

Hyperspectral Image Classification via Fusing Correlation Coefficient and Joint Sparse Representation

Bing Tu, *Member, IEEE*, Xiaofei Zhang, *Student Member, IEEE*,
Xudong Kang^{ID}, *Senior Member, IEEE*, Guoyun Zhang,
Jinping Wang, *Student Member, IEEE*, and Jianhui Wu

Abstract—The joint sparse representation (JSR)-based classifier assumes that pixels in a local window can be jointly and sparsely represented by a dictionary constructed by the training samples. The class label of each pixel can be decided according to the representation residual. However, once the local window of each pixel includes pixels from different classes, the performance of the JSR classifier may be seriously decreased. Since correlation coefficient (CC) is able to measure the spectral similarity among different pixels efficiently, this letter proposes a new classification method via fusing CC and JSR, which attempts to use the within-class similarity between training and test samples while decreasing the between-class interference. First, the CCs among the training and test samples are calculated. Then, the JSR-based classifier is used to obtain the representation residuals of different pixels. Finally, a regularization parameter λ is introduced to achieve the balance between the JSR and the CC. Experimental results obtained on the Indian Pines data set demonstrate the competitive performance of the proposed approach with respect to other widely used classifiers.

Index Terms—Correlation coefficient (CC), hyperspectral imagery, joint sparse representation (JSR).

I. INTRODUCTION

HYPERSPECTRAL images (HSIs) have the ability to provide spatial structures and the spectral signatures of different materials. Taking advantage of those useful information into account, HSI remote sensing has been widely used in many applications, such as environmental monitoring and

precision agriculture. Some popular classifiers, such as support vector machine (SVM) [1] and relevance vector machine [2], have been widely used in HSI analysis.

Recently, sparse representation (SR), serving as a powerful image processing tool, has been applied for HSI classification [3]. SR relies on the assumption that pixels from the same class are supposed to hold similar spectral characteristics, and thus, a test sample can be linearly represented by a small number of training samples from the same class. However, the traditional SR approach just considers the spectral information of a test pixel and ignores the spatial neighbors surrounding the test pixel. Based on the assumption that pixels from a local region usually have similar spectral materials and characteristics, Chen *et al.* [4] proposed a JSR-based classification (JSRC) algorithm, which takes spatial information into account by representing the pixels in a local window jointly, so as to obtain a better classification performance. In [5] and [6], some sparse-based classifiers, such as the kernel-based SR and the l_2 -norm regularized sparse subspace clustering, have been proposed which show much better performance. However, a common limitation of those local window-based methods is that the local windows may include pixels from different classes. In other words, pixels in the edge areas are suitable with small-sized windows, whereas a large-sized window is preferred for smooth areas. To address this problem, Fang *et al.* [7] proposed the multiscale adaptive SR (MASR) method, which obtained an improved performance. Other advancing tools, such as the adaptive mean-shift analysis method [8], are also an effective way to solve this problem.

A K nearest neighbor (KNN) classifier is one kind of classical classification methods. It makes full use of the nonlocal information in the HSI by finding a predefined number of training samples which are closest in Euclidean distance to the test sample and then assign the test sample with the label which has the largest number of nearest training samples. Several extensions of this classifier have been studied. In 2015, Cui and Prasad [9] proposed a class-dependent SR classification (CDSRC) for HSI classification, which effectively combines the ideas of SR and KNN classifier in a classwise manner to exploit both correlation and Euclidean distance among training and test samples. Li *et al.* [10] proposed a KNN and collaborative representation (CR)-based

Manuscript received September 24, 2017; revised November 29, 2017; accepted December 21, 2017. Date of publication January 24, 2018; date of current version February 23, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 51704115, in part by the Key Laboratory Open Fund Project of Hunan Province University under Grant 17K040 and Grant 15K051, in part by the Hunan Provincial Natural Science Foundation under Grant 2016JJ2064, in part by the Fund of Education Department of Hunan Province under Grant 16C0723, and in part by the Science and Technology Program of Hunan Province under Grant 2016TP1021. (Corresponding author: Xudong Kang.)

B. Tu, X. Zhang, G. Zhang, J. Wang, and J. Wu are with the College of Information and Communication Engineering, Hunan Institute of Science and Technology, Yueyang 414000, China (e-mail: tubing@hnist.edu.cn; xiaofei_zhang@vip.hnlist.edu.cn; gyzzhang@hnist.edu.cn; jinpings_wang@foxmail.com; jhwu@hnist.edu.cn).

X. Kang is with the College of Electrical and Information Engineering, Hunan University, 410082 Changsha, China (e-mail: xudong_kang@163.com).

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

Digital Object Identifier 10.1109/LGRS.2017.2787338

classifier which achieves better classification performance than several previous algorithms, such as the original KNN classifier. In [11], a kernel-fused representation-based classifier has been proposed for HSI classification, which combines SR and CR into an effective kernel fused representation-based classification framework.

One limitation of the KNN classifier is that it uses the Euclidean distance which is not always the optimal option for measuring the distance between two hyperspectral pixels. In this letter, it is found that the correlation coefficient (CC) can recognize those pixels with strong relationships effectively. Therefore, this letter proposes a classification method by combining CC and JSR (CCJSR) for HSI. It can be accomplished by three main steps. In the first step, a test sample can be linearly represented by the atoms in an over complete dictionary and sparse vectors. In this step, JSR is used to produce the residual for every class. In the second step, CC is used to calculate the degree of similarity between the training and test samples. In the last step, a decision function is used for classification based on the residual of JSR and the degree of correlation. The proposed method combines two major factors, i.e., spectral similarity and local spatial consistency, for HSI classification. The major contributions are concluded as follows. First, this letter introduces the CC to model the spectral similarity among the training and test pixels. Second, by fusing the CC and JSR, the proposed CCJSR method can effectively overcome within-class variations and between-class interference among pixels. At last, an effective decision fusion strategy is proposed to achieve the balance between CC and JSR. Experimental results performed on a real hyperspectral data set demonstrate the effectiveness of the proposed method in terms of classification accuracies.

II. RELATED WORK

A. Joint Sparse Representation

Based on the assumption that neighboring hyperspectral pixels usually consist of similar materials and share the same spectral characteristics, the JSRC method was proposed in [4], and the specific details are shown as follows.

The JSRC model assumes that pixels from the same class commonly represent the same spectral characteristics. Assume that two neighboring hyperspectral pixels y_1 and y_2 have similar materials, the SR of y_1 can be written as

$$y_1 = \mathbf{X}a_1 = a_{1,\mu_1}x_{\mu_1} + a_{1,\mu_2}x_{\mu_2} + \dots + a_{1,\mu_n}x_{\mu_n} \quad (1)$$

where the index set $\{\mu_1, \mu_2, \dots, \mu_n\}$ is the support of the sparse vector a_1 , and $n = \|a_1\|_0$ denotes the ℓ_0 -norm (or sparsity level) of a_1 . Since y_1 and y_2 consist of similar materials, y_2 can be approximated by the same index set of training samples $\{x_{\mu_1}, x_{\mu_2}, \dots, x_{\mu_n}\}$ with a different sparse vector $\{a_{2,\mu_1}, a_{2,\mu_2}, \dots, a_{2,\mu_n}\}$

$$y_2 = \mathbf{X}a_2 = a_{2,\mu_1}x_{\mu_1} + a_{2,\mu_2}x_{\mu_2} + \dots + a_{2,\mu_n}x_{\mu_n}. \quad (2)$$

For the JSRC, it is assumed that the m pixels in a $\sqrt{m} \times \sqrt{m}$ window are all from the same class j . In this situation, the training samples \mathbf{X}^j are HSI pixels from class j . a^j is a sparse vector whose entries correspond to the training samples

\mathbf{X}^j . The full $\mathbf{X} = [\mathbf{X}^1 \dots \mathbf{X}^j \dots \mathbf{X}^c]$ consists of all the subdivisions of total c classes. Therefore, the unknown test samples $\mathbf{Y} = [y_1^j \ y_2^j \ \dots \ y_m^j]$ which consist of those pixels in a small window can be represented as

$$\begin{aligned} \mathbf{Y} &= [y_1^j \ y_2^j \ y_m^j] = [\mathbf{X}^j a_1^j \ \mathbf{X}^j a_2^j \ \dots \ \mathbf{X}^j a_m^j] \\ &= [\mathbf{X}^1 \dots \mathbf{X}^j \dots \mathbf{X}^c] \cdot [\mathbf{0} \ \dots \ \beta^j \ \dots \ \mathbf{0}]^T = \mathbf{X}\beta \end{aligned} \quad (3)$$

where $y^j = \mathbf{X}^j a^j$. Since the number of columns in the subdictionary \mathbf{X}^j is much less than the number of columns in dictionary \mathbf{X} , the number of rows in β^j is much less than the number of rows in β . Therefore, β is a sparse vector matrix with only a few nonzero rows which can be obtained as follows:

$$\hat{\beta} = \arg \min \|\beta\|_0 \quad \text{s.t.} \quad \mathbf{Y} = \mathbf{X} \cdot \beta \quad (4)$$

where $\|\beta\|_0$ represents the number of nonzero rows of β . The optimization problem in (4) can be relaxed to an inequality problem

$$\hat{\beta} = \arg \min \|\mathbf{X}\beta - \mathbf{Y}\|_2 \quad \text{s.t.} \quad \|\beta\|_0 \leq \eta \quad (5)$$

where η is the given sparsity level. The simultaneous orthogonal matching pursuit method can be used to solve the problem in (5). Once β is obtained, the reconstruction residual errors-based classification of the central pixels can be described as

$$\text{Class}(y) = \arg \min_{j=1,2,\dots,c} r^j(y) \quad (6)$$

where $r^j(y) = \|\mathbf{Y} - \mathbf{X}^j \cdot \beta^j\|_2$, $j = 1, 2, \dots, c$ is the corresponding reconstruction residual errors of the j th class.

B. Correlation Coefficient

The CC is an effective metric that determines whether two variables are associated. In this letter, the CCs among different pixels are used to determine whether these pixels belong to the same class. To objectively measure the correlation of two pixels, \mathbf{A} and \mathbf{B} with d dimension spectral values the CC can be calculated as follows:

$$\begin{aligned} \rho &= \frac{\text{cov}(\mathbf{A}, \mathbf{B})}{\sqrt{\text{var}(\mathbf{A})} \cdot \sqrt{\text{var}(\mathbf{B})}} \\ &= \frac{\sum_{z=1}^d (a_z - u_a)(b_z - u_b)}{\sqrt{\sum_{z=1}^d (a_z - u_a)^2} \cdot \sqrt{\sum_{z=1}^d (b_z - u_b)^2}} \end{aligned} \quad (7)$$

where $\text{var}(\mathbf{A})$ and $\text{var}(\mathbf{B})$ refer to the variance of \mathbf{A} and \mathbf{B} . $\mathbf{A} = \{a_z\}_{z=1}^d$, $\mathbf{B} = \{b_z\}_{z=1}^d$, $u_a = (1/d) \sum_{z=1}^d a_z$, $u_b = (1/d) \sum_{z=1}^d b_z$ and $|\rho| \leq 1$. More generally, $\rho > 0$ means that the positive correlation and vice versa. The closer the absolute value of ρ is to 1, the stronger the relationship of the two vectors is.

III. PROPOSED APPROACH

A. Motivation

Fig. 1 shows a graphical example illustrating the motivation behind the proposed method. There are three randomly selected training samples (from the 14th class, the second class, and the 16th class) and one test sample from the

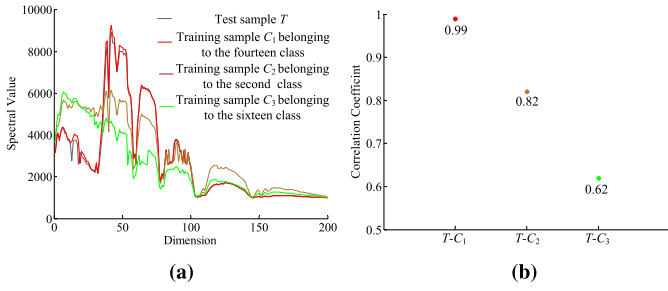


Fig. 1. (a) Spectral values of the four randomly selected samples (three for training and one for test). (b) CCs among the test sample T and the three training samples.

14th class. Fig. 1(a) and (b) shows the spectral values of those pixels, and the CCs among the test and the three training samples, respectively. As shown in Fig. 1(b), it can be seen that the pixels in the same class usually have high correlation with each other and vice versa. However, in the case of high spectral similarity among different materials (caused by some disruptive factors, such as spectral mixing, shadow, and so on), the CC itself is still not able to achieve an efficient classification performance.

The JSR-based method simply assumes that pixels in a local neighborhood are composed of the same material. This assumption is not always correct, since some pixels in a local region may have quite different spectral curves and thus influence the performance of the JSR classifier. Therefore, fusing CC and JSR is expected to be an effective measure to distinguish those pixels with different spectral properties while overcome the influence caused by those disruptive factors. In this letter, the CC is introduced into the JSR classifier, so as to combine the local spatial information and nonlocal spectral information in an effective way.

B. CCJSR Algorithm

This letter introduces an HSI classifier that combines JSR and CC. The proposed CCJSR is composed of two components (CC and JSR). The major steps of the proposed method can be concluded in Algorithm 1. First, we calculate the CCs among the training and test samples. Second, the representation residuals are calculated using the JSR. Third, the class label of each pixel is determined based on the defined decision function. The details are shown as follows.

Assuming $\mathbf{X}^j = [\mathbf{X}_1^j \ \mathbf{X}_2^j \ \dots \ \mathbf{X}_{k_j}^j]$, in which $\mathbf{X}_{k_j}^j$ belongs to j th class, and the j th class consists of k_j training samples. This letter first calculates the CCs among the training and test samples. For one test sample y and one training sample $\mathbf{X}_{k_j}^j$, the CC $\rho_{k_j}^j$ can be calculated as follows:

$$\begin{aligned} \rho_{k_j}^j &= \frac{\text{cov}(\mathbf{X}_{k_j}^j, y)}{\sqrt{\text{var}(\mathbf{X}_{k_j}^j)} \cdot \sqrt{\text{var}(y)}} \\ &= \frac{\sum_{z=1}^d [(\mathbf{X}_{k_j}^j)_z - u_{\mathbf{X}_{k_j}^j}] \cdot [(y)_z - u_y]}{\sqrt{\sum_{z=1}^d [(\mathbf{X}_{k_j}^j)_z - u_{\mathbf{X}_{k_j}^j}]^2} \cdot \sqrt{\sum_{z=1}^d [(y)_z - u_y]^2}} \end{aligned} \quad (8)$$

where d represents as the dimension of the samples. $\text{var}(\mathbf{X}_{k_j}^j)$ and $\text{var}(y)$ refer to the variances of $\mathbf{X}_{k_j}^j$ and y . And $u_{\mathbf{X}_{k_j}^j} = (1/d) \sum_{z=1}^d (\mathbf{X}_{k_j}^j)_z$, $u_y = (1/d) \sum_{z=1}^d (y)_z$, $\rho^j = \{\rho_1^j, \rho_2^j, \dots, \rho_{k_j}^j\}$.

Then, for the training samples in each class, the matrix ρ^j is sorted in descending order based on the CCs of different training samples. After that, this letter calculates the mean of N largest ρ^j as the CC cor^j . Suppose that N largest ρ^j consist of $\{\rho_1^j, \rho_2^j, \dots, \rho_N^j\}$, the CC cor^j can be calculated by

$$\text{cor}^j = \frac{1}{N} (\rho_1^j + \rho_2^j + \dots + \rho_N^j). \quad (9)$$

Algorithm 1 CCJSR

Inputs: Training set $\mathbf{X}_\tau = \{(x_1, j_1), \dots, (x_\tau, j_\tau)\} \in (\mathbb{R}^d \times j)^\tau$ (d is the number of spectral band, x_τ refers to the τ th training sample, j_τ is the class label of x_τ and class label $j = \{1, 2, \dots, c\}$); test samples $\mathbf{Y} = (y_1, \dots, y_n) \in \mathbb{R}^d$; the HSI \mathbf{I} ; sparsity level S_j ; number of nearest neighbors N ; the regularization parameter λ ; and joint neighboring scale S_e .

Step 1 CC

```

for  $i = 1, \dots, n$ 
  for  $j = 1, \dots, c$ 
    Calculate the CC between
      the training and test sample based on (8);
  end for
  Calculate the mean of  $N$  largest  $\rho_i^j$  as the
     $j$ th class CC  $\text{cor}_i^j$ 
end for

```

Step 2 JSR

```

for  $i = 1, \dots, n$ 
  Update joint neighboring scale  $S_e$  and calculate
    the sparse coefficient by SOMP;
  for  $j = 1, \dots, c$ 
    Calculate the residual  $r_i^j$  based on (12);
  end for
  Calculate the norm of the  $j$ th class residual
     $r_i^j = \|r_i^j\|_2$ 
end for

```

Step 3 Determine the class label of y_i based on (13)

Outputs: The classification map.

Next, we calculate the residual of the JSR. The training samples $\mathbf{X} = [\mathbf{X}^1 \ \mathbf{X}^2 \ \dots \ \mathbf{X}^j]$ are used as the dictionary for representation, and $\boldsymbol{\alpha}^j = [\alpha_1^j \ \alpha_2^j \ \dots \ \alpha_{k_j}^j]$ is a sparse vector related to \mathbf{X}^j . The pixel y can be represented by a linear combination of these training samples

$$y = \mathbf{X}^j \boldsymbol{\alpha}^j. \quad (10)$$

To obtain the sparsest solution, the sparse matrix $\hat{\boldsymbol{\alpha}}^j$ can be obtained as follows:

$$\hat{\boldsymbol{\alpha}}^j = \arg \min \|y - \mathbf{X}^j \boldsymbol{\alpha}^j\|_2 \quad \text{s.t.} \quad \|\boldsymbol{\alpha}^j\|_0 \leq \eta. \quad (11)$$

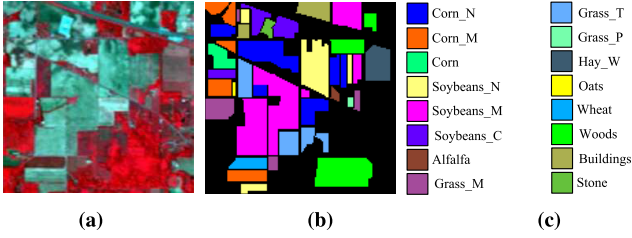


Fig. 2. (a) False-color composite of the Indian Pines image. (b) and (c) Reference data for the Indian Pines image.

After calculating the sparse matrix $\hat{\alpha}^j$, the residual of each class r^j can be calculated via

$$r^j(y) = \|y - \mathbf{X}^j \hat{\alpha}^j\|_2 \quad j = 1, 2, \dots, c. \quad (12)$$

Finally, this letter combines the CC and the JSR in the decision level to exploit both the CCs among training and test samples and the representation residuals. By introducing a regularization parameter λ into the decision function, the class label of the test pixel y can be obtained as follows:

$$\text{Class}(y) = \arg \min_{j=1,2,\dots,c} (r^j(y) + \lambda \cdot (1 - \text{cor}^j(y))) \quad (13)$$

where $\text{cor}^j \in [0, 1]$, which represents the CCs among pixels.

IV. EXPERIMENTAL RESULTS

A. Data Set

One real hyperspectral data set is used for experimental evaluation in this letter. The Indian Pines image shows the Indian Pines Test Field in the northwest of Indiana, which is captured by the Airborne Visible/Infrared Imaging Spectrometer remote-sensing device. The image is of a size 145×145 pixels with a spatial resolution of 20 m per pixel and 220 wave bands. With 20 water absorption wave bands removed, the 200 bands are used in the experiment. As the scene is captured in June, some crops, such as corn and soybean, are still in the early stage of growth. In the reference classification map obtained from site exploration, the scene is divided into 16 different classes. Fig. 2 indicates the false-color composite of the Indian Pines image, the corresponding reference date, and color labels.

B. Parameter Analysis

Here, the relationship among overall accuracy (OA), the joint neighboring scale S_e , and the sparsity level S_l is analyzed in Fig. 3(a). As S_e is increasing, the classification accuracy increases first and then decreases. This phenomenon indicates that the joint representation scheme is useful, since neighboring hyperspectral pixels usually consist of similar materials and share the same spectral characteristics. However, when the joint neighboring scale S_e is very large, the samples belonging to different classes may be included into the same local window and thus results in the decrease of classification accuracy. In addition, with the parameter S_l increasing, the OA decreases slowly. When S_e is set to 6 and S_l is set to 2, the proposed method obtains the highest classification accuracy. Therefore, $S_e = 6$ and $S_l = 2$ are set as the default parameters setting for the proposed method.

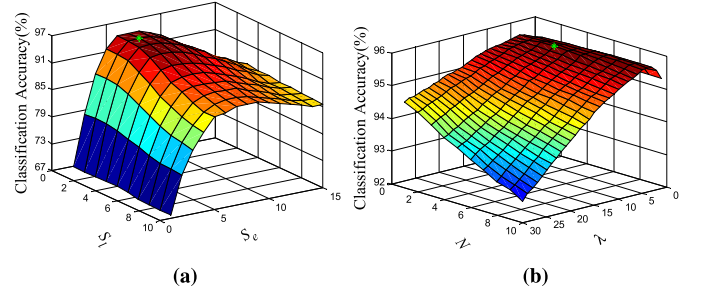


Fig. 3. (a) Analysis of the joint neighboring scale S_e and sparsity level S_l . (b) Analysis of the number of nearest neighbors N and regularized parameter λ .

The number of nearest neighbors N and the regularized parameter λ are also important parameters for the proposed CCJSR. Fig. 3(b) shows the performance of the proposed method with different values of N and λ . The parameter N is selected from the interval of $\{0, 2, 4, 6, 8, 10\}$, and the parameter λ is selected from the interval of $\{0, 5, 10, 15, 20, 25, 30\}$. If the regularization parameter λ is set to 0, it means that only the residual information of JSR is exploited during the classification stage. In addition, with λ increasing, the relationship between CC and JSR is also changing. As shown in Fig. 3(b), the proposed method can achieve the highest accuracy (95.85%) when λ equals to 0.6. The performance is much better than those obtained when $\lambda = 0$, which can demonstrate the importance of using the CC. Alternatively, as N is increasing, the classification accuracy increases first and then decreases. The reason is that those pixels belonging to the same class are usually composed of similar materials but sometimes influenced by shadow and other factors, calculating the mean of CC among all the training and test samples will decrease the classification accuracy. Therefore, $N = 4$ is used as the default parameter setting.

C. Comparisons With Other Approaches

In this section, the proposed method is compared with the SVMs method [1], SRC method [3], the joint sparse representation classification (JSRC) method [4], the class-dependent SRC (CDSRC) method [9], and the edge-preserving filtering (EPF) method [12]. The SVM is implemented by using the Gaussian kernel with fivefold cross validation. Other methods are implemented with the default parameters given by the authors. The experiment has been repeated ten times to obtain the accuracies in Table I.

The experiment is performed on the Indian Pines data set and 10% of the labeled data are randomly selected as training samples and the remaining 90% of data are used as test samples. The classification performances and classification maps of the compared methods are summarized in Table I and Fig. 4. It can be seen that, for some classes, such as Grass_T, Grass_P, Hay_W, Soybean_M, and Woods, the class accuracies of the proposed method are much higher than 96%. In addition, for some classes with small number of training samples, such as Grass_M and Oats, the proposed method can achieve class accuracies of 98.26% and 88.00%, whereas

TABLE I
CLASSIFICATION ACCURACY (IN PERCENT) OF THE INDIAN PINES IMAGE IN THE SVM, SRC, JSRC, CDSRC, EPF, AND CCJSR METHODS

The number of training samples is 10% of the reference data.								
Class	Training	Test	SVM	SRC	JSRC	CDSRC	EPF	CCJSR
Alfalfa	12	34	68.92(12.0)	64.56(12.5)	93.99(5.01)	100.0(0.00)	95.60(11.3)	95.29(3.00)
Corn_N	140	1288	77.92(2.16)	54.55(2.65)	92.64(4.66)	81.41(2.18)	95.74(2.05)	96.04(0.90)
Corn_M	83	747	79.04(3.97)	50.82(2.21)	87.23(16.8)	86.84(3.07)	95.68(2.37)	95.05(1.62)
Corn	20	217	67.92(3.99)	36.47(6.81)	92.87(5.62)	95.37(2.82)	95.60(2.79)	88.57(4.43)
Grass_P	50	433	88.65(3.03)	82.23(3.16)	93.56(5.19)	95.41(1.11)	98.37(1.19)	97.69(1.70)
Grass_T	73	657	89.13(1.91)	91.10(2.20)	93.68(4.83)	96.00(0.69)	95.29(2.27)	99.63(0.21)
Grass_M	5	23	95.43(6.52)	81.74(9.63)	96.38(3.45)	10.00(31.6)	100.0(0.00)	98.26(3.48)
Hay_W	50	428	97.34(0.92)	91.00(2.81)	92.27(10.4)	93.90(0.47)	100.0(0.00)	100.0(0.00)
Oats	5	15	57.25(10.6)	56.00(17.3)	91.09(7.39)	10.00(31.6)	90.00(31.6)	88.00(9.80)
Soybean_N	100	872	78.75(1.64)	66.41(2.71)	93.96(4.99)	85.52(2.53)	92.42(5.75)	93.78(1.44)
Soybean_M	250	2205	81.13(1.38)	71.92(1.83)	92.28(4.73)	75.29(1.29)	88.38(3.48)	95.91(0.24)
Soybean_C	60	533	77.75(2.66)	43.82(3.96)	87.02(16.8)	95.35(2.06)	92.31(5.34)	91.97(2.55)
Wheat	20	185	93.69(3.22)	90.35(3.67)	94.48 (4.92)	100.0(0.00)	100.0(0.00)	99.14(0.55)
Woods	130	1135	92.79(1.39)	89.41(1.78)	93.29(5.11)	92.69(1.13)	95.58(2.54)	99.42(0.33)
Buildings	40	346	74.45(5.28)	36.47(2.96)	93.81(4.89)	89.00(2.09)	95.07(1.89)	90.52(3.13)
Stone	10	83	99.15(0.74)	89.52(3.67)	94.27(6.27)	99.30(0.99)	98.71(0.09)	87.23(3.46)
OA			83.09(0.40)	68.86(0.93)	94.31(0.46)	85.63(0.55)	94.52(1.13)	96.00(0.46)
AA			82.46(1.78)	68.52(1.17)	92.91(1.32)	81.63(2.71)	95.55(1.94)	95.44(0.52)
Kappa			80.65(0.43)	64.43(1.06)	93.50(0.53)	83.40(0.66)	93.58(1.30)	94.78(0.94)

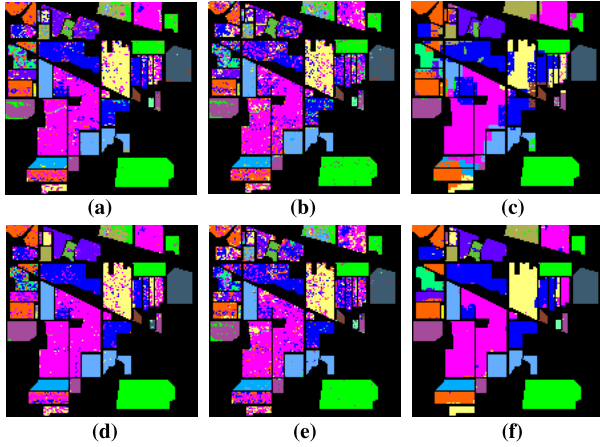


Fig. 4. Classification results (Indian Pines) obtained by (a) SVM method (83.09%), (b) SRC method (68.86%), (c) JSRC method (94.31%), (d) CDSRC method (85.63%), (e) EMP method (94.52%), and (f) CCJSR method (96.00%).

the class accuracies of other compared methods are much lower than it. Furthermore, the proposed method can achieve an OA of 96.00%, whereas the OA of the original JSRC is only 94.31%, which can demonstrate the advantage of taking CC among training and test samples into account. Generally, the proposed method always shows the best classification performance in terms of the highest OA, AA, and Kappa.

V. CONCLUSION

This letter proposes an HSI classification method based on JSR and CC. Considering that JSRC may include between-class interference, CC is introduced to model the spectral similarity among pixels in the CCJSR method. Compared with the original JSRC, the CCJSR method can make full use of the spatial contextual information and spectral similarity information at the same time. Furthermore, a decision function is introduced to achieve the balance between JSR and CC. Experiments performed on the Indian Pines data set demonstrate that CCJSR can improve the performance of the

JSRC effectively. The major objective of this letter is to demonstrate that spectral similarity among the training and test pixels is an important factor for HSI classification. It is obvious that, besides the CC metric, other spectral measures such as spectral angle measure can also be used in the proposed framework. Whether these metrics could lead to further improvement will be a future research topic.

REFERENCES

- [1] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.
- [2] F. A. Mianji and Y. Zhang, "Robust hyperspectral classification using relevance vector machine," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 6, pp. 2100–2112, Jun. 2011.
- [3] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.
- [4] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification using dictionary-based sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3973–3985, Oct. 2011.
- [5] 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.
- [6] H. Zhai, H. Zhang, L. Zhang, P. Li, and A. Plaza, "A new sparse subspace clustering algorithm for hyperspectral remote sensing imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 1, pp. 43–47, Jan. 2017.
- [7] L. Fang, S. Li, X. Kang, and J. A. Benediktsson, "Spectral-spatial hyperspectral image classification via multiscale adaptive sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 12, pp. 7738–7749, Dec. 2014.
- [8] X. Huang and L. Zhang, "An adaptive mean-shift analysis approach for object extraction and classification from urban hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 12, pp. 4173–4185, Dec. 2008.
- [9] M. Cui and S. Prasad, "Class-dependent sparse representation classifier for robust hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2683–2695, May 2015.
- [10] W. Li, Q. Du, F. Zhang, and W. Hu, "Collaborative-representation-based nearest neighbor classifier for hyperspectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 2, pp. 389–393, Feb. 2015.
- [11] L. Gan, P. Du, J. Xia, and Y. Meng, "Kernel fused representation-based classifier for hyperspectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 5, pp. 684–688, May 2017.
- [12] X. Kang, S. Li, and J. A. Benediktsson, "Spectral-spatial hyperspectral image classification with edge-preserving filtering," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2666–2677, May 2014.