $\alpha_1 : \alpha_2 = 0.8$. Note that best performances are obtained for higher ratios than in the experiments of Table VI (where the ratio was $\alpha_0 : \alpha_2 = 0.5$ and $\alpha_1 : \alpha_2 = 0.5$) which are the same ratios used in [18]. The optimized ratios using the validation set for all datasets used are summarized in Table VII. Default values of $\alpha_0 : \alpha_2 = 0.9$ and $\alpha_1 : \alpha_2 = 0.8$ clearly give superior performance than those of [18].

VIII. COMPARISON TO PREVIOUS RESULTS

A. Scene classification

We compare the performance of our scene classification algorithm to the supervised approaches of Vogel and Schiele [32] and Oliva and Torralba [24], and the semi-supervised approach of Fei-Fei and Perona [9] and Lazebnik et *al*. [18], using the same datasets that they tested their approaches on and the same number of training and testing images. For each dataset we use the

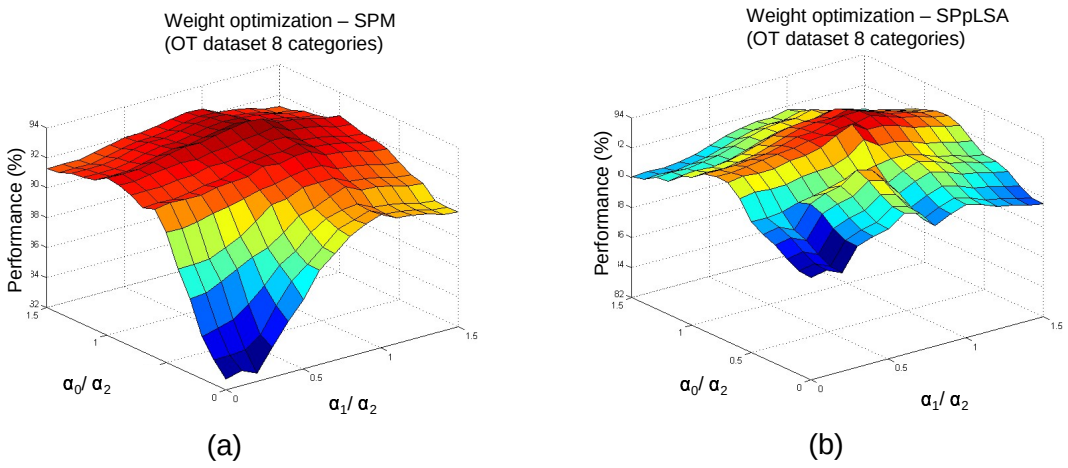


Fig. 10. Optimization rates between the weights at each pyramid level using the validation set from the OT dataset: (a) SPM is used; (b) SP-pLSA is used.

TABLE VII

OPTIMIZED WEIGHT RATIOS $\alpha_0 : \alpha_2$ AND $\alpha_1 : \alpha_2$ FOR EACH DATASET USING THE VALIDATION SET. $4N$ = 4 NATURAL CATEGORIES; $4MM$ = 4 MAN-MADE CATEGORIES.

ratios	OT - 8	OT - $4N$	OT - $4MM$	VS	FP	LSP
SPM weights						
$\alpha_0 : \alpha_2$	1	0.8	0.9	1	0.9	1
$\alpha_1 : \alpha_2$	0.9	0.9	0.8	0.9	0.8	0.8
SP-pLSA weights						
$\alpha_0 : \alpha_2$	0.7	0.9	0.9	0.9	1	1
$\alpha_1 : \alpha_2$	0.8	0.8	0.8	0.7	0.9	0.8

SVM classifier, SIFT and four circular supports spaced at $M = 10$; the parameters V,Z,α_0, α_1 and α_2 have the validation set optimised values for each dataset (see Table V and Table VII). We used colour for OT and VS and grey for FP and LSP. The visual vocabulary is computed independently for each dataset, as described in Section V-B. We return to the issue of sharing vocabularies across datasets in Section IX. The results are given in Table VIII.

Note that much better results are obtained with the four natural scenes of OT, than with the six of VS. This is because the images in VS are much more ambiguous than those of OT and consequently more difficult to classify. Without using spatial information (5th column in

TABLE VIII

COMPARISON OF OUR ALGORITHM WITH OTHER METHODS USING THEIR OWN DATABASES. VALIDATION SET OPTIMIZED VALUES ARE USED FOR EACH DATASET. $L = 2$ FOR SP-PLSA AND SPM.

Dataset	# of categ.	# train	# test	pLSA	SP-pLSA	SPM	Authors
OT	8	800	1888	82.5	**87.8**	87.1	83.7 [24]
OT	4 Natural	1000	472	90.7	**93.9**	93.3	89.0 [24]
OT	4 Man-Made	1000	216	91.7	**94.8**	94.2	89.0 [24]
VS	6	600	100	87.8	88.3	**88.6**	74.1 [32]
FP	13	1300	2459	74.3	**85.9**	85.5	65.2 [9]
LSP	15	1500	2986	72.7	**83.7**	83.5	81.4 [18]

TABLE IX

CLASSIFICATION OF CALTECH 101 WITH 15 OR 30 TRAINING IMAGES PER CLASS, AND 50 TEST IMAGES PER CLASS. FOR SP-PLSA AND SPM FOUR CONCENTRIC CIRCLES SPACED AT $M = 10$ ARE USED, $V = 1500$, $Z = 80$, AND SVM IS USED AS THE DISCRIMINATIVE CLASSIFIER.

# train	SP-pLSA	SPM	[18]	[1]	[14]	[23]	[33]	[35]
15	**59.8**($\pm$1.4)	58.7($\pm$0.8)	56.4	52.0	49.5	51.9	44.0	59.0
30	**67.7**($\pm$1.5)	66.5($\pm$0.7)	64.6	–	58.2	56.0	63.0	66.0

Table VIII) our method outperforms the previous methods in [24], [9], [18], [32], despite the fact that our training is unsupervised in the sense that the scene identity of each image is unknown at the pLSA stage and is not required until the KNN or SVM training step. This is in contrast to [9], [18], where each image is labelled with the identity of the scene to which it belongs during the training stage. In [32], the training requires manual annotation of 9 semantic concepts for 60000 patches, while in [24] training requires manual annotation of 6 properties for thousands of scenes. It is worth noting that in [24], [32] the intermediate information which represents the images has a semantic meaning, while in [9], [18] and our approach the intermediate information need not have a semantic meaning from the human point of view. However this is not a problem: we are interested in the semantic meaning of the whole scene, and not the intermediate information, because our final goal is to give a label for each scene.

As we noted in Section VII better results are obtained with spatial information (6th and 7th columns in Table VIII). We have better performances than [18] when using SP-pLSA and also

when using their own method with our features. Moreover, for a better comparison we use the same number of words and weight ratios as in [18] ($V = 200$ and $L = 2$): they achieve 81.1% of correct classified scenes, and we increase this to 82.2% (with SPM) when using four concentric circles to represent each pixel in the image. In both our and their experiments grey SIFT descriptors are used. This demonstrates again that using more than one patch to represent each pixel increases performance.

Discussion

The superior performance (compared to [9], [32]) could be due to the use of better features and how they are used. In the case of Vogel and Schiele [32], they learn 9 topics (called *semantic concepts*) that correspond to those that humans can observe in the images: *water*, *trees*, *sky* etc. for 6 categories. In our case, we discover between 22 and 30 topics in the case of 8 categories. These topics can vary depending if we are working with colour features (where topics can distinguish objects with different colours like *light sky*, *blue sky*, *orange sky*, *orange foliage*, *green foliage* etc...) or only grey SIFT features (objects like *trees* and *foliage*, *sea*, *buildings* etc...). In contrast to [32] we discover objects that sometimes would not be distinguished in a manual annotation, for example *mountains with snow* and *mountains without snow*. Fei-Fei and Perona learn 40 topics (called *themes*) for 13 categories, but it is left unsaid whether these topics correspond to natural objects.

Our superior performance compared to [24] could be due to their method of scene interpretation. They propose a set of perceptual dimensions (e.g. naturalness, openness) that represent the dominant spatial structure of a scene. These dimensions are estimated using spectral and coarsely localized information, using a very low dimensional representation of the scene (Spatial Envelope) which bypasses the segmentation and the preprocessing of individual objects or regions. In contrast, in our approach specific information about objects/topics is used for scene categorization. We also outperform the (SPM) classifier proposed in [18] when working with our pixel representation and features. So we have demonstrated that representing a pixel with more than one patch is better. Moreover we successfully incorporated spatial information into the pLSA framework (SP-pLSA) obtaining slightly better performances than SPM, though optimizing the level weights is responsible for the more significant part of the improvement.

B. Caltech 101

The Caltech-101 data set (collected by Fei-Fei et *al*. [20]) consists of images from 101 object categories and an additional background class, making the total number of classes 102. This database contains from 31 to 800 images per category. Most images are medium resolution, about 300×300 pixels. The significance of this database is its large inter-class variability. A number of previously published papers have reported results on this data set: Lazebnik et *al*. [18], Berg et *al*. [1], Grauman and Darrell [14], Zhang et *al*. [35] etc.

For the experiments, dense colour SIFT with four support regions are used to represent each pixel, spaced at $M = 10$, $V = 1500$, and $Z = 80$ topics. The weight ratios are $\alpha_0/\alpha_2 = 0.9$ and $\alpha_1/\alpha_2 = 0.8$ for SP-pLSA and $\alpha_0/\alpha_2 = 1$ and $\alpha_1/\alpha_2 = 0.9$ for SPM. An SVM is used as the classifier. We carried out experiments using 15 and 30 random training images per category, and 50 random testing images per class (disjoint from the training images). The mean recognition rate per class is used so that more populous (and easier) classes are not favoured. This process is repeated 10 times and the average correctness rate is reported. Table IX shows our results and those reported by other authors. Our best performance is when using the SP-pLSA algorithm with a mean recognition rate of 59.8% with 15 training images per class, and 67.7% with 30 training images per class. This outperforms the results reported by Zhang et *al*. [35] that to our knowledge are the best until now.

IX. Applications

We applied the pLSA based classifier in four other situations. The first one is also a classification task, but combining the images of two different datasets, the second is a relevance feedback application, the third is scene retrieval for the film *Pretty Woman* [Marshall, 1990], and in the fourth we apply pLSA for image segmentation. In all the following the descriptor is dense colour SIFT with circular support and $V = 700$, $Z = 22$ and $K = 10$ (these are the optimal parameter values when working with the four natural scenes from the OT dataset).

Vocabulary generalization. In this classification test, we train the system with the four natural scenes of the OT dataset (*coast*, *forest*, *mountains* and *open country*) and test using the same four scene categories from the VS dataset. This tests whether the vocabulary and categories learnt from one dataset generalize to another. We obtain a performance of 88.2% of correctly classified images for KNN and 88.9% for SVM. This performance is only slightly worse than the 89.8%

Fig. 11. Example frames from the film Pretty Woman with their classification. The classifier is trained on the OT dataset.

obtained when classifying the same four categories in the VS dataset with no generalization (i.e. using training images only from VS). This slight performance drop is because (i) images within the same database are more similar, and (ii) the images in VS are more ambiguous than OT, so this ambiguity is not represented in training the OT classifier. However, 88.9% compared to 89.8% does demonstrate excellent generalization. To address (i) we investigate using a vocabulary composed from both databases and find this improves the performance to 89.6%.

Relevance Feedback (RF). [37] proposed a method for improving the retrieval performance, given a probabilistic model. It is based on moving the query point in the visual word space towards good example points (relevant images) and away from bad example points (irrelevant images). The vector moving strategy uses the Rocchio's formula [26]:

$$q_{pos} = \alpha q + \beta(\frac{1}{a}\sum_{i=1}^{a} rel_i) - \gamma(\frac{1}{b}\sum_{j=1}^{b} irel_j) \tag{3}$$

where q is the BoW for the query image, a is the number of relevant images b is the number of irrelevant images, and rel, $irel$ are the BoW representations for the relevant and irrelevant retrieved images. The parameters α, β and γ are set to 1. With the modified query vector q_{pos} and a constructed negative example q_{neg}:

$$q_{neg} = \alpha(\sum_{j=1}^{b} irel_j) + \beta(\frac{1}{b}\sum_{j=1}^{b} irel_j) - \gamma(\frac{1}{a}\sum_{i=1}^{a} rel_i) \tag{4}$$

their representations in the discovered concept space are obtained $P(z|q_{pos})$ and $P(z|q_{neg})$ and their similarities sp_i and sn_i to each image $i \epsilon I$ in the database are measured using the cosine

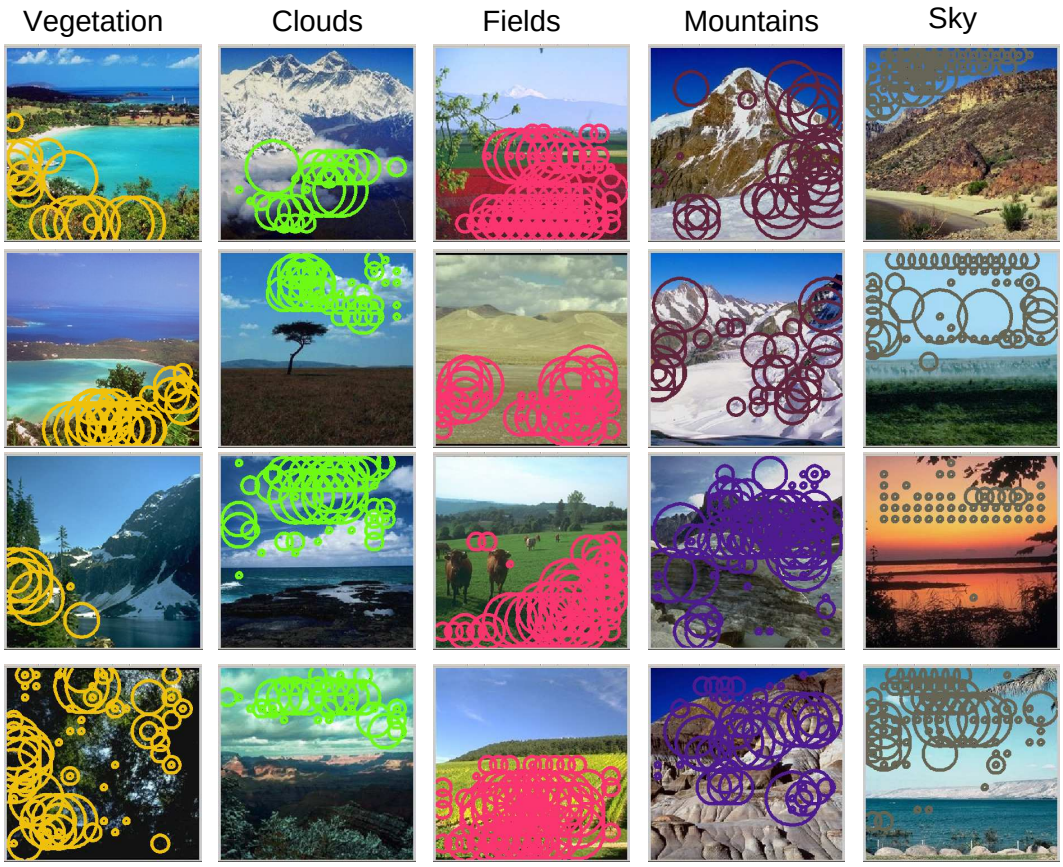


Fig. 12. Topics segmentation. Five topics (vegetation, clouds, fields, mountains and sky) are shown. Only circular regions with a topic posterior $P(z|w,d)$ greater than 0.8 are shown.

metric of the corresponding vectors in the topic space, respectively. Then the images are ranked based on the similarity $s_i = sp_i - sn_i$.

To test RF we simulate the user's feedback using 25 random images of each category. For each query image, we carry out *n* iterations. At each iteration the system examines the top 20, 40 or 60 images that are most similar to the query excluding the positive examples labelled in previous iterations. Images from the same category as the initial query will be used as positive examples, and other images as negative examples. We used 200 query images, 25 of each category, in the OT dataset. Best results are obtained when considering the top 60 images, The first 100 images can be retrieved with an average precision of 0.75. The most difficult category to retrieve is *open country* while the better retrieved are *forest* and *highway* followed by *tall buildings*. This is in accordance with the classification results.

Classifying film frames into scenes. In this test the images in OT are again used as training images (8 categories), and key frames from the movie *Pretty Woman* are used as test images. We used $V = 1500$ and $Z = 25$ which are the optimized values for the 8 categories in the OT dataset. Note, this is a second example of vocabulary and topic generalization as we are using training images from a different dataset. We used every hundredth frame from the movie to form the test set. In this movie there are only a few images that could be classified as the same categories used in OT, and there are many images containing only people. So it is a difficult task for the system to correctly classify the key frames. Although the results obtained (see Figure 11) are purely anecdotal, they are very encouraging and show again the success of using pLSA in order to classify scenes according to their topic distribution.

Segmentation. Figure 12 shows examples of segmentation of five topics using the colour SIFT vocabulary. Circular patches are painted according to the maximum posterior $P(z|w,d)$:

$$P(z|w,d) = \frac{P(w|z)P(z|d)}{\sum_{z_l \epsilon Z} P(w|z_l)P(z_l|d)} \tag{5}$$

For each visual word in the image we choose the topic with maximum posterior $P(z|w,d)$ and paint the patch with its associated colour, so each colour represents a different topic (the topic colour is chosen randomly). To simplify the figures we only paint one topic each time. Note that topics represent consistent regions across images (enabling a coarse segmentation) and there is a straightforward correspondence between topic and object.

X. Discussion – the Scene Classification Task

Figure 13a shows the confusion matrix between the 8 categories in OT dataset when no spatial information is used. The best classified scenes are *highway* and *forest* with a performance of 89.8% and 98.8% respectively. The most difficult scenes to classify are *open country*. There is confusion between the *open country* and *coast* scenes, and between the *open country* and *mountain scenes*. The most confused man made images are *street*, *inside city* and *highway*. These are also the most confused categories in [24]. We can also establish some relationship amongst the categories by looking at the distances among the topic distributions between them (see the dendrogram in Figure 13b). When the topic distributions are close, the categories are also close to each other on the dendrogram. For example, the closest natural categories are *open country* and *coast* and the closest man-made are *inside city* and *street*.

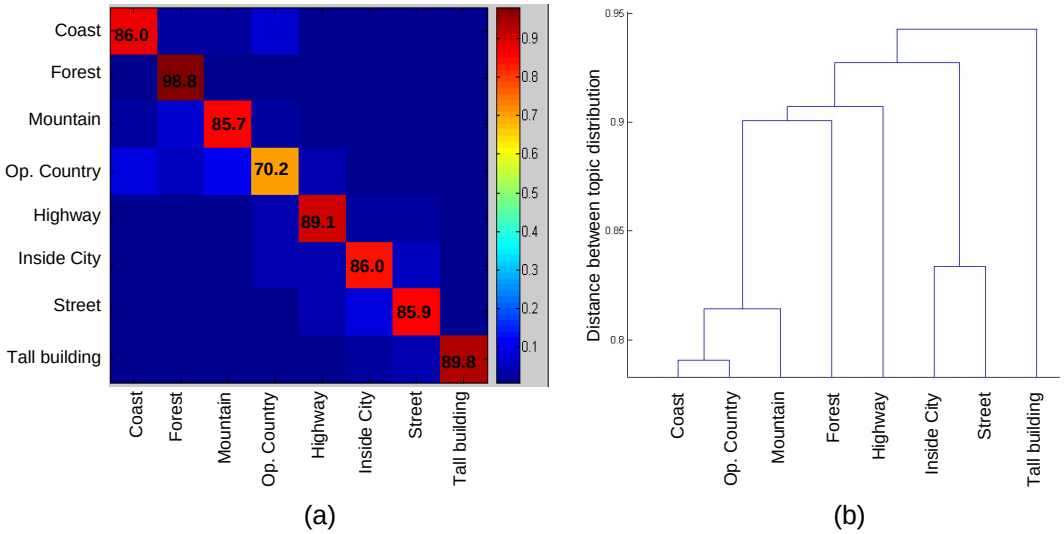


Fig. 13. (a) Confusion matrix for the 8 categories in the OT dataset. (b) Dendrogram showing the closest categories, which are also the most confused.

Figure 14 shows some images confused between categories showing the ambiguity between some of them. Scene categorization is characterized by potential ambiguities since it depends strongly on the subjective perception of the viewer. For example some of the *open country* images shown in Figure 14a can be easily classified as *mountain* for some humans as the system did. Obviously, the obtainable classification accuracies depend strongly on the consistency and accuracy of the manual annotations, and sometimes annotation ambiguities are unavoidable. For example, the annotation of *mountains* and *open country* is quite challenging. Imagine an image with *fields* and *snow hills* in the far distance: is it *open country* or *mountain*? Even more confused are *coast* and *open country* scenes (Figure 14b) yet both of them have a similar structure: *water* or *fields* and the *sky* in the distance. For that reason, it is not surprising that *coast* and *open country* are confused in both directions. Another major confusion appears between *streets* and *highway*. This results mainly from the fact that each street scene contains a *road* whereas the most important part of highway scenes is the *road*. *Streets* and *inside city* images are confused because normally streets occur in cities.

Figure 15a shows the confusion matrix for the 8 categories in the OT dataset when using SP-pLSA with $L = 2$. Now for the *forest* scenes we obtain a rate of 100% of correct classified images, and all the classification rates for the other scenes are also increased. Again the most

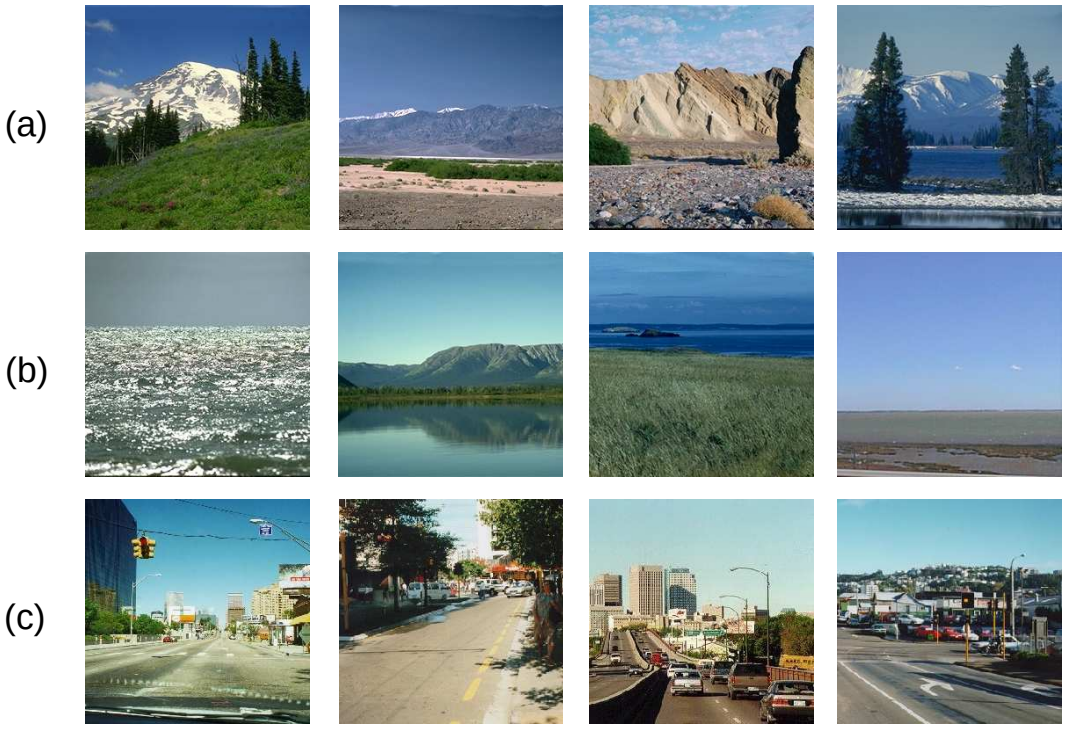


Fig. 14. Images showing the most confused categories: (a) open country images classified as mountains; (b) coast images classified as open country; (c) highway images classified as street.

difficult scenes to classify are the *open country*. Figure 15b shows some images well classified using SP-pLSA and poorly classified without spatial information. This demonstrates that spatial distribution can reduce the ambiguity – or at least that spatial distribution correlates with the annotator's choices. However we are still far from 100% correct classification, again due to the ambiguities between the scene categories used. Vogel and Schiele [32] analyzed in detail the ambiguities between scene categories, showing that there is a semantic transition between categories. Their experiments with human subjects showed that many images cannot be clearly assigned to one category. How far away must a *mountain* be so that the image moves from the *mountains* category to the *open country* category? How much *road* is necessary to make a *street* image into a *highway* image and vice versa? And we arrive at the same conclusion as [32]: it is not wise to aim for a hard decision categorization of scenes. However, since scenes, that is full images, contain very complex semantic details, hard scene categorization is an appropriate task for: (i) testing the image representation [32], in this case provided by topics,

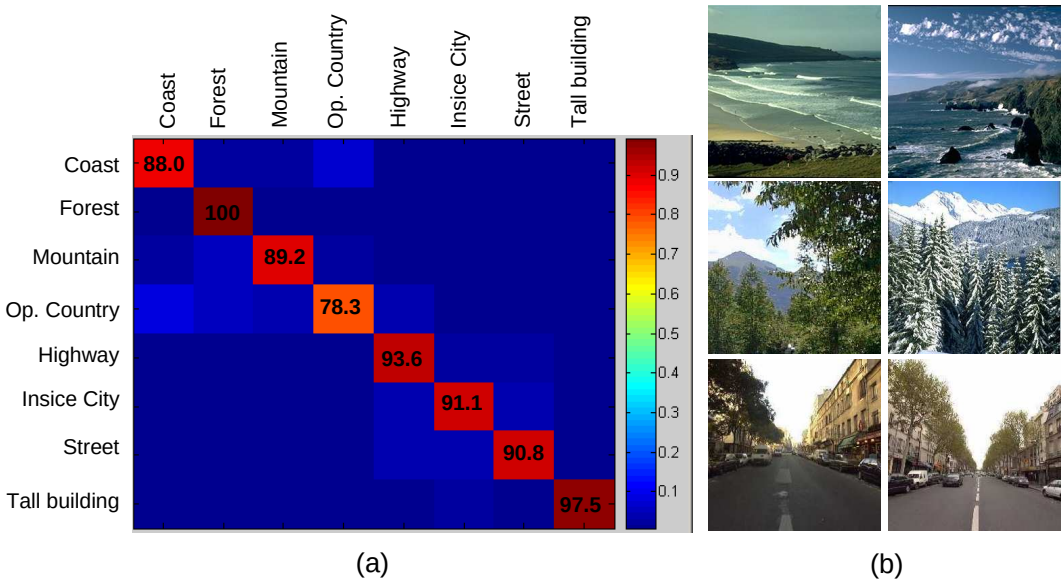


Fig. 15. (a) Confusion matrix for the 8 categories in the OT dataset when using SP-pLSA. (b) Top: two coast scenes classified as mountain when spatial information is not used, and correctly classified using SP-pLSA; Middle: two forest scenes classified as mountains without spatial information, and correctly classified with SP-pLSA; Bottom: two street scenes classified as highway without spatial information, and correctly classified using SP-pLSA.

(ii) as an approximation for how the ranking on an image retrieval system would work, and (iii) classifying mutually exclusive scenes such as indoor/outdoor, garden/bathroom or coast/kitchen.

We have done some preliminary experiments with k-means clustering the image topics provided by SP-pLSA to automatically detect visually similar categories. The results are interesting because the resulting clusters have a semantic meaning such as fields with mountains at the back, fields with flowers, coasts with rocks, sunshine coast, highway with cars and without cars etc. Nevertheless, the images with a semantic transition between categories are not well clustered (because there are not sufficient ambiguous images). A solution would be to use EM soft assignment in the clustering.

XI. Conclusions

We have proposed a scene classifier that learns topics and their distributions in unlabelled training images using pLSA, and then uses their distribution in test images as a feature vector in a supervised discriminative classifier. In contrast to previous approaches [9], [24], [32], our topic learning stage is completely unsupervised and we obtain significantly superior performance

in the pure bag of words situation (no spatial information). We also have shown that the pLSA adapted to incorporate spatial information at different resolution levels (SP-pLSA) has comparable/slightly superior performance with the spatial pyramid matching proposed in [18]. We also demonstrated that using more than one patch to represent each pixel gives better performance, outperforming the method in [18] when using their own approach with spatial information.

We studied the influence of various descriptor parameters and have shown that using dense SIFT descriptors with overlapping patches gives the best results for man-made as well as for natural scene classification. Furthermore, discovered topics correspond fairly well with different textural objects (grass, mountains, sky) in the images, and topic distributions are consistent between images of the same category. It is probably this freedom in choosing appropriate topics for a dataset, together with the optimized features and vocabularies, that is responsible for the superior performance of the scene classifier over previous non-spatial work (even in cases where manual annotation was provided). Moreover, the use of pLSA is never detrimental to performance, and it gives a significant improvement over the original BoW model when a large number of scene categories are used.

ACKNOWLEDGEMENTS

Thanks to Antonio Torralba, Julia Vogel, Fei-Fei Li and Lazebnik Lazebnik for providing their datasets, and to Ondra Chum, Rob Fergus, Jan-Mark Geusebroek, David Lowe, Cordelia Schmid, Josef Sivic, Bill Triggs and Tinne Tuytelaars for discussions. This work was partially funded by the research grant BR03/01 from the University of Girona, by the EU NOE PASCAL and by the EU Project CLASS.

REFERENCES

[1] A. C. Berg. *Shape Matching and Object Recognition*. PhD thesis, Computer Science Division, University of California., 2005.

[2] A. Bosch, X. Muoz, and J. Freixenet. Segmentation and description of natural outdoor scenes. *Image and Vision Computing*, 25(5):727–740, May 2006.

[3] A. Bosch, X. Muoz, and R. Marti. A review: Which is the best way to organize/classify images by content? *Image and Vision Computing*, 25(6):778–791, June 2007.

[4] A. Bosch, A. Zisserman, and X. Muoz. Scene classification via pLSA. In *European Conference on Computer Vision*, volume 4, pages 517–530, Graz, Austria, March 2006.

[5] C. Chang. and C. Lin. *LIBSVM: a library for support vector machines*, 2001.

[6] J. L. Crowley and F. Berard. Multi-modal tracking of faces for video communications. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 640–645, San Juan, Puerto Rico, 1997.

[7] 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, Prague, Czech Republic, May 2004.

[8] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893, San Diego, California, June 2005.

[9] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 524–531, Washington, DC, USA, 2005.

[10] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning object categories from google's image search. In *International Conference on Computer Vision*, volume II, pages 1816–1823, Beijing, China, October 2005.

[11] G. D. Finlayson, B. Schiele, and J. L. Crowley. Comprehensive colour normalization. In *European Conference on Computer Vision*, volume 1, pages 475–490, Freiburg, Germany, 1998.

[12] T. Geodeme, T. Tuytelaars, G. Vanacker, M. Nuttin, and L. Van Gool. Omnidirectional sparse visual path following with occlusion-robust feature tracking. In *OMNIVIS Workshop, International Conference on Computer Vision*, volume 3115, pages 207–215, Beijing, China, October 2005.

[13] K. Grauman and T. Darrell. The pyramid match kernel: Discriminative classification with sets of image features. In *International Conference on Computer Vision*, pages 1458–1465, 2005.

[14] K. Grauman and T. Darrell. Pyramid match kernels: Discriminative classification with sets of image features (version 2). Technical Report CSAIL-TR-2006-020, Massachusetts Institute of Technology Computer Science and Artificial Intelligence Laboratory, 2006.

[15] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR Conference on Research and Development in Information Retrieval*, Melbourne, Australia, 1998.

[16] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 41(2):177–196, 2001.

[17] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using affine-invariant regions. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 319–324, Madison, Wisconsin, June 2003.

[18] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178, New York, June 2006.

[19] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International Journal of Computer Vision*, 43(1):29–44, June 2001.

[20] F. Li, 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, CVPR*, page 178, Washington D.C., USA, June 2004.

[21] D. Lowe. Distinctive image features from scale invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[22] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

[23] J. Mutch and D. Lowe. Multiclass object recognition using sparse, localized features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 11–18, New York, June 2006.

[24] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

[25] 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 *International Conference on Computer Vision*, pages 883–890, Beijing, China, October 2005.

[26] J.J. Rocchio. *Relevance feedback in information retrieval.* In the SMART Retrieval System - Experiments in Automatic Document Processing, Prentice Hall, Englewood Cliffs, NJ, 1971.

[27] J. Sivic, B.C. Russell, A.A. Efros, A. Zisserman, and W. T Freeman. Discovering objects and their locations in images. In *International Conference on Computer Vision*, pages 370–377, Beijing, China, October 2005.

[28] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *International Conference on Computer Vision*, volume 2, pages 1470–1477, Nice, France, October 2003.

[29] M. Szummer and R. W. Picard. Indoor-outdoor image classification. In *ICCV Workshop on Content-based Access of Image and Video Databases*, pages 42–50, Bombay, India, 1998.

[30] A. Vailaya, A. Figueiredo, A. Jain, and H. Zhang. Image classification for content-based indexing. *IEEE Transactions on Image Processing*, 10:117–129, 2001.

[31] M. Varma and A. Zisserman. Texture classification: Are filter banks necessary? In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 691–698, Madison, Wisconsin, June 2003.

[32] J. Vogel and B. Schiele. Semantic modeling of natural scenes for content-based image retrieval. *International Journal of Computer Vision*, 72(2):133–157, January 2007.

[33] G. Wang, Y. Zhang, and L. Fei-Fei. Using dependent regions for object categorization in a generative framework. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1597–1604, New York, June 2006.

[34] J. Weijer and C. Schmid. Coloring local feature extraction. In *European Conference on Computer Vision*, volume 2, pages 332–348, Graz, Austria, May 2006.

[35] H. Zhang, A. Berg, M. Maire, and J. Malik. SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2126–2136, New York, June 2006.

[36] J. Zhang, M. Marszałek, and C. Lazebnik, S. Schmid. Local features and kernels for classification of texture and object categories: a comprehensive study. *International Journal of Computer Vision*, 2007.

[37] R. Zhang and Z. Zhang. Hidden semantic concept discovery in region based image retrieval. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 996–1001, Washington, DC, USA, June 2004.