Pattern Recognition Letters 45 (2014) 197–204

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Undoing the codebook bias by linear transformation with sparsity and F-norm constraints for image classification ☆

Chunjie Zhang [a], Chao Liang [b,*], Junbiao Pang [c], Yifan Zhang [d], Jing Liu [d], Lei Qin [e], Qingming Huang [a,e]

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

ARTICLE INFO

Article history:
Received 24 September 2013
Available online 13 April 2014

Keywords:
Codebook bias
Linear transformation
Sparsity
Alternative optimization

ABSTRACT

The bag of visual words model (BoW) and its variants have demonstrated their effectiveness for visual applications. The BoW model first extracts local features and generates the corresponding codebook where the elements of a codebook are viewed as visual words. However, the codebook is dataset dependent and has to be generated for each image dataset. Besides, when we only have a limited number of training images, the codebook generated correspondingly may not be able to encode images well. This requires a lot of computational time and weakens the generalization power of the BoW model. To solve these problems, in this paper, we propose to undo the dataset bias by linear codebook transformation in an unsupervised manner. To represent each point in the local feature space, we need a number of linearly independent basis vectors. We view the codebook as a linear transformation of these basis vectors. In this way, we can transform the pre-learned codebooks for a new dataset using the pseudo-inverse of the transformation matrix. However, this is an under-determined problem which may lead to many solutions. Besides, not all of the visual words are equally important for the new dataset. It would be more effective if we can make some selection and choose the discriminative visual words for transformation. Specifically, the sparsity constraints and the F-norm of the transformation matrix are used in this paper. We propose an alternative optimization algorithm to jointly search for the optimal linear transformation matrixes and the encoding parameters. The proposed method needs no labeled images from either the source dataset or the target dataset. Image classification experimental results on several image datasets show the effectiveness of the proposed method.

1. Introduction

The bag of visual words model (BoW) [1,2] plays a very important role for visual applications (*e.g.*, image classification, retrieval and segmentation). Basically, the BoW model can be divided into four components: local feature extraction, codebook generation, local feature encoding and histogram based image representation. It has been widely used on various datasets [3–9]. To consider the spatial information, spatial pyramid matching (SPM) [5] and its variants [10–15] are also widely used.

Although proven effective, there is one problem with the BoW model and its variants. The codebook has to be learned for each image dataset separately and the performances of directly using codebooks generated by other datasets are less competitive than using the codebook generated with the corresponding dataset. This is because the state-of-the-art image datasets are collected for particular purposes [16]. To overcome this problem, a lot of works [17–24] have been done and the usefulness of considering the dataset bias is widely proved. However, most of these methods ignore the codebook bias problem with different datasets and only try to adapt the pre-learned classifiers instead. In fact, if we take a close look at the four components of the BoW model, we can find that the codebook is the only component which varies from datasets. The other three components of the BoW model have no such dataset dependence. For example, dense SIFT feature is used for local feature extraction, sparse coding or nearest neighbor assignment is used for local feature encoding and images are represented

☆ This paper has been recommended for acceptance by F. Tortorella.

* Corresponding author. Tel./fax: +86 2768772100.

E-mail addresses: cjzhang@jdl.ac.cn (C. Zhang), cliang@whu.edu.cn (C. Liang), junbiao_pang@bjut.edu.cn (J. Pang), yfzhang@nlpr.ia.ac.cn (Y. Zhang), jliu@nlpr.ia.ac.cn (J. Liu), qinlei@ict.ac.cn (L. Qin), qmhang@jdl.ac.cn (Q. Huang).

http://dx.doi.org/10.1016/j.patrec.2014.04.002
0167-8655/© 2014 Elsevier B.V. All rights reserved.

by visual word histogram. Finally, SVM classifiers are trained for classification. Hence, if we can generate the codebook which performs consistently on different datasets, we will be able to make the BoW model less dataset dependent and improve the performance of visual applications using the BoW representation.

Researchers have also explored the generation of universal codebooks and then adapted it for specific visual applications [25–29]. However, there is one problem with this strategy. To learn a universal codebook, you should collect a universal image dataset. This costs a lot of labor and is also time consuming. The universal codebook is actually also dataset dependent as long as the universal image dataset is collected. Transfer learning [30–32] and semi-supervised learning [33–36] techniques are also used to alleviate this problem, however, most of them does not consider the codebook bias which makes the proposed algorithms less dataset independent.

To solve the codebook bias problem mentioned above, in this paper, we propose a novel linear transformation based unsupervised codebook adaption method. In linear algebra, to represent each point in a space, the number of basis vectors needed should be no less than the dimension of this space, depending on the linear independence of the basis vectors. Similarly, to represent one point in the local feature space, the number of basis vectors should be no less than the local feature space dimension. Hence, we follow the work of [37] and view each codebook as a linear transformation of these basis vectors. In this way, we can linearly transform the pre-learned codebooks for a new dataset using the pseudo-inverse of the corresponding transformation matrix. However, the transformation matrix often has thousands of values. This means searching for the optimal transformation matrix is a under-determined problem which may leads to many solutions. Besides, not all of the visual words are equally important, it is more effective if we can choose the most discriminative visual words from pre-learned codebooks for transformation. We use the sparsity constraints [10,12] and F-norm over the transformation matrixes in this paper for visual word selection. We use the F-norm constraints for two reasons. First, the optimization over the transformation matrixes is under-determined, adding some constraints can help solve the problem. Second, the F-norm is differentiable which means the optimization problem can be solved much easier than non-differentiable constraints. We propose an alternative optimization algorithm to jointly search for the optimal linear transformation matrixes and the encoding parameters. Note that the proposed method for undoing the codebook bias requires no training images from either the source dataset or the target dataset. To test the effectiveness of the proposed method, we conduct image classification experiments on several image datasets. The results show the effectiveness of codebook transformation for undoing the dataset bias. Fig. 1 gives the flowchart of the proposed method.

Compared with [37], the contributions of this paper lie in three aspects. First, we add F-norm to solve the under-determined problem of finding the optimal transformation matrix. Second, by jointly using sparsity constraints and F-norm regularization, we can transform the pre-learned codebooks more effectively than using sparsity constraints alone [37], especially when we have a limited number of images with the target dataset. Third, we are able to achieve better classification performance than [37] did.

The rest of this paper is organized as follows: in Section 2 we give the related work. The details of the proposed linear codebook transform method for undoing the codebook bias is given in Section 3. The experimental results are given in Section 4. Finally, we conclude in Section 5.

2. Related work

Recently, many image datasets have been introduced for various visual applications, such as the Bird dataset [3], the Butterfly dataset [4], the Scene-15 dataset [5], the Event dataset [6], the Indoor dataset [7], the Corel-5K dataset [8] and the Caltech-256 dataset [9]. A lot of works have been done to improve the image classification performances on these datasets. However, datasets were also blamed for hindering the improvement of classification performance [16]. To overcome this problem, a lot of works [17–24] have been done. Khosla et al. [17] proposed to undo the dataset bias by jointly learning the bias vectors and visual words' weights in a discriminative manner. An online domain adaption

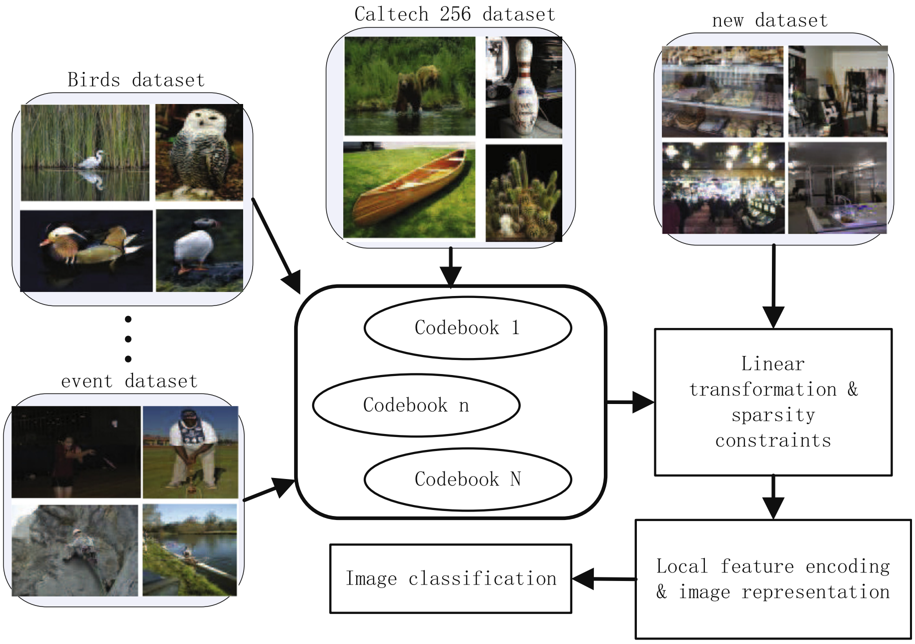

Fig. 1. Flowchart of the proposed linear codebook transformation for image classification method.

of cascade classifiers was proposed by Jain and Miller [18]. Kulis et al. [19] proposed an asymmetric kernel transformation based object categorization method. The dataset shift problem was systematically analyzed by Candela et al. [20]. Gopalan et al. [21] took an unsupervised approach to adapt the object categorization problem. Saenko et al. [22] tried to adapt object model of a particular visual domain to new domain by minimizing the effect of feature distribution discrepancy. A heterogeneous transfer learning algorithm was proposed by Zhu et al. [23] for image classification with good performance. To cope with the lack of training images, Wang et al. [24] proposed a dyadic knowledge transfer approach for cross-domain classifications. Although proven effective, all of the above mentioned methods made use of the training images from either the source dataset or the target dataset.

Researchers also explored the generation of universal codebook and then adapted it for specific visual applications [25–29]. Zhang et al. [25] tried to learn a general codebook and used the descriptive visual words and visual phases for visual applications. Spatial pyramid codebook was used by Zhang et al. [26] to combine the spatial information of local features. Perronnin et al. [27] first generated a universal codebook using the gaussian mixture model (GMM) and then adapted it for specific applications. Winn et al. [28] used the information bottleneck principle to obtain a more discriminative codebook. Chatfield et al. [29] systematically compared several the state-of-the-art codebook construction methods. However, the dataset bias problem is still unsolved after the imageset is collected.

Transfer learning [30–32] and semi-supervised learning [33–36] techniques were also used to alleviate the dataset bias problem. Yang et al. [30] tried to transfer the structured knowledge and achieved good performance. Pan and Yang [31] studied several transfer learning methods for undoing the dataset bias. A topographic subspace model with transfer learning was proposed by Liu et al. [32] for sparse representation. Chova et al. [33] proposed a semi-supervised one class SVM classifier to predict the categories of images. Guillaumin al. [34] used a multimodal semi-supervised strategy for classifying images by considering the discriminative information of different modalities. A sclable semi-supervised multiple kernel learning algorithm was proposed by Wang et al. [35] for mining the useful information while Sang et al. [36] used ternary semantic analysis strategy for image tag refinement. However, the labeled images are still needed. Besides, most of these methods does not explicitly consider the codebook bias problem for different datasets.

3. Linear codebook transformation for visual applications

In this section, we give the details of the proposed linear codebook transformation method to undo the codebook bias and apply it for image classification problems. The proposed method requires no training images from the source dataset or the target dataset.

3.1. Linear codebook transformation

In linear algebra, to represent each point in a space, the number of basis vectors needed should be no less than the dimension of this space, depending on the linear independence of the basis vectors. Similarly, to represent one point in the local feature space, the number of basis vectors should be no less than the local feature space dimensions. Suppose we have a set of basis vectors $B = [b_1, b_2, \ldots, b_Q] \in \mathbb{R}^{P\times Q}$ which can completely represent each local feature in this space. P is the dimension of local feature space and Q is the number of basis vectors with $Q > P$. Let $D_1 = [d_1^1, d_1^2, \ldots, d_1^M] \in \mathbb{R}^{P\times M}$ be a codebook generated using a particular dataset where M is the number of visual words. Since each visual word in codebook D_1 can be viewed as a point in the local feature space, each element of D_1 can be linearly represented by the basis vectors B as:

$$d_1^i = a_1^{i,1} b_1 + a_1^{i,2} b_2 + \cdots + a_1^{i,Q} b_Q, \quad \forall i = 1, \ldots, M \tag{1}$$

where $a^{i,1}$ is the linear combination parameters. This can be rewritten in a matrix form as:

$$D_1 = BA_1^T \tag{2}$$

where $A_1 = [a_j^{i,1}]_{i=1,\ldots M, j=1,\ldots,Q}$ is the corresponding linear transformation matrix. In this way, we can generate a codebook D_1 by linearly combining the basis vectors of local feature space. This can also be written as:

$$B = D_1 (A_1^T)^+ \tag{3}$$

where $(A_1^T)^+$ is the psedoinverse of matrix A_1^T. Similarly, we can generate a codebook D_2 as:

$$D_2 = BA_2^T = D_1 (A_1^T)^+ A_2^T \tag{4}$$

Let $A = A_2 A_1^+$, Eq. (4) can be rewritten as:

$$D_2 = D_1 A^T \tag{5}$$

Suppose we have learnt the codebook D_1 for dataset 1, to generate the codebook D_2 for dataset 2, all we need to do is to find the corresponding transformation matrix A. If the transformation matrix A has been learnt, we can use the corresponding codebook D_1 for local feature encoding. We use the sparse coding technique [38] in this paper as it has been shown to be very effective for encoding local features. Let $x \in \mathbb{R}^{P\times 1}$ be the local feature to be encoded, α is the corresponding sparse coding parameter with λ being the parameter which controls the sparsity of α as:

$$min_{\alpha, D_2} \|x - D_2 \alpha\|^2 + \lambda \|\alpha\|_1 \tag{6}$$

This can be optimized over α and A as:

$$min_{\alpha, A} \|x - D_1 A^T \alpha\|^2 + \lambda \|\alpha\|_1 \tag{7}$$

This problem can be solved efficiently by alternatively optimizing over α/A while keeping the other fixed. When α is fixed, Problem 7 equals to solving the following optimization problem as:

$$min_A \|x - D_1 A^T \alpha\|^2 \tag{8}$$

Let $\overrightarrow{A}^T = D_1 A^T$, Problem 8 can be rewritten as:

$$min_A \|x - \overrightarrow{A} \alpha\|^2 \tag{9}$$

When A is fixed, Problem 7 equals to solving the following optimization problem as:

$$min_\alpha \|x - D_1 A^T \alpha\|^2 + \lambda \|\alpha\|_1 \tag{10}$$

Since D_1 is pre-learned and A is fixed, let $D = D_1 A^T$, Problem 10 can be rewritten as:

$$min_\alpha \|x - D\alpha\|^2 + \lambda \|\alpha\|_1 \tag{11}$$

Problems 9 and 11 can be effectively optimized by the feature-sign search and the Lagrange dual algorithms proposed in [38]. In this way, we can transform the codebook D_1 generated using dataset 1 to dataset 2. However, the transformation of only one codebook is often too weak, especially when the two image datasets are quite different. It would be more effective if we can transform a number of codebooks instead.

3.2. Multiple codebook transformation for undoing dataset bias

Formally, suppose we have N pre-learned codebooks $D_1, \ldots, D_N$ generated using the corresponding image datasets. To encode local feature x, the optimization problem can be written similarly as:

$$min_{\alpha_n, A_n, n=1,2,\ldots,N} \left\| x - \sum_{n=1}^{N} D_n A_n^T \alpha_n \right\|^2 + \lambda_n \sum_{n=1}^{N} \|\alpha_n\|_1 \tag{12}$$

where λ_n is the sparsity constraint parameter for the nth dataset, α_n is the corresponding encoding parameter. However, the transformation matrixes also have influences for the performances. Hence, it would be more effective if we can impose some constraints on the transformation matrices. We use the popular F-norm in this paper because it is derivable and try to solve the following problem as:

$$min_{\alpha_n, A_n, n=1,2,\ldots,N} \left\| x - \sum_{n=1}^{N} D_n A_n^T \alpha_n \right\|^2 + \sum_{n=1}^{N} \lambda_n \|\alpha_n\|_1 + \sum_{n=1}^{N} \gamma_n \|A_n\|_F^2 \tag{13}$$

Let $\beta = [\alpha_1; \alpha_2; \ldots; \alpha_N]$, $\lambda = [\lambda_1; \ldots; \lambda_N]$, $\gamma = [\gamma_1; \gamma_2; \ldots; \gamma_N]$, $E = [D_1; D_2; \ldots; D_N]$ and $C = diag\{A_1, A_2, \ldots, A_N\}$, Problem 13 can be rewritten as:

$$min_{\beta, C} \|x - EC^T\beta\|^2 + \lambda\|\beta\|_1 + \gamma\|C\|_F^2 \tag{14}$$

Since E is pre-learned and fixed, this problem can be solved similarly as Problem 7 by alternatively optimizing over β and C. When β is fixed, Problem 14 equals to solving the following optimization problem as:

$$min_C \|x - EC^T\beta\|^2 + \gamma\|C\|_F^2 \tag{15}$$

with

$$\frac{\partial \|x - EC^T\beta\|^2 + \gamma\|C\|_F^2}{\partial C} = 2\beta\beta^T CE^T E - 2\beta x^T E + \gamma Tr(CC^H) \tag{16}$$

When C is fixed, the optimization of Problem 14 over β equals:

$$min_\beta \|x - EC^T\beta\|^2 + \lambda\|\beta\|_1 \tag{17}$$

This problem can be solved in a similar way as the feature-sign-search algorithm [38]. Algorithm 1 gives the proposed linear codebook transformation method for undoing the dataset bias.

Algorithm 1. The proposed linear codebook transformation algorithm for undoing the dataset bias

```
Input:
  The local features $X, \lambda_1, \ldots, \lambda_N, \gamma_1, \gamma_2; \ldots, \gamma_N, D_1; D_2, \ldots, D_N$,
  the threshold parameter $\theta$ and max iteration number
  $maxiter$;
Output:
  The learned $C$ and encoding parameters $\beta$;
1: for $iter = 1, 2, \ldots, maxiter$
2:   Find the optimal $C$ with encoding parameters $\beta$ fixed by
     solving Problem 15 with Eq. (16);
3:   Find the optimal encoding parameters $\beta$ with codebook $C$
     fixed by solving Problem 17;
4:   Check whether the decrease of objective function of
     Problem 14 falls below the threshold $\theta$.
       If unsatisfied
         go to step 1
     Else
         stop;
5: return $\beta, C$;
```

3.3. Max pooling based image representation for image classification

After learning the corresponding linear transformation matrix C, we can use it to encode local features by solving Problem 17 with C fixed. To represent images using these encoded parameters, we follow the popular max pooling scheme [39,40] to extract information from local features. The max pooling has been proven effective when combined with sparse coding for image representation. It chooses the max value of each dimension of the encoded sparse parameters within a particular image region. Besides, to combine the spatial information of local features, we use the spatial pyramid matching (SPM) technique [5]. The first three pyramids with $2^L \times 2^L, L = 0, 1, 2$ are used in this paper.

To test the effectiveness of the proposed linear codebook transformation method for undoing the dataset bias, we evaluate image classification performances on several public image datasets. This is achieved by training a set of classifiers. We follow Yang et al. [39] and use the one-vs-all linear SVM classifier as it has been shown to be effective with sparse coding techniques [12,39,40].

4. Experiments

4.1. Experimental setup

To evaluate the effectiveness of the proposed linear codebook transformation method, we conduct image classification performances on several public image datasets: the Bird dataset [3], the Butterfly dataset [4], the Scene-15 dataset[5], the Event dataset [6], the Indoor dataset [7], the Corel-5K dataset [8], the Caltech-256 dataset [9], Lazebnik's texture dataset [41] and the PASCAL VOC 2007 dataset [42]. The Bird dataset has 100 images each of six different classes (egrets, mandarin ducks, snowy owls, puffins, toucans, and wood ducks). The Butterfly dataset consists of 619 images of seven classes. The Scene-15 dataset has fifteen classes ranging from 200 to 400 images per class. The Event dataset has eight sports event categories with the number of images in each category varying from 137 to 250. The indoor dataset has 15,620 images of 67 classes. The Corel-5K dataset consists of 50 categories of images while the Caltech-256 dataset has 30,607 images of 256 classes. The Lazebnik's texture dataset [41] has 1000 images of 25 different textures. The PASCAL VOC 2007 dataset has around 10,000 images of twenty classes which are more difficult to classify than the above datasets. Fig. 2 shows some example images of these image datasets.

We densely extract SIFT features [43] of size 16 × 16 pixels with an overlap of 6 pixels for all the datasets except the PASCAL VOC 2007 dataset. For the PASCAL VOC 2007 dataset, we densely extract SIFT features with an overlap of 4 pixels on various patch sizes. The patch size varies from 16 to 64 pixels with a step of 4 pixels. For the nine datasets, we randomly choose 50, 16, 100, 70, 80, 50, 30, 10 and 50 images per class from the corresponding image dataset as the training set and use the rest of images as the testing set. These training numbers are chosen as the same as [3–9,41]. This process is repeated for five times to get reliable results. The codebook size for each dataset is set to 1024. The λ and γ are two important parameters which control the transformation of codebooks. Larger λ and γ increase the sparsity of the transformation process than smaller λ and γ. Since we use nine datasets for evaluation, we have eight λ_i and $\gamma_i, i = 1, \ldots, 8$ to set. We set $\lambda_i, i = 1, \ldots, 8$ to the same value and $\gamma_i, i = 1, \ldots, 8$ to the same value, respectively. We try to find the optimal parameters by grid search. λ_i ranges from 0.1 to 1.5 with a step of 0.1 while γ_i ranges from 0.1 to 1 with a step of 0.1. We use the classification rate for performance evaluation. Instead of re-implementing the algorithms, we directly use

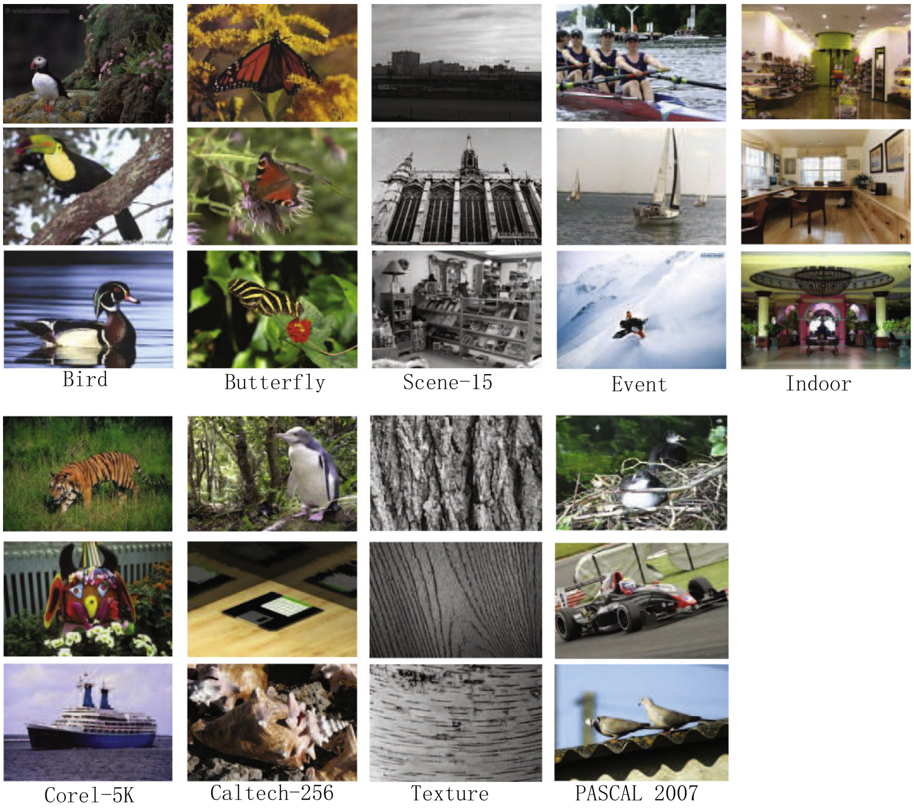

Fig. 2. Example images for the Bird, Butterfly, Scene-15, Event, Indoor, Corel-5K and Caltech-256 datasets.

the performances reported in [3–9] for fair comparison. We use CT-SFC as the abbreviation of the proposed linear codebook transfer with sparsity and F-norm constraints for undoing the dataset bias algorithm.

4.2. Image classification performance

We give the performance comparison of the proposed CT-SFC with the method [37] in Table 1. When generating the codebook for one particular dataset with the proposed CT-SFC method, we use all the other codebooks generated from the corresponding image datasets. We also give the performances of combining the visual words (Combined-VW) and SIFT features (Combined-SIFT) on these datasets. The Combined-VW is generated by first clustering the visual words of these image datasets and then using the cluster centers as the new codebook. The Combined-SIFT is generated by clustering the SIFT features extracted from these image datasets directly. The codebook sizes are also set to 1024 for consistency. To demonstrate the effectiveness of the proposed method over [37], we also give the performances with and without F-norm constraints over the transformation matrix. The horizontal row indicates the dataset that the codebook is generated while the vertical column indicates the dataset that the classification is performed. We also give the performance of the proposed linear codebook transferring algorithm on the corresponding vertical column by transferring the codebook generated by the other image datasets.

We can see from Table 1 that the codebook generated by one particular image dataset achieves the best classification performance on the corresponding dataset. However, we can achieve better results by transferring the codebooks instead of directly using the codebooks generated by other datasets. In fact, the codebook generated by the corresponding image dataset is the upper performance bound of the proposed codebook transfer algorithm. Besides, the proposed CT-SFC is able to outperform Combined-VW and Combined-SIFT. We believe this is because the combined methods used all the information from various datasets without selection. The noisy information is also used by the combination methods for classification which hinders the final performances.

Besides, the relative improvement of the proposed codebook transfer algorithm varies depending on the image datasets. For example, the proposed method achieves equal performances on the Butterfly Scene-15, Event and Corel-5K datasets while performs one percent less on the Bird, Caltech-256 and Indoor dataset compared with the codebook generated by the corresponding datasets. We believe this is because the difficulties of these datasets are different. The Caltech-256 dataset, the Indoor dataset and the PASCAL VOC 2007 dataset are more difficult to classify that the Corel-5K dataset and the Butterfly dataset. This is not only because of the increased number of image classes but also because of the large intra and inter class variations. Moreover, the Combined-VM method performs not as well as Combined-SIFT method. This is because k-means clustering prefers large clusters. Since visual words are generated by k-means clustering, the re-clustering of visual words will result in large clusters that encode most of the local features. The Combined-SIFT method makes clustering on the SIFT features directly which helps to alleviate this problem. However, using all the SIFT features from various datasets for one particular dataset classification may introduce noise and decrease the classification performance. The proposed CT-SFC method can also cope with codebooks of different size. Of course, the codebook size also has influence on the codebook transfer performance. Basically, a larger codebook can help to classify images than a relatively small codebook. However, the computational cost also increases when a larger codebook is used.

On analyzing the details of the classification performance, we can have three conclusions. First, compared with the codebook generated by the corresponding dataset, the use of other datasets generated codebooks perform better on similar image classes than

Table 1
Mean classification rates on the image datasets: Bird, Butterfly, Scene-15, Event, Indoor, Corel-5 K, Caltech-256, Lazebnik's texture dataset and PASCAL VOC 2007 dataset. The horizontal row indicates the dataset that the codebook is generated while the vertical column indicates the dataset that the classification is performed. The Combined-VW is generated by first clustering the visual words of these image datasets and then use the cluster centers as the new codebook. The Combined-SIFT is generated by clustering the SIFT features extracted from these image datasets directly. We also give the performance of the proposed linear codebook transfer algorithm by transfering the codebook generated by the other image datasets on the corresponding vertical column. CT-SC: codebook transfer with sparse constraints [35], CT-SFC: codebook transfer with sparse and F-norm constraints (Problem 14). We split the table into two sub-tables for space reasons. The bold values are used to indicate the best classification performances.

datasets	Bird	Butterfly	Scene-15	Event	Indoor	Corel-5K	Caltech-256	Texture	PASCAL07
Bird	**0.83 ± 0.07**	0.72 ± 0.09	0.78 ± 0.06	0.79 ± 0.07	0.39 ± 0.08	0.61 ± 0.04	0.29 ± 0.06	0.48 ± 0.12	0.28 ± 0.07
Butterfly	0.75 ± 0.08	**0.72 ± 0.08**	0.77 ± 0.06	0.78 ± 0.07	0.38 ± 0.06	0.60 ± 0.04	0.29 ± 0.05	0.49 ± 0.14	0.27 ± 0.06
Scene-15	0.72 ± 0.06	0.69 ± 0.08	**0.79 ± 0.05**	0.78 ± 0.08	0.40 ± 0.07	0.62 ± 0.05	0.28 ± 0.06	0.53 ± 0.12	0.29 ± 0.06
Event	0.73 ± 0.06	0.73 ± 0.07	0.74 ± 0.07	**0.81 ± 0.07**	0.41 ± 0.07	0.61 ± 0.03	0.31 ± 0.06	0.50 ± 0.13	0.30 ± 0.05
Indoor	0.70 ± 0.08	0.72 ± 1.01	0.77 ± 0.05	0.79 ± 0.08	**0.43 ± 0.06**	0.62 ± 0.04	0.32 ± 0.07	0.55 ± 0.12	0.32 ± 0.06
Corel-5K	0.72 ± 0.09	0.70 ± 0.09	0.76 ± 0.06	0.78 ± 0.08	0.39 ± 0.08	**0.67 ± 0.05**	0.31 ± 0.05	0.48 ± 0.14	0.31 ± 0.06
Caltech-256	0.71 ± 0.08	0.72 ± 0.07	0.75 ± 0.05	0.79 ± 0.09	0.40 ± 0.05	0.64 ± 0.04	**0.38 ± 0.06**	0.56 ± 0.11	0.32 ± 0.07
Texture	0.63 ± 0.12	0.61 ± 0.10	0.65 ± 0.08	0.70 ± 0.12	0.32 ± 0.09	0.54 ± 0.08	0.20 ± 0.11	**0.77 ± 0.09**	0.25 ± 0.05
PASCAL07	0.71 ± 0.08	0.71 ± 0.08	0.75 ± 0.05	0.79 ± 0.09	0.35 ± 0.07	0.63 ± 0.06	0.28 ± 0.07	0.57 ± 0.13	**0.40 ± 0.04**
Combined-VW	0.65 ± 0.09	0.67 ± 0.08	0.71 ± 0.04	0.72 ± 0.08	0.33 ± 0.09	0.58 ± 0.06	0.26 ± 0.07	0.56 ± 0.09	0.30 ± 0.05
Combined-SIFT	0.77 ± 0.08	0.69 ± 0.09	0.75 ± 0.06	0.78 ± 0.07	0.38 ± 0.08	0.64 ± 0.05	0.33 ± 0.06	0.68 ± 0.11	0.35 ± 0.06
CT-SC [37]	0.81 ± 0.07	0.71 ± 0.08	0.78 ± 0.05	0.80 ± 0.06	0.41 ± 0.07	0.66 ± 0.04	0.35 ± 0.06	0.74 ± 0.11	0.38 ± 0.05
CT-SFC	**0.82 ± 0.05**	**0.72 ± 0.09**	**0.79 ± 0.07**	**0.81 ± 0.08**	**0.42 ± 0.05**	**0.67 ± 0.06**	**0.37 ± 0.06**	**0.76 ± 0.10**	**0.39 ± 0.05**

on dissimilar image classes. For example, the Scene-15 dataset can be roughly divided into the indoor class and the outdoor class. When using the Indoor dataset generated codebook for classification, the performances are comparable or a little less that the Scene-15 generated codebook on the indoor class (*e.g.,* kitchen, livingroom, store). However, for the outdoor class (*e.g.,* highway/mountain), the performance is less competitive than on the indoor dataset. Second, the proposed codebook transferring method can alleviate this problem by transferring the elements of datasets with similar image classes for better image representation. We can achieve comparable classification rates by codebook transformation (*e.g.,* the Scene-15 dataset and the Corel-5K dataset). These results prove the effectiveness of transferring codebook for undoing the dataset bias and improving the classification performance. Third, the addition of F-norm constraints on the linear transformation matrix improves the codebook transformation performance. The training sample numbers are relatively small compared with the transformation matrix parameters while the F-norm constraints help to alleviate this problem and improve the classification rates.

Compared with CT-SC [37], we add F-norm regularization over the transformation matrix. This strategy helps to transform the useful information more efficiently, especially when we have limited number of training images. This also means the proposed CT-SFC method can also be used in an incremental way. We can gradually adapt the pre-learned codebooks to new image datasets with the addition of training images. To show the effectiveness of this property, we also give the classification performance with the number of training images on these image datasets in Fig. 3. The solid/dotted line represents the performance of CT-SFC/CT-SC over these image datasets, respectively. This is achieved by randomly choosing the training images per class for five times and uses the mean of the classification rates for evaluation. We can have three conclusions from Fig. 3. First, the proposed method can gradually improve the classification performance with the increase of training images. Second, the use of F-norm helps to improve the performance over CT-SC [37], especially when we only have a limited number of training images. Third, the relative improvements over these datasets are varied. We believe this is because of the relative difficulties of different image datasets. The Caltech-256, Indoor and PASCAL VOC 207 datasets are more difficult to classify, hence needs more training images to get reliable performance. Besides, the reconstruction error for local features is reduced to about 20% to 40% on average compared with directly using the source dataset's codebook. This means we can encode local features more efficiently.

To intuitively show the influences of different image datasets, we sum the absolute value of α_i corresponding to each dataset and plot it in Fig. 4. The horizontal rows represent the image datasets to be transferred while the vertical columns represent the influences of the other eight datasets. The diagonal values are zero. We can have three conclusions from Fig. 4. First, image datasets with similar objects have relatively larger influences. For example, Butterfly and Bird are more correlated than other datasets, the Indoor dataset has relatively larger influence for the Scene15's codebook. Second, the influence of Texture dataset is relatively low. This is because the other datasets do not have so much texture related features. To alleviate the codebook bias, we should choose similar or related datasets to transfer. Third, when learning the PASCAL07's codebook, the influences of different datasets are more

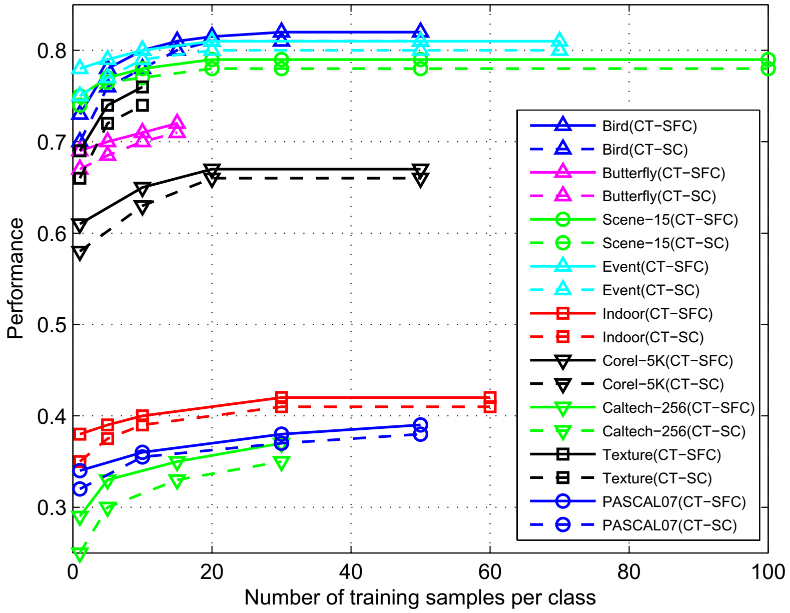

Fig. 3. The performance changes with the number of training images per class for the Bird, Butterfly, Scene-15, Event, Indoor, Corel-5K, Caltech-256, Texture and PASCAL VOC 2007 datasets. The solid/dotted line represents the performance of CT-SFC/CT-SC, respectively. It is best viewed in color. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

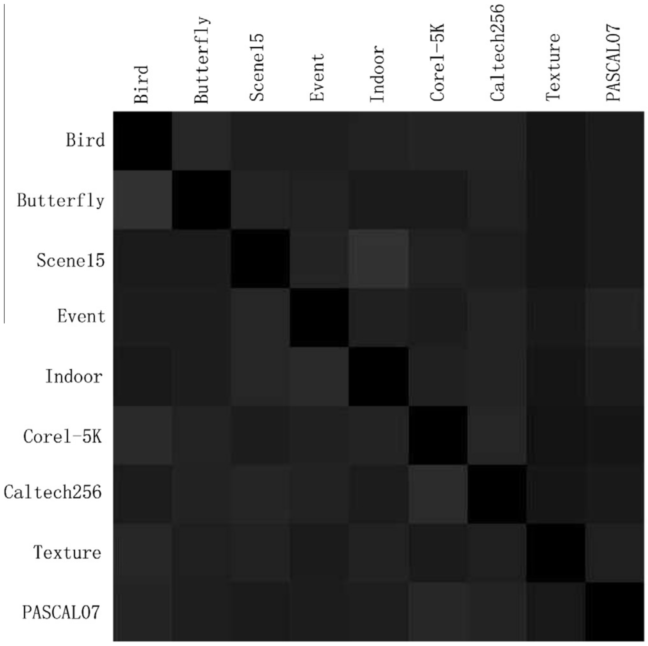

Fig. 4. The influences of different datasets for codebook transfer. The horizontal rows represent the image datasets to be transferred while the vertical columns represent the influences of the other eight datasets. Codebooks with larger influences are brighter than those with smaller influences.

evenly distributed. This is because the PASCAL07 dataset has various images and harder to represent well compared with other datasets.

4.3. Computational complexity analysis

The computational complexity of k-means clustering is $O(J * k * t)$, where J is the number of local features, k is the number of cluster centers and t is the number of iterations. To generate a codebook, hundreds of thousands of local features (J) are needed. The computational cost of the proposed CT-SFC is low compared with k-means clustering. This is because the optimization of Eq. (13) over each local feature x can be jointly considered as:

$$min_{\alpha_{n,j},A_n,n=1,2,\ldots,N}\sum_{j=1}^{J}\left\{\left\|x_j-\sum_{n=1}^{N}D_nA_n^T\alpha_{n,j}\right\|^2+\sum_{n=1}^{N}\lambda_n\|\alpha_{n,j}\|_1+\sum_{n=1}^{N}\gamma_n\|A_n\|_F^2\right\} \quad (18)$$

Without causing confusion, let $\gamma=[\beta_1,\ldots,\beta_J]$ with $\beta_j=[\alpha_{1,j};\alpha_{2,j},\ldots,\alpha_{N,j}], j=1,\ldots,J$, Problem 18 can also be optimized alternatively over β and C. When γ is fixed, Problem 18 can be optimized as:

$$min_C\sum_{j=1}^{J}\{\|x_j-EC^T\beta_j\|^2+\gamma\|C\|_F^2\} \quad (19)$$

with

$$\frac{\partial\sum_{j=1}^{J}\{\|x_j-EC^T\beta_j\|^2+\gamma\|C\|_F^2\}}{\partial C}=\sum_{j=1}^{J}\left\{2\beta_j\beta_j^TCE^TE-2\beta_jx_j^TE+\gamma Tr(CC^H)\right\} \quad (20)$$

When C is fixed, Problem 18 can be solved as:

$$min_\gamma\sum_{j=1}^{J}\left\{\left\|x_j-\beta_j^TEC^T\right\|^2+\lambda\|\beta_j\|_1\right\} \quad (21)$$

This can be solved in a similar way as Problem 17. Generally, the proposed method takes about 1/10 ∽ 1/5 time of k-means clustering when 100,000 local features are used. This means we can save the time for codebook generation and concentrate on the design of classification models in order to improve the performance.

5. Conclusion

In this paper, we propose a novel linear codebook transformation method to undo the codebook bias. This is achieved by linearly transforming the pre-learned codebooks for new visual applications. We also impose sparsity and F-norm constraints for discriminative visual words transformation. An alternative optimization algorithm is proposed to jointly learn the optimal transformation matrixes and encoding parameters. Experimental results on several public datasets demonstrate the effectiveness of the proposed method.

Acknowledgement

This work is supported by the National Basic Research Program of China (973 Program): 2012CB316400, National Natural Science Foundation of China: 61303154, 61303114, 61202234, 61202325, 61272329, 61133003 and 61332016. China Postdoctoral Science Foundation: 2012M520434, 2013T60156, 2013M530350. 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: Proceedings of IEEE International Conference on Computer Vision, UK, 2003, pp. 1470–1477.

[2] G. Csurka, C. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints, in: Proceedings of ECCV International Workshop on Statistical Learning in Computer Vision, 2004, pp. 1–22.

[3] S. Lazebnik, C. Schmid, J. Ponce, A maximum entropy framework for part-based texture and object recognition, in: Proceedings of IEEE International Conference on Computer Vision, China, 2005, pp. 832–838.

[4] S. Lazebnik, C. Schmid, J. Ponce, Semi-local affine parts for object recognition, in: Proceedings of the British Machine Vision Conference, 2004, pp. 959–968.

[5] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, USA, 2006, pp. 2169–2178.

[6] L. Li, L. Fei-Fei, What, where and who? Classifying event by scene and object recognition, in: Proceedings of IEEE International Conference on Computer Vision, 2007, pp. 1–8.

[7] A. Quattoni, A. Torralba, Recognizing indoor scenes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, USA, 2009, pp. 413–420.

[8] G. Liu, Z. Li, L. Zhang, Y. Xu, Image retrieval based on micro-structure descriptor, Pattern Recognit. 44 (9) (2011) 2123–2133.

[9] G. Griffin, A. Holub, P. Perona, Caltech-256 Object Category Dataset, Technical Report, CalTech, 2007.

[10] 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.

[11] M. Zerler, G. Taylor, R. Fergus, Adaptive deconvolutional networs for mid and high level feature learning, in: Proceedings of IEEE International Conference on Computer Vision, 2011, pp. 2018–2025.

[12] 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: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, USA, 2011, pp. 1673–1680.

[13] G. Sharma, F. Jurie, C. Schmid, Discriminative spatial saliency for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, USA, 2012, pp. 3506–3513.

[14] S. Battiato, G. Farinella, G. Gallo, D. Ravi, Scene categorization using bag of textons on spatial hierarchy, in: Proceedings of the IEEE International Conference on Image Processing, USA, 2008, pp. 2536–2539.

[15] 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.

[16] A. Torralba, A. Efros, Unbiased look at dataset bias, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, USA, 2011, pp. 1521–1528.

[17] A. Khosla, T. Zhou, T. Malisiewicz, A. Efros, A. Torralba, Undoing the damage of dataset bias, in: Proceedings of the European Conference on Computer Vision, 2012, pp. 158–171.

[18] V. Jain, E. Miller, Online domain adaption of a pre-trained cascade of classifiers, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, USA, 2011, pp. 577–584.

[19] B. Kulis, K. Saenko, T. Darrell, What you saw is not what you get: domain adaptation using asymmetric kernel transforms, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, USA, 2011, pp. 1785–1792.

[20] J. Candela, M. Sugiyama, A. Schwaighofer, N. Lawrence, Dataset Shift in Machine Learning, MIT Press, 2009.

[21] R. Gopalan, R. Li, R. Chellappa, Domain adaptation for object recognition: an unsupervised approach, in: Proceedings of IEEE International Conference on Computer Vision, 2011, pp. 999–1006.

[22] K. Saenko, B. Kulis, M. Fritz, T. Darrell, Adapting visual category models to new domains, in: Proceedings of the European Conference on Computer Vision, 2010, pp. 213–226.

[23] Y. Zhu, Y. Chen, Z. Lu, S. Pan, G. Xue, Y. Yu, Q. Yang, Heterogeneous transfer learning for image classification, in: Proceedings of the AAAI Conference on Artificial Intelligence, USA, 2011.

[24] H. Wang, F. Nie, H. Huang, C. Ding, Dyadic transfer learning for cross-domain image classification, in: Proceedings of IEEE International Conference on Computer Vision, 2011, pp. 551–556.

[25] S. Zhang, Q. Huang, G. Hua, S. Jiang, W. Gao, Q. Tian, Building contextual visual vocabulary for large-scale image applications, ACM Multimedia (2010) 501–510.

[26] C. Zhang, S. Wang, Q. Huang, J. Liu, C. Liang, Q. Tian, Image classification using spatial pyramid robust sparse coding, Pattern Recognit. Lett. 34 (9) (2013) 1046–1052.

[27] F. Perronnin, C. Dance, G. Csurka, M. Bressan, Adapted vocabularies for generic visual categorization, in: Proceedings of the European Conference on Computer Vision, 2006, pp. 464–475.

[28] K. Winn, A. Criminisi and T. Minka, Object categorization by learned universal visual dictionary, in: Proceedings of IEEE International Conference on Computer Vision, 2005, pp. 1800–1807.

[29] K. Chatfield, V. Lempitsky, A. Vedaldi, A. Zisserman, The devil is in the details: an evaluation of recent feature encoding methods, in: Proceedings of the British Machine Vision Conference, 2011, pp. 1–12.

[30] Q. Yang, V. Zheng, B. Li, H. Zhou, Transfer learning by reusing structured knowledge, AI Mag. 32 (2) (2011) 95–106.

[31] S. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 20 (10) (2010) 1345–1359.

[32] Y. Liu, J. Cheng, C. Xu, H. Lu, Building topographic subspace model with transfer learning for sparse representation, Neurocomputing 73 (10–12) (2010) 1662–1668.

[33] L. Chova, G. Valls, J. Mari, J. Calpe, Semi-supervised one-class support vector machines for classification of remote sensing data, IEEE Trans. Geosci. Remote Sens. 48 (8) (2010) 3188–3197.

[34] M. Guillaumin, J. Verbeek, C. Schmid, Multimodal semi-supervised learning for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 902–909.

[35] S. Wang, S. Jiang, Q. Huang, Q. Tian, S3MKL: scalable semi-supervised multiple kernel learning for image data mining, ACM Multimedia (2010) 163–172.

[36] J. Sang, C. Xu, J. Liu, User-aware image tag refinement via ternary semantic analysis, IEEE Trans. Multimedia 14 (3–2) (2012) 883–895.

[37] C. Zhang, Y. Zhang, S. Wang, J. Pang, C. Liang, Q. Huang, Q. Tian, Undo the codebook bias by linear transformation for visual applications, in: Proceedings of the 21st ACM Internation Conference on Multimedia, Spain, 2013, pp. 533–536.

[38] H. Lee, A. Battle, R. Raina, A. Ng, Efficient sparse coding algorithms, in: Proceedings of the Neural Information Processing Systems, 2007.

[39] J. Yang, K. Yu, Y. Gong, T. Huang, Linear spatial pyramid matching using sparse coding for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, USA, 2009, pp. 1794–1801.

[40] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, USA, 2010, pp. 3360–3367.

[41] S. Lazebnik, C. Schmid, J. Ponce, A sparse texture representation using local affine regions, IEEE Trans. Pattern Anal. Mach.Intell. 27 (8) (2005) 1265–1278.

[42] M. Everingham, A. Zisserman, C. Williams, L. Van Gool, The PASCAL Visual Object Classes Challenge 2007 (VOC 2007) Results, Technical report, Pascal Challenge, 2007.

[43] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vision 60 (2004) 91–110.