A NOVEL KERNEL COLLABORATIVE REPRESENTATION APPROACH FOR IMAGE CLASSIFICATION

Weiyang Liu, Lijia Lu, Hui Li, Wei Wang, and Yuexian Zou**

School of Electronic and Computer Engineering, Peking University, P. R. China

ABSTRACT

Sparse representation classification (SRC) plays an important role in pattern recognition. Recently, a more generic method named as collaborative representation classification (CRC) has greatly improved the efficiency of SRC. By taking advantage of recent development of CRC, this paper explores to smoothly apply the kernel technique to further improve its performance and proposes the kernel CRC (KCRC) approach. Tested by multiple databases in experiments, KCRC has shown that it can perfectly classify the data with the same direction distribution with limited complexity, and outperforms CRC, SRC and some other conventional algorithms.

Index Terms— Collaborative Representation, Kernel Technique, Regularized Least Square Algorithm, Augmented Lagrange Multiplier Method, Image Classification

1. INTRODUCTION

Sparsity of signals has been commonly considered as a powerful principle in classification and is of increasing importance in pattern recognition recently in academics. Sparse representation technique has been widely studied in [1–3], partially thanks to the great progress of l_0 and l_1 minimization methods proposed in [4, 5].

It has been found that most natural images can be sparsely represented by over-complete basises, which inspires SRC [1]. Compared to conventional face recognition techniques, SRC can perform better according to experiments in [1]. Although the improvement of recognition accuracy was considered as the contribution of the sparsity of signals, Zhang et al. [6] have revealed that the collaborative representation (CR) mechanism is the real key to improve the recognition accuracy. Moreover, the necessity of sparsity in face recognition was also questioned in [7] and [8]. In summary, developed from SRC, CRC emphasizes CR mechanism, indicating the sparsity constraint only helps robustness, not accuracy. Experiments in [6] showed that CRC method has close or even better recognition results with far less complexity than SRC.

*Corresponding Email:{lih64,zouyx}@pkusz.edu.cn. This work was supported by National Basic Research Program of China (973Program, No.2012CB315904), National Natural Science Foundation of China (No.NSFC61179028), Postdoc National Fund (No.2013M540019), Shenzhen Fundamental Research Program (No.JCYJ20130329175141512).

SRC performs poorly when facing data of the same direction distribution [9], because relatively different direction distributions usually are required for the training samples with different labels in SRC , and CRC also shares the same limitation. Therefore, the kernel function, which was originated from SVM [10, 11] and was successfully applied to KPCA [12] and KFDA [13], is adopted in this paper to overcome such shortcoming. A Mercer kernel implicitly defines a nonlinear mapping which maps data in the input space into a high or even infinite dimensional kernel feature space where features of the same class are more easily grouped together and therefore features of different classes become linear separable. It was shown in [9, 14] that SRC performs better by adopting the kernel function. The contribution of this paper is that we smoothly combine CRC method and the kernel technique to establish a novel kernel CRC approach that has absorbed the advantages of both techniques.

The outline of the paper is as follows. In Section II, CRC and the kernel technique are briefly introduced. In Section III, the proposed KCRC and two specific KCRC algorithms are presented. Experimental results are provided and discussed in Section IV, and concluding remarks are given in Section V.

2. PRELIMINARIES

2.1. Principle of CRC

The principle of CRC [6] is briefly presented in this section. In CRC, the dictionary $\boldsymbol{D}$ is constructed by all training samples and a test sample $\boldsymbol{y}$ is coded collaboratively over the whole dictionary, which is the essence of CRC.

Assume there is a set of training samples containing total k classes, the ith class with n_i samples. The training data matrix $\boldsymbol{D}$, also called dictionary, can be formulated as

$$\boldsymbol{D} = \{\boldsymbol{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_k\} \in \mathbb{R}^{m \times n} \tag{1}$$

where $n{=}\sum_{j=1}^{k} n_j$ and $\boldsymbol{D}_i$ denotes the dictionary associated with ith class, namely

$$\boldsymbol{D}_i = \{\boldsymbol{v}_{i1}, \boldsymbol{v}_{i2}, \cdots, \boldsymbol{v}_{in_i}\} \in \mathbb{R}^{m \times n_i} \tag{2}$$

in which m is the dimension of each sample and equals to $w \times h$ in the context of image recognition. $\boldsymbol{v}_{ij}$, also called atom, stands for the jth training image in the ith class.

According to the representation principle, a test image $\boldsymbol{y}$ belonging to the ith class can be represented by

$$\boldsymbol{y} \approx x_{i1}\boldsymbol{v}_{i1} + x_{i2}\boldsymbol{v}_{i2} + \cdots + x_{in_i}\boldsymbol{v}_{in_i} \tag{3}$$

where $x_{ij}\in\mathbb{R}, j=1,\cdots,n_i$ is the coding coefficient of vector $\boldsymbol{v}_{ij}$. By defining the vector $\boldsymbol{x}=\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_i,\cdots,\boldsymbol{x}_k\}^T$ in which $\boldsymbol{x}_i$ is the coding coefficients associated with $\boldsymbol{D}_i$, the test sample $\boldsymbol{y}$ can also be formulated as

$$\boldsymbol{y} \approx \boldsymbol{D}\boldsymbol{x} \in \mathbb{R}^m \tag{4}$$

where $\{0,\cdots,0,x_{i1},\cdots,x_{in_i},0,\cdots,0\}$ is the ideal form of the coefficient vector $\boldsymbol{x}$ in SRC. Only coefficients of the class to which the test sample belongs are non-zero. However, $\boldsymbol{x}$ is usually not necessary to be sparse in CRC.

The test sample $\boldsymbol{y}$ can be written as $\boldsymbol{y}=\hat{\boldsymbol{y}}+\boldsymbol{e}$ where $\hat{\boldsymbol{y}}=\boldsymbol{D}\boldsymbol{x}$ is the estimated test sample and $\boldsymbol{e}$ is the residual caused by noise, occlusion and etc. The vector $\boldsymbol{x}$ can be found by solving the constrained optimization problem as follows:

$$\hat{\boldsymbol{x}} = \arg\min_x \|\boldsymbol{x}\|_{l_p} \text{ subj. to } \|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{x}\|_{l_q} \le \varepsilon \tag{5}$$

where ε is a small error constant. After Lagrangian formulation, a general model of CRC method is formulated as

$$\hat{\boldsymbol{x}} = \arg\min_x \left(\|\boldsymbol{y}-\boldsymbol{D}\boldsymbol{x}\|_{l_q} + \mu\|\boldsymbol{x}\|_{l_p} \right) \tag{6}$$

where μ is the regularization parameter and $p, q \in \{1,2\}$. The combinations of p, q values lead to different instantiations of CRC model. For instance, SRC method is under the condition of $p=1, q\in\{1,2\}$ and different settings of q is used to handle recognition with or without occlusion. Similar to SRC, CRC uses the coding coefficients $\hat{\boldsymbol{x}}$ to obtain the reconstructed image and determine its class label with following equation:

$$identity(\boldsymbol{y}) = \arg\min_i \left(\|\boldsymbol{y}-\boldsymbol{D}_i\hat{\boldsymbol{x}}_i\|_2 / \|\hat{\boldsymbol{x}}_i\|_2 \right) \tag{7}$$

In fact, the key to reduce the computational complexity is to reasonably set the value of p, q. Based on different combinations of p, q, two CRC algorithms were proposed in [6, 15]. One is the CRC regularized least square (CRC-RLS) algorithm with $p=2, q=2$. Another one is the robust CRC (RCRC) algorithm with $p=2, q=1$. It has been comprehensively analyzed in [6] that sparsity of signals is useful but not crucial for face recognition and has been proven that the collaborative representation mechanism does play an important role.

2.2. Kernel Technique

Kernel function [12, 13], also described as kernel technique, has become a simple yet useful tool in pattern recognition since its success in SVM [10, 11], KPCA [12] and KFDA [13]. Kernel function is used to create a nonlinear mapping mechanism $\boldsymbol{y}\in\mathbb{R}\mapsto\boldsymbol{\phi}(\boldsymbol{y})\in\mathbb{H}$ where $\mathbb{H}$ is a unique associated reproducing kernel Hilbert space. If every data point is mapped into higher dimensional space via transformation $\boldsymbol{\phi}$, the inner product, which is called kernel function, becomes

$$K(\boldsymbol{y}_i,\boldsymbol{y}_j) = \langle \boldsymbol{\phi}(\boldsymbol{y}_i), \boldsymbol{\phi}(\boldsymbol{y}_j) \rangle = \boldsymbol{\phi}(\boldsymbol{y}_i)^T \boldsymbol{\phi}(\boldsymbol{y}_j) \tag{8}$$

where $\boldsymbol{y}_i, \boldsymbol{y}_j$ are different samples and $\boldsymbol{\phi}$ denotes the implicit nonlinear mapping associated with the kernel function $K(\boldsymbol{y}_i,\boldsymbol{y}_i)$. There are some empirical frequently-used kernel functions satisfying the Mercer condition such as the linear kernel $K(\boldsymbol{y}_i,\boldsymbol{y}_j)=\boldsymbol{y}_i^T\boldsymbol{y}_j$ and Gaussian radial basis function (RBF) kernels $K(\boldsymbol{y}_i,\boldsymbol{y}_j)=\exp(-\beta\|\boldsymbol{y}_i^T-\boldsymbol{y}_j\|_2^2)$.

3. PROPOSED KCRC APPROACH

3.1. Formulation of KCRC

It is well-known that data nonlinearly transformed to high dimensional space is more separable. It could also avoid the same direction distribution of data. Obviously, the new features in kernel space also can be applied to CRC. However, mapping to high dimensional space makes optimization problem in CRC more complicated so that the dimensionality in the kernel feature space has to be reduced. The nonlinear mapping mechanism is formulated as

$$\boldsymbol{y}\in\mathbb{R}^m \mapsto \boldsymbol{\phi}(\boldsymbol{y}) = [\phi_1(\boldsymbol{y}),\phi_2(\boldsymbol{y}),\cdots,\phi_s(\boldsymbol{y})] \in \mathbb{R}^s \tag{9}$$

where $\boldsymbol{\phi}(\boldsymbol{y})$ is the high dimensional feature associated to the sample $\boldsymbol{y}$ and $s\gg m$. Define that $\boldsymbol{d}_i^{[j]}$ denotes the ith atom in $\boldsymbol{D}$ with the jth class label, in which $1\le i\le n=\sum_{j=1}^k n_j$, but the class label $[j]$ will be removed for the convenience of description. According to the nonlinear mapping mechanism, the original dictionary $\boldsymbol{D}$ becomes a much higher dimensional one: $\boldsymbol{\Phi}=\{\boldsymbol{\phi}(\boldsymbol{d}_1),\boldsymbol{\phi}(\boldsymbol{d}_2),\cdots,\boldsymbol{\phi}(\boldsymbol{d}_n)\}\in\mathbb{R}^{s\times n}$, and the test sample becomes $\boldsymbol{\phi}(\boldsymbol{y})=\boldsymbol{\Phi}\boldsymbol{x}$. The model of KCRC can be formulated as

$$\hat{\boldsymbol{x}} = \arg\min_x \|\boldsymbol{x}\|_{l_p} \text{ subj. to } \|\boldsymbol{\phi}(\boldsymbol{y})-\boldsymbol{\Phi}\boldsymbol{x}\|_{l_q} \le \varepsilon. \tag{10}$$

However, Eq. (10) is even harder to solve than the previous problem Eq. (5) because the high dimensionality leads to high complexity.

A dimensionality reduction matrix $\boldsymbol{R}$, namely a projection matrix, can be defined or constructed by utilizing the methodology of KPCA [12] and KFDA [13]. With the matrix $\boldsymbol{R}\in\mathbb{R}^{s\times c}$, we could obtain

$$\boldsymbol{R}^T\boldsymbol{\phi}(\boldsymbol{y}) = \boldsymbol{R}^T\boldsymbol{\Phi}\boldsymbol{x}. \tag{11}$$

where $\boldsymbol{R}$ is related to the samples in the kernel space. In fact, each atom in $\boldsymbol{R}$ is a linear combination of samples in kernel feature space. Namely

$$\boldsymbol{R} = \boldsymbol{\Phi}\boldsymbol{\Psi} = \{\boldsymbol{\phi}(\boldsymbol{d}_1),\cdots,\boldsymbol{\phi}(\boldsymbol{d}_n)\}\cdot\{\boldsymbol{\psi}_1,\cdots,\boldsymbol{\psi}_c\} \tag{12}$$

where $\boldsymbol{R}=\{\boldsymbol{R}_1,\cdots,\boldsymbol{R}_c\}$ and $\boldsymbol{\psi}_i$ is the $\boldsymbol{n}$-dimensional linear projection coefficients vector corresponding to the $\boldsymbol{R}_i=$

$\sum_{j=1}^{n} \psi_{i,j}\phi(\boldsymbol{d}_j)=\boldsymbol{\Phi}\boldsymbol{\psi}_i$. In [9], $\boldsymbol{\Psi}\in\mathbb{R}^{n\times c}$ is called pseudo-transformation matrix. Then we put Eq. (12) into Eq. (11):

$$(\boldsymbol{\Phi}\boldsymbol{\Psi})^T\boldsymbol{\phi}(\boldsymbol{y}) = (\boldsymbol{\Phi}\boldsymbol{\Psi})^T\boldsymbol{\Phi}\boldsymbol{x} \tag{13}$$

from which we can get $\boldsymbol{\Psi}^T\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y})=\boldsymbol{\Psi}^T\boldsymbol{G}\boldsymbol{x}$, where $\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y})=[K(\boldsymbol{d}_1,\boldsymbol{y}),\cdots,K(\boldsymbol{d}_n,\boldsymbol{y})]^T$. $\boldsymbol{G}\,(G_{ij}=K(\boldsymbol{d}_i,\boldsymbol{d}_j))$ is defined as the kernel Gram matrix that is symmetric and positive semi-definite according to Mercer's theorem. Since $\boldsymbol{G}$ and $K(\boldsymbol{D}_i,\boldsymbol{y})$ are given a prior, the last step is to find $\boldsymbol{\Psi}$ instead of finding $\boldsymbol{R}$. Several methods are introduced in [9, 12, 13] to determine the pseudo transformation matrix $\boldsymbol{\Psi}$. Particularly, when $\boldsymbol{\Psi}$ is an identity matrix, no dimensionality reduction is applied. Moreover, $\boldsymbol{\Psi}$ can also be a random projection matrix to achieve dimensionality reduction.

After substituting the constraint in CRC model with equivalent kernel function constraint, we have

$$\hat{\boldsymbol{x}} = \arg\min_{x} \|\boldsymbol{x}\|_{l_p} \text{ subj. to } \|\boldsymbol{\Psi}^T\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y}) - \boldsymbol{\Psi}^T\boldsymbol{G}\boldsymbol{x}\|_{l_q} < \varepsilon \tag{14}$$

which is the model of KCRC approach. Additionally, a small perturbation would be added to $\boldsymbol{\Psi}^T\boldsymbol{G}$ if the norm of a column is close to 0. Namely, add a small number to every element of matrix $\boldsymbol{\Psi}^T\boldsymbol{G}$. The model of KCRC can be formulated to another form shown as follows:

$$\hat{\boldsymbol{x}} = \arg\min_{x} \left(\|\boldsymbol{\Psi}^T\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y}) - \boldsymbol{\Psi}^T\boldsymbol{G}\boldsymbol{x}\|_{l_q} + \mu\|\boldsymbol{x}\|_{l_p} \right) \tag{15}$$

from which we could derive two specific algorithms. With $p=2, q=2$, the l_2 minimization model can efficiently solve $\boldsymbol{x}$ with low computational complexity. The regularized least square algorithm is applied to solve this optimization problem (Algorithm 1). Facing images with occlusion and corruption, we could set $p=2, q=1$. More robustness is gained by making the first term a l_1 regularized one. Let $\boldsymbol{e}=\boldsymbol{\Psi}^T\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y}) - \boldsymbol{\Psi}^T\boldsymbol{G}\boldsymbol{x}$ and $p=2, q=1$. Eq. (14) is rewritten as

$$\begin{aligned} \hat{\boldsymbol{x}} &= \arg\min_{x} \left(\|\boldsymbol{e}\|_{l_1} + \mu\|\boldsymbol{x}\|_{l_2} \right) \\ \text{subj. to } & \boldsymbol{\Psi}^T\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y}) = \boldsymbol{\Psi}^T\boldsymbol{G}\boldsymbol{x} + \boldsymbol{e} \end{aligned} \tag{16}$$

which is a constrained convex optimization problem that can be solved by Augmented Lagrange Multiplier (ALM) method [16, 17] as Algorithm 2 shown.

3.2. Proposed KCRC Algorithms

Two algorithms are designed for KCRC. For normal situations, p, q are both set as 2. The regularized least square algorithm is applied to solve the model with $p, q=2$. Specifically, we give the new dictionary $\boldsymbol{D}'=\boldsymbol{\Psi}^T\boldsymbol{G}$ and define $\boldsymbol{P}'$ as the coding basis in kernel CRC-RLS (KCRC-RLS). Namely

$$\boldsymbol{P}' = \left((\boldsymbol{\Psi}^T\boldsymbol{G})^T(\boldsymbol{\Psi}^T\boldsymbol{G}) + \mu\cdot\boldsymbol{I} \right)^{-1} \left(\boldsymbol{\Psi}^T\boldsymbol{G} \right)^T. \tag{17}$$

The test sample is transformed to $\boldsymbol{\Psi}^T\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y})$. For high level corruption and occlusion, kernel robust CRC (KRCR-C) algorithm ($p=2, q=1$) is designed. $\boldsymbol{D}''=\boldsymbol{\Psi}^T\boldsymbol{G}$ is the new dictionary and $\boldsymbol{P}_k''$ is the coding basis in the kernel space.

Algorithm 1: KCRC-RLS

1. Normalize the columns of $\boldsymbol{D}'=\boldsymbol{\Psi}^T\boldsymbol{G}$ to unit l_2-norm.
2. Represent $\boldsymbol{y}' = \boldsymbol{\Psi}^T\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y})$ over dictionary $\boldsymbol{D}'$ by
$$\hat{\boldsymbol{p}}' = \boldsymbol{P}'\boldsymbol{y}'$$
where $\boldsymbol{P}' = (\boldsymbol{D}'^T\boldsymbol{D}' + \mu\boldsymbol{I})^{-1}\boldsymbol{D}'^T$.
3. Obtain the regularized residuals
$$r_i = \frac{\|\boldsymbol{y} - \boldsymbol{D}'\hat{\boldsymbol{p}}_i'\|_2}{\|\hat{\boldsymbol{p}}_i'\|_2}.$$
where $\hat{\boldsymbol{p}}_i'$ is the coding coefficients associated with class i over $\boldsymbol{P}'$.
4. Output the identity of $\boldsymbol{y}'$ (class label) as
$$identity(\boldsymbol{y}') = \arg\min_i(r_i).$$

Algorithm 2: KRCRC

1. Normalize the columns of $\boldsymbol{D}''=\boldsymbol{\Psi}^T\boldsymbol{G}$ to unit l_2-norm.
2. Input $\boldsymbol{y}''=\boldsymbol{\Psi}^T\boldsymbol{K}(\boldsymbol{D},\boldsymbol{y})$, $\boldsymbol{x}_0$, $\boldsymbol{e}_0$, $k=1$ and $\tau > 0$.
3. Proceed if $|\boldsymbol{x}_{k+1} - \boldsymbol{x}_k| > \tau$ is true. If not, output $\hat{\boldsymbol{e}}, \hat{\boldsymbol{x}}$ and go to step 5.
4. Do the following iteration:
$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{P}_k''(\boldsymbol{y}'' - \boldsymbol{e}_k + \boldsymbol{z}_k/\sigma_k) \\ \boldsymbol{e}_{k+1} &= S_{1/\sigma_k}(\boldsymbol{y}'' - \boldsymbol{D}''\boldsymbol{x}_{k+1} + \boldsymbol{z}_k/\sigma_k) \\ \boldsymbol{z}_{k+1} &= \boldsymbol{z}_k + \sigma_k(\boldsymbol{y}'' - \boldsymbol{D}''\boldsymbol{x}_{k+1} - \boldsymbol{e}_{k+1}) \end{aligned}$$
where $\boldsymbol{P}_k''=(\boldsymbol{D}''^T\boldsymbol{D}'' + 2\mu/\sigma_k\boldsymbol{I})^{-1}\boldsymbol{D}''^T$ and $S_m, m\geq 0$ is the shrinkage coefficient. $k=k+1$ and go to step 3.
5. Represent $\boldsymbol{y}''$ over dictionary $\boldsymbol{D}''$ by the converged $\boldsymbol{x}$.
6. Obtain the regularized residuals
$$r_i = \frac{\|\boldsymbol{y}'' - \boldsymbol{D}''\hat{\boldsymbol{x}}_i\|_2}{\|\hat{\boldsymbol{x}}_i\|_2}.$$
where $\hat{\boldsymbol{x}}_i$ is the coding coefficients related to class i.
7. Output the identity of $\boldsymbol{y}''$ (class label) as
$$identity(\boldsymbol{y}'') = \arg\min_i(r_i).$$

4. EXPERIMENTS AND RESULTS

4.1. Experiments on Data Sets with the Same Direction Distribution

Comparisons between CRC and KCRC on self-built data sets are presented. Two class p-dimensional data sets are built: $\boldsymbol{Q}=\{\boldsymbol{q}_1,\cdots,\boldsymbol{q}_n\}\in\mathbb{R}^{p\times n}$, $\boldsymbol{W}=\{\boldsymbol{w}_1,\cdots,\boldsymbol{w}_n\}\in\mathbb{R}^{p\times n}$, where $\boldsymbol{q}_1, -\boldsymbol{w}_1\in[1,3]$ and $\boldsymbol{q}_i=-\boldsymbol{q}_{i+1}, \boldsymbol{w}_j=-\boldsymbol{w}_{j+1}$. Then we uniformly pick value from $[1,3]$ and $[-3,-1]$ to construct $\boldsymbol{q}_1, \boldsymbol{w}_1$ respectively. Then all data is corrupted by Gaussian nose with zero mean and 0.04 variance. By experiments, 2-dimensional situation is shown in Figure 1(a) and Figure 1(b). With different dimension $p = 2, 8, 16, 32, 64, 128, 256$, recognition rates of CRC and KCRC are shown in Figure 1(c). It can be found out that KCRC-RLS performs much better than CRC-RLS when facing data with the same direction distribution.

4.2. Experiments on Face Data Sets

To evaluate the performance of the proposed KCRC algorithms, a public face data set (Extended YaleB database) has been used. The specific experiment setup is the same as [1].

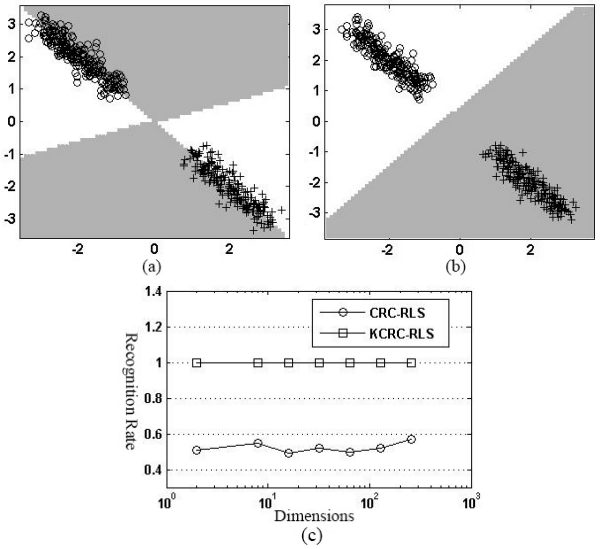

Fig. 1. Recognition result on self-built data set. In (a) and (b), "+" represents data set $\boldsymbol{Q}$ while "○" stands for data set $\boldsymbol{W}$. Gray area and white area are judged to class $\boldsymbol{Q}$ and $\boldsymbol{W}$ respectively. (a)Decision boundary by CRC-RLS. (b)Decision boundary by KCRC-RLS. (c)Recognition rate vs. dimension.

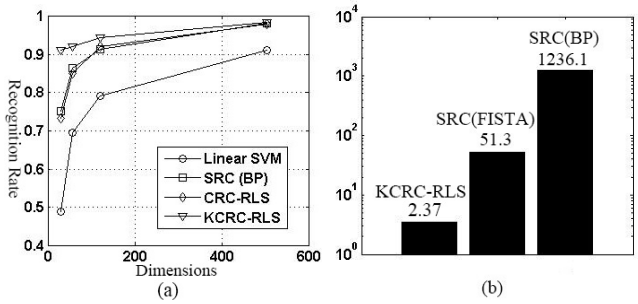

Fig. 2. (a)Comparison of recognition accuracy. SRC (BP) is solved by l_1 solver in Matlab sparselab toolbox (b)Comparison of test time while the test time of CRC-RLS is set to 1 for better comparison.

4.2.1. Recognition Performance Test

The recognition performance test has used random test [1] based on Extended Yale B database. Each result is averaged out by 20 times experiments. Recognition accuracy of KCRC, CRC, SRC and linear SVM are compared in experiments shown in Figure 2, in which we use the random projection matrix as our $\boldsymbol{\Psi}$. Moreover, downsample is adopted for face feature extraction. It is obtained that KCRC-RLS can achieve the best recognition accuracy only with slightly more running time than CRC-RLS, but much less than SRC.

4.2.2. Robustness Test

Since KCRC-RLS algorithm is developed to cope with the recognition task under low occlusion and corruption, KRCRC algorithm is designed to process the recognition with high occlusion and corruption. Pixel corruption and random block occlusion are adopted to do robustness test whose results are shown in Table 1. Experiments show the robustness of KRCRC is slightly better than RCRC. So KRCRC preserves good robustness.

Table 1. Recognition performance comparison on occlusion and corruption between R-SRC, RCRC and KCRC. R-SRC is the robust SRC [1]. The way we simulate the occlusion and corruption situation is also the same as [1].

Occlusion	**0%**	**10%**	**20%**	**30%**	**40%**	**50%**
R-SRC(BP)	100%	100%	99.3%	97.9%	89.7%	63.9%
RCRC	100%	100%	100%	97.3%	92.5%	83.1%
KRCRC	100%	100%	100%	97.8%	93.1%	83.7%
Corruption	**0%**	**10%**	**30%**	**50%**	**70%**	**90%**
R-SRC(BP)	100%	100%	100%	100%	91.0%	6.9%
RCRC	100%	100%	100%	100%	91.1%	18.2%
KRCRC	100%	100%	100%	100%	91.9%	20.2%

4.3. Experiments on Car Logos Data Sets

This experiment is to further test the performance of KCRC approach. A portion of car logos are automatically collected by a video surveillance camera while others are downloaded from the internet. So the car logo images have large variations in terms of image resolution, lighting condition, noise condition, background et al. Data samples are given in Figure 3(a). There are 3 data sets: 300 images (10 classes), 500 images (13 classes) and 1000 images (21 classes). We randomly select 10 images in each class as training samples and the remaining as test samples. Results are averaged across 20 times experiments for accuracy. It can be seen that KCRC-RLS and KRCRC perform well in the experiment. Stable recognition rates show that KRCRC is more robust than KCRC-RLS.

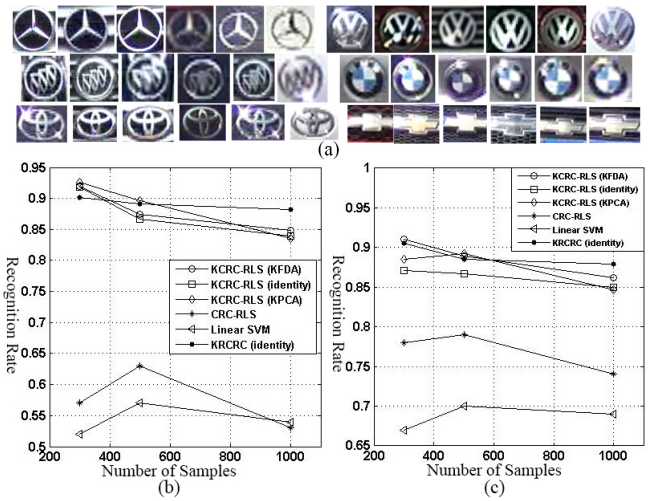

Fig. 3. Experiment on car logos database. KFDA and KPCA stand for the kernel space dimensionality reduction method used in KFDA, KPCA respectively. Identity means no dimensionality reduction in the kernel space. (a)Car logos data (b)All images are downsampled and resized to 32×32 (1024 dimensions). (c)Eigenface is adopted to extract the feature (80 dimensions).

5. CONCLUSION

In this work, we have elaborated the CRC method [6] and proposed the kernel CRC approach. Kernel technique is smoothly combined with the CRC method. The proposed KCRC enhances the discrimination ability of CRC, making the decision boundary more reasonable, and achieves the high recognition accuracy with low complexity. Experiments verified that KCRC approach outperforms CRC, SRC and linear SVM.

6. REFERENCES

[1] John Wright, Allen Y Yang, Arvind Ganesh, Shankar S Sastry, and Yi Ma, "Robust face recognition via sparse representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 2, pp. 210–227, 2009.

[2] Julien Mairal, Michael Elad, and Guillermo Sapiro, "Sparse representation for color image restoration," *Image Processing, IEEE Transactions on*, vol. 17, no. 1, pp. 53–69, 2008.

[3] Ke Huang and Selin Aviyente, "Sparse representation for signal classification," in *Advances in neural information processing systems*, 2006, pp. 609–616.

[4] David L Donoho, "For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution," *Communications on pure and applied mathematics*, vol. 59, no. 6, pp. 797–829, 2006.

[5] Joel A Tropp and Stephen J Wright, "Computational methods for sparse solution of linear inverse problems," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 948–958, 2010.

[6] Lei Zhang, Meng Yang, and Xiangchu Feng, "Sparse representation or collaborative representation: Which helps face recognition?," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 471–478.

[7] Roberto Rigamonti, Matthew A Brown, and Vincent Lepetit, "Are sparse representations really relevant for image classification?," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1545–1552.

[8] Qinfeng Shi, Anders Eriksson, Anton van den Hengel, and Chunhua Shen, "Is face recognition really a compressive sensing problem?," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 553–560.

[9] Li Zhang, Wei-Da Zhou, Pei-Chann Chang, Jing Liu, Zhe Yan, Ting Wang, and Fan-Zhang Li, "Kernel sparse representation-based classifier," *Signal Processing, IEEE Transactions on*, vol. 60, no. 4, pp. 1684–1695, 2012.

[10] Vladimir N Vapnik, "Statistical learning theory," *Wiley*, 1998.

[11] Christopher JC Burges, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[12] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller, "Kernel principal component analysis," in *Artificial Neural NetworksłICANN'97*, pp. 583–588. Springer, 1997.

[13] Sebastian Mika, Gunnar Ratsch, Jason Weston, Bernhard Scholkopf, and KR Mullers, "Fisher discriminant analysis with kernels," in *Neural Networks for Signal Processing IX, 1999. Proceedings of the 1999 IEEE Signal Processing Society Workshop.* IEEE, 1999, pp. 41–48.

[14] Shenghua Gao, Ivor Wai-Hung Tsang, and Liang-Tien Chia, "Kernel sparse representation for image classification and face recognition," in *Computer Vision–ECCV 2010*, pp. 1–14. Springer, 2010.

[15] Lei Zhang, Meng Yang, Xiangchu Feng, Yi Ma, and David Zhang, "Collaborative representation based classification for face recognition," *arXiv Preprint arXiv:1204.2358*, 2012.

[16] Dimitri P Bertsekas, "Nonlinear programming," *Athena Scientific*, 1999.

[17] Zhouchen Lin, Minming Chen, and Yi Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *arXiv Preprint arXiv:1009.5055*, 2010.