Spatial-spectral Classification of Hyperspectral Images using Discriminative Dictionary Designed by Learning Vector Quantization

Zhaowen Wang, *Student Member, IEEE,* Nasser Nasrabadi, *Fellow, IEEE,* and Thomas Huang, *Life Fellow, IEEE*

Abstract

In this paper, a novel discriminative dictionary learning method is proposed for Sparse Representation-based Classification (SRC) to label high-dimensional Hyperspectral Imagery (HSI). In SRC, a dictionary is conventionally constructed using all the training pixels, which is not only inefficient due to the large size of typical HSI images, but is also ineffective in capturing class-discriminative information crucial for classification. We address the dictionary design problem with the inspiration from the Learning Vector Quantization (LVQ) technique and propose a hinge loss function that is directly related to the classification task as the objective function for dictionary learning. The resulting online learning procedure systematically "pulls" and "pushes" dictionary atoms so that they become better adapted to distinguish between different classes. In addition, the spatial context for a test pixel within its local neighborhood is modeled using a Bayesian graph model and incorporated with the sparse representation of a single test pixel in a unified probabilistic framework, which enables further refinement of our dictionary to capture the spatial class dependency that complements the spectral information. Experiments on different HSI images demonstrate that the dictionaries optimized using our method can achieve higher classification accuracy with substantially reduced dictionary size than using the whole training set. The proposed method also outperforms existing dictionary learning methods and attains the state-of-the-art results in both of the spectral-only and spatial-spectral settings.

Zhaowen Wang and Thomas Huang are with Beckman Institute, Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Illinois, IL 61801, USA.

Nasser Nasrabadi is with U.S. Army Research Laboratory, Adelphi, MD 20783, USA .

The work is supported by the U.S. Army Research Laboratory and U.S. Army Research Office under grant number W911NF-09-1-0383.

Index Terms

Classification, hyperspectral imagery, sparse representation, dictionary learning, learning vector quantization, spatial dependency.

I. INTRODUCTION

Hyperspectral Imagery (HSI) is an important tool in remote sensing, which can measure the distinct radiance patterns of different ground materials; and it is widely applied in agriculture, military, mineralogy, *etc.* [1], [2]. Each pixel in a hyperspectral image is a vector with hundreds or thousands of entries corresponding to spectral bands spanning a wide range of frequencies. Due to the high dimensionality of the pixels, it is very difficult to estimate the distribution of hyperspectral data especially with limited training samples, which poses a big challenge to classification task.

To deal with the curse of dimensionality problem, dimension reduction techniques have been exploited, including feature extraction and feature selection approaches [3]. Unfortunately, some discriminative information will get lost if too many dimensions are discarded. In the past decade, kernel methods such as Support Vector Machine (SVM) have been widely used for classifying high dimensional data, and have achieved great success in HSI applications [4], [5]. However, the kernel trick employed in SVM is not scalable to large data set and requires proper tuning of regularization parameters.

More recently, Sparse Representation-based Classification (SRC) [6] has been applied to HSI classification, which enjoys a straightforward generative interpretation and also achieves competitive results [7], [8]. Sparse representation expresses a signal as the linear combination of very few atoms from an over-complete dictionary, and the resulting sparse code can reveal its class information if signals from different classes lie in different subspaces. The effectiveness of SRC has already been demonstrated in face recognition [6], expression recognition [9], and speaker verification [10]. Good performance on HSI classification is also expected because the high correlation among different bands of hyperspectral images intrinsically induces a low dimensional manifold in which samples can be sparsely represented.

A good dictionary characterizing the subspace structure of each class is the key factor for SRC to attain high classification accuracy. Conventionally, an SRC dictionary is constructed by directly including all the training samples [6], [7], which is not efficient for HSI data when a

huge number of data samples are available. Random sampling or clustering methods such as K-means can give a compact dictionary. However, generative as well as discriminative capabilities are lost in such sub-optimal dictionaries. Lately, there has been active research in the computer vision and machine learning communities studying the design of compact dictionaries driven by large-scale training data. Generative approaches, such as Method of Optimal Direction (MOD) [11], K-SVD [12], [13], projected gradient [8], and convex relaxed l_1 formulations [14], [15], have focused on minimizing signal reconstruction errors. For better performance on classification, discrimination costs have been incorporated in the dictionary design in a supervised manner; *e.g.*, concatenating labels with signals in [16], combining with linear discriminant analysis in [17], and regularizing sparse codes pattern in [18]. Classification models other than SRC, such as linear classifier [19]–[21] and logistic regression [22]–[24], have also been used with sparse codes as inputs. However, it is unclear how the discrimination metrics used in existing methods are geared to the mechanism of SRC; moreover, the employment of an extra classification model (often requiring one-versus-rest paradigm in multi-class cases) will multiply the number of parameters and increase the risk of over-fitting.

In this paper, a new dictionary learning algorithm is proposed particularly for classification of hyperspectral data using a generic SRC classifier. We optimize the dictionary by minimizing the hinge loss of residual difference between competing classes, which is inspired by the idea behind the Learning Vector Quantization (LVQ) [25]. The LVQ technique was first applied to dictionary learning by Chen et al. [26] for HSI classification in an ad-hoc way; while here we adapt the philosophy of LVQ to SRC in a more principled manner and hence name the algorithm Learning Sparse Representation-based Classification (LSRC). The resultant dictionary updating rules mimic the "pulling" and "pushing" actions of LVQ, generating more discriminative atomic features that are useful for separating similar classes.

Besides spectral information, modeling the spatial dependency between the labels of neighboring pixels is also important for HSI classification. Spatial information has been previously used in morphological pre-processing [27], [28] or post-processing steps [29] independent of classification; and it has also been exploited with SVM classifiers simultaneously by composite kernels [30], [31] or a Markov Random Field (MRF) [32]. In methods based on sparse representation, joint sparsity model [7] and block-wise joint sparsity model [8] have been employed to enforce the spatial consistency of sparse codes. All the above-mentioned methods are effective

in enhancing the spatial smoothness of the same class labels; however, they are unable to capture the co-occurrence of different classes within a neighborhood. To address this problem, we propose a patch-based Bayesian network with a kernel-smoothed spatial dependency that fully characterizes the joint distribution of neighboring pixels' labels and at the same time takes a compact parametric form. More importantly, the sparse representation model for a single pixel spectrum can be combined with the spatial Bayesian network in a unified probabilistic framework, which allows us to apply the same LSRC algorithm in optimizing our dictionary with spatial contextual information taken into account. It is observed that both the proposed spatial Bayesian network and the dictionary trained with spatial context can further improve the classification performance.

The remainder of this paper is organized as follows. We first introduce the background of sparse representation and the framework of LSRC dictionary design inspired by LVQ techniques in Section II. An online dictionary optimization algorithm is then presented in Section III. In Section IV, spatial information is further incorporated in our dictionary learning model, leading to superior classification results on several HSI images as demonstrated by the experiments in Section V. We draw concluding remarks in Section VI.

II. Dictionary Learning for Sparse Representation-based Classification

A. Sparse Representation-based Classification (SRC)

Sparse representation has been shown to be effective in describing high-dimensional hyperspectral data, which intrinsically lie on a low-dimensional manifold [7], [8]. Suppose we have a data set containing N labeled HSI pixels of m spectral channels coming from C classes: $\{\mathbf{x}_i, y_i\}_{i=1...N}$, $\mathbf{x}_i \in \mathbb{R}^m$, $y_i \in \{1...C\}$. For each class c, there exists a dictionary $\mathbf{D}^c \in \mathbb{R}^{m \times n^c}$ of n^c atoms, such that n^c is much smaller than the number of samples in class c (might be larger than m) and any data sample in this class can be well approximated as the linear combination of a small number of active atoms selected from $\mathbf{D}^c$:

$$\mathbf{x}_i \approx \mathbf{D}^c \boldsymbol{\alpha}_i^c, \ |\boldsymbol{\alpha}_i^c|_0 \leq K, \quad \forall \ y_i = c. \tag{1}$$

where $\boldsymbol{\alpha}_i^c \in \mathbb{R}^{n^c}$ is the sparse code for pixel $\mathbf{x}_i$ with respect to the dictionary $\mathbf{D}^c$; and the ℓ_0 norm counts the number of non-zero elements in the sparse code, which is restricted not to exceed a predefined sparsity level K.

The sparse model in (1) lays the foundation for the SRC classifier [6]. Given a test pixel $\mathbf{x}_i$ with unknown label, we can also sparsely represent it using a concatenated dictionary $\mathbf{D} = \left[\mathbf{D}^1, ..., \mathbf{D}^C\right]$ with $n = \sum_{c=1}^{C} n^c$ atoms. The corresponding concatenated sparse code $\boldsymbol{\alpha}_i \in \mathbb{R}^n$ can be expressed as $\boldsymbol{\alpha}_i = \left[\boldsymbol{\alpha}_i^1; ...; \boldsymbol{\alpha}_i^C\right]$. According to compressive sensing theory [33], when $\boldsymbol{\alpha}_i$ is sparse enough and $\mathbf{D}$ satisfies conditions such as low mutual coherence, we can recover $\boldsymbol{\alpha}_i$ from $\mathbf{x}_i$ by efficiently solving the following ℓ_1 problem, which gives the same sparse solution as solving the combinatorial ℓ_0 problem:

$$\boldsymbol{\alpha}_i = \arg\min_{\mathbf{z}} \|\mathbf{D}\mathbf{z} - \mathbf{x}_i\|_2^2 + \lambda\|\mathbf{z}\|_1, \quad \text{with } \lambda > 0. \tag{2}$$

There exist many convex programming methods to solve (2); the feature sign algorithm in [14] is used in this paper. If pixel $\mathbf{x}_i$ has class label y_i and certain conditions hold on $\mathbf{D}$ (such as block-coherence [34] and sparse subspace clustering property [35]), it is expected that the non-zero coefficients of its sparse code $\boldsymbol{\alpha}_i$ will concentrate in its sub-code corresponding to class y_i, *i.e.*, $\boldsymbol{\alpha}_i^{y_i}$ as defined in (1). Therefore, SRC makes classification decision based on the residual of signal approximated by the sub-code of each class: $r_i^c = \|\mathbf{e}_i^c\|_2^2$, where $\mathbf{e}_i^c = \mathbf{x}_i - \mathbf{D}^c\boldsymbol{\alpha}_i^c$ is the class-wise reconstruction error. The predicted class label is obtained as

$$\hat{y}_i = \arg\min_{c} r_i^c. \tag{3}$$

In this paper, our goal is to find an optimal dictionary $\mathbf{D}^*$ that achieves the best classification on the training data set:

$$\mathbf{D}^* = \arg\min_{\mathbf{D}\in\mathcal{D}} \frac{1}{N}\sum_i I(\hat{y}_i \neq y_i), \tag{4}$$

where $I(\cdot)$ is the indicator function that returns 1 if its operand is true and 0 otherwise, and $\mathcal{D}$ is the $\mathbb{R}^{m\times n}$ matrix space with unit-length columns. Note that Eq. (4) minimizes the overall error rate. It can be easily extended to weighted average of errors for each class, if the numbers of training samples for each class are unbalanced or when we care more about the accuracy for some particular classes than the others.

B. Objective Function with Insights from LVQ

Although directly related to our task of classification, Eq. (4) is not easy to solve. A recent work in [26] applied the LVQ technique to learn the dictionary for SRC, which motivated us to design a more appropriative objective function based on the insight from LVQ.

LVQ [25] is a supervised learning algorithm that generates a codebook optimized for a prototype-based classifier. In testing, LVQ classifies a sample with the same label as the closest prototype in the codebook to it, which is essentially the same as the nearest neighbor classification. During its training process, LVQ (in its simplest version) iteratively goes through each training sample $\mathbf{x}_i$ and moves its nearest prototype $\mathbf{m}_{n(i)}$ towards or away from $\mathbf{x}_i$ based on whether $\mathbf{m}_{n(i)}$ belongs to the same class as $\mathbf{x}_i$:

$$\mathbf{m}_{n(i)} = \begin{cases} \mathbf{m}_{n(i)} + \rho(\mathbf{x}_i - \mathbf{m}_{n(i)}), & \text{if } \mathbf{m}_{n(i)} \text{ has label } y_i \\ \mathbf{m}_{n(i)} - \rho(\mathbf{x}_i - \mathbf{m}_{n(i)}), & \text{otherwise} \end{cases} \tag{5}$$

where $0 < \rho < 1$ is a monotonically decreasing step size.

LVQ shares a common spirit with SRC in several ways. Both of them represent data samples with a subset of elements in a codebook or dictionary, and classify the samples based on the energy distribution in the selected prototypes or atoms. This justifies the attempt in [26] to use updating rules similar to Eq. (5) in learning dictionary for SRC. However, the underlying principles of sparse coding and vector quantization are quite different, which makes the performance of the ad-hoc approach in [26] not guaranteed.

A deeper insight into LVQ has been developed in [36], which regards the learning procedure as a scholastic gradient descent algorithm with a loss function defined on any *misclassified* sample $\mathbf{x}_i$:

$$\mathcal{L}_{LVQ}(\mathbf{x}_i, y_i) \propto \|\mathbf{x}_i - \mathbf{m}^+_{n(i)}\|_2^2 - \|\mathbf{x}_i - \mathbf{m}^-_{n(i)}\|_2^2, \tag{6}$$

where $\mathbf{m}^+_{n(i)}$ and $\mathbf{m}^-_{n(i)}$ are the nearest prototypes to $\mathbf{x}_i$ with label y_i and other than y_i, respectively. We adopt an objective function with a similar form as in (6) so that the merits of LVQ can be exploited in building an SRC dictionary, and hence we refer to this new dictionary design algorithm as the Learning SRC (LSRC). Specifically, a hinge loss function is enforced on each data point:

$$\mathcal{L}_{LSRC}(\mathbf{x}_i, y_i; \mathbf{D}) = \max(0, r_i^{y_i} - r_i^{\hat{c}_i} + b), \tag{7}$$

where $\hat{c}_i$ is the most competitive class in reconstructing the signal excluding the true class y_i:

$$\hat{c}_i = \arg \min_{c \in \{1,\ldots,C\} \setminus y_i} r_i^c, \tag{8}$$

and b is a non-negative parameter controlling the "margin" between the classes. The loss function in (7) is zero when the residual of true class is smaller than any other class by at least an amount

of b. Otherwise, it gives a penalty proportional to the residual difference between the true class and the most competitive "imposter" class. Intuitively, this loss function is also related to the misclassification rate of SRC. Thus, we can formulate the problem of LSRC dictionary design as

$$\mathbf{D}^* = \arg\min_{\mathbf{D}\in\mathcal{D}} \frac{1}{N}\sum_i \mathcal{L}_{LSRC}(\mathbf{x}_i, y_i; \mathbf{D}), \tag{9}$$

which is optimized over the whole training set.

III. Dictionary Optimization for LSRC

A. Online Dictionary Learning

Following the approach used in LVQ, we propose a stochastic gradient descent method to obtain the optimal dictionary in Eq. (9). As opposed to batch mode gradient method, stochastic gradient descent has been favored in dictionary learning for its faster convergence and better scalability to a training set with large or growing size when the objective function is an expectation over all the training samples [15]. We first initialize the dictionary with a reasonable guess $\mathbf{D}^0$ (by performing K-means or unsupervised training on each class individually), and then update it iteratively by going through the whole data set multiple epochs until the convergence. In the t-th iteration, a single sample $(\mathbf{x}_i, y_i)$[1] is drawn from the data set randomly and the dictionary is updated in the gradient direction of its cost term:

$$\mathbf{D}^t = \mathbf{D}^{t-1} - \rho^t \nabla_{\mathbf{D}} \mathcal{L}_{LSRC}(\mathbf{x}, y; \mathbf{D}^{t-1}), \tag{10}$$

where ρ^t is the step size at iteration t, which is updated as $\rho^t = \frac{\rho^0}{\sqrt{(t-1)/N+1}}$ with initial step size ρ^0. With a sufficiently small step size ρ^0, $\mathbf{D}^t$ is guaranteed to converge to a local minimum of (9) as t goes to infinity [36]. The gradient of hinge loss is

$$\nabla_{\mathbf{D}} \mathcal{L}_{LSRC}(\mathbf{x}, y; \mathbf{D}) = \begin{cases} \mathbf{0} & \text{if } r^y - r^{\hat{c}} + b < 0 \\ \text{undefined} & \text{if } r^y - r^{\hat{c}} + b = 0 \\ \nabla_{\mathbf{D}} r^y - \nabla_{\mathbf{D}} r^{\hat{c}} & \text{if } r^y - r^{\hat{c}} + b > 0 \end{cases}. \tag{11}$$

We can ignore the case of undefined gradient, because it occurs with very low probability in practice (only when $r^y - r^{\hat{c}} + b = 0$) and thus it will not affect the convergence of the stochastic

[1]For simplicity, we drop all the data indices i hereafter.

gradient descent as long as a suitable step size is chosen [36]. The same argument could also be obtained if we consider $\mathbf{0}$ as a sub-gradient in the case when the gradient is undefined. Therefore, we only need update dictionary for those samples with $r^y - r^{\hat{c}} + b > 0$.

To evaluate the gradient $\nabla_{\mathbf{D}} r^c$ for a particular class c, we calculate the derivative of r^c with respect to each element d_{ij} of $\mathbf{D}$ as

$$\begin{aligned} \frac{\partial r^c}{\partial d_{ij}} &= -2(\mathbf{e}^c)^T \frac{\partial \mathbf{D}\mathbf{P}_c\boldsymbol{\alpha}}{\partial d_{ij}} \\ &= -2(\mathbf{e}^c)^T \left[\mathbf{P}_c(j,j)\alpha_j \mathbf{u}_i + \mathbf{D}\mathbf{P}_c \frac{\partial \boldsymbol{\alpha}}{\partial d_{ij}} \right], \end{aligned} \tag{12}$$

where $\mathbf{P}_c$ is a $n \times n$ diagonal matrix with 1 at positions corresponding to class c and 0 otherwise, and $\mathbf{u}_i$ is a $m \times 1$ unit column vector with i-th element equal to 1.

B. Finding Sparse Code Derivative

The only thing that remains to be found now is the derivative of sparse code $\boldsymbol{\alpha}$ with respect to the dictionary elements. The sparse code $\boldsymbol{\alpha}$ can be regarded as an implicit function of $\mathbf{D}$, and it is not differentiable everywhere due to the ℓ_1 term in (2). The problem of evaluating sparse derivative has been investigated independently in [24], [37] together with the analysis of differentiability, which we use here directly.

Define the active set of sparse code as $\Lambda = \{j | \alpha_j \neq 0\}$. It has been shown in [24] that $\boldsymbol{\alpha}$ is differentiable with respect to any dictionary atom $\mathbf{d}_j$ with index $j \in \Lambda$. For the other atoms, the gradient is zero with probability 1 and thus can be ignored for the same reason mentioned in the previous subsection. The sparse code derivative with respect to active dictionary atoms can be expressed as

$$\frac{\partial \boldsymbol{\alpha}}{\partial d_{ij}} = \frac{\partial \boldsymbol{\alpha}}{\partial \boldsymbol{\alpha}_\Lambda} \frac{\partial \boldsymbol{\alpha}_\Lambda}{\partial d_{ij}}, \quad \text{if} \quad j \in \Lambda \tag{13}$$

and it is given in [37] that

$$\frac{\partial \boldsymbol{\alpha}_\Lambda}{\partial \mathbf{D}_\Lambda} = -\mathbf{A}^{-1} \frac{\partial [\mathbf{D}_\Lambda^T (\mathbf{D}_\Lambda \boldsymbol{\alpha}_\Lambda - \mathbf{x})]}{\partial \mathbf{D}_\Lambda}, \tag{14}$$

where $\boldsymbol{\alpha}_\Lambda$ and $\mathbf{D}_\Lambda$ denote the sparse coefficients and dictionary columns corresponding to the active set Λ. $\mathbf{A} = \mathbf{D}_\Lambda^T \mathbf{D}_\Lambda$, and in practice we set $\mathbf{A} = \mathbf{D}_\Lambda^T \mathbf{D}_\Lambda + \epsilon \cdot \mathbf{I}$ to ensure the stability of the inverse of $\mathbf{A}$, where ϵ is a small positive constant. $\boldsymbol{\alpha}_\Lambda$ is related to $\boldsymbol{\alpha}$ by $\boldsymbol{\alpha} = \mathbf{P}_\Lambda \boldsymbol{\alpha}_\Lambda$, where $\mathbf{P}_\Lambda \in \mathbb{R}^{n \times |\Lambda|}$, $\mathbf{P}_\Lambda(j,k) = I(j = \psi(k))$, and $\psi(k)$ denotes the k-th element of Λ sorted in ascending order.

Algorithm 1 Dictionary learning with LSRC

Input: labeled data set $\mathcal{S} = \{\mathbf{x}_i, y_i\}$, sparse regularization λ, margin b

Output: dictionary $\mathbf{D}$

initialize $\mathbf{D}$
set $t = 1$
while not converge **do**
  randomly permute data set $\mathcal{S}$
  for each $(\mathbf{x}, y) \in \mathcal{S}$ **do**
    find sparse code $\boldsymbol{\alpha}$ using Eq. (2)
    find $r^c = \|\mathbf{x} - \mathbf{D}^c\boldsymbol{\alpha}^c\|_2^2$ for any $c = 1...C$
    find $\hat{c}$ using Eq. (8)
    if $r^y - r^{\hat{c}} + b > 0$ **then**
      $\mathbf{d}_j \leftarrow \mathbf{d}_j + \Delta\mathbf{d}_j$ for any $j \in \Lambda$ by Eq. (15)
      $\mathbf{d}_j \leftarrow \mathbf{d}_j / \|\mathbf{d}_j\|_2$ for any $j \in \Lambda$
    end if
    $t \leftarrow t + 1$
  end for
end while
return $\mathbf{D}$

After plugging (13) and (12) into (11) and doing some manipulations, we get the update equation for each dictionary atom in the active set Λ as

$$\begin{aligned}\Delta\mathbf{d}_j^t &= \mathbf{d}_j^t - \mathbf{d}_j^{t-1} \\ &= 2\rho^t\Big[\alpha_j I(cls(j) = y)\cdot\mathbf{e}^y - \alpha_j I(cls(j) = \hat{c})\cdot\mathbf{e}^{\hat{c}} \\ &\quad +(\beta^y_{\psi^{-1}(j)} - \beta^{\hat{c}}_{\psi^{-1}(j)})\cdot\mathbf{e} - \alpha_j\mathbf{D}_\Lambda(\boldsymbol{\beta}^y - \boldsymbol{\beta}^{\hat{c}})\Big]. \end{aligned} \quad (15)$$

where $cls(j)$ is the class label for j-th dictionary atom, $\psi^{-1}(\cdot)$ denotes the inverse function of $\psi(\cdot)$, $\mathbf{e} = \mathbf{x} - \mathbf{D}\boldsymbol{\alpha} = \mathbf{x} - \mathbf{D}_\Lambda\boldsymbol{\alpha}_\Lambda$, and $\boldsymbol{\beta}^c = \mathbf{A}^{-1}\mathbf{P}_\Lambda^T\mathbf{P}_c\mathbf{D}^T\mathbf{e}^c$. The resulting dictionary atoms are projected to unit length to ensure $\mathbf{D} \in \mathcal{D}$. The overall procedure of LSRC is summarized in Algorithm 1.

C. Connection with LVQ

Looking at the first two terms in Eq. (15), we can find that they have the effects of "pulling" the active dictionary atoms of correct class towards the signal, and *at the same time* "pushing" the active dictionary atoms of the most competitive wrong class away from the signal. This is exactly what LVQ does in its improved version – LVQ2 [25]. The difference lies in that LVQ2 finds a single nearest neighbor to approximate signal, while our LSRC method employs sparse coding to find multiple atoms and simultaneously updates the atoms selected from the sub-dictionaries of both correct and incorrect classes using the first two terms in (15). The amount of updating is proportional to the sparse coefficient of each atom, *i.e.*, the contribution of each atom in approximating the signal. It should be noted that performing the action of "pushing" is critical in learning a discriminative dictionary, which captures the decision boundary structure between different classes; in contrast, solely performing the action of "pulling" will result in a reconstructive dictionary.

The third and fourth terms in (15) are unique in our approach. They use the overall reconstruction error and every active atom as the ingredients for dictionary update on the active atoms from *all the classes*, which makes sense as the sparse code is jointly determined by all active atoms. As shown latter in the experiments, these terms play a crucial role for our algorithm to achieve stable behavior as well as significantly improved performance over the ad-hoc combination of LVQ with dictionary learning in [26], which is essentially similar as using only the first two terms in (15).

Furthermore, the strategy defined by Eq. (11), which focuses on difficult samples while updating the dictionary, is similar to the LVQ2.1 [25], which performs an update only when the sample falls inside a small "window" around the middle plane between the correct and wrong atoms. Therefore, the choice of hinge loss function as our objective function is also corroborated from the perspective of LVQ2.1.

IV. Spatial-spectral Classification using Patch-based LSRC

So far LSRC has only used the spectral information of each single pixel in designing a discriminative dictionary for classification. In most of the recent HSI classification approaches [28], [29], [31], [32], the spatial correlation between neighboring pixels is exploited together with the spectral information to attain the state-of-the-art results. We can enforce spatial smoothness

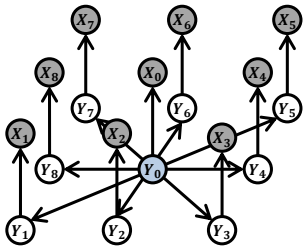


Fig. 1. Graphical model for a 3×3 hyperspectral image patch. There is a statistical dependency between the class label of central pixel X_0 and the label of each other pixel in the neighborhood. Nodes shaded in grey are observable, and the node Y_0 shaded in blue is the variable to be inferred.

in SRC's class prediction map simply by averaging the reconstruction residuals of the pixels within a local neighborhood before making decision for the central pixel:

$$\hat{y}_i = \arg\min_c \sum_{j \in \{i, \mathcal{N}(i)\}} w_j r_j^c. \tag{16}$$

where $\mathcal{N}(i)$ denotes the indices of the neighboring pixels for pixel $\mathbf{x}_i$; and w_j is an averaging weight, which can be obtained from a Gaussian kernel centered at $\mathbf{x}_i$. The spatial smoothness assumption employed by (16) holds well in the homogenous regions of hyperspectral images; however, it usually fails in the elongated regions formed by any single class or on the borders between different classes. The same problem has been encountered by other sparse representation models attempting to capture local spatial distribution of hyperspectral images, such as the joint sparsity constraint used in [7] and the block-wise joint sparsity constraint used in [8]. In the following, we propose a patch-based Bayesian network model capable of describing all kinds of class spatial relationship in a compact form, and it can be efficiently inferred and learned with the sparse representation model under a unified probabilistic framework.

A. Patch-based Bayesian Network

Probabilistic graph models, especially MRF [38], have been applied to encode 2D spatial information for HSI classification in the literature [32], [39]. With pixels in the entire image connected together by a huge graph, the inference of a MRF model usually requires sampling methods such as the Metropolis algorithm used in [32], which are computationally intensive and take a long period of time to converge.

In order to make inference more efficient and scalable, we break the whole image into local patches centered at each test pixel and model each of them using a directed acyclic Bayesian network. In this way, we can infer the label for one test pixel at a time and process them independently.

Specifically, consider a test pixel at image location i represented by random variable X_i. A patch centered at the test pixel X_i is defined by a neighborhood system $\mathcal{N}$ and contains its surrounding pixels $\{X_j\}_{j\in\mathcal{N}(i)}$. We assume that any pixel X is conditionally independent of other pixels given its corresponding class label represented by random variable Y. The labels of surrounding pixels $\{Y_j\}_{j\in\mathcal{N}(i)}$ are conditionally independent given the central test pixel label Y_i. The resultant patch-based Bayesian network can be graphically represented in Fig. 1.

Given all the observed pixels in a patch, the posterior probability of the test pixel label is

$$\begin{aligned} & p(Y_i|X_i,\{X_j\}_{j\in\mathcal{N}(i)}) \\ \propto\ & p(X_i,\{X_j\}_{j\in\mathcal{N}(i)}|Y_i)p(Y_i) \\ \propto\ & p(Y_i)p(X_i|Y_i)\prod_{j\in\mathcal{N}(i)}p(X_j|Y_i) \\ \propto\ & p(Y_i)p(X_i|Y_i)\prod_{j\in\mathcal{N}(i)}\sum_{Y_j}p(X_j|Y_j)p(Y_j|Y_i), \end{aligned} \tag{17}$$

where $p(Y)$ is the class prior, $p(Y_j|Y_i)$ is the conditional label co-occurrence probability, and $p(X_i|Y_i)$ is the observation likelihood. As in Eq. (4), Eq. (17) maximizes the overall likelihood of the inferred pixel label. It is also possible to adjust $p(Y)$ according to the different misclassification costs associated with each class; or we could simply drop $p(Y)$ so that the accuracy of each class is not biased to its frequency. The observation model should give a probabilistic explanation to HSI data with underlying sparse representation, which can be formulated as in [23]:

$$p(X_i=\mathbf{x}|Y_i=c)\propto e^{-\|\mathbf{x}-\mathbf{D}^c\boldsymbol{\alpha}^c\|_2^2},\ \ \forall i \tag{18}$$

where dictionary $\mathbf{D}$ is regarded as a given parameter and sparse code $\boldsymbol{\alpha}$ is a latent variable with Laplace prior: $p(\boldsymbol{\alpha})\propto e^{-\lambda\|\boldsymbol{\alpha}\|_1}$. Eq. (18) provides a probabilistic perspective towards the single-pixel SRC classifier, and $\boldsymbol{\alpha}$ can be estimated in the same way as it is in Eq. (2). The patch-based Bayesian network combines the evidence from all the single-pixel SRC classifiers within the neighborhood, where the spatial relationship is modeled directly through label co-occurrence

probability $p(Y_j|Y_i)$. The class label of the test pixel is predicted as the one giving the maximal posterior probability in (17), and we call this classification scheme *patch-based SRC*. Note that the smoothed SRC prediction defined in (16) can be roughly regarded as a special case of the patch-based SRC, where only the same-class co-occurrence probability $p(Y_j = c|Y_i = c)$ is considered in (17). In our patch-based SRC, a large probability will be assigned to $p(Y_j = c_2|Y_i = c_1)$ if pixels of class c_2 tend to appear around those of class c_1; in this way, pixels of class c_1 will not be merged with c_2, even if they are sparsely distributed and class c_2 has a much more dominant size.

B. Kernel-smoothed Spatial Dependency

If $p(Y_j|Y_i)$ is stationary and only depends on the displacement between the image positions of pixels X_i and X_j, then there should be $|\mathcal{N}(i)|C^2$ parameters[2] to be found in the estimation of $p(Y_j|Y_i)$ for all $j \in \mathcal{N}(i)$. When the neighborhood size $|\mathcal{N}(i)|$ is large, a great amount of training data are required to avoid model overfitting; such data are usually unavailable from one hyperspectral image. We discuss how to regularize and estimate so many co-occurrence probabilities in the following.

Since homogenous regions are dominant in natural hyperspectral images, we can safely assume that the probability of finding a pixel of the same class as the central test pixel will decay gradually as we move away from the test pixel. Following the philosophy in Eq. (16), we use a smoothing kernel to model the same-class co-occurrence probability $p(Y_j = c|Y_i = c)$ as a function of the distance between pixels X_i and X_j. In particular, we choose Gaussian Radial Basis Function (GRBF) as the parametric form for this function:

$$p(Y_j = c|Y_i = c) = \exp\{-\gamma_{c,u}u_{ij}^2 - \gamma_{c,v}v_{ij}^2\}, \tag{19}$$

where u_{ij} and v_{ij} are the distances from X_i to X_j in horizontal and vertical directions, respectively. $\gamma_{c,u}$ and $\gamma_{c,v}$ are positive parameters for GRBF. Note that the range of Eq. (19) is $(0, 1]$, which is consistent with the definition of probability. The co-occurrence probabilities for different classes are defined accordingly as

$$p(Y_j = c'|Y_i = c) = \left[1 - \exp\{-\gamma_{c,u}u_{ij}^2 - \gamma_{c,v}v_{ij}^2\}\right]\xi_{c,c'}, \quad \forall\, c' \neq c, \tag{20}$$

[2] $|\mathcal{N}(i)|C(C-1)$ degrees of freedom

where $\xi_{c,c'} \geq 0$ and $\sum_{c' \neq c} \xi_{c,c'} = 1$. Now the unknown parameter set to encode the spatial dependency becomes $\{\gamma_{c,u}, \gamma_{c,v}, \xi_{c,c'}\}$, and the total number of parameters is reduced to $C(C+1)$. In this way, both of the descriptiveness of Eq. (17) and the compactness of Eq. (16) are maintained in our kernel-smooth spatial dependency model.

Given the class labels $\{y_i\}_{i=1...N}$ and spatial locations of all the pixels in the training set, we can learn the spatial dependency parameters by Maximum Likelihood Estimation (MLE). The log likelihood of all co-occurred class pairs within neighborhood is expressed as

$$\begin{aligned} ll = & \sum_{i=1}^{N} \sum_{\substack{j \in \mathcal{N}(i) \\ y_i = y_j}} \left[-\gamma_{y_i,u} u_{ij}^2 - \gamma_{y_i,v} v_{ij}^2 \right] \\ & + \sum_{i=1}^{N} \sum_{\substack{j \in \mathcal{N}(i) \\ y_i \neq y_j}} \left\{ \log \left[1 - \exp(-\gamma_{y_i,u} u_{ij}^2 - \gamma_{y_i,v} v_{ij}^2) \right] + \log \xi_{y_i,y_j} \right\}. \end{aligned} \tag{21}$$

Under the constraint that $\sum_{c' \neq c} \xi_{c,c'} = 1, \ \forall c$, we can easily obtain the MLE value of $\xi_{c,c'}$:

$$\hat{\xi}_{c,c'} = \frac{< c, c' >}{\sum_{c'' \neq c} < c, c'' >}, \ \ \forall c, c' \neq c, \tag{22}$$

where $< c, c' >$ represents the number of the training samples of class c' falling into the neighborhood around samples of class c.

To find the MLE of $\gamma_{c,u}$ and $\gamma_{c,v}$, we need to maximize the Eq. (21) under the constraints of $\gamma_{c,u} > 0$ and $\gamma_{c,v} > 0$. Barrier method [40] is used to convert this constrained problem into an unconstrained form, and the augmented log likelihood function for the parameters $\{\gamma_{c,u}, \gamma_{c,v}\}$ becomes

$$\begin{aligned} \tilde{ll}_c = & \sum_{\substack{1 \leq i \leq N \\ y_i = c}} \sum_{\substack{j \in \mathcal{N}(i) \\ y_i = y_j}} \left[-\gamma_{c,u} u_{ij}^2 - \gamma_{c,v} v_{ij}^2 \right] \\ & + \sum_{\substack{1 \leq i \leq N \\ y_i = c}} \sum_{\substack{j \in \mathcal{N}(i) \\ y_i \neq y_j}} \left\{ \log \left[1 - \exp(-\gamma_{c,u} u_{ij}^2 - \gamma_{c,v} v_{ij}^2) \right] \right\} + \mu(\log \gamma_{c,u} + \log \gamma_{c,v}), \end{aligned} \tag{23}$$

where $\mu > 0$ is the barrier parameter. Then we can employ the Newton-Raphson method to search the maximum of (23). The first-order derivative of $\tilde{ll}_c$ is

$$\frac{\partial \tilde{ll}_c}{\partial \gamma_{c,u}} = - \sum_{\substack{1 \leq i \leq N \\ y_i = c}} \sum_{\substack{j \in \mathcal{N}(i) \\ y_i = y_j}} u_{ij}^2 + \sum_{\substack{1 \leq i \leq N \\ y_i = c}} \sum_{\substack{j \in \mathcal{N}(i) \\ y_i \neq y_j}} \frac{\exp(-\gamma_{c,u} u_{ij}^2 - \gamma_{c,v} v_{ij}^2) u_{ij}^2}{1 - \exp(-\gamma_{c,u} u_{ij}^2 - \gamma_{c,v} v_{ij}^2)} + \frac{\mu}{\gamma_{c,u}}, \tag{24}$$

and the derivative with respect to $\gamma_{c,v}$ is defined similarly. The second-order derivatives of $\tilde{ll}_c$ are

$$\frac{\partial \tilde{ll}_c}{\partial \gamma_{c,u}^2} = \sum_{\substack{1\le i\le N \\ y_i=c}} \sum_{\substack{j\in\mathcal{N}(i) \\ y_i\neq y_j}} \frac{-\exp(-\gamma_{c,u}u_{ij}^2 - \gamma_{c,v}v_{ij}^2)u_{ij}^4}{[1-\exp(-\gamma_{c,u}u_{ij}^2 - \gamma_{c,v}v_{ij}^2)]^2} - \frac{\mu}{\gamma_{c,u}^2}, \tag{25}$$

$$\frac{\partial \tilde{ll}_c}{\partial \gamma_{c,u}\partial \gamma_{c,v}} = \sum_{\substack{1\le i\le N \\ y_i=c}} \sum_{\substack{j\in\mathcal{N}(i) \\ y_i\neq y_j}} \frac{-\exp(-\gamma_{c,u}u_{ij}^2 - \gamma_{c,v}v_{ij}^2)u_{ij}^2v_{ij}^2}{[1-\exp(-\gamma_{c,u}u_{ij}^2 - \gamma_{c,v}v_{ij}^2)]^2}. \tag{26}$$

$\partial \tilde{ll}_c/\partial \gamma_{c,v}^2$ and $\partial \tilde{ll}_c/(\partial \gamma_{c,v}\partial \gamma_{c,u})$ are defined similarly. Denote the parameter vector $\boldsymbol{\gamma}_c = [\gamma_{c,u}, \gamma_{c,v}]^T$. The Hessian matrix $\nabla^2_{\boldsymbol{\gamma}_c}\tilde{ll}_c$ is negative definite given the form of the second-order derivatives above. Therefore, $\tilde{ll}_c(\boldsymbol{\gamma}_c)$ is a concave function of $\boldsymbol{\gamma}_c$ and the Newton-Raphson method can converge to a global maximum. Starting from an initial value $\boldsymbol{\gamma}_c^{(0)}$, we update $\boldsymbol{\gamma}_c$ iteratively by

$$\boldsymbol{\gamma}_c^{(t+1)} = \boldsymbol{\gamma}_c^{(t)} + [-\nabla^2_{\boldsymbol{\gamma}_c}\tilde{ll}_c(\boldsymbol{\gamma}_c^{(t)})]^{-1}\nabla_{\boldsymbol{\gamma}_c}\tilde{ll}_c(\boldsymbol{\gamma}_c^{(t)}). \tag{27}$$

The MLE parameter $\hat{\boldsymbol{\gamma}}_c$ is obtained upon convergence. As the parameter search is carried out in a 2-dimensional space, given the pixel labels and their spatial locations, our training procedure is very efficient and converges quickly. Once these parameters are learned, the computational overhead in classifying a test pixel is also very small compared to the simple smoothing approach in Eq. (16).

C. Dictionary Learning for Patch-based SRC

With the dictionary learned using the Algorithm 1 and the label co-occurrence probabilities found in the previous subsection, we can conduct patch-based SRC according to Eq. (17). However, the dictionary $\mathbf{D}$ obtained using the Algorithm 1 is optimized for single-pixel SRC, but not for the patch-based SRC. To further improve the classification performance, we customize the Algorithm 1 for the patch-based SRC, leading to a *patch-based LSRC* algorithm, which is derived in the following.

Ignoring the class prior, which is independent of the dictionary $\mathbf{D}$, we find the log likelihood of (17) is equivalent to

$$\log p(X_i|Y_i) + \sum_{j\in\mathcal{N}(i)} \log\left(\sum_{Y_j} p(X_j|Y_j)p(Y_j|Y_i)\right) = \sum_{j\in\{i,\mathcal{N}(i)\}} \log\left(\sum_{Y_j} p(X_j|Y_j)p(Y_j|Y_i)\right), \tag{28}$$

where the equality comes from the fact that $p(Y_i = c'|Y_i = c) = \delta(c - c')$. Applying the same idea of Eq. (7), we want to minimize the log likelihood gap between the true class and the most competitive imposter class if the true class does not have the highest likelihood. The loss function for data sample $(\mathbf{x}_i, y_i)$ at location i with the neighboring pixels $\{\mathbf{x}_j\}_{j\in\mathcal{N}(i)}$ is thus defined as

$$\mathcal{L}(\mathbf{x}_i, y_i; \mathbf{D}) = \max(0, \sum_{j\in\{i,\mathcal{N}(i)\}} \left[\log\left(\sum_{c'} p(X_j = \mathbf{x}_j|Y_j = c')p(Y_j = c'|Y_i = \hat{c}_i)\right)\right.$$
$$\left. - \log\left(\sum_{c'} p(X_j = \mathbf{x}_j|Y_j = c')p(Y_j = c'|Y_i = y_i)\right)\right] + b), \tag{29}$$

where $\hat{c}_i$ is defined as

$$\hat{c}_i = \arg\max_{c\in\{1,\ldots,C\}\setminus y_i} \sum_{j\in\{i,\mathcal{N}(i)\}} \log\left(\sum_{c'} p(X_j = \mathbf{x}_j|Y_j = c')p(Y_j = c'|Y_i = c)\right). \tag{30}$$

The optimal dictionary is obtained by minimizing the total sum of the loss function over all the training data, as we have done in Eq. (9). The same stochastic gradient descent method can be employed here, with only one difference that the gradient $\nabla_{\mathbf{D}} r^c$ is replaced by the gradient of the negative log likelihood:

$$\begin{aligned}
& -\nabla_{\mathbf{D}} \sum_{j\in\{i,\mathcal{N}(i)\}} \log\left(\sum_{c'} p(X_j = \mathbf{x}_j|Y_j = c')p(Y_j = c'|Y_i = c)\right) \\
= & -\sum_{j\in\{i,\mathcal{N}(i)\}} \frac{\sum_{c'} p(Y_j = c'|Y_i = c)\nabla_{\mathbf{D}} p(X_j = \mathbf{x}_j|Y_j = c')}{\sum_{c'} p(Y_j = c'|Y_i = c)p(X_j = \mathbf{x}_j|Y_j = c')} \\
= & \sum_{j\in\{i,\mathcal{N}(i)\}} \frac{\sum_{c'} p(Y_j = c'|Y_i = c)p(X_j = \mathbf{x}_j|Y_j = c')\nabla_{\mathbf{D}} r_j^{c'}}{p(X_j = \mathbf{x}_j|Y_i = c)} \\
= & \sum_{j\in\{i,\mathcal{N}(i)\}} \sum_{c'} p(Y_j = c'|X_j = \mathbf{x}_j, Y_i = c)\nabla_{\mathbf{D}} r_j^{c'}
\end{aligned} \tag{31}$$

where the second equality is obtained by substituting the observation model in (18), and $\nabla_{\mathbf{D}} r_j^{c'}$ can be evaluated in the same manner as in (12). From Eq. (31), it is clear that the patch-based LSRC extends the single-pixel version by considering the residuals of neighboring pixels under all class hypotheses. As to be shown in the experiments, this makes classification more robust if the spatial relationship between pixel labels can be estimated correctly.

V. Experimental Results

A. Experiment Setup

In this section, we examine the classification accuracies of our proposed LSRC and patch-based LSRC algorithms on three benchmark hyperspectral images. Throughout the experiments, we use only 5 atoms per class and obtain dictionaries with sizes of 80 and 45. This is a great reduction with respect to the sizes of the whole training sets, which have an order of magnitude of 1000. With a compactly designed dictionary, our approach requires much less computation in testing and is more scalable to large data sets.

In dictionary learning with the LSRC algorithm, we use the k-means clustering to obtain the initial sub-dictionary for each class, and LSRC usually converges in around 100 iterations. When learning the spatial-spectral dictionaries with the patch-based LSRC, we start from the dictionaries learned with LSRC and convergence is reached in around 20 iterations. We set $\epsilon = 0.001$ and $\mu = 10$ in all the optimization tasks. The neighborhood system $\mathcal{N}$ is a square patch centered at the test pixel. Unless stated otherwise, all the pixels within $\mathcal{N}$ are used for inference including background pixels that do not belong to any class of our interest. This setting is appropriate and realistic for practical applications.

Our method is compared with the various HSI classification approaches. With the same SRC classifier, our learned dictionary (LSRC) is compared with dictionaries obtained from the full training set (Full), the K-means clustering (K-means), the unsupervised training (Unsup) [15], and the ad-hoc LVQ approach (LVQ) [26]. For fair comparison, the same dictionary size is used for all dictionary learning approaches, except for the Full dictionary that uses all the training data. We also compare with SVM classifiers with a linear kernel (SVM) and an RBF-kernel (KSVM), the latter of which is known to give the state-of-the-art results on HSI data [5]. For spatial-spectral classification, the proposed patch-based SRC method is compared with the composite kernel SVM (SVM-CK) [30] and the joint sparsity models (SOMP, SP-S) with the highest accuracies in [7]. Different configurations are tested for our patch-based SRC method: using dictionary optimized by LSRC (LSRC) or patch-based LSRC (pLSRC); with spatial co-occurrence probabilities $p(Y_j|Y_i)$ estimated point-wise without any regularization (suffix -P) or with the kernel-smoothed regularization (suffix -K) as was introduced in Section IV-B. In addition, we also test the smoothed SRC prediction (suffix -S) given in Eq. (16) and apply

TABLE I

THE 16 CLASSES IN THE INDIA PINES IMAGE.

Class No.	Class name	Train	Test
1	Alfalfa	6	48
2	Corn-notill	144	1290
3	Corn-min	84	750
4	Corn	24	210
5	Grass/Pasture	50	447
6	Grass/Trees	75	672
7	Grass/Pasture-mowed	3	23
8	Hay-windrowed	49	440
9	Oats	2	18
10	Soybeans-notill	97	871
11	Soybeans-min	247	2221
12	Soybean-clean	62	552
13	Wheat	22	190
14	Woods	130	1164
15	Building-Grass-Trees-Drives	38	342
16	Stone-steel Towers	10	85
Total		1043	9323

a modified patch-based LSRC to learn a dictionary customized for it[3]. Thus, we have six configurations (LSRC/pLSRC-S/P/K) for the patch-based SRC in total.

In the following experiments, we report for each method the classification accuracy of each class, the overall accuracy (OA), the class-averaged accuracy (AA), and the κ coefficient [3]. The κ coefficient incorporates both of the diagonal and off-diagonal entries in the confusion matrix, which is widely adopted as a robust accuracy measure for remote sensing data.

B. AVIRIS Indian Pines Data Set

We first conduct experiments on the commonly-used Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Indian Pines image [41]. The AVIRIS sensor generates 220 bands across the spectral range from 0.2 to 2.4 μm. Following the same setting as [7], we remove the 20 water absorption bands from the data and obtain a 200-dimensional feature for each pixel. The image

[3]This is simply done by just considering the same-class co-occurrence probability in Eq. (31)

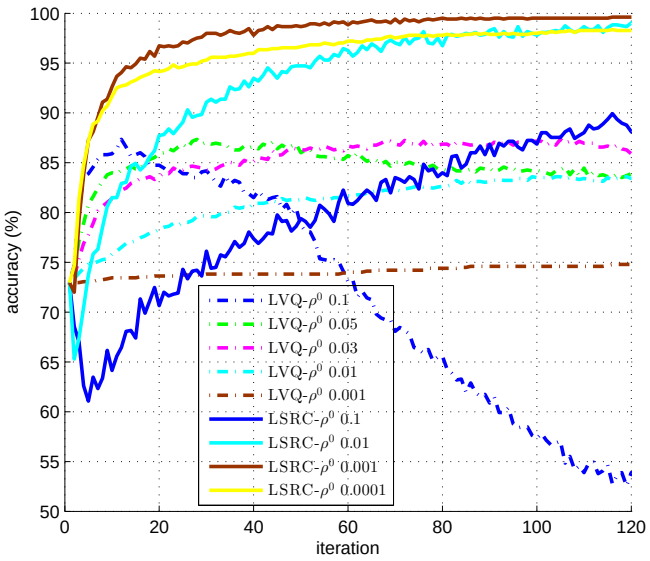


Fig. 2. The overall accuracy on the training set during the training process at each iteration of the LVQ and LSRC algorithms on the Indian Pines image. Different values of the initial step size ρ^0 are tried.

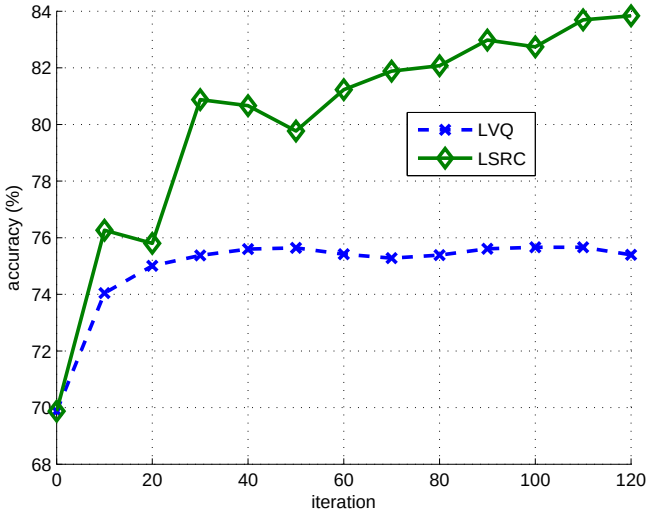


Fig. 3. The OA on the test set during the training process at each iteration of the LVQ and LSRC algorithms on the Indian Pines image. The step size ρ^0 is set as 0.01 for LSRC and 0.03 for LVQ.

has a spatial resolution of 20 m per pixel and a spatial dimension of 145×145. There are 16 labeled classes as summarized in Table I, most of which are crops. We randomly select 10% samples from each class for training and use the remaining 90% for testing. The distribution of the training and test samples in the image are shown in Fig. 6 (a) and (b), respectively.

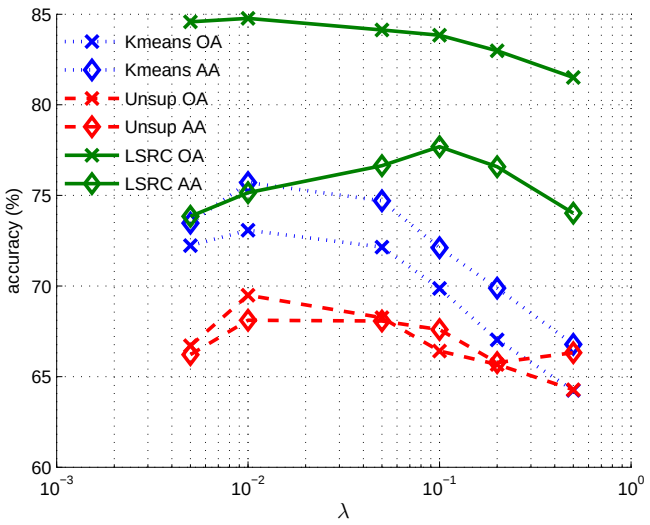


Fig. 4. The effect of the sparsity regularization parameter λ on the test accuracy for the Indian Pines image using the dictionaries obtained by the Kmeans, Unsup, and LSRC dictionary design methods.

The optimization behavior of the LSRC algorithm is first investigated by plotting the accuracies on the training set during the learning iterations in Fig. 2. It is observed that LSRC can always converge to a higher training accuracy with different choices for the initial step size ρ^0. With a suitable step size, the training accuracy quickly converges to almost 100%. We have set ρ^0 between the range of 0.001$\sim$0.01 throughout our experiments. Fig. 2 also shows the learning behavior of LVQ, which achieves a much lower training accuracy than LSRC and may not even converge if the step size is not chosen wisely. This proves that the principled formulation of our LSRC algorithm has resulted in a more effective and stable optimization procedure, although it shares a similar spirit with the ad-hoc LVQ approach. This is further confirmed by the accuracy improvement on the test set during the learning process as shown in Fig. 3.

The sparsity regularization coefficient λ in Eq. (2) is an important parameter in the sparse representation model. Its effect on classification performance using the dictionaries learned with Kmeans, Unsup and LSRC is shown in Fig. 4. Generally, the OAs of all the methods tend to improve as λ decreases up to 0.01. The OA of LSRC is much higher than Kmeans and Unsup, and is less sensitive to the change of λ. In terms of AA, LSRC is also better than the two baseline methods for almost all values of λ. It should be noted that a smaller λ produces less sparse codes and adds more time in solving (2). We set λ as 0.1 or 0.05 throughout our experiments,

TABLE II

THE EFFECT OF THE MARGIN PARAMETER b ON THE ACCURACY FOR THE INDIAN PINES IMAGE USING LSRC.

b	0.0	0.1	0.2	0.3	0.4
Train OA. (%)	99.81	99.14	98.85	98.27	97.60
Test OA. (%)	81.30	83.51	**83.84**	83.19	82.88

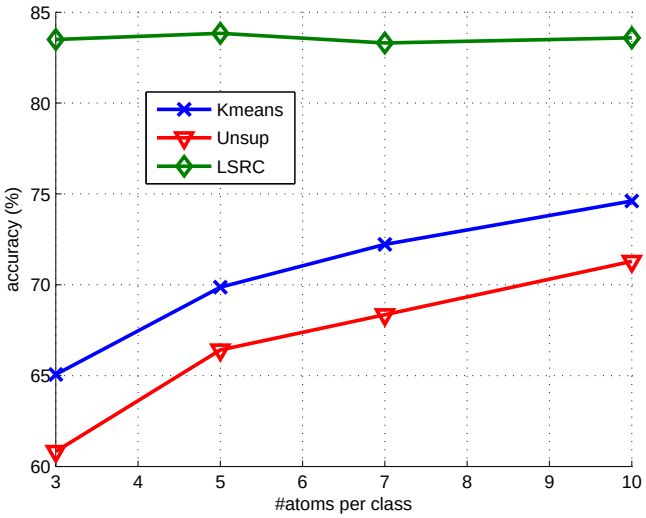


Fig. 5. The effect of dictionary size on the overall accuracy of the Indian Pines image using the Kmeans, Unsup, and LSRC dictionary design methods.

which is observed to give stable and fast solutions.

The effect of the margin parameter b in LSRC is illustrated in Table II. A too small value of b leads to an over-fitting of the training set, while a large value leads towards biasing the goal of classification. b is set as 0.2 or 0.3 in all our experiments in order to achieve a balance between the two factors.

Different dictionary sizes are also tested for Kmeans, Unsup, and LSRC, with the comparison of their accuracies given in Fig. 5. The OAs of Kmeans and Unsup grow as the dictionary size increases, while LSRC maintains a much higher and stabler accuracy for all the sizes. This property allows us to use a compact dictionary (5 atoms per class used throughout the experiments) for LSRC without much compromise in the classification accuracy.

The classification results for all the methods on the Indian Pines image are shown in Fig. 6 (c)-(j), and the accuracies are listed and compared in Table III and Table IV. For the methods

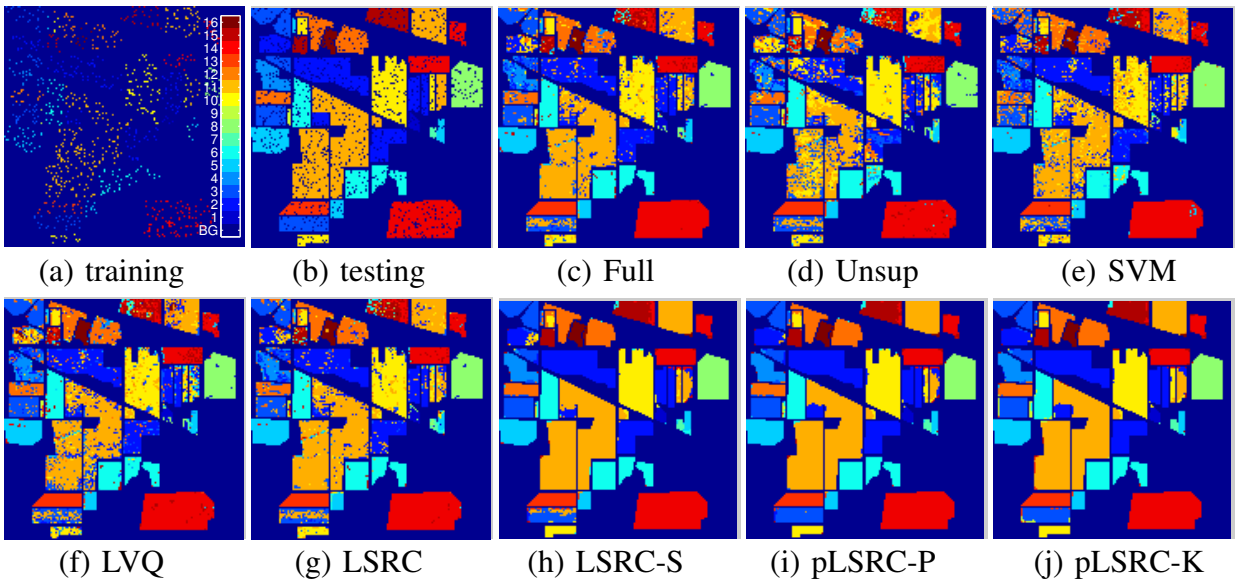


Fig. 6. Classification maps on the Indian Pines image.

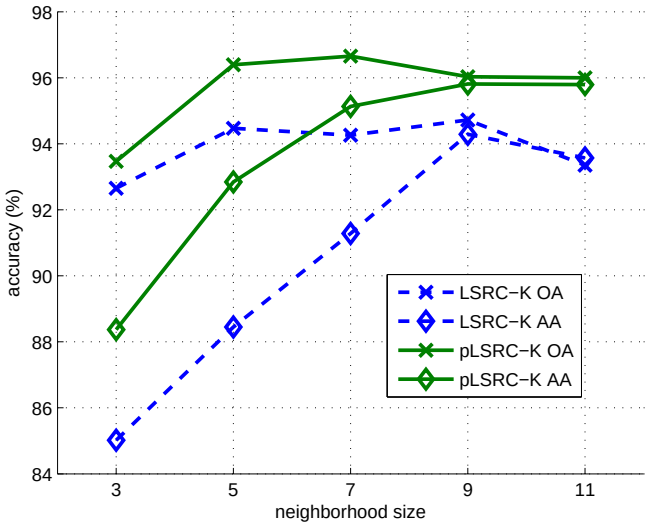


Fig. 7. The effect of neighborhood size on the classification accuracies of the Indian Pines image for LSRC and pLSRC with kernel-smoothed spatial dependency.

using spectral information only (single pixel-based) shown in Table III, the proposed LSRC outperforms most of the other methods by a large margin, and its accuracy is close to the top method KSVM. More importantly, the fixed dictionary size of LSRC offers a great scalability advantage over the nonlinear KSVM when the size of the training set is huge. From Table IV, we find the methods that exploit the spatial-spectral information (patch-based) gain substantial

TABLE III

CLASSIFICATION ACCURACY FOR THE INDIA PINES IMAGE BASED ON SINGLE PIXEL.

Method	Full	K-means	Unsup	LVQ [26]	SVM	KSVM	LSRC
class 1	68.75	68.75	70.83	72.92	56.25	**81.25**	75.00
class 2	80.08	40.39	45.81	69.61	80.00	**86.28**	82.48
class 3	70.40	69.87	65.33	68.40	48.80	**72.80**	68.93
class 4	62.38	**77.14**	52.86	62.38	25.24	58.10	58.57
class 5	91.05	89.26	87.47	89.49	89.49	**92.39**	86.80
class 6	**97.02**	93.15	92.56	93.30	93.01	96.88	96.43
class 7	26.09	17.39	17.39	17.39	13.04	43.48	**47.83**
class 8	97.73	96.59	95.00	97.27	**99.09**	98.86	98.18
class 9	55.56	**66.67**	55.56	61.11	11.11	50.00	38.89
class 10	77.04	83.24	**85.19**	72.90	58.55	71.53	78.53
class 11	83.66	57.14	48.27	64.34	70.42	84.38	**84.65**
class 12	74.64	57.61	57.43	77.36	66.67	**85.51**	83.15
class 13	**100.00**	**100.00**	**100.00**	**100.00**	99.47	**100.00**	**100.00**
class 14	95.36	90.64	88.83	90.21	94.42	93.30	**95.96**
class 15	53.80	49.42	24.85	49.12	56.43	**64.91**	52.34
class 16	92.94	96.47	94.12	**97.65**	85.88	88.24	95.29
OA	82.96	69.87	66.41	75.39	74.44	**84.52**	83.84
AA	76.66	72.11	67.59	73.97	65.49	**79.24**	77.69
κ	0.805	0.662	0.624	0.723	0.708	**0.823**	0.816

improvement over the pixel-based methods. Our patch-based SRC configured with pLSRC-K achieves the best performance in this category, owing to the benefits from both dictionary learning with pLSRC (the improvement of pLSRC-S/P/K over LSRC-S/P/K) and the kernel-smoothed dependency modeling (the improvement of pLSRC-K over pLSRC-S/P). Note that pLSRC-K performs especially well on minority classes (*e.g.*, class 7 and 9), which indicates that the intricate spatial structure around small regions has been correctly learned as opposed to being filtered out by those smoothness-based approaches (SOMP, LSR-S, pLSR-S). The visual comparison shown in Fig. 6 also confirms pLSRC-K achieves better performance on class borders and edge-like class regions.

The size of neighborhood system $\mathcal{N}$ plays an important role in patch-based SRC. As shown in Fig. 7, a too small neighborhood cannot capture enough spatial label information, while a too large neighborhood may bring in noisy information without statistical significance. Nevertheless,

TABLE IV

CLASSIFICATION ACCURACY FOR THE INDIA PINES IMAGE BASED ON LOCAL PATCH.

Method	SOMP [7]	SVM-CK [30]	LSRC-S	LSRC-P	LSRC-K	pLSRC-S	pLSRC-P	pLSRC-K
class 1	85.42	**95.83**	91.67	93.75	91.67	91.67	91.67	91.67
class 2	94.88	96.67	92.64	92.79	93.41	96.20	98.14	**98.60**
class 3	94.93	90.93	85.33	89.73	80.80	90.00	**95.47**	92.53
class 4	91.43	85.71	92.38	91.43	89.05	91.90	92.86	**93.81**
class 5	89.49	**93.74**	91.50	91.28	92.17	91.72	92.39	93.51
class 6	98.51	97.32	**99.85**	**99.85**	99.26	**99.85**	**99.85**	98.51
class 7	91.30	69.57	82.61	91.30	**100.00**	82.61	91.30	**100.00**
class 8	99.55	98.41	**100.00**	**100.00**	**100.00**	**100.00**	99.77	99.77
class 9	0.00	55.56	0.00	16.67	44.44	0.00	44.44	**72.22**
class 10	89.44	93.80	89.90	90.13	92.77	91.73	93.11	**96.10**
class 11	**97.34**	94.37	96.76	96.67	96.40	96.17	95.72	96.13
class 12	88.22	93.66	92.03	94.02	95.83	91.85	94.38	**97.28**
class 13	**100.00**	99.47	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
class 14	**99.14**	**99.14**	98.80	98.71	98.71	98.80	98.37	98.37
class 15	**99.12**	87.43	84.50	84.21	85.96	85.67	90.06	93.57
class 16	96.47	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
OA	95.28	94.86	94.03	94.53	94.26	94.96	96.10	**96.65**
AA	88.45	90.73	87.37	89.41	91.28	88.01	92.35	**95.13**
κ	0.946	0.941	0.932	0.938	0.935	0.943	0.956	**0.962**

under all neighborhood sizes, dictionaries learned with pLSRC give better results than those learned with LSRC. We set the neighborhood size as 7×7 for achieving the best results on this image.

We also compare our classification accuracy with another dictionary learning method proposed in [8], which is called dictionary modeling with spatial coherence (DMS). Following the same experimental settings as in [8], we use half of the samples for training, and use up to 50 atoms for each class sub-dictionary. We achieve overall test accuracies of 96.93%, 98.13% and 99.04%, respectively with algorithms pLSRC-S, pLSRC-P and pLSRC-K; while the DMS method only gives 93.52%. The performance gain is due to the discriminative dictionary optimization as well as the sophisticated spatial dependency modeling in our approach. This experiment also shows that our proposed approach has the potential to achieve even better performance with a larger training data set and a dictionary with more atoms.

TABLE V

THE 9 CLASSES IN THE UNIVERSITY OF PAVIA IMAGE.

Class No.	Class name	Train	Test
1	Asphalt	548	6304
2	Meadows	540	18146
3	Gravel	392	1815
4	Trees	524	2912
5	Metal sheets	265	1113
6	Bare soil	532	4572
7	Bitumen	375	981
8	Bricks	514	3364
9	Shadows	231	795
Total		3921	40002

C. ROSIS Pavia Urban Data Set

The other two hyperspectral images used in our experiments are acquired by the Reflective Optics System Imaging Spectrometer (ROSIS) [42] from the urban area near the University of Pavia and the center of Pavia city, respectively. The ROSIS sensor generates 115 spectral bands ranging from 0.43 to 0.86 μm and has a spatial resolution of 1.3 m per pixel.

The University of Pavia image consists of 610$\times$340 pixels, each having 103 bands with the 12 noisiest bands removed. The description about the nine classes under our consideration are listed in Table V. We follow the same experimental setting for the training and test sets as in [7], using about 9% of all the data for training and the rest for testing. Details about the training and test sets can be found in Table V and Fig. 8 (a) and (b).

The classification results on the University of Pavia image are given in Fig. 8 (c)-(j), and the accuracies are compared in Tables VI and VII. LSRC achieves the best performance in this image among the methods using spectral information only. For spatial-spectral classification, patch-based SRC with dictionaries learned using LSRC already outperforms other competing methods. It is also noticed that the performance difference among the three spatial model configurations is very small. This is because the training data of this image are selected in spatial chunks, and samples from different classes are located away from each other. In this way, a homogeneous spatial relationship is learned by any spatial model with a small 3$\times$3 neighborhood, which is not

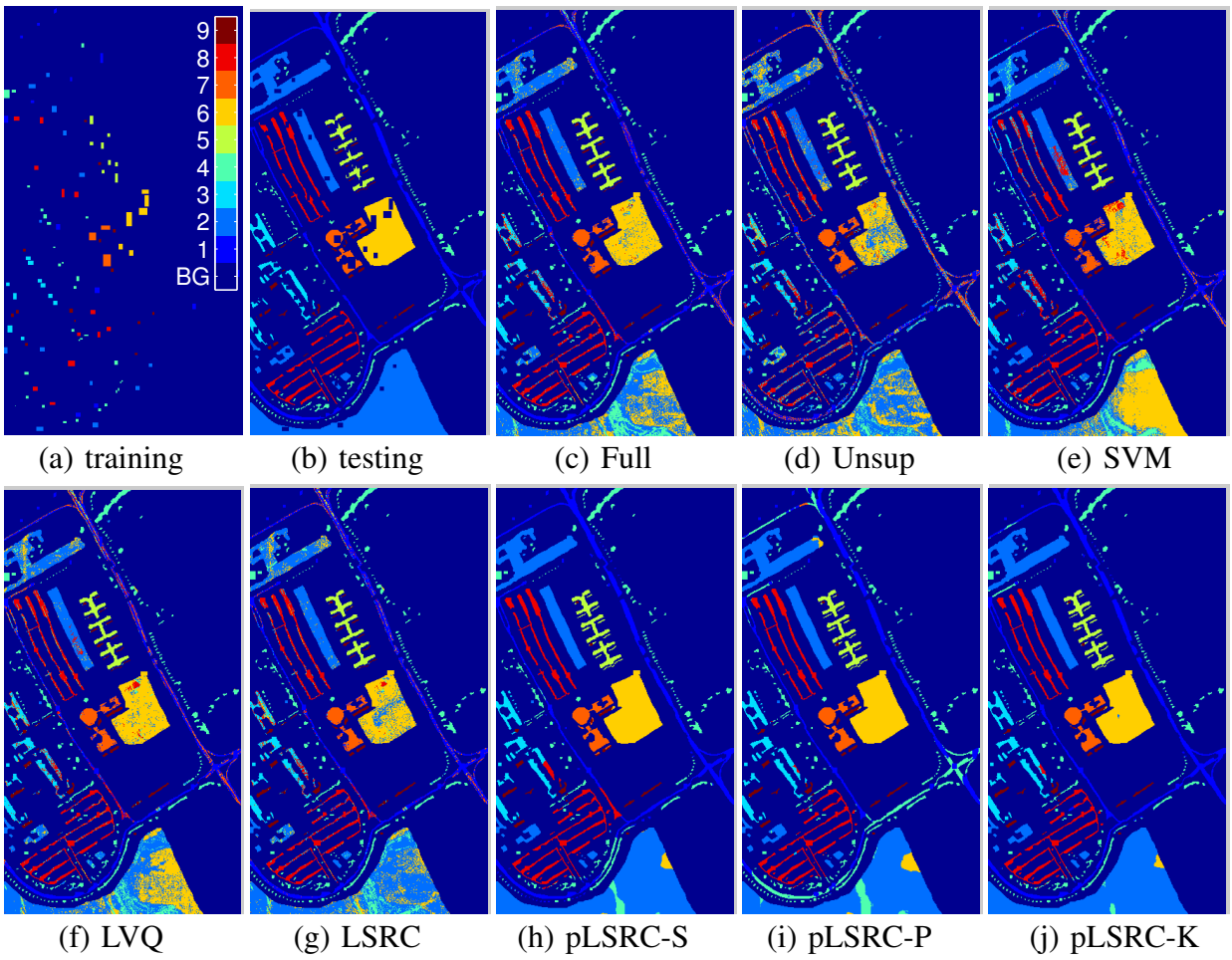


Fig. 8. Classification maps on the University of Pavia image. (h), (i) and (j) are obtained based on local patches excluding background pixels, corresponding to the results in Table. VIII.

consistent with the spatial distribution of testing samples. Since the spatial distributions learned in such a way are incorrect, we do not continue the experiments for pLSRC.

Instead, to better illustrate the difference among the three spatial models, we enlarge the neighborhood size to 21×21 so that more spatial contextual information of the labels can be captured. The counter effect of a large neighborhood is that many background pixels (not belonging to any of the nine classes considered in the experiments) within the neighborhood cannot be well represented by our dictionaries. Therefore, we exclude those background pixels in our dictionary learning and classification. The classification results obtained with background pixels removed are given in Table VIII, which demonstrates the effectiveness of our proposed kernel-smoothed spatial model as well as the pLSRC dictionary learning method when sufficient

TABLE VI

CLASSIFICATION ACCURACY FOR THE UNIVERSITY OF PAVIA IMAGE BASED ON SINGLE PIXEL.

Method	Full	K-means	Unsup	LVQ [26]	SVM	KSVM	LSRC
class 1	81.47	46.92	39.61	68.19	72.94	**84.30**	80.41
class 2	67.18	65.22	66.37	62.99	50.71	67.01	**78.03**
class 3	68.93	55.65	65.40	**70.30**	64.08	68.43	68.32
class 4	**98.80**	86.37	78.67	95.81	95.60	97.80	97.29
class 5	**99.91**	99.82	**99.91**	99.82	**99.91**	99.37	**99.91**
class 6	89.79	75.92	64.94	88.56	83.86	**92.45**	77.06
class 7	87.16	**93.99**	91.64	85.73	79.20	89.91	83.79
class 8	89.83	79.22	67.36	82.19	81.21	**92.42**	86.59
class 9	**97.99**	90.31	71.07	92.58	89.43	97.23	95.97
OA	78.31	68.01	64.57	73.24	67.28	79.15	**81.08**
AA	86.78	77.05	71.66	82.91	79.66	**87.66**	85.26
κ	0.726	0.596	0.549	0.666	0.599	0.737	**0.754**

TABLE VII

CLASSIFICATION ACCURACY FOR THE UNIVERSITY OF PAVIA IMAGE BASED ON LOCAL PATCH.

Method	SP-S [7]	SVM-CK [30]	LSRC-S	LSRC-P	LSRC-K
class 1	83.79	79.85	93.80	**93.86**	**93.86**
class 2	72.35	**84.86**	84.54	84.56	84.56
class 3	71.84	**81.87**	74.27	74.05	74.05
class 4	**98.94**	96.36	98.90	98.90	98.90
class 5	**100.00**	99.37	**100.00**	**100.00**	**100.00**
class 6	92.63	**93.55**	88.21	88.21	88.21
class 7	**91.44**	90.21	91.03	90.93	90.93
class 8	95.57	92.81	**97.86**	**97.86**	**97.86**
class 9	**98.24**	95.35	97.99	97.99	97.99
OA	82.09	87.18	88.97	**88.98**	**88.98**
AA	89.42	90.47	**91.84**	91.82	91.82
κ	0.772	0.833	0.855	**0.855**	**0.855**

knowledge about the class spatial distribution can be inferred from the training data set. In particular, modeling class co-occurrence using Bayesian network (methods with suffix -P, -K) performs much better than simply enforcing spatial smoothness (methods with suffix -S) on small-sized classes (*e.g.*, about 10% higher accuracy on class 9). The classification maps of

TABLE VIII

CLASSIFICATION ACCURACY ON THE UNIVERSITY OF PAVIA IMAGE BASED ON LOCAL PATCH EXCLUDING BACKGROUND PIXELS.

Method	LSRC-S	LSRC-P	LSRC-K	pLSRC-S	pLSRC-P	pLSRC-K
class 1	93.89	63.64	94.48	93.70	66.78	**97.84**
class 2	96.67	91.33	98.34	96.39	92.73	**98.42**
class 3	88.82	88.43	87.93	88.93	**97.30**	95.54
class 4	81.59	94.06	87.74	82.07	**94.20**	87.50
class 5	**100.00**	99.73	**100.00**	**100.00**	99.73	**100.00**
class 6	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	99.74
class 7	**100.00**	98.37	99.29	99.90	98.37	99.29
class 8	99.08	**99.52**	**99.52**	98.90	98.69	98.28
class 9	58.62	68.30	68.55	58.62	68.68	**69.31**
OA	94.78	88.66	96.25	94.65	90.14	**97.03**
AA	90.96	89.26	92.87	90.95	90.72	**93.99**
κ	0.930	0.851	0.950	0.928	0.870	**0.960**

pLSRC with three different spatial models are further compared in Fig. 8 (h)-(j). The distinctive characteristics of the three can be best demonstrated by the bottom left part of the image, where there are some isolated trees (class 4) placed along a narrow and curved path (class 1). The tiny areas of trees are merged with the path in (h) due to the false smoothness assumption held by pLSCR-S. On the other hand, pLSRC-P confuses the path with trees in (i) due to its unreliable estimation of spatial co-occurrences. A more robust estimation of class spatial relationship is obtained with pLSRC-K method, which produces a much better classification result in (j).

The second image collected by the ROSIS sensor, the Center of Pavia, has a spatial dimension of 1096×492, and each pixel has 102 spectral bands after 13 noisy bands being removed. The description of the nine labeled classes and the training/test samples are given in Table IX and Fig. 9 (a) and (b). About 5% of all the data are used for training.

Classification results of the Center of Pavia image are given in Fig. 9 (c)-(j), and the accuracies are compared in Tables X and XI. Again, LSRC achieves the highest accuracy among the single pixel-based methods, and the patch-based SRC methods with the neighborhood size of 3×3 give better results than other competing methods. It is observed that pLSRC cannot make further improvement over the dictionaries given by LSRC, since the accuracy on the training data set

TABLE IX

THE 9 CLASSES IN THE CENTER OF PAVIA IMAGE.

Class No.	Class name	Train	Test
1	Water	745	64533
2	Trees	785	5722
3	Meadow	797	2094
4	Brick	485	1667
5	Soil	820	5729
6	Asphalt	678	6847
7	Bitumen	808	6479
8	Tile	223	2899
9	Shadow	195	1970
Total		5536	97940

already saturates. At the end of the pLSRC learning algorithm, we obtain zero training error and the performance on the test set remains the same or improves very little.

TABLE X

CLASSIFICATION ACCURACY ON THE CENTER OF PAVIA IMAGE BASED ON SINGLE PIXEL.

Method	Full	K-means	Unsup	LVQ [26]	SVM	KSVM	LSRC
class 1	98.79	98.82	98.70	98.96	96.97	**100.00**	98.97
class 2	90.63	87.40	87.64	88.50	91.09	89.74	**94.22**
class 3	**96.66**	92.93	91.98	94.84	96.08	95.70	96.28
class 4	87.46	75.94	81.22	86.68	86.32	**90.40**	87.94
class 5	94.92	91.85	88.83	93.91	88.57	90.59	**97.14**
class 6	95.76	93.53	94.45	93.68	95.27	94.98	**96.47**
class 7	**95.32**	83.49	85.60	90.74	94.03	95.28	94.89
class 8	99.59	99.69	99.62	99.62	**99.83**	99.07	99.69
class 9	**99.59**	98.48	99.54	98.43	95.74	11.88	99.39
OA	97.45	95.86	95.91	96.85	95.68	96.13	**97.93**
AA	95.41	91.35	91.95	93.93	93.77	85.29	**96.11**
κ	0.954	0.925	0.926	0.943	0.923	0.928	**0.962**

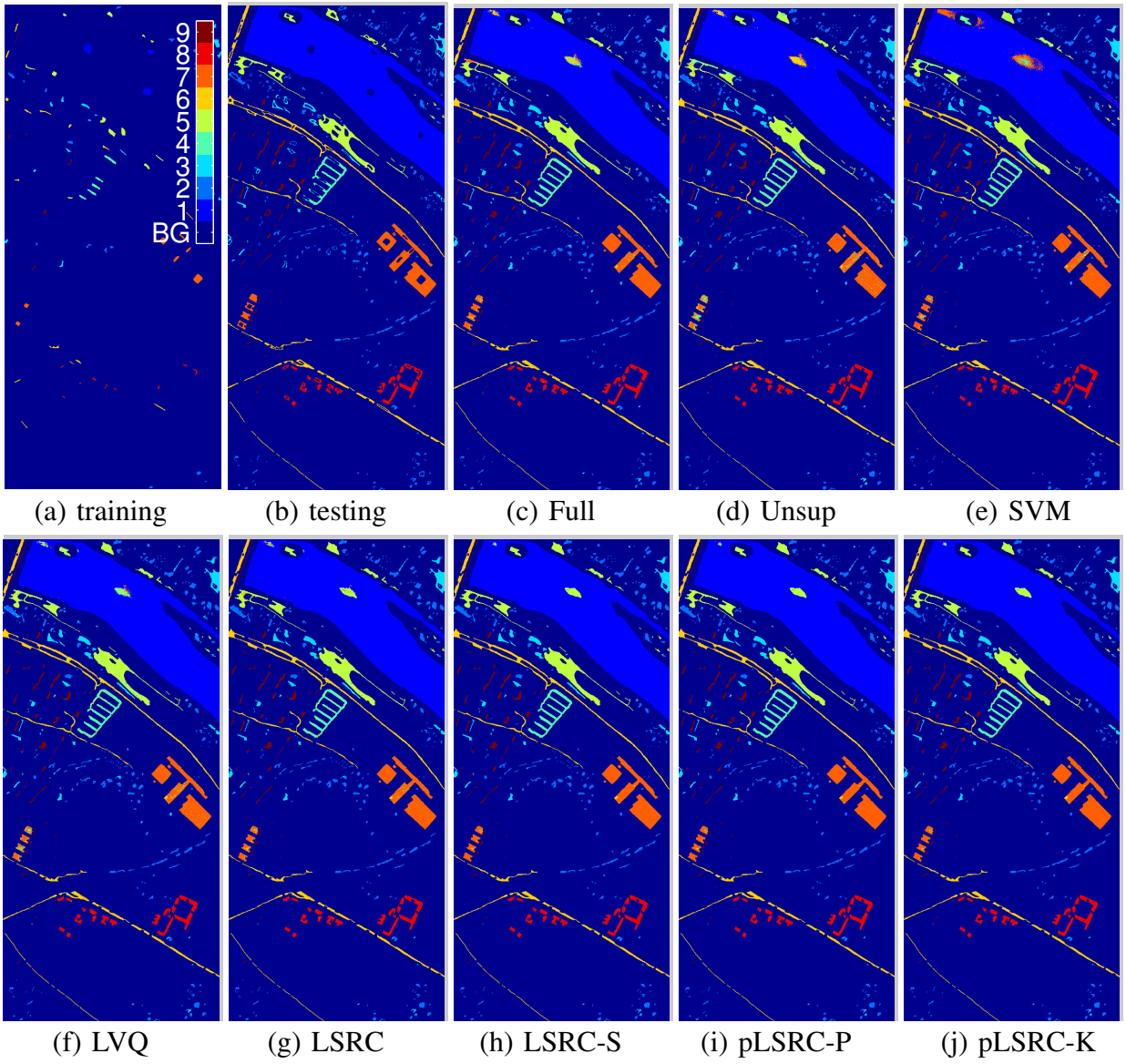


Fig. 9. Classification maps on the Center of Pavia image.

VI. Conclusion

A new discriminative dictionary learning method for the SRC classifier is proposed in this paper, which generates dictionaries with enhanced discriminative power as well as reduced size. The optimization scheme shares the same spirit as LVQ, and we have derived a stochastic gradient descent algorithm which adapts the "pulling" and "pushing" actions of LVQ to sparse coding. We also extended the SRC classifier and the dictionary learning method to spatial-spectral domain so that the spatial dependency among the HSI pixel labels in a local patch is modeled

TABLE XI

CLASSIFICATION ACCURACY ON THE CENTER OF PAVIA IMAGE BASED ON LOCAL PATCH.

Method	SOMP [7]	SVM-CK [30]	LSRC-S	LSRC-P	LSRC-K	pLSRC-S	pLSRC-P	pLSRC-K
class 1	**99.32**	97.46	99.14	99.13	99.13	99.14	99.14	99.14
class 2	92.38	93.08	95.88	95.86	95.86	95.89	**95.93**	**95.93**
class 3	95.46	97.09	**97.66**	**97.66**	**97.66**	**97.66**	97.61	97.61
class 4	85.66	77.02	**97.54**	97.48	97.48	**97.54**	97.36	97.36
class 5	96.37	98.39	99.20	**99.21**	**99.21**	99.20	**99.21**	**99.21**
class 6	92.83	94.32	**99.02**	99.01	99.01	99.01	98.88	98.88
class 7	94.68	97.50	98.07	98.10	98.10	98.09	**98.12**	**98.12**
class 8	99.69	99.83	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
class 9	98.68	**99.95**	99.59	99.59	99.59	99.59	99.64	99.64
OA	97.66	96.81	98.85	98.85	98.85	**98.85**	98.84	98.84
AA	95.01	94.96	98.45	98.45	98.45	**98.46**	98.43	98.43
κ	0.958	0.943	0.979	0.979	0.979	**0.979**	0.979	0.979

and encoded in our dictionary. Smoothing kernels are explored to limit the degrees of freedom on the spatial dependency of neighboring pixel labels, and at the same time to preserve the discriminative information. Experimental results on three hyperspectral images demonstrate that the proposed method is effective under both spectral and spatial-spectral settings; and our learned dictionaries perform substantially better than dictionaries learned using other methods including the ad-hoc adaptation of LVQ in [26]. Comparable or higher accuracies are also observed when our approach is compared with the state-of-the-art SVM classifiers.

The "pulling" and "pushing" actions during the dictionary update implicitly induce a representation with a large margin between the different classes. We will pursuit the margin analysis of the SRC classifier as a future research direction, which may lead to a theoretical guarantee on the classification performance and a guideline to design even better dictionaries.

REFERENCES

[1] D. A. Landgrebe, *Signal theory methods in multispectral remote sensing*, ser. Wiley series in remote sensing. Wiley, 2003.

[2] J. R. Jensen, *Remote Sensing of the Environment: An Earth Resource Perspective*. Pearson Education, 2009.

[3] J. A. Richards and X. Jia, *Remote Sensing Digital Image Analysis: An Introduction*, 4th ed. Springer-Verlag, 2006.

[4] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, pp. 1778–1790, 2004.

[5] G. Camps-valls and L. Bruzzone, "Kernel-based methods for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, pp. 1351–1362, 2005.

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

[7] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification using dictionary-based sparse representation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 10, pp. 3973–3985, 2011.

[8] A. Castrodad, Z. Xing, J. Greer, E. Bosch, L. Carin, and G. Sapiro, "Learning discriminative sparse representations for modeling, source separation, and mapping of hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 11, pp. 4263–4281, 2011.

[9] S. F. Cotter, "Sparse representation for accurate classification of corrupted and occluded facial expressions," in *ICASSP*, 2010, pp. 838–841.

[10] B. C. Haris and R. Sinha, "Sparse representation over learned and discriminatively learned dictionaries for speaker verification," in *ICASSP*, 2012, pp. 4785–4788.

[11] K. Engan, S. O. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," in *ICASSP*, 1999, pp. 2443–2446.

[12] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[13] H. V. Nguyen, V. M. Patel, N. M. Nasrabadi, and R. Chellappa, "Kernel dictionary learning," in *ICASSP*, 2012, pp. 2021–2024.

[14] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *NIPS*, 2007, pp. 801–808.

[15] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009, pp. 689–696.

[16] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *CVPR*, 2010, pp. 2691–2698.

[17] F. Rodriguez and G. Sapiro, "Sparse representations for image classification: Learning discriminative and reconstructive non-parametric dictionaries," University of Minnesota, Tech. Rep., 2007, IMA Preprint 2213.

[18] J. Yang, J. Wang, and T. S. Huang, "Learning the sparse representation for classification," in *ICME*, 2011, pp. 1–6.

[19] J. Yang, K. Yu, and T. S. Huang, "Supervised translation-invariant sparse coding," in *CVPR*, 2010, pp. 3517–3524.

[20] D.-S. Pham and S. Venkatesh, "Joint learning and dictionary construction for pattern recognition," in *CVPR*, 2008.

[21] Z. Jiang, Z. Lin, and L. S. Davis, "Learning a discriminative dictionary for sparse coding via label consistent K-SVD," in *CVPR*, 2011, pp. 1697–1704.

[22] D. Bradley and J. A. D. Bagnell, "Differentiable sparse coding," in *NIPS*, December 2008.

[23] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Supervised dictionary learning," in *Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems*, 2008, pp. 1033–1040.

[24] J. Mairal, F. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 4, 2012.

[25] T. Kohonen, "Improved versions of learning vector quantization," in *IJCNN International Joint Conference on Neural Networks*, vol. 1, 1990, pp. 545–550.

[26] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Discriminative dictionary design using LVQ for hyperspectral image classification," in *IEEE 4th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, 2012.

[27] J. A. Benediktsson, J. A. Palmason, and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, pp. 480–491, 2005.

[28] M. Dalla Mura, J. Atli Benediktsson, B. Waske, and L. Bruzzone, "Morphological attribute profiles for the analysis of very high resolution images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 10, pp. 3747–3762, 2010.

[29] Y. Tarabalka, J. Chanussot, and J. A. Benediktsson, "Segmentation and classification of hyperspectral images using watershed transformation," *Pattern Recognition*, vol. 43, no. 7, pp. 2367–2379, 2010.

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

[31] J. Li, P. R. Marpu, A. Plaza, J. M. Bioucas-Dias, and J. A. Benediktsson, "Generalized composite kernel framework for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. accepted, 2013.

[32] Y. Tarabalka, M. Fauvel, J. Chanussot, and J. A. Benediktsson, "SVM-and MRF-based method for accurate classification of hyperspectral images," *IEEE Geoscience and Remote Sensing Letters*, vol. 7, no. 4, pp. 736–740, 2010.

[33] E. Candes, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on pure and applied mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[34] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3042–3054, 2010.

[35] M. Soltanolkotabi and E. J. Candes, "A geometric analysis of subspace clustering with outliers," *The Annals of Statistics*, vol. 40, no. 4, pp. 2195–2238, 2012.

[36] L. Bottou, "Stochastic learning," in *Advanced Lectures on Machine Learning*, ser. Lecture Notes in Artificial Intelligence, LNAI 3176, O. Bousquet and U. von Luxburg, Eds. Springer Verlag, 2004, pp. 146–168.

[37] J. Yang, Z. Wang, Z. Lin, X. Shu, and T. Huang, "Bilevel sparse coding for coupled feature spaces," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

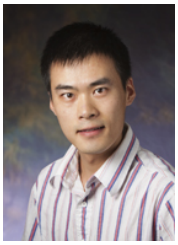
[38] S. Z. Li, *Markov Random Field Modeling in Computer Vision*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1995.

[39] B. C. Tso and P. M. Mather, "Classification of multisource remote sensing imagery using a genetic algorithm and markov random fields," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no. 3, pp. 1255–1260, 1999.

[40] G. B. Dantzig and M. N. Thapa, *Linear Programming 2: Theory and Extensions*. Springer, 2003.

[41] D. Landgrebe, "AVIRIS NW Indiana's Indian Pines 1992 data set," 1992, https://engineering.purdue.edu/~biehl/MultiSpec/hyperspectral.html.

[42] A. Plaza, J. A. Benediktsson, J. W. Boardman, J. Brazile, L. Bruzzone, G. Camps-Valls, J. Chanussot, M. Fauvel, P. Gamba, A. Gualtieri *et al.*, "Recent advances in techniques for hyperspectral image processing," *Remote Sensing of Environment*, vol. 113, pp. S110–S122, 2009.



Zhaowen Wang (S'12) received the B.E. and M.S. degrees in electrical engineering from the Shanghai Jiao Tong University, Shanghai, China, in 2006 and 2009, respectively. He is currently pursuing the Ph.D. degree in the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign (UIUC), Urbana, IL. Since spring 2010, he has been with Department of Electrical and Computer Engineering, UIUC.

His research interests include image super-resolution, image classification, statistical learning, and video events analysis.

Nasser M. Nasrabadi (S'80-M'84-SM'92-F'01) received the B.Sc. (Eng.) and Ph.D. degrees in electrical engineering from the Imperial College of Science and Technology (University of London), London, England, in 1980 and 1984, respectively.

From October 1984 to December 1984, he worked with IBM (UK) as a Senior Programmer. During 1985 to 1986, he worked with Philips Research Laboratory in NY as a Member of the Technical Staff. From 1986 to 1991, he was an Assistant Professor with the Department of Electrical Engineering at Worcester Polytechnic Institute, Worcester, MA. From 1991 to 1996, he was an Associate Professor with the Department of Electrical and Computer Engineering at State University of New York at Buffalo, Buffalo. Since September 1996, he has been a Senior Research Scientist with the US Army Research Laboratory, Adelphi, MD, working on image processing and automatic target recognition. He has served as an Associate Editor for the IEEE Transactions on Image Processing, the IEEE Transactions on Circuits, Systems and Video Technology, and the IEEE Transactions on Neural Networks. He is a Fellow of ARL, SPIE, and IEEE. His current research interests are in hyperspectral imaging, automatic target recognition, statistical machine learning theory, robotics, and neural networks applications to image processing.

Thomas S. Huang (LF'01) received his B.S. Degree in Electrical Engineering from National Taiwan University, Taipei, Taiwan, R.O.C; and his M.S. and Sc.D. Degrees in Electrical Engineering from the Massachusetts Institute of Technology, Cambridge, Massachusetts. He was on the Faculty of the Department of Electrical Engineering at MIT from 1963 to 1973; and on the Faculty of the School of Electrical Engineering and Director of its Laboratory for Information and Signal Processing at Purdue University from 1973 to 1980. In 1980, he joined the University of Illinois at Urbana-Champaign, where he is now William L. Everitt Distinguished Professor of Electrical and Computer Engineering, and Research Professor at the Coordinated Science Laboratory, and at the Beckman Institute for Advanced Science he is Technology and Co-Chair of the Institute's major research theme Human Computer Intelligent Interaction.

Dr. Huang's professional interests lie in the broad area of information technology, especially the transmission and processing of multidimensional signals. He has published 21 books, and over 600 papers in Network Theory, Digital Filtering, Image Processing, and Computer Vision. He is a Member of the National Academy of Engineering; a Member of the Academia Sinica, Republic of China; a Foreign Member of the Chinese Academies of Engineering and Sciences; and a Fellow of the International Association of Pattern Recognition, IEEE, and the Optical Society of America.

Among his many honors and awards are: Honda Lifetime Achievement Award, IEEE Jack Kilby Signal Processing Medal, the King-Sun Fu Prize of the International Association for Pattern Recognition, and the Azriel Rosenfeld Life Time Achievement Award of the International Conference on Computer Vision.