Information Sciences 310 (2015) 118–129

Contents lists available at ScienceDirect

Information Sciences

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

Image classification using boosted local features with random orientation and location selection

Chunjie Zhang [a], Jian Cheng [b,*], Yifan Zhang [b], Jing Liu [b], Chao Liang [c], Junbiao Pang [d], Qingming Huang [a,e], Qi Tian [f]

[a] School of Computer and Control Engineering, University of Chinese Academy of Sciences and Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, 100049 Beijing, China
[b] National Laboratory of Pattern Recognition, Institute of Automation Chinese Academy of Sciences, P.O. Box 2728, Beijing, China
[c] National Engineering Research Center for Multimedia Software, School of Computer, Wuhan University, 430072 Wuhan, China
[d] Beijing Key Laboratory of Multimedia and Intelligent Software Technology, College of Metropolitan Transportation, Beijing University of Technology, 100124, China
[e] Key Lab of Intell. Info. Process, Institute of Computing Technology, Chinese Academy of Sciences, 100190 Beijing, China
[f] Department of Computer Sciences, University of Texas at San Antonio, TX 78249, USA

ARTICLE INFO

Article history:
Received 15 August 2014
Received in revised form 4 January 2015
Accepted 9 March 2015
Available online 17 March 2015

Keywords:
Sparse coding
Image classification
Random orientation
Boosting
Local feature selection

ABSTRACT

The combination of local features with sparse technique has improved image classification performance dramatically in recent years. Although very effective, this strategy still has two shortcomings. First, local features are often extracted in a pre-defined way (e.g. SIFT with dense sampling) without considering the classification task. Second, the codebook is generated by sparse coding or its variants by minimizing the reconstruction error which has no direct relationships with the classification process. To alleviate the two problems, we propose a novel boosted local features method with random orientation and location selection. We first extract local features with random orientation and location using a weighting strategy. This randomization process makes us to extract more types of information for image representation than pre-defined methods. These extracted local features are then encoded by sparse representation. Instead of generating the codebook in a single process, we construct a series of codebooks and the corresponding encoding parameters of local features using a boosting strategy. The weights of local features are determined by the classification performances of learned classifiers. In this way, we are able to combine the local feature extraction and encoding with classifier training into a unified framework and gradually improve the image classification performance. Experiments on several public image datasets prove the effectiveness and efficiency of the proposed method.

1. Introduction

Image classification is a classic problem in computer vision. It tries to classify one image to a pre-defined class by analyzing the image's content. Recently, the use of sparse coding for image classification becomes popular. Sparse coding [6] tries to minimize the reconstruction error of one given feature by selecting a relatively small subset of basis sets. Since its introduction, the sparse coding technique and its variants have attracted more and more researchers' attention and have been proved effective for many vision applications [12,31,44,45].

* Corresponding author.
E-mail addresses: cjzhang@jdl.ac.cn (C. Zhang), jcheng@nlpr.ia.ac.cn (J. Cheng), yfzhang@nlpr.ia.ac.cn (Y. Zhang), jliu@nlpr.ia.ac.cn (J. Liu), cliang@whu.edu.cn (C. Liang), junbiao_pang@bjut.edu.cn (J. Pang), qmhuang@jdl.ac.cn (Q. Huang), qitian@cs.utsa.edu (Q. Tian).

http://dx.doi.org/10.1016/j.ins.2015.03.011
0020-0255/

Basically, sparse coding based image classification method can be divided into three steps. First, local features are extracted either by detection or dense sampling. Second, the codebook is generated (using sparse coding or its various variants) and local features are encoded accordingly. Finally, images are represented using the encoded parameters and SVM classifiers are trained to predict the categories of images. Although this strategy has been proven very effective for image classification. It still has two shortcomings. On one hand, the local feature extraction process and the classification task are relatively independent. On the other hand, the optimal parameters for minimizing reconstruction error cannot be able to help training discriminative classifiers for prediction. For example, the classifying of tiger and cat is different from separating cat and dog. Although researchers have tried [2,9,46] to alleviate the two problems, the results are far from satisfactory with heavy computational cost. If we can combine the local feature encoding and classifier training into a unified framework, we are able to model images better for classification.

To alleviate the two problems mentioned above, we propose to classify images using boosted local features with random orientation and location selection. We first extract local features by randomly choosing the locations and orientations with re-weighting to extract more types of information than pre-defined local feature extraction strategies (e.g. dense sampling of SIFT features). For each random extraction strategy, we generate the corresponding codebook with re-weighting and encode the features accordingly. Then we train SVM classifiers to make prediction of image classes. The outputs of the trained classifiers are used to re-weight images. This process is iterated in a boosting way to combine the discriminative power of a series of classifiers for classification. In this way, we can unify the extraction of local features, the generation of codebook and the training of classifiers into a unified framework.

Our main contribution are as follows.

- We propose a random local feature extraction strategy with orientation and location selection. This helps us to extract more types of information for classification. Besides, a re-weighting scheme is also imposed to extract more information from the 'hard' images which may help the classification task.
- We iteratively generate codebooks using the randomly extracted local features with re-weighting. The weights are determined by the predictions of learned classifiers. The misclassified images gains more weights compared with the correctly classified images, making the proposed method concentrates on the 'hard' images for each round.
- We unify the local feature extraction, codebook generation and classifier training into a unified framework iteratively by using the boosting strategy. The discrepancy between the predicted classes and groundtruth is used to re-weight the training images which are then used for local feature extraction. In this way, we can gradually improve the image classification performances by combining the sparse coding based image representation with the boosting strategy.

The rest of this paper is organized as follows. Section 2 introduces some related work. Section 3 gives the details of the proposed boosted local feature with random orientation and location selection method for image classification. Experimental comparisons are given in Section 4 and finally we conclude in Section 5.

2. Related work

The bag-of-visual-words (BoW) model [38] is widely used for image classification in recent years due to its simplicity and efficiency. k-means is usually used for codebook generation and nearest neighbor assignment is leveraged to quantize local features. This hard assignment strategy causes information loss which hinders final classification performance. Gemert et al. [14] tried to softly encode local features while Yang et al. [45] used the sparse coding technique. Motivated by this, a lot of works have been done [13,42,48–51]. Wang et al. [42] added locality constraints during the sparse coding process to speed up the computation and improve the performance. Gao et al. [13] explored the relationship of local feature similarities and encoded parameter similarities with Laplacian sparse coding. Zhang et al. [48] used non-negative sparse coding instead of sparse coding to ensure consistency with the max pooling strategy.

Most image classification methods used histogram based features for local region description, such as SIFT [29] and HoG [5]. To speed up computation of local features, many works have been done [1,18]. Speeded up robust features (SURF) was proposed by Bay et al. [1] which can be computed 3–7 folds faster than SIFT. The hashing technique [18] was also proposed by Indyk and Motwani. These local features are then encoded either by sparse coding or its variants to get the image representation for classification. This is often achieved by minimizing the reconstruction error with sparsity constraints. Although very effective, most of the above mentioned methods treated local feature generation and classifier training separately for image classification. The objectives of codebook generation and local feature encoding are minimizing the reconstruction errors while the objective of classifier training is to minimize the classification error. To solve this problem, Yang et al. [46] tried to unify the codebook generation with classifier training for object recognition by modeling on the SIFT features directly while the use of nearest neighbor information for direct image classification was also proposed by Boiman et al. [2]. Lowe [30] extended it by using nearest neighbor information to speed up the computation. However, the computational cost of [2,30,46] are very high compared with the BoW model for classification.

Instead of using pre-defined local feature extraction strategy, the use of features learned from the images also becomes popular [8,16,20,37] in recent years. Deep belief network (DBN) [16] and convolutional neural network (CNN) [20] tried to learn multiple layers of nonlinear features from images. Shao et al. [37] used multi-objective genetic programming

technique for automatic feature learning in order to classify images. Fan et al. [8] tried to generate binary features by receptive field selection for object matching. These learning based methods have been proven very effective for handling with large image datasets. However, how to tune the parameters of these methods to get reliable performances is still an open problem that needs to be solved. The use of matrix factorization technique was also proven very effective by researchers [23–27]. Besides, annotation [17,39–41,43] was also adopted to alleviate the deficiency between visual features and semantic meanings.

Using randomness for image classification is also very popular. Zhang et al. [52] experimentally found that the randomly selected codebook performs no worse than the codebook generated by k-means clustering. Instead of generating one codebook, Moosmann et al. [32] used random clustering forests to construct a series of trees for local feature coding. These encoding parameters are then concatenated to represent images. A randomization based multiclass boosting method is also proposed by Paisitkriangkrai et al. [34]. Because their is no objective function to optimize, the randomness based methods can be computed very efficiently.

Boosting was proposed by Schapire [36] to turn a set of weak learners into a strong learner as long as the weak learners are not too 'weak' [53]. Boosting was widely used and extended since its introduction, e.g. AdaBoost [11], RankBoost [10] and random forest [3]. The usages of boosting technique for image classification [28,34,50] are also very popular both for image representation and classifier training with good performances. By iteratively adding weak learners to improve the performance, we can finally get reliable image classification rates.

3. Image classification using boosted local features with random orientation and location selection

In this section, we give the details of the proposed image classification using boosted local features with random orientation and location selection algorithm. First, local features are extracted with random orientation and location. These local features are then used to learn the codebook by sparse coding with re-weighting with the weights of each local feature is determined by the discrepancy of the predicted label and the groundtruth of images from which it is extracted. We then use the learned codebook for local feature encoding and extract image representation by spatial pyramid max pooling. Linear SVM classifiers are then used to predict image categories and we use the predicted values for local feature re-weighting. This process is iterated in a boosting way to get reliable image classification performance. Fig. 1 shows the outline of the proposed method.

3.1. Local feature extraction with random orientation and location

Given an image, extracting proper local features is very important for image classification. Histogram based features are often used. We follow this paradigm and extract local feature with random orientation and location. Images are first densely divided into small spatial regions with multi-scale and overlap. We then compute the gradients of these local regions and sub-divide it into spatial cells. We use the 4×4 cells with 8 gradient orientations in this paper, as [5] did. Note that other cell partitions strategies can also be used. The 1-D histogram of gradients over the pixels of one cell is extracted to represent this cell. The gradient information within each local region is called rare local feature in this paper.

Instead of using these extracted features for classification directly. We impose a random selection process to jointly consider the classification task. Let $w_n,\ n = 1, \ldots, N$ be the weight of the nth images. N is the number of training images. The weights of images are set to be the same before the first iteration and are updated after each classification iteration. We give the rare local features extracted from one particular image the same weight as the image's. The rare local features are weighted accordingly. We then summarize these rare local features to find the number of dominant gradients. The dominant gradients are defined as the first K largest values of the summed rare local features. We randomly select K gradient orientations from the 4×4 cells of each local region as our local feature representation during this iteration. This selected gradient has both orientation and the corresponding location information. We repeat this random selection process for M times to

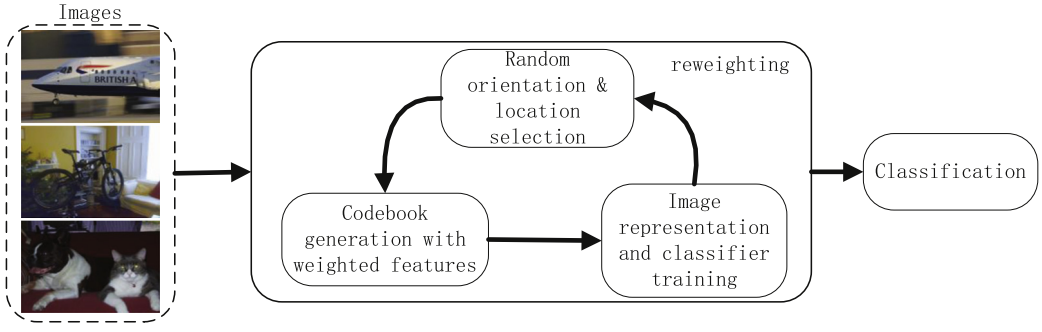

Fig. 1. Outline of the proposed image classification using boosted local features with random orientation and location selection method.

extract more information. In this way, we can extract local feature for image representation which can jointly consider the classification task. In other words, the local feature extraction process is task dependent. This is different from most image classification methods which extract local features independent of the classification task in a single round. Besides, this local feature extraction process is repeated in an iterative way with the weighs updated by the learned classifiers. This means the random selection process can extract different types of features for classification during different iterations.

3.2. Codebook generation with weighted local features

After the local features are extracted, we can use them to learn the codebooks for feature encoding. Since we extract different types of local features for each iteration and random selection strategy, we need to generate the corresponding codebooks. Formally, let $x_{m,i}^{p,n} \in \mathbb{R}^{K\times 1}$ denote the ith local feature extracted during the mth random selection from the nth training image of the pth iteration, $w_{m,i}^{p,n}$ is the corresponding weight. Where $m = 1,\ldots,M, n = 1,\ldots,N, p = 1,\ldots,P$, P is the number of iterations. The sparse coding technique [35] tries to reconstruct the local features by jointly minimizing the reconstruction error and sparseness as:

$$min_{U_m^p, v_{m,i}^{p,n}} \left\| x_{m,i}^{p,n} - U_m^p \times v_{m,i}^{p,n} \right\|_2^2 + \lambda_m^p \left\| v_{m,i}^{p,n} \right\|_1 \tag{1}$$

where $U_m^p \in \mathbb{R}^{K\times Q}$ is the corresponding codebook to be learned for the mth random selection of pth iteration, $v_{m,i}^{p,n} \in \mathbb{R}^{Q\times 1}$ is the sparse coded parameters of $x_{m,i}^{p,n}$ with λ_m^p is the sparsity parameter. However, since our extracted local features have weight information, it should also be included for local feature encoding. Hence, we added the weight constraints into the sparse coding technique and use weighted sparse coding instead:

$$min_{U_m^p, v_{m,i}^{p,n}} w_{m,i}^{p,n} \left\| x_{m,i}^{p,n} - U_m^p \times v_{m,i}^{p,n} \right\|_2^2 + \lambda_m^p \left\| v_{m,i}^{p,n} \right\|_1 \tag{2}$$

In this way, we can jointly consider the codebook generation and classifier training. Recently, Gao et al. [12] found that imposing encoding parameter consistency into the sparse coding process can further improve the performance. We follow this strategy and propose the weighted Laplacian sparse coding as:

$$min_{U_m^p, v_{m,i}^{p,n}, \ldots, v_{m,l}^{p,n}} \sum_i w_{m,i}^{p,n} \left\| x_{m,i}^{p,n} - U_m^p \times v_{m,i}^{p,n} \right\|_2^2 + \lambda_m^p \left\| v_{m,i}^{p,n} \right\|_1 + \beta_m^p \sum_{i,j} \left\| v_{m,i}^{p,n} - v_{m,j}^{p,n} \right\|^2 S_{i,j} \tag{3}$$

where β_m^p is the similarity smooth parameter. $S_{i,j}$ measures the similarity of weighted local features $x_{m,i}^{p,n}$ and $x_{m,i}^{p,n}$ as:

$$S_{i,j} = w_{m,i}^{p,n} \times w_{m,j}^{p,n} \times Similarity\left(x_{m,i}^{p,n}, x_{m,j}^{p,n}\right) \tag{4}$$

With the similarities between $x_{m,i}^{p,n}$ and $x_{m,i}^{p,n}$ measured by histogram intersection kernel as:

$$Similarity\left(x_{m,i}^{p,n}, x_{m,j}^{p,n}\right) = \sum_{k=1}^{K} min\left(x_{m,ik}^{p,n}, x_{m,jk}^{p,n}\right) \tag{5}$$

Problem (3) can be alternatively solved by fixing the codebook and encoding parameters. When the encoding parameters are fixed, Problem (3) can be rewritten as:

$$min_{U_m^p} \sum_i w_{m,i}^{p,n} \left\| x_{m,i}^{p,n} - U_m^p \times v_{m,i}^{p,n} \right\|_2^2 \tag{6}$$

Which can be optimized using conjugate gradient decent method [21]. We optimize for the M codebooks during the pth iteration independently. When the codebook is fixed, Problem (3) can be rewritten as:

$$min_{v_{m,i}^{p,n}, \ldots, v_{m,l}^{p,n}} \sum_i w_{m,i}^{p,n} \left\| x_{m,i}^{p,n} - U_m^p \times v_{m,i}^{p,n} \right\|_2^2 + \lambda_m^p \left\| v_{m,i}^{p,n} \right\|_1 + \beta_m^p \sum_{i,j} \left\| v_{m,i}^{p,n} - v_{m,j}^{p,n} \right\|^2 S_{i,j} \tag{7}$$

Jointly solving for the optimal encoding parameters for all local features is impossible. Hence we optimize them one by one. This can be achieved by optimizing each local feature $x_{m,i}^{p,n}$ iteratively by fixing the encoding parameters of other local features fixed as:

$$min_{v_{m,i}^{p,n}} w_{m,i}^{p,n} \left\| x_{m,i}^{p,n} - U_m^p \times v_{m,i}^{p,n} \right\|_2^2 + \lambda_m^p \left\| v_{m,i}^{p,n} \right\|_1 + \beta_m^p \sum_{i,j} \left\| v_{m,i}^{p,n} - v_{m,j}^{p,n} \right\|^2 S_{i,j} \tag{8}$$

Problem (8) can be solved by using the feature-sign-search algorithm [12,21,45].

After the codebooks are learned, we can use them for encoding. The local features used for codebook generation is kept as template features for new local feature encoding. The new features are encoded one by one based on the assumption that the new feature does not affect the Laplacian graph [12]. This corresponds to solving for the optimization Problem (8) while keeping the codebook and the encoding parameters of other local features. These encoded parameters are then used as the image representation during the pth iteration.

For sparse coded local features, max pooling strategy with spatial pyramid matching [19] is often used to extract image representation. It selects the maximum response of all the parameters within one particular region for representation. This strategy is inspired by the biological model [33] with its performances for image classification proved by many researchers [12,13,42,45,48]. Formally, let $h_m^{p,n}$ represent the max pooled features over one region with D local features of the nth image and pth iteration, the jth dimension of $h_m^{p,n}$ can be calculated as:

$$h_{m,j}^{p,n} = max\left\{\left|v_{m,1j}^{p,n}\right|, \left|v_{m,2j}^{p,n}\right|, \ldots, \left|v_{m,D}^{p,n}\right|\right\} \tag{9}$$

The final image representation $h^{p,n}$ is then obtained by concatenating the max pooled features $h_m^{p,n}$ of each random sampling strategy as:

$$h^{p,n} = (h_1^{p,n}; h_2^{p,n}; \ldots; h^{p,n} p_M) \tag{10}$$

3.3. Boosted local features for image classification

After the final image representation for the pth iteration is obtained, we can use it to train classifiers for classification prediction. Given the training images $\{h^{p,n}, y_n\},\ n = 1, \ldots, N, y_n \in \{1, \ldots, L\}$ with their corresponding weights $w^{p,n}$, L is the number of image classes. For the pth iteration, we want to train a classifier such that the predicted value $\bar{y}_n = \sum_p \bar{y}_n^p$ matches with the groundtruth y_n. In this paper, we use the multi-class linear SVM classifier for each iteration by training L linear classifiers:

$$\bar{y}_n^p = max_c \gamma_c^T h^{p,n} \tag{11}$$

by optimizing the following problem:

$$min_{\gamma_c} \|\gamma_c\|^2 + C\sum_{n=1}^{N} \ell(\gamma_c h^{p,n}, y_n) \tag{12}$$

with the quadratic hinge loss function as:

$$\ell(\gamma_c h^{p,n}, y_n) = \left(max(0, \gamma_c^T h^{p,n} \times y_n - 1)\right)^2 \tag{13}$$

The linear classifier is shown to performance comparable with other kernel based methods when combined with sparse coding and max pooling strategy. Besides, the parameters needed to be tuned are less for linear SVM classifier than non-linear classifiers.

We combine the outputs of these non-linear SVM classifiers for final image classification in a boosting way. At the pth iteration, the predicted value is $\bar{y}_n = \sum_p \bar{y}_n^p$. We use the exponential loss as the loss function which has the form as:

$$exp(-\bar{y}_n \times y_n) = exp\left(-\sum_{i}^{p-1} \bar{y}_n^i \times y_n\right) \cdot exp(-\bar{y}_n^p \times y_n) \tag{14}$$

Let $w_{p-1,n} = exp\left(-\sum_i^{p-1} \bar{y}_n^i \times y_n\right)$, Eq. 14 can be rewritten as:

$$exp(-\bar{y}_n \times y_n) = w_{p-1,n} \cdot exp(-\bar{y}_n^p \times y_n) \tag{15}$$

And the overall loss function is defined as the summed exponential loss of all training images:

$$Loss = \sum_n exp(-y_n \times \bar{y}_n) \tag{16}$$

In this way, we can use the predicted values of the former iterations to weight images. These weighted images are then used to extract local features and generate the corresponding codebooks. Instead of sequently extract local features, construction codebook and make classification of images, we conduct it in an iterative way to combine them into a unified process. This can not only makes the objectives of each process more consistent but also can improve the final classification by concentrating on the 'hard' to classify images. Algorithm 1 gives the detailed procedure of the proposed image classification using boosted local features with orientation and location selection method.

Algorithm 1. The proposed image classification using boosted local features with orientation and location selection algorithm.

Input: The training images and labels with initial weights, test images, boosting iteration number *maxiter*;
Output: The predicted classes of test images;
1: **for** $iter = 1, 2, \ldots, maxiter$
2: Extract local features with random orientation and location and weights from images as described in Section 3.1
3: Generate the codebooks and encoding schemes with weighted local features by alternatively optimizing over Problems (6)–(8) as described in Section 3.2;
4: Train multi-class SVM classifiers to predict image categories for this iteration and re-weight training images accordingly.
5: Check if $iter > maxiter$ or the decrease of the summed exponential loss falls below a pre-defined threshold.
If unsatisfied
go to step 1
Else
stop, go to step 6
6: end **for**.
7: **return** The predicted classes of test images;

4. Experiments

To evaluate the performance of the proposed image classification method using boosted local features with random orientation and location selection (B-ROL), we conduct image classification experiments on several public image datasets, the Scene-15 dataset [19], the Caltech-101 dataset [22], the Caltech-256 dataset [15] and the PASCAL VOC 2007 dataset [7].

4.1. Experimental settings

We select image regions of multi-scale with the smallest image region is set to 16×16 pixels in this paper. Each image region is sub-divided into 4×4 cells and eight gradient orientations are calculated for each cell as the rare local region description for random selection. Since the images of PASCAL VOC 2007 dataset are more hard to classify than the other three datasets, we first map the images into different color spaces (e.g. RGB, HSV), as Sande et al. [35] did in order to extract more information. We randomly select about 50,000 local features for each codebook generation and feature encoding. The codebook size is set to 1024, as Yang et al. [45] did. Max pooling with three scales of spatial pyramid (1×1, 2×2, 4×4) is used to combine the spatial information for image representation. During each iteration, the random dimension number K as well as the random selection times M are two important parameters that control the performances. Using more dimensions for representation and selecting more times will help to improve the classification accuracy. However, this also costs more computational power both for codebook generation and feature encoding. Besides, since different image datasets have different difficulties to classify, K and M are also dataset dependent. A larger sparsity parameter λ_m^p results in more sparse parameters than a smaller one while a larger smooth parameter β_m^p ensures more similar encoding parameters for similar features. Following the parameter settings of [2,12], we set the sparsity parameter λ to 0.3–0.4 and the smoother parameter β to 0.1–0.3 respectively.

We randomly select the training images per class and use the rest images for performance evaluation. This process is repeated for ten times for each image dataset to get reliable results. Multi-class classification is done via the one-versus-all rule: a SVM classifier is learned to separate each class from the rest and a test image is assigned the label of the classifier with the highest response. As to the boosting process, the maximum boosting iteration is set to 50 in this paper. The average of per-class classification rates is used to quantitatively measure the performance for all the datasets expect the PASCAL VOC 2007 dataset. As to the PASCAL VOC 2007 dataset, the mean average precision (mAP) is used for performance evaluation. The randomly selected local feature dimension K and the random selection times M are two important parameters which control the discriminative power of image representation during each iteration. We experimentally found that the performances increases with large feature dimension and more random selection times. By selecting local features many times with more dimensions, we can extract more information for each iteration. However, using larger K and M also costs more computational power. Hence, we empirically set M to 5, 10, 10, 30 for the Scene-15, the Caltech-101, the Caltech-256 and the PASCAL VOC 2007 datasets respectively.

4.2. Scene-15 dataset

The first image dataset we consider is the Scene-15 dataset. The fifteen scene dataset has 4485 images of fifteen classes, which vary from natural scenes to man-made environments. We randomly select 100 images per class for training in order to be consistent with other methods [13,14,19,45].

Table 1 gives the performance comparison of the proposed method with other methods [13,14,19,45]. We also give the performance of using boosted SIFT features for classification. This is achieved by directly extracting the SIFT features instead of random local feature extraction. We can see from Table 1 that the proposed method is able to outperform other sparse coding based methods. Specially, by imposing the boosting process, we can improve the classification over LScSPM by more than 3%. Besides, by randomly select local features, we can extract more types of information with the re-weighting strategy, hence improves the performance by about 1.5% over B-ROL (without random selection). Moreover, the combination of codebook generation with classifier training by re-weighting also helps to encode local features more suitable for classification. Finally, the jointly consideration of feature extraction, codebook generation and classifier training helps to improve the classification performance over sparse coding by about 13%. This proves the effectiveness of the proposed method. We show some example images that our method classified correctly and misclassified on the Scene-15 dataset in Fig. 2.

4.3. Caltech-101 dataset

The Caltech-101 dataset has 9144 images of 101 classes with the number of images per classes varies from 31 to 800. We follow the experimental setup as [42] and randomly select 15, 30 training images respectively per class and use the rest images for evaluation.

Table 1
Performance comparison of the proposed method with other methods on the Fifteen Scene dataset. Numerical values in the table stand for mean and standard derivation. The bold values are used to indicate the best classification performances.

Algorithms	Classification rate
KSPM [45]	76.73 ± 0.65
KC [14]	76.67 ± 0.39
ScSPM [45]	80.28 ± 0.93
KSPM [13]	81.40 ± 0.50
LScSPM [12]	89.75 ± 0.50
B-ROL (without random selection)	**91.85 ± 0.53**
B-ROL	**93.38 ± 0.55**

Fig. 2. Example images correctly classified and misclassified on the Scene-15 dataset. For each class, the left/right images are correctly classified/misclassified respectively.

Table 2
Performance comparison on the Caltech-101 dataset. The bold values are used to indicate the best classification performances.

Methods	15 training	30 training
KSPM [19]	56.40	64.40 ± 0.80
KC [14]	–	64.14 ± 1.18
NBNN [2]	65.00 ± 1.14	70.40
ScSPM [45]	67.00 ± 0.45	73.20 ± 0.54
LLC [42]	65.43	73.44
KMTJSRC [47]	65.00 ± 0.70	–
B-ROL	**72.25 ± 0.70**	**76.92 ± 0.59**

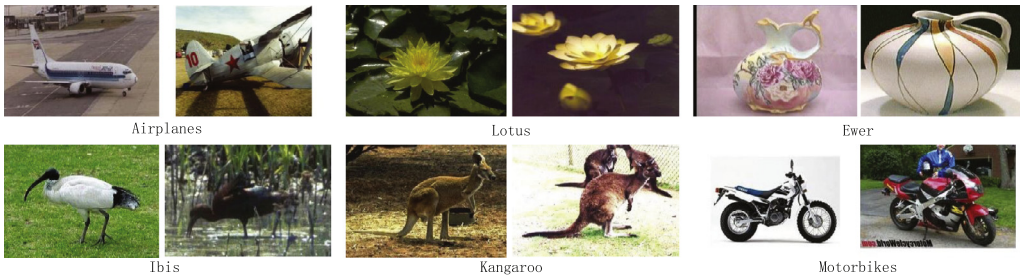

Fig. 3. Example images correctly classified and misclassified on the Caltech-101 dataset. For each class, the left/right images are correctly classified/misclassified respectively.

Table 3
Performance comparisons on the Caltech-256 dataset. The bold values are used to indicate the best classification performances.

Methods	15 training	30 training	45 training
KC [14]	–	27.17 ± 0.46	–
KSPM [15]	–	34.10	–
NBNN(1 Desc)[2]	30.45	38.18	–
KSPM [45]	23.34 ± 0.42	29.51 ± 0.52	–
ScSPM [45]	27.73 ± 0.51	34.02 ± 0.35	37.46 ± 0.55
LLC [42]	34.36	41.19	45.31
LScSPM[13]	30.00 ± 0.14	35.74 ± 0.10	38.54 ± 0.36
B-ROL	**37.55 ± 0.29**	**42.12 ± 0.25**	**45.91 ± 0.27**

We give the classification performances comparison with [2,14,19,42,45,47] in Table 2. Compared with the ScSPM [45] which used sparse coding along with max pooling for image classification, the proposed method improves the performance by about 5.2/3.7% with 15/30 training images per class respectively. Besides, the proposed method also outperforms sparse reconstruction for direct classification method KMTJSRC [47] by about 7.2%. Moreover, our method also improves over NBNN [2] which uses local features directly for classification without training classifiers. By iteratively training classifiers in a boosting way, we can extract proper information and improve the performance. The relative improvements of the proposed method over other methods decreases with the increasing of training images. This is because the proposed method can adaptively select local features and make full use of training images by the boosting strategy. Finally, we can see the effectiveness of the proposed method from Table 2. We also give some example images that our method classified correctly and misclassified on the Caltech-101 dataset in Fig. 3.

4.4. Caltech-256 dataset

The Caltech-256 dataset has 29,780 images of 256 classes with larger intra-class variability compared with the Caltech-101 dataset. Each class has at least 80 images. We randomly select 15/30/45 training images per class for performance evaluation and compared with [2,13–15,42,45].

Table 3 gives the performance comparison. We can see from Table 3 that the proposed method outperforms the baseline methods which shows the effectiveness of the proposed method. We can have similar conclusions as on the Caltech-101 dataset. The proposed outperforms the sparse coding with max pooling method [45] and directly SIFT distance [2] based method. B-ROL outperforms LScSPM [13] which also considers the encoding parameter similarities of local features by about 7.5% with 15 training images. This proves the usefulness of selecting local features and jointly considering the codebook generation with classifier training. Besides, we can see that the relative performance improvement increases with the decrease of training images. This is because we can extract more types of information and combine the feature extraction, encoding and classification into a unified process via the boosting strategy for better image representation and classification. We give some example images that our method classified correctly and misclassified on the Caltech-256 dataset in Fig. 4.

4.5. PASCAL VOC 2007 dataset

The PASCAL VOC 2007 dataset has about 10,000 images of 20 classes (person, bird, cat, cow, dog, horse, sheep, airplane, bicycle, boat, bus, car, motorbike, train, bottle, chair, dining table, potted plant, soft and tv/monitor). The images are divided into train/validation/test sets. This dataset is more difficult to classify than the other three datasets.

We compared the performances of the proposed method with LLC [42], the best result's of 2007 competition [7], the re-implementation of fisher kernel [4] and the super vector based method [54] in Table 4. We are able to outperform the

Fig. 4. Example images correctly classified and misclassified on the Caltech-256 dataset. For each class, the left/right images are correctly classified/misclassified respectively.

Table 4
Performance comparison on the PASCAL VOC 2007 dataset.

Object class	LLC [42]	Best'07 [7]	FK [4]	SV [4]	SV [54]	B-ROL
Airplane	74.8	77.5	80.0	74.3	87.1	82.6
Bicycle	65.2	63.6	67.4	63.8	67.4	68.5
Bird	50.7	56.1	51.9	47.0	65.8	59.3
Boat	70.9	71.9	70.9	69.4	72.3	72.8
Bottle	28.7	33.1	30.8	29.1	40.9	38.2
Bus	68.8	60.6	72.2	66.5	78.3	75.7
Car	78.5	78.0	79.9	77.3	69.7	80.9
Cat	61.7	58.8	61.4	60.2	69.7	62.8
Chair	54.3	53.5	56.0	50.2	58.5	57.6
Cow	48.6	42.6	49.6	46.5	50.1	51.6
Table	51.8	54.9	58.4	51.9	55.1	57.4
Dog	44.1	45.8	44.8	44.1	56.3	48.3
Horse	76.6	77.5	78.8	77.9	71.8	79.8
Motorbike	66.9	64.0	70.8	67.1	70.8	69.5
Person	83.5	85.9	85.0	83.1	84.1	85.2
Plant	30.8	36.3	31.7	27.6	31.4	33.7
Sheep	44.6	44.7	51.0	48.5	51.5	52.5
Sofa	53.4	50.9	56.4	51.1	55.1	56.8
Train	78.2	79.2	80.2	75.5	84.7	83.1
Tv	53.5	53.2	57.5	52.3	65.2	61.9
mAP	59.3	59.4	61.7	58.2	64.3	63.9

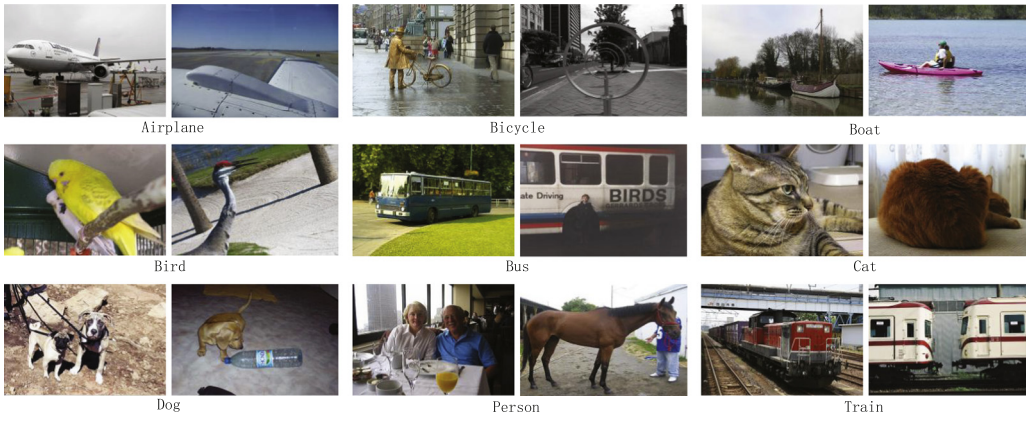

Fig. 5. Example images correctly classified and misclassified on the PASCAL VOC 2007 dataset. For each class, the left/right images are correctly classified/misclassified respectively.

performance of the compared methods which again proves the effectiveness of the proposed method. Besides, the proposed method performs not as good as the super vector based method reported by [54]. We believe this is because of the implementation details, as pointed out by [4]. However, we are able to outperform the re-implemented super vector based method by [4]. This also prove the proposed method's effectiveness. On analyzing the per class performance, we can see that the rigid objects are easier to classify than non-rigid objects. Besides, the proposed method improves the classification performance

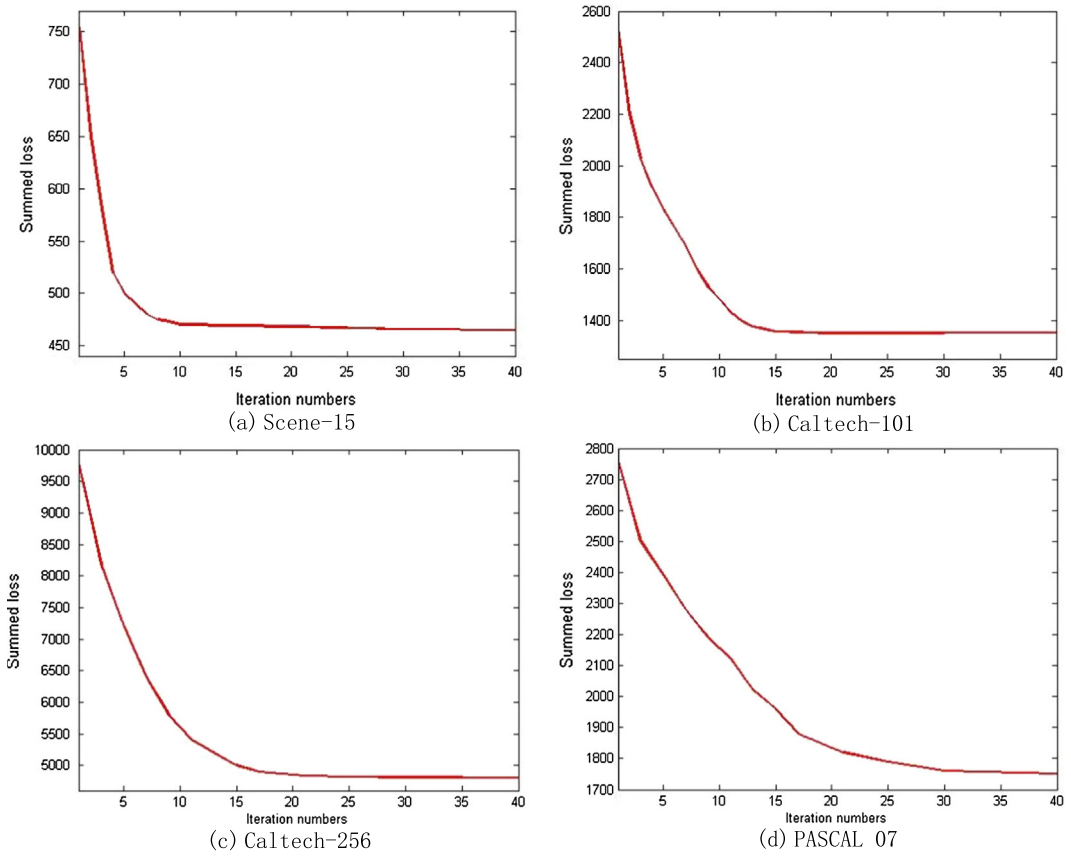

Fig. 6. The changes of the summed exponential losses on the four datasets with the increase of iterations.

mainly on the non-rigid objects over [4,7,42]. This is because non-rigid objects have large inter-class variations which should be modeled with different types of models. The proposed method can generate a series of models to solve this problem. We also give some example images that our method classified correctly and misclassified on the PASCAL VOC 2007 dataset in Fig. 5.

4.6. Convergence of B-ROL

We iteratively reweight the training images for classification. During each iteration, the corresponding codebooks and encoding parameters are alternatively optimized by minimizing the objective values. Besides, the summed exponential loss is reduced in each iteration. In this way, we can reduce the loss gradually. Moreover, because the summed loss is positive, the proposed B-ROL converges. To intuitively show the convergence of the proposed method, we plot the summed exponential losses on the four datasets in Fig. 6. With the iteration, the summed loss of training images decreases. Besides, the decrease rate varies from different datasets. We believe this is because some datasets are more difficult to classify than others. For example, of the four datasets, the PASCAL VOC 2007 dataset is the most difficult. Hence, the summed loss also decreases relatively slow. However, the Scene-15 dataset is easier to classify which needs only about 10 iterations to converge.

5. Conclusion

This paper proposed a novel image classification method by using boosted local features with random orientation and location selection. By randomly extract local features with varied orientation and location, we can get more types of information than pre-defined features for classification. We learn the codebook and encoding parameters by weighted sparse coding. The weights are determined by the predicted values of the learned classifiers and groundtruth in order to let the classifier pay more attention to the images which are hard to classify in the next iteration. The proposed method works in a boosting way to gradually improve classification performances. In this way, the proposed method can unify the local feature extraction, the codebook generation and the classifier training into a unified process for better classification accuracy. We conducted experiments on several public image datasets and compared with other the state-of-the-art methods to demonstrate the effectiveness of the proposed method.

Acknowledgements

This work is supported by National Basic Research Program of China (973 Program): 2012CB316400 and 2015CB351802, National Natural Science Foundation of China: 61303154, 61170127, 61272329, 61202234, 61303114, 61025011 and 61332016. The President Fund of UCAS. Beijing Municipal Natural Science Foundation of China No. 4132010. Specialized Research Fund for the Doctoral Program of Higher Education (No. 20130141120024).

References

[1] H. Bay, A. Ess, T. Tuytelaars, L.V. Gool, SURF: speeded up robust features, Comput. Vis. Image Underst. 110 (3) (2008) 346–359.
[2] O. Boiman, E. Shechtman, M. Irani, In defense of nearest-neighbor based image classification, in: CVPR, 2008.
[3] Leo Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[4] K. Chatfield, V. Lempitsky, A. Vedaldi, A. Zisserman, The devil is in the details: an evaluation of recent feature encoding methods, in: BMVC, 2011, pp. 1–12.
[5] N. Dalal, W. Triggs, Histograms of oriented gradients for human detection, in: Proc. CVPR, 2005.
[6] D. Donoho, For most large underdetermined systems of linear equations, the minimal L1 norm is also the sparsest solution, Commun. Pure Appl. Math. 59 (6) (2006) 797–829.
[7] M. Everingham, A. Zisserman, C. Williams, L. Van Gool, The PASCAL Visual Object Classes Challenge 2007 (VOC 2007) Results, Technical Report, Pascal Challenge, 2007.
[8] B. Fan, Q. Kong, T. Trzcinski, Z. Wang, C. Pan, P. Fua, Receptive fields selection for binary feature description, IEEE Trans. Image Process. 23 (6) (2014) 2583–2595.
[9] C. Farabet, C. Couprie, L. Najman, Y. LeCun, Learning hierarchical features for scene labeling, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1915–1929.
[10] Y. Freund, R. Iyer, R. Schapire, Y. Singer, An efficient boosting algorithm for combining preferences, J. Mach. Learn. Res. 4 (2003) 933–969.
[11] Y. Freund, R. Schapire, A short introduction to boosting, J. Jpn. Soc. Artif. Intell. 14 (5) (1999) 771–780.
[12] S. Gao, L. Chia, I. Tsang, Z. Ren, Concurrent single-label image classification and annotation via efficient multi-layer group sparse coding, IEEE Trans. Multimedia 16 (3) (2014) 762–771.
[13] S. Gao, I. Tsang, L. Chia, Laplacian sparse coding, hypergraph Laplacian sparse coding, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 92–104.
[14] J. Gemert, C. Veenman, A. Smeulders, J. Geusebroek, Visual word ambiguity, IEEE Trans. Pattern Anal. Mach. Intell. 32 (7) (2010) 1271–1283.
[15] G. Griffin, A. Holub, P. Perona, Caltech-256 Object Category Dataset, Technical Report, CalTech, 2007.
[16] G. Hinton, S. Osindero, Y. Teh, A fast learning algorithm for deep belief nets, Neural Comput. 18 (7) (2006) 1527–1554.
[17] R. Hong, M. Wang, Y. Gao, D. Tao, X. Li, X. Wu, Image annotation by multiple-instance learning with discriminative feature mapping and selection, IEEE Trans. Cybernet. 44 (5) (2014) 669–680.
[18] P. Indyk, R. Motwani, Approximate nearest neighbors: towards removing the curse of dimensionality, in: Proc. Symp. Theory Comput., 1998, pp. 604–613.
[19] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in: Proc. CVPR, 2006.
[20] Y. LeCun, K. Kavukcuoglu, C. Farabet, Convolutional networks and applications in vision, in: Proc. Int. Symp. Circuits Syst., 2010, pp. 253–256.
[21] H. Lee, A. Battle, R. Raina, A.Y. Ng, Efficient sparse coding algorithms, in: Proc. NIPS, 2006.
[22] L. Li, L. Fei-Fei, What, where and who? Classifying events by scene and object recognition, in: ICCV, 2007, pp. 1–8.
[23] Z. Li, J. Liu, H. Lu, Structure preserving non-negative matrix factorization for dimensionality reduction, Comput. Vis. Image Underst. 117 (9) (2013) 1175–1189.
[24] Z. Li, J. Liu, J. Tang, H. Lu, Robust structured subspace learning for data representation, IEEE Trans. Pattern Anal. Mach. Intell., http://dx.doi.org/10.1109/TPAMI.2015.2400461.
[25] Z. Li, J. Liu, Y. Yang, X. Zhou, H. Lu, Clustering-guided sparse structural learning for unsupervised feature selection, IEEE Trans. Knowl. Data Eng. 26 (9) (2014) 2138–2150.
[26] Z. Li, J. Liu, X. Zhu, T. Liu, H. Lu, Image annotation using multi-correlation probabilistic matrix factorization, in: ACM Multimedia, 2010, pp. 1187–1190.
[27] Z. Li, Y. Yang, J. Liu, X. Zhou, H. Lu, Unsupervised feature selection using nonnegative spectral analysis, in: AAAI, 2012.
[28] X. Liu, L. Zhang, M. Li, H. Zhang, D. Wang, Boosting image classification with LDA-based feature combination for digital photograph management, Pattern Recogn. 38 (6) (2005) 887–901.
[29] D. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.
[30] D. Lowe, Local Naive Bayes nearest neighbor for image classification, in: Proc. CVPR, 2012, pp. 3650–3656.
[31] J. Mairal, F. Bach, J. Ponce, G. Sapiro, Sparse representation for color image restoration, IEEE Trans. Image Process. 17 (1) (2008) 53–69.
[32] F. Moosmann, E. Nowak, F. Jurie, Randomized clustering forests for image classification, IEEE Trans. Pattern Anal. Mach. Intell. 30 (9) (2008) 1632–1646.
[33] B. Olshausen, D. Field, Emergence of simple-cell receptive field properties by learning a sparse code for natural images, Nature 381 (1996) 607–609.
[34] S. Paisitkriangkrai, C. Shen, Q. Shi, A. Hengel, RandomBoost: simplified multiclass boosting through randomization, IEEE Trans. Neural Netw. Learn. Syst. 25 (4) (2014) 764–779.
[35] K. Sande, T. Gevers, C. Snoek, Evaluating color descriptors for object and scene recognition, IEEE Trans. Pattern Anal. Mach. Intell. 32 (9) (2010) 1582–1596.
[36] R. Schapire, The strength of weak learnability, Mach. Learn. 5 (2) (1990) 197–227.
[37] L. Shao, L. Liu, X. Li, Feature learning for image classification via multiobjective genetic programming, IEEE Trans. Neural Netw. Learn. Syst. 25 (7) (2014) 1359–1371.
[38] J. Sivic, A. Zisserman, Video Google: a text retrieval approach to object matching in videos, in: ICCV, 2003.
[39] J. Tang, R. Hong, S. Yan, T. Chua, G. Qi, R. Jain, Image annotation by kNN-sparse graph-based label propagation over noisily tagged web images, ACM TIST 2 (2) (2011) 14.
[40] J. Tang, H. Li, G. Qi, T. Chua, Image annotation by graph-based inference with integrated multiple/single instance representation, IEEE Trans. Multimedia 12 (2) (2010) 131–141.
[41] J. Tang, Z. Zha, D. Tao, T. Chua, Semantic-gap-oriented active learning for multilabel image annotation, IEEE Trans. Image Process. 21 (4) (2012) 2354–2360.
[42] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: Proc. CVPR, 2010.
[43] M. Wang, R. Hong, X. Yuan, S. Yan, T. Chua, Movie2comics: towards a lively video content presentation, IEEE Trans. Multimedia 14 (3) (2012) 858–870.
[44] J. Wright, A. Yang, A. Ganesh, S. Satry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 210–227.
[45] J. Yang, K. Yu, Y. Gong, T. Huang, Linear spatial pyramid matching using sparse coding for image classification, in: CVPR, 2009, pp. 1794–1801.
[46] L. Yang, R. Jin, R. Sukthankar, F. Jurie, Unifying discriminative visual codebook generation with classifier training for object category recognition, in: CVPR, 2008.

[47] X. Yuan, S. Yan, Visual classification with multi-task joint sparse representation, in: CVPR, 2010, pp. 3493–3500.
[48] C. Zhang, J. Cheng, J. Liu, J. Pang, C. Liang, Q. Huang, Q. Tian, Object categorization in sub-semantic space, Neurocomputing 142 (2014) 248–255.
[49] C. Zhang, J. Liu, C. Liang, Z. Xue, J. Pang, Q. Huang, Image classification by non-negative sparse coding, correlation constrained low-rank and sparse decomposition, Comput. Vis. Image Underst. 123 (2014) 14–22.
[50] C. Zhang, J. Liu, Q. Tian, Y. Han, H. Lu, S. Ma, A boosting, sparsity-constrained bilinear model for object recognition, IEEE Multimedia 19 (2) (2012) 58–68.
[51] 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: CVPR, 2011, pp. 1673–1680.
[52] J. Zhang, M. Marszalek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: a comprehensive study, Int. J. Comput. Vis. 73 (2) (2007) 213–238.
[53] T. Zhang, Statistical behavior and consistency of classification methods based on convex risk minimization, Ann. Stat. 32 (1) (2004) 56–85.
[54] X. Zhou, K. Yu, T. Zhang, T. Huang, Image classification using super-vector coding of local image descriptors, in: ECCV, 2010, pp. 141–154.