Noname manuscript No.
(will be inserted by the editor)

Improving Image Classification using Semantic Attributes

Yu Su · **Frédéric Jurie**

Received: date / Accepted: date

Abstract The Bag-of-Words (BoW) model – commonly used for image classification – has two strong limitations: on one hand, visual words lack semantic meanings, on the other hand, they are often polysemous. This paper proposes to address these two limitations by introducing an intermediate representation based on the use of semantic attributes. Specifically, two different approaches are proposed. Both approaches consist of predicting a set of semantic attributes for the entire images as well as for local image regions, and in using these predictions to build the intermediate level features. Experiments on four challenging image databases (PASCAL VOC 2007, Scene-15, MSRCv2 and SUN-397) show that both approaches improve performance of the BoW model significantly. Moreover, their combination achieves state-of-the-art results on several of these image databases.

Keywords image classification · bag-of-words model · semantic attribute · visual words disambiguation

1 Introduction

Image classification, including object and scene classification, is a central area in computer vision research. Among the recent advances made in this field, the most significant one is perhaps the representation of images by statistics of local features, in particular through the introduction of histograms of textons [21] and the bag-of-words (BoW) model [4,32] which is borrowed from natural language processing. In the BoW model, local features extracted from images are first mapped to a set of visual words obtained by vector quantizing the feature descriptors (e.g. with k-means). An image is then represented as a histogram of visual words occurrences. Combined with some powerful classifiers such as the Support Vector Machine (SVM), the BoW model has demonstrated impressive performances on several challenging image classification tasks [7,12,40].

As it is presented above, the BoW model suffers from two strong limitations. First, despite the fact that visual words are more meaningful than single pixels, they still lack explicit semantic meanings. Indeed, extracting semantic features is a very important characteristic of the human visual system. Humans learn about new object categories by using existing knowledge of visual categories, which is often encoded as high-level semantic attributes [28]. For example, if a new animal is seen, it can be connected to some previously learned concepts (e.g. *grey, head, hooves and wings*) which can be used to recognize this animal. Besides colors and object parts, this kind of shared semantic attributes might describe common scenes properties (e.g. *road*), common shapes (e.g. *box*) and common materials (e.g. *wood*). Second, like written words of natural languages, visual words are also frequently polysemous, i.e. the same visual word may have different meanings. As a simple example, we can imagine that two local features which represent similar image structures (e.g. windows) are assigned to the same visual word, one being sampled from a 'car' while the other is sampled from a 'plane'.

In this paper, we propose to address the above mentioned limitations of the BoW model by (a) predicting semantic attributes for both entire images and image regions (illustrated in Fig. 1) and (b) using them as additional information for

Yu Su
GREYC - UMR6072 CNRS , University of Caen, France
Tel.: +33-231567434
Fax: +33-231567330
E-mail: yu.su@unicaen.fr

Frédéric Jurie
GREYC - UMR6072 CNRS, University of Caen, France
Tel.: +33-231567434
Fax: +33-231567330
E-mail: frederic.jurie@unicaen.fr

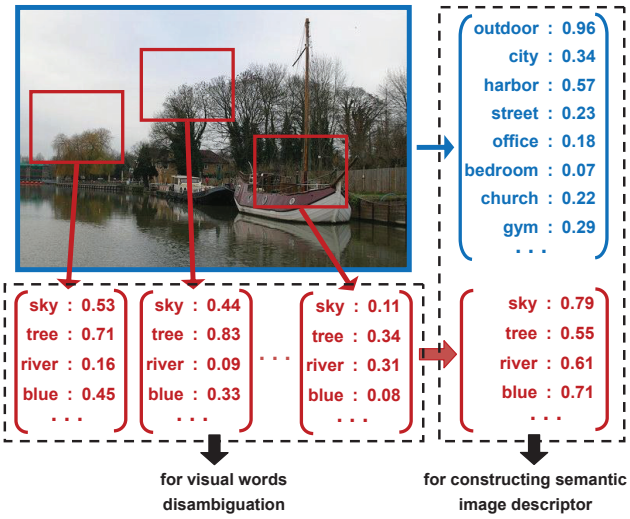


Fig. 1 Illustration of semantic attribute prediction. For the attributes which describe the global properties of images (e.g., *outdoor*, *city*, etc.), the attribute classifiers are applied to entire images. For the attributes which describe the local characteristics of images (e.g. *sky*, *tree*, etc.), the attribute classifiers are applied to a set of image regions. The figure is better viewed in color.

the BoW model. Specifically, we train a set of classifiers for individual visual semantic attributes – whose list has been manually specified – from BoW features, and use them to make predictions on new images or image regions. We then use the outputs of these classifiers as a low-dimensional image descriptor which has explicit semantic meanings. The performance of this semantic descriptor alone is close to that of much higher dimensional BoW histogram, while the combination of both consistently improves their performances. As to the problem of visual word disambiguation, we propose two methods to utilize the *contexts* which is defined as the occurrence probabilities of a set of semantic attributes on entire images or image regions. In the first method, a single vocabulary is learned from local features (e.g. SIFT) for all contexts (attributes). Then we select one context for each visual word to reduce its ambiguity. In the second method, multiple vocabularies are learned from local features, each of which corresponds to a single context. Visual words in these context-specific vocabularies are less ambiguous than those in the universal vocabulary. For a specific classification task, only the relevant contexts are selected, resulting in a low dimensional final image descriptor.

The organization of this paper is as follows: In Section 2, we review the related works on semantic attributes, semantic vocabulary and visual word disambiguation. Then, we explain how we utilize semantic information to construct image descriptors with explicit semantic meanings (Section 3) and then how we disambiguate visual words (Section 4). Experiments and results are conducted in Section 5, followed by conclusions and discussions in the last section.

2 Related Works

The recent literature abounds with approaches making interesting use of visual semantic attributes and giving proofs-of-concept. Roughly speaking, these methods can be divided into two categories. One is representing objects or images by vectors of semantic attributes, which are usually in a much lower dimensional space than BoW histograms. The other is learning semantic vocabularies that are more discriminative than traditional ones (e.g. those computed by k-means). In the following, a comprehensive review of these methods is given. Besides, we also review the methods related to visual words disambiguation.

Visual semantic attributes. Farhadi *et al.* [8] were among the first to propose to use a set of visual semantic attributes such as *hairy* and *four-legged* to identify familiar objects, and to describe unfamiliar objects when new images are provided. At the same time, Lampert *et al.* [19] showed that high-level descriptions in terms of semantic attributes can also be used to recognize object classes without any training image, once the semantic attribute classifiers have been trained from other classes of data. Kumar *et al.* [18] have also proposed to describe faces by vectors of visual attributes (e.g., *gender*, *race*, *age*, *hair color*) which are predicted by using corresponding attribute classifiers.

In addition to describing objects semantically, several works described the whole image by semantic features, for image retrieval or image classification tasks. Vogel and Schiele [37] used visual attributes describing scenes to characterize image regions and combined these local semantics into a global image description, used for the retrieval of natural scene images. Wang *et al.* [38] proposed to represent images by their similarities with Flickr image groups which have explicit semantic meanings, and showed that these semantic features give similar or even better performance than pure visual features, for different image classification tasks. Torresani *et al.* [35] used the outputs of a large number of object category classifiers to represent images and showed good performances for both image classification and image retrieval tasks. A similar idea was also adopted by [22], in which an image is represented as the localized outputs of object detectors. In these methods, classifiers are trained for each individual semantic attribute and the classifier outputs are used to represent images. Besides using attribute classifiers, some researchers proposed to utilize the hierarchical structure of semantic attributes to represent images [23] or measure the similarity of images [6]. For example, Li *et al.* [23] built a semantically meaningful image hierarchy by using both visual and semantic information, and represent images by the estimated distributions of concepts over the entire hierarchy. Deselaers and Ferrari [6] represented an image by the labels of its nearest neighbors in ImageNet dataset

and measured the semantic similarity of two images through ImageNet hierarchy.

In this work, we also use semantic classifiers to describe images. However, we additionally propose to use the semantic attributes to disambiguate visual words in the BoW framework.

Semantic vocabulary. Several attempts have been made to embed semantic information into the vocabulary. In [37], Vogel and Schiele proposed to manually assign to each image region a semantic label (e.g. *sky*, *water*, *grass*), and then constructed a semantic vocabulary based on these labeled image regions. The visual words in this vocabulary have explicit semantic meanings. However, the manual labeling prevents to use this method in large-scale applications. In [24], Liu *et al.* proposed a two-steps procedure to construct semantic vocabularies. First, visual words (also called *mid-level features*) are obtained by vector quantizing the local features (using k-means), as in the traditional BoW model. Second, mid-level features are embedded into a lower dimensional semantic space using diffusion maps and then clustered again by k-means to obtain a semantic vocabulary. In [15], Ji *et al.* considered both visual and semantic similarities of local features. The semantic similarities of local features are learned from 60,000 labeled Flickr images as well as the correlation of image labels provided by WordNet. In addition, the methods based on topic models such as *Probabilistic Latent Semantic Analysis (pLSA)* [1,29] or *Latent Dirichlet Allocation (LDA)* [9,31] represent an image as a mixture distribution of hidden topics which are more related to meaningful concepts than the visual words.

The above mentioned works utilize either additional semantic annotation of images [15,37] or manifold structure of mid-level feature space [1,9,24,29,31] to learn the more semantic meaningful vocabulary. Our method bears similarities with the former, but our aim is not only to learn semantic meaningful vocabulary but also to make the visual word less ambiguous (and therefore more discriminative) which is more important for image classification tasks.

Visual words disambiguation. To deal with synonymy and polysemy, one solution is to eliminate the most frequent words which are supposed to be the most ambiguous ones, as proposed in [32]. Another solution is to utilize task-specific information: as an example, supervised learning methods can be used to obtain *category-specific* vocabularies [25]. In addition, Yuan *et al.* [42] combined the spatially co-occurrent visual words to form *visual phrases*, which usually have higher level meanings and therefore are less ambiguous. A similar idea was also presented in [44].

Synonymy can be caused by the quantization process used to obtain the visual vocabulary. Indeed, the hard assignment of the standard BoW model can lead to large loss of information if some visual words have close representations. To address this problem, soft assignment in which a local feature is assigned to different number (including zero) of visual words was proposed [11] and can help to alleviate synonymy.

Polysemy of visual words is partly due to the discard of spatial information. Hence, the use of spatial information can help to disambiguate visual words. A typical example is the well-known spatial pyramid matching [20], in which multiple histograms are constructed from increasing finer sub-regions and then concatenated to give the image representation.

Topic models, such as the *Probabilistic Latent Semantic Analysis (pLSA)* [14], also address polysemy [30]. For example, both the topics of 'bird' and 'equipment' can give high probability to the word 'crane', but the occurrence probabilities of different topics reduce this uncertainty. In contrast to the topic model, our method uses semantic contexts rather than topics learned from data collection. Please refer to Section 4 for more details.

As context plays a major role in the disambiguation of natural language words, our opinion is that it can be also useful for visual word disambiguation. In [5], the foreground (object of interest) and background are modeled separately, resulting in two BoW histograms which are combined by summing the corresponding kernels. In [36], videos are decomposed into regions with different semantic meanings, from which multiple region-specific BoW histograms are computed and concatenated. Both [5] and [36] showed promising results on action recognition tasks. The differences between our method and them are twofold. First, in our method, BoW histograms are context-specific rather than region-specific. Second, our method compresses multiple histograms rather than computing multiple kernels for them [5] or concatenating them [36], resulting therefore in a more compact image representation.

In another related work, Khan *et al.* [16] proposed to use some category-specific color attention maps to weight local shape features and then concatenate multiple histograms. Our method for visual word disambiguation also uses the idea of weighting local features. However, we adopt semantic contexts (rather than color) to generate attention maps and reduce the dimension of final image descriptors by selecting the relevant contexts (rather than concatenate all histograms). In [40], four geometry contexts (*ground*, *vertical*, *porous* and *sky*) were adopted to build the *geometry specific histograms*. Different from it, our method uses much more contexts and combines multiple context-specific histograms by context selection rather than concatenating them as in [40]. Experiments in Section 5.4 show that our method performs better than the geometry specific histograms.

Finally, compared with our previous works on semantic attributes [33,34] corresponding to Section 3 and 4.1 re-

spectively, this paper makes three extensions. First, we extend the method for visual word disambiguation described in [34] by learning a specific vocabulary for each context and selecting contexts for each classification task by simulated annealing (see Section 4.2). Second, we give a more comprehensive review of the semantic-related methods for image classification. Third, we give more experimental results to validate the benefit of using semantic information for image classification.

3 Image Representation by Semantic Attribute Features

In this work, six groups of visual semantic attributes are introduced to cover the spectrum of (1) *global scenes* (e.g. *train station*, *bedroom*), (2) *local scene elements* (e.g. *sky*, *tree*), (3) *color* (e.g. *green*, *red*), (4) *shape* (e.g. *box*, *cylinder*), (5) *material* (e.g. *leather*, *wood*) and (6) *object* (e.g. *face*, *motorbike*). It makes a total of 110 different attributes. We define these semantic attributes by hand with the intention of providing abundant semantic information for image description. Fig. 2 gives the full list of semantic attributes and some typical images. These semantic attributes can be divided into two categories. The attributes in the group of *global scene* (group 1) describe the characteristics of whole images we refer to them as *global attributes*, while the attributes in other groups (groups 2 to 6) describe the characteristic of image regions we refer to them as *local attributes*.

We learn a set of independent attribute classifiers (SVMs with Battacharyya kernel), each of which corresponds to a semantic attribute, and use them to construct semantic image descriptors. For global attributes, the classifiers are learned on whole images described by BoW histograms. For local attributes, the classifiers are learned from some randomly sampled image regions described again by BoW histograms. In the training process, the label of a region is the same as the label of the image from which it is sampled. In practice, the Battacharyya kernel is implemented by square-rooting BoW histograms before training linear SVMs (the equivalence was proved in [27]). Using more complex kernels (e.g. chi-square [3]) does not significantly improve the accuracy of attribute classifiers and the performance of resultant semantic image descriptor (see Fig.7).

As to the training images, there are two cases. For the semantic attributes that appear in PASCAL 2007 and Scene-15 databases (e.g. *motorbike*, *bedroom*), the training images as well as the annotations are directly obtained from the training images of these databases. For other semantic attributes, training images are automatically downloaded from *Google image search* by using the name of attribute as query. We manually reject the irrelevant images, leaving about 400 relevant images for each attribute. When training a classifier for

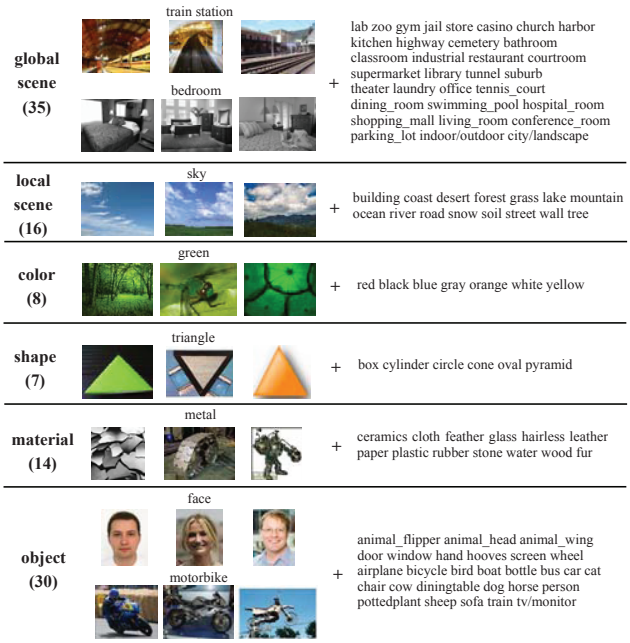


Fig. 2 Semantic attributes, grouped by type, including some illustrative training images. The values in parentheses are the number of semantic attributes within corresponding groups. In this paper, the attributes of global scene are refereed as *global attributes*, while the attributes of local scene, color, shape, material and object are referred as *local attributes*.

a given attribute, the images of this attribute are considered as positive samples. The images of other attributes within the same group are considered as negative samples. Take the attribute *wood* as an example; its images are used as positive samples, while the images of other materials are used as negative samples. However, there exist two exceptions: *indoor/outdoor* and *city/landscape*. For these two attributes, the images of *indoor* and *city* are used as positive samples respectively, while the representative images of *outdoor* and *landscape* are used as negative samples respectively.

Similar to the training process, there are also two cases in attribute prediction, when processing test images. For global attributes, the predictions are the result of running the attribute classifiers on the whole image. For local attributes, the predictions are generated by running the attribute classifiers on some randomly sampled image regions and then pooling the classifier outputs (see Fig. 1). We evaluated the performances of two pooling methods: average pooling which averages the classifier outputs of image regions and maximum pooling which assign to each context only the maximum score of image regions, and experimentally demonstrate that the average pooling performs better. It is worthwhile to point out that, in the prediction process, the classifier outputs are transformed into probabilities by sigmoid function (refer to [2]). An image is finally represented by a 110-D descriptor, each element of which can be consid-

ered as the occurrence probability of the corresponding semantic attribute. This image descriptor has two advantages compared with BoW histogram. First, it has explicit semantic meanings while BoW histogram does not. Second, its dimensionality is much lower than that of BoW histograms (usually up to several thousands). In the experiment section, we show that this semantic image descriptor performs close to BoW histogram. Furthermore, when combining it with BoW histogram, the performance always increases, which demonstrates that they are complementary to each other.

4 Visual Words Disambiguation by Semantic Contexts

As pointed out in the introduction, context plays a major role in the disambiguation of natural language words. By analogy, this motivates us to put a special emphasis on extracting contextual information from images with the idea of using it to disambiguate visual words. Here we use the local semantic attributes defined in the previous section to describe the local characteristics of image, which are referred as *semantic contexts*. In the following, we will introduce two methods to embed semantic contexts into BoW histogram and therefore reduce the ambiguity of visual words.

4.1 Context embedding with a single vocabulary

In this first method, a single vocabulary is learned from a set of local features (e.g. SIFT) which are extracted from image patches with randomly selected positions and scales. The main idea of our method for visual words disambiguation is illustrated in Fig. 3. Specifically, for an image, we construct multiple BoW histograms, each of which corresponds to a visual semantic context: in this case, a given visual word has different occurrence frequencies when different contexts are considered. For example, in Fig. 3, the occurrence frequency of the visual word denoted by square is higher in context *sky* than in *tree*, because this visual word often appears in sky area. By embedding contextual information, the visual words in each single histogram are less ambiguous. Considering the huge resulting dimensionality if these context-specific histograms were combined (e.g. concatenated), we propose to reduce the dimensionality by selecting only a single context for each visual word. The resultant histogram is called *context-embedded BoW histogram* (contextBoW-s for short) which has the same dimensionality as the standard BoW histogram. Here '-s' denotes 'single vocabulary' in order to distinguish with that of multiple vocabularies (introduced in the next subsection).

In the following, we first formulate the process of embedding semantic contexts into BoW model, and then introduce how to construct the context-embedded BoW histogram by using previously learned attribute classifiers (also referred as context classifiers).

4.1.1 Formulation of embedding contexts into BoW model

Let $\{f_i, i = 1,\ldots,N\}$ be the set of local features extracted from image I, where N is the total number of local features. The visual vocabulary consists of V visual words denoted by $\{v_j, j = 1,\ldots,V\}$. The traditional BoW feature, for v_j, measures the occurrence probability of v_j on image I, say $p(v_j|I)$. In practice, $p(v_j|I)$ is usually computed as the occurrence frequency of visual word v_j on image I by:

$$p(v_j|I) = \frac{1}{N}\sum_{i=1}^{N}\delta(f_i, v_j), \tag{1}$$

where

$$\delta(f_i, v_j) = \begin{cases} 1 \text{ if } j = \underset{j=1,\ldots,V}{\arg\min}\, d(f_i, v_j) \\ 0 \text{ else} \end{cases} \tag{2}$$

and d is a distance function (e.g. the Euclidean distance).

As mentioned in Section 1, a visual word can have different meanings in different contexts. Marginalizing $p(v_j|I)$ over different contexts gives:

$$p(v_j|I) = \sum_{k=1}^{C} p(v_j|c_k, I)p(c_k|I), \tag{3}$$

where c_k is the k-th context, C is the number of contexts, $p(v_j|c_k, I)$ is the context-specific occurrence probability of v_j on image I, and $p(c_k|I)$ is the occurrence probability of context c_k on image I.

Eq. (3) is similar to the formulation of *Probabilistic Latent Semantic Analysis (pLSA)* [14]. But different from pLSA, we do not assume the conditional independence that, conditioned on the context c_i, visual words v_i are generated independently from the specific image I, i.e. $p(v_j|c_k, I) \neq p(v_j|c_k)$. Instead, we believe that the words generated by a given context are characteristic signatures of the image. As an illustration, if for a particular image, a *window*-like visual word occurs simultaneously with the *blue* context, it could be a good cue for hypothesizing the presence of a plane in the image. Another difference from pLSA is that we do not consider contexts as latent variables, which we believe would be hard to estimate, but define them offline and predict them for every image by using the context classifiers.

It is worthwhile to point out that the second term of Eq. (3) ($p(c_k|I)$), which is equivalent to the semantic image

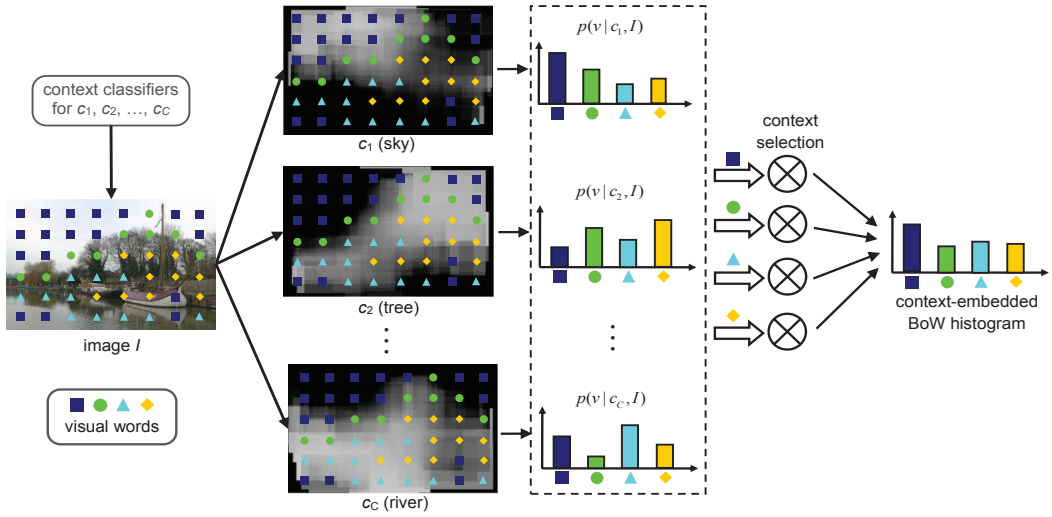


Fig. 3 Construction of context embedded BoW histogram. For an image, multiple probability maps are generated by the pre-learned context classifiers to measure the occurrence probabilities of corresponding contexts. Then, a BoW histogram is constructed for each context by weighting local features according to its probability map. Finally, a context selection process is used to choose a single context for each visual word and therefore result in a compact image descriptor. Note that in this method, the same vocabulary is used for all contexts.

descriptor (using here only the local attributes) proposed in Section 3, can also provide rich information to describe the image as shown by [37]. For example, knowing an image is composed of one third of *sky*, one third of *sea* and one third of *beach*, brings a lot of information regarding the content of this image. Thus, when classifying images, $p(v_j|c_k,I)$ and $p(c_k|I)$ are combined to take advantage of the complementary information embedded in them. In this work, the combination is performed at decision level, i.e. by training classifiers on $p(v_j|c_k,I)$ and $p(c_k|I)$ separately and then combining their scores (e.g. with the weighted sum rule, product rule or max rule). The detailed description of these combination rules can be found in [17].

4.1.2 Implementation of context-embedded BoW histogram

In this work, $p(v_j|c_k,I)$ is constructed by modeling the probabilistic distribution of context c_k on image I. In practice, the probabilistic distribution is estimated by randomly dividing image I into a set of regions and predicting the occurrence probabilities of c_k on these regions. By denoting $I_p(f_i) = \{g_l, l = 1,\ldots,L_i\}$ the set of image regions which cover the local feature f_i, we define:

$$p(v_j|c_k,I) = \frac{1}{N}\sum_{i=1}^{N}\delta(f_i,v_j)p(c_k|I_p(f_i)), \tag{4}$$

where $p(c_k|I_p(f_i)$ can be considered as the weight of local feature f_i. In practice, $p(c_k|I_p(f_i))$ is computed by averaging the outputs of ϕ_k (context classifier for c_k) on the regions within $I_p(f_i)$:

$$p(c_k|I_p(f_i)) = \frac{1}{L_i}\sum_{l=1}^{L_i}\phi_k(H(g_l)), \tag{5}$$

where $H(g_l)$ is the BoW histogram built on region g_l. Here the classifier outputs have already been transformed into probabilities (see Section 3).

Concatenating $p(v_j|c_k,I)$ for all visual words and contexts would lead to a $V \times C$-dimensional descriptor. In this work C is 75 (i.e. the number of *local attributes*) since only the *local attributes* are used to construct $p(v_j|c_k,I)$ while V is usually from hundreds to thousands. Training classifiers using this high dimensional descriptor would be very time-consuming especially when the non-linear kernel is used. Our intuition is that, for a given classification task, a visual word usually appears only within a limited set of contexts. For example, as in Fig. 3, the visual word denoted by square almost exclusively appears in the context *sky* and *river*. In practice, we show in Section 5 that using only one context per visual word already gives very good results. By doing that, for a given classification task, an image is finally represented by

$$[p(v_1|c_{k_1},I),\ldots,p(v_j|c_{k_j},I),\ldots,p(v_V|c_{k_V},I)],$$

where c_{k_j} is the selected context for visual word v_j and the given classification task. representation as context-embedded BoW histogram (contextBoW-s for short).

Up to now, the only remaining problem is how to choose a single context for each visual word (i.e. the c_{k_j}). This is a feature selection problem and in theory any criterion can

be used for that, e.g. max-likelihood. Although more consistent with the proposed probabilistic framework, the max-likelihood criterion does not allow the use of category labels of images and therefore performs worse than the supervised ones in practice. In this work, we adopt a supervised *t-test* based criterion. Specifically, for each visual word v_j and each context c_k, we assume that the value of $p(v_j|c_k,I)$ follows the Gaussian distribution $\mathcal{N}(\mu^+_{j,k},\sigma^+_{j,k})$ on positive images and $\mathcal{N}(\mu^-_{j,k},\sigma^-_{j,k})$ on negative images. It is worth pointing out that while the probability $p(v_j|c_k,I)$ is bounded between 0 and 1, we observe in experiments that its distribution is usually near-Gaussian. Thus, the assumption is approximately satisfied. For a given visual word, we compute the *t-test* score between these two distributions for every possible context and take the context giving the highest value:

$$k_j = \arg\min_{k=1,\ldots,C} \frac{(\mu^+_{j,k}-\mu^-_{j,k})^2}{{\sigma^+_{j,k}}^2+{\sigma^-_{j,k}}^2}. \quad (6)$$

It therefore selects the context for which the representation of positive images is as different as possible from the representation of negative images, i.e. the most discriminative context. As this context selection process is supervised, the selected context depend on the classification task to be addressed. That is to say, the context selected for 'aeroplane' classification and 'person' classification will be very different. The whole procedure of constructing contextBoW-s is summarized in Algorithm 1.

Algorithm 1 Construction of ContextBoW-s

Input: image I, visual vocabulary $\{v_j, j=1,\ldots,V\}$, local attribute classifiers $\{\phi_k, k=1,\ldots,C\}$
Extract a set of local features $\{f_i, i=1,\ldots,N\}$ from image I.
for $i=1,\ldots,N$ **do**
  Construct $I_p(f_i)=\{g_l, l=1,\ldots,L_i\}$ which is the set of image regions covering f_i.
  for $k=1,\ldots,C$ **do**
    Compute $p(c_k|I_p(f_i))$ by Eq. (5).
  end for
end for
for $j=1,\ldots,V, k=1,\ldots,C$ **do**
  Compute $p(v_j|c_k,I)$ by Eq. (4).
end for
for $j=1,\ldots,V$ **do** {offline context selection}
  for $k=1,\ldots,C$ **do**
    Compute $(\mu^+_{j,k},\sigma^+_{j,k})$ and $(\mu^-_{j,k},\sigma^-_{j,k})$ for a classification task.
  end for
  Select the context with the highest *t-test* score by Eq. (6)
end for
Output: ContextBoW-s $[p(v_1|c_{k_1},I),\ldots,p(v_j|c_{k_j},I),\ldots,p(v_V|c_{k_V},I)]$

4.2 Context embedding with multiple vocabularies

As above mentioned, another choice for visual word disambiguation is to learn a specific vocabulary for each semantic context. In this case, each visual word is learned within a given context and therefore is much less ambiguous. For example, if a *window*-like visual word is learned within the context *sky*, it is very likely to be a *plane window* rather than a *car window*, and will therefore be modeled more accurately. In the following, we will introduce how to learn context-specific vocabulary and construct compact image representations by selecting the most discriminative contexts for a specific classification task.

4.2.1 Learning context-specific vocabulary

In the traditional vocabulary learning process, local features extracted from a set of images are uniformly sampled (at random positions or regularly) and then vector quantized to get visual words. Differently, when learning our context-specific vocabulary, the sampling of local features is based on the distribution of this context on images. Specifically, more local features are sampled at the image regions with higher context-occurring probabilities (brighter image regions in Fig. 4). In practice, this process is implemented by assigning each local feature f_i a probability $p(c_k|I_p(f_i))$ (defined in Section 4.1.2) and sampling local features based on their probabilities, which is formulated as follows.

$$s(f_i) = \begin{cases} 1 \text{ if } p(c_k|I_p(f_i)) \geq r_i \\ 0 \text{ else} \end{cases} \quad (7)$$

where $s(f_i)$ indicates whether the local feature f_i is selected or not and r_i are random numbers which are uniformly sampled between 0 and 1.

After sampling local features for each context, k-means is used multiple times, to build one specific vocabulary per context. Finally, an image can be represented by multiple context-specific BoW histograms. The construction of context-specific BoW histogram is the same as that in 4.1.2 (see Eq. (4))

4.2.2 Context selection by simulated annealing

As in Section 4.1.2, concatenating all context-specific BoW histograms would lead to a $V \times C$-dimensional descriptor. Training a classifier based on such high dimensional descriptors would be very time consuming, especially when non-linear kernels are used. However, we can not perform context selection for each visual word, as introduced in Section 4.1.2, because the visual words are context specific

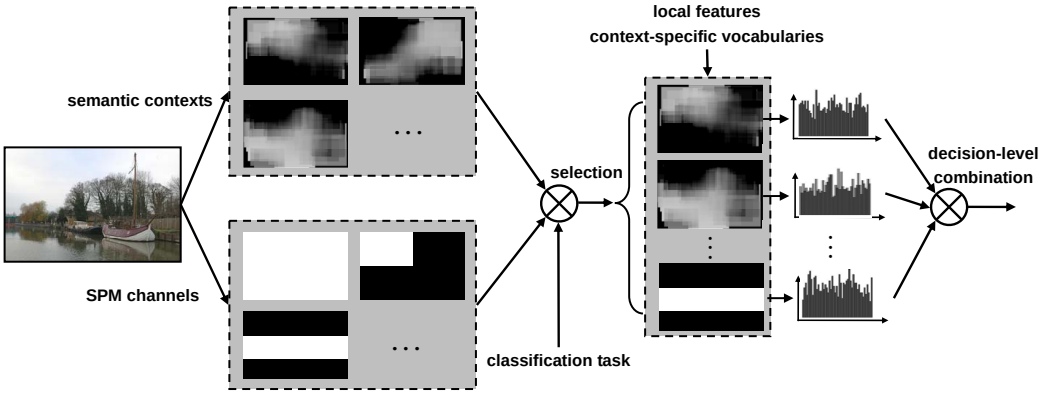


Fig. 4 Context selection from both semantic contexts and SPM channels. For an image, multiple probability maps are generated by both context classifiers and SPM channels, from which multiple BoW histograms are constructed. Then, a context selection process is used to choose a small number of the most discriminative contexts for a specific classification task. Finally, multiple BoW histograms are combined at decision level.

rather than unique for all contexts. In this work, we adopt a *divide and conquer* strategy to learn the final image classifier. More specifically, we train a classifier for each context based histogram, which is of much lower dimensionality, and then combine all the classifiers by averaging their outputs. The benefit of this strategy is also noted in [10].

Although the divide and conquer strategy effectively reduce the dimensionality of features used for each classifier, constructing multiple histograms and running multiple classifiers in test phase is very time consuming. Furthermore, for a specific classification task, the contexts are not equally important. For example, when classifying 'aeroplane', the context *sky* is much more useful than *building*. This provides a possibility to select only a subset of useful contexts (classifiers) without losing much performance. It is worth pointing out that the context selection is performed for each classification task separately in the training stage rather than for each individual test image. The context selection process is introduced in the follows.

Let $\{h_k, k = 1,\ldots,C\}$ denotes the classifier trained on the k-th context-specific BoW histogram. $w_k \in \{0,1\}, i = 1,\ldots,C$ indicates whether the k-th context is selected ("1" means selected). $F(h)$ is an evaluation function whose output is the performance of classifier h on the classification task to be addressed, where $h = \sum_{k=1}^{C} w_k h_k$ is a linear combination of the selected classifiers. The performance measure is the average precision or the classification accuracy, depending on the tasks (see Section 5 for details). Our aim is to get the optimal value of $W = [w_1, w_2, \ldots, w_C]$ which maximizes $F(h)$:

$$W^* = \arg\max_{W} F(h) \tag{8}$$

It is a combinatorial optimization problem; therefore the exhaustive search is computationally prohibitive when the number of contexts C is large (75 in our case). Thus, in this work, we adopt simulated annealing which is a stochastic optimization method to search for the global optima. As to the number of selected contexts, there are two options. It can be either considered as a parameter set by hand or chosen by simulated annealing automatically. In this work, we choose the former setting with which we can control the dimensionality of final image descriptor and therefore make fair comparisons with other methods (e.g. spatial pyramid matching).

In our work, the simulated annealing process starts from a random initial state. During each iteration, the new state (W) is generated by randomly selecting a context (e.g. k-th context) and flip its indicator w_k. Meanwhile, we need to flip the indicator of another randomly selected context to guarantee that the number of selected contexts does not change. A cooling temperature is involved in the iterative process and works like that: the choice between the previous and current state is almost done by chance when the temperature is large, but increasingly tends to select the better state as the temperature goes to zero. This cooling mechanism prevents the simulated annealing from stacking at local optima and therefore makes it outperform the simpler greedy search (validated by experiments in 5.5).

After context selection, an image is eventually represented by a small set of context-specific BoW histograms. Image classification is performed by running the classifiers trained on these context-specific BoW histograms and averaging their outputs. We refer to the selected histogram as contextBoW-m to distinguish with contextBoW-s introduced in Section 4.1. The whole procedure of constructing contextBoW-m is summarized in Algorithm 2.

Algorithm 2 Construction of ContextBoW-m

Input: image I, context-specific vocabulary $\{v_{k,j}, k = 1,\ldots,C, j = 1,\ldots,V\}$, local attribute classifiers $\{\phi_k, k = 1,\ldots,C\}$.
Extract a set of local features $\{f_i, i = 1,\ldots,N\}$ from image I.
for $j = 1,\ldots,V, k = 1,\ldots,C$ **do**
Compute $p(v_{k,j}|c_k,I) = \frac{1}{N}\sum_{i=1}^{N}\delta(f_i,v_{k,j})p(c_k|I_p(f_i))$.
end for
Solve $W^* = \arg\max_W F(\sum_{k=1}^{C} w_k h_k)$ {offline context selection}
where h_k is the classifier trained on $\{p(v_{k,j}|c_k,I), j = 1,\ldots,V\}$, $W = [w_1,w_2,\ldots,w_C]$, $w_k \in \{0,1\}$ indicates whether the k-th context is selected, and $F(h)$ gives the performance of classifier h on the classification task to be addressed.
Output: ContextBoW-m
$\{p(v_{k,j}|c_k,I), w_k = 1, j = 1,\ldots,V\}$

4.2.3 Relation to Spatial Pyramid Matching

Recall that the way we embed contextual information into the BoW model is based on weighting local features (see Eq. (4)). This is similar to the Spatial Pyramid Matching (SPM) [20] which divides an image into grids and build a histogram for each grid. Specifically, the SPM can be also considered as weighting local features: at a given level, features within a given bin are weighted by 1 while others are set to 0. However, there are two differences between our method and the SPM. First, in our method, the weights of the local features are continuous rather than binary. Secondly, the weights in SPM are the same for all images, while the weights given by the context classifiers are image-specific. Although less flexible than context-based weights, the binary weights in SPM are more stable which is also favorable. Thus, we add the SPM grids into the context selection process to balance the tradeoff between flexibility and stability. It is worthwhile to point out that, different from traditional SPM, we learn a specific vocabulary for each SPM grid based on local features within this grid. The context selection process with both semantic contexts and SPM grids is illustrated Fig. 4.

5 Experiments

This section presents the experimental validation of the proposed methods. The databases used for the experiments as well as some parameters of our algorithms are given in Section 5.1. Then we show the accuracy of the attribute classifiers and give some examples of attribute prediction in Section 5.2. The performance of semantic image descriptor, contextBoW-s, contextBoW-m as well as the demonstration of some aspects of the algorithms are given Section 5.3, 5.4 and 5.5 respectively. Finally, Section 5.6 gives the comparison with state-of-the-art results.

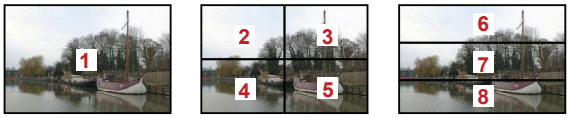


Fig. 5 Illustration of three-level spatial pyramid. Number in each bin denotes its index.

5.1 Experimental setup

Databases. Four publicly available image databases are used for the experiments: PASCAL VOC 2007 [7], Scene-15 [20], MSRCv2 [39] and SUN-397 [40].

PASCAL VOC 2007 is the last challenge for which the test data annotations are publicly available. The database contains 9963 images for 20 object classes, which were collected from users uploads to the Flickr website. The database has already partitioned into "training","validation" and "testing" sets. For the challenge's classification task, the goal is to determine whether or not each test image contains at least one instance of each object class of interest. Performance is measured by calculating the average precision (AP) for each class, and the mean average precision over the 20 categories (mAP), following the protocols given in [7].

Scene-15 database contains 15 scene categories, each of which has 200 to 400 gray-level images. These images come from the COREL collection, personal photographs, and Google image search. Following the experimental setup used in [20], 100 images per category are randomly sampled as training samples (remaining as testing samples). One-versus-all strategy is used for multiclass classification and the performance is reported as the average classification rate on the 15 categories.

MSRCv2 is an object category database. We follow the experimental setup used in [43] which chose 9 categories out of 15: cow, airplane, face, car, bike, book, sign, sheep and chair in order to make objects from different categories not to appear in the same image. In the experiments, 15 training images and 15 testing images are randomly sampled for each category. One-versus-all strategy is used for multiclass classification and the performance is reported as the average classification rate on 9 categories.

SUN-397 database contains 397 scene categories, each of which has at least 100 images collected from the Internet. Following the experimental setup used in [40], 50 images per category are randomly sampled as training samples (remaining as testing samples). One-versus-all strategy is used for multiclass classification and the performance is reported as the average classification rate on the 397 categories.

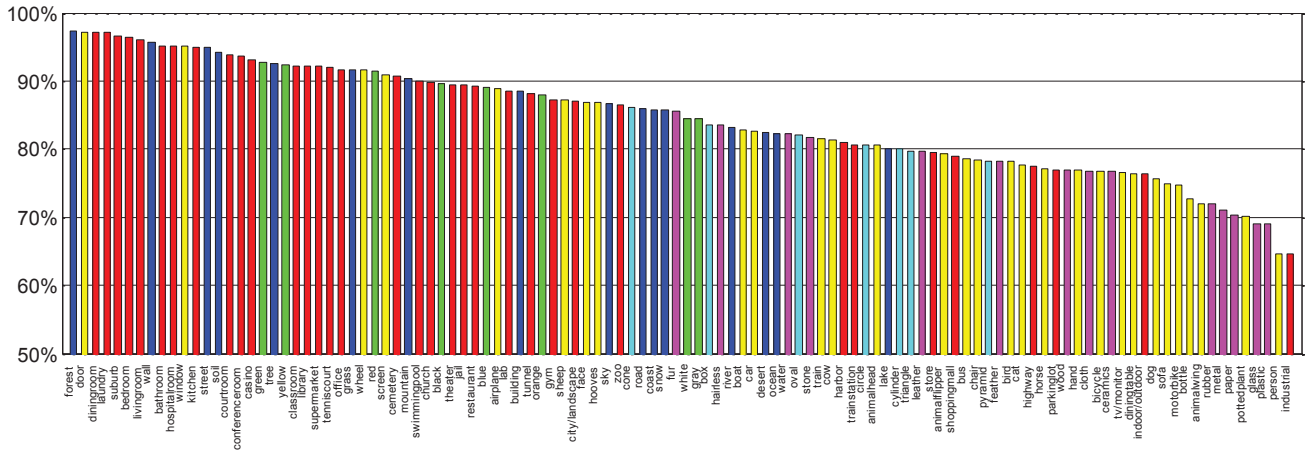


Fig. 6 Accuracy of individual attribute classifiers computed by fivefold cross-validation on training images. The colors show the groups of attributes: global scene, local scene, color, shape, material, part. The figure is better viewed in color.

Local features. Four types of local features, the ones proposed in [8], are used in our experiments: SIFT, Texton filterbanks (36 Gabor filters at different scales and orientations), LAB and Canny edge detection. Specifically, SIFT features are computed for 2000 image patches with randomly selected positions and scales (with scales from 16 to 64 pixels), and are quantized to 1024 k-means centers. Texton and LAB features are computed for each pixel, and quantized to 256 and 128 k-means centers respectively, while Canny edge features are quantized to 8 orientation bins. Combining these features gives a 1416-dimensional BoW feature vector.

Attribute classifiers. As mentioned in Section 3, attribute classifiers are learned by SVM with Battacharyya kernel (here we use the implementation of LIBSVM [2]), the inputs to which are BoW feature vectors constructed by pooling local features within image regions (for region-level classifiers) or whole images (for image-level classifiers). In order to estimate the occurrence probabilities of contexts, we use non-negative SVM scores obtained by fitting a sigmoid function to the original SVM decision value [2]. The SVM parameter C is set to 10, which is determined by fivefold cross-validation. As to the image regions used for local attribute classifiers, on each training image we sample 100 regions with random positions and scales (with scales from 20% to 40% of the image size). When training a local context classifier, 10,000 regions are randomly selected from positive and negative training images respectively. When training the global context classifiers, the average number of positive training images is about 400 and the same number of negative training images are randomly selected. distribution maps of local contexts from image regions on test images.

Image classification. For image classification, a SVM classifier with chi-square kernel (also implemented by using LIBSVM) is learned for each category. The value of the SVM parameter C and the normalization factor γ of chi-square kernel are determined by fivefold cross-validation. As to spatial pyramid matching (SPM), we use a three-level pyramid, 1×1, 2×2, 3×1 (totally 8 channels as shown in Fig. 5)

5.2 Evaluation of attribute classifiers

The prediction of semantic attributes plays the key role in our method. Thus, in this subsection, we evaluate the performances of attribute classifiers and give some examples of attribute prediction.

Fig. 6 shows the accuracy achieved by individual attribute classifiers, which are computed by fivefold cross-validation on training images. When training and testing attribute classifiers, the negative examples were sampled to balance the positive examples, so making a random prediction would give a 50% accuracy. As illustrated in Fig. 6, most of the classifiers achieve higher than 80% accuracy; the lowest accuracies are seen on the material attributes, while on average the *global scene* attribute classifiers perform the best. Using more negative training samples produces attribute classifiers with slightly better accuracy but does not improve the performance of the resultant semantic image descriptor (see Fig. 7). As mentioned in Section 3, the attribute classifiers are learned by SVM with Battacharyya kernel. It is shown in Fig. 7 that Battacharyya kernel significantly outperforms linear kernel but the more complex chi-square kernel does not lead to better performance. Thus, Battacharyya kernel gives the best trade-off between computational cost and performance.

We use these attribute classifiers to make soft predictions of attribute occurrence, and use those predictions as features

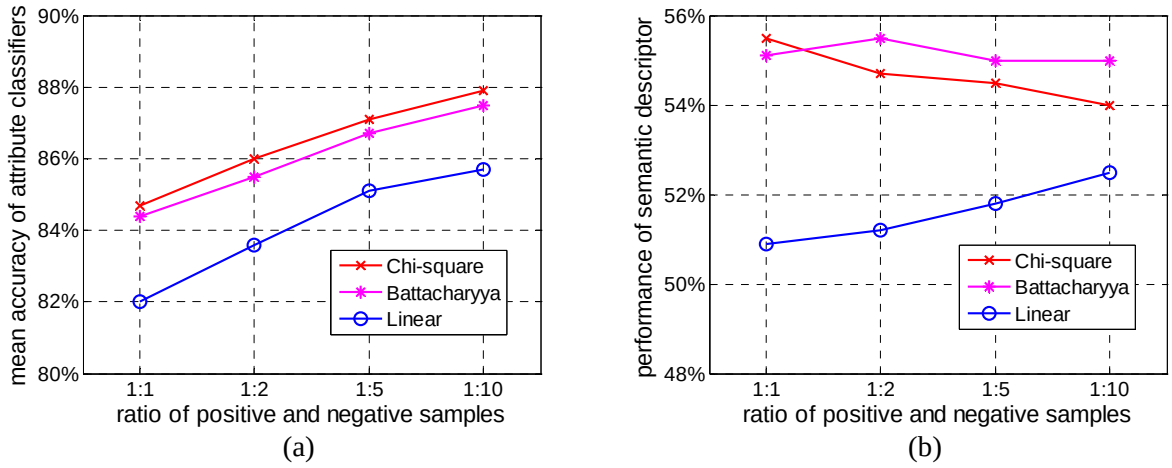


Fig. 7 Influence of the number of negative training samples and of the type of kernel on (a) the accuracy of attribute classifiers and (b) the final image classification performance given by those attribute classifiers. In (b), the performance is measured as the mAP of semantic image descriptor on PASCAL VOC 2007 database.

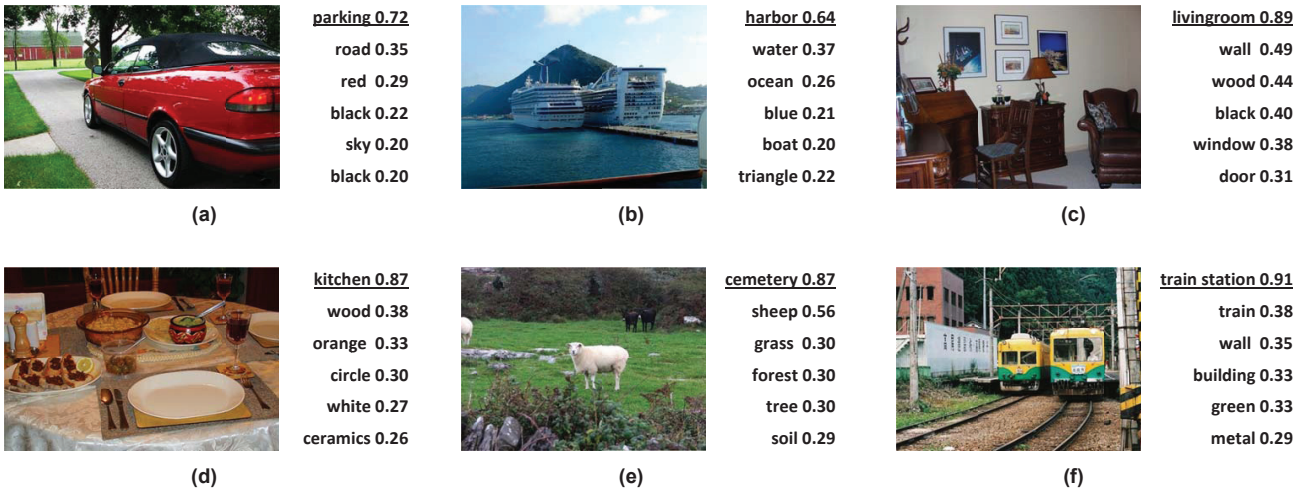


Fig. 8 Examples of semantic attribute prediction. For each image, we give the strongest prediction of global attribute (underlined) and the top 5 predictions of local attributes. The value after each prediction denotes the confidence given by the corresponding attribute classifier.

to build semantic image descriptor and disambiguate the visual words. In Fig. 8, we give some examples of attribute prediction. In many cases where the prediction is not accurate enough, it is possible to understand why the attribute classifier makes such predictions. For example, the *car* and *road* regions of the image given in Fig. 8(a) make the scene look like a parking lot; the photo frames hanging on the wall look, in Fig. 8(c), similar to windows and doors; the grass and stone in Fig. 8(e) make the scene similar to a cemetery.

5.3 Evaluation of semantic image descriptor

Recall that the semantic descriptors for local attributes are computed by running the local attribute classifiers on image regions and then pooling the classifier outputs. We experiment both average pooling and maximum pooling to construct the semantic descriptor (75-D). The performances of average pooling on PASCAL 2007 is 52.3% (mAP), which is much better than that of maximum pooling, i.e. 46.8%.

Fig. 9 gives the performances of different groups of semantic attributes. The attributes of *global scene*, *local scene* and *object* perform better than others. The worse performances of *color* and *shape* attributes are mainly due to their lower dimensionalities while the worse performance of *material* attributes lies in the difficulty of predicting them (see Fig. 6).

Table 1 summarizes the performances of semantic descriptor, BoW histogram, and their combinations by different rules. Three conclusions can be drawn from Table 1. First, semantic descriptors perform close to BoW histogram while their dimensionality (110-D) is much lower than that of BoW histogram (1416×8=11328-D). Second, combining

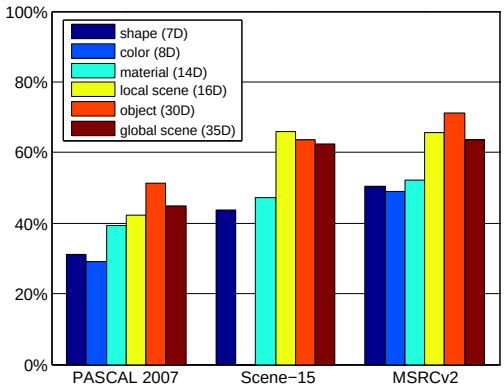


Fig. 9 Performances of different groups of semantic attributes. We do not give the performance of *color* attributes on Scene-15 database because it contains only grey-level images. Values marked off by brackets denote the number of attributes in the corresponding group. The figure is better viewed in color.

semantic descriptors with BoW histogram improves the performance, which validates that they are complementary to each other. Third, the weighted sum rule performs best to combine them.

For a more detailed comparison, Fig. 10 gives the performance achieved by semantic descriptors, BoW histogram and their combination (weighted sum) on every object category of PASCAL 2007 and Scene-15 databases. On the majority of categories, semantic descriptors perform worse than BoW histogram, while on eight categories ('bird', 'bottle', 'chair', 'dog', 'person', 'potted plant', 'suburb', 'coast') semantic descriptors performs better. The performance on each category is increased by combining the two feature types, instead of using only one of them.

In [1], images are represented by the mixing coefficients of topics, obtained with pLSA. This representation bears similarities with the proposed semantic descriptors. Thus, we re-implement the method in [1] and compare it with our semantic descriptor. To be fair, the number of topics is set to the dimensionality of semantic descriptor and the same classifier is used for classification. The performance of this pLSA-based descriptor is 52.8% (mAP) on PASCAL 2007 which is worse than that of semantic descriptor (55.1%). In addition, we compare our method with another attribute-based method [38]. In this method, an image is represented by a descriptor of 103 dimensions, each of which corresponds to the similarity of this image to a Flickr image group. Although its dimensionality is a little lower, our semantic descriptor gives much better performance (55.1%) on PASCAL 2007 than this 103-D similarity-based descriptor (44.9% reported in [38]).

	PASCAL 2007	Scene-15	MSRCv2	SUN-397
Semantic	55.1	79.1 ± 0.9	82.8 ± 2.8	25.4 ± 0.6
BoW+SPM	59.2	83.3 ± 0.7	86.2 ± 2.3	30.9 ± 0.4
weighted sum	**62.2**	$\mathbf{86.1 \pm 0.3}$	$\mathbf{88.0 \pm 2.6}$	$\mathbf{33.8 \pm 0.4}$
product	61.8	85.4 ± 0.6	86.9 ± 2.2	33.3 ± 0.5
max	60.9	83.1 ± 0.4	86.6 ± 1.8	33.1 ± 0.5

Table 1 Performance of semantic descriptors, standard BoW+SPM model and their combination by weighed sum, product and max rules. The optimal weight in the weighted sum rule is learned on the validation set of Pascal 2007 database.

5.4 Evaluation of contextBoW-s

5.4.1 Qualitative results

In this subsection, we give some examples illustrating the context selection process. As mentioned in Section 4.1, we choose only one context for each visual word, the most relevant for the category to be classified. Hence, for each category, we can compute the frequency each context is selected, and higher frequency means higher relevance for this category. Fig. 11 gives the frequencies of contexts for category 'cow', 'motorbike' and 'living room'. It can be seen that even if the relevance of different contexts vary greatly, the contexts that are related to the category to be classified tend to have higher relevance. Take Fig. 11(b) as an example, besides *motorbike*, the context *street* and *wheel* also play important roles in 'motorbike' classification.

As explained before, the context selection depends on the classification task to be addressed. It means an image is described differently for different classification tasks. For example, in Fig. 12, for 'motorbike' classification, the two most relevant contexts are *motorbike* and *street*. This result can be easily explained. For 'person' classification, the contexts *black* and *sky* dominate the image description. These two local contexts seem to have no relation with 'person', whereas one possible explanation is that in daily life people often wears dark or blue clothes.

5.4.2 Parameter evaluation

In the computation of context-embedded BoW histogram (contextBoW-s), the number of randomly sampled regions per image is an important parameter. Hence, we do several experiments on the validation set of PASCAL 2007 to evaluate the effect of the number of regions as well as the way to chose their locations (random sampling vs. regular grid). From these experiments, we conclude that sampling regions on a regular grid does not give better results than sampling them randomly. However, random sampling raises questions about the stability of results and the number of regions to sample. If we sample 10, 50 and 100 regions per image, the mAP are respectively 56.2%, 56.8% and 57.3%. Taking

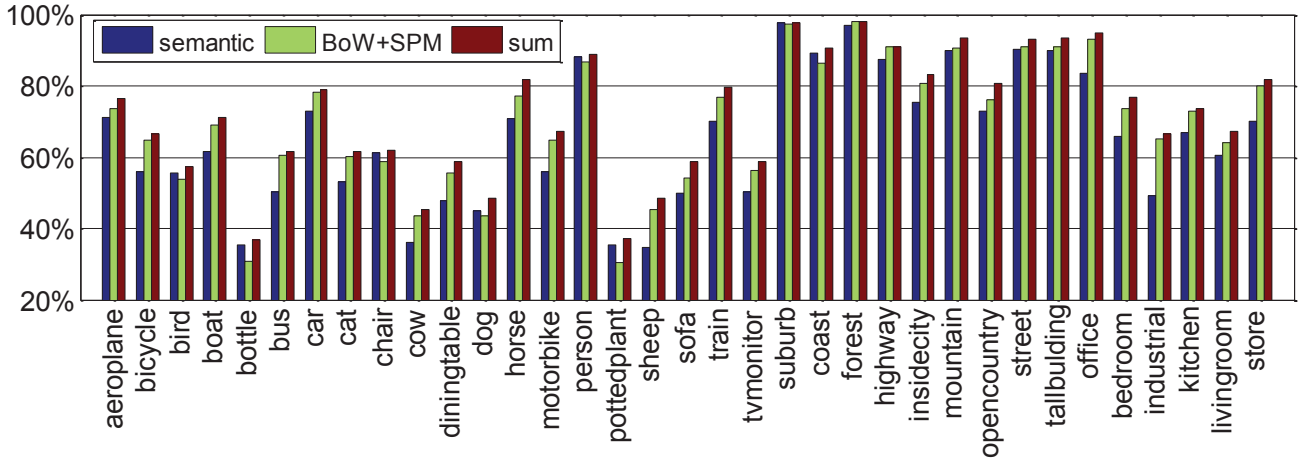
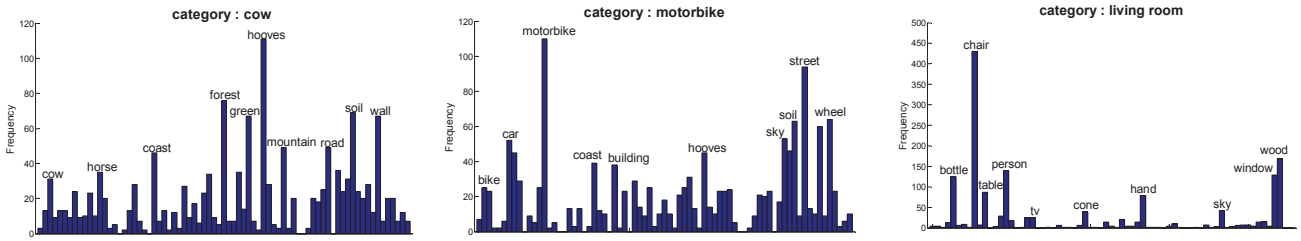


Fig. 10 Average precision achieved using bag of words histogram (with SPM), semantic descriptors and their combination, on PASCAL VOC 2007 and Scene-15 database.



(a) (b) (c)

Fig. 11 Selection frequencies of different contexts for three categories: 'cow', 'motorbike' and 'living room'. The contexts with high frequency are marked by their names.

more than 100 regions does not improve the results significantly. Regarding stability, the standard deviations observed over 5 runs, if we sample 10, 50 or 100 regions per image, are respectively 0.5%, 0.3% and 0.2%. Hence, if 100 regions are randomly sampled, the choice for these regions does not have a great effect on the performance of contextBoW-s.

As mentioned in Section 4.1, we rank contexts for each visual word and select only the best one, resulting in the V-dimensional descriptor (contextBoW-s). Although it is also possible to use more contexts (e.g. top 2, 3 or 5) for each visual word, with the cost of higher dimensionality of image description, Fig. 13 shows that it does not result in a significant performance improvement (at most 0.2%). Furthermore, instead of context selection, we can use other dimensionality reduction methods, such as *Principal Component Analysis (PCA)* or *Linear Discriminant Analysis (LDA)*, to obtain a low dimensional image descriptor. To validate the effect of them, we use PCA and LDA to project the C-dimensional descriptor $(p(v|c_1,I),p(v|c_2,I),...,p(v|c_C,I))$ for each visual word into a lower dimensional subspace. Fig. 13 gives the performance of PCA (up to 5-D) and LDA (only 1-D due to the binary classification task on PASCAL 2007 database), which are worse than that of our context selection. In short, selecting a single context for each visual word gives the best tradeoff between performance and dimensionality.

Finally, we evaluate the influence of the number of visual words. It can be seen from Fig. 14 that, as the number of visual words increases, the performance of context-embedded BoW histogram (on validation set of PASCAL 2007) continues to increase. However, the performance saturates when the number of visual words exceeds 1024. Thus, the number of visual words is set to 1024 for the following experiments, on all four databases. Note that Fig. 14 gives the performance of the visual words learned from SIFT features. Similar experiments are also done for texton and LAB features to determine the optimal number of visual words (256 and 128 respectively).

5.4.3 Comparison with standard BoW+SPM model

In this subsection, we compare our methods with the standard BoW model. Table 2 summarizes the performances of BoW model, contextBoW-s, semantic descriptor and their

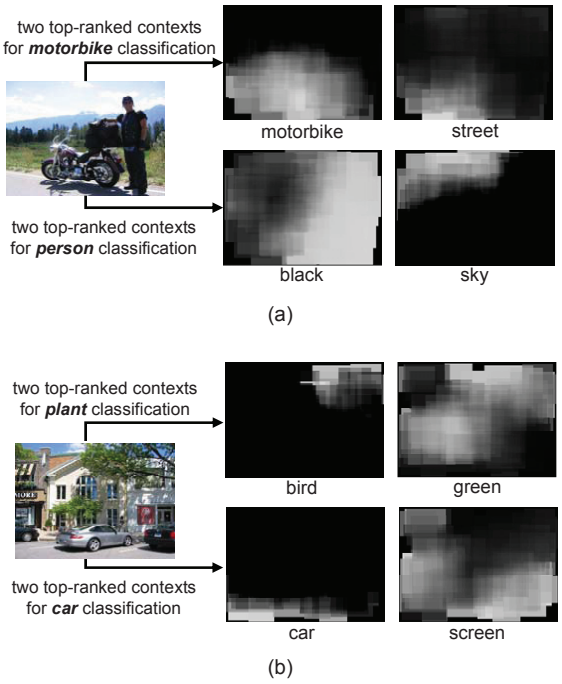


Fig. 12 Probability maps of the two top-ranked contexts for different classification tasks. The value of each pixel on the probability map is computed by averaging the outputs of corresponding context classifiers on the image regions covering this pixel.

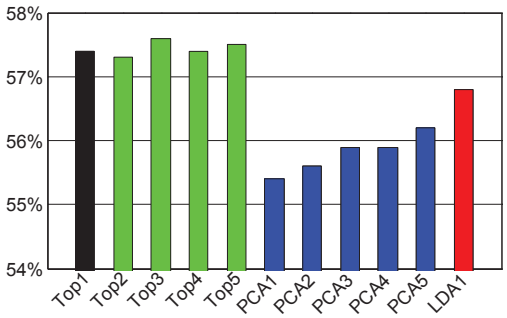


Fig. 13 Performance comparison of different dimension reduction methods on the validation set of PASCAL 2007. Top N means that the top-ranked N contexts are kept. The numbers after PCA and LDA denote the dimensionality of the subspace.

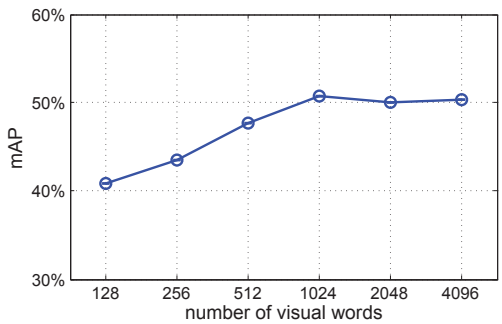


Fig. 14 Performances of ContextBoW-s with different number of visual words learned from SIFT features. The experiment is done on the validation set of PASCAL 2007.

	PASCAL 2007	Scene-15	MSRCv2	SUN-397
BoW+SPM	59.2	83.3 ± 0.7	86.2 ± 2.3	30.9 ± 0.4
semantic	55.1	79.1 ± 0.9	82.8 ± 2.8	25.4 ± 0.6
contextBoW-s	62.0	85.4 ± 0.5	88.5 ± 2.4	32.5 ± 0.7
contextBoW-s+semantic	**64.5**	**87.8 ± 0.5**	**90.7 ± 1.8**	**34.7 ± 0.5**
result from database creator	59.4 [7]	81.4 [20]	80.4 ± 2.5 [43]	38.0 [40]

Table 2 Performance comparison between our methods and the standard BoW+SPM model.

combination on four databases. Here the spatial pyramid (SPM) is applied on both BoW model and contextBoW-s to enhance their performances. It can be concluded from Table 2 that, by embedding contextual information, the performance of BoW model is improved, say 2.8% on PASCAL 2007, 2.1% on Scene-15, 2.3% on MSRCv2 and 1.6% on SUN-397. As observed in previous experiments, although semantic descriptors do not give better performance than BoW model, combining them with contextBoW-s leads to additional improvement, demonstrating that they are somewhat complementary. Finally, the improvement of our method (contextBoW-s+semantic) to BoW model is 5.3% on PASCAL 2007, 4.5% on Scene-15, 4.5% on MSRCv2 and 3.8% on SUN-397.

For more detailed comparisons, Fig. 15 gives the performance improvement for each category of PASCAL 2007 and Scene-15 databases. It can be seen that contextBoW-s performs better than BoW model on 31 of 35 categories (except for 'bus', 'cat', 'highway' and 'kitchen'), whereas contextBoW-s+semantic performs better than BoW model on all categories. In particular, for category 'pottedplant', the improvement of average precision is more than 10%. We believe the reason of this large improvement is that pottedplants are very diverse in appearance and have small sizes; therefore their classification mainly depends on the contextual information.

5.5 Evaluation of contextBoW-m

5.5.1 Qualitative results

In this subsection, we first give some examples illustrating the context selection at task-level. As mentioned in Section 4.2, we select a subset of contexts for each individual classification task by using simulated annealing. As it is a stochastic process, we ran the context selection procedure 10 times for each classification task and then reported the selection frequency of every context. In this experiment, there is no constraint on the number of selected contexts and 8 SPM channels are also involved in the context selection process. Fig. 16 shows the selection frequencies of contexts for 'bottle', 'car' and Scene-15 database. Note that, different from PASCAL 2007 database in which the binary classification tasks are independent from each other, the multi-class classification task in Scene-15 database is considered as a whole

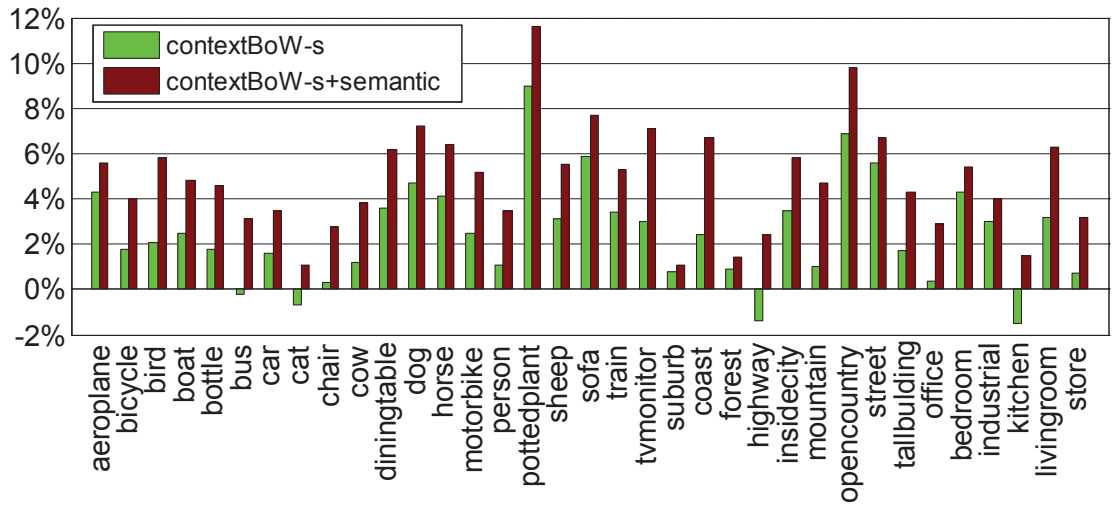


Fig. 15 Performance improvement of our methods over the standard BoW+SPM model on PASCAL 2007 database.

for which the context selection is performed. Similar to the previous observation, the contexts which are more relevant to the classification task tend to be selected. For example, in Fig. 16(a), some indoor contexts (e.g. *wall*, *door* and *screen*) play important roles in 'bottle' classification since bottle often appears in indoor scenes. Another interesting observation is that the importance of SPM channels also depends on the classification task itself. For example, in Fig. 16(a), SPM channels of entire image play more important role in 'bottle' classification since bottles usually appear in clutter backgrounds whose characteristics are better modeled by entire image than image regions. In Fig. 16(b), the bottom region of an image (probably road) is much more important than other parts for 'car' classification. It can be also observed from Fig. 16 that SPM channels plays a more important role in scene classification than in object classification. This is reasonable because the spatial configurations of scene images are more consistent than those of object images.

5.5.2 Comparison between SPM channels and semantic contexts

As mentioned in Section 4.2, some SPM channels are also involved in the context selection process. Some qualitative results have already shown the complementarities of SPM channels and semantic contexts (see Fig. 16). In the following, we quantitatively evaluate the effect of additional SPM channels in context selection. Fig. 17 gives the performance of our method in three different settings, i.e. SPM channels, semantic contexts and both. In the first setting, 8 SPM channels (1×1, 2×2, 3×1) are used without any selection process. It is worth pointing out that the SPM used here is a little different from the traditional one as different vocabulary is learned for each channel and the combination of different channels is performed at decision level rather than at feature level. In the second and third settings, to keep the final image description the same dimensionality as that in the first setting, the number of contexts selected by simulated annealing is also set to 8. It can be observed from Fig. 17 that when classifying outdoor scenes (e.g. 'mountain', 'street') or objects in outdoor scenes (e.g. 'boat', 'car') the semantic contexts often give good results without using SPM channels. On the contrary, when classifying indoor scenes (e.g. 'bedroom', 'kitchen') and objects in indoor scenes (e.g. 'bottle', 'sofa') the SPM channels performs similar to semantic contexts and combining them improves the performances. The reason behind this observation is that there are much more attributes in our method to describe outdoor scenes than indoor scenes, therefore when classifying indoor scenes and objects the SPM channels are needed as a supplement. Furthermore, the global layout of indoor images within the same category is more consistent.

5.5.3 Evaluation of context selection

In context selection, the number of selected contexts is an important parameter. Fig. 18 gives the performances of contextBoW-m with different number of contexts. In this experiment, the candidate contexts include both semantic contexts and SPM channels. Besides, in order to validate the effectiveness of simulated annealing, we also compare it with random selection, greedy search and logistic regression. Since logistic regression learns a weight for each context, we can either combine all contexts by weighted sum or select contexts with higher weights (the absolute values are considered). It can be seen from Fig. 18 that simulated annealing gives better performance than other methods. Moreover, the performance of using selected contexts quickly approaches that of combining all contexts uniformly (horizontal solid line in Fig. 18), which validates the importance of

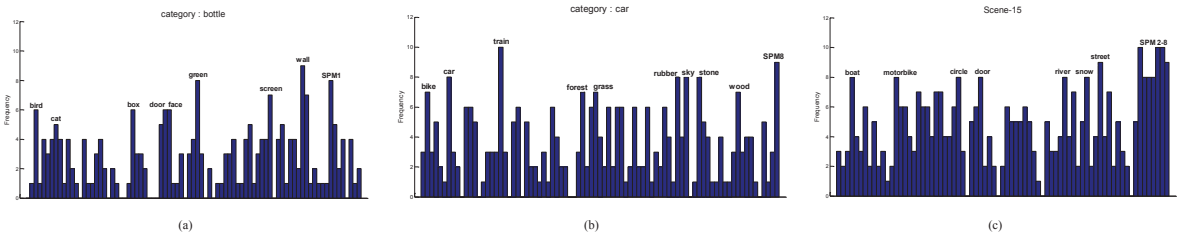


Fig. 16 Selection frequencies of different contexts for category 'bottle' and 'car', as well as Scene-15 database. The contexts with high frequency are marked by their names.

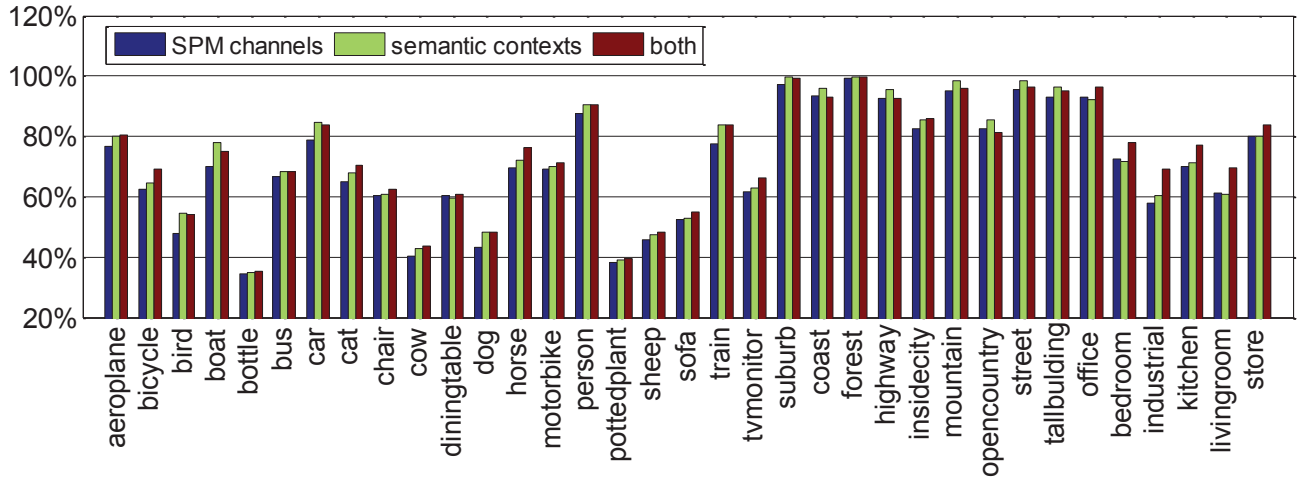


Fig. 17 Performances of contextBoW-m with SPM channels, semantic contexts and both. In these cases, the feature dimensionality of contextBoW-m is kept the same for fair comparison.

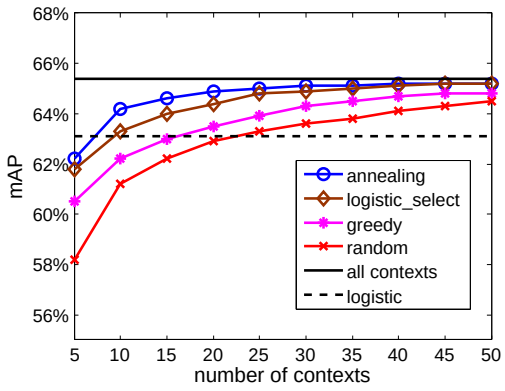


Fig. 18 Performances of contextBoW-m on PASCAL 2007 database with different number of contexts. The horizontal solid and dash lines denote the performance of combining all contexts with uniform weights and the weights learned by logistic regression respectively.

context selection. It is worth noting that combining all the contexts by weighted sum performs worse than selecting a subset of contexts according to the weights. The reason we think is that the weight optimization is not directly related to the final performance measure (i.e. mAP).

5.5.4 Comparison with standard BoW+SPM model

Table 3 summarizes the performances of BoW+SPM model, contextBoW-s, contextBoW-m and ther combination with semantic descriptors. The number of selected contexts in contextBoW-s is set to 8 so that the dimensionality of image representation is the same as that of BoW+SPM and ContextBoW-m. It can be concluded that, by learning a vocabulary for each context, contextBoW-m not only outperforms standard BoW+SPM model but also outperforms contextBoW-s in which a single vocabulary is learned for all contexts. Moreover, the performance of contextBoW-m can be enhanced by combining it with semantic descriptors. Finally, the improvement of our method (contextBoW-m+semantic) to BoW+SPM model is 7.4% on PASCAL 2007, 6.5% on Scene-15, 6.3% on MSRCv2 and 4.7% on SUN-397.

We also compare the contextBoW-m with the geometry texton histograms [40] which are built using texton features and four geometry contexts. To be fair, we build the contextBoW-m using only texton features. As in [40], the SVM with chi-square kernel is used to learn the final classifier. On SUN-397 database, the performance of this reduced contextBoW-m is 27.4% which is better than that of geometry texton histograms (23.5%).

	PASCAL 2007	Scene-15	MSRCv2	SUN-397
BoW+SPM	59.2	83.3 ± 0.7	86.2 ± 2.3	30.9 ± 0.4
semantic	55.1	79.1 ± 0.9	82.8 ± 2.8	25.4 ± 0.6
contextBoW-s	62.0	85.4 ± 0.9	88.5 ± 2.4	32.5 ± 0.7
contextBoW-m	64.2	87.5 ± 0.9	90.6 ± 2.2	33.8 ± 0.5
contextBoW-m+semantic	**66.6**	$\mathbf{89.8 \pm 0.7}$	$\mathbf{92.5 \pm 2.0}$	$\mathbf{35.6 \pm 0.4}$
result from database creator	59.4 [7]	81.4 [20]	80.4 ± 2.5 [43]	38.0 [40]

Table 3 Performance comparison between standard BoW+SPM model and different combinations of our methods.

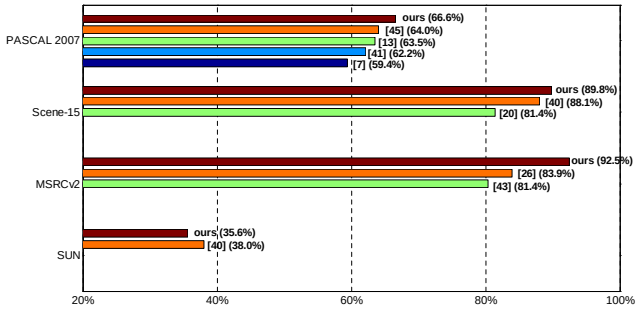


Fig. 19 Comparison between our method (contextBoW-m+semantic) and several state-of-the-art approaches.

5.6 Comparison with state-of-the-art results

It is worthwhile to point out that the results of our method (contextBoW-m+semantic) on PASCAL 2007, Scene-15 and MSRCv2 databases are better than the state-of-of-art results on these databases (as illustrated in Fig. 19). More specifically, on PASCAL 2007, our method achieves the mAP of 66.6%, which is better than [41] (reporting 62.2%), [13] (reporting 63.5%), [45] (reporting 64.0%), as well as the top results obtained at the PASCAL 2007 challenge [7] (59.4%).

On Scene-15, our method achieves the mean classification accuracy of 89.8%, which is better than 88.1% reported in [40], while we use much less features than they do (they combine 8 different types of features for the experiments on Scene-15) and outperforms the 81.4% reported in [20].

On MSRCv2, our method achieves the mean classification accuracy of 92.5%, which is much better than the 80.4% and 83.9% reported in [43] and [26] respectively.

On SUN-397, our method achieves the mean classification accuracy of 35.6% which is worse than the 38.0% reported in [40], but we use much less features than they do (they combine 15 different types of features for the experiments on SUN-397).

5.7 Summary

This subsection summarizes the conclusions drawn from the performed experiments.

First, we have observed that learned from manually labeled images, the attribute classifiers are able to give meaningful attribute predictions for unseen images (see Fig. 8). When learning attribute classifiers, the number of randomly sampled negative samples does not have big influence on the final classification performance, and the Battacharyya kernel gives the best trade-off between computational cost and performance (see Fig. 7).

Second, semantic image descriptor performs only a little worse than BoW histograms but with much lower dimensionality; its combination with BoW histogram leads to significant performance improvement (see Table.1).

Third, the performance of BoW histograms can be significantly improved by embedding semantic information, i.e. by learning context-specific vocabularies and building context-specific BoW histograms (see Table.2 and 3). Moreover, the context-embedded BoW histograms (contextBoW-s and contextBoW-m) are also complementary to the semantic image descriptor (see Table.2 and 3).

Fourth, context selection (t-test score for contextBoW-s and simulated annealing for contextBoW-m) gives the best trade-off between the performance and dimensionality of context-embedded BoW histogram (see Fig. 13 and Fig. 18).

Finally, our method performs better or similarly to the state-of-the-art results on all the used databases.

6 Conclusion and Discussion

In this paper, we have presented two novel methods to improve the performance of the bag-of-words model for image classification, via the prediction of semantic attributes. One is combining bag-of-words histograms with semantic image descriptors at decision level. The other is embedding semantic information into the visual vocabulary. Extensive experimental results demonstrated that both methods enhance the performance of bag-of-words model by a large margin. Moreover, combining two methods brought even further improvement. In short, our method outperformed bag-of-words model by 7.4% on PASCAL VOC 2007, 6.5% on Scene-15, 6.3% on MSRCv2 and 4.7% on SUN-397, and also achieved the state-of-the-art results on several of these challenging image databases.

At last, we will give some discussions on our method. The first one is about its practicality. Indeed, it takes some time to collect images and train classifiers for semantic attributes. However, this is an off-line training phase and the attribute classifiers are generic and task-independent; therefore they can be reused. In the testing phase, since the attribute classifiers are linear SVMs (performed on square-rooted BoW histograms), the construction of the probabilistic distribution of contexts is quite efficient. Thus, the computation time of context-embedded BoW histogram is comparable to that of traditional bag-of-words histogram. As to the training images of attribute classifiers, in our method, they are collected by web search and then manually labeled.

However, it would also be possible to train attribute classifiers directly from the top ranked images which includes outliers, at the cost of degrading the classifier accuracy. This approach would become more compelling if larger numbers of attributes were used in future work.

In our method, the local attribute classifiers and the semantic information embedded in them play key roles in enhancing the traditional BoW histogram. To validate this point, we tried to learn the local attribute classifiers on regions sampled from random training images and then repeat the same procedure to build context-embedded BoW histogram. In this case, the attribute classifiers do not have any semantic meaning. Experimental results on PASCAL VOC 2007 database shows that the mAP of context-embedded BoW histogram built by using random attribute classifier is about 4% to 6% worse that of ContextBoW-s and ContextBoW-m built by using semantic attribute classifiers.

7 Acknowledgement

This work was partly realized under the Quaero Programme, funded by OSEO, French State agency for innovation.

References

1. Bosch, A., Zisserman, A., Munoz, X.: Scene classification via pLSA. In: ECCV (2006)
2. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology **2**, 27:1–27:27 (2011). Software available at http://www.csie.ntu.edu.tw/čjlin/libsvm
3. Chapelle, O., Haffner, P., Vapnik, V.: Support vector machines for histogram-based image classification. Neural Networks, IEEE Transactions on **10**(5), 1055–1064 (1999)
4. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: Proc. Workshop on Statistical Learning in Computer Vision, at ECCV (2004)
5. Delaitre, V., Laptev, I., Sivic, J.: Recognizing human actions in still images: a study of bag-of-features and part-based representations. In: BMVC (2010)
6. Deselaers, T., Ferrari, V.: Visual and semantic similarity in ImageNet. In: CVPR (2011)
7. Everingham, M., Van Gool, L., Williams, C., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007 results. http://www.pascal-network.org/challenges/VOC/voc2007/
8. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: CVPR (2009)
9. Fei-Fei, L., Perona, P.: A Bayesian hierarchical model for learning natural scene categories. In: CVPR (2005)
10. Gehler, P., Nowozin, S.: On feature combination for multiclass object classification. In: ICCV (2009)
11. van Gemert, J., Veenman, C., Smeulders, A., Geusebroek, J.M.: Visual word ambiguity. IEEE Transactions on Pattern Analysis and Machine Intelligence **32**(7), 1271–1283 (2010)
12. Griffin, G., Holub, A., Perona, P.: Caltech-256 object category dataset. Tech. Rep. 7694, California Institute of Technology (2007)
13. Harzallah, H., Jurie, F., Schmid, C.: Combining efficient object localization and image classification. In: ICCV (2009)
14. Hofmann, T.: Probabilistic latent semantic analysis. In: Proc. of Uncertainty in Artificial Intelligence (1999)
15. Ji, R., Yao, H., Sun, X., Zhong, B., Gao, W.: Towards semantic embedding in visual vocabulary. In: CVPR (2010)
16. Khan, F., van de Weijer, J., Vanrell, M.: Top-down color attention for object recognition. In: ICCV (2009)
17. Kittler, J., Hatef, M., Duin, R., Matas, J.: On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence **20**(3), 226–239 (1998)
18. Kumar, N., Berg, A., Belhumeur, P., Nayar, S.: Attribute and simile classifiers for face verification. In: ICCV (2009)
19. Lampert, C., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: CVPR (2009)
20. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR (2006)
21. Leung, T., Malik, J.: Representing and recognizing the visual appearance of materials using three-dimensional textons. International Journal of Computer Vision **43**, 29–44 (2001)
22. Li, L., Su, H., Xing, E., Fei-Fei, L.: Object bank: a high-Level image representation for scene classification and semantic feature sparsification. In: NIPS (2010)
23. Li, L.J., Wang, C., Lim, Y., Blei, D., Fei-Fei, L.: Building and using a semantivisual image hierarchy. In: CVPR (2010)
24. Liu, J., Yang, Y., Shah, M.: Learning semantic visual vocabularies using diffusion distance. In: CVPR (2009)
25. Moosmann, F., Triggs, B., Jurie, F.: Fast discriminative visual codebooks using randomized clustering forests. In: NIPS (2007)
26. Morioka, N., Satoh, S.: Building compact local pairwise codebook with joint feature space clustering. In: ECCV (2010)
27. Perronnin, F., Senchez, J., Liu, Y.: Large-scale image categorization with explicit data embedding. In: CVPR (2010)
28. Rosch, E., Mervis, C., Gray, W., Johnson, D., Boyes-Braem, P.: Basic objects in natural categories. Cognitive psychology **8**(3), 382–439 (1976)
29. Saghafi, B., Farahzadeh, E., Rajan, D., Sluzek, A.: Embedding visual words into concept space for action and scene recognition. In: BMVC (2010)
30. Sivic, J., Russell, B., Efros, A., Zisserman, A., Freeman, W.: Discovering objects and their location in images. In: ICCV (2005)
31. Sivic, J., Russell, B., Zisserman, A., Freeman, W., Efros, A.: Unsupervised discovery of visual object class hierarchies. In: CVPR (2008)
32. Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: ICCV (2003)
33. Su, Y., Allan, M., Jurie, F.: Improving object classification using semantic attributes. In: BMVC (2010)
34. Su, Y., Jurie, F.: Visual word disambiguation by semantic contexts (2011)
35. Torresani, L., Szummer, M., Fitzgibbon, A.: Efficient object category recognition using classemes. In: ECCV (2010)
36. Ullah, M., Parizi, S., Laptev, I.: Improving bag-of-features action recognition with non-local cues. In: BMVC (2010)
37. Vogel, J., Schiele, B.: Semantic modeling of natural scenes for content-based image retrieval. International Journal on Computer Vision **72**(2), 133–157 (2007)
38. Wang, G., Hoiem, D., Forsyth, D.: Learning image similarity from flickr groups using stochastic intersection kernel machines. In: ICCV (2009)
39. Winn, J., Criminisi, A., Minka, T.: Object categorization by learned universal visual dictionary. In: ICCV (2005)
40. Xiao, J., Hays, J., Ehinger, K., Oliva, A., Torralba, A.: Sun database: Large-scale scene recognition from abbey to zoo. In: CVPR (2010)
41. Yang, J., Li, Y., Tian, Y., Duan, L., Gao, W.: Group-sensitive multiple kernel learning for object categorization. In: ICCV (2009)

42. Yuan, J., Wu, Y., Yang, M.: Discovery of collocation patterns: from visual words to visual phrases. In: CVPR (2007)
43. Zhang, Y., Chen, T.: Efficient kernels for identifying unbounded-order spatial features. In: CVPR (2009)
44. Zheng, Y., Zhao, M., Neo, S., Chua, T., Tian, Q.: Visual synset: towards a higher-level visual representation. In: CVPR (2008)
45. Zhou, X., Yu, K., Zhang, T., Huang, T.: Image classification using super-vector coding of local image descriptors. In: ECCV (2010)