Robust Scene Categorization via Scale-Rotation Invariant Generative Model and Kernel Sparse Representation Classification

**Jinjun KUANG††, Member and Yi CHAI††, Nonmember**

**SUMMARY** This paper presents a novel scale-rotation invariant generative model (SRIGM) and a kernel sparse representation classification (KSRC) method for scene categorization. Recently the sparse representation classification (SRC) methods have been highly successful in a number of image processing tasks. Despite its popularity, the SRC framework lacks the abilities to handle multi-class data with high inter-class similarity or high intra-class variation. The kernel random coordinate descent (KRCD) algorithm is proposed for \( \ell_1 \) minimization in the kernel space under the KSRC framework. It allows the proposed method to obtain satisfactory classification accuracy when inter-class similarity is high. The training samples are partitioned in multiple scales and rotated in different resolutions to create a generative model that is invariant to scale and rotation changes. This model enables the KSRC framework to overcome the high intra-class variation problem for scene categorization. The experimental results show the proposed method obtains more stable performances than other existing state-of-art scene categorization methods.

**key words**: kernel sparse representation, scale-rotation invariant generative model, scene categorization

1. Introduction

Automatic scene categorization is an important task in computer vision research, which has attracted a lot of interests in recent years. The scene images are generally consist of a number of different visual components such as sky, water, house and tree. These visual components are usually randomly organized in scene images and cannot be predicted easily. Therefore, the high intra-class variation is one of most difficult problems in scene categorization. As a result, designing an over-completed model to represent each scene image is crucial. On the other hand, it is common that different scene categories share similar visual components. For example, trees are found in many scene categories. Hence, finding a proper method to handle this inter-class similarity problem is a key to scene categorization.

In order to solve those problems, Yang et al. [1] proposed a scene classification framework based on sparse coding and spatial pyramid matching (ScSPM). This method obtained satisfactory results because it has created a robust generative model by using the powerful SIFT features and the spatial pyramid matching technique. However, since the classification accuracy of this framework relies on the linear SVM, its classification rate may be reduced by the nonrobust nature of the SVM classifiers if scene images are corrupted by illumination variation, occlusion and random noise.

In the field of image classification, the sparse representation classification (SRC) has recently drawn a lot of attentions. The work of Wright et al. [2] and Mei et al. [3] based on sparse representation classification obtained impressive results against illumination variations, occlusions, and random noise on face recognition and vehicle recognition datasets. However, their method cannot be directly applied to the tasks with high intra-class variation such as scene categorization. In order to sparsely represent a scene image under the SRC framework, the training set must contain all possible combinations of all visual components. In realistic applications, the number of the training samples may be insufficient.

In order to solve this “intra-class variation” problem, the kernel trick could be used to map the linear inseparable features into high dimensional feature space, in which features of the same class are more easily grouped together and then linear separable. Gao et al. [4] proposed a KSRC method which is based on kernel feature-sign search (KFSS) and SPM. Experimental results show KSRC has outperformed ScSPM.

The details of the proposed method are described as follows:

1. A kernel random coordinate descent (KRCD) algorithm is designed for solving the convex optimization problem under the kernel sparse representation classification (KSRC) framework. The KRCD algorithm is more capable of handling the high inter-class similarity problem than standard SRC methods. The performances of different Mercer kernel functions are analyzed as well.

2. In order to solve the intra-class variation problem and adapt the KSRC method to scene categorization tasks, a scale-rotation invariant generative model (SRIGM) is proposed. In addition, a simple dimension reduction method is proposed to further enhance the efficiency of the proposed method.

3. Finally, a novel scene categorization framework based on the KSRC and the SRIGM is proposed, resulting in satisfactory performance with only a few training samples.

At last, several case studies are presented, and experimental results show the proposed method obtained performances superior to three existing state-of-art scene catego-
zation methods.

This paper is organized as follows: Section 2 describes the SRIGM. Section 3 explains the KSRC in detail. Section 4 presents the experimental setup and test results. Section 5 concludes this paper.

2. Formulation of the Scale-Rotation Invariant Generative Model

In this section, we explain how to create an SRIGM that is capable of overcoming the high intra-class variation nature of scene images.

The generative training samples (atom vectors) used to construct the over-completed dictionary is created from a small number of raw training samples by a rotating and multiscale-partitioning process. The over-completed dictionary created by this process could be called a generative model.

Consider $k$ classes of training images and the resolution of each image is $w \times h$. The atom vector $v$ is transformed from each training image so that $v \in \mathbb{R}^d$, $d = w \times h$. Given the dictionary of $l$th class $D_l = [v_{1l}, \ldots, v_{nl}] \in \mathbb{R}^{dzl}$, then the over-completed dictionary is defined as $D = [D_1, \ldots, D_l, \ldots, D_k] \in \mathbb{R}^{dzk}$, $z = n \times k$.

The first step is to get the local blocks $b_j$ of the training images by segmenting the training images at different scales. Each image can be cut into $4^l$ blocks at scale $l$, $l = \{0, 1, 2, \ldots \}$. The second step is to get the extended blocks $b_{j1}, b_{j2}, \ldots, b_{jn}$ by rotating each block $b_j$ at scale $l$. Then, new training data are acquired: $A_l = [b_{j1}, b_{j2}, \ldots, b_{jn}, b'_{j1}, b'_{j2}, \ldots, b'_{jn}, b''_{j1}, b''_{j2}, \ldots, b''_{jn}, b'''_{j1}, b'''_{j2}, \ldots, b'''_{jn}]$ and $A' = [A^0, A^1, \ldots, A^M]$.

This process is called the $M \times s$-fold rotation-expansion. Then, the $k$th class of dictionary is $D_k = f(A'_k)$, where $D_k = [D'_k, D''_k, \ldots, D'''_k]$ and $v = f(b)$, where function $f(\cdot)$ transforms the 2D image to 1D vector. Finally, the over-completed dictionary is created.

In order to reduce the computational cost of the convex optimization process, the dimension of the dictionary $D$ has to be reduced. The basic idea of dimension reduction is to abandon those blocks that are similar to each other. Given a threshold $\delta$, and the Euclidean distance between two blocks is defined as $\kappa$. If $\kappa < \delta$ then abandon one of the blocks.

After the dimension reduction process, one can expect to improve the classification. When the threshold $\delta$ is set to 0.001, and the reconstruction errors remain at the same level. That suggests this method is capable of reducing the dimension of the dictionary $D$, meanwhile, leaving the completeness of the dictionary intact.

3. Kernel Sparse Representation Classification

The basic principle of sparse representation framework can be formulated as follows. The test vector $y$ could be represented by the dictionary $D$ and the sparse coefficients vector $C = [c_1, \ldots, c_n]$, the problem of this framework is to minimize the objective function $J(C)$

$$\min_C J(C) = \frac{1}{2} \left\| y - \sum_{i=1}^{n} v_i \cdot c_i \right\|_2^2 + \lambda \left\| C \right\|_0$$

where $\left\| C \right\|_0$ is the $\ell_0$ norm and which counts the non-zero components in the vector $C$, the parameter $\lambda > 0$ is a scalar regularization parameter that balances the tradeoff between the sparsity and the reconstruction error. Because $d \gg z$, solving Eq. (1) is an NP-hard problem. Luckily, the compressed sensing theory indicates that an approximate solution could be obtained by replacing the $\ell_0$ norm in Eq. (1) with the $\ell_1$ norm, thus

$$\min_C J(C) = \frac{1}{2} \left\| y - \sum_{i=1}^{n} v_i \cdot c_i \right\|_2^2 + \lambda \left\| C \right\|_1$$

where $\left\| C \right\|_1 = \sum_{i=1}^{n} |c_i|$. This is a well-known convex optimization problem. However, due to the high inter-class similarity nature of the scene images, the $C$ computed by Eq. (2) is sometimes not reliable. Thus a kernel-based method is proposed to project the features to a higher or even an infinite dimensional space, where those features are grouped better and more suitable for linear classification.

Consider the Lasso problem of Eq. (2) in the kernel space, i.e.

$$\min_C J(C) = \frac{1}{2} \left\| \varphi(y) - \sum_{i=1}^{n} \varphi(v_i) \cdot c_i \right\|_2^2 + \lambda \left\| C \right\|_1$$

where $\varphi(\cdot)$ maps a feature vector from feature space $d$ to a kernel space $h \in \mathbb{R}^d \rightarrow \varphi(h) \in \mathbb{R}^h$. And $D \rightarrow \tilde{D} = (\varphi(v_1), \varphi(v_2), \ldots, \varphi(v_n))$, given $K(x, x) = \varphi(x)^T \varphi(x) = 1$.

To solve the convex optimization problem of Eq. (3), we develop the Kernel Random Coordinate Descent (KRCGD) method which employs the Random Coordinate Descent approach [5]. To apply the KRCGD to Eq. (3), Nesterov [5] showed that the gradient of $J(C)$ must be coordinate-wise Lipschitz continuous

$$\left| \nabla_c J(C + he_j) - \nabla_c J(C) \right| \leq L_j |h|$$

When $c_j \neq 0$, take partial derivative of $J(C)$ with respect to $c_j$, then

$$k(v_j, v_i) = \phi(v_j)^T \cdot \phi(v_i)$$

$$k(v_j, y) = \phi(v_j)^T \cdot \phi(y)$$

$$\nabla_c J(C) = \sum_{i=1}^{n} c_i \cdot K(v_j, v_i) - K(v_j, y) + \lambda$$

After calculations, one can get

$$L_j = \max \left( \frac{\partial (\nabla_c J(C))}{\partial c} \right) = k(v_j, v_j)$$

Because $k(v_j, v_j) = 1$, the gradient of $\nabla_c J(C)$ is differentiable. And since its first derivative is bounded, the objective function is coordinate-wise Lipschitz continuous. Then, the coefficient vector $C$ could be updated as follows

$$c_j^{k+1} = c_j^k - L_j^{-1} \cdot \nabla_c J(C) \cdot e_j$$
where \( j \) is a random integer and \( j \in [0, n] \). Note that the kernel function has to satisfy \( K(x, x) = \phi(x)^T \phi(x) = 1 \), which means some kernels (e.g. polynomial or sigmoid kernel) are incompatible with the KRCD framework. Nevertheless, there are many kernel functions that could fit into the proposed framework (i.e. Gaussian kernel, Laplace kernel and Rational quadratic). Experimental results show the Laplace kernel obtained best performance for the proposed method. The mathematical expression for the Laplace kernel is

\[
K(x_i, x_j) = \exp\left(-\frac{|x_i - x_j|}{\sigma}\right)
\]

In order to assign the categorization label to each test scene image, the framework of the proposed method is introduced as follow.

Create the dictionary \( D \): as shown in Fig. 1 (a), the first step is to create the SRIGM by the \( M \times s \)-fold rotation-expansion. The second step is the dimension reduction process.

The classification component of the algorithm involves the following steps: as illustrated in Fig. 1 (b), the first step is to break down the test image and get block images at multiple scales \( l = \{0, 1, 2, \ldots\} \). Then, in order to fully utilize the scale invariant property of the SRIGM, the resolution of the block images should be adjusted. In general, one could directly resize the image using various interpolation techniques, and one block image should have \( M \) instances with different resolutions. The next step is to calculate the reconstruction residuals \( r_l^j \) of all classes at scale \( l = 0 \). If the smallest one belongs to class \( k \), the \( r_{lk}^j \) is chosen. Note that if \( l \neq 0 \), the test image will be cut into many blocks, thus the reconstruction residual is

\[
r_{lk}^j = \frac{1}{2^{2l}} \sum_{i=1;2^{2l}}^{} r_{lk}^j
\]

where \( i \) indicates the \( i \)th block. Since each block image has several resized instances, the reconstruction residual \( r_{lk}^j \) of the block image \( i \) should be equal to the reconstruction residual \( r_{lk}^j \) of the instance \( j \), where the \( r_{lk} \) is the smallest one of all instances that belong to the block image \( i \). Given a threshold \( B \), for class \( k \) at scale \( l \), if \( r_{lk}^j < \beta \) then the test image belongs to class \( k \); If \( r_{lk}^j \geq \beta \), then try again at a smaller scale (larger \( l \)). If there is no smaller scale, the system will refuse to categorize this test image.

4. Experiments and Discussion

In this section, the results of comparison experiments are presented to demonstrate the superiority of the proposed method. The limitations of the proposed method are discussed as well.

4.1 Experiment Setup and Dataset

The experiments are run with Matlab 2010a. The workstation has a dual-core CPU running at 3.2 GHz, the memory size is 4 Gb.

The experiments are carried out by using the 15-Scene Database [8]. The 15-Scene Database is built up with 4,485 images which fall into 15 categories. Each category includes 200-450 images. The resolutions of all images are resized to 256x256, and the color images are transformed to grayscale images. The partition scales for the proposed method are \( l = \{0, 1, 2, 3\} \). In comparison experiments, each category is divided into two sets of images, 50-images for training and the rest of images for testing.

4.2 Performance Analysis of the Proposed Method

The average recognition precisions of the proposed method with different rotation extensions and different training sample numbers are presented in Fig. 2. The best categorization score is obtained when the size of the training sample is 100, the rotation extension is 8 and the category number is 10. Those results suggest when there are a small number of training samples, higher rotation extension can increase the recognition precision. On the other hand, if the size of training sample is large, solely enforcing the rotation extension can degrade the algorithm performance. Higher

![Fig. 1](image1.png)  The general framework of the proposed method. (a) the multi-scale partition process; (b) the framework of the proposed method.

![Fig. 2](image2.png)  The categorization precision of the proposed method by using different rotation extension.
rotation extension also leads to larger dictionary, which degrades the system efficiency as well. These results suggest while the higher rotation extension could increase the intra-class completeness, the inter-class similarities are increasing as well. Thus, the suggested strategy is to balance the rotation extensions and the size of training sample. The optimized parameters vary according to unique properties of scene categories.

The categorization results of different sets of scene images are shown in Fig. 3. The best result is 87.1%, which is obtained on “open country” dataset.

4.3 Qualitative Comparison

The proposed method is compared with the ScSPM [1], the Gist [6], the BoV [7] and the CC-LDA [8]. The SVM with linear kernel is trained for the above 4 state-of-art methods. The training images are segmented at 4 levels for the ScSPM. Nine orientations (0, 15, 30, 45, 60, 75, 90, 105, 130) and four scales (1:32, 1:64, 1:128, 1:256) are used for implementing the Gist. Considering the intra-class variation problem, we choose 15 topics for the CC-LDA. The size of codebook is set to 1500 for the BoV. The rotation extension of the proposed method is set to 10. The results are average values of classification results from all 15 categories.

The experimental results are shown in Table 1. The CC-LDA has acquired the best score when the category number is 5. However, the proposed method and the ScSPM are only slightly worse in performance. The ScSPM has acquired the best categorization rate and outperform the proposed method by 0.8% when the category number is 10. The proposed method has acquired the highest performance when the category number is 15. The average results show that the proposed method respectively outperforms the “ScSPM”, the “Gist” feature based method, the “BoV” and the “CC-LDA” by 0.4%, 8.5%, 4.6%, and 14.1%. These results suggest that although some methods could obtain better results in cases that the category number is small, the proposed method has demonstrated reliable performances in all category sizes.

It must be noted that the average processing time of the proposed method for a single image is 37s. Thus, when applying the proposed method to some time-sensitive tasks, the process may be accelerated by reducing both the number of the partition scales and the rotation resolution, an interesting area for our further research to improve the efficiency of the system.

5. Conclusion

This paper presents a novel scale-rotation invariant generative model (SRIGM) and a kernel sparse representation classification (KSRC) method for scene categorization. The KRC is proposed to efficiently solve the $\ell_1$ minimization in kernel space and handling the high inter-class similarity problem; The SRIGM is proposed to create a generative model that invariant to scale and rotation changes, and enables the KSRC framework to solve the intra-class variation problem. The optimal parameters selection strategy is discussed as well. Experimental results show the proposed method has obtained more stable results comparing to other existing state-of-art scene categorization methods.

Table 1  Performance comparison between different methods with different category numbers.

<table>
<thead>
<tr>
<th>Category Number</th>
<th>Proposed</th>
<th>ScSPM</th>
<th>Gist</th>
<th>BoV</th>
<th>CC-LDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>79.2%</td>
<td>82.4%</td>
<td>75.1%</td>
<td>77.9%</td>
<td>83.6%</td>
</tr>
<tr>
<td>10</td>
<td>78.8%</td>
<td>79.6%</td>
<td>68.1%</td>
<td>73.5%</td>
<td>51.7%</td>
</tr>
<tr>
<td>15</td>
<td>66.4%</td>
<td>61.3%</td>
<td>55.7%</td>
<td>59.3%</td>
<td>43.8%</td>
</tr>
<tr>
<td>Avg</td>
<td>74.8%</td>
<td>74.4%</td>
<td>66.3%</td>
<td>70.2%</td>
<td>60.7%</td>
</tr>
</tbody>
</table>

References