Fisher Kernels on Visual Vocabularies for Image Categorization

Florent Perronnin and Christopher Dance
Xerox Research Centre Europe
6 chemin de Maupertuis, 38240 Meylan, France
Firstname.Lastname@xrce.xerox.com
http://www.xrce.xerox.com

Abstract

Within the field of pattern classification, the Fisher kernel is a powerful framework which combines the strengths of generative and discriminative approaches. The idea is to characterize a signal with a gradient vector derived from a generative probability model and to subsequently feed this representation to a discriminative classifier. We propose to apply this framework to image categorization where the input signals are images and where the underlying generative model is a visual vocabulary: a Gaussian mixture model which approximates the distribution of low-level features in images. We show that Fisher kernels can actually be understood as an extension of the popular bag-of-visterms. Our approach demonstrates excellent performance on two challenging databases: an in-house database of 19 object/scene categories and the recently released VOC 2006 database. It is also very practical: it has low computational needs both at training and test time and vocabularies trained on one set of categories can be applied to another set without any significant loss in performance.

1. Introduction

Image categorization is the pattern classification problem which consists in assigning one or multiple labels to an image based on its semantic content. This is a very challenging task as one has to cope with inherent object/scene variations as well as changes in viewpoint, lighting and occlusion. Hence, although much progress has been made in the past few years, image categorization remains an open problem. Several approaches consist in modeling the distribution of low-level features contained in images irrespective of their absolute or relative locations within the image. Despite their relative simplicity, such approaches have shown state-of-the-art performance in a recent evaluation [1].

The most popular approach, which was inspired by the bag-of-words used in text categorization, is referred to as the bag-of-keypatches [2] or bag-of-visterms (BOV) [11]. In the following, we use the latter denomination which is more general (the term keypatches assumes the use of an interest point detector for the extraction of low-level feature vectors). Given a *visual vocabulary*, the idea is to characterize an image with the number of occurrences of each visual word. Any classifier can then be used for the categorization of this histogram representation. Most of the work on bags-of-visual-words has focused on the estimation of the visual vocabulary. This is done through the clustering of low-level feature vectors using for instance K-means [2, 15], Gaussian Mixture Models (GMM) [3, 14] or mean-shift [7].

It has been observed that, even on databases containing a restricted number of categories (< 10), the best performance is generally obtained with large vocabularies containing from several hundreds to several thousands of visual words [2, 7, 11, 14, 15, 18]. As the cost of histogram computations depends directly on the number of visual words, one way to reduce the computational cost is to have more compact vocabularies. In [17] an approach based on the information bottleneck principle was proposed. A vocabulary containing initially several thousands of words was reduced down to approximately 200 words without any loss of performance. Another way to reduce the computational cost is to organize the vocabulary in a tree structure, *e.g.* using Extremely Randomized Clustering Forests [12]. However, in both cases, the derived vocabularies are not *universal*: they are tailored to the categories under consideration and would have to be learned again for a new set of categories. This is an issue when one wants to add new categories incrementally to a category set without fully retraining the system. This is likely to happen when dealing with a large number of categories as the full set of categories may not be known beforehand.

As the two objectives of having a truly universal and compact vocabulary seem irreconcilable, some researchers have departed from the idea of having one unique visual vocabulary across images and proposed to have one (much smaller) per-image vocabulary. In [18], K-means clustering is applied to estimate 40 visual words per image and the similarity between image signatures is measured with the Earth Mover's Distance (EMD). In [3], a single visual word (a Gaussian with full covariance matrix) is estimated per image and the Bhattacharyya distance is used to measure the similarity between Gaussian distributions. In both cases kernel-based classification was performed using the Support Vector Machine (SVM). However, the use of small per-image vocabularies does not necessarily lead to a reduced computational cost. Indeed, for such approaches the vocabulary has to be learned online. Moreover, both the EMD and Bhattacharyya distance are significantly more costly than the linear or χ^2 kernels which are traditionally used to classify bags-of-visual-words.

To overcome the limitations of the previously mentioned techniques, we propose to apply *Fisher kernels* to image categorization. The Fisher kernel is a powerful framework which combines the strengths of generative and discriminative approaches to pattern classification [5]. The idea is to characterize a signal with a gradient vector derived from a probability density function (pdf) which models the generation process of the signal. This representation can then be used as input to a discriminative classifier. For the problem of image categorization the input signals are images and we propose to use as a generative model a GMM which approximates the distribution of low-level features in images, i.e. a visual vocabulary. Note that Fisher kernels have already been applied to the problem of image categorization but on a very different model: the constellation model [4]. Also, Fisher kernels on GMMs have been successfully applied to audio indexing [13] and speaker recognition [16].

The gradient representation of the Fisher kernel has a major advantage over the histogram of occurrences of the BOV: for the same vocabulary size, it is much larger (in our experiments, a hundred times larger). Hence, there is no need to use costly kernels to (implicitly) project these very high-dimensional gradient vectors into a still higher dimensional space: linear classifiers already provide excellent results.

One important choice in the design of the generative model of a Fisher kernel is the presence or the absence of the class label as a latent variable. When the model contains the label as latent variable, [5] shows that the Fisher kernel has the desirable property to be asymptotically as good as the Maximum a Posteriori (MAP) decoder. However, in this case the visual vocabulary has to be learned in a supervised manner and cannot be easily extended to a new task.

The remainder of this paper is organized as follows. In 2 we introduce the principle of Fisher kernels. In 3 we apply Fisher kernels to visual vocabularies modeled by GMMs and show that the Fisher kernel generalizes the traditional BOV approach. In 4 we discuss the design of the GMM, *i.e.* whether the GMM should contain the class label as latent variable. In 5 we show experimentally the excellent performance of our approach on two challenging datasets: an in-house database of 19 object/scene categories and the recently released VOC 2006 database which contains 10 objects. We also show how the visual vocabularies derived for one of these tasks can be directly applied to the other task without any significant loss of performance. Finally, we draw conclusions.

2. Fisher Kernels Principle

Pattern classification techniques can be divided into the classes of *generative* approaches and *discriminative* approaches. While the first class focuses on the modeling of class-conditional probability density functions, the second one focuses directly on the problem of interest: classification. This explains the theoretical superiority of discriminative methods over generative ones. However, generative approaches have a number of properties which still make them attractive, including the possibility to handle variable length data.

Fisher kernels have been introduced to combine the benefits of generative and discriminative approaches [5]. Let p be a pdf whose parameters are denoted λ. Then one can characterize the samples $X = \{x_t, t = 1...T\}$ with the following gradient vector:

$$\nabla_\lambda \log p(X|\lambda) \ . \tag{1}$$

Intuitively, the gradient of the log-likelihood describes the direction in which parameters should be modified to best fit the data. It transforms a variable length sample X into a fixed length vector whose size is only dependent on the number of parameters in the model.

This gradient vector can then be classified using any discriminative classifier. For those discriminative classifiers which use an inner product term it is important to normalize the input vectors. In [5], the Fisher information matrix F_λ is suggested for this purpose:

$$F_\lambda = E_X \left[\nabla_\lambda \log p(X|\lambda) \nabla_\lambda \log p(X|\lambda)' \right] \ . \tag{2}$$

The normalized gradient vector is thus given by:

$$F_\lambda^{-1/2} \nabla_\lambda \log p(X|\lambda) \ . \tag{3}$$

Because of the cost associated with its computation and inversion, F_λ is often approximated by the identity matrix and no normalization is performed. In the next section, we will derive a diagonal approximation of F_λ (this corresponds to a dimension-wise normalization of the dynamic range) and in section 5, we will show that using such a normalization impacts favorably the performance.

3. Fisher Kernels on Visual Vocabularies

We propose to apply Fisher kernels on visual vocabularies, where the vocabularies of visual words are represented by means of a GMM. $X = \{x_t, t = 1...T\}$ denotes the set of low-level feature vectors extracted from an image and λ the set of parameters of the GMM. $\lambda = \{w_i, \mu_i, \Sigma_i, i = 1...N\}$ where w_i, μ_i and Σ_i denote respectively the weight, mean vector and covariance matrix of Gaussian i and where N denotes the number of Gaussians. Each Gaussian represents a word of the visual vocabulary: w_i encodes the relative frequency of word i, μ_i the mean of the word and Σ_i the variation around the mean.

We denote $\mathcal{L}(X|\lambda) = \log p(X|\lambda)$. Under an independence assumption, we have:

$$\mathcal{L}(X|\lambda) = \sum_{t=1}^{T} \log p(x_t|\lambda) \, . \tag{4}$$

The likelihood that observation x_t was generated by the GMM is:

$$p(x_t|\lambda) = \sum_{i=1}^{N} w_i p_i(x_t|\lambda) \, . \tag{5}$$

The weights are subject to the constraint:

$$\sum_{i=1}^{N} w_i = 1 \tag{6}$$

and the components p_i are given by:

$$p_i(x|\lambda) = \frac{\exp\left\{-\frac{1}{2}(x-\mu_i)'\Sigma_i^{-1}(x-\mu_i)\right\}}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \, , \tag{7}$$

where D is the dimensionality of the feature vectors and $|.|$ denotes the determinant operator. We assume that the covariance matrices are diagonal as (i) any distribution can be approximated with an arbitrary precision by a weighted sum of Gaussians with diagonal covariances and (ii) the computational cost of diagonal covariances is much lower than the cost involved by full covariances. We use the notation $\sigma_i^2 = \text{diag}(\Sigma_i)$.

In the following, $\gamma_t(i)$ denotes the occupancy probability, i.e. the probability for observation x_t to have been generated by the i-th Gaussian. Bayes formula gives:

$$\gamma_t(i) = p(i|x_t, \lambda) = \frac{w_i p_i(x_t|\lambda)}{\sum_{j=1}^{N} w_j p_j(x_t|\lambda)} \, . \tag{8}$$

The superscript d denotes the d-th dimension of a vector. Straightforward derivations provide the following results:

$$\frac{\partial \mathcal{L}(X|\lambda)}{\partial w_i} = \sum_{t=1}^{T}\left[\frac{\gamma_t(i)}{w_i} - \frac{\gamma_t(1)}{w_1}\right] \text{ for } i \geq 2 \, , \tag{9}$$

$$\frac{\partial \mathcal{L}(X|\lambda)}{\partial \mu_i^d} = \sum_{t=1}^{T} \gamma_t(i)\left[\frac{x_t^d - \mu_i^d}{(\sigma_i^d)^2}\right] \, , \tag{10}$$

$$\frac{\partial \mathcal{L}(X|\lambda)}{\partial \sigma_i^d} = \sum_{t=1}^{T} \gamma_t(i)\left[\frac{(x_t^d - \mu_i^d)^2}{(\sigma_i^d)^3} - \frac{1}{\sigma_i^d}\right] \, . \tag{11}$$

Note that (9) is defined for $i \geq 2$ as there are only $(N-1)$ free weight parameters due to the constraint (6) (w_1 was supposed to be given knowing the value of the other weights). The gradient vector is just a concatenation of the partial derivatives with respect to all the parameters.

To normalize the dynamic range of the different dimensions of the gradient vectors, we need to compute the diagonal of the Fisher information matrix F. Let us denote by f_{w_i}, $f_{\mu_i^d}$ and $f_{\sigma_i^d}$ the terms on the diagonal of F which correspond respectively to $\partial\mathcal{L}(X|\lambda)/\partial w_i$, $\partial\mathcal{L}(X|\lambda)/\partial\mu_i^d$ and $\partial\mathcal{L}(X|\lambda)/\partial\sigma_i^d$. The normalized partial derivatives are thus $f_{w_i}^{-1/2}\partial\mathcal{L}(X|\lambda)/\partial w_i$, $f_{\mu_i^d}^{-1/2}\partial\mathcal{L}(X|\lambda)/\partial\mu_i^d$ and $f_{\sigma_i^d}^{-1/2}\partial\mathcal{L}(X|\lambda)/\partial\sigma_i^d$. It can be shown that we have approximately:

$$f_{w_i} = T\left(\frac{1}{w_i} + \frac{1}{w_1}\right) \, , \tag{12}$$

$$f_{\mu_i^d} = \frac{Tw_i}{\left(\sigma_i^d\right)^2} \, , \tag{13}$$

$$f_{\sigma_i^d} = \frac{2Tw_i}{\left(\sigma_i^d\right)^2} \, . \tag{14}$$

To the best of our knowledge, this is the first time a closed form approximation is proposed for the Fisher information matrix of a GMM. For more details of these derivations, the reader is referred to the appendix.

Let us now relate the traditional BOV to Fisher kernels. In the BOV representation, the relative number of occurrences of the i-th word is given by:

$$\frac{1}{T}\sum \gamma_t(i) \, . \tag{15}$$

From equations (9) and (15), it is clear that the BOV is directly related to the Fisher kernel when one considers only the gradient with respect to the weight parameters: they both consider 0-th order statistics (word counting). However, when taking the derivatives with respect to the means and standard deviations, the Fisher kernel also considers 1-st and 2-nd order statistics (c.f. equations (10) and (11)). With a given vocabulary of size N, the BOV leads to an N-dimensional histogram while the full gradient representation gives a vector of dimensionality $(2 \times D+1) \times N-1$. As

$D = 50$ in our experiments (c.f. section 5), the dimensionality of the gradient representation is approximately 100 times larger. This enables to characterize images with very high-dimensional vectors, even with fairly small vocabularies containing on the order of 100 words.

4. Design of the Visual Vocabulary

We now discuss the design of the visual vocabulary (i.e. of the GMM) that is used as the generative model for the Fisher kernel. The simplest idea is to train the GMM in an unsupervised manner with the low-level feature vectors from all categories or even on a separate dataset [2, 7, 11, 15]. However, in [5], the authors state that

> "A kernel classifier employing the Fisher kernel derived from a model that contains the label as latent variable is, asymptotically, at least as good a classifier as the MAP labeling based on the model".

In the case of a GMM, for a K-class problem where classes are denoted ω_k, having the label as latent variable means the pdf has the form:

$$p(x) = \sum_{k=1}^{K} p(\omega_k) p(x|\omega_k) \, , \tag{16}$$

where each class-conditional $p(x|\omega_k)$ is itself a GMM. These class-conditional pdfs have to be learned in a supervised manner using the training material of the corresponding category. In this case, the same vocabulary cannot be used across tasks and has to be learned for each different set of categories.

Supervised learning of GMMs has already been considered in the BOV framework. In [3], the authors propose the training of one vocabulary $p(x|\omega_k)$ per category and the creation of a single vocabulary by folding together all the Gaussians. A significant improvement was demonstrated compared to unsupervised vocabulary estimation. Unfortunately, this approach is impractical for a large number of categories as the vocabulary size increases linearly with the number of categories. To make this approach more practical, the authors of [14] transform the K-class problem into K 2-class problems. They make use of both a universal vocabulary, which describes the visual content of all the considered classes, and class-vocabularies. For each class, a new combined vocabulary is created by merging the universal and class-vocabulary. For a given image, one histogram is computed for each category on the combined vocabularies. Each of these histograms describes whether the image content is best modeled by the universal vocabulary or the corresponding class vocabulary. Due to the strong link between the BOV and the Fisher kernel (c.f. previous section), the fact that the models of [14] and [3] include the label as a latent variable may explain why they outperform those approaches where the vocabulary is trained in an unsupervised manner.

Hence we will test the Fisher kernels on vocabularies trained in an unsupervised manner (*i.e.* in the case where the model does not contain the label as a latent variable) and also in a supervised manner (*i.e.* in the case where the model contains the label as a latent variable). For the latter case, we preferred the approach of [14] over [3] due to its practicality and its better performance. This means that, for a given image, we will derive one gradient representation per category.

5. Experimental Validation

In this section, we first describe our experimental setup. We then carry out a comparative evaluation of the proposed Fisher kernel on two challenging databases: an in-house database of 19 object/scene categories and the recently released VOC 2006 database [1]. We finally perform cross-database experiments where the vocabulary derived from one database is used for the other one.

5.1. Experimental setup

We used two types of low-level local feature vectors in our experiments. They are extracted on regular grids at different scales. As all images were resized to contain approximately the same number of pixels, roughly the same number of features was extracted from all images (between 500 and 600 for each feature type). The first features are based on local histograms of orientations as described in [10] (128 dimensional features). The second ones are simple local RGB statistics (96 dimensional features). In both cases, the dimensionality of the feature vectors was reduced to 50 through Principal Component Analysis (PCA).

To train a GMM in an unsupervised manner we used the Maximum Likelihood (ML) criterion. The strategy we employed consists in starting with one Gaussian and then increasing progressively the number of Gaussians in the system as in [14]. This vocabulary is later referred to as "universal". To train class-vocabularies we also followed the approach of [14] and adapted them from the universal vocabulary using the MAP criterion.

For the classification of word histograms and gradient representations, we experimented with the SVM using the SVMlight package [6] and our own implementation of Sparse Logistic Regression (SLR) [8], i.e. logistic regression with a Laplacian prior. In both cases, one linear classifier was trained per category in a one-versus-all manner. As both classifiers performed very similarly (as observed experimentally in [8]), we report results only for SLR.

We ran two systems separately, one for each feature type.

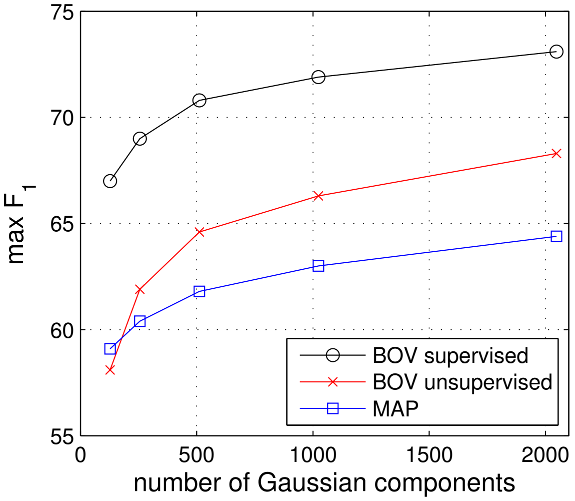

Figure 1. Performance of the three baseline systems as a function of the number of Gaussian components on our in-house dataset. Performance plateaus after 2048 Gaussians.

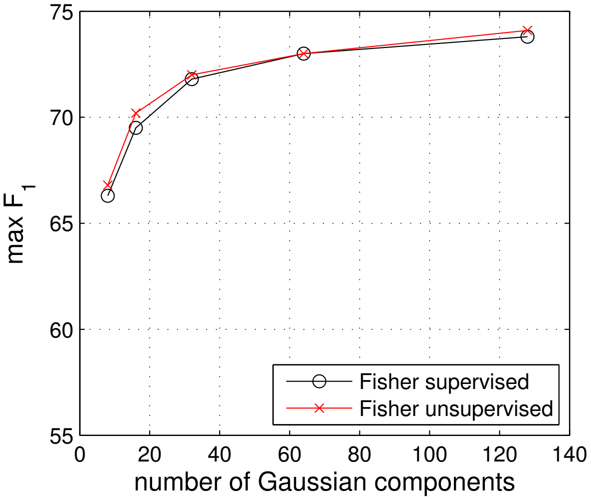

Figure 2. Performance of the Fisher kernels as a function of the number of Gaussian components on our in-house database. Performance plateaus after 128 Gaussians.

The final score is simply the average of the scores of the two systems.

5.2. In-house database

The first set of experiments was carried out on an in-house dataset of 19 object/scene categories: beach, bicycling, birds, boating, cats, clouds/sky, desert, dogs, flowers, golf, motorsports, mountains, people, sunrise/sunset, surfing, underwater, urban, waterfalls and wintersports. This is a very challenging dataset as the training and test images were collected independently. Approximately 30K images were available for training and 5K for testing. Both sets were manually multi-labeled. Our measure of performance is the maximum of the F_1 criterion which is heavily used in the text categorization literature. The F_1 measure is defined as the harmonic mean of the precision and the recall.

As this is not a publicly available dataset, we ran three systems which will serve as a baseline for Fisher kernels: (i) the traditional BOV where the universal vocabulary is trained in an unsupervised manner (unsupervised BOV), (ii) the approach of [14] which makes use of both a universal and class vocabularies (supervised BOV) and (iii) the MAP decoder based on the class vocabularies. The main parameter which will affect the performance of these methods is the number of Gaussian components, i.e. of visual words. Hence, the performance is shown on figure 1 for a varying number of Gaussian components. The approach of [14] is the best performing one and its best performance is reached for 2048 Gaussians with $\max F_1 = 73.1\%$.

As for the Fisher kernel approach, two parameters will mainly affect its performance: (i) the number of Gaussian components and (ii) the parameters with respect to which the gradient is computed. We first study on figure 2 the influence of the number of Gaussian components in the case where the gradient is taken with respect to all three parameter types (weights, means and standard deviations). We provide results in the cases where (i) the class label is a latent variable of the visual vocabulary (Fisher supervised) and (ii) the class label is not a latent variable of the visual vocabulary (Fisher unsupervised). Despite the significantly higher complexity of the supervised approach (one gradient vector per image per category instead of a single gradient vector per image for the unsupervised approach) both approaches perform very similarly. Hence, the unsupervised approach is preferred for its simplicity. The best performance of the Fisher kernel on the visual vocabulary learned in an unsupervised manner is 74.1% with a GMM containing 128 Gaussians. Note that, if we had approximated the Fisher information matrix with the identity matrix instead of using the diagonal approximation proposed in section 3, the performance of this model would have decreased down to 70.7%. Also, if we had used a simpler normalization in the range [-1,1], as done in [4], the performance would have been 72.2%.

While the improvement in terms of $\max F_1$ is modest (+1.0% absolute compared to the approach of [14]), the reduction of the computational cost at both training and test time is very significant due to the fact that we can use a much more compact vocabulary. With our implementation of [14], training the whole system from scratch for both types of low-level feature vectors with 30K images takes approximately 19h of CPU time on a 2.4GHz AMD

step		BOV	Fisher
feature extraction	low-level	150	
	high-level	550	20
Training			
vocabulary		$1000 + 30 \times C$	40
SLR		$3 \times C$	$4 \times C$
Testing			
classification		$0.02 \times C$	$0.06 \times C$

Table 1. Breakdown of the computational cost (in ms) per image for the BOV approach of [14] and the proposed Fisher kernel. C is the number of considered categories. The first two steps are common to both training and testing phases. The high-level feature extraction corresponds to the computation of the histograms of word occurrences for the BOV and to the gradient computation in the case of Fisher kernels.

gradient	max F_1 (in %)	gradient dimension
w	58.1	127
μ	69.4	6,400
σ	70.4	6,400
$\mu\sigma$	74.1	12,800
w$\mu\sigma$	74.1	12,927

Table 2. Contribution of each parameter (w = weights, μ = mean and σ = standard deviation) to the classification accuracy and to the dimensionality of the gradient space for a GMM with 128 Gaussians.

Opteron$^{\text{TM}}$with 4GB Ram. With Fisher kernels, the training cost is reduced down to approximately 2h30. As for the test time, it is reduced from 700ms down to 170ms per image. One can refer to table 1 for a breakdown of the training and testing costs.

We now analyze the contribution of each parameter type of the GMM to the classification accuracy. This is done by taking the gradient of the log-likelihood with respect to only a subset of the parameters. Experiments were carried out on our best system with 128 Gaussians. Results are shown in table 2, along with the dimensionality of the gradient representation. When one takes the gradient with respect to weights only (equivalent to the traditional BOV histogram), one obtains a much poorer performance than when taking the gradient with respect to means or standard deviations only. This is not surprising as the gradient with respect to weights has a much lower dimensionality (50 times smaller). When taking the gradient with respect to both means and standard deviations one obtains a significant improvement over either parameter alone. However, when taking the gradient with respect to means, standard deviations and weights, no further improvement is obtained.

5.3. VOC 2006 database

The second set of experiments was carried out on the recently released VOC 2006 database [1]. It consists of 10 object categories: bicycle, bus, car, cat, cow, dog, horse, motorbike, person and sheep. 2,618 images are available for training and 2,686 for testing. As was the case for our in-house database, we trained one GMM with 128 Gaussians for both types of low-level feature vectors and, to build our representation, we took only the gradient with respect to the means and standard deviations (c.f. previous subsection).

In Spring 2006, two rounds of public evaluations were carried out on this database. Our focus is on the competition called "comp1" for which only the provided training material could be used to train the classifiers. In the following, we consider those 20 systems which ran against all categories in the "comp 1" challenge (18 during the first round and 2 during the second round). The measure of performance used during the competition was the Area Under the Curve (AUC). In figure 3 we provide our per-category results and compare them with the median and the best results reported in [1]. In figure 4 we compare the average AUC of our system over the 10 categories, which is equal to 0.931, with the average AUCs of the other systems. The best average AUC reported on this database is 0.936. The proposed system is thus very close to the state-of-the-art.

5.4. Cross database experiments

We wanted to make sure that the vocabulary trained for one set of categories could be used for another set of categories without any significant performance degradation. Hence, we trained a visual vocabulary on one database and used it as the generative model for the Fisher kernel for the other database. When training on the VOC 2006 database and testing on our in-house database, the decrease in performance was insignificant (from 74.1% down to 73.9%). When training on our in-house database and testing on VOC 2006, no decrease in performance was observed. This means that Fisher kernels are fairly insensitive to the quality of the generative model (as observed in [5]) and therefore that the same vocabulary can be used for different category sets. Obviously, this property might not hold if the sets of categories are widely different, for instance, if the visual vocabulary is trained on sketches and used to categorize natural images.

6. Conclusion

In this paper we introduced a novel approach to image categorization which consists in applying the Fisher kernel framework on a visual vocabulary, i.e. a GMM which models the generative process of the low-level feature vectors extracted from images.

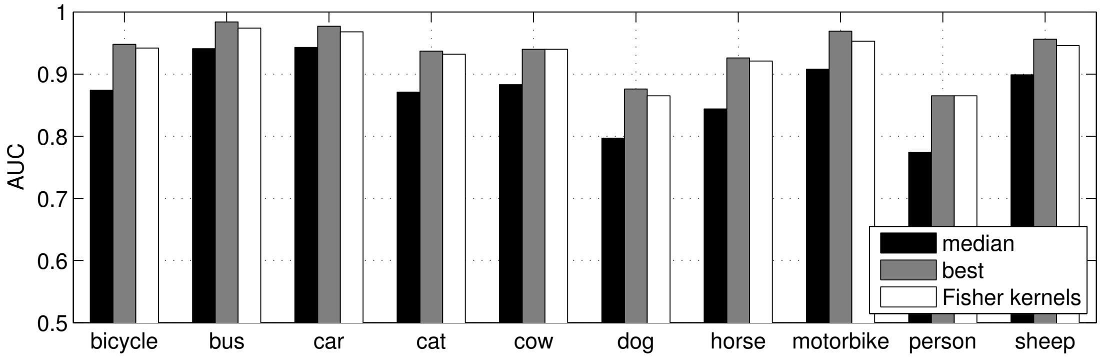

Figure 3. Per category results on the PASCAL VOC 2006: AUC for the Fisher kernel approach (in white) compared to the median and best AUC reported in [1] (in black and grey respectively).

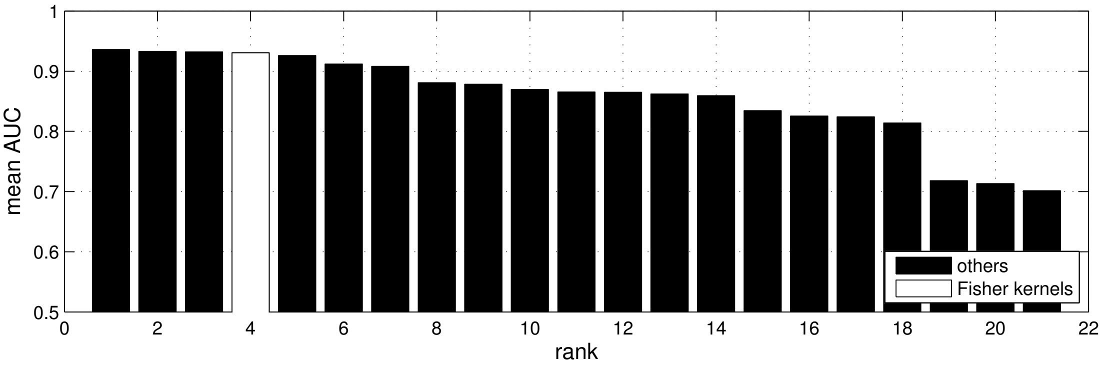

Figure 4. Average AUC over the 10 categories of the PASCAL VOC 2006: ranking of the proposed Fisher kernel approach with respect to the other 20 systems reported in [1].

We showed that the proposed approach is actually a generalization of the popular BOV. The main advantage over the BOV is that, for the same vocabulary size, the gradient representation of the Fisher kernel has a much higher dimensionality than the histogram representation (100 times larger in our experiments). Hence, high dimensional and highly informative representations can be derived from images, even with very compact vocabularies containing on the order of 100 words. This makes the proposed approach very attractive from a computational standpoint. The ability to use compact vocabularies was obtained without sacrificing the generalization ability of our vocabularies. Indeed, it was shown experimentally that vocabularies trained on one set of categories could be exported to another set of categories without any significant loss of performance.

We evaluated our approach on two challenging datasets, including the recently released VOC 2006 database, and showed that the proposed approach produces state-of-the-art results.

It is important to notice that any kernel based on a generative model can be applied to image categorization with the framework proposed in this paper, i.e. using the visual vocabulary as the generative model. Hence, other kernels such as the log-likelihood ratio kernel [9] introduced in the field of automatic speech recognition, will most certainly be worth testing in the future.

A. Derivation of the Fisher Information Matrix

Our derivations are based on two assumptions. The first one is that the number of low-level features x_t extracted from each image is constant and equal to T. This is a reasonable assumption in our case (c.f. section 5.1). The second one is that for each observation x_t, the distribution of the occupancy probability $\gamma_t(i)$ is sharply peaked. This means that there is one Gaussian index i such that $\gamma_t(i) \approx 1$ and that $\forall j \neq i$, $\gamma_t(j) \approx 0$. This second property is based on empirical observation. In the following, we just provide the details of the computation of f_{w_i} as similar derivations lead to the values of $f_{\mu_i^d}$ and $f_{\sigma_i^d}$.

We recall that $\partial \mathcal{L}(X|\lambda)/\partial w_i$ and thus f_{w_i} are defined for $i \geq 2$. Using the definition of the Fisher kernel (2) and the value of $\partial\mathcal{L}(X|\lambda)/\partial w_i$ (9) we get:

$$f_{w_i} = \int_X p(X|\lambda) \left[\sum_{t=1}^{T} \left(\frac{\gamma_t(i)}{w_i} - \frac{\gamma_t(1)}{w_1} \right) \right]^2 dX \ . \quad (17)$$

Using the following notation:

$$A_t(i) = \frac{\gamma_t(i)}{w_i} - \frac{\gamma_t(1)}{w_1}, \quad (18)$$

we have:

$$\begin{aligned} f_{w_i} &= \sum_{\substack{t=1 \ldots T \\ u=1 \ldots T \\ t \neq u}} \int_{x_t, x_u} A_t(i) A_u(i) p(x_t, x_u|\lambda) dx_t dx_u \\ &+ \sum_{t=1}^{T} \int_{x_t} A_t(i)^2 p(x_t|\lambda) dx_t \ . \quad (19) \end{aligned}$$

For $t \neq u$, using the independence of x_t and x_u, we get:

$$\int_{x_t, x_u} A_t(i) A_u(i) p(x_t, x_u|\lambda) dx_t dx_u \quad (20)$$

$$= \int_{x_t} A_t(i) p(x_t|\lambda) dx_t \int_{x_u} A_u(i) p(x_u|\lambda) dx_u \ . \quad (21)$$

Using the definition of the occupancy probability $\gamma_t(i)$ (8), we obtain:

$$A_t(i) p(x_t|\lambda) = p_i(x_t|\lambda) - p_1(x_t|\lambda) \quad (22)$$

and integrating both sides we get:

$$\int_{x_t} A_t(i) p(x_t|\lambda) dx_t = 0. \quad (23)$$

Turning to the second term of equation (19), we have:

$$A_t(i)^2 = \left(\frac{\gamma_t(i)}{w_i} \right)^2 + \left(\frac{\gamma_t(1)}{w_1} \right)^2 - 2 \frac{\gamma_t(i)\gamma_t(1)}{w_i w_1} \ . \quad (24)$$

Using the property that $\gamma_t(i)$ is sharply peaked (i.e. is either close to 0 or 1), we can write $\gamma_t(i)^2 \approx \gamma_t(i), \forall i$ and $\gamma_t(i)\gamma_t(1) \approx 0, \forall i \geq 2$. Thus:

$$A_t(i)^2 \approx \frac{\gamma_t(i)}{w_i^2} + \frac{\gamma_t(1)}{w_1^2} \ . \quad (25)$$

We obtain:

$$A_t(i)^2 p(x_t|\lambda) \approx \frac{p_i(x_t|\lambda)}{w_i} + \frac{p_1(x_t|\lambda)}{w_1} \ . \quad (26)$$

Integrating both sides we get:

$$\int_{x_t} A_t(i)^2 p(x_t|\lambda) dx_t \approx \frac{1}{w_i} + \frac{1}{w_1} \ , \quad (27)$$

which leads to the following formula for f_{w_i}:

$$f_{w_i} \approx T \left(\frac{1}{w_i} + \frac{1}{w_1} \right) \ . \quad (28)$$

References

[1] The PASCAL visual object classes challenge 2006. http://www.pascal-network.org/challenges/VOC/voc2006/index.html.

[2] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *ECCV Workshop on Statistical Learning for Computer Vision*, 2004.

[3] J. Farquhar, S. Szedmak, H. Meng, and J. Shawe-Taylor. Improving "bag-of-keypoints" image categorisation. Technical report, University of Southampton, 2005.

[4] A. Holub, M. Welling, and P. Perona. Combining generative models and Fisher kernels for object recognition. In *ICCV*, volume 1, pages 136–143, 2005.

[5] T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *Advances in Neural Information Processing Systems 11*, pages 487–493, 1999.

[6] T. Joachims. *Advances in Kernel Methods - Support Vector Learning*, chapter Making large-Scale SVM Learning Practical. MIT-PRESS, 1999.

[7] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *ICCV*, volume 1, pages 604–610, 2005.

[8] B. Krishnapuram, L. Carin, M. Figueiredo, and A. Hartemink. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *IEEE PAMI*, 27(6):957–968, 2005.

[9] M. Layton and M. Gales. Maximum margin training of generative kernels. Technical report, Cambridge University, 2005.

[10] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[11] F. Monay, P. Quelhas, D. Gatica-Perez, and J.-M. Odobez. Constructing visual models with a latent space approach. Technical report, IDIAP, 2005.

[12] F. Moosmann, B. Triggs, and F. Jurie. Randomized clustering forests for building fast and discriminative visual vocabularies. In *NIPS*, 2006.

[13] P. Moreno and R. Rifkin. Using the Fisher kernel method for web audio classification. In *ICASSP*, volume 4, pages 2417–2420, 2000.

[14] F. Perronnin, C. Dance, G. Csurka, and M. Bressan. Adapted vocabularies for generic visual categorization. In *ECCV*, volume 4, pages 464–475, 2006.

[15] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, volume 2, pages 1470–1477, 2003.

[16] V. Wan and S. Renals. Speaker verification using sequence discriminant support vector machines. *IEEE PAMI*, 13(2):203–210, 2005.

[17] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned visual dictionary. In *ICCV*, volume 2, pages 1800–1807, 2005.

[18] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: An in-depth study. Technical report, INRIA, 2005.