Kernel Collaborative Representation With Tikhonov Regularization for Hyperspectral Image Classification

Wei Li, *Member, IEEE*, Qian Du, *Senior Member, IEEE*, and Mingming Xiong

***Abstract*—In this letter, kernel collaborative representation with Tikhonov regularization (KCRT) is proposed for hyperspectral image classification. The original data is projected into a high-dimensional kernel space by using a nonlinear mapping function to improve the class separability. Moreover, spatial information at neighboring locations is incorporated in the kernel space. Experimental results on two hyperspectral data prove that our proposed technique outperforms the traditional support vector machines with composite kernels and other state-of-the-art classifiers, such as kernel sparse representation classifier and kernel collaborative representation classifier.**

***Index Terms*—Hyperspectral classification, kernel methods, nearest regularized subspace (NRS), sparse representation.**

I. Introduction

KERNEL-based methods, such as support vector machines (SVM) [1], [2], have been proved to be powerful in hyperspectral image classification. The central idea behind kernel-based methods is to map the data from its original input space into a high-dimensional kernel-induced feature space where the data may become more separable. For instance, an SVM seeks to learn an optimal decision hyperplane that best separates the training samples in the kernel-induced feature space and to define the model for classification task by exploiting the concept of margin maximization. Spatial extensions to kernel methods, such as the methods exploiting the properties of Mercer's conditions to construct a family of composite kernels for the combination of both spectral and spatial information, have been recently presented. In [3], a composite kernel (CK) has been designed for the combination of nonlinear transformations of spectral and contextual signatures for SVM, and the resulting classifier is referred to as SVM-CK.

Sparse representation-based classification (SRC) [4], originally developed for face recognition, has attracted considerable attention in the past few years. The essence of a SRC classifier is built on the concept that a pixel can be represented as a linear combination of labeled samples via the sparse regularization techniques, such as the ℓ_0-norm regularization and the ℓ_1-norm regularization. It does not need a training process (but does require labeled data), which is different from the conventional training-test fashion (e.g., as in SVM). A test sample to be classified is sparsely approximated by the training data, and it is directly assigned to the class whose labeled samples provide the smallest representation error. In [5], SRC was applied to hyperspectral image classification, and demonstrated good performance. Furthermore, kernel version of SRC classifier (KSRC) was developed in [6] and [7]. In [8], spatial-spectral kernel sparse representation was introduced to improve the classification performance.

In fact, [9] argued that it was the collaborative representation (CR) rather than sparse representation playing the essential role for classification. In [10], the kernel version of a collaborative representation-based classification (CRC) was proposed and denoted as KCRC. A nonlinear nearest subspace (NNS) classifier was proposed for high-dimensional face data [11]. The essential difference between NNS [and its linear version called nearest subspace (NS)] and KCRC (and its linear version CRC) is that the former employs within-class training data for collaborative representation (also called *pre-partitioning*) while the latter uses all the training data from different classes simultaneously (also called *post-partitioning*).

The key difference between SRC and CRC (and NS) implementations is that the former employs an ℓ_0 or ℓ_1-norm regularization while the latter employs an ℓ_2-norm regularization; thus, the latter can have a closed-form solution, resulting in much lower computational cost. In [12], the NS was extended to the nearest regularized subspace (NRS) classifier, where an ℓ_2-norm penalty has been designed in the style of a distance-weighted Tikhonov regularization to measure the similarity between the pixel under test and the within-class labeled samples; consequently, collaborative performance can be improved by considering within-class variations.

As mentioned before, when classes are not linearly separable, kernel methods [3], [6], [10] can project the data into a nonlinear feature space where class separability can be improved. On the other hand, it is highly probable that two spatially adjacent pixels belong to the same class; as a matter of fact, spatial information has been verified to be helpful for high-spatial-resolution hyperspectral image classification [3], [8]. In this letter, we first extend the idea of Tikhonov regularization to the

Manuscript received February 9, 2014; revised April 14, 2014; accepted May 13, 2014. This work was supported by the National Natural Science Foundation of China under Grant NSFC-61302164.

W. Li and M. Xiong are with the College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China (e-mail: liwei089@ieee.org).

Q. Du is with the Department of Electrical and Computer Engineering, Mississippi State University, Mississippi State, MS 39762 USA (e-mail: du@ece.msstate.edu).

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

Digital Object Identifier 10.1109/LGRS.2014.2325978

original CRC using all the labeled samples, and the resulting method is denoted as CRT. And then, a nonlinear version of CRT is introduced by incorporating the kernel trick, which is referred to as KCRT. The extended CRT classifier only employs the spectral signatures while ignoring the spatial information at neighboring locations. Thus, we further consider the use of a spatial-spectral kernel, capturing both spatial and spectral features, for the proposed KCRT, such as a KCRT version with a composite kernel which is called KCRT-CK. We believe the findings of this letter, especially the one that KCRT tends to outperform KSRC and KCRC, is important, given that KSRC and KCRC are widely accepted as the state-of-the-art classifiers in remote sensing applications.

II. Related Work

A. Nearest Regularized Subspace

Consider a data set with training samples $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{n}$ in $\mathbb{R}^d$ (d is the dimensionality) and class labels $\omega_i \in \{1, 2, \ldots, C\}$, where C is the number of classes, and n is the total number of training samples. Let n_l be the number of available training samples for the lth class, and $\sum_{l=1}^{C} n_l = n$.

An approximation of test sample $\mathbf{y}$ is represented via a linear combination of available training samples per class, $\mathbf{X}_l$. That is, for each class, only from the training samples particular to class l, the class-specific approximation, $\widehat{\mathbf{y}}_l$, is calculated as $\widehat{\mathbf{y}}_l = \mathbf{X}_l\boldsymbol{\alpha}_l$, where $\mathbf{X}_l$ is of size $d \times n_l$, and $\boldsymbol{\alpha}_l$ is a $n_l \times 1$ vector of weighting coefficients. The weight vector $\boldsymbol{\alpha}_l$ for the linear combination is solved by an ℓ_2-norm regularization

$$\boldsymbol{\alpha}_l = \arg\min_{\boldsymbol{\alpha}_l^*} \|\mathbf{y} - \mathbf{X}_l\boldsymbol{\alpha}_l^*\|_2^2 + \lambda \|\boldsymbol{\Gamma}_{l,\mathbf{y}}\boldsymbol{\alpha}_l^*\|_2^2 \tag{1}$$

where $\boldsymbol{\Gamma}_{l,\mathbf{y}}$ is a biasing Tikhonov matrix specific to each class l as well as the current test sample $\mathbf{y}$, and λ is a global regularization parameter which balances the minimization between the residual part and the regularization term. Note that $\boldsymbol{\alpha}_l^*$ is a various representation of $\boldsymbol{\alpha}_l$ with size of $n_l \times 1$. Specifically, the regularization term is designed in the form of

$$\boldsymbol{\Gamma}_{l,\mathbf{y}} = \begin{bmatrix} \|\mathbf{y} - \mathbf{x}_{l,1}\|_2 & & 0 \\ & \ddots & \\ 0 & & \|\mathbf{y} - \mathbf{x}_{l,n_l}\|_2 \end{bmatrix} \tag{2}$$

where $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n_l}$ are the columns of matrix $\mathbf{X}_l$ for the lth class. And then, the weight vector $\boldsymbol{\alpha}_l$ can be recovered in a closed-form solution

$$\boldsymbol{\alpha}_l = \left(\mathbf{X}_l^T\mathbf{X}_l + \lambda^2\boldsymbol{\Gamma}_{l,\mathbf{y}}^T\boldsymbol{\Gamma}_{l,\mathbf{y}}\right)^{-1}\mathbf{X}_l^T\mathbf{y}. \tag{3}$$

Once obtaining the weight vector, class label of the test sample is then determined according to the class which minimizes the residual between $\widehat{\mathbf{y}}_l$ and $\mathbf{y}$. That is

$$r_l(\mathbf{y}) = \|\widehat{\mathbf{y}}_l - \mathbf{y}\|_2 = \|\mathbf{X}_l\boldsymbol{\alpha}_l - \mathbf{y}\|_2 \tag{4}$$

and $\text{class}(\mathbf{y}) = \arg\min_{l=1,\ldots,C} r_l(\mathbf{y})$.

B. Kernel Trick

Appropriate selection of a kernel function is able to accurately reflect the similarity among samples; however, not all metric distances can be applied in kernel methods. In fact, valid kernels are only those satisfying the Mercer's conditions [3], requiring to be positive semidefinite. For a given nonlinear mapping function Φ, the Mercer kernel function $k(\cdot, \cdot)$ can be represented as

$$k(\mathbf{x}, \mathbf{x}') = \Phi(\mathbf{x})^T\Phi(\mathbf{x}'). \tag{5}$$

Commonly used kernels include linear kernel $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T\mathbf{x}'$, the t-degree polynomial kernel $k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^T\mathbf{x}' + 1)^t (t \in \mathbb{Z}^+)$, and the Gaussian radial basis function (RBF) kernel $k(\mathbf{x}, \mathbf{x}') = \exp(-\gamma\|\mathbf{x} - \mathbf{x}'\|_2^2)$ ($\gamma > 0$ is the parameter of RBF kernel).

III. Kernel Collaborative Representation With Tikhonov Regularization

The essential difference between CRT and NRS is *post-partitioning* and *pre-partitioning* for the training samples. The *post-partitioning* actually has been applied in both SRC [4] and CRC [9]. The *post-partitioning* indicates an approximation of the test samples $\mathbf{y}$ is calculated via a linear combination of *all* available training samples. That is, using all the samples in matrix $\mathbf{X} \in \mathbb{R}^{d\times n}$, the weight vector $\boldsymbol{\alpha} \in \mathbb{R}^{n\times 1}$ is supposed to obtain so that $\mathbf{X}\boldsymbol{\alpha}$ is close to $\mathbf{y}$, and then $\mathbf{X}$ and $\boldsymbol{\alpha}$ are separated into l different class-specific sub-dictionaries according to the given class labels of the training samples. In *pre-partitioning*, the training data is first partitioned into $\mathbf{X}_l$, as stated for the original NRS in Section II-A. In this letter, we provide the comparison of CRT and NRS as well as KCRT and the kernel version of NRS (denoted as KNRS) using two partitionings in the next section.

Algorithm 1 Proposed KCRT-CK Classifier

```
Input: Training data $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{n}$, class labels $\omega_i$, test sample $\mathbf{y} \in \mathbb{R}^d$, and regularization parameter $\lambda$.
Step 1: Select a Mercer kernel $k(.,.)$ and its parameters;
Step 2: Concatenate the spectral and spatial features;
Step 3: Calculate biasing Tikhonov matrix $\boldsymbol{\Gamma}_{\Phi(\mathbf{y})}$ according to (7);
Step 4: Obtain weight vector $\boldsymbol{\alpha}'$ according to (8);
Step 5: Decide class label class$(\mathbf{y})$ according to (9).
Output: class$(\mathbf{y})$.
```

As for how to choose appropriate kernels, we employ the previously mentioned CK[1] [3], [8]. The reason is that CK takes advantages of the properties of Mercer's conditions, and simultaneously utilize spatial and spectral features, providing rich feature information. Two of common composite kernels

[1] CK is not the only method for providing the spatial-spectral kernel representation. A number of other methods can be found in the literatures.

TABLE I
CLASSIFICATION ACCURACY (%) FOR THE INDIAN PINES DATA SET

Class	# samples		Classification algorithms										
	Train	Test	SVM	SVM-CK [3]	SRC [4]	KSRC [7]	KSRC-CK [8]	CRC [9]	KCRC [10]	KCRC-CK	CRT	KCRT	KCRT-CK
1	6	48	81.04	95.74	87.04	70.37	94.44	65.19	66.67	94.44	79.63	77.78	**98.15**
2	144	1290	87.10	93.65	80.96	83.54	92.19	68.13	85.50	95.63	79.66	84.38	**98.12**
3	84	750	72.94	89.33	65.59	75.30	90.65	34.53	65.59	94.95	67.31	74.94	**99.76**
4	24	210	61.53	82.65	54.27	65.81	75.64	60.84	61.54	84.87	61.54	61.54	**95.30**
5	50	447	95.77	93.65	95.17	93.36	95.37	90.54	94.37	93.58	95.17	93.76	**96.98**
6	75	672	96.79	99.20	95.31	97.46	99.60	93.71	98.53	**99.73**	95.58	98.26	99.46
7	3	23	70.92	84.62	61.54	84.62	84.62	65.38	73.08	88.39	65.38	80.27	**96.15**
8	49	440	97.96	97.96	97.14	97.34	99.59	88.55	99.18	**100**	95.71	99.18	**100**
9	2	18	45.00	**100**	20.00	35.00	90.00	45.00	20.00	**100**	35.00	45.00	**100**
10	97	871	80.27	85.64	53.20	83.99	86.05	24.17	77.07	85.87	67.36	81.10	**96.28**
11	247	2221	86.47	91.47	88.09	87.12	93.19	91.25	88.13	93.85	90.96	87.32	**98.10**
12	62	552	85.83	88.76	79.48	77.20	87.13	29.09	79.97	91.15	68.40	82.41	**99.02**
13	22	190	99.06	99.53	95.75	97.64	99.53	84.43	99.53	99.53	95.28	99.53	**99.53**
14	130	1164	90.73	95.36	92.81	91.34	96.45	94.28	94.36	94.30	90.34	92.81	**99.92**
15	38	342	64.21	87.47	76.32	71.84	77.89	53.42	65.79	85.68	78.42	66.58	**99.47**
16	10	85	95.79	**100**	95.79	92.63	85.26	73.16	77.89	67.89	94.74	81.05	67.37
Overall Accuracy (%)			84.45	94.15	82.23	85.14	92.18	70.35	85.02	94.97	83.87	86.09	**98.22**
Kappa Coefficient (κ)			0.8112	0.9230	0.7956	0.8306	0.9107	0.6507	0.8308	0.9296	0.8100	0.8411	**0.9797**

introduced in [3] are the "stacked" kernel and the "weighted summation" kernel, which have similar performance. We choose the former but not the latter due to an additional balance parameter required for the latter. Assume $\mathbf{x}^w \equiv \mathbf{x}$ represents the spectral content of a sample, $\mathbf{x}^s$ represents spatial feature of a sample by applying some feature extraction (i.e., a mean value *per* spectral band) within its surrounding area (depends on the choice of the spatial window size). The spectral and spatial features are concatenated into a new representation $\mathbf{x} \equiv \{\mathbf{x}^w; \mathbf{x}^s\}$. The "stacked" kernel matrix can be then calculated using (5).

In the chosen kernel-induced feature space, we can linearly represent a test sample in terms of all available training samples. The new weight vector $\boldsymbol{\alpha}'$ for the linear combination is still solved by an ℓ_2-norm regularization

$$\boldsymbol{\alpha}' = \arg\min_{\boldsymbol{\alpha}^*} \|\Phi(\mathbf{y}) - \boldsymbol{\Phi}\boldsymbol{\alpha}^*\|_2^2 + \lambda \left\|\boldsymbol{\Gamma}_{\Phi(\mathbf{y})}\boldsymbol{\alpha}^*\right\|_2^2 \tag{6}$$

where the mapping function Φ maps the test sample to the kernel-induced feature space: $\mathbf{y} \rightarrow \Phi(\mathbf{y}) \in \mathbb{R}^{D\times 1}$ ($D \gg d$ is the dimension of kernel feature space) and $\boldsymbol{\Phi} = [\Phi(\mathbf{x}_1), \Phi(\mathbf{x}_2), \ldots, \Phi(\mathbf{x}_n)] \in \mathbb{R}^{D\times n}$. The new biasing Tikhonov matrix $\boldsymbol{\Gamma}_{\Phi(\mathbf{y})}$ then has the form of

$$\boldsymbol{\Gamma}_{\Phi(\mathbf{y})} = \begin{bmatrix} \|\Phi(\mathbf{y}) - \Phi(\mathbf{x}_1)\|_2 & & 0 \\ & \ddots & \\ 0 & & \|\Phi(\mathbf{y}) - \Phi(\mathbf{x}_n)\|_2 \end{bmatrix} \tag{7}$$

where $\|\Phi(\mathbf{y}) - \Phi(\mathbf{x}_i)\|_2 = [k(\mathbf{y},\mathbf{y}) + k(\mathbf{x}_i,\mathbf{x}_i) - 2k(\mathbf{y},\mathbf{x}_i)]^{1/2}$, $i = 1, 2, \ldots, n$. After constituting $\boldsymbol{\Gamma}_{\Phi(\mathbf{y})}$, the weight vector $\boldsymbol{\alpha}'$ with size of $n \times 1$ can be recovered in a closed-form solution

$$\boldsymbol{\alpha}' = \left(\mathbf{K} + \lambda^2 \boldsymbol{\Gamma}_{\Phi(\mathbf{y})}^T \boldsymbol{\Gamma}_{\Phi(\mathbf{y})}\right)^{-1} \mathbf{k}(\cdot,\mathbf{y}) \tag{8}$$

where $\mathbf{k}(\cdot,\mathbf{y}) = [k(\mathbf{x}_1,\mathbf{y}), k(\mathbf{x}_2,\mathbf{y}), \ldots, k(\mathbf{x}_n,\mathbf{y})]^T \in \mathbb{R}^{n\times 1}$, and $\mathbf{K} = \boldsymbol{\Phi}^T\boldsymbol{\Phi} \in \mathbb{R}^{n\times n}$ is the Gram matrix with $\mathbf{K}_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$. The weight vector $\boldsymbol{\alpha}'$ is "partitioned" into $\boldsymbol{\alpha}'_l = \{\alpha'_i | \ \forall i \ s.t. \ \omega_i = l\} \in \mathbb{R}^{n_l\times 1}$. Class label of the test sample is finally determined by

$$\text{class}(\mathbf{y}) = \arg\min_{l=1,\ldots,C} \|\boldsymbol{\Phi}_l\boldsymbol{\alpha}'_l - \Phi(\mathbf{y})\|_2 . \tag{9}$$

In (9), $\boldsymbol{\Phi}_l = [\Phi(\mathbf{x}_{l,1}), \Phi(\mathbf{x}_{l,2}), \ldots, \Phi(\mathbf{x}_{l,n_l})]$ represents the kernel sub-dictionary in class l, and it can be further expressed as

$$\begin{aligned} \|\boldsymbol{\Phi}_l\boldsymbol{\alpha}'_l - \Phi(\mathbf{y})\|_2 &= \sqrt{(\Phi(\mathbf{y}) - \boldsymbol{\Phi}_l\boldsymbol{\alpha}'_l)^T (\Phi(\mathbf{y}) - \boldsymbol{\Phi}_l\boldsymbol{\alpha}'_l)} \\ &= \sqrt{k(\mathbf{y},\mathbf{y}) + \boldsymbol{\alpha}'^T_l \mathbf{K}_l \boldsymbol{\alpha}'_l - 2\boldsymbol{\alpha}'^T_l \mathbf{k}_l(\cdot,\mathbf{y})} \end{aligned} \tag{10}$$

where $\mathbf{K}_l$ is the Gram matrix of the samples in class l, and $\mathbf{k}_l(\cdot,\mathbf{y}) = [k(\mathbf{x}_{l,1},\mathbf{y}), k(\mathbf{x}_{l,2},\mathbf{y}), \ldots, k(\mathbf{x}_{l,n_l},\mathbf{y})]^T \in \mathbb{R}^{n_l\times 1}$.

IV. EXPERIMENTS AND ANALYSIS

A. Experimental Data

The first experimental data employed was acquired using National Aeronautics and Space Administration's (NASA) Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor and was collected over northwest Indiana's Indian Pine test site in June 1992. The image represents a rural scenario with 145 × 145 pixels and 220 bands in the 0.4- to 2.45-μm region of the visible and infrared spectrum with a spatial resolution of 20 m. In this letter, a total of 202 bands is used after removal of water-absorption bands. There are 16 different land-cover classes in the original ground truth map, and 10% training samples are used as shown in Table I.

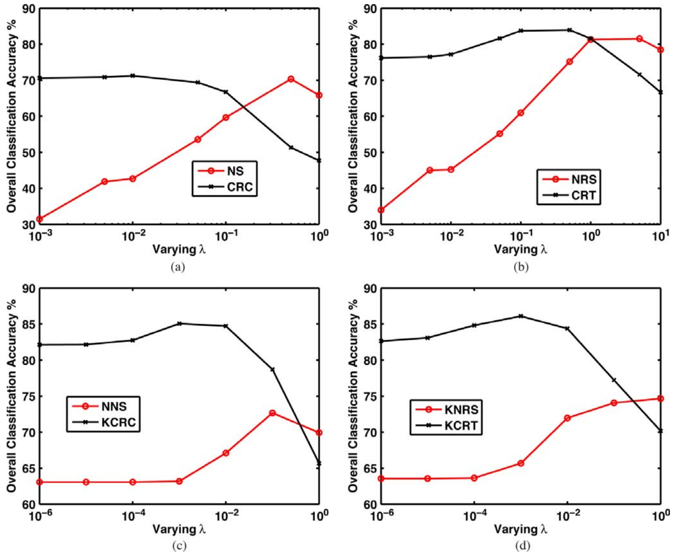
The second experimental hyperspectral data set employed was collected by the Reflective Optics System Imaging Spectrometer (ROSIS) sensor. The image, covering the city of Pavia, Italy, was collected under the HySens project managed by DLR. The data has a spectral coverage from 0.43 to 0.86 μm, and a spatial resolution of 1.3 m. The scene used in our experiment is the university area which has 103 spectral bands with a spatial coverage of 610 × 340 pixels. There are nine classes for the data set, and only 60 samples per class are used for training as shown in Table II.

B. Post-Partitioning versus Pre-Partitioning

We first investigate the effect of *post-partitioning* and *pre-partitioning* on the performance of NS/NNS, CRC/KCRC, NRS/KNRS, and CRT/KCRT, respectively. For convenience, only spectral signatures are employed, RBF kernel is used, and

TABLE II
CLASSIFICATION ACCURACY (%) FOR THE UNIVERSITY OF PAVIA DATA SET

Class	# samples		Classification algorithms										
	Train	Test	SVM	SVM-CK [3]	SRC [4]	KSRC [7]	KSRC-CK [8]	CRC [9]	KCRC [10]	KCRC-CK	CRT	KCRT	KCRT-CK
1	60	6631	80.29	84.68	81.75	80.00	93.33	68.86	70.08	83.59	82.05	81.72	**91.67**
2	60	18649	84.32	96.85	67.35	65.72	91.67	65.55	85.28	96.79	80.05	84.15	**98.33**
3	60	2099	82.84	89.52	74.85	73.75	85.00	64.41	86.80	89.43	79.80	85.99	**90.00**
4	60	3064	92.26	96.64	94.65	95.48	90.00	93.44	94.81	**97.03**	95.76	94.42	93.33
5	60	1345	99.11	99.70	99.55	99.26	98.33	99.78	99.48	100	99.55	99.26	**100**
6	60	5029	89.12	**93.64**	80.73	83.45	90.00	85.58	87.55	92.78	92.27	86.18	93.33
7	60	1330	92.01	**97.44**	74.51	78.16	85.00	63.31	94.56	98.89	94.89	94.89	96.67
8	60	3682	79.71	88.10	71.62	63.30	81.67	68.03	79.20	89.88	87.06	88.73	**90.00**
9	60	947	99.79	99.79	100	100	100	81.84	100	100	99.47	100	**100**
Overall Accuracy (%)			84.06	93.63	75.94	82.04	91.56	71.94	84.25	93.72	83.94	84.98	**94.81**
Kappa Coefficient (κ)			0.8002	0.9322	0.6950	0.7979	0.9238	0.6483	0.8012	0.9340	0.7974	0.8043	**0.9417**



(a)

(b)

(c)

(d)

Fig. 1. For Indian Pines data, the classification performance as a function of varying λ using (a) linear versions without Tikhonov regularization: CRC (*post-partitioning*) and NS (*pre-partitioning*), (b) linear versions with Tikhonov regularization: CRT (*post-partitioning*) and NRS (*pre-partitioning*), (c) nonlinear versions without Tikhonov regularization: KCRC and NNS, and (d) nonlinear versions with Tikhonov regularization: KCRT and KNRS.

the parameter γ of RBF kernel is set by the median value of $1/(\|\mathbf{x}_i - \bar{\mathbf{x}}\|_2^2)$, $i = 1, 2, \ldots, n$, where $\bar{\mathbf{x}} = (1/n)\sum_{i=1}^{n}\mathbf{x}_i$ is the mean of all available training samples [7]. The experimental results for the Indian Pines data are shown in Fig. 1. The following observations can be summarized: (1) with optimal parameter λ, NS (*pre-partitioning*) has similar performance to CRC (*post-partitioning*), NRS (*pre-partitioning*) basically is also similar to CRT (*post-partitioning*) due to the regularization, and NRS/CRT is superior to NS/CRC; and (2) in kernel domain, NNS (*pre-partitioning*) is obvious worse than KCRC (*post-partitioning*), and KNRS (*pre-partitioning*) is also worse than the proposed KCRT (*post-partitioning*).

C. Parameter Tuning

We study the window size as well as the regularization parameter λ for the proposed KCRT-CK. The window size determines the number of spatially neighboring pixels averaged, which is a significant parameter to measure the local homogeneity. Additionally, as a global regularization parameter, the adjustment of λ is also important to the algorithm performance. We report experiments demonstrating the sensitivity of the proposed method over a wide range of the parameter space. In general, leave-one-out cross validation (LOOCV) strategy based on available training samples is considered for the parameter tuning. Fig. 2 illustrates the classification accuracy versus varying λ and different window sizes for the proposed KCRT-CK. Optimal parameters (e.g., window size and regularization parameter λ) for two experimental data are obviously shown in Fig. 2: optimal window size is 9×9 and $\lambda = 10^{-3}$ for the Indian Pines data, and optimal window size is 3×3 and $\lambda = 5 \times 10^{-4}$ for the University of Pavia data. Note that for the Indian Pines data, it covers a rural area with large homogenous regions; comparatively, for the other data, it represents an urban area with dense and individual buildings, which causes the optimal window size is relatively smaller. For other classifiers, such as SVM,[2] cross validation is also employed to determine the related parameters, and all optimal ones are used in the following experiments.

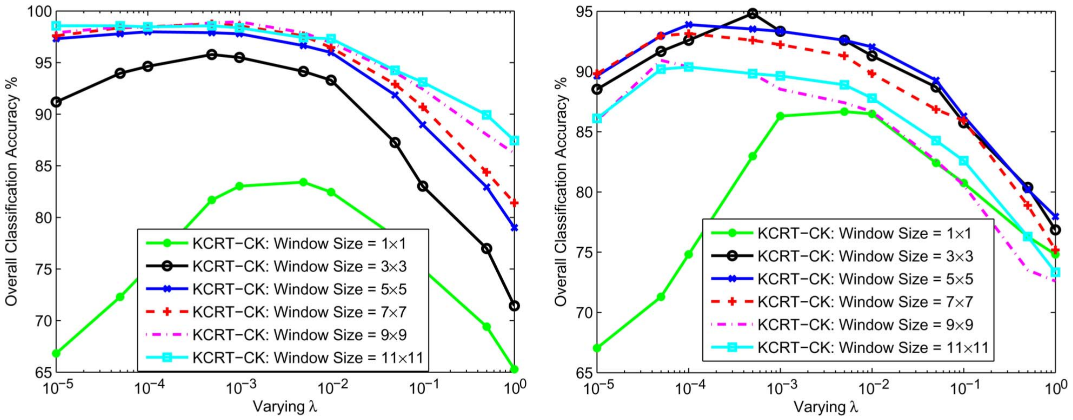
D. Classification Performance

We mainly compare the classification accuracy (e.g., overall accuracy, kappa coefficient and the individual class accuracy) of the proposed KCRT/KCRT-CK with KCRC/KCRC-CK and KSRC/KSRC-CK.[3] Both CRC- and SRC-based classifiers can be viewed as the representation-based classifiers by connecting the linear relationship between the training and test samples. Even though SVM does not belong to this type of classifiers, we consider it and its extension (e.g., SVM-CK) in comparison due to their popularity. To avoid any bias, we randomly choose the training and testing samples, repeat the experiments 20 times, and report the average classification accuracy.

The performance of aforementioned classification methods is summarized in Tables I, II. The kernel version of representation-based methods leads to better performance than the original version (e.g., CRC, SRC, and CRT) as it is expected. We also observe that KCRT/KCRT-CK always outperforms KCRC/KCRC-CK, which indicates that the biasing Tikhonov matrix works effectively to measure the distance in the kernel-induced space, especially with the spatial-spectral kernel, and adaptively influences the calculation of weight vector. The proposed KCRT-CK achieves the highest classification accuracy (around 98% and 95% for two data, respectively) and

[2] SVM with RBF kernel is implemented using the `libsvm` package; http://www.csie.ntu.edu.tw/cjlinn/libsvm

[3] SRC with the ℓ_1—minimization is implemented using the `l1_ls` package; http://www.stanford.edu/boyd/software.html

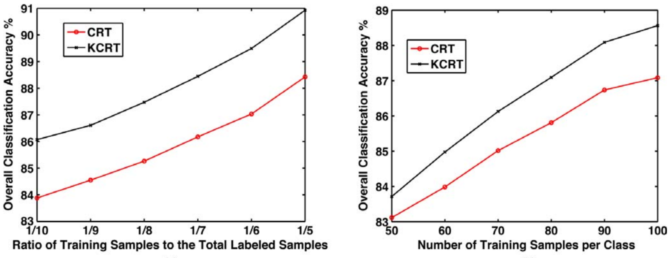


(a) (b)

Fig. 2. Classification performance of the proposed KCRT-CK as a function of varying λ in the two experimental data. (a) Indian Pines; (b) University of Pavia.

TABLE III
STATISTICAL SIGNIFICANCE FROM THE STANDARDIZED MCNEMAR'S TEST ABOUT THE DIFFERENCE BETWEEN METHODS

	Indian Pines data	University of Pavia data
	$\|z\|$/significant?	$\|z\|$/significant?
KCRC *vs* CRC	42.06/yes	84.38/yes
KCRC-CK *vs* CRC	59.41/yes	103.80/yes
KCRT *vs* CRT	8.53/yes	8.91/yes
KCRT-CK *vs* CRT	35.73/yes	53.64/yes
KCRT *vs* KCRC	3.61/yes	4.53/yes



(a) (b)

Fig. 3. Classification performance of both CRT and KCRT with different numbers of training-sample sizes in the two experimental data. (a) Indian Pines; (b) University of Pavia.

obviously outperforms the state-of-the-art KSRC, KSRC-CK as well as SVM-CK. The standardized McNemar's test has been employed to verify the statistical significance in accuracy improvement of the proposed methods. As listed in Table III, the $|z|$ values of McNemar's test larger than 1.96 and 2.58 mean that two results are statistically different at the 95% and 99% confidence levels.

Fig. 3 further illustrates the comparisons between the proposed KCRT and CRT with different numbers of training-sample sizes. For the Indian Pines data, the training size is changed from 1/10 to 1/5 (note that 1/10 is the ratio of number of training samples to the total labeled data), while for the University of Pavia data, it is changed from 50 samples per class to 100 samples. It is obvious that the classification performance of KCRT is consistently better than that of CRT (around 2% improvement for the Indian Pines data and 1% improvement for the University of Pavia data).

V. CONCLUSION

In this letter, KCRT was proposed for hyperspectral image classification. It is found that *post-partitioning* as used in KCRT is more appropriate particularly in the kernel-induced feature space. Moreover, KCRT-CK was proposed to incorporate the spatial information at neighboring locations in the kernel-induced space. Experimental results on real hyperspectral images verified that the proposed KCRT-CK outperforms the traditional SVM-CK and the state-of-the-art kernel classifiers, such as KSRC/KSRC-CK and KCRC/KCRC-CK.

REFERENCES

[1] R. Archibald and G. Fann, "Feature selection and classification of hyperspectral images with support vector machines," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 4, pp. 674–677, Oct. 2007.

[2] W. Li, S. Prasad, and J. E. Fowler, "Decision fusion in kernel-induced spaces for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 6, pp. 3399–3411, Jun. 2014.

[3] G. Camps-Valls, L. Gomez-Chova, J. Muñoz-Marí, J. Vila-Francés, and J. Calpe-Maravilla, "Composite kernels for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 93–97, Jan. 2006.

[4] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via space representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[5] Q. Sami ul Haq, L. Tao, F. Sun, and S. Yang, "A fast and robust sparse approach for hyperspectral data classification using a few labeled samples," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 6, pp. 2287–2302, Jun. 2012.

[6] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification via kernel sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 1, pp. 217–231, Jan. 2013.

[7] L. Zhang *et al.*, "Kernel sparse representation-based classifier," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 1684–1695, Apr. 2012.

[8] J. Liu, Z. Wu, Z. Wei, L. Xiao, and L. Sun, "Spatial-spectral kernel sparse representation for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 6, pp. 2462–2471, Dec. 2013.

[9] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: Which helps face recognition?" in *Proc. Int. Conf. Comput. Vis.*, Barcelona, Spain, Nov. 2011, pp. 471–478.

[10] B. Wang, W. Li, N. Poh, and Q. Liao, "Kernel collaborative representation-based classifier for face recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Vancouver, BC, Canada, May 2013, pp. 2877–2881.

[11] L. Zhang, W. Zhou, and B. Liu, "Nonlinear nearest subspace classifier," in *Proc. 18th Int. Conf. Neural Inf. Process.*, Shanghai, China, Nov. 2011, pp. 638–645.

[12] W. Li, E. W. Tramel, S. Prasad, and J. E. Fowler, "Nearest regularized subspace for hyperspectral classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 477–489, Jan. 2014.