Orthogonal and Smooth Subspace Based on Sparse Coding for Image Classification

Fushuang Dai[1,2], Yao Zhao[1,2(✉)], Dongxia Chang[1,2], and Chunyu Lin[2]

[1] Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China
yzhao@bjtu.edu.cn
[2] Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing, China

Abstract. Many real-world problems usually deal with high-dimensional data, such as images, videos, text, web documents and so on. In fact, the classification algorithms used to process these high-dimensional data often suffer from the low accuracy and high computational complexity. Therefore, we propose a framework of transforming images from a high-dimensional image space to a low-dimensional target image space, based on learning an orthogonal smooth subspace for the SIFT sparse codes (SC-OSS). It is a two stage framework for subspace learning. Firstly, a sparse coding followed by spatial pyramid max pooling is used to get the image representation. Then, the image descriptor is mapped into an orthonormal and smooth subspace to classify images in low dimension. The proposed algorithm adds the orthogonality and a Laplacian smoothing penalty to constrain the projective function coefficient to be orthogonal and spatially smooth. The experimental results on the public datasets have shown that the proposed algorithm outperforms other subspace methods.

Keywords: Image classification · Orthogonal and smooth subspace · Sparse coding · Max pooling

1 Introduction

Image classification is one of the most fundamental problems in computer vision and pattern recognition. The primary objective of image classification is to assign one or more category labels to an image. It has been applied in a variety of fields such as video surveillance, image and video retrieval, web content analysis and so on. In image classification, image representation plays a very important role. In the past, many representation methods based on color, texture, shape, etc. were proposed. For the images of single color, single texture and single shape, the representations use these simple features are very excellent for image classification. Moreover, they have a small amount of computation and low computational complexity. However, in real life, a large number of images contain various and colorful content, in this case, the above-mentioned representations are far not enough to do image classification. In recent years, the Scale Invariant Feature Transform (SIFT) [1] descriptor has been

Y.-S. Ho et al. (Eds.): PCM 2015, Part II, LNCS 9315, pp. 41–50, 2015.
DOI: 10.1007/978-3-319-24078-7_5

widely used. SIFT descriptor is the local feature of the image, it keeps invariant to rotate, scale zoom, brightness variation, and a certain stability to the change of the viewing angle, affine transformations and the noise.

However, due to SIFT descriptor' high dimension, it leads to the "curse of dimensionality" [2]. So, the work of pre-processing for SIFT descriptor is quite important. In recent years, the bag of visual words (BoW) model [3] is widely applied for image analysis. But, there are two main recognized drawbacks [4] existing in BoW model: (1) In dictionary, the approach selects randomly visual words, which severely limits the descriptive power of image representation. (2) The model loses the spatial information of raw image. To overcome the first problem, sparse coding succinctly represents data vectors with several basis vectors in the dictionary, which requires only a relatively small number of bases to represent the signal. There are two basic requirements in sparse coding. One is that sparse coding feature is similar to the original as possible. Another is that coding coefficient is sparse. This shows that sparse coding not only can well retain the original features, but also greatly simplifies the complexity of the subsequent calculation. By overcoming the second problem of the BoW model, a spatial pyramid method is proposed [4, 5], which extended the BoW model by partitioning the image into sub-regions and computing histograms of local features.

From a space perspective, image classification can be considered as a classifier design problem. In this circumstance, an image of $n \times m$ pixels can be represented by a $n \times m$-dimensional vector. Then, the researchers focus on designing a classifier which can classify images in the $n \times m$-dimensional space effectively. In fact, the linear low-dimensional subspace [6] can significantly improve the performance of image classification. Principle Component Analysis (PCA) [7] and Linear Discriminate Analysis (LDA) [8] are very prominent subspace methods and they have got promising accuracies in image classification. However, they failed to consider the specific structures of images and cannot fully explore the spatial information. Hence, the orthogonal smooth subspace [9] is proposed. However, the algorithm uses the color features to classify. For single colorful images with very obvious target, the algorithm gets good classification performance. But, for various colorful target images e.g. in each class, the colors of images are various, and there is interference of other objects in the background of the target image, such as these images in Caltech101, this algorithm will lose its superior classification performance.

Based on the above-mentioned analysis, considering the low accuracy and high computational complexity of high-dimensional data in the classification algorithms, an orthogonal smooth subspace for the SIFT sparse codes (SC-OSS) was proposed. In the new algorithm, we transform images from a high-dimensional image space to a low-dimensional space. In the new algorithm, sparse coding followed by spatial pyramid max pooling is used to get the image representation. Then, the obtained representation is mapped into an orthonormal and smooth subspace in low dimension. Experiments will show that our approach get higher classification accuracy on the public datasets.

The rest of the paper is organized as follows. In Sect. 2, we describe and analyze sparse codes of SIFT descriptor using spatial pyramid max pooling. Section 3 presents the orthogonal smooth subspace learning model. Section 4 explains the parameters setting. Section 5 shows some experimental results, and Sect. 6 concludes the paper.

2 SIFT Sparse Codes Using Spatial Pyramid Max Pooling

2.1 The Sparse Codes

Let $\mathbf{X}$ be a set of SIFT appearance descriptors, $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M]^T \in \mathbf{R}^{M \times D}$. The sparse coding method mainly solves the following problem

$$\min_{\mathbf{u},\mathbf{v}} \sum_{m=1}^{M} \|\mathbf{x}_m - \mathbf{u}_m \mathbf{V}\|^2 + \lambda |\mathbf{u}_m| \tag{1}$$

where $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_k]^T$ is the codebook, K is the size of the codebook. Normally, the codebook $\mathbf{V}$ is an overcomplete basis set, i.e.,$K > D$. $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_M]^T$ is the coding matrix, $\mathbf{u}_m$ is constrainedL_1-norm regularization, i.e., after the optimization, the only nonzero element in $\mathbf{u}_m$ denotes the vector $\mathbf{x}_m$.

The problem (1) is not convex with respect to both variables $\mathbf{U}$ and $\mathbf{V}$. Thus, we cannot directly get the global optima for the problem. So, the way is to solve (1) iteratively by alternatively optimizing over $\mathbf{V}$ or $\mathbf{U}$ while fixing the other. Fixing $\mathbf{V}$, we can solve the following optimization by optimizing over each coefficient $\mathbf{u}_m$ individually

$$\min_{\mathbf{u}_m} \|\mathbf{x}_m - \mathbf{u}_m \mathbf{V}\|^2 + \lambda |\mathbf{u}_m| \tag{2}$$

The optimization (2) is well known as Lasso in the Statistics, and it is essentially a linear regression problem with L_1-norm regularization on the coefficients. Fixing $\mathbf{U}$, we can solve the optimization (3) by the Lagrange dual as used in [10]. The optimization (3) is a least square problem with quadratic constraints, where typically applied a unit L_2-norm constraint on $\mathbf{v}_k$ to avoid trivial solutions.

$$\begin{aligned} &\min_{\mathbf{V}} \|\mathbf{X} - \mathbf{U}\mathbf{V}\|_F^2 \\ &s.t. \|\mathbf{v}_k\| \le 1, \forall k = 1, 2, \ldots K \end{aligned} \tag{3}$$

2.2 Spatial Pyramid Max Pooling

In spatial pyramid model, the feature will be partitioned into $2^l \times 2^l$ blocks in different scales. Here, we construct two levels spatial pyramid, i.e., $l = 0, 1$. From sparse coding, we already get the coefficient $\mathbf{U}$. Then, in each level of spatial pyramid model, the max pooling function [13] is used to compute the image features:

$$r_j = \max\{|u_{1j}|, |u_{2j}|, \ldots, |u_{Mj}|\} \quad (4)$$

where r_j is the j-th element of the image feature, u_{ij} is the element of j-th column and i-th row of the coding matrix $\mathbf{U}$, and M is the number of local descriptors in this region.

3 Orthogonal and Smooth Subspace Method

From the above framework of sparse coding followed by spatial pyramid max pooling, we get a 5 × K-dimensional feature space. And it is a high dimensional space. The performance of image classification can be improved significantly in a linear low-dimensional subspace. So we propose an orthogonal smooth subspace for the SIFT sparse codes (SC-OSS). A more detailed description of the proposed algorithm is shown in Fig. 1. Firstly, we extract SIFT feature to get the local descriptor. Then, sparse coding and max pooling is applied to get the high dimensional description. At last, the orthonormal and smooth subspace method is proposed to obtain the low dimensional description of the image.

In the following, we will project each image feature into a low dimensional subspace by $\mathbf{y}_i = \mathbf{W}^T\mathbf{x}_i$ after getting the 5 × K-dimensional vector. $\mathbf{W} \in \mathbf{R}^{(n_1 \times n_2) \times d}$ is the transformation matrix and d is the dimensionality of the subspace.

A popular image classification aim is to find a set of centers for which the within-cluster spread is small; the between-clustering spread is large in some sense. Therefore, the optimization objection used of our paper is:

$$\max \ \mathrm{trace}\left(\mathbf{S}_w^{-1}\mathbf{S}_b\right) \quad (5)$$

where $\mathbf{S}_w$ is the within-class scatter matrix and $\mathbf{S}_b$ is the between-class scatter matrix. $\mathbf{S}_w$ measures how compact or tight the classes are and it is defined as

$$\mathbf{S}_w = \sum_{j=1}^{C}\sum_{i=1}^{q_j}\left(\mathbf{y}_i - \mathbf{m}_j\right)\left(\mathbf{y}_i - \mathbf{m}_j\right)^{\mathrm{T}} \quad (6)$$

where $\mathbf{m}_j = \frac{1}{q_j}\sum_{i=1}^{q_j}\mathbf{y}_i$ (j = 1, 2, …, C), q_j is the number of samples in class j. $\mathbf{S}_b$ measures how scatter the class centers are from the sample mean and it is given by

$$\mathbf{S}_b = \sum_{j=1}^{C} q_j\left(\mathbf{m}_j - \mathbf{m}\right)\left(\mathbf{m}_j - \mathbf{m}\right)^{\mathrm{T}} \quad (7)$$

where $\mathbf{m} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{y}_i$. By making the substitution $\mathbf{y}_i = \mathbf{W}^{\mathrm{T}}\mathbf{x}_i$ into (6) and (7), we can get

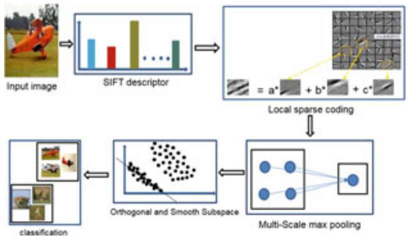

Fig. 1. The flow-chart of the proposed SC-OSS algorithm

$$\mathbf{S}_w = \sum_{j=1}^{C}\sum_{i=1}^{q_j}\left(\mathbf{W}^{\mathrm{T}}\mathbf{x}_i - \mathbf{W}^{\mathrm{T}}\mathbf{m}_j\right)\left(\mathbf{W}^{\mathrm{T}}\mathbf{x}_i - \mathbf{W}^{\mathrm{T}}\mathbf{m}_j\right)^{\mathrm{T}} = \mathbf{W}^{\mathrm{T}}\sum_{j=1}^{C}\sum_{i=1}^{q_j}\left(\mathbf{x}_i - \mathbf{m}_j\right)\left(\mathbf{x}_i - \mathbf{m}_j\right)^{\mathrm{T}}\mathbf{W} = \mathbf{W}^{\mathrm{T}}\mathbf{S}_{wx}\mathbf{W} \quad (8)$$

$$\begin{aligned}\mathbf{S}_b &= \sum_{j=1}^{C} q_j\left(\mathbf{W}^{\mathrm{T}}\mathbf{m}_j - \mathbf{W}^{\mathrm{T}}\mathbf{m}\right)\left(\mathbf{W}^{\mathrm{T}}\mathbf{m}_j - \mathbf{W}^{\mathrm{T}}\mathbf{m}_j\right)^{\mathrm{T}} \\ &= \mathbf{W}^{\mathrm{T}}\sum_{j=1}^{C} q_j\left(\mathbf{m}_{jx} - \mathbf{m}_x\right)\left(\mathbf{m}_{jx} - \mathbf{m}_x\right)^{\mathrm{T}}\mathbf{W} = \mathbf{W}^{\mathrm{T}}\mathbf{S}_{bx}\mathbf{W}\end{aligned} \quad (9)$$

Then we can rewrite (5) as

$$\mathrm{maxtrace}\left(\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_{wx}\mathbf{W}\right)^{-1}\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_{bx}\mathbf{W}\right)\right) \quad (10)$$

where $\mathbf{S}_{wx}$ and $\mathbf{S}_{bx}$ are the within-class scatter matrix and between-class scatter matrix in the original image feature space. Here, we add the orthogonal constraints, i.e., $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$. It guarantees that the embedding coefficients are invariant only when one multiples the transformation matrix by an orthogonal matrix. And then, we use the 2-D discretized Laplacian smoothing term, which has been successfully used in [11]. It is a $n_1 n_2 \times n_1 n_2$ matrix:

$$\boldsymbol{\Delta} = \mathbf{D}_1 \otimes \mathbf{I}_2 + \mathbf{I}_1 \otimes \mathbf{D}_2 \quad (11)$$

where $\mathbf{I}_j$ is the $n_j \times n_j$ identity matrix for $j = 1, 2$. $\otimes$ is the Kronecker product. $\mathbf{D}_j$ is

$$\mathbf{D}_j = \frac{1}{h_j^2}\begin{pmatrix} -1 & 1 & & & 0 \\ 1 & -2 & 1 & & \\ & \cdots & \cdots & \cdots & \\ & & 1 & -2 & 1 \\ 0 & & & 1 & -1 \end{pmatrix} \quad (12)$$

where $h_j = \frac{1}{n_j}$, for$j$ = 1, 2.

Therefore, the objective function can be written as:

$$\arg \max_{\mathbf{W}^{\mathrm{T}}\mathbf{W}=\mathrm{I}} \operatorname{trace}\left(\left(\mathbf{W}^{\mathrm{T}}\left(\mathbf{S}_{wx}+\lambda\mathbf{\Delta}^{\mathrm{T}}\mathbf{\Delta}\right)\mathbf{W}\right)^{-1}\times\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_{bx}\mathbf{W}\right)\right) \tag{13}$$

Where λ is the balance parameter and **I** is an identity matrix. Use the Lagrangian method to solve the following problem.

$$\Phi(\mathbf{W}) = \operatorname{trace}\left(\left(\mathbf{W}^{\mathrm{T}}\left(\mathbf{S}_{wx}+\lambda\mathbf{\Delta}^{\mathrm{T}}\mathbf{\Delta}\right)\mathbf{W}\right)^{-1}\times\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_{bx}\mathbf{W}\right)\right) - \operatorname{trace}\left(\varGamma\left(\mathbf{W}^{\mathrm{T}}\mathbf{W}-\mathrm{I}\right)\right) \tag{14}$$

Let $\mathbf{A} = \mathbf{S}_{wx} + \lambda\mathbf{\Delta}^{\mathrm{T}}\mathbf{\Delta}$, $\mathbf{B} = \mathbf{S}_{bx}$. Differentiating $\Phi(\mathbf{W})$ with respect to **W** and set it to be zero. Then, we have

$$\left(\mathbf{W}^{\mathrm{T}}\mathbf{A}\mathbf{W}\right)^{-1}\left(\mathbf{A}\mathbf{W}\left(\mathbf{W}^{\mathrm{T}}\mathbf{A}\mathbf{W}\right)^{-1}\mathbf{W}^{\mathrm{T}}\mathbf{B}\mathbf{W}-\mathbf{B}\mathbf{W}\right) - \mathbf{W}\varGamma = \mathbf{0} \tag{15}$$

Multiple both sides of Eq. (15) by $\mathbf{W}^{\mathrm{T}}$, we can get $\mathbf{W}^{\mathrm{T}}\mathbf{W}\varGamma=\mathbf{0}$. Note that there always exists $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$. Therefore, we can deduce $\mathbf{\Gamma}=\mathbf{0}$. Hence,

$$\mathbf{A}\mathbf{W}\left(\mathbf{W}^{\mathrm{T}}\mathbf{A}\mathbf{W}\right)^{-1}\mathbf{W}^{\mathrm{T}}\mathbf{B}\mathbf{W} - \mathbf{B}\mathbf{W} = \mathbf{0} \tag{16}$$

Let $\mathbf{D}=\mathbf{W}^{\mathrm{T}}\mathbf{A}\mathbf{W}$, $\mathbf{E}=\mathbf{W}^{\mathrm{T}}\mathbf{B}\mathbf{W}$. Substituting **D** and **E** into (16), we can get

$$\mathbf{B}\mathbf{W} = \mathbf{A}\mathbf{W}\mathbf{D}^{-1}\mathbf{E} \tag{17}$$

Since **D** and **E** are symmetric, there always exists a nonsingular matrix **P**, satisfying $\mathbf{P}^{\mathrm{T}}\mathbf{D}\mathbf{P} = \mathbf{I}$ and $\mathbf{P}^{\mathrm{T}}\mathbf{E}\mathbf{P} = \mathbf{\Lambda}$, where $\mathbf{\Lambda}$ is a diagonal matrix. Hence, $\mathbf{D} = \left(\mathbf{P}^{\mathrm{T}}\right)^{-1}\mathbf{P}^{-1}$, $\mathbf{E} = \left(\mathbf{P}^{\mathrm{T}}\right)^{-1}\varLambda\mathbf{P}^{-1}$. By making the substitution **D** and **E** into (17), we can get

$$\mathbf{B}\mathbf{W}\mathbf{P} = \mathbf{A}\mathbf{W}\mathbf{P}\varLambda \tag{18}$$

Let $\mathbf{V} = \mathbf{W}\mathbf{P}$, then $\mathbf{W} = \mathbf{V}\mathbf{P}^{-1}$. Therefore, **V** is the generalized Eigen matrix of the matrix pairs **B** and **A**. Substituting $\mathbf{W} = \mathbf{V}\mathbf{P}^{-1}$ into $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$, we have $\left(\mathbf{P}^{-1}\right)^{\mathrm{T}}\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{P}^{-1} = \mathrm{I}$. Perform the following eigenvalue decomposition of the matrix $\mathbf{V}^{\mathrm{T}}\mathbf{V}$: $\mathbf{V}^{\mathrm{T}}\mathbf{V} = \mathbf{U}\Sigma\mathbf{U}^{\mathrm{T}}$, then

$$\left(\mathbf{P}^{-1}\right)^{\mathrm{T}}\mathbf{U}\Sigma\mathbf{U}^{\mathrm{T}}\mathbf{P}^{-1} = \mathrm{I} \tag{19}$$

Thus, we can get $\mathbf{P}^{-1} = \mathbf{U}\Sigma^{-1/2}$. Therefore, we can get the transformation matrix

$$\mathbf{W} = \mathbf{V}\mathbf{P}^{-1} = \mathbf{V}\mathbf{U}\Sigma^{-1/2} \tag{20}$$

After getting the low dimensional feature $\mathbf{y}_i$, the Nearest Neighbor is used to classify images.

4 Parameters Selecting

For the orthogonal smooth subspace, we get the balance parameter λ by another parameter α, which is irrelevant to numerical scale.

$$\lambda = \alpha \max(\operatorname{diag}(\mathbf{S}_{wx})) \tag{21}$$

Here, we determine $\alpha = 0.001$ by cross validation based on previous work [9].

Another parameter is the dimensionality: d. The optimal dimensionality of subspace is equal to $\operatorname{rank}(\mathbf{S}_{bx})$. In real applications, we know that $\operatorname{rank}(\mathbf{S}_{bx}) = C - 1$. Thus, we also empirically choose $d = C - 1$ in the following experiments.

5 Experiments

For the purpose of testing the performance of our SC-OSS algorithm, experiments are conducted on the Caltech 101 data set (only the twenty maximum data sets are used), the Coil20 data set and the Flickr 13 animal images data set selected from a subset of NUS-WIDE data sets. In order to verify the classification performance of the SC-OSS, we compare it with LDA [8], OLDA [12], OSSL [9], KSPM [4], ScSPM [5] and LLC [14].

Firstly, we compare the performance of the algorithms with different number of training images and the experiments are repeated for 10 times. The average classification accuracy is shown in Tables 1, 2 and 3.

From Table 1, we observe that SC-OSS performs best among all the methods. It outperforms OSSL by more than 20 % and even outperforms LDA and OLDA by a large margin. And, it also achieves higher accuracy than ScSPM, KSPM and LLC. From Tables 2 and 3, the SC-OSS algorithm again achieves much better performance

Table 1. Classification rate (%) comparison on twenty maximum datasets of Caltech101 dataset (Mean ± Std-Dev%).

Algorithms	***15training***	**30training**	**45training**	**60training**
LDA	20.5 ± 1.8	26.2 ± 4.1	32.8 ± 2.4	32.8 ± 2.0
OLDA	27.6 ± 3.8	34.7 ± 3.0	36.0 ± 1.2	39.2 ± 4.4
OSSL	43.0 ± 1.9	52.2 ± 1.5	57.1 ± 0.7	61.1 ± 1.0
KSPM	68.3 ± 1.5	76.4 ± 2.1	80.5 ± 1.0	81.5 ± 1.8
ScSPM	**76.3 ± 1.5**	80.0 ± 0.7	82.4 ± 0.6	83.9 ± 0.8
LLC	68.9 ± 1.8	74.8 ± 0.9	77.8 ± 0.8	80.3 ± 0.8
SC-OSS	73.1 ± 1.5	**81.8 ± 1.0**	**87.2 ± 0.8**	**90.2 ± 0.7**

Table 2. Classification rate (%) comparison on Coil20 dataset (Mean ± Std-Dev%).

Algorithms	***5training***	**10training**
LDA	43.4 ± 8.0	45.4 ± 11.9
OLDA	55.7 ± 6.0	60.3 ± 4.1
OSSL	84.5 ± 2.0	89.7 ± 0.8
KSPM	76.2 ± 2.0	80.5 ± 0.9
ScSPM	89.2 ± 1.4	94.2 ± 1.6
LLC	88.8 ± 0.6	93.8 ± 0.8
SC-OSS	**89.4 ± 0.8**	**96.9 ± 0.8**

Table 3. Classification rate (%) comparison on Flickr 13 animal images dataset (Mean ± Std-Dev%).

Algorithms	***50training***	**75training**	**100training**
LDA	9.7 ± 0.5	10.2 ± 0.6	9.1 ± 0.5
OLDA	10.1 ± 0.8	10.6 ± 0.7	9.2 ± 0.5
OSSL	10.8 ± 0.7	10.4 ± 0.9	11.2 ± 0.6
KSPM	26.9 ± 0.8	28.8 ± 0.6	28.7 ± 0.6
ScSPM	34.5 ± 0.8	36.8 ± 1.1	37.4 ± 0.8
LLC	**34.8 ± 1.0**	36.8 ± 1.2	38.0 ± 1.4
SC-OSS	30.9 ± 0.8	**37.2 ± 0.8**	**40.7 ± 1.2**

than listing algorithms. Meanwhile, from experimental standard derivation, we generally find that our method standard derivations are relatively smaller than another algorithms, it means that our method is more stable to training samples.

In the following, we compare the max pooling with other pooling methods, namely, the mean of absolute values (Abs) and the square root of mean squared statistics (Sqrt). The other two pooling functions are defined as

$$\text{Abs}: \quad z_j = \frac{1}{M}\sum_{i=1}^{M} |u_{ij}| \tag{22}$$

$$\text{Sqrt}: \quad z_j = \sqrt{\frac{1}{M}\sum_{i=1}^{M} u_{ij}^2} \tag{23}$$

As shown in Table 4, the performance of max pooling is the best, probably due to its robustness to local spatial variations. We also investigate the effects of codebook

Table 4. The performance comparison using different pooling methods on ten maximum datasets of Caltech 101 dataset (Mean ± Std-Dev%).

Pooling	Max	Abs	Sqrt
Caltech101	**88.7 ± 0.8**	78.9 ± 1.3	84.3 ± 1.2

sizes. Here, we design three sizes: 256, 512 and 1024. As shown in Table 5, we find that the performance for our method keeps increasing when the codebook size goes up to 1024.

Table 5. The effects of codebook size on our method on ten maximum datasets of Caltech 101 dataset (Mean ± Std-Dev%).

Codebook size	***256***	**512**	**1024**
15training	80.8 ± 1.2	81.1 ± 1.5	**81.5 ± 1.1**
30training	84.5 ± 1.3	87.1 ± 1.5	**88.7 ± 0.8**
45training	85.9 ± 1.0	88.5 ± 0.6	**90.1 ± 0.8**
60training	88.1 ± 0.7	89.7 ± 0.8	**91.6 ± 0.5**

6 Conclusion

In this paper, we propose a framework of transforming images from a high-dimensional image space to a low-dimensional target image space, based on learning an orthogonal smooth subspace for the SIFT sparse codes (SC-OSS). By sparse coding followed by spatial pyramid max pooling, we get the excellent image representation. Then, the high-dimensional representation is mapped into a low dimension subspace to classify images. The low dimension subspace adds the orthogonality and a Laplacian smoothing penalty to constrain the projective function coefficient to be orthogonal and spatially smooth. So, comparing with other methods, our method performs better on the public datasets. Moreover, in terms of the sparse coding, it also has much value to be worth studying. Further research of this proposed method is a valuable direction.

Acknowledgement. This paper was supported in part by National Natural Science Foundation of China (61210006, 61100141), Program for Changjiang Scholars and Innovative Research Team in University (IRT201206), the Fundamental Research Funds for the Central Universities of China (2013JBM021).

References

1. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
2. Bai, E.: Big data: The curse of dimensionality in modeling. In: Control Conference, pp.6–13. IEEE press, Chinese (2014)
3. Lou, X., Huang, D., Fan, L., et al.: An image classification algorithm based on bag of visual words and multi-kernel learning. J. Multimedia **9**(2), 269–277 (2014)
4. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recogn. **2**, 2169–2178 (2006)

5. Yang, J., Yu, K., Gong, Y., et al.: Linear spatial pyramid matching using sparse coding for image classification. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1794–1801 (2009)
6. Yan, Y., Zhang, Y.: Discriminant projection embedding for face and palmprint recognition. Neurocomputing **17**, 3534–3543 (2008)
7. Zhou, J., Jin, Z., Yang, J.: Multiscale saliency detection using principle component analysis. In: The 2012 International Joint Conference on Neural Networks (IJCNN), pp. 1–6. IEEE press, Brisbane (2012)
8. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. Wiley-Inter science, Hoboken, NJ (2000)
9. Hou, C., Nie, F., Zhang, C., et al.: Learning an orthogonal and smooth subspace for image classification. Signal Process. Lett. **16**(4), 303–306 (2009). IEEE press
10. Lee, H., Battle, A., Raina, R., et al.: Efficient sparse coding algorithms. In: Advances in neural information processing systems, pp.801–808 (2006)
11. Liu, F., Liu, X.: Locality enhanced spectral embedding and spatially smooth spectral regression for face recognition. In: Information and Automation, pp.299–303. IEEE press, Shenyang (2012)
12. Ye, J.: Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems. J. Mach. Learn. Res. **6**, 483–502 (2005)
13. Serre, T., Wolf, L., Poggio, T.: Object recognition with features inspired by visual cortex. Comput. Vis. Pattern Recogn. **2**, 994–1000 (2005)
14. Wang, J., Yang, J., Yu, K., et al.: Locality-constrained linear coding for image classification. In: IEEE Conference on Computer Vision and Pattern Recognition, pp.3360–3367 (2010)