J. Vis. Commun. Image R. 25 (2014) 1387–1398

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Beyond visual word ambiguity: Weighted local feature encoding with governing region

Chunjie Zhang [a], Xian Xiao [b,*], Junbiao Pang [c], Chao Liang [d], Yifan Zhang [e], Qingming Huang [a,f]

[a] School of Computer and Control Engineering, University of Chinese Academy of Sciences, 100049 Beijing, China
[b] Institute of Automation, Chinese Academy of Sciences, Beijing, China
[c] College of Computer Science and Technology, Beijing University of Technology, 100124 Beijing, China
[d] National Engineering Research Center for Multimedia Software, School of Computer, Wuhan University, 430072 Wuhan, China
[e] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, P.O. Box 2728, Beijing, China
[f] Key Lab of Intell. Info. Process, Institute of Computing Technology, Chinese Academy of Sciences, 100190 Beijing, China

ARTICLE INFO

Article history:
Received 16 October 2013
Accepted 23 May 2014
Available online 12 June 2014

Keywords:
Visual word ambiguity
Governing region
Weighted encoding
Image classification
Sparse
Bag-of-visual words
Object categorization
Locality constraint

ABSTRACT

Typically, *k*-means clustering or sparse coding is used for codebook generation in the bag-of-visual words (BoW) model. Local features are then encoded by calculating their similarities with visual words. However, some useful information is lost during this process. To make use of this information, in this paper, we propose a novel image representation method by going one step beyond visual word ambiguity and consider the governing regions of visual words. For each visual application, the weights of local features are determined by the corresponding visual application classifiers. Each weighted local feature is then encoded not only by considering its similarities with visual words, but also by visual words' governing regions. Besides, locality constraint is also imposed for efficient encoding. A weighted feature sign search algorithm is proposed to solve the problem. We conduct image classification experiments on several public datasets to demonstrate the effectiveness of the proposed method.

1. Introduction

The bag-of-visual words (BoW) model has been very popular for image applications in recent years. It is inspired by the bag-of-words model for text analysis. Basically, the BoW model can be divided into four steps. First, it selects image regions and extracts local features to represent these regions. Second, a codebook is generated either by *k*-means clustering [1] or sparse coding [2]. Third, each local feature is then encoded accordingly to form image representation. Finally, classifiers are trained to predict the classes of images.

Of the four modules, the generation of codebook and local feature encoding modules are very important for the final image representation. A codebook is a collection of patterns which are used for local feature encoding. Usually, *k*-means clustering is used and local features are quantized by nearest neighbor assignment. However, as pointed out by many researchers [3–10], this hard assignment strategy causes sever information loss, especially when local features lie on the boundary of visual words. To reduce the information loss during local feature encoding process, a lot of works [3–10] have been done by softly assigning local features, such as kernel codebook [3] and sparse coding [2].

Although proven effective, there are some problems with previous codebook generation and local feature encoding strategies. First, each local feature is treated equally for codebook generation and image representation. The generation of codebook and encoding of local features should be task dependent for efficient image representation. Second, the extracted local features are often too many to be used for codebook generation due to the computational power and memory usage required. To solve this problem, researchers choose to randomly select local features for codebook generation. The cluster centers are viewed as visual words. However, during the new local feature encoding process, only the visual words are used. This strategy discards the useful information of local features during the codebook generation process [8]. In fact, what we learned during the codebook generation is a partition of local feature space, not merely cluster centers. We believe this information should also be made use of for efficient codebook learning and local feature encoding. Third, many previous practices only try to reduce the information loss during local feature encoding without considering the codebook generation process. If we can

* Corresponding author.
E-mail addresses: cjzhang@jdl.ac.cn (C. Zhang), xaioxain@gmail.com (X. Xiao), junbiao_pang@bjut.edu.cn (J. Pang), cliang@whu.edu.cn (C. Liang), yfzhang@nlpr.ia.ac.cn (Y. Zhang), qmhuang@jdl.ac.cn (Q. Huang).

http://dx.doi.org/10.1016/j.jvcir.2014.05.010
1047-3203/

generate the codebook and encode local features by jointly reducing the information loss, we will be able to get more representative codebook and encoding parameters simultaneously.

To preserve the spatial information of local features, spatial pyramid matching (SPM) is introduced by Lazebnik et al. [11] and is widely used by researchers. This method partitions an image into increasingly finer sub-regions of different scales with $2^l \times 2^l,\ l = 0, 1, 2$. The histogram representation of each sub-region is then concatenated to form the final image representation. Many works [12–16] have demonstrated the effectiveness of this method for visual applications, such as image classification, image/video retrieval. In cases only the first scale $l = 0$ is used, SPM will reduce to BoW. Researchers have empirically found that k-means clustering based codebook generation method should be combined with SPM and non-linear kernels for efficient image classification. However, the computational cost is high. To speed up the computation and improve the performance, Yang et al. [2] proposed to use sparse coding with linear SVM classifier instead. Inspired by this, a lot of works were proposed [4–6] which further improve the performance. However, local features are still treated equally. Besides, the partition of local feature space information during the codebook generation process is still not considered by these methods.

To solve the problems mentioned above, in this paper, we propose a novel image representation method by going one step beyond visual word ambiguity and consider the governing regions of visual words generated by weighted local features. Each local feature is weighted according to specific visual application task by using the output of pre-trained traditional classifiers [1]. Each weighted local feature is then encoded not only by considering its similarities with visual words, but also by visual words' governing regions. A visual word's governing region is defined as the area covered by local features that are encoded with this visual word. We first find the local features nearby used during the codebook generation process, each weighted local feature is encoded by considering its similarities between visual words and the coding parameters of these nearby neighbors. This governing region based encoding scheme is consistently used both during the codebook generation and local feature encoding processes. We propose a weighted feature sign search algorithm to solve this governing region based weighted sparse coding problem. Experiments on several public datasets demonstrate the effectiveness of the proposed governing region based weighted sparse coding algorithm for image representation.

The rest of this paper is organized as follows. Section 2 gives the related work. In Section 3, we give the details of the proposed weighted local feature encoding by governing region method for image representation and conduct experiments on several public datasets to demonstrate its effectiveness in Section 4. Finally, we conclude in Section 5.

2. Related work

Inspired by text analysis, the bag-of-visual word model (BoW) [1] has been widely used for visual applications. One important component of the BoW model is the generation of codebook and encoding of local features. Traditional BoW model used the k-means clustering algorithm and viewed the cluster centers as visual words. Local features are then quantized to the nearest visual word. However, this scheme causes sever information loss which limits its discriminative power. To reduce the information loss during the local feature encoding process, a lot of works have been proposed [2–10]. Yang et al. [2] proposed to use sparse coding for codebook generation. Linear classifier was then trained to save computational cost. Gemert et al. [3] explored the visual word ambiguity problem and tried to quantize local features softly with codebook uncertainty. Wang et al. [4] found that locality is also very important and proposed the locality constrained linear coding algorithm which can be viewed as an extension of the sparse coding algorithm [2]. To consider the smooth property of sparse coding parameters, Gao et al. [5] proposed a Laplacian sparse coding algorithm and improved the image classification performance. The sparse coding along with max pooling strategy is suboptimal because negative coding parameters are useless for image representation. To solve this problem, non-negative sparse coding along with max pooling is proposed by Zhang et al. [6]. Liu et al. [7] conducted experiments on several public datasets and found soft assignment is very necessary for local feature encoding. Perronnin and Dance [8] proposed to use the fisher kernel for codebook construction which extended the BoW model by going beyond counting the zero order statistics and tried to encode second order statistics. Jurie and Triggs [9] proposed an acceptance-radius based clustering method to generate better codebooks than k-means clustering. To avoid the heavy computation of k-means clustering, Randomized clustering forest was proposed by Moosmann et al. [10] which speeded up the computation and improved the final image classification performance.

The traditional BoW model lacks the spatial information, inspired by the work of Grauman and Darrell [12] and Lazebnik et al. [11] proposed the spatial pyramid matching (SPM) algorithm which was widely used by researchers. Motivated by the SPM algorithm, a lot of works [13–16] have been done to combine the spatial information of local features. Chen et al. [13] proposed a hierarchical matching method with side information and used it for image classification. A weighting scheme was used to select discriminative visual words. Yao et al. [14] combined randomization and discrimination into a unified framework and used it for fine-grained image categorization. Zhang et al. [15] proposed to represent images with components and used a bilinear model for object recognition. A pose pooling kernel was proposed by Zhang et al. [16] to recognize sub-category birds.

To avoid the information loss, researchers also tried to classify images by using local features directly [17–19]. Boiman et al. [17] proposed to measure the similarities of local features and used it for image classification with encouraging results. To speed up the computation, Timofte and Gool [18] proposed to iteratively search for the nearest neighbors and used it for image classification and dimensionality reduction. A local naive Bayes nearest neighbor method [19] was also proposed by McCann and Lower. The use of neighbor information speeds up the computation and also helps to improve classification performance. In fact, the use of local neighbor information has been widely used for visual applications, such as locality constrained linear coding [4], Laplacian sparse coding [5] and kernel codebook [3].

The use of sparsity [20] has also been very popular in recent years. Sparse coding [21] is used by Yang et al. [2] for image classification. Sparse representation is used for robust face recognition by Wright et al. [22]. Yang et al. [23] combined fisher discrimination information with sparse coding to learn the dictionary for face, digit and gender classification. Structured sparse representation is also proposed by Elhamifar and Vidal [24] for robust face classification.

Researchers [13,25–33] have also tried to treat local features differently which has shown its effectiveness for various image classification tasks. Chen et al. [13] combined object confidence map and visual saliency map for side information which achieved good image classification performance. Fernando et al. [25] tried to fuse discriminative feature with a regression model to boost image classification performance. Cinbis et al. [26] viewed local features as non-iid sampled and used the fisher kernel to classify images. Sharma et al. [27] also explored discriminative saliency with spatial information. Moosmann et al. [28] tried to learn the saliency

map and use it for object categorization. Su and Jurie [29,30] proposed to solve the visual word disambiguation by semantic context modeling. Fei-Fei et al. [31] proposed an incremental Bayesian approach for the Caltech 101 dataset while Griffin et al. [32] evaluated the classification performance on the Caltech 256 dataset. Li et al. [33] tried to classify events by scene and object recognition. Using the SIFT feature [34] for local region description, many works [35–43] have been proposed. Fei-Fei and Perona [35] proposed a Bayesian hierarchical approach for scene classification. Oliva and Torralba [36] modeled the shape of the scene in a holistic way while Chatfield et al. [37] compared the performances of different methods and found the details are very important for the final classification performance. Rasiwasia and Vasconcelos [38] used a low semantic space based representation while Li et al. [39] proposed an ObjectBank representation. Wu and Rehg [40] also explored the histogram intersection kernel for codebook generation. He et al. [41] used Laplacian similarities for face recognition while Zhang et al. [42] combined the k nearest neighbor approach with SVM classifier training. Fisher kernel is also used to reduce the quantization loss during local feature encoding by Perronnin et al. [43].

3. Weighted local feature encoding with governing region for image representation

In this section, we give the details of the proposed governing region based weighted local feature encoding method for image representation.

3.1. Local feature encoding: k-means clustering, kernel codebook and sparse coding

Traditional BoW model generates codebook by k-means clustering and quantizes local feature by nearest neighbor based visual word assignment. Formally, let X be the set of D-dimensional local features, $X = [x_1, x_2, \ldots, x_N]^T \in \mathbb{R}^{D\times N}$ where N is the number of local features. The k-means clustering tries to finding the M cluster centers $C = [c_1, c_2, \ldots, c_M]^T$ by solving the following optimization problem as:

$$\begin{aligned} \min_{C,U} \quad & \sum_{i=1}^{N} \|x_i - u_i^T C\|^2 \\ \text{s.t.} \quad & Card(u_i) = 1, \quad |u_i| = 1,\ u_i \succeq 0, \quad \forall i \end{aligned} \tag{1}$$

where M is the codebook size, $U = [u_1, u_2, \ldots, u_N]^T$, $u_i \in \mathbb{R}^{M\times 1}$, $i = 1, \ldots, N$ are the cluster membership indicators. $Card(u_i) = 1$ means that only one element of u_i is nonzero. $|*|$ indicates the L_1 norm. During the codebook generation phase, C and U are optimized iteratively. After the codebook C has been learned, local features can then be quantized to the nearest visual word c_i.

The hard assignment of Problem 1 is too restrictive and causes sever information loss, especially when local features lie on the boundary of visual words. To alleviate this problem, Gemert et al. [3] proposed to use kernel codebook by softly assigning the local features. Instead of assigning each local feature to one visual word, the kernel codebook scheme tries to encode each local feature proportionally by measuring their similarities with all visual words as:

$$u_{i,j} = \frac{K_\sigma(D(x_i, c_j))}{\sum_{j=1}^{M} K_\sigma(D(x_i, c_j))}, \quad \forall i = 1, 2, \ldots, N, \quad j = 1, 2, \ldots, M \tag{2}$$

where $D(x_i, c_j)$ is the distance between x_i and c_j. $K(*)$ is a kernel function. Gemert et al. [3] used the Gaussian-shaped kernel $K_\sigma(x, c) = e^{-\frac{(x-c)^2}{2\sigma^2}}$, where σ is the corresponding smoothing parameter. Compared with hard assignment based model, the explicitly modeling of visual word ambiguity helps to improve the image representation power. Besides, it also performs consistently with the increase of codebook size. However, this soft assignment strategy is only applied during the local feature encoding phase while the codebook is still learnt by traditional k-means clustering. Moreover, both k-means clustering method [1] and the visual word ambiguity method [3] ignore the governing regions of each visual word and only uses the clustering centers instead of the partition of local feature space for local feature encoding.

Sparse coding is proposed as another way to reduce the information loss of hard assignment [2]. The use of sparsity also helps to avoid each feature being assigned to too many visual words. This is achieved by minimizing the reconstruction error with encoding parameter sparsity over the codebook C and encoding parameters U as:

$$\begin{aligned} \min_{C,U} \quad & \sum_{i=1}^{N} \|x_i - u_i^T C\|^2 + \lambda \|u_i\|_1 \\ \text{s.t.} \quad & \|c_j\| \leqslant 1, \quad j = 1, 2, \ldots, M \end{aligned} \tag{3}$$

where λ is the sparse regularization parameter. The unit L_2 norm is applied to avoid trivial solutions. Similar with k-means clustering and kernel codebook, the sparse coding algorithm also has a training phase and a coding phase. During the codebook learning process, the optimization of Problem 3 is alternatively solved over codebook C and sparse coding parameters U while keeping the other fixed. After the codebook has been learned, local features can be encoded by fixing codebook C. The joint optimization over all local features is very hard to solve. Hence, local features are often encoded individually over each local feature x_i with encoding parameter u_i in practice as:

$$\min_{u_i} \|x_i - u_i^T C\|^2 + \lambda \|u_i\|_1, \quad \forall i \tag{4}$$

The sparse coding alleviate the information loss caused by hard assignment. Besides, sparsity constraints also help to choose the most representative visual words instead of using all of them for local feature encoding. Another advantage of sparse coding is the objective function consistency for both the codebook generation and local feature encoding. However, the sparse coding only uses the visual words instead of the partition of local feature space for local feature encoding. If we can make use of this partition information, we will be able to further improve the coding efficiency which eventually helps to boost the final image representation power.

3.2. Learning local feature's weight

For a specific task, different local features are not equally important. Some local features are not very necessary or even can be viewed as noise while representative local features should be explored more efficiently. Hence it is necessary to weight local features for efficient image representation.

In this paper, we use a simple but efficient method to weight local features. This method uses the state-of-the-art image application method (*e.g.* image classification, object detection) to evaluate the discriminative power of each local feature. Take the two-class classification task for example, we first train SVM classifier using the sparse coding along with max pooling based image representation proposed by Yang et al. [2]. For each image, we densely sample sub image regions (64×64 pixels) with overlap. Images are fist resized to the same size. The sub-region number is set to $P \times Q$ for each image (20×15 in this paper), as Zhang et al. [15] did. We then use the learned classifier to predict this sub-region's category and assign each local feature within this sub-region with the same category. Formally, let y^i be the output of the learnt classifier for sub-region i. Then the m_i local features $X^i = \left[x_1^i, x_2^i, \ldots, x_{m_i}^i\right]$ within this sub-region are all set to the value y^i. If a local feature

j belongs to more than one sub-region, its category prediction value is aggregated over all sub-regions as $y_j = \sum_{i=1}^{n_i} y^i$ where n_i is the sub-region number this local feature belongs. The larger the $|y_j|$ is, the more discriminative this local feature is. We use the upper indices to denote the index of sub-regions and use the lower indices to denote the training images. The smaller the $|y_j|$ is, the less discriminative the local feature is. Hence the weight of each local feature can be defined as $w_j = e^{\frac{y_j}{\sigma_1}}$. We use the median of all y_j as σ_1. In this way, we are able to make use of the discriminative power of each local feature differently and help to generate codebook and encode local features more effectively and efficiently.

3.3. Governing region based weighted local feature encoding

Although proven very effective, the state-of-the-art local feature encoding algorithms only use the visual words for quantization without paying too much attention to the governing regions of visual words. We define a visual word's governing region as the area covered by local features that are encoded with this visual word. For the visual word generated by k-means clustering, the visual word's governing region corresponds to the hard segmented local feature space for this corresponding visual word and does not overlap with each other. For the soft assignment based visual word generation methods (e.g. kernel codebook/sparse coding), the governing regions of visual words are overlapped because of the soft assignment strategy. However, what we learned during the codebook generation process is a partition of the local feature space, not merely the visual words (or cluster centers for k-means). As pointed out by many researchers [8–10], these local feature partitions are often unbalanced. If we only use the visual words to encode local features, the governing region information of visual words will be lost. Fig. 1 shows a toy example of this problem. Suppose we have learned a codebook of two visual words with the corresponding partition of local feature space separated by the black curve in Fig. 1. Local features A and B should be quantized to visual words 2 and 1 respectively based on the local feature space partition. However, for nearest neighbor assignment strategy, one local feature will be encoded based on the relative position of this local feature with the perpendicular bisector (denoted by the red line) between the two visual words. Hence, local features A and B are quantized to visual words 2 and 1 for the nearest neighbor assignment strategy. The use of large codebook helps to alleviate this information loss while the problem remains unsolved. The same problem also happens both for the kernel codebook and sparse coding based local feature encoding algorithms. This may decrease the representative ability of encoded parameters which further hinders the final image representation power and finally decrease the image application performance.

To solve the mentioned problems above and encode local features more effectively and consistently, in this subsection, we introduce the governing region based weighted local feature encoding algorithm. We use the weighted local features for efficient encoding. To consider the governing region information of visual words, besides calculating the similarities of local features with each visual word, we also use the encoding information of visually similar local features used during the codebook generation process. Formally, let x_i be the i-th local feature to be encoded, u_i is the corresponding encoding parameter, $i = 1, 2, \ldots, N$. x_j, $j = 1, 2, \ldots, k$ be the k nearest neighbors of x and u_j, $j = 1, 2, \ldots, k$ are the corresponding encoding parameters. Sparsity constraint is also used to choose discriminative visual words for encoding. The proposed governing region based weighted local feature encoding algorithm tries to find the optimal encoding parameters $u_i, i = 1, 2, \ldots, N$ and the codebook C by solving the following optimization problem as:

$$\begin{aligned} \min_{C,U} \quad & \|wX - UC\|^2 + \beta\|U\|_1 \\ & + \frac{\alpha}{2}\sum_{i=1}^{N}\sum_{j=1}^{k}(u_i - u_j)^2 e^{-\frac{(w_i x_i - w_j x_j)^2}{\sigma^2}} \\ \text{s.t.} \quad & \|c_j\| \leqslant 1, \quad j = 1, 2, \ldots, M \end{aligned} \tag{5}$$

where σ is the smoothing parameter, $w = [w_1, w_2, \ldots, w_n]$ are local features' weights. α controls the influence of visual word's governing regions. β is the sparse regularization parameter which controls the sparsity of U. $\|c_j\| \leqslant 1$ is used to avoid trivial solutions. The third term in the above optimization problem is used to combine the governing region of visual words. In this way, we are able to combine the reconstruction of weighted local feature and the governing regions of visual words along with sparsity constraints into a unified framework.

Problem 5 is not convex over U and C simultaneously, but is convex for U/C while fixing C/U. Hence, we optimize U and C alternatively while keeping the other fixed. When C is fixed, the optimization of Problem 5 over the encoding parameters U equals to:

$$\min_{U} \|wX - UC\|^2 + \frac{\alpha}{2}\sum_{i=1}^{N}\sum_{j=1}^{k}(u_i - u_j)^2 e^{-\frac{(w_i x_i - w_j x_j)^2}{\sigma^2}} + \beta\|U\|_1 \tag{6}$$

The joint optimization over all local features U simultaneously is often very hard or impossible. An alternative way is to encode each local feature $x_i, i = 1, \ldots, N$ with the corresponding parameter u_i individually. In this way, we are able to solve Problem 6 over each local feature x_i by:

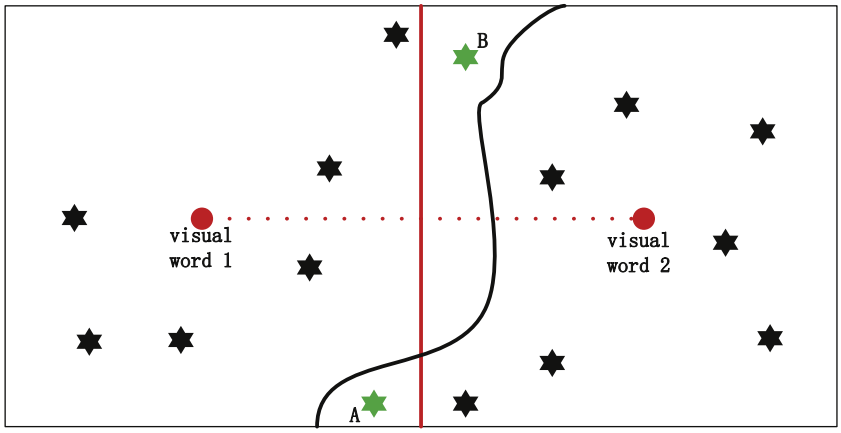

Fig. 1. A toy example showing the problem with traditional visual word only based local feature encoding method. Local features A and B should be quantized to visual words 2 and 1 respectively based on the local feature space partition. However, based on nearest neighbor based assignment strategy, local features A and B are quantized to visual words 2 and 1 respectively. It is best viewed in color. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$$\min_{u_i}\|wx_i - u_i^T C\|^2 + \frac{\alpha}{2}\sum_{j=1}^{k}(u_i - u_j)^2 e^{-\frac{(w_i x_i - w_j x_j)^2}{\sigma^2}} + \beta\|u_i\|_1,$$
$$\forall i = 1, 2, \ldots, N \tag{7}$$

When the encoding parameter U is fixed, the optimization of Problem 5 over the codebook C equals to:

$$\min_{C} \quad \|wX - UC\|^2 \tag{8}$$
$$\text{s.t.} \quad \|c_j\| \leqslant 1, \quad j = 1, 2, \ldots, M$$

$\|c_j\| \leqslant 1$ is used to avoid trivial solutions. This is a least squares problem which can be solved by using the Lagrange dual method proposed by [21].

Due to the amount of local features, it is computational impossible to use all the local features for codebook generation. In practice, researchers [2–6] often randomly select a portion of local features instead. A codebook is generated using these randomly selected local features and all the local features are then encoded with the learnt codebook. During the codebook generation process, we iteratively optimize over problem (6) and problem (8) until the maximum number of iteration is achieved or the objective function value of problem (5) falls below a pre-defined threshold γ. After the codebook has been learnt, we can encode each local feature by solving problem (7). Algorithm 1 gives the proposed governing region based codebook generation algorithm.

Algorithm 1. The proposed governing region based codebook generation algorithm.

Input:
The local features X, α, β, threshold parameter γ and max iteration number *maxiter*;
Output:
The learned codebook C and coding parameters U;
for $iter = 1, 2, \ldots, maxiter$
Find the optimal encoding parameters U with codebook C fixed by solving Problem 6; this is achieved by encoding each local feature individually by solving Problem 7;
Find the optimal codebook C with U fixed by solving Problem 8;
Check whether the decrease of objective function of Problem 5 falls below the threshold γ.
If unsatisfied
go to step 1
Else
stop;
Return: U, C;

To find the optimal encoding parameter u_i for local feature x_i, we need to solve Problem 7. Let $u_{i,m}$ denote the m-th coefficient of u_i, $i = 1, 2, \ldots, N$, $m = 1, 2, \ldots, M$. Problem 7 can be rewritten as:

$$\min_{u_i}\|wX - u_i^T C\|^2 + \frac{\alpha}{2}\sum_{j=1}^{k}(u_i - u_j)^2 e^{-\frac{(w_i x_i - w_j x_j)^2}{\sigma^2}} + \beta\sum_{m=1}^{M}|u_{i,m}| \tag{9}$$

Note that if we know the signs of each $u_{i,m}$, we will be able to replace $|u_{i,m}|$ with $u_{i,m}$ (if $u_{i,m} \geqslant 0$), or $-u_{i,m}$ (if $u_{i,m} < 0$). This reduces Problem 9 to a simpler problem which can be solved efficiently. Hence, we follow the feature sign search strategy proposed in [21] and try to search for the signs of $u_{i,m}$, $m = 1, 2, \ldots, M$. Based on these signs, we can efficiently solve Problem 9. Besides, because Problem 9 is convex over u_i, this search strategy will systematically reduce the objective value of Problem 9 and refines the signs of $u_{i,m}$ if it is set incorrect initially.

To simplify representation, let

$$f(u_{i,m}) = \frac{\partial\|w_i x_i - u_i^T C\|^2}{\partial u_{i,m}} + \frac{\partial\frac{\alpha}{2}\sum_{j=1}^{k}(u_i - u_j)^2 e^{-\frac{(w_i x_i - w_j x_j)^2}{\sigma^2}}}{\partial u_{i,m}} \tag{10}$$

We give the details of the weighted feature sign search strategy for solving Problem 9 in Algorithm 2. We maintain an active set of nonzero coefficients and their signs. In each step, the optimal active set and coefficients' signs are systematically learned by reducing the objective values of Problem 9. This is achieved by computing the new solution $\hat{u}_i^{new}$ to Problem 9 based on current active set and signs. A discrete line search between the current solution u_i and $\hat{u}_i^{new}$ is then made. The solution, active set and signs are then updated accordingly. We also give the theoretical justifications in Section 3.4 that the proposed algorithm consistently reduces the objective value of Problem 9 and the overall algorithm systematically reduces the objective value of Problem 7 which eventually helps to minimize the objective value of Problem 5. The locality constraint [4] is also used to speed up computation and make use of the locality constraints.

Algorithm 2. The proposed weighted feature sign search algorithm for solving Problem 9.

Input:
The local feature $x_i, x_j, C, \alpha, \beta, w$, initialize
$u_i = \vec{0},\ \theta = \vec{0},\ activeset = \{\}$, where
$\theta = sign(u_i),\ \theta_m \in \{-1, 0, 1\}$;
Output:
The learned encoding parameter u_i;
For zero coefficients of u_i, select
$m = argmax_m|f(u_{i,m})|$.
Activate $u_{i,m}$ only if it locally reduces the objective as:
(i): If $f(u_{i,m}) > \beta$, then set
$\theta_m = -1,\ activeset = \{m\} \cup activset$.
(ii): If $f(u_{i,m}) < -\beta$, then set
$\theta_m = 1,\ activeset = \{m\} \cup activset$.
Feature sign search:
(i): Let $\widehat{C}$ be a submatrix of C which is made up only by the columns corresponding to the active set. $\widehat{u}_i$ and $\hat{\theta}$ be the corresponding subvectors of u_i and θ.
(ii): Finding the optimal solution to the unconstrained optimization problem as:
$\hat{u}_i^{new} = argmin_{\hat{u}_i^{new}} f(\widehat{u}_i) + \beta\hat{\theta}^T\widehat{u}_i$
(iii): Perform a discrete line search on the closed line segment from u_i to $\hat{u}_i^{new}$.
(iv): Check the objective value at $\hat{u}_i^{new}$ and all points where any coefficient changes sign and update $\widehat{u}_i$ to the point with the lowest objective value. Then remove zero coefficients of $\widehat{u}_i$ from the active set and update $\theta = sign(u_i)$
Check the optimality conditions:
(1): **For** nonzero coefficients:
$f(u_{i,m}) + \beta sign(u_{i,m}) = 0,\ \forall u_{i,m} \neq 0$
If not satisfied, go to step 3; **else** check condition (2).
(2): For zero coefficients:
$|f(u_{i,m})| \leqslant \beta,\ \forall u_{i,m} = 0$
If not satisfied, go to step 2; **else** go to step 5.
Return: u_i;

3.4. Convergence of Algorithms 1 and 2

In this subsection, we give the theoretical justification of the convergence of the proposed governing region based codebook generation and weighted local feature encoding algorithm. u_i is called consistent with a given active set and sign vector θ if m is in the active set, then $sign(u_{i,m}) = \theta_m$ and if m is not in the active set, then $u_{i,m} = 0$.

Lemma 3.1. *Consider optimizing over Problem 9 with u_i is consistent with a given active set and sign vector. If the current coefficients u_i^c are consistent, but are not optimal for the augmented problem at the start of Step 3 in Algorithm 2, then the weighted feature sign search is able to reduce the objective value of Problem 9.*

Proof. Let $\widehat{u_i}$ be the subvector of u_i corresponding to coefficients in the given active set. Since $\widehat{u_i}$ is not an optimal point for Problem 9. There are two possible cases: (i) if $\hat{u}_i^{new}$ is consistent with the give active set, updating $\hat{u}_i := \hat{u}_i^{new}$ decreases the objective function of Problem 9; (ii) if $\hat{u}_i^{new}$ is not consistent with the active set, let $\hat{u}_i^d$ be the first zero-crossing point on a line segment from $\hat{u}_i$ to $\hat{u}_i^{new}$, then the discrete line search described in step 3 of Algorithm 2 decreases the objective value of Problem 9. □

Lemma 3.2. *Consider optimizing over Problem 9 with u_i is consistent with a given active set and sign vector. if the current coefficients u_i^c are optimal at step 2 in Algorithm 2, but are not optimal for Problem 9, then the weighted feature sign search algorithm is able to reduce the objective value of problem 9.*

Proof. Since u_i^c are optimal at step 2 in Algorithm 2, it satisfies the optimality condition (1) but not (2). Thus, in step 2, there is some m is activated and added to the active set. In step 3, note that a Taylor expansion of the objective function around u_i^c has a first order term in u_i only, we have that any direction that decreases the objective function must be consistent with the active set. Besides, since u_i^c is not an optimal point for (2), it must decrease local near u_i^c along the direction from u_i^c to u_i^{new}. Obviously, when u_i^c is consistent with the active set, either u_i^{new} is consistent, or the first zero-crossing from $\hat{u}_i$ to $\hat{u}_i^{new}$ has a lower objective value. □

Theorem 3.3. *Algorithm 2 of the optimization of Problem 9 converges to a global optimum in a finite number of steps.*

Proof. From Lemmas 1 and 2, it follows that the proposed weighted feature sign search algorithm always strictly reduce the objective value of Problem 9. At the start of step 2 in Algorithm 2, u_i either satisfies optimality condition iv(4) or is $\vec{0}$. In either case, u_i is consistent with the current active set and sign vector and must be optimal for the problem described in the above lemmas. Besides, as the number of possible active sets is finite and since the objective value is strictly decreasing, the outer loop of steps ii–iv(2) cannot repeat indefinitely and will reach step iv(2) in a finite number of steps.

Based on Lemmas 3.1 and 3.2 and Theorem 3.3, we are able to prove the convergence of Algorithm 1. □

Theorem 3.4. *The alternative optimization of codebook C and encoding parameters U while keeping the other fixed is able to reduce the objective value of Problem 5, hence is able to converge in a finite number of steps.*

Proof. For each local feature x_i, finding its governing region based encoding parameter by Algorithm 2 reduces the objective function of Problem 5. Since we optimize each local feature separately, the objective function of Problem 5 will be consistently reduced. Hence step 1 of Algorithm 1 reduces the objective value of Problem 5. When the encoding parameters U are fixed, solving Problem 8 also reduces the objective value. Besides, since we alternatively optimize over U and C while keeping C and U fixed, we are able to reduce the objective value of problem 5 alternatively. Moreover, since the objective value of Problem 5 is lower bounded non-negative and each step of Algorithm 1 reduces the objective value of Problem 5. Hence Algorithm 1 will converge in a finite number of steps. □

3.5. Representing images by max pooling along with spatial pyramid matching

After encoding each local feature with governing regions, we can use the encoding parameters for image representation. As proven by many researchers [2,4–6], the sparse coding parameters should be used with max pooling for image representation in order to achieve good image classification performance. The max pooling strategy is inspired by biophysical studies in visual cortex [20] and empirically proven. It can also help to get rid of some noise response which often has small coding parameters. Hence, we use max pooling to extract the useful information from the governing region based sparse coding parameters. Formally, let $[u_1, u_2, \ldots, u_P]$ be the P coding parameters of local features within the selected image region, h is the extracted representation of this region as:

$$h = \max\{|u_1|, |u_2|, \ldots, |u_P|\} \tag{11}$$

Besides, to consider the spatial information of local features, we follow the spatial pyramid matching method by using the first three layers ($L = 0, 1, 2$), as Lazebnik et al. [11] did. Note that the proposed governing region based weighted sparse coding algorithm can also be combined with other more efficient spatial methods to further improve the final image representation power, such as [13]. The ScSPM [2] can be viewed as a special case of the proposed method when the governing region and reweighting scheme are not considered. If we simply set the weights to one, the proposed method will degenerate to the method proposed in [5].

4. Experiments

To evaluate the effectiveness of the proposed governing region based codebook generation and weighted local feature encoding method (WGRSPM) for image representation, We choose to only conduct image classification experiments on several public datasets because of the page limits. Note that the proposed WGRSPM can also be applied to other visual applications such as image retrieval and object detection. The datasets are the Scene 15 dataset [11] by Lazebnik et al. [11], the Caltech 101 dataset by Fei-Fei et al. [31], the 256 dataset by Griffin et al. [32] and the UIUC-Sport dataset by [33].

4.1. Parameters setting

Local features play a very important role for efficient image representation. Choosing the proper local features helps to improve the final image classification performance substantially. In this paper, we follow the successful work of [1–3] and choose the SIFT descriptors [34] as our local feature descriptor. We densely extract SIFT features on overlapping 16×16 pixels with an overlap of 6 pixels. The dense extraction of local features has been shown more

effective than interest point detectors. We normalize these extracted SIFT features with L_2 norm. We randomly select the training images and use the rest images for testing. This process is repeated for six times to get reliable results. We use the mean of per-class classification rates for performance measurement and report the final results by the mean and standard deviation of the classification rates. The one-vs-all linear SVM classifier is used for classification because of its advantages with max pooling based image representation. We use fivefold cross validation to choose the optimal parameters. The maximum iteration number during the governing region based codebook generation process is set to 30. The codebook size is set to 1024 for all these datasets, as [2,4,5] did. SPM with three pyramids is used to combine the spatial information of local features. Hence each image is represented with a vector of $21 \times 1024 = 21{,}504$ dimensions. We choose to compare with other methods by their reported results instead of re-implementing them and test on every datasets for fair comparison.

The sparsity parameter β is the most important parameter in our algorithm. A larger β ensures more sparse solution than smaller β. Yang et al. [2] empirically found that good results can be achieved when β is fixed to be 0.3–0.4. We follow this setup and set $\beta = 0.4$ for the Scene 15 dataset and the Caltech 101 dataset. $\beta = 0.3$ is set for the Caltech 256 dataset and the UIUC Sport dataset.

4.2. Analysis on the Scene 15 dataset

We first consider the Scene 15 dataset which is collected by [11,35,36]. There are 4485 images of 15 categories (*bedroom, coast, forest, highway, industrial, insidecity, kitchen, livingroom, mountain, office, opencountry, store, street, suburb* and *tallbuilding*) in this dataset. Each category has 200–400 images. Fig. 2 shows some example images of the Scene 15 dataset. We follow the same experimental setup as Lazebnik et al. [11] did and randomly select 100 images per class for classifier training and use the rest images for testing.

Because of computational cost, we randomly choose a subset of local features for codebook generation, as [2,4–6] did. Hence, the number of local features may affect image classification accuracy. Intuitively, the more local features we use, the more accurate the final classification performance is. However, the computational cost also increases with the number of local features. Hence it is necessary to explore the influence of the number of local features. Fig. 3 shows the mean classification accuracies with different amount of local features. We can see from Fig. 3 that the increase of local features helps to boost the final classification performance. However, as the number of local features increases from 3×10^4 to 5×10^4, the classification performance becomes relatively stable. To get comparable results and reduce the computational cost, we choose 5×10^4 local features to generate codebook and encode local features for all the datasets in this paper. Another advantage of the proposed governing region based image classification method is the use of relatively less local features which is still able to get comparable results with others [2,4,5]. We believe this is because of the consideration of governing regions of visual words instead of using the visual words only which allows us to generate codebook and encode local features more smoothly and efficiently.

During the local feature encoding process, α and k jointly control the neighbor's influences. Fixing k, a larger α imposes bigger impact of neighbors compared with a smaller α. Fixing α, a larger k means we can consider more nearby local features' influence compared with a smaller k. Hence, it is necessary to explore the influence of different α and k. Our experimental results suggest

Fig. 2. Example images (Scene 15 dataset).

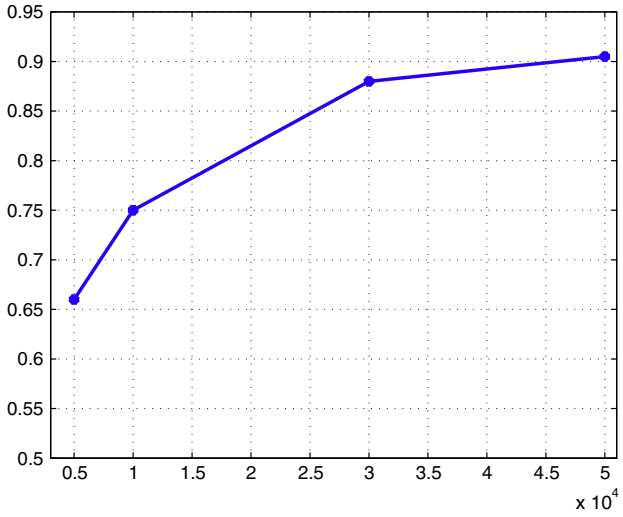

Fig. 3. The influence of the number of local features used for codebook generation and encoding. The vertical axis indicates the classification rate while the horizontal axis indicates the number of local features.

that choosing $k = 5$ (with $\alpha = 0.1$) local features achieves the best performances when using $10^4, 3 \times 10^4$ and 5×10^4 local features respectively. This is in accordance with the experimental setup as [4,5] did. The local feature's encoding information and its neighbors' information should be jointly considered in order to improve image classification performance. When a large α and k are used, the performance would decrease dramatically. This is because information is lost if we replace a local feature with its nearby local features, as Chatfield et al. [37] found.

Finally, we give the performance of the proposed weighted governing region based image classification method and compare with methods proposed by [2,3,5,11,38–40] in Table 1. Yang et al. [2] used the sparse coding alone with max pooling technique while Gemert et al. [3] used soft-assignment of local features. Gao et al. [5] explored the smooth relationship in the encoding parameters. Lazebnik et al. [11] used the spatial pyramid technique to combine the spatial relationship of local features. Rasiwasia and Vasconcelos [38] used semantic based image representation while Li et al. [39] used web based semantic representation. Wu et al. [40] used histogram intersection kernel for codebook generation. We choose these methods for comparison because they are closely related with the proposed method or are the-state-of-the-art methods. We use the reported results instead of re-implementing them for fair comparison with the same experimental setting. We also give the classification results without local feature weighting (GRSPM). We can see from Table 1 that the proposed WGRSPM algorithm achieves comparable results with other methods which shows the effectiveness of the proposed method. Compared with other methods [2,3,11,38] which only used the visual words information for local feature encoding, the proposed method uses the partition of local feature space which is more accurate and representative, hence helps to encode local features more efficiently. Besides, the proposed WGRSPM also outperforms the soft assignment based methods [2,3]. Moreover, The 3.5% improvement of WGRSPM over GRSPM also demonstrates the effectiveness of the weighting scheme. Since the weighting parameters are determined by their discriminative power, we are able to encode local features more robustly and get ride of the noisy local features. However, the GRSPM does not work as good as LScSPM which used Euclidean distance for reconstruction and histogram intersection similarity for Laplacian graph construction. We believe this is because local features are not randomly generated but inherently related [41]. A more appropriate distance measurement method which can make full use of the local features' inherent information should be used instead of only using Euclidean distance. However, this is a problem which is still unsolved and we will leave it to our future work.

Table 1
Performance comparison (Scene 15 dataset). The bold values indicates the best performances.

Algorithm	Performance
KSPM [2]	76.73 ± 0.65
ScSPM [2]	80.28 ± 0.93
KSPM [11]	81.40 ± 0.50
LSS [38]	72.20 ± 0.20
OB [39]	80.9
DSS [27]	85.50 ± 0.60
KCSPM [3]	76.70 ± 0.40
HIK + OCSVM [40]	84.00 ± 0.46
LScSPM [5]	89.75 ± 0.50
GRSPM	86.52 ± 0.45
WGRSPM	**90.06 ± 0.53**

4.3. Caltech 101 dataset

The Caltech 101 dataset has 8677 images of 101 classes. The number of images per class varies from 31 to 800 images. Objects are often in the center of images. Some images are manually rotated and cropped after alignment. The background noise in the Caltech 101 dataset is relatively small compared with the Caltech 256 dataset. Fig. 4 shows some example images of the Caltech 101 dataset. We randomly choose 15, 30 images per class for training and use the rest images for testing. 5×10^4 local features are randomly chosen for codebook generation, as did on the Scene 15 dataset.

Table 2 gives the image classification performance comparison of the proposed WGRSPM with other methods [2–4,11,17,42]. Wang et al. [4] used locality constraints for sparse coding along with max pooling approach. Boiman et al. [17] used local features directly for classification with heavy computational cost. Zhang et al. [42] combined discriminative nearest neighbor with SVM classifier training for improving the classification performance. We choose these methods for comparison because they are closely related with the proposed method or are the-state-of-the-art methods. We can have similar conclusions as on the Scene 15 dataset. The proposed governing region based weighted local feature encoding method treats local features discriminatively and uses the local feature space partition information for local feature encoding. This scheme helps to get more representative encoding parameters which eventually improves the image classification performance, hence helps the WGRSPM algorithm outperform the soft assignment and sparse coding based methods [2–4] which only used the visual words information for local feature encoding. Besides, the WGRSPM algorithm also outperforms the NBNN [17] algorithm which uses the local features directly for image classification. The direct use of local features totally avoids the information loss during local feature encoding process. However, NBNN cannot take advantage of the discriminatively trained classifiers which also limits its performance. The WGRSPM algorithm is able to reduce the information loss by going beyond visual words and consider visual words' governing regions, besides, it also can take advantage of the well trained classifiers for robust and effective image classification.

4.4. Caltech 256 dataset

The Caltech 256 dataset has 256 categories of 29,780 images. Each category has at least 80 images. Compared with the Caltech

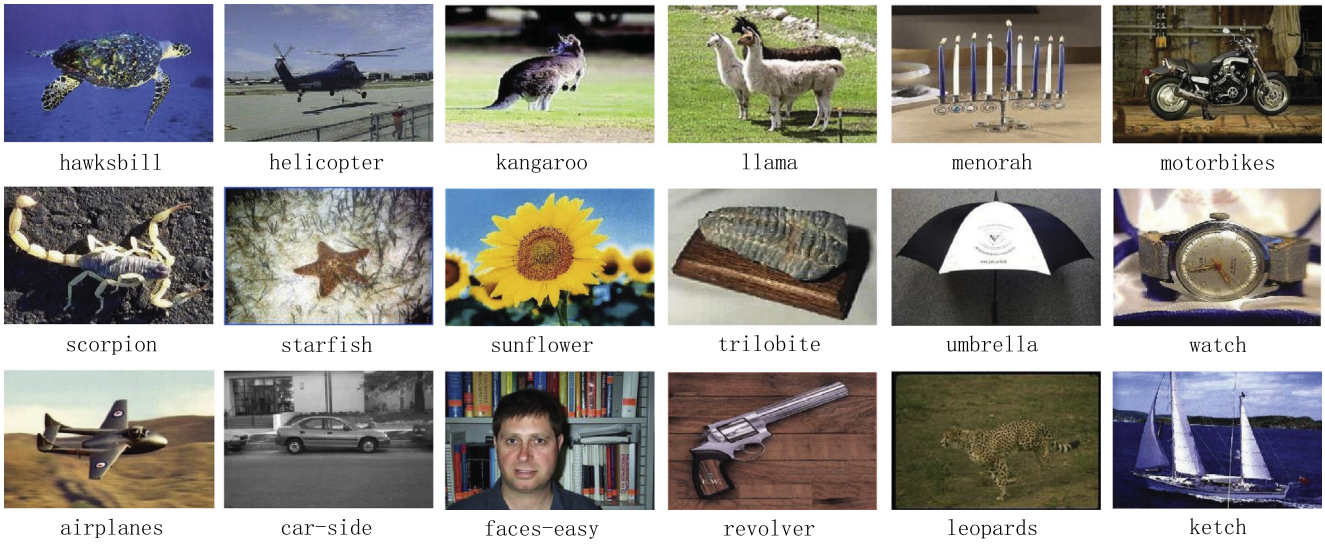

Fig. 4. Example images (Caltech 101 dataset).

Table 2
Performance comparison (Caltech-101 dataset). The bold values indicates the best performances.

Algorithm	15 training	30 training
ScSPM [2]	67.00 ± 0.45	73.20 ± 0.54
KCSPM [3]	–	64.14 ± 1.18
LLC [4]	65.43	73.44
KSPM [11]	56.40	64.40 ± 0.80
NBNN [17]	65.00 ± 1.14	70.40
SVM-KNN [42]	59.10 ± 0.60	66.20 ± 0.50
GRSPM	68.45 ± 0.52	74.18 ± 0.65
WGRSPM	**69.39 ± 0.46**	**74.95 ± 0.78**

Table 3
Performance comparison (Caltech-256 dataset). The bold values indicates the best performances.

Algorithm	15 training	30 training	45 training
KSPM [2]	23.34 ± 0.42	29.51 ± 0.52	–
ScSPM [2]	27.73 ± 0.51	34.02 ± 0.35	37.46 ± 0.55
LScSPM [5]	30.00 ± 0.14	35.74 ± 0.10	38.54 ± 0.36
NBNN[17]	30.45	38.18	–
KSPM [32]	–	34.10	–
LLC [4]	34.36	41.19	45.31
KCSPM [3]	–	27.17 ± 0.46	–
IFK [43]	34.70	40.80	45.00
GRSPM	35.45 ± 0.50	41.39 ± 0.48	45.47 ± 0.48
WGRSPM	**36.32 ± 0.48**	**41.70 ± 0.51**	**45.62 ± 0.59**

101 dataset, images within the Caltech 256 dataset are more diverse and are with larger intra-class variances. This makes the Caltech 256 dataset harder to classify than the Caltech 101 dataset. Fig. 5 shows some example images of the Caltech 101 dataset. 5×10^4 local features are randomly chosen for codebook generation, as did on the Scene 15 dataset. We randomly choose 15, 30, 45 images per class for training and use the rest images for testing.

We give the performance comparison in Table 3. The proposed WGRSPM achieves comparable performance with other methods [2–5,17,19,43] which demonstrates the effectiveness of the proposed WGRSPM method. McCann and Lowe [19] used local naive Bayes based nearest neighbor for classification while Perronnin

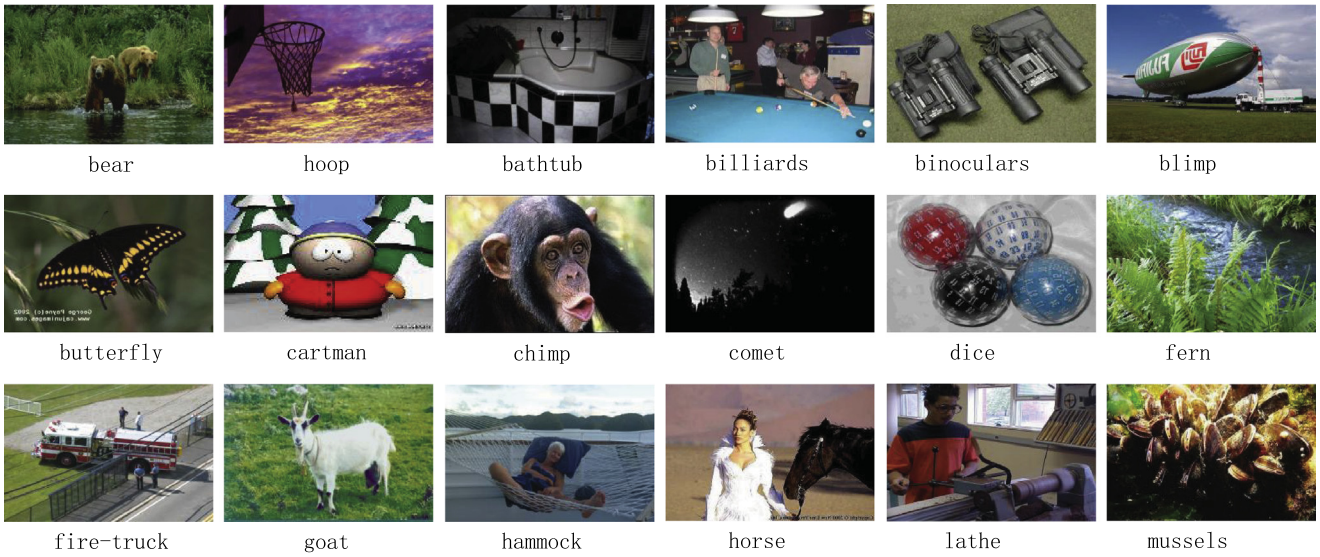

Fig. 5. Example images (Caltech 256 dataset).

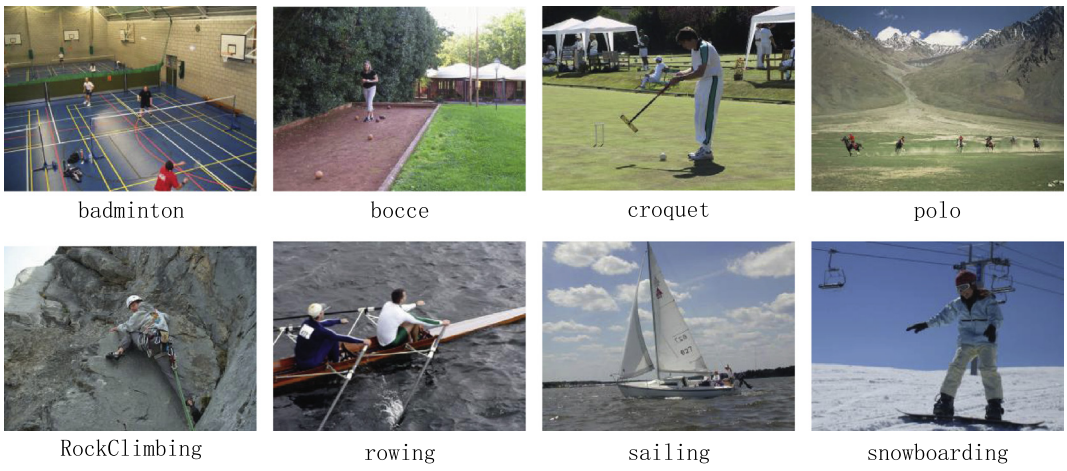

Fig. 6. Example images (UIUC-Sport dataset).

Table 4
Performance comparison (UIUC-Sport dataset). The bold values indicates the best performances.

Algorithm	Performance
ScSPM [2]	82.74 ± 1.46
HIK + OCSVM [40]	83.54 ± 1.13
LScSPM [5]	85.31 ± 0.51
Sc^{+}SPM [6]	83.77 ± 0.97
GRSPM	86.49 ± 0.75
WGRSPM	**87.25 ± 0.82**

et al. [43] tried to encode local features with more information using fisher kernels. We choose these methods for comparison because they are closely related with the proposed method or are the-state-of-the-art methods. Compared with ScSPM [2], KSPM [32] and LLC [4], the consideration of visual word's governing region information boosts the representation power and improves the classification accuracy. Besides, WGRSPM outperforms the NBNN [17] method which shows the effectiveness of discriminatively trained classifiers. As the number of training images increases, the performance improvement of WGRSPM over other methods decreases. This is because with the increase of training image numbers, we can learn more discriminative classifiers. This makes the improvement of weighted local feature encoding with governing region scheme less significant. However, we are still able to improve the final image classification performance over other methods. The proposed method is also able to outperform the fisher kernel based method [43]. The fisher kernel extends the traditional BoW model by going beyond count statistics. Besides, we can see from Tables 2 and 3 that the performance improvement of WGRSPM decreases from Caltech 101 dataset to Caltech 256 dataset. This is because the Caltech 256 dataset is more difficult to classify than the Caltech 101 dataset as images in the Caltech 256 dataset have more class numbers and are not rotated or cropped with alignment. Besides, the intra and inter class variations of the Caltech 256 dataset are larger that the Caltech 101 dataset.

4.5. UIUC-Sport dataset

The UIUC-Sport dataset contains eight categories (*badminton, bocce, croquet, polo, rock climbing, rowing, sailing* and *snow boarding*) of 1792 images. Each category has 137–250 images. 5×10^4 local features are randomly chosen for codebook generation, as did on the Scene 15 dataset. We randomly select 70 images per

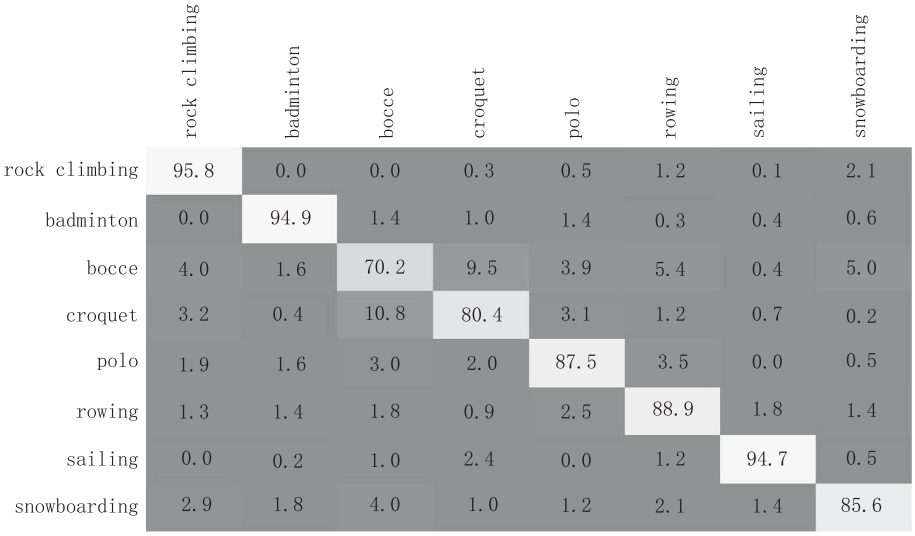

Fig. 7. Confusion matrix (UIUC-Sport dataset).

class as the training set and use the rest for testing. Fig. 6 shows some example images of the UIUC-Sport dataset.

We give the performance comparison of WGRSPM with [2,5,6,40] on the UIUC-Sport dataset in Table 4. We choose these methods for comparison because they are closely related with the proposed method or are the-state-of-the-art methods. The image classification results again demonstrate the effectiveness of the proposed WGRSPM method. Combining the discriminative information of local features and the governing regions of visual words help to encode local features more efficiently than using visual words only [2,6] or using histogram intersection similarity [5,40]. We also give the confusion matrix in Fig. 7 for per class performance evaluation. We can see from Fig. 7 that the proposed WGRSPM method is less efficient to classify *bocce* and *croquet* classes than other classes. This is because *bocce* and *croquet* images are more difficult and are with large intra class variations compared with other image classes. The *bocce* and *croquet* images are more likely to be confused. This is because both the two classes are with people on the grass land while the bocce and croquet are relatively small to be well encoded. This also happens with the LScSPM method which achieved 64.7% and 77.8% classification accuracy on the *bocce* and *croquet* classes respectively.

5. Conclusion

This paper proposes a novel image representation method by going one step beyond visual word ambiguity and consider the governing regions of visual words generated by weighted local features. For each visual application task, local features are weighted to take their discriminative power into consideration. Besides, each local feature is encoded not only by considering its similarities with visual words, but also by visual word's governing region. Sparsity constraint is also used to encode local features efficiently. A weighted feature sign search algorithm is proposed to solve this governing region based weighted sparse coding problem. We also give the theoretical justification of the convergence of the proposed algorithm. Finally, we conduct experiments on several public datasets to demonstrate the effectiveness of the proposed method for image classification. Note that the proposed image representation method can also be applied to other visual applications, such as object detection and image retrieval.

Our future work will focus on how to speed up the computation will also be studied. Besides, the adaption of the proposed image representation method for other visual applications will also be explored.

Acknowledgments

This work is supported by National Basic Research Program of China (973 Program): 2012CB316400, National Natural Science Foundation of China: 61303154, 61202234, 61305018, 61303114, 61202325 and 61332016, the Open Project Program of the National Laboratory of Pattern Recognition (NLPR), the President Fund of University of Chinese Academy of Sciences (UCAS). Beijing Municipal Natural Science Foundation of China No. 4132010. Specialized Research Fund for the Doctoral Program of Higher Education (No. 20130141120024).

References

[1] J. Sivic, A. Zisserman, Video google: a text retrieval approach to object matching in videos, in: Proc. of 9th IEEE Int. Conf. on Computer Vision, Nice, France, 2003, pp. 1470–1477.
[2] J. Yang, K. Yu, Y. Gong, T. Huang, Linear spatial pyramid matching using sparse coding for image classification, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Miami, USA, 2009, pp. 1794–1801.
[3] J.C. Gemert, C.J. Veenman, A. Smeulders, J. Geusebroek, Visual word ambiguity, IEEE Trans. Pattern Anal. Mach. Intell. 32 (2010) 1271–1283.
[4] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, San Francisco, USA, 2010, pp. 3360–3367.
[5] S. Gao, I. Tsang, L. Chia, P. Zhao, Local features are not lonely-Laplacian sparse coding for image classification, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, San Francisco, USA, 2010, pp. 3555–3561.
[6] C. Zhang, J. Liu, Q. Tian, C. Xu, H. Lu, S. Ma, Image classification by non-negative sparse coding, low-rank and sparse decomposition, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, CO, USA, 2011, pp. 1673–1680.
[7] L. Liu, L. Wang, X. Liu, In defense of soft-assignment coding, in: Proc. of IEEE Int. Conf. of Computer Vision, DC, USA, 2011, pp. 2486–2493.
[8] F. Perronnin, C. Dance, Fisher kernels on visual vocabularies for image categorization, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Minnesota, USA, 2007, pp. 1–8.
[9] F. Jurie, B. Triggs, Creating efficient codebooks for visual recognition, in: Proc. of IEEE Int. Conf. on Computer Vision, Beijing, China, 2005, pp. 604–610.
[10] F. Moosmann, B. Triggs, F. Jurie, Randomized clustering forests for building fast and discriminative visual vocabularies, in: Proc. of the Neural Information Processing Systems, MIT Press, British Columbia, Canada, 2006, pp. 985–992.
[11] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, New York, USA, 2006, pp. 2169–2178.
[12] K. Grauman, T. Darrell, Pyramid match kernels: discriminative classification with sets of image features, in: Proc. of IEEE Int. Conf. on Computer Vision, Beijing, China, 2005, pp. 725–760.
[13] Q. Chen, Z. Song, Y. Hua, Z. Huang, S. Yan, Hierarchical matching with side information for image classification, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Rhode Island, USA, 2012, pp. 3426–3433.
[14] B. Yao, A. Khosla, Li Fei-Fei, Combining randomization and discrimination for fine-grained image categorization, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Rhode Island, USA, 2012, pp. 1577–1584.
[15] C. Zhang, J. Liu, Q. Tian, Y. Han, H. Lu, S. Ma, A boosting, sparsity-constrained bilinear model for object recognition, IEEE Multimedia 19 (2012) 58–68.
[16] N. Zhang, R. Farrell, T. Darrell, Pose pooling kernels for sub-category recognition, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Rhode Island, USA, 2012, pp. 3665–3672.
[17] O. Boiman, E. Shechtman, M. Irani, In defense of nearest-neighbor based image classification, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Alaska, USA, 2008, pp. 1–8.
[18] R. Timofte, L. Gool, Iterative nearest neighbors for classification and dimensionality reduction, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Rhode Island, USA, 2012, pp. 2456–2463.
[19] S. McCann, D. Lowe, Local naive Bayes nearest neighbor for image classification, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Rhode Island, 2012, pp. 3650–3656.
[20] B. Olshausen, D. Field, Emergence of simple-cell receptive field properties by learning a sparse code for natural images, Nature 381 (1996) 607–609.
[21] H. Lee, A. Battle, R. Raina, A. Ng, Efficient sparse coding algorithms, in: Advances in Neural Information Processing Systems, MIT Press, Vancouver, Canada, 2006, pp. 801–808.
[22] J. Wright, A. Yang, A. Ganesh, S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2009) 210–227.
[23] M. Yang, L. Zhang, X. Feng, D. Zhang, Fisher discrimination dictionary learning for sparse representation, in: Proc. of IEEE Int. Conf. on Computer Vision, Barcelona, Spain, 2011, pp. 543–550.
[24] E. Elhamifar, R. Vidal, Robust classification using structured sparse representation, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, CO, USA, 2011, pp. 1873–1879.
[25] B. Fernando, E. Fromont, D. Muselet, M. Sebban, Discriminative feature fusion for image classification, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Rhode Island, USA, 2012, pp. 3434–3441.
[26] R. Cinbis, J. Verbeek, C. Schmid, Image categorization using fisher kernels of non-iid image models, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Rhode Island, USA, 2012, pp. 2184–2191.
[27] G. Sharma, F. Jurie, C. Schmid, Discriminative spatial saliency for image classification, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Rhode Island, USA, 2012, pp. 3506–3513.
[28] F. Moosmann, D. Larlus, F. Jurie, Learning saliency maps for object categorization, in: ECCV Workshop on the Representation and Use of Prior Knowledge in Vision, Springer, 2006.
[29] Y. Su, F. Jurie, Visual word disambiguation by semantic context, in: IEEE Int. Conf. on ICCV, 2011.
[30] Y. Su, F. Jurie, Improving image classification using semantic attributes, Int. J. Comput. Vis. 100 (1) (2012) 59–77.
[31] L. Fei-Fei, R. Fergus, P. Perona, Learning generative visual models from few training examples: an incremental Bayesian approach test on 101 object categories, Comput. Vis. Image Understand. 106 (1) (2007) 59–70.
[32] G. Griffin, A. Holub, P. Perona, Caltech-256 Object Category Dataset, Caltech, Tech. Rep. UCB/CSD-04-1366, 2007.
[33] L.-J. Li, L. Fei-Fei, What, where and who? Classifying events by scene and object recognition, in: Proc. of IEEE Int. Conf. on Computer Vision, Rio, Brazil, 2007, pp. 1–8.

[34] D. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2004) 91–110.
[35] L. Fei-Fei, P. Perona, A Bayesian hierarchical model for learning natural scene categories, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, San Diego, USA, 2005, pp. 524–531.
[36] A. Oliva, A. Torralba, Modeling the shape of the scene: a holistic representation of the spatial envelope, Int. J. Comput. Vis. 42 (2001) 145–175.
[37] K. Chatfield, V. Lempitsky, A. Vedaldi, A. Zisserman, The devil is in the details: an evaluation of recent feature encoding methods, in: British Machine Vision Conference, Dundee, British, 2011, pp. 1–12.
[38] N. Rasiwasia, N. Vasconcelos, Scene classification with low-dimensional semantic spaces and weak supervision, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, Alaska, USA, 2008, pp. 1–6.
[39] L. Li, H. Su, E. Xing, Li. Fei-Fei, ObjectBank: a high-level image representation for scene classification and semantic feature sparsification, in: Proc. of the Neural Information Processing Systems, MIT Press, Vancouver, Canada, 2010, pp. 1378–1386.
[40] J. Wu, J. Rehg, Beyond the euclidean distance: creating effective visual codebooks using the histogram intersection kernel, in: Proc. of IEEE Int. Conf. on Computer Vision, GA, USA, 2009, pp. 630–637.
[41] X. He, S. Yan, Y. Hu, P. Niyogi, H. Zhang, Face recognition using Laplacianfaces, IEEE Trans. Pattern Anal. Mach. Intell. 27 (2005) 328–340.
[42] H. Zhang, A. Berg, M. Maire, J. Malik, Svm-knn: discriminative nearest neighbor classification for visual category recognition, in: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition, NY, USA, 2006, pp. 2126–2136.
[43] F. Perronnin, J. Sanchez, T. Mensink, Improving the fisher kernel for large-scale image classification, in: Proc. of European Conference on Computer Vision, Springer, 2010, pp. 143–156.