Towards Optimal Bag-of-Features for Object Categorization and Semantic Video Retrieval

Yu-Gang Jiang
Dept of Computer Science
City University of Hong Kong
Kowloon, Hong Kong
yjiang@cs.cityu.edu.hk

Chong-Wah Ngo
Dept of Computer Science
City University of Hong Kong
Kowloon, Hong Kong
cwngo@cs.cityu.edu.hk

Jun Yang
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
juny@cs.cmu.edu

ABSTRACT

Bag-of-features (BoF) deriving from local keypoints has recently appeared promising for object and scene classification. Whether BoF can naturally survive the challenges such as reliability and scalability of visual classification, nevertheless, remains uncertain due to various implementation choices. In this paper, we evaluate various factors which govern the performance of BoF. The factors include the choices of detector, kernel, vocabulary size and weighting scheme. We offer some practical insights in how to optimize the performance by choosing good keypoint detector and kernel. For the weighting scheme, we propose a novel soft-weighting method to assess the significance of a visual word to an image. We experimentally show that the proposed soft-weighting scheme can consistently offer better performance than other popular weighting methods. On both PASCAL-2005 and TRECVID-2006 datasets, our BoF setting generates competitive performance compared to the state-of-the-art techniques. We also show that the BoF is highly complementary to global features. By incorporating the BoF with color and texture features, an improvement of 50% is reported on TRECVID-2006 dataset.

Categories and Subject Descriptors

I.4.7 [**Image Processing and Computer Vision**]: Feature Measurement; H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing

General Terms

Algorithms, Measurement, Experimentation.

Keywords

Object categorization, semantic video retrieval, bag-of-features, keypoint detector, soft-weighting, kernel.

1. INTRODUCTION

The problem of classifying images and video shots according to their semantic content is currently one of the most difficult challenges, especially in the presence of within-class variation, occlusion, background clutter, pose and lighting changes. While global features are known to be limited in face of these difficulties, bag-of-features (BoF) which captures the invariance aspects of local keypoint features has recently attracted numerous research attentions. The basic idea of BoF is to depict each image as an orderless collection of local keypoint features. For compact representation, a visual vocabulary is usually constructed to describe BoF through the clustering of keypoint features. Each keypoint cluster is treated as a "visual word" in the visual vocabulary. Through mapping the keypoints in an image to the visual vocabulary, we can describe the image as a feature vector according to the presence or count of each visual word. Under the supervised learning platform (e.g., SVM), the feature vector forms the basic visual cue for object and scene classification. The BoF approach, although is simple and do not contain any geometry information, has demonstrated excellent performance for various visual classification tasks [11, 17, 21, 25].

In this paper, we study and evaluate several factors which could impact the performance of BoF. These factors include the choices of keypoint detector, size of visual vocabulary, weighting scheme of visual words, and kernel function used in supervised learning. Besides offering a thorough study and practical insights into these choices, we also propose a novel soft-weighting method of visual words. We find that the proposed soft-weighting method is consistently better than the traditional weighting schemes used in other recent works. We experimentally show that, by jointly considering all these factors, the performance of BoF could be significantly boosted. Our experiments indicate that BoF is the best single feature on TRECVID-2006 dataset. By combining BoF with grid-based global features (color and texture), the performance is further upgraded (as much as 50%) without sophisticated fusion technique. This indeed signifies the potential of BoF: it is not only effective by itself, but also complementary to global features popularly adopted in the content-based retrieval.

There exist several pioneering researches on BoF including [21, 25]. These works basically adopt techniques in text information retrieval (IR) for modeling BoF. The factors such as the choice of weighting scheme are not addressed and indeed migrated directly from IR without empirical evidence showing their effectiveness. This paper investigates the best possible choices of these factors through empirical verification, aiming to explore the upper limit of BoF performance. The remaining sections are organized as follows. Section 2 describes the existing approaches in object recognition and semantic video retrieval. Section 3 outlines the major factors that dominate BoF, including keypoint detector, vocabulary size, weighting scheme and kernel. Section 4 presents our experimental results and comparisons with state-of-the-art techniques. Finally, Section 5 concludes this paper.

2. RELATED WORKS

Object categorization is a well studied problem in computer vision. Recently, BoF exhibits surprisingly good performance for this problem across several datasets (e.g., [11, 17, 25] among others). In [25], Zhang et al. gave a comprehensive study on the local feature based object and texture classification. They provided comparisons on the choice of some local detectors and proposed to use χ^2 RBF kernel for SVM learning. In [17], Nowak et al. studied the sampling strategies of BoF to contrast dense (local patches) and sparse (keypoints) representation. They claimed that sample size is critical for building vocabulary and thus the randomly sampled local patches could offer a more powerful representation than keypoints. Our empirical findings, however, show that sparse BoF is as good as dense BoF while enjoying the merit of speed efficiency. In [11], Lazebnik et al. integrated the location information of keypoints into BoF. Although we do not investigate this issue, it is expected that spatial information is likely to have positive effect on BoF.

Semantic video retrieval, on the other hand, is to rank shots according to the detection confidence of a semantic concept. Compared to object categorization, the task is conducted in a more diverse setting where the emphasis usually includes feature selection, multi-modality fusion, and machine learning on huge multimedia dataset. Here we focus our review on feature-level analysis which is related to our latter experimental comparison. In [4], the rich sets of features (visual, motion, text, face) and classifiers are demonstrated to have excellent performance on semantic retrieval. Visual features, in particular, are extracted simultaneously from global, grid, region and keypoints levels, activating more than 100 SVMs for learning a single concept. While technically impressive, it becomes expensive to scale up such system, for instance, when thousands of semantic concepts are considered for retrieval. Meanwhile, the approaches in [3, 9, 22] used less features but yet still shown competitive performance to [4]. The features include color and texture (in global and grid levels), motion, text, etc. BoF is also used in [9, 22]. [9] adopted single keypoint detector and descriptor, while [22] used a combination of different keypoint detectors (Harris Laplace and Boosted ColorHarris Laplace) and keypoint descriptors (SIFT and HueSIFT). The ColorHarris Laplace and HueSIFT are constructed by integrating color information into Harris Laplace and SIFT respectively. Improvements of the color boosted features over the traditional ones are observed in [22].

In addition, [19] also used local feature for semantic video retrieval, but in a different way. They adopted geometric blur features [2] with point-to-point matching. The features are computed based on 200 randomly sampled points with high edge energy from a keyframe. Given a test sample, an online point-to-point matching is required between the sample and training exemplars. To avoid computational overhead, a total of 1291 training examples are picked as references. Each keyframe is then represented as a 1291 dimensional vector with each component indicates the distance of the keyframe to a reference. The feature vectors are used directly for SVM learning. Note that although exemplars are adopted in [19] to improve the computational efficiency, this method is still much slower than BoF which uses a visual vocabulary with acceptable amount of visual words (c.f. Section 4.6).

In this paper, we assess while improve the performance of BoF for object categorization and semantic concept retrieval. Different from [9, 22, 11, 17, 25], we separately and jointly consider various factors such as feature weighting and vocabulary size, which could govern the BoF performance but have not yet been seriously addressed in other works.

3. BAG-OF-FEATURES FOR VISUAL CLASSIFICATION

This section introduces the various factors that can affect the performance of BoF for visual classification. We first discuss popular keypoint detectors and the choice of vocabulary size. We then describe our proposed soft-weighting scheme, and different kernels suitable for SVM learning with BoF.

3.1 Keypoint Detector

Keypoint detector samples a sparse set of locally stable points (and their support regions) which forms the basis of BoF. The sampled keypoints are expected to be invariant to geometric and photometric changes. Different detectors, nevertheless, emphasize different aspects of invariances, resulting in keypoints of varying properties and sampled sizes. Here we evaluate six popular keypoint detectors, including Laplacian of Gaussian (LoG) [12], Difference of Gaussian (DoG) [13], Harris Laplace [15], Hessian Laplace [14], Harris Affine [15], and Hessian Affine [14]. In LoG, the scale-space representation is built by successive smoothing of high resolution image with Gaussian based kernels of different sizes. A feature point is then detected if a local 3D extremum is present and if its absolute value is higher than a threshold. The LoG detector is circularly symmetric and it detects blob-like structures. In DoG, the input image is successively smoothed with a Gaussian kernel and sampled. The DoG representation is obtained by subtracting two successive smoothed images. Thus, all the DoG levels are constructed by combined smoothing and sub-sampling. The DoG is an approximate but more efficient version of LoG. The Harris Laplace detector responds to corner-like regions. It uses a scale-adopted Harris function to localize points in scale-space, and then selects the points for which the Laplacian of Gaussian attains a maximum over a scale. Harris-Affine, which is derived from Harris-Laplace, estimates the affine neighborhood by the affine adaptation based on the second moment matrix. Keypoints of Hessian Laplace are points which reach the local maxima of Hessian determinant in space and fall into the local maxima of Laplacian of Gaussian in a scale, while Hessian Affine is achieved after the affine adaptation procedure based on Hessian Laplace.

The performance evaluation of detectors can also be found in [14]. The evaluation criterion in [14] is to measure the repeatability/matching score based on general image transforms, e.g., viewpoint, scale, blur, light, etc. However, the discriminative power of keypoints from different detectors is not studied in [14] for visual classification. In [25], Zhang et al. performed an evaluation of detectors on object and texture classification. Two detectors (Harris Laplace and LoG) and their rotation and affine versions are compared. However, the issue of sampled size, which is an important factor as claimed by [17], is not addressed. Naturally more sampled keypoints could mean more discriminative information for classification. Comparing the effectiveness of detectors without taking into account the sample size can actually bias the detectors which tend to sample more points.

In contrast to [17, 25], we conduct the evaluation of keypoint detectors by assessing their discriminative power, while considering their sampling mechanisms. We use SIFT (scale-invariant feature transform) [13] to describe the regions around the keypoints. SIFT is a 128 dimensional feature vector that captures the spatial structure and the local orientation distribution of a region surrounding a keypoint. Recently studies have shown that SIFT is one of the best descriptors for keypoints [11, 17, 25].

3.2 Vocabulary Size

A visual vocabulary is generated by clustering the detected keypoints in their feature space and treating each cluster as a unique visual word of the vocabulary. Different from text vocabulary in information retrieval, the size of visual vocabulary is determined by the number of keypoint clusters. A small vocabulary may lack the discriminative power since two keypoints may be assigned into the same cluster even if they are not similar to each other. A large vocabulary, on the other hand, is less generalizable, less forgiving to noises, and incurs extra processing overhead.

The trade-off between discrimination and generalization motivates the studies of visual vocabulary size. Our survey shows that previous works used a wide range of vocabulary sizes, leading to difficulty in interpreting their findings. For instance, Lazebnik et al. [11] adopted 200-400 visual words, Zhang et al. [25] adopted 1000, Sivic et al. [21] adopted 6,000 -10,000, etc. In our study, we experiment with vocabularies of 500-10,000 visual words on two different datasets (PASCAL and TRECVID). The vocabulary sizes cover most of the implementation choices in existing works. By using two datasets with very different properties (c.f. Section 4.1), we aim to provide some practical insights on this issue.

3.3 Keyword Weighting

Term weighting is known to have critical impact to text information retrieval. Whether such impact extends to visual keywords remains an interesting question. A fundamental difference is that: text words are sampled naturally according to language context; visual words are the outcomes of data clustering. The former carries semantic sense, while the latter infers statistical information. The existing approaches with BoF mostly adopted conventional term frequency (*tf*) and inverse document frequency (*idf*). In [21], Sivic et al. adopted *tf-idf*, while most of the other works chose *tf* directly [11, 25]. In [17], binary weighting, which indicates the presence and absence of a visual word with values 1 and 0 respectively, was used. Generally speaking, all the weighting schemes perform the nearest neighbor search in the vocabulary in the sense that each keypoint is mapped to the most similar visual word (i.e., the nearest cluster centroid). We argue that, for visual words, directly assigning a keypoint to its nearest neighbor is not an optimal choice, given the fact that two similar points may be clustered into different clusters when increasing the size of visual vocabulary. On the other hand, simply counting the votes (e.g. *tf*) is not optimal as well. For instance, two keypoints assigned to the same visual word are not necessarily equally similar to that visual word, meaning that their distances to the cluster centroid are different. Ignoring their similarity with the visual word during weight assignment cause the contribution of two keypoints equal, and thus more difficult to assess the importance of a visual word in an image.

In order to tackle the aforementioned problems, in [16], Agarwal and Triggs proposed to fit a probabilistic mixture model to the distribution of a set of training local features in the descriptor space, and code new features by their vectors of posterior mixture-component membership probabilities. This method, although interesting, involves a training stage which is not very efficient. Here we propose an straightforward *soft-weighting* approach to weight the significance of visual words. For each keypoint in an image, instead of searching only for the nearest visual word, we select the top-N nearest visual words. Suppose we have a visual vocabulary of K visual words, we use a K-dimensional vector $T = [t_1, ..., t_k, ..., t_K]$ with each component t_k representing the weight of a visual word k in an image such that

$$t_k = \sum_{i=1}^{N} \sum_{j=1}^{M_i} \frac{1}{2^{i-1}} sim(j, k), \tag{1}$$

where M_i represents the number of keypoints whose ith nearest neighbor is visual word k. The measure $sim(j, k)$ represents the similarity between keypoint j and visual word k. Notice that in Eqn 1 the contribution of a keypoint is dependent on its similarity to word k weighted by $\frac{1}{2^{i-1}}$, representing the word is its ith nearest neighbor. Empirically we find $N = 4$ is a reasonable setting.

By using the proposed soft-weighing scheme, we expect to address the fundamental drawbacks of the conventional weighing schemes (e.g., *tf* and *tf-idf*) which are directly migrated from the text retrieval domain.

3.4 Kernels for BoF

Support Vector Machines (SVM) have been one of the most popular classifiers for BoF. For two-class case, the decision function for a test sample x has the following form:

$$g(x) = \sum_{i} \alpha_i y_i K(x_i, x) - b, \tag{2}$$

where $K(x_i, x)$ is the response of a kernel function for the training sample x_i and the test sample x; y_i is the class label of x_i; α_i is the learned weight of the training sample x_i, and b is a learned threshold parameter.

The choice of a good kernel function $K(x_i, x)$ is critical for statistical learning. Although there is a number of general purpose kernels off the shelf, it is unclear which one is the most effective for BoF in the context of visual classification. In [25], Zhang et al. adopted the χ^2 RBF kernel which have shown good performance, while the authors of many other existing works, to our knowledge, chose the traditional linear kernel or Gaussian RBF kernel. In this paper, we will evaluate the following kernels for BoF visual classification:

- *Linear kernel.*

$$K_{linear}(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \mathbf{y}. \tag{3}$$

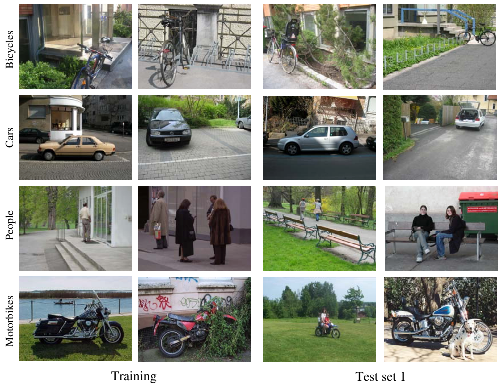

Figure 1: PASCAL Challenge 2005 image examples of different categories.

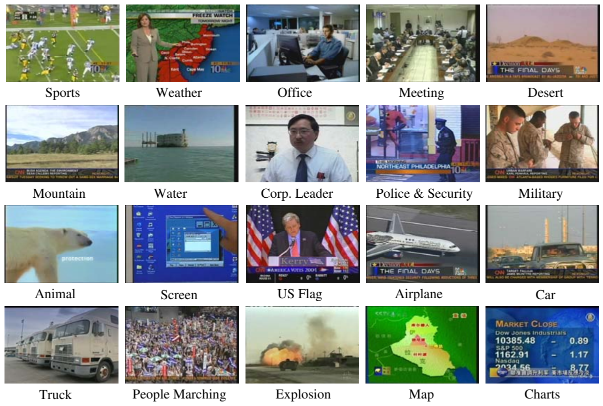

Figure 2: Keyframe examples of 20 semantic categories in TRECVID-2006 evaluation.

- *Generalized forms of RBF kernels.*

$$K_{d-RBF}(\mathbf{x}, \mathbf{y}) = e^{-\rho d(\mathbf{x}, \mathbf{y})}, \quad (4)$$

where $d(\mathbf{x}, \mathbf{y})$ can be chosen to be any distance in the feature space. Since BoF is a histogram of visual words with discrete densities, the functions such as χ^2 distance are more appropriate:

$$d_{\chi^2}(\mathbf{x}, \mathbf{y}) = \sum_i \frac{(x_i - y_i)^2}{x_i + y_i}, \quad (5)$$

which gives a χ^2 RBF kernel. The χ^2 RBF kernel satisfies Mercer's condition [8].

In addition, Chapelle et al. [6] introduced another series of kernels for color histogram based image classification, with the distance function defined as:

$$d_b(\mathbf{x}, \mathbf{y}) = \sum_i |x_i - y_i|^b. \quad (6)$$

Eqn 4 becomes Laplacian and sub-linear RBF kernels respectively when $b = 1, 0.5$. These kernels are popularly used in image retrieval with color histogram as feature, and shown to have good performance than Gaussian RBF kernel ($b = 2$)[6]. The functions $e^{-\rho d_b(\mathbf{x}, \mathbf{y})}$ satisfy Mercer's condition if and only if $0 \leq b \leq 2$ (page 434 in [24]).

- *Histogram Intersection Kernel.* The Histogram Intersection (HI) kernel was proposed and proven to be Mercer kernel in [18]:

$$K_{HI}(\mathbf{x}, \mathbf{y}) = \sum_i \min\{x_i, y_i\}, \quad (7)$$

4. EXPERIMENTS

4.1 Datasets and Visual Word Generation

We evaluate the performance of BoF on two datasets: 1) PASCAL-2005 VOC Challenge [7] - a typical dataset for object categorization, 2) TRECVID-2006 [23] - a popular and huge video dataset for semantic retrieval.

The PASCAL-2005 datset contains four object classes: bicycles, cars, people, motorbikes. It has one training dataset of 684 images and two test sets (test set 1: 689 images, test set 2: 956 images). We use test set 1 in our experiments since many recent works evaluate their approaches on this test set. Figure 1 shows the image examples of PASCAL dataset.

In TRECVID-2006 dataset, the training and testing sets consist of 61,901 and 79,484 video shots respectively. In the experiments, we use the 20 semantic concepts which are selected in TRECVID-2006 evaluation [23]. The class labels of training set are provided by LSCOM [1]. We use one keyframe per shot for experiments. Figure 2 shows the keyframes of the 20 semantic concepts. These concepts cover a wide variety of types, including objects, indoor/outdoor scenes, people, events, etc. Note that this dataset is a multi-label dataset, which means each keyframe may belong to multiple classes or none of the classes, e.g. the example of *weather news* in Figure 2 also belongs to concept *map*. Compared with PASCAL, TRECVID dataset is more diverse and represents the real world scenario in the sense that the videos are from broadcast footage without any manual selection.

The TRECVID and PASCAL represent two very different datasets which are popularly and respectively used by the multimedia and computer vision community. By conducting experiments on these datasets, we expect to have a more insightful and convincing conclusion on BoF.

In the experiments, we use k-means algorithm to generate separate visual vocabularies for the two datasets. In PASCAL, we use all the keypoint features in the training set for clustering. In TRECVID, we subsample the training set and cluster 80k features. While there is a issue of data dependent vocabulary versus universal vocabulary, we do not plan to elaborate this challenging question in this paper due to space limitation. With the vocabularies, a two-class SVM classifier is trained for every object class (semantic concept).

4.2 Evaluation Criteria

The PASCAL and TRECVID communities use two different criteria, equal error rate (EER) and inferred average precision (InfAP) respectively, for performance evaluation. To make our experimental results comparable to both communities, we use ERR for PASCAL and InfAP for TRECVID. The EER is a point on the Receiver Operating Characteristic (ROC) curve, which measures the accuracy at which the number of false positives and false negatives are equal [7]. The InfAP is an approximation of the conventional average

Table 1: The mean EER of object classification tasks on PASCAL-2005 dataset using different keypoint detectors. We use *tf* for feature weighting, 1,000 visual keywords, and χ^2 RBF kernel.

Detector	Mean EER	Average # of keypoints per image
Harris Laplace	0.904	859
Harris Affine	0.871	778
Hessian Laplace	0.912	925
Hessian Affine	0.897	777
LoG	0.908	807
DoG	0.925	753

Table 2: The mean EER of object classification on PASCAL-2005 dataset using different weighting schemes and vocabulary sizes. The best result at each vocabulary size is shown in bold.

Vocabulary size	Weighting schemes			
	binary	*tf*	*tf-idf*	*soft-weighting*
500	0.906	0.924	**0.928**	0.922
1,000	0.913	0.925	0.929	**0.931**
2,000	0.916	0.913	0.914	**0.935**
5,000	0.921	0.917	0.913	**0.931**
10,000	0.904	0.902	0.908	**0.927**

Table 3: The mean InfAP of semantic video retrieval on TRECVID-2006 dataset using different weighting schemes and vocabulary sizes. The best result at each vocabulary size is shown in bold.

Vocabulary size	Weighting schemes			
	binary	*tf*	*tf-idf*	*soft-weighting*
500	0.048	0.088	0.081	**0.110**
1,000	0.076	0.082	0.078	**0.105**
5,000	0.082	0.083	0.089	**0.100**
10,000	0.083	0.090	0.096	**0.111**

precision (AP). The main advantage of InfAP is that it can save lots of judging effort during the annotation of ground-truth for large dataset [23]. Following the TRECVID evaluation, the InfAP is computed over the top 2,000 retrieved shots.

4.3 Comparison of Keypoint Detectors

Our first experiment aims to evaluate different keypoint detectors in the context of object categorization. We only experiment with PASCAL dataset since detecting different kinds of keypoints on the large TRECVID dataset is time-consuming. To make fair comparison, we adjust the cornerless threshold in each detector to detect an average of 750-950 keypoints per image. However, please note that even when the thresholds are set to zero, the sparse keypoint detectors can only return limited number of keypoints. The results are shown in Table 1 in terms of mean EER. As shown in Table 1, DoG achieves the best performance, followed by Hessian Laplace and LoG. Overall, the DoG and LoG perform well on this dataset, possibly due to the fact that they extract blob-like regions, while the others extract corner-like regions, which mainly lie around objects. An evaluation by Zhang et al. [25] shows that the background scene information contains lots of discriminative information on this dataset. Thus, we conclude that the blob-like region extractors maybe better than corner-like ones in the sense to represent background scene, so as to generate more discriminative features.

Next, we move on to examine the impact of affine adaption on Hessian Laplace and Harris Laplace. As shown in Table 1, both Hessian Laplace and Harris Laplace indeed win a large margin over Hessian Affine and Harris Affine, respectively. One possible reason is that the normalization process of affine adaption process may lose discriminative information. On the other hand, the affine transform may be rare in the real world applications. Since the affine adaption is done on the original -Laplace detectors (detailed in Section 3.1), and it will filter out a number of keypoints. In order to examine whether the quantity of keypoints is the one reason for the less satisfactory results of -Affine detectors, we lower the thresholds of both -Affine detectors to extract more keypoints. We find that Hessian Affine needs as high as 1,135 keypoints per image to generate a mean EER of 0.911, which is still a little bit lower than Hessian Laplace. This indicates that affine adaption of the two detectors is indeed not as good as the original -Laplace detectors.

It is also worth to note that, since different detectors extract keypoints with different properties, the keypoint detectors are complementary to some extent. However, the number of keypoints basically doubles when two detectors are used. Using keypoints from multiple detectors could greatly affect the processing time. For large dataset, such as TRECVID which contains more than 140,000 shots (keyframes), the speed is especially critical. For this reason, we only adopt DoG in the remaining experiments on the TRECVID dataset. By using the same cornerless threshold, the average number of DoG keypoints per keyframe in the TRECVID dataset is only 235. This is simply because the size of the keyframes in TRECVID dataset is smaller than that of the images in PASCAL. Note that DoG performs the best on the PASCAL dataset does not mean that it is only good for object modeling. Indeed DoG is also good for scene representation as demonstrated in our experiment that it can detect background scene information which is shown to be discriminative for object categorization in PASCAL. Hence, based on the nice properties observed on this small dataset, we expect that DoG is also a good choice for the more diversified TRECVID dataset.

4.4 Weighting Schemes and Vocabulary Sizes

In this section, we examine the keyword weighting schemes, vocabulary sizes, and study their relationships. We use DoG as keypoint detector and χ^2 RBF kernel for SVM learning. The results on PASCAL and TRECVID dataset are summarized in Table 2 and Table 3 respectively.

First, let us evaluate the influences of different weighting schemes. Our soft-weighting outperforms the other popular weighting schemes across different vocabulary sizes on both datasets. This indicates that the visual words, unlike traditional text words, are indeed correlated to each other and such correlation needs to be considered in feature representation. For that reason, our soft-weighting method which is

Table 4: The classification performances on both datasets using SVM with different *kernels*. The best results are given in bold. (Note that the evaluation metrics are different on the two datasets)

	PASCAL (mean EER)	TRECVID (mean InfAP)
Linear	0.874	0.041
HI	0.909	0.052
Gaussian RBF	0.892	0.075
Laplacian RBF	**0.921**	**0.087**
Sub-linear RBF	**0.922**	**0.084**
χ^2 RBF	**0.925**	**0.083**

tailored for the feature weighting of visual words performs much better. Next, we move on the see the relationship between *binary* and *tf*. We see that *tf* outperforms *binary* by a large margin only when the vocabulary size is small. This is due to the fact that, with a larger vocabulary size, the count of most visual keywords is either 0 or 1 and thus *tf* features are similar with *binary* features.

The *idf*, which weights visual words according to their distribution among the images, is only slightly useful in some of our experiments. We observe that the impact of *idf* is sensitive to vocabulary size. The is not surprising because a frequent visual word (cluster) may be split into several rare words (clusters) when increasing the vocabulary size. Thus the *idf* weight of a certain keypoint is not stable at all.

Finally, let us examine the impact of different vocabulary sizes. While using *binary* weighting, we observe that an appropriate size of vocabulary is around 5,000 for PASCAL dataset, and 10,000 (or larger) for TRECVID dataset. This is reasonable, due to the fact that TRECVID dataset is much more diversified than PASCAL dataset, and thus should contain more visual words. Another interesting observation is that when more sophisticated weighting schemes are employed, the impact of vocabulary size turns to be insignificant, especially for our soft-weighting method. Our explanation on this is based on the virtue of soft-weighting scheme discussed in Section 3.3. For this reason, we did not try any larger vocabulary size on both datasets.

4.5 Kernel Choice

In this experiment, we investigate the impact of different kernel choices for the BoF based visual classification. We use *tf* weighting on one thousand (1,000-d) vocabulary and five thousands (5,000-d) vocabulary for PASCAL and TRECVID respectively. Table 4 summarizes the results. The results of other weighting and vocabulary choices are similar. For the generalized RBF kernels, we adjust the parameter ρ in a reasonably range and choose the best one by cross validation. Overall, the generalized RBF kernels perform better than Linear kernel and HI kernel with non-trivial margin. This is probably due to the fact that visual words are correlated to each other, and are not linearly separable.

Among all the generalized RBF kernels, the χ^2 RBF kernel, Laplace RBF kernel, and sub-linear RBF kernel consistently outperform the traditional Gaussian RBF kernel. This can be attributed to the responses of kernels to background variance. Ideally, a kernel should only emphasize

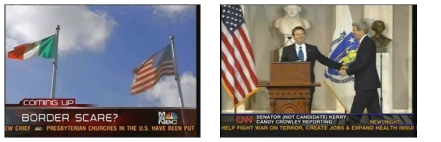

Figure 3: Instances of *US Flag* with different backgrounds in TRECVID dataset.

regions containing the target concept, while tolerating the background variance without amplifying the effect. Take Figure 3 as an example, intuitively one only perceives the common region (flag) when comparing their relevancy to the concept *US Flag*. An ideal kernel should thus reduce the impact of backgrounds. With reference to Figure 3, suppose there is a bin (visual word) representing people. This bin should have a nonzero weight w for the keyframe I_1 on the right hand side, but its weight is zero for the other keyframe. The responses of different kernels at this particular bin are:

$$K_{sub-linear}(I_1, I_2) = e^{-\rho|w-0|^{0.5}} = e^{-\rho w^{0.5}}$$
$$K_{Laplacian}(I_1, I_2) = e^{-\rho|w-0|} = e^{-\rho w}$$
$$K_{\chi^2}(I_1, I_2) = e^{-\rho\frac{(w-0)^2}{w+0}} = e^{-\rho w}$$
$$K_{Gaussian}(I_1, I_2) = e^{-\rho(w-0)^2} = e^{-\rho w^2}.$$

The kernel has a sub-linear exponential decay in sub-linear case, while it has linear exponential decay in the Laplacian and χ^2 cases, and quadratic exponential decay in the Gaussian case [6]. An ideal distance function should give small response (or equivalently a larger kernel response). Thus the kernels with linear/sub-linear exponential decay appear as better choices than the Gaussian RBF kernel.

Among different kernel choices, the computational time of linear kernel and HI kernel is shorter than that of the generalized RBF kernels. The sub-linear RBF kernel is the slowest since it contains a time-consuming square root for nonzero components of every support vector. For BoF representation, as shown in our experiments, we suggest to use kernels with linear exponential decay, i.e. the Laplace RBF kernel or the χ^2 RBF kernel. In our following experiments, χ^2 RBF kernel is employed.

4.6 Fusion with Color/Texture Features

Global features such as color and texture are popularly used in image and video classification. While keypoints are extracted from the grey level images and do not contain any color information, global features are statistics about the overall distribution of visual information. In this experiment, we investigate the complementary power of BoF when fused with color/texture features. We only choose the TRECVID dataset for evaluation, simply because the keypoint based features already saturate the performance on the PASCAL dataset.

We examine the fusion of BoF with two types of global features: color moment (CM) and wavelet texture (WT). In CM, we calculated the first 3 moments of 3 channels in *Lab* color space over 5×5 grid partitions, and aggregate the features into a 225-d feature vector. For WT, we use 3×3 grids and each grid is represented by the variances in 9 Haar wavelet sub-bands to form a 81-d feature vector.

The combination of different features is done by "late fu-

Table 5: Performance of fusing BoF with color moment (CM) and/or wavelet (WT) on TRECVID dataset. The 2nd column indicates the size of vocabulary, while the columns 4-6 show the mean InfAP performances varying with respect to the fused features and vocabulary size. The percentage in the parenthesis shows the degree of improvement over BoF only feature (3rd column).

			Global features		
			CM	WT	CM+WT
			0.076	0.031	0.100
Local feature	500-d	0.110	0.147 (34%)	0.106 (-4%)	0.155 (41%)
	1,000-d	0.105	0.149 (42%)	0.107 (2%)	0.156 (49%)
	5,000-d	0.100	0.147 (47%)	0.106 (6%)	0.155 (55%)
	10,000-d	0.111	0.152 (37%)	0.111 (0%)	0.158 (42%)

Table 6: Performance comparison on PASCAL-2005 dataset.

	Our BoF		PASCAL challenge 2005		Nowak et al. [17]
	DoG	DoG+Hessian Laplace	Zhang et al. [25]	Larlus et al. [10]	
Mean EER	0.931	0.947	0.928	0.946	0.954

sion", i.e. the final decision is made by fusing of the outputs of separate classifiers. Generally, the raw output of SVM in Eqn 2 can be used as detector response. We prefer the Platt's method [5, 20] to convert the raw output into a posterior probability. This is more reasonable especially for multi-modality fusion, since the raw outputs of SVM for different modalities may result in different scales, which will make the feature with larger scale dominating the others. In our experiments, we use "average fusion" to combine different feature channels.

Table 5 shows the fusion performance. The results show that BoF (with the best possible choices of detector, weighting scheme and kernel) outperforms CM, WT and their combination. This indeed proves the effectiveness of local features, even though they contain no color information. By fusing BoF with color feature, the performance is improved by around 40% over BoF only feature. The improvement from the fusion of BoF and WT is not as high as that of fusing BoF and CM. This is firstly because the WT itself is not as good as CM, and most importantly because both of them describe the textural information of the images (of course, from local and global point of view respectively). Overall, the best performance is attained when fusing the three features together. The results demonstrate that BoF is indeed highly complementary to these global features, and fusion should be used for good performance.

4.7 Performance Comparison

With careful selection of detector (DoG), soft-weighting scheme, and χ^2 RBF kernel, BoF alone indeed exhibits excellent performance on both datasets. In this section, we further compare and analyze the performance of BoF with the state-of-the-art techniques.

For PASCAL dataset, we compare our results with that of Zhang et al. [25], Larlus et al. [10], and Nowak et al. [17], which is to our knowledge the best reported results on this dataset. [25] adopted sparse sampling, while [10, 17] employed dense sampling. Note that [10] is the winner of PASCAL Challenge 2005. Table 6 summarizes the performance comparison. Zhang et al. used two local detectors (Harris Laplace and Laplacian of Gaussian), 1,000-d visual keywords, *tf* weighting, and χ^2 kernel for SVM. As shown in the table, our results by DoG detector with soft-weighting scheme is already better than that of Zhang et al., which is the best reported result with sparse sampling in the PASCAL challenge. We further combine keypoints from our two best detectors (DoG and Hessian Laplace), and fuse the two groups of keypoints in the χ^2 kernel with the approach in [25]. With this setting, our BoF already outperforms the winner of PASCAL with EER as high as 0.947. Although the performance is still lower than that of Nowak [17] who adopted a dense set of multi-scale local patches, our sparse BoF has the advantage of speed efficiency. The number of local patches used by Nowak [17] is large (10,000 per image). This will significantly increase the computational time, and thus prohibit the scalability of this approach to larger dataset such as TRECVID. In our approach, even by a combination of two detectors, the average number of keypoints per image is only 1,678 which is much lower than [17].

For TRECVID-2006 dataset, we first compare our BoF to the local feature approaches of Berkeley [19] and Mediamill [22]. The results are shown in Table 7. Compared to Berkeley who adopted point-to-point matching of local features, our BoF performs slightly better. Note that although Berkeley used exemplars to avoid online point-to-point matching with every training examples, the number of keypoint comparison per test sample is still as high as $51{,}640{,}000$ $(200 \times 200 \times 1,291)$, where there are 200 sampled points and 1,291 exemplars. While for our BoF, the number of keypoint comparison for one keyframe is only 2,350,000 $(235 \times 10,000)$ for a vocabulary size of 10,000. The BoF of Mediamill used late fusion to combine differnt keypoint detectors and descriptors. However, our results show that, by simply using DoG and SIFT, a single run (rather than fusion of different detectors/descriptors) with well representation achieves a mean InfAP of 0.111, which already doubles that of Mediamill.

We conclude this section by comparing our results with the best results of the top 3 teams in TRECVID-2006 evaluation [23]. As shown in Table 7, our best results using only 3 visual features are comparable to CMU [9] and IBM [3]. CMU used both visual (color, texture, BoF) and text

Table 7: Performance comparison on TRECVID-2006 dataset.

	Our results		Local feature systems in TRECVID'06		Best of TRECVID'06		
	BoF	Global + BoF	Mediamill (Run 5)	Berkeley (Run 2)	CMU	IBM	Tsinghua
Mean InfAP	0.111	0.158	0.055	0.110	0.159	0.177	0.199

features, while IBM used global and localized color and textures, motion features, as well as text. Compared to Tsinghua [4] who emphasizes rich features and rich classifiers, our method is obviously more efficient and can be easily scaled up to a thousand of semantic concepts.

5. CONCLUSION

We have investigated various factors in BoF for object categorization and semantic video retrieval. By jointly considering the choice of keypoint detector, vocabulary size, weighting scheme, and kernel, the BoF shows surprisingly strong performance regardless of the orderless and colorless representation.

We have shown that all the four investigated factors are influential to the performance of BoF. The vocabulary size, however, exhibits less or even insignificant impact when our proposed soft-weighting scheme is in use. This indeed motivates and verifies the need of a weighting scheme specifically for visual words to alleviate the impact of clustering on vocabulary generation. Our experiments also demonstrate that the BoF is highly complementary to the global features. By incorporating the BoF with two global visual features, the performance is already competitive enough to the best few systems in TRECVID-2006 evaluation, while enjoying the merit of simplicity and efficiency.

There is still room for further improvement of BoF. One interesting direction is to use the geometric blur [2] as keypoint descriptor. This is first motivated by the fact that the local patches used in [19] are not invariant to scale. Secondly, the SIFT descriptor is easily suffered from the quantization effect when dividing the regions around the keypoints into fixed grids.

Acknowledgements

The work described in this paper was fully supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 118906).

6. REFERENCES

[1] LSCOM lexicon definitions and annotations. In *DTO Challenge Workshop on Large Scale Concept Ontology for Multimedia, Columbia University ADVENT Technical Report #217-2006-3*, 2006.

[2] A. C. Berg and J. Malik. Geometric blur for template matching. In *IEEE CVPR*, 2001.

[3] M. Campbell et al. IBM research trecvid-2006 video retrieval system. In *TRECVID*, 2006.

[4] J. Cao et al. Intelligent multimedia group of Tsinghua university at trecvid 2006. In *TRECVID*, 2006.

[5] C. C. Chang and C. J. Lin. LIBSVM: a library for support vector machines. *Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm*, 2001.

[6] O. Chapelle, P. Haffner, and V. N. Vapnik. Support vector machines for histogram-based image classification. *IEEE Trans. on NN*, 10(5), 1999.

[7] M. Everingham et al. The 2005 pascal visual object classes challenge. In *LNAI*, volume 3944, pages 117–176. Springer-Verlag, 2005.

[8] C. Fowlkes, S. Belongie, F. Chung, and J. Malik. Spectral grouping using the nystrom method. *IEEE Trans. on PAMI*, 26(2), 2004.

[9] A. G. Hauptmann et al. Multi-lingual broadcast news retrieval. In *TRECVID*, 2006.

[10] D. Larlus, G. Dorko, and F. Jurie. Creation de vocabulaires visuels efficaces pour la categorisation d'images. In *Reconnaissance des Formes et Intelligence Artificielle*, 2006.

[11] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE CVPR*, 2006.

[12] T. Lindeberg. Feature detection with automatic scale selection. *Int. J. of Computer Vision*, 30:79–116, 1998.

[13] D. Lowe. Distinctive image features from scale-invariant keypoints. *Int. Journal on Computer Vision*, 60(2):91–110, 2004.

[14] K. Mikolajczyk et. al. A comparison of affine region detectors. *Int. Journal on Computer Vision*, 65(1/2):43–72, 2005.

[15] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *Int. Journal of Computer Vision*, 60:63–86, 2004.

[16] A. Agarwal, and B. Triggs. Hyperfeatures – multilevel local coding for visual recognition. In *ECCV*, 2006.

[17] E. Nowak et al. Sampling strategies for bag-of-features image classification. In *ECCV*, 2006.

[18] F. Odone et al. Building kernels from binary strings for image matching. *IEEE Trans. on IP*, 14(2), 2005.

[19] S. Petrov et al. Detecting categories in news video using acoustic, speech, and image features. In *TRECVID*, 2006.

[20] J. Platt. Probabilities for SV machines. In *Advances in Large Margin Classifiers*, pages 61–74, 2000.

[21] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, 2003.

[22] C. G. M. Snoek et al. The mediamill trecvid 2006 semantic video search engine. In *TRECVID*, 2006.

[23] TREC Video Retrieval Evaluation (TRECVID). *http://www-nlpir.nist.gov/projects/trecvid/.*

[24] V. Vapnik. *The Nature of Statistical Learning Theory.* New York: Springer-Verlag, 1995.

[25] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: An in-depth study. In *INRIA Technical Report RR-5737*, 2005.