

Kernel sparse representation-based classifier ensemble for face recognition

Li Zhang · Wei-Da Zhou · Fan-Zhang Li

Published online: 19 April 2013
© Springer Science+Business Media New York 2013

Abstract Kernel sparse representation-based classifier (KSRC) has been proposed, which has good representation and classification performance on face image data. The performance of KSRC on face image data is partly dependent on the random projection matrix when using the random projection method and the kernel Gram matrix. This paper develops the kernel sparse representation-based classifier ensemble (KSRCE), which does not require to consider the effect of random projection and kernel Gram matrix on KSRC. Actually, the random projection matrix and the kernel Gram matrix could be used for designing the diversity schemes for KSRCE. In the combination stage, we can combine the labels or the reconstruction errors of a test sample. Experimental results on three face data sets show that KSRCE is very promising.

Keywords Sparse representation-based classifier · Kernel method · Ensemble learning

1 Introduction

At present, face recognition is an active research area in pattern recognition and machine learning [7, 16]. The face image data is the typical high dimensionality

Supported in part by the National Natural Science Foundation of China under Grant Nos. 60970067 and 61033013, by the Natural Science Foundation of Jiangsu Province of China under Grant Nos. BK2011284, BK201222725, by the Natural Science Pre-research Project of Soochow University under Grant No. SDY2011B09 and by the Qing Lan Project.

L. Zhang (✉) · F.-Z. Li
School of Computer Science and Technology, Soochow University, Suzhou,
215006 Jiangsu, China
e-mail: lizhang.ml@gmail.com

W.-D. Zhou
AI Speech Ltd., Suzhou 215123, China

data, and face recognition is the typical small sample size problem. There are many classes (different persons), and only a few images per person. Usually, we need to reduce the dimension of face image when performing face recognition tasks. Recently, sparse representation-based classifier (SRC) is proposed in [18, 19]. Similar to nearest neighbor (NN) [4] and nearest subspace (NS) [11, 12], SRC is a non-parameter learning method which does not need a training process and can directly assign a class label to a test sample. SRC implements sparse representation of data by using the methods for sparse signal reconstruction in CS (Compressed sensing) and classifies data in terms of reconstruction errors. Random projection (RP) is a good choice for reducing dimensionality, since Wright et al. state that “the precise choice of feature space is no longer critical, even random features contain enough information to recover the sparse representation and hence correctly classify any test image” [18]. In RP, the random matrix is regarded as the projection matrix and is generally generated according to the standard Gaussian distribution. Experimental results in [18] show SRC outperforms NN and NS. However, SRC can not work well when data points have the same direction distribution [23].

To Remedy it, Zhang et al. propose kernel sparse representation-based classifier (KSRC) [23]. KSRC implicitly map the data into a high-dimensional kernel feature space by using some nonlinear mapping associated with a kernel function. In the kernel feature space, the dimensionality can be reduced by exploiting kernel-based dimensionality reduction methods, such as kernel principle component analysis (KPCA) [17], kernel fisher discriminant analysis (KFDA) [14], and random projection [23]. Experimental results in both [18] and [23] indicate that random projection is a nice scheme of dimensionality reduction for face recognition. But random projection has randomness when generating the random projection matrix, which would lead to slightly different classification performance. In addition, the selection of kernel parameters is also difficult.

Classifier ensemble has been considered as a very efficient technique, which can lead to good classification performance by combining multiple classifiers. General speaking, the classification performance of ensemble classifier is better than the single best classifier, which is supported by experimental results [10, 15, 20]. The diversity of individual classifiers and the combination rule for the outputs of these classifiers are two important issues in classifier ensemble. In order to get better performance, the individual classifiers must be both diverse and accurate. Diversity can ensure that all the individual classifiers make uncorrelated errors. In addition, accuracy of individual classifiers is important, since too many poor classifiers can suppress correct predictions of good classifiers. In order to make individual classifiers diverse, one common way is to train individual classifiers by using different training sets, randomly selected [1, 4, 5]. Bagging [1] and Boosting [5] are well known examples of successful iterative methods for reducing a generalization error. The other way is to train multiple classifiers by using different feature sets [8]. Furthermore, combination rules is also an important issue, which is related to how to combine the outputs of individual classifiers. If labels are available, a simple (majority) voting (SV) rule can be used [9]. If the continuous outputs like posteriori probabilities are supplied, an average or some other linear combination rule or some nonlinear combination rules can be used [6, 9, 20].

This paper deals with the ensemble of KSRCs (KSRCE), which can eliminate the randomness of random projection and improve the classification of KSRCs. In KSRCE, the diversity is guaranteed by generating random projection matrix and/or

the different kernel or its parameters and/or the random sample of kernel matrix. In the combination stage, we can combine the labels or the reconstruction errors of a test sample. Usually, SRC and KSRC are cast into a quadratically constrained ℓ_1 minimization problem. Here, we recast KSRC into a linear programming (LP) problem, which is more easy to solve than quadratically constrained ℓ_1 minimization problems.

The rest of this paper is organized as follows. In Section 2, we briefly review KSRC and give the LP problem of KSRC. Section 3 discusses kernel sparse representation-based classifier ensemble (KSRCE), and describes the diversity and the combination rules in KSRCE, respectively. KSRCE is compared with KSRC and SRC and SRC ensemble (SRCE) on the face data in Section 4. Section 5 concludes this paper.

2 Kernel sparse representation-based classifier

This section reviews KSRC and presents an LP for KSRC.

2.1 KSRC

In the following, we simply introduce KSRC [23]. Consider a c -class classification task. Let the training set be $\{\mathbf{x}_i, y_i\}_{i=1}^n$, where $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^m$, $y_i \in \{1, 2, \dots, c\}$, n is the total number of training samples, and m is the dimensionality of the input space \mathcal{X} . Given an arbitrary sample \mathbf{x} in \mathcal{X} , the goal is to assign a label to it.

Let Φ be the nonlinear mapping function corresponding to a kernel $k(\cdot, \cdot)$. Usually, a Mercer kernel $k(\cdot, \cdot)$ can be expressed as

$$k(\mathbf{x}, \mathbf{x}') = \Phi(\mathbf{x})^T \Phi(\mathbf{x}') \tag{1}$$

where T denotes the transpose of a matrix or vector, \mathbf{x} and \mathbf{x}' are any two points in \mathcal{X} . To make the training samples separable, we employ Φ to map the data from the input space \mathcal{X} to a high-dimensional (possibly infinite dimensional) kernel feature space \mathcal{F} . In the finite dimensional case, there have

$$\Phi : \mathbf{x} \in \mathcal{X} \rightarrow \Phi(\mathbf{x}) = [\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \dots, \phi_D(\mathbf{x})]^T \in \mathcal{F} \tag{2}$$

where $\Phi(\mathbf{x}) \in \mathbb{R}^D$ is the image of \mathbf{x} in \mathcal{F} , $D \gg m$ is the dimension of the feature space \mathcal{F} , and $\phi_i(\mathbf{x}) \in \mathbb{R}$. The conclusions obtained from the finite dimensional case can be applied to the case of infinite dimension. Thus, hereafter our discussion only focus on the finite dimensional case for the convenience of description. The images of the training samples \mathbf{x}_i are $\Phi(\mathbf{x}_i)$, $i = 1, \dots, n$. In KSRC, we can linearly represent the image of test sample in terms of the images of all training samples in this kernel feature space \mathcal{F} . Namely,

$$\Phi(\mathbf{x}) = \sum_{i=1}^n \alpha_i \Phi(\mathbf{x}_i) = \Phi \alpha \tag{3}$$

where the sample matrix in \mathcal{F} can be expressed as

$$\Phi = [\Phi(\mathbf{x}_1), \Phi(\mathbf{x}_2), \dots, \Phi(\mathbf{x}_n)] \in \mathbb{R}^{D \times n} \tag{4}$$

$\alpha = [\alpha_1, \alpha_2, \dots, \alpha_n]^T$ is the coefficient vector, and α_i are the coefficients corresponding to the images $\Phi(\mathbf{x}_i)$. If the test sample \mathbf{x} belongs to the j th class, then the entries of α are expected to be zero except those associated with this class. Namely,

$$\alpha = [0, \dots, 0, \alpha_{j,1}, \dots, \alpha_{j,n_j}, 0, \dots, 0]^T \tag{5}$$

where n_j is the number of samples in the j th class, and $n = \sum_{j=1}^c n_j$. The coefficient vector α is expected to be sparse.

In KSRC, the problem of finding the coefficient vector is formulated as a convex programming problem:

$$\begin{aligned} \min_{\alpha} \quad & \|\alpha\|_1 \tag{6} \\ \text{subject to} \quad & \|\mathbf{B}^T \mathbf{k}(\cdot, \mathbf{x}) - \mathbf{B}^T \mathbf{K} \alpha\|_2 \leq \varepsilon \end{aligned}$$

where $\mathbf{K} = \Phi^T \Phi \in \mathbb{R}^{n \times n}$ is the kernel Gram matrix which is symmetric and positive semi-definite, and $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, $\mathbf{k}(\cdot, \mathbf{x}) = [k(\mathbf{x}_1, \mathbf{x}), \dots, k(\mathbf{x}_n, \mathbf{x})]^T = \Phi^T \Phi(\mathbf{x})$, \mathbf{B} is a pseudo-transformation matrix, and ε is a small positive constant, say 10^{-3} . In the case of random projection, \mathbf{B} is a random matrix.

Equation (6) is a quadratically constrained ℓ_1 minimization problem. By solving it, we can get the coefficient vector α . Now we need to classify \mathbf{x} in terms of α . Likewise, we also use the minimum residual between \mathbf{x} and its c approximations in the reduced subspace to determine the label of \mathbf{x} . For class j , we define a characteristic function δ_j which can pick up the coefficients corresponding to the j -th class. Namely,

$$\delta_j(a_i) = \begin{cases} \alpha_i, & \text{if } y_i = j \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

By using which, we get only the coefficients of samples belonging to class j and denote them by a new vector:

$$\delta_j = [\delta_j(a_1), \delta_j(a_2), \dots, \delta_j(a_n)]^T \tag{8}$$

Thus, the j -th approximation to the test sample \mathbf{x} in the reduced subspace can be expressed as $\mathbf{B}^T \mathbf{K} \delta_j$. We get the estimated label \hat{y} for \mathbf{x} by minimizing residual between the $\mathbf{B}^T \mathbf{k}(\cdot, \mathbf{x})$ and its approximations. Then, we get

$$\hat{y} = \arg \min_{j=1, \dots, c} r_j(\mathbf{x}) = \arg \min_{j=1, \dots, c} \|\mathbf{B}^T \mathbf{k}(\cdot, \mathbf{x}) - \mathbf{B}^T \mathbf{K} \delta_j\|_2 \tag{9}$$

2.2 Linear programming for KSRC

The convex problem (6) can be efficiently solved [3, 18, 19]. In [18], ℓ_1 – magic software package [2] is used to solve the problem. Here, we modify the convex problem (6) as follows.

$$\begin{aligned} \min_{\alpha} \quad & \|\alpha\|_1 \tag{10} \\ \text{subject to} \quad & \mathbf{B}^T \mathbf{k}(\cdot, \mathbf{x}) - \mathbf{B}^T \mathbf{K} \alpha \leq \varepsilon \\ & \mathbf{B}^T \mathbf{K} \alpha - \mathbf{B}^T \mathbf{k}(\cdot, \mathbf{x}) \leq \varepsilon \end{aligned}$$

which is a linear programming problem. Typically, solving a linear programming is more easy than solving quadratically constrained ℓ_1 minimization problem.

3 KSRC ensemble

In this section, we propose the kernel sparse representation-based classifier ensemble (KSRCE), and discuss the diversity and the combination rules for KSRCE.

Also consider a c -class classification task. Suppose that the training sample set is $\{\mathbf{x}_i, y_i\}_{i=1}^n$, and \mathbf{x} is an arbitrary test sample. Assume that there are T kernel sparse representation-based classifiers (KSRCs). Let $r_j^t(\mathbf{x})$ and \hat{y}^t denote the reconstruction error of the j th class and the label obtained by the t -th KSRC for \mathbf{x} , respectively. We try to assign a label to \mathbf{x} according to outputs $r_j^t(\mathbf{x})$ or \hat{y}^t , $t = 1, 2, \dots, T$ of the T KSRCs. To get good classification performance, we need to design the diversity and the combination rules for KSRCE, respectively.

3.1 Diversity

Diversity can ensure that all the individual classifiers make uncorrelated errors. If each classifier makes the identical errors, these errors will propagate to the ensemble and thereby no improvement can be achieved in combining multiple classifiers. Thus, it is important to assure the diversity in classifier ensembles. Generally, the diversity in classifier ensemble is generated by using different individual classifiers, or different training samples for individual classifiers as mentioned in Section 1.

Here, we only focus on the ensemble of KSRCs and give the schemes of generating the diversity for KSRCE in the following.

1. Random matrix: Random projection is a very efficient and simple way for dimensionality reduction. In this case, \mathbf{B} in (6) and (10) is a random matrix. Different \mathbf{B} would lead to different representation and different classification performance.
2. Random selection of kernel functions and kernel parameters: There are some common used kernels, such as linear kernel, radial basis function (RBF) kernel and wavelet kernels [21, 22]. Obviously, different kernel leads to different kernel Gram matrix \mathbf{K} and $\mathbf{k}(\cdot, \mathbf{x})$ in (6) and (10), which would result in different classification performance. Except for the linear kernel, other kernels have their own parameters. The linear kernel has the form

$$k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{x}' \quad (11)$$

and RBF kernels can be expressed as

$$k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|_2^2) \quad (12)$$

where $\gamma > 0$ is the parameter for RBF kernels. When using the RBF kernel, we must give the value of γ . Of course, different γ also results in different classification performance.

3. Random sample of kernel matrix: Each column of the kernel Gram matrix \mathbf{K} consists of all training samples. Thus, \mathbf{K} is fixed when setting the kernel parameters. We can randomly select a part of training samples to construct a new kernel matrix $\bar{\mathbf{K}} \in \mathbb{R}^{n' \times n}$, where n' is the number of selected training samples. We would get different representation and classification results by replacing \mathbf{K} with $\bar{\mathbf{K}}$. In this case, \mathbf{B} must be replaced by $\bar{\mathbf{B}} \in \mathbb{R}^{n' \times n'}$. Moreover, \mathbf{B} could be a random matrix or just be an identity matrix.

These three schemes can be used alone or used together.

3.2 Combination rules

For an individual KSRC, we can obtain the reconstruction errors for a test sample and assign a label to it according to these errors. In the combination step, we have two ways to combine the outputs of multiple KSRCs. One way is that we firstly combine the reconstruction errors in terms of some combination rules and then assign a label to the test sample. The other way is to use the simple voting (SV) rule when getting the labels from multiple KSRCs for the test samples. In the following, the two ways are described.

1. Combining reconstruction errors: Given the T reconstruction errors r_j^t , it requires to combine them to obtain a robust estimate of the reconstruction r_j for the j -th class using some rules. In [18], there proposes a rule, called the mean rule. Namely,

$$r_j = \frac{1}{T} \sum_{t=1}^T r_j^t \quad (13)$$

Here, we propose two other rules for combining these reconstruction errors: the max rule and the min rule. The two combination rules are developed from the mean rule (13) and to approximate the mean by the maximum and the minimum of the reconstruction error, respectively. The max rule can be expressed as

$$r_j = \max_{t=1, \dots, T} r_j^t \quad (14)$$

and the min rule can be described as

$$r_j = \min_{t=1, \dots, T} r_j^t \quad (15)$$

From the estimated reconstruction errors r_j , we can assign an estimated label \hat{y} according to

$$\hat{y} = \arg \min_{j=1, \dots, c} r_j \quad (16)$$

2. Combining labels: Given the T estimated labels \hat{y}^t , we can use the simple vote rule to combine them to obtain an estimated label \hat{y} . Here, the simple vote can be described as:

$$\hat{y} = \arg \max_{j=1, \dots, c} \sum_{t=1}^T \theta(\hat{y}^t - j) \quad (17)$$

where $\theta(\hat{y}^t - j)$ is a function defined by

$$\theta(\hat{y}^t - j) = \begin{cases} 1, & \text{if } \hat{y}^t = j \\ 0, & \text{otherwise} \end{cases} \quad (18)$$

3.3 Algorithm of KSRCE

The complete procedure of KSRCE is shown in Algorithm 1.

Algorithm 1 Kernel sparse representation-based classifier ensemble

1. **Input:** A set of training samples $\{\mathbf{x}_i, y_i\}_{i=1}^n$, where $\mathbf{x}_i \in \mathbb{R}^m$, $y_i \in \{1, 2, \dots, c\}$, a test sample $\mathbf{x} \in \mathbb{R}^m$, the ensemble size T and an optional error tolerance $\epsilon > 0$.
 2. Generate T different kernel sparse representation-based classifiers by using the diversity schemes mentioned in Section 3.1.
 3. Combine the outputs of the T KSRCs by using the combination rules mentioned in Section 3.2.
 4. **Output:** The estimated label \hat{y} for \mathbf{x} .
-

4 Simulation

This section gives numerical experimental results of KSRCE on three face data sets and compare KSRCE with SRC, SRCE (SRC ensemble) and KSRC. The LP problem for both KSRC and SRC is solved by exploiting the GLPK software package [13].

All numerical experiments are performed on the personal computer with a 2.93 GHz Intel(R) Core(T)2 Duo CPU and 2 G bytes of memory. This computer runs on Windows 7, with Matlab 7.01 and VC++ 6.0 compiler installed.

4.1 Experiments on ORL data set

Firstly, we perform experiments on the ORL face database [16] and take into account the ensemble size and diversity schemes. The ORL face data set has 10 different images for each subject and consists of 40 distinct subjects. Figure 1 shows 6 images of the same subject. All the subjects are in up-right, frontal position (with tolerance for some side movement). The size of each face image is 112×92 , and the resulting standardized input vectors are of dimensionality 10,304. The feature values of all samples are normalized by 255. Namely, features take values from the interval $[0, 1]$. The number of images for both training and test is 200.

Experiment on ensemble size Now, we analysis the affection of ensemble size on the classification performance of KSRCE. The ensemble size is varied from 2 to 10. Only the random matrix scheme is adopted to generating the diversity for KSRCE with the linear kernel. The reduced dimensionality range is 10–140 in [23]. Here, we

Fig. 1 Images of a subject from the ORL database



select three typical subspaces from this range, or 20, 60 and 120 when performing the dimensionality reduction.

Figure 2 shows the classification error on the test set for the three subspaces. “Ensemble-Vote” means KSRCE using the simple voting rule, “Ensemble-Max”, “Ensemble-Min” and “Ensemble-Mean” denote KSRCE using the max rule, the min rule and the mean rule, respectively. “Single-Min” and “Single-Average” denote that the classifier with the minimal error among and the average error over multiple KSRCs, respectively.

In Fig. 2a, the classification performance of KSRCE is obviously much better than the single best KSRC (or “Single-Min”) except for the simple voting rule when ensembling two KSRCs. In the simple voting rule, the diversity of any two classifiers only gives confusion. Thus, it could not improve performance by employing the simple voting rule when combining two classifiers. Figure 2b and c indicate that KSRCE always outperforms the average performance of multiple individual KSRCs (or “Single-Average”), and mostly achieves the better performance than the single best KSRC. Although the effect of ensemble on the classification performance decreases as the increasing of dimensionality, KSRCE can still improve the classification performance of KSRC since we always take the average performance of KSRC as the final result instead of the minimal ones. In addition, the larger the ensemble

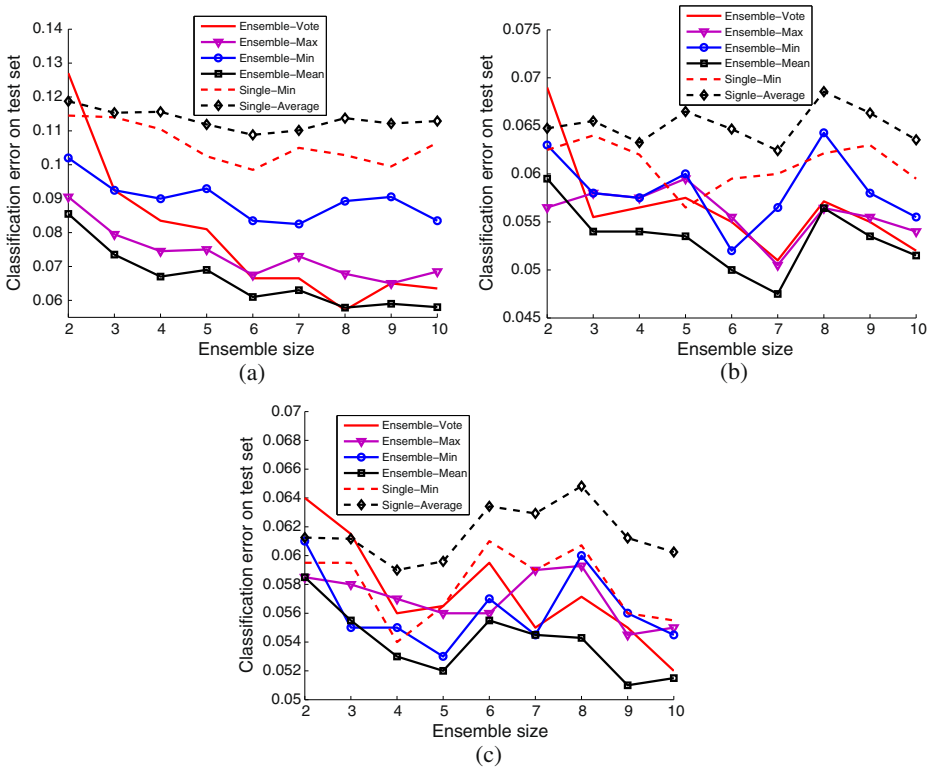


Fig. 2 Classification error vs. ensemble size. **a** 20 dimensionality, **b** 60 dimensionality, and **c** 120 dimensionality

size is, the better the ensemble performance in the case of low dimensionality. With the increasing of dimensionality, the ensemble performance is not proportion to the ensemble size, but dependent on the performance of individual KSRCs.

In summary, the ensemble size is important only in the relative low subspace. In the following experiments, we take the ensemble size of five as in [18].

Experiment on diversity schemes Here, we compares different diversity schemes for KSRCE on the ORL face data set. Let the ensemble size be five. We design four methods for generating the diversity for KSRCE according to the three diversity schemes given in Section 3.1.

- Random matrix: We randomly generate \mathbf{B} five times and get five individual KSRCs.
- Random sample of kernel matrix: The linear kernel is used. We randomly select 50 % training samples to construct kernel matrix, repeat it five times and get five individual KSRCs.
- Random matrix+random sample: We integrate the two above methods. In each KSRC, both different random matrix and different (linear) kernel matrix are generated.
- Random matrix+random sample+ random kernel selection: In each KSRC, different random matrix and different kernel matrix and different kernel are adopted. Here, the linear kernel and the RBF kernel are considered. In [23], the parameter γ of RBF kernel is set by the median value of $1/(\|\mathbf{x}_i - \bar{\mathbf{x}}\|^2)$, $i = 1, \dots, n$, where $\bar{\mathbf{x}}$ is the mean of all training samples. Let $\gamma = \omega \times 1/(\|\mathbf{x}_i - \bar{\mathbf{x}}\|^2)$, where ω is a random number chosen from a uniform distribution on the interval [1, 3].

Figure 3 gives the classification error on the test set for the four diversity schemes. The dimensionality of subspace is 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 120, and 140. Figure 3a–d appear similar. The classification performance of six classifiers is improved with the increasing of the dimensionality. But the improvement is not evident when the dimensionality of subspace is larger than or equal to 60. In Figs. 3a–c, we can see that KSRCE with the simple voting rule, the max rule and the mean rule outperforms both the average and minimal of KSRCs in the case of the lower dimensionality, say 20. But we can not clearly distinguish the six classifiers in the case of higher dimensionality, say 120. In Fig. 3d, we can still identify that the ensemble performance of KSRCs is better than the performance of the single KSRC. In a nutshell, the diversity scheme, “Random matrix+random sample+random kernel selection”, is the best choice for KSRCE. In the following, we use this scheme to generate diversity for KSRCE.

4.2 Experiments on three face data sets

UMIST face database [7] and Extended Yale B database [11] are also considered here. The original features of each face image is obtained by stacking its columns.

- UMIST: The UMIST face database is a multi-view database which consists of 574 cropped gray-scale images of 20 subjects, each covering a wide range of poses from profile to frontal views as well as race, gender and appearance. Each image in the database is resized into 112×92 . Figure 4a depicts some sample images of

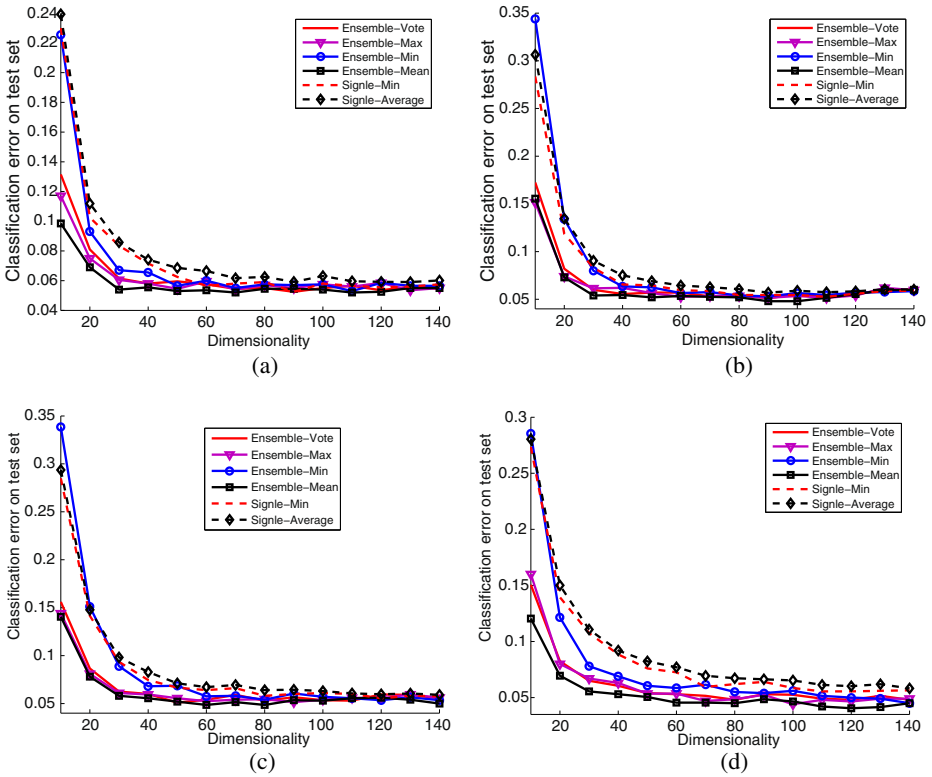
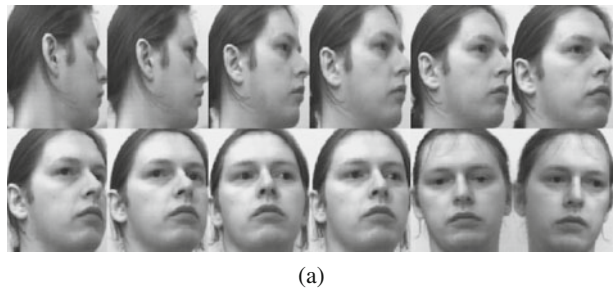


Fig. 3 Classification error vs. dimensionality. **a** Random matrix, **b** random sample of kernel matrix, **c** random matrix+random sample, and **d** random matrix+random sample+random kernel selection

Fig. 4 Face data. **a** Images of a subject from the UMIST database, and **b** images of a subject from the Extended Yale B database



- a typical subset in the UMIST database. The total number of the training samples is 290, and that of the test samples is 284.
- **Extended Yale B:** It consists of 2,414 frontal-face images of 38 subjects which are manually aligned, cropped, and then re-sized to 168×192 images [11]. These images were captured under various laboratory-controlled lighting conditions. Figure 4b shows some sample images of a typical subset in the Extended Yale B database. The total number of the training samples could be 1,207, and that of the test samples is also 1,207. Each sample has 32,256 features.

This section compares KSRCE to SRC, SRCE and KSRC on three face data sets, including the ORL, the UMIST and the Extended Yale B face data sets. The dimensionality of subspace in the ORL data set is 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 120, and 140. For both the UMIST and Extended Yale B data sets, the dimensionality of subspace is 20, 40, 60, 80, 100, 120, 140, 160, 180 and 200.

In KSRCE and SRCE, the ensemble size is five. From Figs. 2 and 3, we can see that the mean rule is relative stable. Thus, the mean rule is taken as the combination rule for SRCE and KSRCE. The diversity scheme for SRCE is the random matrix one, which is the only one SRCE can use. While for KSRCE, the scheme of random matrix+random sample+random kernel selection is used, whose setting is the same as that in Section 4.1. In KSRC, the linear kernel or the RBF kernel is used. We take

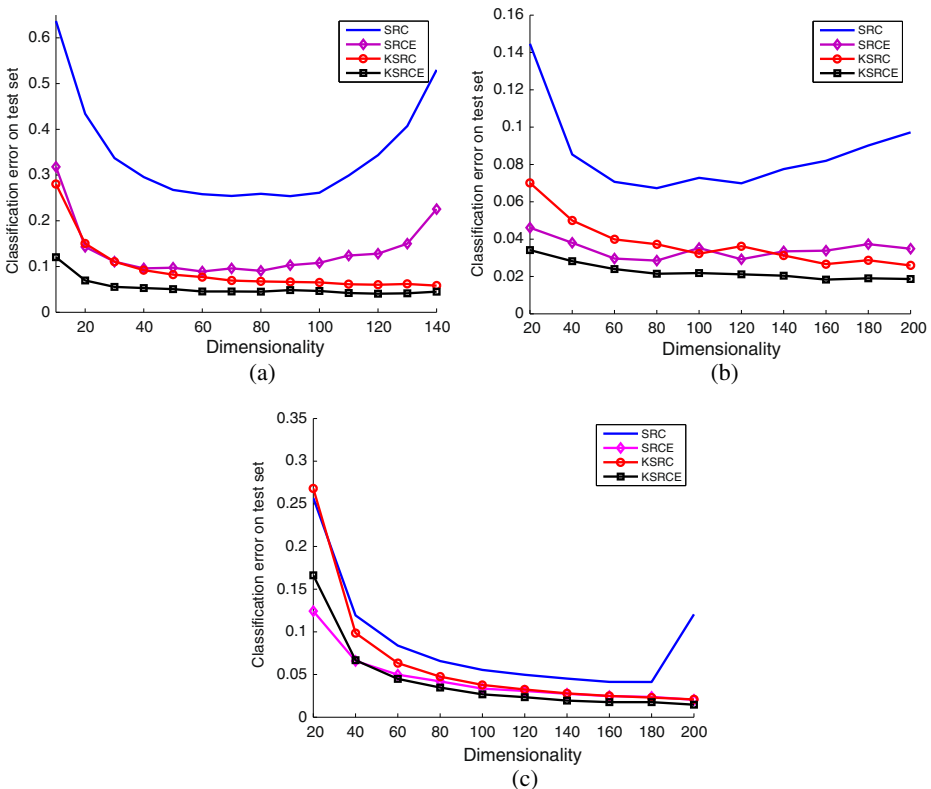


Fig. 5 Classification error vs. dimensionality. **a** ORL, **b** UMIST, and **c** Extended Yale B

the average classification error on the test set over five SRCs and KSRCs as the final results of SRC and KSRC, respectively.

Figure 5 shows the classification errors obtained by four methods. Inspection on Fig. 5a and b indicate that the performance of KSRCE is the best among the four methods. SRCE is much better than SRC, which further supports that the classifier ensemble is effective. In addition, KSRC is compared with SRCE. From Fig. 5c, we can see that KSRCE outperforms SRCE when the dimensionality is larger than or equal to 40. KSRC is always better than SRC except that the case of $d = 20$.

Figure 6 gives curves of the CPU running time vs. dimensionality. All curves show that the CPU running time greatly increases when increasing the dimensionality. Moreover, ensemble algorithms require more running time compared to single classifiers. For both ORL and UMIST, the running time of KSCRE is close to that of SRCE. But, KSRCE is much better than SRCE on the Extended Yale B set. In fact, the CPU running time depends on the training sample number, the test sample number, the sample dimensionality and others. On the Extended Yale B set, KSRCE (or KSRC) runs faster than SRCE (or SRC) because SRCE (or SRC) performs dimensionality reduction from very high dimensionality (32,256), and KSRCE (or KSRC) only from 1,207 which is the number of training samples.

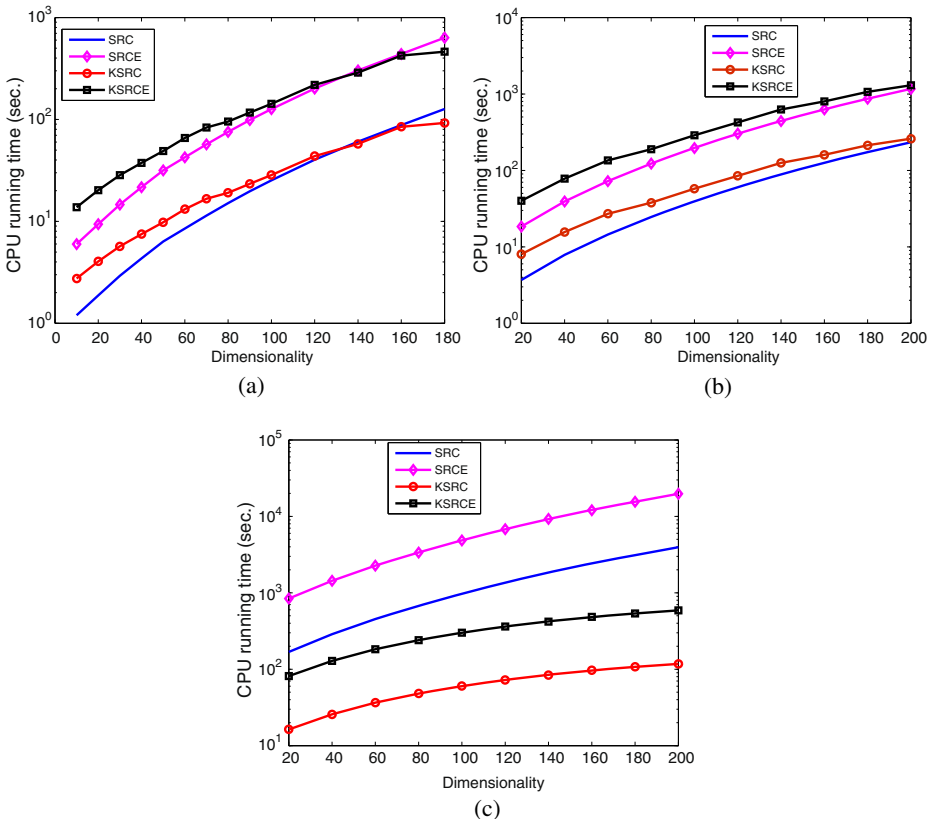


Fig. 6 Classification error vs. dimensionality. **a** ORL, **b** UMIST, and **c** Extended Yale B

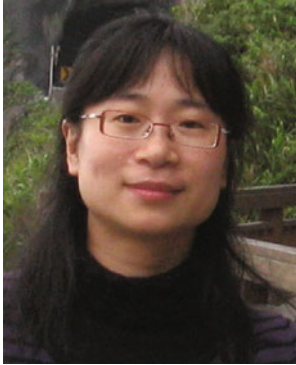
5 Conclusions

In this paper, we develop the kernel sparse representation-based classifier ensemble. We design the diversity schemes and describe some combination rules for KSRCE. Generally, SRC and KSRC are cast into a quadratically constrained ℓ_1 minimization problem. We recast KSRC into a linear programming (LP) problem, which is more easy to solve than quadratically constrained ℓ_1 minimization problems. Experimental results on three face data sets show that KSRCE is mostly much better than SRC, SRCE and KSRC. In the case of relative low dimensionality, the effect of KSRCE is very obvious. In addition, the mixture diversity scheme performs well.

References

- Breiman, L.: Bagging predictors. *Machine Learning* **24**, 123–140 (1996)
- Candès, E., Romberg, J.: ℓ_1 -magic: recovery of sparse signals via convex programming (2005). <http://www.acm.caltech.edu/l1magic/>. Accessed 10 March 2010
- Chen, S., Donoho, D., Saunders, M.: Atomic decomposition by basis pursuit. *SIAM Rev.* **43**(1), 129–159 (2001)
- Duda, R., Hart, P., Stork, D.: *Pattern Classification*, 2nd edn. John Wiley & Sons (2000)
- Freund, Y., Shapire, R.: Experiments with a new boosting algorithm. In: *Proceedings of the Thirteenth International Conference on Machine Learning*, pp. 148–156. Morgan Kaufmann, Bary, Italy (1996)
- Fumera, G., Roli, F.: A theoretical and experimental analysis of linear combiners for multiple classifier systems. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(6), 942–956 (2005)
- Graham, D.B., Allinson, N.M.: Characterizing virtual Eigensignatures for general purpose face recognition. In: *Face Recognition: from Theory to Applications*, NATO ASI Series F, Computer and Systems Sciences, vol. 163, pp. 446–456 (1998)
- Ho, T.K., Hull, J.J., Srihari, S.N.: Decision combination in multiple classifier systems. *IEEE Trans. Pattern Anal. Mach. Intell.* **16**(1), 66–75 (1994)
- Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On combining classifiers. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**(3), 226–239 (1998)
- Kuncheva, L.I.: *Combining Pattern Classifiers: Methods and Algorithms*. John Wiley & Sons, Inc., Hoboken, N.J. (2004)
- Lee, K.C., Ho, J., Kriegman, D.: Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(5), 684–698 (2005)
- Li, S.Z.: Face recognition based on nearest linear combinations. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 839–844. IEEE Computer Society, Washington, DC, USA (1998)
- Makhorin, A.: Introduction to GLPK. <http://www.gnu.org/software/glpk/glpk.html> (2004). Accessed 5 May 2009
- Mika, S., Rätsch, G., Weston, J., Schölkopf, B., Müller, K.R.: Fisher discriminant analysis with kernels. In: *IEEE International Workshop on Neural Networks for Signal Processing IX*, Madison, USA, pp. 41–48 (1999)
- Roli, F., Kittler, J., Windeatt, T. (eds.): *Multiple Classifier Systems*. Lecture Notes in Computer Science, vol. 3077. Springer (2004)
- Samaria, F.S., Harter, A.C.: Parameterisation of a stochastic model for human face identification. In: *Proceedings of the 2nd IEEE International Workshop on Applications of Computer Vision*, Sarasota Florida, pp. 138–142 (1994)
- Schölkopf, B., Smola, A.J., Müller, K.R.: Nonlinear component analysis as a kernel Eigenvalue problem. *Neural Comput.* **10**(5), 1299–1319 (1998)
- Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(2), 210–226 (2009)
- Yang, A.Y., Wright, J., Ma, Y., Sastry, S.S.: Feature selection in face recognition: a sparse representation perspective. Tech. Rep. UCB/EECS-2007-99. EECS Department, University of California, Berkeley (2007). <http://www.eecs.berkeley.edu/Pubs/TechRpts/2007/EECS-2007-99.html>. Accessed 10 May 2008

20. Zhang, L., Zhou, W.D.: Sparse ensembles using weighted combination methods based on linear programming. *Pattern Recogn.* **44**(1), 97–106 (2011)
21. Zhang, L., Zhou, W.D., Jiao, L.C.: Wavelet support vector machine. *IEEE Trans. Syst. Man Cybern., Part B* **34**(1), 34–39 (2004)
22. Zhang, L., Zhou, W.D., Jiao, L.C.: Support vector machines based on the orthogonal projection kernel of father wavelet. *Int. J. Comput. Intell. Appl.* **5**(3), 283–303 (2005)
23. Zhang, L., Zhou, W.D., Chang, P.C., Liu, J., Yan, Z., Wang, T., Li, F.Z.: Kernel sparse representation-based classifier. *IEEE Trans. Signal Process.* **60**(4), 1684–1695 (2012)



Li Zhang (M'08) received the B.S. degree in 1997 and the Ph.D. degree in 2002 in electronic engineering from Xidian University, Xi'an, China. From 2003 to 2005, she was a postdoctor at the Institute of Automation of Shanghai Jiao Tong University, Shanghai, China. From 2005 to 2010, she worked as an associate professor at Xidian University, Xi'an, China. Now she is a full professor of Soochow University in Suzhou, China. Her research interests have been in the areas of machine learning, pattern recognition, neural networks and intelligent information processing.



Wei-Da Zhou received the B.S. in 1996 and the Ph.D. degree in 2003 in electronic engineering from Xidian University, Xi'an, China. He was an associate professor at the School of Electronic Engineering at Xidian University, Xi'an, China from 2006 to 2009. From 2009, he works for AI Speech Ltd., Suzhou 215123, Jiangsu, China. His research interests include machine learning, learning theory and intelligent information processing.



Fan-Zhang Li received the B.S. degree in computer science and technology from the University of Science and Technology of China, Hefei, Anhui, in 1995. From 1996 to 2000, he worked in Yunnan University. He is a full professor of Soochow University since 2000, Suzhou, China. He has published three books on dynamic fuzzy logic and one book on Lie group machine learning. His research interests include artificial intelligence, machine learning and dynamic fuzzy logic.