Fine-Grained Image Classification via Low-Rank Sparse Coding With General and Class-Specific Codebooks

Chunjie Zhang, Chao Liang, Liang Li, Jing Liu, Qingming Huang, and Qi Tian, *Senior Member, IEEE*

***Abstract*— This paper tries to separate fine-grained images by jointly learning the encoding parameters and codebooks through low-rank sparse coding (LRSC) with general and class-specific codebook generation. Instead of treating each local feature independently, we encode the local features within a spatial region jointly by LRSC. This ensures that the spatially nearby local features with similar visual characters are encoded by correlated parameters. In this way, we can make the encoded parameters more consistent for fine-grained image representation. Besides, we also learn a general codebook and a number of class-specific codebooks in combination with the encoding scheme. Since images of fine-grained classes are visually similar, the difference is relatively small between the general codebook and each class-specific codebook. We impose sparsity constraints to model this relationship. Moreover, the incoherences with different codebooks and class-specific codebooks are jointly considered. We evaluate the proposed method on several public image data sets. The experimental results show that by learning general and class-specific codebooks with the joint encoding of local features, we are able to model the differences among different fine-grained classes than many other fine-grained image classification methods.**

Manuscript received July 15, 2015; accepted March 18, 2016. Date of publication April 7, 2016; date of current version June 15, 2017. This work was supported in part by the Open Project of Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, in part by the National Basic Research Program of China under Grant 2012CB316400 and Grant 2015CB351802, and in part by the National Natural Science Foundation of China under Grant 61272329, Grant 61303114, Grant 61303154, Grant 61332016, and Grant 61402431. *(Corresponding author: Chunjie Zhang.)*

C. Zhang is with Institute of Automation, Chinese Academy of Sciences, 100190, Beijing, China. He is with the School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing 100049, China, and also with the Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing 100190, China (e-mail: zhangcj@ucas.ac.cn).

C. Liang is with the National Engineering Research Center for Multimedia Software, School of Computer, Wuhan University, Wuhan 430072, China (e-mail: cliang@whu.edu.cn).

L. Li is with the School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing 100049, China, and also with the Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing 100190, China (e-mail: liang.li@vipl.ict.ac.cn).

J. Liu is with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: jliu@nlpr.ia.ac.cn).

Q. Huang is with the School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing 100049, China, also with the Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing 100190, China, and also with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: qmhuang@jdl.ac.cn).

Q. Tian is with the Department of Computer Sciences, The University of Texas at San Antonio, San Antonio, TX 78249 USA (e-mail: qitian@cs.utsa.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2016.2545112

***Index Terms*— Fine grained, image representation, semantic space, visual recognition.**

I. Introduction

As a typical problem in computer vision, image classification is widely studied by researchers with various methods [1]–[4]. Of these methods, the bag-of-visual-words model is often used using scale invariant feature (SIFT) [5] features. These local features are quantized to form a histogram representation of images followed by classifier training and evaluation. Although very effective for general image classification, the relationships among local features are treated independently. This becomes a drawback when being applied to a fine-grained classification task where the differences between images are more subtle. This means the resulting bag-of-words (BoW) representation of images is cluttered together, which makes the trained classifiers unable to separate them apart properly. Hence, how to represent images in a more discriminative way becomes urgent.

There are two stages with the BoW scheme for image representation. First, a codebook is learned by minimizing the summed reconstruction errors of local features with some constraints (nearest neighbor [1], soft assignment [2], and sparsity [6], etc). Since the number of local features is large, it is impossible to learn the codebook with all the local features simultaneously. Usually, local features are randomly selected to learn the codebook accordingly. Second, after the codebook is learned, each local feature is encoded accordingly with the learned codebook. This strategy works well for general image classification tasks. However, for the fine-grained classification task, there are more pieces of information that can be explored.

Since images of fine-grained classes are visually and contextually similar, only one codebook is unable to model this well even with a large codebook size. It is more effective to learn a number of codebooks for joint and per-class representations, respectively [7], [8]. Usually the codebook incoherence is also combined for balanced image representation with general codebook and per-class codebooks. Although very effective, the differences between general codebook and each per-class codebook are often ignored. Since images of fine-grained classes are visually very similar, the differences are relatively small between the general codebook and each class-specific codebook. As to the encoding of local features, many works treat each local feature individually [1], [4], [6]. Since local features are incoherently spatially and contextually correlated, it is more reasonable to encode them jointly with spatial constraints [4], [9], [10] and visual similarities [2], [11], [12].

Since visual features are incoherently correlated with the degenerated structure [13], the low-rank technique is also used for encoding [14]–[16]. This technique works well when visual features are correlated. However, how to collect similar visual features for low-rank decomposition is still an open problem. Researchers tried to define various rules with class constraints [14] or segmentation [15]. However, the spatial information of local features within different images is ignored. If we can automatically bundle the visual features together for decomposition, we are able to represent images more efficiently.

To solve the problems mentioned above, in this paper, we propose a novel fine-grained image classification method by leveraging the low-rank sparse coding (LRSC) technique and combine it with general and class-specific codebook generation. We learn a general codebook and a number of codebooks per class for joint encoding of local features. The general codebook represents the universal information of all classes while each class-specific codebook encodes the distinctive character of each class. To model the differences between general codebook and each class-specific codebook, the sparsity constraint is used along with the codebook incoherences. As to the encoding of local features, the low-rank constraint is leveraged to consider the spatial and structure information of local features within a particular image region. Instead of treating each region separately, we encoded the corresponding regions of the same position within the training images to make use of the spatial information. We conduct fine-grained image classification experiments on several public image data sets and the results show the effectiveness of the proposed method.

The main contributions of this paper lie in following aspects. First, we construct a general codebook and a number of class-specific codebooks by exploring the sparsity correlation and incoherences between general codebook and class-specific codebooks. This makes the resulting codebooks more discriminative and representative. Second, local features within one image region are jointly encoded with low-rank constraints to model the correlations of local features. We densely select image regions with overlap to improve the discriminative power and robustness of the encoding parameters. The encoding parameters are max pooled for each region. In this way, we are able to outperform many baseline methods for the fine-grained image classification task on several public image data sets.

The rest of this paper is organized as follows. Related work is given in Section II. The details of the proposed method for fine-grained image classification via low-rank sparse coding with general and class-specific codebooks (LRSC-GCC) are given in Section III. We give the experimental results and analysis in Section IV. Finally, we conclude in Section V.

II. Related Work

Various approaches [1]–[4] had been proposed for image classification. Sivic and Zisserman [1] quantized the local features with k-means clustering to harvest the discriminative power of local features. To alleviate the information loss of hard assignment, van Gemert *et al.* [2] proposed to soft-assignment local features based on their relative distances to a number of visual words. Boiman *et al.* [3] used the local features directly without quantization to preserve the discriminative information. However, the computational cost was relatively high compared with quantization-based methods. To combine the spatial information, Lazebnik *et al.* [4] used an image partition strategy by dividing images with multiscales. This technique was effective and easy to implement. These methods used the SIFT feature [5] for local region description. To reduce the information loss during the quantization process, sparse coding was proposed by Yang *et al.* [6] with max pooling to simulate the human brain and achieved improved performances.

To classify the images of fine-grained classes, more than one general codebook was needed [7], [8]. Ramirez *et al.* [7] tried to classify images by learning a number of codebooks with structured incoherences and shared features. Gao *et al.* [8] extended this by learning a general codebook and a reweighting scheme to balance the reconstruction error with the incoherence influences. The experimental results proved its effectiveness. A randomization strategy was proposed by Yao *et al.* [17] for fine-grained categorization. Berg and Belhumeur [18] studied the distinctiveness of fine-grained images for human understanding with impressive results. Since the codebook played an important role for classification, many works [19]–[21] had been done to improve the representative power of codebooks. Winn *et al.* [19] tried to learn a universal visual dictionary and adapt it for specific classification tasks, while Moosmann *et al.* [20] proposed to use random clustering forests to classify images. Yang *et al.* [21] tried to learn a sparse variation codebook with a single training image for face recognition.

After the codebook was learned, how to encode local features was another problem that needed to be solved. Many previous works treated each local feature individually [1], [4], [6] by nearest neighbor assignment or sparse coding. As local features were correlated, researchers also explored how to jointly encoded the local features [9]–[12], [22]–[24]. Zhang *et al.* [9] used the nearby local features with Harr-like transformation, while Jiang *et al.* [10] tried to randomly select the contextual clues of local features and applied it for fast object search. Gao *et al.* [11] modeled the similarity information among local features and used Laplacian sparse coding for image classification. Yuan and Yan [12] tried to combine the multitask classification with sparse representation. An affine-constrained group sparse coding technique was proposed by Chi *et al.* [22] for classification with multiple-input samples. Chiang *et al.* [23] combined the sparse coding with multiattribute for dictionary learning, while Bristow *et al.* [24] proposed a fast convolutional sparse coding method to speed up the computation. Since local features were correlated, they exhibited some degenerated structure [13]. Inspired by this observation, Peng *et al.* [13] used the low-rank and sparse decomposition for face recognition and achieved good performances. Zhang *et al.* [14] applied it to the histogram representation of images and combined non-negative sparse coding for image classification, while Zhang *et al.* [15] used the segmented image regions for joint encoding with low-rank and sparse constraints. An accelerated low-rank recovery

method was proposed by Mu *et al.* [16] using the random projection technique. The use of locality information was also studied very widely [25]–[28]. Wang *et al.* [25] used nearby visual words for local feature encoding instead of using all the visual words with improved classification accuracy. Boureau *et al.* [26] studied the multiway pooling method by combining the local information for image classification, while Shabou and Le-Borgne [27] combined locality and spatial regularization for scene categorization. Bulò and Kontschieder [28] used the image pixels directly for image labeling with the neural decision forests.

The spatial and structure information [4], [29]–[34] were also very important for efficient classification. The spatial pyramid matching was widely used to boost the performance, while Grauman and Darrell [29] proposed the pyramid matching in feature space. Torralba *et al.* [30] developed a context-based vision system for recognition. Wu *et al.* [31] tried to bundle the local features for partial-duplicate image search. The random walk strategy was used by Ni *et al.* [32] to explore the contextual information of histogram representation of images, while Belongie *et al.* [33] used shape context for shape matching. Zhang *et al.* [34] applied the sparse coding with spatial partition with improved classification accuracy. Sparsity constraint was often used to boost the classification performances [35]–[37]. Chen *et al.* [35] used sparse representation for image classification, while Xiao *et al.* [36] combined it with kernel reconstruction of ICA. Li *et al.* [37] used ordinal distance metric learning for image ranking with good performance. Bai *et al.* [38] proposed to use feature-fusion-based marginalized kernels, which improved the performance, while Zhang *et al.* [39] used structural feature selection with a shape model for object detection. Yu and Grauman [40] proposed a local learning approach for fine-grained visual comparisons, while Qian *et al.* [41] used multistage metric learning for fine-grained categorization with good performance. The use of deep ranking technique is proposed by Wang *et al.* [42]. To avoid explicit usage of the part annotation, Krause *et al.* [43] proposed an automatic part detection scheme.

III. Low-Rank Sparse Coding With General and Class-Specific Codebooks for Fine-Grained Image Classification

In this section, we give the details of the proposed method. Image regions are densely selected with overlap. We then generate the general codebook and class-specific codebooks by minimizing the reconstruction error with discrepancy constraints between general codebook and class-specific codebook. The codebook incoherences are also used for reliable codebook learning. Besides, local features are jointly encoded within each image region to combine the spatial and structure information. Max pooling is then used to get the image representation and we train one-versus-all linear support vector machine (SVM) classifiers for class prediction.

A. Low-Rank Sparse Coding With General and Class-Specific Codebooks

Let $\boldsymbol{X} = [\boldsymbol{x_1}, \ldots, \boldsymbol{x_N}]$ be the N local features, $\boldsymbol{B}$ is the codebook, and $\boldsymbol{\alpha_n}$ is the encoded parameter for the nth local feature. The sparse coding tries to learn the optimal codebook and the corresponding encoding parameters $\boldsymbol{\alpha_n} \in \mathbb{R}^{d\times 1}, n = 1, \ldots, N$ by solving the optimization problem as

$$[\boldsymbol{B}, \boldsymbol{\alpha_n}] = \arg\min_{\boldsymbol{B},\boldsymbol{\alpha_n}} \sum_{n=1}^{N} \left\|\boldsymbol{x}_n^T - \boldsymbol{B}\boldsymbol{\alpha_n}\right\|_2^2 + \lambda_1 \|\boldsymbol{\alpha_n}\|_1 \quad (1)$$

with λ_1 as the parameter for sparsity control. Let $\mathbf{A} = [\alpha_1; \ldots; \alpha_N]$. This can be written in a matrix form as

$$[\boldsymbol{B}, \mathbf{A}] = \arg\min_{\boldsymbol{B},\mathbf{A}} \|\boldsymbol{X} - \boldsymbol{B}\mathbf{A}\|_F^2 + \lambda_1 \|\mathbf{A}\|_{1,1} \quad (2)$$

with $|\mathbf{A}\|_{1,1} = \sum_{n=1}^{N} \|\boldsymbol{\alpha_n}\|_1$.

Instead of learning single codebook, we try to generate multiple codebooks for fine-grained classification. Especially, a general codebook $\boldsymbol{B^0} \in \mathbb{R}^{r_0\times d}$ and C class-specific codebooks $\boldsymbol{B^c} \in \mathbb{R}^{r_c\times d}, c = 1, \ldots, C$, are learned, where C is the number of image classes. Let $\boldsymbol{X^c}$ be the local features of the cth class and $\hat{\boldsymbol{X}} = [\boldsymbol{X^1}, \ldots, \boldsymbol{X^C}]$ be the concatenated local features of all classes, then we concatenate $\boldsymbol{B^0}$ and $\boldsymbol{B^c}$, $c = 1, \ldots, C$, together for feature encoding. We can use the same procedure as Problem 2 directly for codebook learning and local feature encoding. However, for fine-grained tasks, the visual discrepancy between the general codebook and each class-specific codebook is relatively small and correlated. We use the sparsity constraints to model this relationship. Let $\hat{\boldsymbol{B}} = [\boldsymbol{B^0}, \boldsymbol{B^1}, \ldots, \boldsymbol{B^C}]$, then we can learn the general codebook $\boldsymbol{B^0}$ and C class-specific codebooks $\boldsymbol{B^c}$ by solving the optimization problem as

$$[\hat{\boldsymbol{B}}, \hat{\boldsymbol{A}}] = \arg\min_{\hat{\boldsymbol{B}},\hat{\boldsymbol{A}}} \|\hat{\boldsymbol{X}} - \hat{\boldsymbol{B}}\hat{\boldsymbol{A}}\|_F^2 + \lambda_1 \|\hat{\boldsymbol{A}}\|_{1,1} + \gamma_1 \sum_{c=1}^{C} \|\boldsymbol{B^0} - \boldsymbol{B^c}\|_{1,1}. \quad (3)$$

To avoid the local features being only concentrated on the class-specific codebook, we follow the strategy as in [7] and [8] and add an incoherent term to the optimization problem 3 as:

$$[\hat{\boldsymbol{B}}, \hat{\boldsymbol{A}}] = \arg\min_{\hat{\boldsymbol{B}},\hat{\boldsymbol{A}}} \|\hat{\boldsymbol{X}} - \hat{\boldsymbol{B}}\hat{\boldsymbol{A}}\|_F^2 + \lambda_1 \|\hat{\boldsymbol{A}}\|_{1,1} + \gamma_1 \sum_{c=1}^{C} \|\boldsymbol{B^0} - \boldsymbol{B^c}\|_{1,1} + \gamma_2 \sum_{i\neq j} \|\boldsymbol{B^i}^T \boldsymbol{B^j}\|_F^2 \quad (4)$$

where γ_1 is the parameter for codebook discrepancy control and γ_2 is the parameter for codebook incoherence weighting.

Nearby local features are likely to be visually similar and hence exhibit some low-rank character. Besides, the spatial location is also very useful for encoding. Hence, we add low-rank constraints into Problem 4 for each image region of the same class. Suppose we divide each image into P regions, then for each region, we view the local features within one region as belonging to this region. Since image regions are densely extracted with overlap, one local feature may belong to multiple image regions. By abuse of notation, let $\hat{\boldsymbol{X}}_{\boldsymbol{p,c}}$ be the concatenated local features of class c that belong to the pth image region of the same position, with

$\hat{\boldsymbol{X}} = [\hat{\boldsymbol{X}}_{\boldsymbol{1,1}}, \ldots, \hat{\boldsymbol{X}}_{\boldsymbol{p,c}}, \ldots, \hat{\boldsymbol{X}}_{\boldsymbol{P,C}}]$. $\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}$ are the corresponding encoding parameters and $\hat{\boldsymbol{A}} = [\hat{\boldsymbol{A}}_{\boldsymbol{1,1}}, \ldots, \hat{A}_{p,c}, \ldots, \hat{\boldsymbol{A}}_{\boldsymbol{P,C}}]$. Problem 4 can be rewritten as

$$[\hat{\boldsymbol{B}}, \hat{\boldsymbol{A}}] = \arg\min_{\hat{\boldsymbol{B}}, \hat{\boldsymbol{A}}} \sum_{p,c} \|\hat{\boldsymbol{X}}_{\boldsymbol{p,c}} - \hat{\boldsymbol{B}}\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}\|_F^2 + \lambda_1 \|\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}\|_{1,1} + \lambda_2 \|\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}\|_* + \gamma_1 \sum_{c=1}^{C} \|\boldsymbol{B^0} - \boldsymbol{B^c}\|_{1,1} + \gamma_2 \sum_{i \neq j} \|\boldsymbol{B^i}^T \boldsymbol{B^j}\|_F^2 \quad (5)$$

where $\|.\|_*$ is the matrix's nuclear norm.

The proposed method differs from [8] in two aspects. First, for fine-grained image classification, the differences between the general codebook and the class-specific codebook are relatively small and correlated. We use the sparsity constraint to model this relationship and jointly optimize for the general and class-specific codebooks. In this way, we are able to model the subtle differences among fine-grained images. Second, we jointly encode spatially nearby local features with low-rank constraints. This ensures that visually similar local features have correlated encoding parameters with spatial and structural constraints. This ensures us to have more consistent and discriminative encoding parameters for better image representation and classification.

B. Alternative Optimization

It is difficult to simultaneously learn the codebooks and the encoding parameters. Hence, we adopt the alternative optimization strategy and try to learn the codebooks/parameters while keeping parameters/codebooks fixed. When the encoding parameters $\hat{\boldsymbol{A}}$ are fixed, Problem 5 equals the following optimization problem as:

$$[\hat{\boldsymbol{B}}] = \arg\min_{\hat{\boldsymbol{B}}} \sum_{p,c} \|\hat{\boldsymbol{X}}_{\boldsymbol{p,c}} - \hat{\boldsymbol{B}}\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}\|_F^2 + \gamma_1 \sum_{c=1}^{C} \|\boldsymbol{B^0} - \boldsymbol{B^c}\|_{1,1} + \gamma_2 \sum_{i \neq j} \|\boldsymbol{B^i}^T \boldsymbol{B^j}\|_F^2. \quad (6)$$

This problem is still hard to optimize as the general codebook $\boldsymbol{B^0}$ and the class-specific codebook $\boldsymbol{B^c}$ are correlated. We alleviate this problem by optimizing over the general codebook $\boldsymbol{B^0}$ and each class-specific codebook $\boldsymbol{B^c}$ alternatively by fixing the others.

1) Learning the General Codebook: When $\boldsymbol{B^c}$, $c = 1, \ldots, C$ are fixed, let $\tilde{\boldsymbol{X}}_{\boldsymbol{p,c}} = \hat{\boldsymbol{X}}_{\boldsymbol{p,c}} - [\boldsymbol{B^1}, \ldots, \boldsymbol{B^C}]\tilde{\boldsymbol{A}}_{\boldsymbol{p,c}}$, where $\tilde{\boldsymbol{A}}_{\boldsymbol{p,c}}$ are the parameters associated with the corresponding codebooks except the general codebook. We use $\tilde{\boldsymbol{A}}^-_{\boldsymbol{p,c}}$ to represent the parameters associated with the general codebook, which can be obtained by removing $\tilde{\boldsymbol{A}}_{\boldsymbol{p,c}}$ from $\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}$. Problem 6 then equals

$$[\boldsymbol{B^0}] = \arg\min_{\boldsymbol{B^0}} \sum_{p,c} \left\|\tilde{\boldsymbol{X}}_{\boldsymbol{p,c}} - \boldsymbol{B^0}\tilde{\boldsymbol{A}}^-_{\boldsymbol{p,c}}\right\|_F^2 + \gamma_1 \sum_{c=1}^{C} \|\boldsymbol{B^0} - \boldsymbol{B^c}\|_{1,1}. \quad (7)$$

This can be optimized using the feature-sign search strategy [44] in the matrix level.

2) Learning the Class-Specific Codebook: When the general codebook $\boldsymbol{B^0}$ is fixed, let $\bar{\boldsymbol{X}}_{\boldsymbol{p,c}} = \hat{\boldsymbol{X}}_{\boldsymbol{p,c}} - [\boldsymbol{B^0}, \boldsymbol{B^1}, \boldsymbol{B^{c-1}}, \boldsymbol{B^{c+1}}, ., \boldsymbol{B^C}]\bar{\boldsymbol{A}}_{\boldsymbol{p,c}}$, where $\bar{\boldsymbol{A}}_{p,c}$ are the parameters associated with the corresponding codebooks except the cth class-specific codebook. We use $\bar{\boldsymbol{A}}^-_{\boldsymbol{p,c}}$ to represent the parameters associated with the cth class-specific codebook, which can be obtained by removing $\bar{\boldsymbol{A}}^-_{\boldsymbol{p,c}}$ from $\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}$. We can optimize for the cth codebook $\boldsymbol{B^c}$ while keeping the other class-specific codebooks fixed as

$$[\boldsymbol{B^c}] = \arg\min_{\boldsymbol{B^c}} \sum_{p,c} \left\|\bar{\boldsymbol{X}}_{\boldsymbol{p,c}} - \boldsymbol{B^c}\bar{\boldsymbol{A}}^-_{\boldsymbol{p,c}}\right\|_F^2 + \gamma_1 \|\boldsymbol{B^0} - \boldsymbol{B^c}\|_{1,1} + \gamma_2 \sum_{i \neq c} \|\boldsymbol{B^i}^T \boldsymbol{B^c}\|_F^2 \quad (8)$$

which can be optimized in a similar way as Problem 7 for each class-specific codebook.

3) Learning the Encoding Parameters: When the codebooks are fixed, Problem 5 equals

$$[\hat{\boldsymbol{A}}] = \arg\min_{\hat{\boldsymbol{A}}} \sum_{p,c} \|\hat{\boldsymbol{X}}_{\boldsymbol{p,c}} - \hat{\boldsymbol{B}}\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}\|_F^2 + \lambda_1 \|\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}\|_{1,1} + \lambda_2 \|\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}\|_*. \quad (9)$$

We can optimize Problem 9 directly. However, there are usually too many local features to be encoded. Besides, since we extract image regions with overlap, each local feature may belong to many image regions. Hence, we choose to encode the local features for each image region with each class jointly instead of optimizing over all the regions. Besides, since the nuclear norm and the sparsity constraints are nonsmooth, it cannot be optimized directly. We follow the strategy as in [13] and [15] and use two slack variables with equality constraints as:

$$[\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}, \mathbf{D_1}, \mathbf{D_2}] = \arg\min_{\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}} \|\hat{\boldsymbol{X}}_{\boldsymbol{p,c}} - \hat{\boldsymbol{B}}\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}\|_F^2 + \lambda_1 \|\mathbf{D_2}\|_{1,1} + \lambda_2 \|\mathbf{D_1}\|_* \quad \text{s.t. } \hat{\boldsymbol{A}}_{\boldsymbol{p,c}} = \mathbf{D_1};\quad \hat{\boldsymbol{A}}_{\boldsymbol{p,c}} = \mathbf{D_2}. \quad (10)$$

This can be optimized with the inexact augmented Lagrange multiplier method (IALM) with the Lagrangian function as

$$\begin{aligned} &L(\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}, \mathbf{D_1}, \mathbf{D_2}) \\ &= \|\hat{\boldsymbol{X}}_{\boldsymbol{p,c}} - \hat{\boldsymbol{B}}\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}\|_F^2 + \lambda_1 \|\mathbf{D_2}\|_{1,1} \\ &\quad + \lambda_2 \|\mathbf{D_1}\|_* + \operatorname{tr}\left(\mathbf{Y}_1^T(\hat{\boldsymbol{A}}_{\boldsymbol{p,c}} - \mathbf{D_1})\right) + w_1 \|\hat{\boldsymbol{A}}_{\boldsymbol{p,c}} - \mathbf{D_1}\|_F^2 \\ &\quad + \operatorname{tr}\left(\mathbf{Y}_2^T(\hat{\boldsymbol{A}}_{\boldsymbol{p,c}} - \mathbf{D_2})\right) + w_2 \|\hat{\boldsymbol{A}}_{\boldsymbol{p,c}} - \mathbf{D_2}\|_F^2. \end{aligned} \quad (11)$$

This can be optimized by alternatively solving for $\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}$, $\mathbf{D_1}$, and $\mathbf{D_2}$ and the multipliers $\mathbf{Y}_1$, $\mathbf{Y}_2$, w_1, and w_2.

When updating $\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}$, Problem 11 equals

$$\begin{aligned} \hat{\boldsymbol{A}}_{\boldsymbol{p,c}} = \arg\min_{\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}} &\|\hat{\boldsymbol{X}}_{\boldsymbol{p,c}} - \hat{\boldsymbol{B}}\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}\|_F^2 \\ &+ \operatorname{tr}(\mathbf{Y}_1^T(\hat{\boldsymbol{A}}_{\boldsymbol{p,c}} - \mathbf{D_1})) + w_1 \|\hat{\boldsymbol{A}}_{\boldsymbol{p,c}} - \mathbf{D_1}\|_F^2 \\ &+ \operatorname{tr}(\mathbf{Y}_2^T(\hat{\boldsymbol{A}}_{\boldsymbol{p,c}} - \mathbf{D_2})) + w_2 \|\hat{\boldsymbol{A}}_{\boldsymbol{p,c}} - \mathbf{D_2}\|_F^2. \end{aligned} \quad (12)$$

We can get $\hat{\boldsymbol{A}}_{\boldsymbol{p,c}}$ as

$$\begin{aligned} \hat{\boldsymbol{A}}_{\boldsymbol{p,c}} &= \mathbf{G}_1\mathbf{G}_2 \\ \mathbf{G}_1 &= (\hat{\boldsymbol{B}}^T\hat{\boldsymbol{B}} + w_1\hat{\boldsymbol{I}} + w_2\hat{\boldsymbol{I}})^{-1} \\ \mathbf{G}_2 &= \hat{\boldsymbol{B}}^T\hat{\boldsymbol{X}}_{\boldsymbol{p,c}} - \frac{1}{2}\mathbf{Y}_1 - \frac{1}{2}\mathbf{Y}_2 + w_1\mathbf{D}_1 + w_2\mathbf{D}_2. \end{aligned} \quad (13)$$

$\mathbf{D_1}$ can be updated by solving problem as

$$\mathbf{D_1} = \arg\min_{\mathbf{D_1}} \lambda_2 \|\mathbf{D_1}\|_* + \mathrm{tr}\big(\mathbf{Y}_1^T(\hat{\boldsymbol{A}}_{\boldsymbol{p},\boldsymbol{c}} - \mathbf{D_1})\big) + w_1 \|\hat{\boldsymbol{A}}_{\boldsymbol{p},\boldsymbol{c}} - \mathbf{D_1}\|_F^2 \tag{14}$$

with the optimal $\mathbf{D_1}$ as

$$\mathbf{D_1} = \mathcal{T}_{\frac{\lambda_2}{2w_1}}\left(\hat{\boldsymbol{A}}_{\boldsymbol{p},\boldsymbol{c}} + \frac{1}{2w_1}\mathbf{Y}_1\right) \tag{15}$$

where $\mathcal{T}_\lambda(\mathbf{A})$ is the singular value soft-thresholding operator as $\mathcal{T}_\lambda(\mathbf{A}) = \mathbf{U_A}\mathcal{S}_\lambda(\sum_{\mathbf{A}})\mathbf{V_A}^T$ with $\mathbf{U_A}\sum_{\mathbf{A}}\mathbf{V_A}^T$ being the singular value decomposition of $\mathbf{A}$. $\mathcal{S}_\lambda(\mathbf{A})$ is a soft-threshold operator of matrix $\mathbf{A}$ with each of the items calculated as $\mathcal{S}_\lambda(\mathbf{A_{i,j}}) = \mathrm{sign}(\mathbf{A_{i,j}})\max(|\mathbf{A_{i,j}} - \lambda|, 0)$.

To solve for the optimal $\mathbf{D_2}$, we can rewrite Problem 11 as

$$\mathbf{D_2} = \arg\min_{\mathbf{D_2}} \lambda_1 \|\mathbf{D_2}\|_{1,1} + \mathrm{tr}\big(\mathbf{Y}_2^T(\hat{\boldsymbol{A}}_{\boldsymbol{p},\boldsymbol{c}} - \mathbf{D_2})\big) + w_2 \|\hat{\boldsymbol{A}}_{\boldsymbol{p},\boldsymbol{c}} - \mathbf{D_2}\|_F^2 \tag{16}$$

which can then be solved as

$$\mathbf{D_2} = \mathcal{S}_{\frac{\lambda_1}{2w_2}}\left(\hat{\boldsymbol{A}}_{\boldsymbol{p},\boldsymbol{c}} + \frac{1}{2w_2}\mathbf{Y}_2\right). \tag{17}$$

The multipliers can then be updated as

$$\begin{aligned} \mathbf{Y}_1 &= \mathbf{Y}_1 + 2W_1(\hat{A}_{p,c} - \mathbf{D_1}), \quad w_1 = \rho w_1 \\ \mathbf{Y}_2 &= \mathbf{Y}_2 + 2W_2(\hat{A}_{p,c} - \mathbf{D_2}), \quad w_2 = \rho w_2 \end{aligned} \tag{18}$$

where ρ is a predefined constant. In this way, we can learn the optimal codebooks and the corresponding encoding parameters, which can then be used for image representation and class prediction. We can reduce the objective values of Problem 5 for each step when optimizing over the general codebook, the class-specific codebooks, and the encoding parameters. Since we alternatively optimize over the codebooks and encoding parameters, we can gradually reduce the objective values. Besides, since the objective value of Problem 5 is non-negative, the final objective value can converge gradually provided that enough alternative optimization steps are made.

We extract image representation using image regions directly with max pooling. The max pooling strategy has been proven effective for choosing the discriminative or representative patterns for various tasks [6], [8], [25]. For each image region, we choose the maximum absolute value of each dimension of the encoding parameters as the image region's representation $\mathbf{h_p}$, $p = 1, \ldots, P$. These image regions are concatenated in a fixed order to get the final image representation $\mathbf{h} = [\mathbf{h_1}; \ldots; \mathbf{h_P}]$. Since we densely select image regions, the resulting image representation contains more spatial information than hard partition of images. To predict image classes, we train one-versus-all linear SVM classifiers. The image is predicted as the class that has the largest positive response.

4) Computational Complexity: The time complexity of the general codebook $\boldsymbol{B^0}$ learning is of $O(r_0 d)$ with the storage complexity as $O(dN)$. Similarly, for each class-specific codebook $\boldsymbol{B^c}$, the time and storage complexity are $O(r_c d)$ and $O(dN)$, respectively. As to the learning of encoding parameters, the time and storage complexity are of $O(d^3)$ as the inverse of matrix should be computed when learning $\hat{\boldsymbol{A}}_{\boldsymbol{p},\boldsymbol{c}}$. Since we often extract relatively more local features compared with the local feature's dimension, the computations of both $\mathbf{D_1}$ and $\mathbf{D_2}$ have a complexity of $O(Nr^2)$. For the server with two Intel Xeon E5-2650 CPUs and 128-GB memories, the learning of codebooks takes about one day, while the encoding of new images takes less than 1 min for each image.

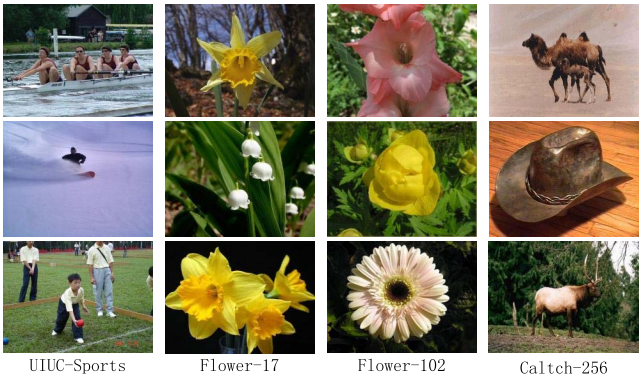


Fig. 1. Example images of the UIUC-Sports data set, the Flower-17 data set, the Flower-102 data set, and the Caltech-256 data set.

IV. Experiments

To evaluate the effectiveness of the proposed LRSC-GCC-based image classification method, we conduct image classification experiments on several public data sets: the UIUC-Sports data set [45], the Flower-17 data set [46], the Flower-102 data set [47], and the Caltech-256 data set [48]. Fig. 1 shows some example images of the four data sets.

A. Experimental Setup

For each image, we first resize it to 300 × 250 pixels. We densely extract local features of 16×16 pixels with 6 pixels overlap. The extracted local features are then normalized with the ℓ_2 norm. As to the image region selection, we densely choose image regions of multiscales with the minimum scale set to 64 × 64 pixels with 16 pixels overlap. In this way, about 430 image regions are chosen for one image. The codebook size is set to 1024 for general and class-specific codebooks. Instead of only extracting SIFT features of gray images, we also extract the color SIFT features (RGB-SIFT, HSV-SIFT, C-invariant SIFT, and the opponent SIFT) as in [49]. The corresponding image representations are concatenated for joint classification. We follow the parameter settings as in [6] and [25] and empirically set λ_1 to 0.3 for the UIUC-Sports data set and the Flower-17 data set. 0.4 is used for the other two data sets. γ_2 is set to 0.1. λ_2 and γ_1 are chosen by cross validation with the parameters ranging from 0.1 to 1.5 with a step of 0.2. The maximum iteration number is set to 50 for the four data sets. We randomly select the train/test images for performance evaluation and repeat this random selection process several times to get reliable results. For fair comparison, we choose to compare with other methods using the reported results with the same experimental setup instead of reimplementing them. The performance is evaluated with the average of per-class classification accuracy.

TABLE I
PERFORMANCE COMPARISONS ON THE UIUC-SPORTS DATA SET

Algorithms	Performance
ScSPM[6]	82.74 ± 1.46
LScSPM[11]	85.31 ± 0.51
LRSC[15]	88.17 ± 0.85
LLC[25]	83.09 ± 1.30
HIK+SVM[50]	83.54 ± 1.13
CSDL [8]	86.54 ± 0.56
GCC	87.23 ± 0.68
LR	88.84 ± 0.65
LR-GCC	$\mathbf{92.58 \pm 0.53}$

B. UIUC-Sports Data Set

The UIUC-Sports data set has eight classes of sports images as *Badminton*, *Bocce*, *Croquet*, *Polo*, *Rock climbing*, *Rowing*, *Sailing*, and *Snow boarding*. There are 1792 images with the number of each class varying from 137 to 250. Following the setup as in [45], we randomly select 70 images per class for training and use the rest of images for testing. This process is repeated ten times. Table I gives the performance comparison of the proposed method with other methods on the UIUC-Sports data set.

We can see from Table I that the proposed LRSC-GCC method is able to outperform many of the state-of-the-art methods. Compared with sparse coding [6], the joint encoding of local features and learning codebooks are necessary for classification performance improvement. Besides, LRSC-GCC is able to improve over LRSC by about 4.4% for learning general and class-specific codebooks. The results in Table I prove the effectiveness of the proposed LRSC-GCC method.

To evaluate the contributions of each component of the proposed method for classification, we also give performances of two baseline methods in Table I: the first method generates the general and class-specific codebooks without low-rank constraints (GCC), while the second method uses low-rank constraints with only the general codebook (LRSC). By imposing low-rank constraints, we are able to model the correlations for local feature encoding, and hence, we are able to improve the performance. Besides, by learning both the general and class-specific codebooks, we are able to model the subtle differences of fine-grained images and get more discriminative encoding parameters compared with only using one general codebook. We also give the average of confusion matrixes and the boxplot of LRSC-GCC on the UIUC-Sports data set in Figs. 2 and 3, respectively. The proposed method performs better on some classes (e.g., *Sailing* and *Polo*) than other classes (e.g., *Rock climbing*). This is because the intraclass variation of rock climbing is larger than the *Sailing* and *Polo* classes. It is relatively harder to model image classes with larger intraclass variations.

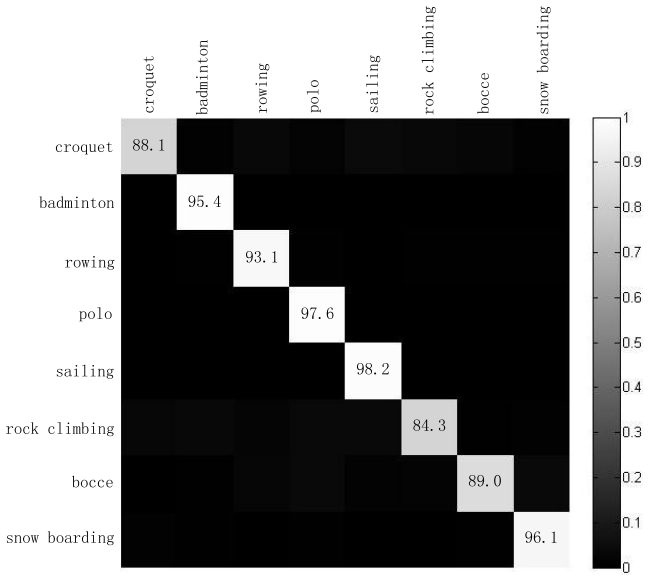


Fig. 2. Confusion matrix of the proposed LRSC-GCC method on the UIUC-Sports data set. The diagonal values indicate the per-class classification rate (%).

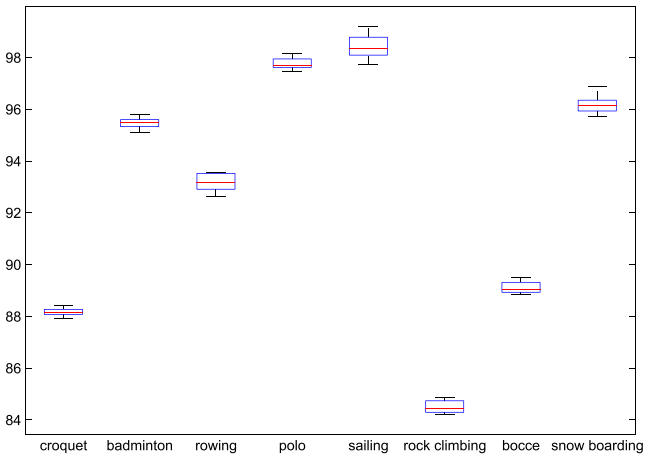


Fig. 3. Boxplot of the performance on the UIUC-Sports data set (%).

TABLE II
PERFORMANCE COMPARISONS ON THE FLOWER-17 DATA SET

Algorithms	Performance
ScSPM[6]	52.35
KMTJSRC-CG [12]	88.90 ± 2.30
Nilsback[46]	71.76 ± 1.76
CSDL[8]	72.65 ± 1.79
Varma [51]	82.55 ± 0.34
LP-B [52]	85.40 ± 2.40
GCC(SIFT)	72.14 ± 1.46
LR(SIFT)	73.17 ± 1.37
LR-GCC(SIFT)	75.63 ± 1.62
LR-GCC	$\mathbf{91.52 \pm 1.24}$

C. Flower-17 Data Set and Flower-102 Data Set

The Flower-17 data set has 17 classes of flower images as *Buttercup*, *Colts' foot*, *Daffodil*, *Daisy*, *Dandelion*, *Fritillary*, *Iris*, *Pansy*, *Sunflower*, *Windflower*, *Snowdrop*, *Lily valley*, *Bluebell*, *Crocus*, *Tigerlily*, *Tulip*, and *Cowslip*. There are 80 images in each class with a total of 1360 images. We use the same train/validate/test (40/20/20) image split as in [46] for fair comparison. Table II gives the performance comparison on the Flower-17 data set. We give the

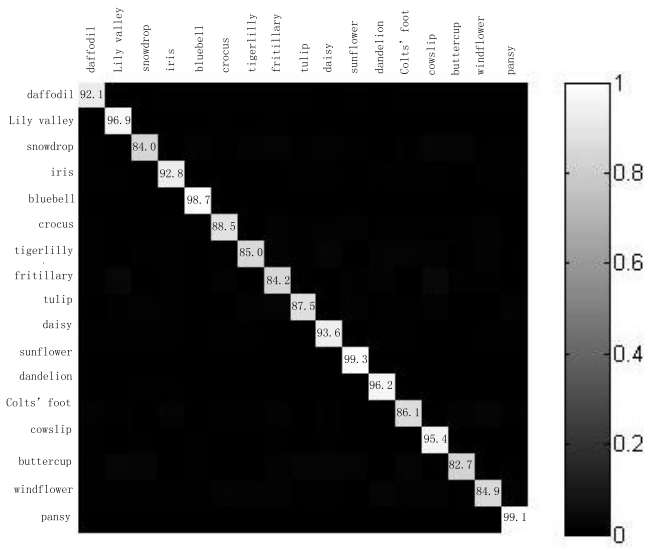


Fig. 4. Confusion matrix of the proposed LRSC-GCC method on the Flower-17 data set. The diagonal values indicate the per-class classification rate (%).

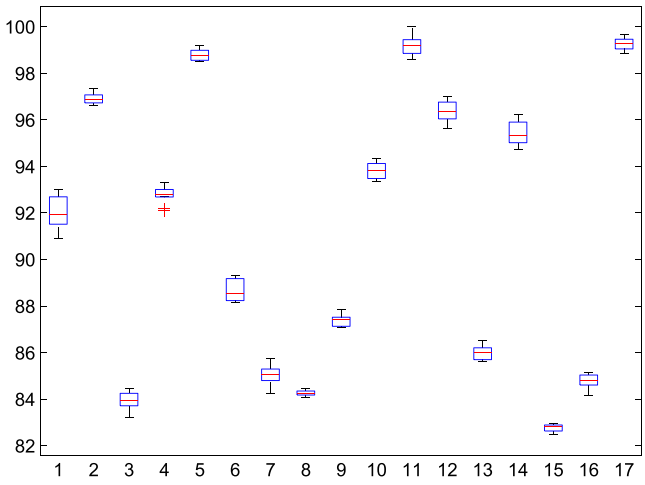


Fig. 5. Boxplot of the performance on the Flower-17 data set (%). The numbers from 1 to 17 in the horizontal row indicate *Daffodil*, *Lily valley*, *Snowdrop*, *Iris*, *Bluebell*, *Crocus*, *Tigerlily*, *Fritillary*, *Tulip*, *Daisy*, *Sunflower*, *Dandelion*, *Colts' foot*, *Cowslip*, *Buttercup*, *Windflower*, and *Pansy*, respectively.

performance of LRSC-GCC when only the SIFT feature is used [LRSC-GCC (SIFT)]. We also give the classification accuracy of LRSC and GCC with the SIFT feature only [LRSC (SIFT) and GCC (SIFT)]. For each random selection process, we calculate the confusion matrix on the corresponding testing images. The average of confusion matrixes is given in Fig. 4. We also give the boxplot in Fig. 5. The proposed method has different performances on varied classes. This is because for some classes (e.g., *Sunflower*), most the images are taken in the front view and occupy large areas of images and hence are relatively easier to model than the classes whose images are taken with multiviews. Besides, the variations of object's shapes of some classes (e.g., *Pansy*) are smaller than those of other classes (e.g., *Cowslip*). Moreover, images of different classes may have very similar visual appearances. For example, some *Daisy* flowers are very similar to the *Colt's foot* flowers. Hence, images of the two classes are often misclassified.

TABLE III
PERFORMANCE COMPARISONS ON THE FLOWER-102 DATA SET

Algorithms	Performance
KMTJSRC-CG[12]	74.1
Nilsback[47]	72.8
GCC	73.5
LR	74.2
LR-GCC	**75.7**

TABLE IV
PERFORMANCE COMPARISON ON THE CALTECH-256 DATA SET

Algorithm	15 training	30 training
KC[2]	–	27.17 ± 0.46
NBNN[3]	30.45	38.18
KSPM[6]	23.34 ± 0.42	29.51 ± 0.52
ScSPM[6]	27.73 ± 0.51	34.02 ± 0.35
LScSPM[11]	30.00 ± 0.14	35.74 ± 0.10
LRSC[15]	–	41.04 ± 0.23
LLC[25]	34.36	41.19
KSPM[48]	–	34.10
FV[53]	38.50 ± 0.20	47.40 ± 0.10
GCC	34.38 ± 0.60	39.44 ± 0.47
LR	35.16 ± 0.52	40.86 ± 0.39
LR-GCC	**39.21 ± 0.48**	**45.87 ± 0.41**
LR-GCC-FV	**41.39 ± 0.36**	**49.13 ± 0.32**

We can see from Table II that when only the SIFT feature is used, the proposed method is able to outperform category-specific dictionary learning, which also learns a number of codebooks by about 3%. This shows the usefulness of considering the local features jointly for encoding. As objects of the Flower-17 data set are visually similar, it is more urgent to jointly model the visual correlations among local features instead of treating them independently. Besides, since color information is useful for flower classification, the combination of various features can greatly improve the performance. Moreover, using the color information with LRSC-GCC, we are able to exceed the performances of many feature combination methods [12], [51], [52]. This again demonstrates the effectiveness of the proposed method.

The Flower-102 data set is an extended version of the Flower-17 data set with more types of flowers and a larger number of images. There are 8189 images of 102 classes, which have 40–250 images for each class. We follow the experimental setup as in [47] and use 10/10/rest for training/validation/testing, respectively. Table III gives the performance comparison on the Flower-102 data set. We can have similar conclusions as on the Flower-17 data set. The proposed LRSC-GCC is able to outperform KMTJSRC-CG, which combines different types of features by sparse reconstruction. Since images of the flower data sets are very similar, the train of discriminative codebooks and joint encoding of parameters help to improve the performance to some extent.

D. Caltech-256 Data Set

There are 256 classes of images with a total number of 29 780 pictures in the Caltech-256 data set. Each class has at least 80 images. We randomly select 15/30 training images per class for performance evaluation and use the rest

TABLE V
PERFORMANCE COMPARISONS ON THE STANFORD DOG DATA SET. ET: EDGE TEMPLATES

Algorithms	Performance
SIFT[54]	22.0
ET[55]	38.0
Symb[56]	44.1
Ali[57]	50.1
LR-GCC	44.3
LR-GCC-FV	49.7

images for testing. We repeat this process ten times for reliable comparison. Table IV gives the performance comparison on the Caltech-256 data set.

Compared with k-means clustering-based codebook generation, the use of sparse coding can preserve more information for classification, and hence LRSC-GCC is able to outperform the kernel codebook. Besides, by training discriminative classifiers, we are able to improve over the nearest-neighbor-based method [3]. Moreover, compared with the methods [6], [25] that treated local features independently, the joint encoding of local features can make use of the spatial and structure information of local features that eventually improve the classification accuracy. Finally, joint encoding local features with codebook generation can improve over only learning discriminative parameters [15]. Although different classes of images within the Caltech-256 data set are not as similar as the Flower-17 and the Flower-102 data sets, the proposed method is still able to encode discriminative information for image representation and classification.

The Fisher vector technique [53] can also be combined with the proposed LRSC-GCC method by fixing the learned codebooks and encoding the local features accordingly (LRSC-GCC-FV). By encoding high-order information, the FV-based method [53] can improve over simple assignment [2], [6] and sparse coding [6], [11], [15] based methods dramatically. Besides, by leveraging the proposed method with FV, we can further improve the classification accuracy.

E. Stanford Dog Data Set

The Stanford Dog data set [54] has 20 580 images of 120 classes. We follow the same experimental setup as in [54] and use 100 images per class for training. Using only the image label, we are able to achieve comparable performance as segmentation- and alignment-based methods [56], [57] in Table V. By generating general and class-specific codebooks with jointly encoding spatial nearby local features, we can outperform using SIFT feature [54] dramatically. Besides, using the Fisher vector technique, we can further improve the performance of the proposed LRSC-GCC method.

F. Convergency

We can gradually reduce the objective value of Problem 5 by alternatively optimizing over the codebooks $\hat{\boldsymbol{B}}$ and the encoding parameters $\hat{\boldsymbol{A}}$. When the encoding parameters $\hat{\boldsymbol{A}}$ are fixed, Problem 5 equals Problem 6, which can be optimized over the general codebook and class-specific codebooks iteratively. During each step, the objective value of Problem 6 is reduced. When the codebooks $\hat{\boldsymbol{B}}$ are fixed, Problem 5 equals Problem 9 whose objective value can be reduced with the IALM method. As we alternatively optimize over $\hat{\boldsymbol{B}}$ and $\hat{\boldsymbol{A}}$, we can reduce the objective value of Problem 5 gradually. Besides, the objective value of Problem 5 is always bigger than zero, and hence the proposed method is able to converge. We give the objective value changes with the number of iterations on the UIUC-Sports data set, the Flower-17 data set, the Flower-102 data set, and the Caltech-256 data set in Fig. 6 for intuitive illustration.

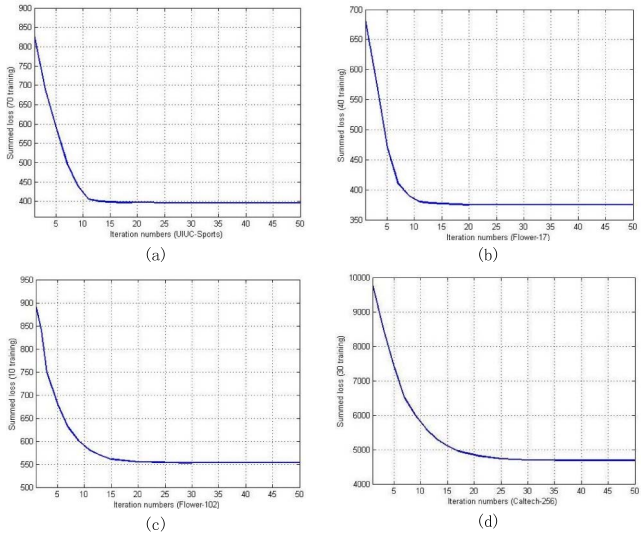


Fig. 6. Convergency of the proposed method on (a) UIUC-Sports data set, (b) Flower-17 data set, (c) Flower-102 data set, and (d) Caltech-256 data set.

V. CONCLUSION

In this paper, we proposed a fine-grained image classification method by combining the codebook generation with LRSC. Instead of generating single codebook, we constructed a general codebook and class-specific codebooks by joint optimization of the summed reconstruction error, the sparsity constraints of codebook discrepancy, and the codebook incoherences. Besides, by adapting the LRSC technique in the image region level, we can model the visual similarities among local features. Moreover, the image regions of the same position were combined for joint representation. In this way, we were able to get more representative and discriminative image representation for fine-grained classification. The classification results on several public image data sets proved the effectiveness of the proposed method.

REFERENCES

[1] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. 9th IEEE ICCV*, Oct. 2003, pp. 1470–1477.
[2] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J.-M. Geusebroek, "Visual word ambiguity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 7, pp. 1271–1283, Jul. 2010.
[3] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest-neighbor based image classification," in *Proc. IEEE Conf. CVPR*, Jun. 2008, pp. 1–8.
[4] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. CVPR*, Jun. 2006, pp. 2169–2178.

[5] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[6] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proc. IEEE Conf. CVPR*, Miami, FL, USA, Jun. 2009, pp. 1794–1801.

[7] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," in *Proc. IEEE Conf. CVPR*, Jun. 2010, pp. 3501–3508.

[8] S. Gao, I. W.-H. Tsang, and Y. Ma, "Learning category-specific dictionary and shared dictionary for fine-grained image categorization," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 623–634, Feb. 2014.

[9] C. Zhang, J. Liu, C. Liang, Q. Huang, and Q. Tian, "Image classification using Harr-like transformation of local features with coding residuals," *Signal Process.*, vol. 93, no. 8, pp. 2111–2118, 2013.

[10] Y. Jiang, J. Meng, J. Yuan, and J. Luo, "Randomized spatial context for object search," *IEEE Trans. Image Process.*, vol. 24, no. 6, pp. 1748–1762, Jun. 2015.

[11] S. Gao, I. W.-H. Tsang, and L.-T. Chia, "Laplacian sparse coding, hypergraph Laplacian sparse coding, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 92–104, Jan. 2013.

[12] X. T. Yuan and S. Yan, "Visual classification with multi-task joint sparse representation," in *Proc. IEEE Conf. CVPR*, Jun. 2010, pp. 3493–3500.

[13] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, "RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images," in *Proc. IEEE Conf. CVPR*, Jun. 2010, pp. 763–770.

[14] C. Zhang, J. Liu, C. Liang, Z. Xue, J. Pang, and Q. Huang, "Image classification by non-negative sparse coding, correlation constrained low-rank and sparse decomposition," *Comput. Vis. Image Understand.*, vol. 123, pp. 14–22, Jun. 2014.

[15] T. Zhang, B. Ghanem, S. Liu, C. Xu, and N. Ahuja, "Low-rank sparse coding for image classification," in *Proc. IEEE ICCV*, Dec. 2013, pp. 281–288.

[16] Y. Mu, J. Dong, X. Yuan, and S. Yan, "Accelerated low-rank visual recovery by random projection," in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 2609–2616.

[17] B. Yao, A. Khosla, and L. Fei-Fei, "Combining randomization and discrimination for fine-grained image categorization," in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 1577–1584.

[18] T. Berg and P. N. Belhumeur, "How do you tell a blackbird from a crow?" in *Proc. ICCV*, 2013, pp. 9–16.

[19] K. Winn, A. Criminisi, and T. Minka, "Object categorization by learned universal visual dictionary," in *Proc. 10th IEEE ICCV*, Oct. 2005, pp. 1800–1807.

[20] F. Moosmann, E. Nowak, and F. Jurie, "Randomized clustering forests for image classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 9, pp. 1632–1646, Sep. 2008.

[21] M. Yang, L. Van Gool, and L. Zhang, "Sparse variation dictionary learning for face recognition with a single training sample per person," in *Proc. ICCV*, 2013, pp. 689–696.

[22] Y. T. Chi, M. Ali, M. Rushdi, and J. Ho, "Affine-constrained group sparse coding and its application to image-based classifications," in *Proc. ICCV*, 2013, pp. 681–688.

[23] C.-K. Chiang, T.-F. Su, C. Yen, and S.-H. Lai, "Multi-attributed dictionary learning for sparse coding," in *Proc. ICCV*, 2013, pp. 1137–1144.

[24] H. Bristow, A. Eriksson, and S. Lucey, "Fast convolutional sparse coding," in *Proc. IEEE Conf. CVPR*, Jun. 2013, pp. 391–398.

[25] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. IEEE Conf. CVPR*, Jun. 2010, pp. 3360–3367.

[26] Y. L. Boureau, N. Le Roux, F. Bach, J. Ponce, and Y. LeCun, "Ask the locals: Multi-way local pooling for image recognition," in *Proc. ICCV*, 2011, pp. 2651–2658.

[27] A. Shabou and H. Le Borgne, "Locality-constrained and spatially regularized coding for scene categorization," in *Proc. IEEE Conf. CVPR*, Jun. 2012, pp. 3618–3625.

[28] S. Bulò and P. Kontschieder, "Neural decision forests for semantic image labelling," in *Proc. IEEE Conf. CVPR*, Jun. 2014, pp. 81–88.

[29] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *Proc. 10th IEEE ICCV*, Oct. 2005, pp. 1458–1465.

[30] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin, "Context-based vision system for place and object recognition," in *Proc. 9th IEEE ICCV*, Oct. 2003, pp. 273–280.

[31] Z. Wu, Q. Ke, M. Isard, and J. Sun, "Bundling features for large scale partial-duplicate Web image search," in *Proc. IEEE Conf. CVPR*, Jun. 2009, pp. 25–32.

[32] B. Ni, S. Yan, and A. Kassim, "Contextualizing histogram," in *Proc. IEEE Conf. CVPR*, Jun. 2009, pp. 1682–1689.

[33] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, Apr. 2002.

[34] C. Zhang, S. Wang, Q. Huang, J. Liu, C. Liang, and Q. Tian, "Image classification using spatial pyramid robust sparse coding," *Pattern Recognit. Lett.*, vol. 34, no. 9, pp. 1046–1052, 2013.

[35] J. Chen, J. Yang, L. Luo, J. Qian, and W. Xu, "Matrix variate distribution-induced sparse representation for robust image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 10, pp. 2291–2300, Oct. 2015, doi: 10.1109/TNNLS.2014.2377477.

[36] Y. Xiao, Z. Zhu, Y. Zhao, Y. Wei, and S. Wei, "Kernel reconstruction ICA for sparse representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1222–1232, Jun. 2015.

[37] C. Li, Q. Liu, J. Liu, and H. Lu, "Ordinal distance metric learning for image ranking," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 7, pp. 1551–1559, Jul. 2015.

[38] X. Bai, C. Liu, P. Ren, J. Zhou, H. Zhao, and Y. Su, "Object classification via feature fusion based marginalized kernels," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 1, pp. 8–12, Jan. 2015.

[39] H. Zhang, X. Bai, J. Zhou, J. Cheng, and H. Zhao, "Object detection via structural feature selection and shape model," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 4984–4995, Dec. 2013.

[40] A. Yu and K. Grauman, "Fine-grained visual comparisons with local learning," in *Proc. IEEE Conf. CVPR*, Jun. 2014, pp. 192–199.

[41] Q. Qian, R. Jin, S. Zhu, and Y. Lin, "Fine-grained visual categorization via multi-stage metric learning," in *Proc. IEEE Conf. CVPR*, Jun. 2015, pp. 3716–3724.

[42] J. Wang *et al.*, "Learning fine-grained image similarity with deep ranking," in *Proc. IEEE Conf. CVPR*, Jun. 2014, pp. 1386–1393.

[43] J. Krause, H. Jin, J. Yang, and L. Fei-Fei, "Fine-grained recognition without part annotations," in *Proc. IEEE Conf. CVPR*, Jun. 2015, pp. 5546–5555.

[44] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Proc. NIPS*, 2006, pp. 801–808.

[45] L. J. Li and L. Fei-Fei, "What, where and who? Classifying events by scene and object recognition," in *Proc. IEEE 11th ICCV*, Oct. 2007, pp. 1–8.

[46] M. E. Nilsback and A. Zisserman, "A visual vocabulary for flower classification," in *Proc. IEEE Conf. CVPR*, 2006, pp. 1447–1454.

[47] M. E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proc. 6th Indian Conf. Comput. Vis., Graph. Image Process.*, 2008, pp. 722–729.

[48] G. Griffin, A. Holub, and P. Perona, "The Caltech 256 dataset," Dept. Vis., California Inst. Technol., Pasadena, CA, USA, Tech. Rep. TR-2007-001, 2006.

[49] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1582–1596, Sep. 2010.

[50] J. Wu and J. M. Rehg, "Beyond the Euclidean distance: Creating effective visual codebooks using the histogram intersection kernel," in *Proc. IEEE 12th ICCV*, Sep./Oct. 2009, pp. 630–637.

[51] M. Varma and D. Ray, "Learning the discriminative power-invariance trade-off," in *Proc. IEEE 11th ICCV*, Oct. 2007, pp. 1–8.

[52] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in *Proc. IEEE 12th ICCV*, Sep./Oct. 2009, pp. 221–228.

[53] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the Fisher vector: Theory and practice," *Int. J. Comput. Vis.*, vol. 105, no. 3, pp. 222–245, Dec. 2013.

[54] A. Khosla, N. Jayadevaprakash, B. Yao, and F.-F. Li, "Novel dataset for fine-grained image categorization," in *Proc. CVPR 1st Workshop Fine-Grained Vis. Categorization*, 2011, pp. 1–2.

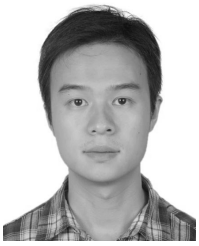
[55] S. Yang, L. Bo, J. Wang, and L. G. Shapiro, "Unsupervised template learning for fine-grained object recognition," in *Proc. NIPS*, 2012, pp. 3131–3139.

[56] Y. Chai, V. Lempitsky, and A. Zisserman, "Symbiotic segmentation and part localization for fine-grained categorization," in *Proc. IEEE ICCV*, Dec. 2013, pp. 321–328.

[57] E. Gavves, B. Fernando, C. G. M. Snoek, A. W. M. Smeulders, and T. Tuytelaars, "Fine-grained categorization by alignments," in *Proc. IEEE ICCV*, Dec. 2013, pp. 1713–1720.

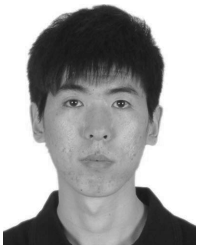
Chunjie Zhang received the B.E. degree from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2006, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2011.

He was an Engineer with the Henan Electric Power Research Institute, Zhengzhou, China, from 2011 to 2012. He held a post-doctoral position with the School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing, where he is currently an Assistant Professor. His current research interests include image processing, machine learning, pattern recognition, and computer vision.



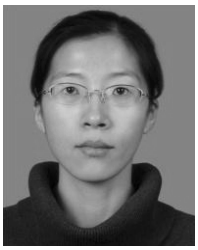
Chao Liang received the B.S. degree in automation from the Huazhong University of Science and Technology, Wuhan, China, in 2006, and the Ph.D. degree in pattern recongition and intelligent system from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2012.

He is currently an Assistant Professor with the National Engineering Research Center for Multimedia Software, School of Computer, Wuhan University, Wuhan. His current research interests include multimedia content analysis, machine learning, computer vision, and pattern recognition.



Liang Li received the B.S. degree in computer science from Xi'an Jiaotong University, Xi'an, China, in 2008, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2013.

He currently holds a post-doctoral position with the University of Chinese Academy of Sciences, Beijing. His current research interests include image processing, large-scale image retrieval, image semantic understanding, multimedia content analysis, computer vision, and pattern recognition.



Jing Liu received the B.E. and M.E. degrees from Shandong University, Jinan, China, in 2001 and 2004, respectively, and the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2008.

She is currently an Associate Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. Her current research interests include multimedia analysis, understanding, and retrieval.

Qingming Huang received the Ph.D. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1994.

He was a Post-Doctoral Fellow with the National University of Singapore, Singapore, from 1995 to 1996, and was with the Institute for Infocomm Research, Singapore, as a Research Staff Member from 1996 to 2002. He joined the Chinese Academy of Sciences, Beijing, China, in 2003. He is currently a Professor with the University of Chinese Academy of Sciences, Beijing. His current research interests include image and video analysis, video coding, pattern recognition, and computer vision.

Qi Tian received the B.E. degree from Tsinghua University, Beijing, China, in 1992, the M.S. degree from Drexel University, Philadelphia, PA, USA, in 1996, and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana–Champaign, Champaign, IL, USA, in 2002.

He is currently a Professor with the Department of Computer Science, The University of Texas at San Antonio, San Antonio, TX, USA, and an Adjunct Professor with Zhejiang University, Hangzhou, China, and Xidian University, Xi'an, China. His current research interests include multimedia information retrieval, computational systems biology, biometrics, and computer vision.