Linear Spectral Mixture Analysis via Multiple-Kernel Learning for Hyperspectral Image Classification

Keng-Hao Liu, *Member, IEEE*, Yen-Yu Lin, *Member, IEEE*, and Chu-Song Chen, *Member, IEEE*

***Abstract*—Linear spectral mixture analysis (LSMA) has received wide interests for spectral unmixing in the remote sensing community. This paper introduces a framework called multiple-kernel learning-based spectral mixture analysis (MKL-SMA) that integrates a newly proposed MKL method into the training process of LSMA. MKL-SMA allows us to adopt a set of nonlinear basis kernels to better characterize the data so that it can enrich the discriminant capability in classification. Because a single kernel is often insufficient to well present all the data characteristics, MKL-SMA has the advantage of providing a broader range of representation flexibilities; it also eases the kernel selection process because the kernel combination parameters can be learned automatically. Unlike most MKL approaches where complex nonlinear optimization problems are involved in their training process, we derived a closed-form solution of the kernel combination parameters in MKL-SMA. Our method is thus efficient for training and easy to implement. The usefulness of MKL-SMA is demonstrated by conducting real hyperspectral image experiments for performance evaluation. Promising results manifest the effectiveness of the proposed MKL-SMA.**

***Index Terms*—Linear spectral unmixing analysis (LSMA), multiple-kernel learning (MKL), spectral unmixing (SU).**

I. Introduction

HYPERSPECTRAL imaging has been a popular topic in the remote sensing community [1]. Recent advances in hyperspectral imaging have been successively applied to many real-world applications such as geology, ecology, agriculture, mineral mapping, land cover classification, chemical, environmental monitoring, and military defense [2]. Opposite to conventional multispectral images that contain tens of discrete bands with a broad bandwidth around 100–200 nm, the hyperspectral images usually consist of hundreds of contiguous bands with fine spectral resolutions that are approximately 10 nm. Owing to the wealth of spectral information collected from the advanced hyperspectral imagine sensors, the hyperspectral image systems have greater potential in data exploration [3]. Although the recently developed sensors may achieve a considerably better resolution in the spatial domain, the ground spatial distance (i.e., the spatial distance covered by a pixel) is still high. As a result, a pixel in the hyperspectral image usually captures the mixture of spectral information of different substances. Classifying or quantifying a so-called "mixed pixel" (or subpixel) is a crucial topic in hyperspectral image processing, which is known as spectral unmixing (SU) [4].

The mixed pixels refer to the image pixels containing different land cover substances or spectral endmembers. Unlike *hard classification* techniques, which assign each pixel to one of the predefined classes, SU is a *soft classification* technique that measures the fractional abundances corresponding to distinct endmembers produced by unsupervised target generation algorithms or given by prior knowledge. Many methods have been developed for SU in the past, such as [3] and [5]–[13]. Among most of them, the theory that has been widely used in the remote sensing community is linear spectral mixture analysis (LSMA) [2], [7], [10]–[13]. LSMA follows the linear mixture model (LMM), which assumes that data samples are linearly mixed by a number of the so-called image endmembers from which they can be further unmixed as abundance fractions of these endmembers. Although the simplest least square approach can achieve LSMA, it could not attain good performance. Over the past years, three representative least-square-based approaches were developed to carry out LSMA. Orthogonal subspace projection (OSP) [8] is an abundance-unconstrained technique for LSMA. Following OSP, nonnegativity constraint least squares (NCLS) [11] and fully constrained least squares (FCLS) [12] were developed for the better abundance estimation. The three methods have shown their effectiveness in remote sensed image classification.

Unfortunately, due to its nature in the inherited constraints, the performance of SU based on LMM, i.e., linear SU, is still limited when the problems are not linearly separable. In order to resolve this dilemma, two branches of nonlinear SU methods have been adopted. Methods in the first branch directly use a nonlinear mixture model (NMM), which considers the scattered light of different materials involved in the mixing process. The intimate spectral mixture [14] was thus proposed to perform nonlinear SU with NMM. Intimate spectral mixture was further investigated in [15] and [16], where neural network approaches were employed to approximate the unknown nonlinear mixing model.

Methods in the second branch, e.g., [17]–[29], are known as kernel-based methods, which generalize linear classification algorithms by nonlinearly mapping data to a high-dimensional

Manuscript received November 18, 2013; revised June 12, 2014; accepted August 30, 2014. This work was supported in part by the Ministry of Science and Technology under Grant 103-2221-E-001-010 and Grant 103-2218-E-110-008 and in part by the Institute for Information Industry under Grant 103-EC-17-A-24-1170.

K.-H. Liu is with the Department of Mechanical and Electromechanical Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan (e-mail: keng3@mail.nsysu.edu.tw).

Y.-Y. Lin is with the Research Center for Information Technology Innovation, Academia Sinica, Taipei 115, Taiwan (e-mail: yylin@citi.sinica.edu.tw).

C.-S. Chen is with the Institute of Information Science and the Research Center for Information Technology Innovation, Academia Sinica, Taipei 115, Taiwan (e-mail: song@iis.sinica.edu.tw).

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

Digital Object Identifier 10.1109/TGRS.2014.2358620

implicit feature space. For example, Camps-Valls and Bruzzone [17], [18] proposed the kernel-based methods for hyperspectral image classification via support vector machine (SVM). Inspired by this trend, LSMA has been also enhanced with its kernel-based version, which is called kernel-based LSMA (KLSMA). Kwon and Nasrabadi [24] first extended OSP to kernel-based OSP (KOSP) for hyperspectral target classification. Later, two techniques, i.e., NCLS and FCLS, have been also extended to their kernel-based counterparts, namely, kernel NCLS (KNCLS) [23], [28] and kernel FCLS (KFCLS) [23], [27], [28], to achieve better SU performance, respectively. KLSMA in general is implemented by casting features in the original space into an implicit feature space via a nonlinear mapping, and thus, it is regarded as a nonlinear SU method.

KLSMA introduces a reproducible kernel in its formulation and performs SU in the high-dimensional feature space so that nonlinearly separable classification problems can be resolved. It has shown noticeable improvement in the classification of multispectral/hyperspectral images [29] and medical images [30]. However, the performance of KLSMA could be limited when a single kernel is insufficient to well represent the whole data. This shortcoming becomes even more evident when addressing increasingly complex unmixing, which means that the variations between pixels are too extensive so that unmixing based on a single similarity measure (a kernel) is generally insufficient. It also imposes the difficulty of selecting an appropriate kernel for a given data set, where trial-and-error-based and empirical settings are commonly adopted for most single-kernel methods. Meanwhile, like traditional kernel methods, KLSMA ignores the fact that the sample values of different dimensions (bands) may require different kernels to produce the best classification results and lacks for a systematic mechanism to choose the appropriate kernel functions. Consequently, the general applicability of KLSMA is still restricted.

Recently, multiple-kernel learning (MKL), which is referred to as learning a kernel machine with a set of basis kernels, has been developed in machine learning society [31]. MKL was developed and originated from solving the optimization problems in SVM. The advantages of MKL are twofold: 1) It can infer the coefficients of combining the basis kernels (or features) directly from the training data, and thus, the kernel selection problem is solved automatically; and 2) it can simultaneously use numerous kernels (or features) to enrich the data similarity representations.

Apart from those SVM-based approaches [32]–[35], there are many extended applications that are also beneficial by MKL, such as fuzzy clustering [36] and dimensionality reduction [37]. Recently, MKL has also shown its availability of improving the conventional single-kernel methods in remote sensed image classification. These studies focus on hard classifications [38], [39] or solve SU under the predeveloped SVM structure [40], [41], where the abundance is estimated by the distance to class boundaries (hyperplanes) without exploiting the spectral mixture model in the feature space. Among them, how to conduct an effective MKL scheme for solving the LSMA problem has not been well explored yet.

In this paper, we propose a new SU method for hyperspectral image classification, which is called MKL-based LSMA (MKL-SMA). Our method integrates MKL into the training process of LSMA. It takes advantages of MKL to enhance the data interpretability and the discriminant capability of the learned LSMA models and leads to better unmixing performance. As a kernel method, MKL-SMA accomplishes SU in a high-dimensional feature space and alleviates the linearly nonseparable problems. Different from KLSMA, the feature space in MKL-SMA is spanned by a set of basis kernels instead of a single kernel. This way, multiple similarity measures corresponding to different kernels are available for depicting the relationship between data samples.

The proposed MKL-SMA possesses the following three main characteristics. First, the flexibility of MKL-SMA allows us to explore more prior knowledge for SU and to employ multiple complementary descriptors to precisely depict the data. Second, we prove that MKL in LSMA has an optimal closed-form solution. Thus, MKL-SMA can be learned very efficiently compared with off-the-shelf MKL algorithms that involve semidefinite programming or other optimization techniques of high computational costs. Third, MKL-SMA is developed in a general way. A family of KLSMA algorithms, such as KLSOSP, KNCLS, and KFCLS, can be generalized to their multikernel versions. LSMA and KLSMA are the special cases of MKL-SMA when only a single basis kernel is used.

In MKL-SMA, users can select any types of KLSMA algorithms as the abundance estimator and select certain types of basis kernels to fulfill the effective SU according to the applications. For performance evaluation, we adopt two real hyperspectral images collected by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) and the Reflective Optics System Imaging Spectrometer (ROSIS) image sensors and demonstrate that MKL-SMA can achieve superior performance than LSMA/KLSMA.

The remainder of this paper is organized as follows. An introduction to LSMA is presented in Section II. Section III describes kernel methods and KLSMA, and Section IV introduces the proposed MKL-SMA. We propose several ways of compiling multiple kernels in Section V. Section VI describes the data and experimental settings. The results conducted on real data are demonstrated in Section VII. Finally, the conclusion is drawn in Section VIII.

II. LSMA

LSMA is a widely used technique to unmix multicomponent composition in remote sensing imagery. It assumes that a spectral pixel is linearly mixed by a number of so-called endmembers or signatures, which are the basic constituents in images and are denoted by $\{\mathbf{s}_i \in \mathbb{R}^d\}_{i=1}^p$, where p and d are the number of endmembers and the spectral dimensionality, respectively. In LMSA, the signature matrix is defined as $\mathbf{S} = [\mathbf{s}_1\ \mathbf{s}_2\ \ldots\ \mathbf{s}_p] \in \mathbb{R}^{d\times p}$, and an input pixel vector $\mathbf{x} \in \mathbb{R}^d$ is supposed to be represented as a linear mixture of signatures, i.e.,

$$\mathbf{x} = \mathbf{S}\boldsymbol{\alpha} + \mathbf{n} \tag{1}$$

where $\mathbf{n} \in \mathbb{R}^d$ accounts for noise or model error, and $\boldsymbol{\alpha} = [\alpha_1\ \alpha_2\ \ldots\ \alpha_p]^\top \in \mathbb{R}^p$ is an unknown p-dimensional abundance

vector with α_i representing the abundance fraction of the signature $\mathbf{s}_i$ in $\mathbf{x}$. Given a set of remote sensing image pixels $D = \{\mathbf{x}_i \in \mathcal{X}\}_{i=1}^N$, LSMA aims to find the abundance vectors $\{\boldsymbol{\alpha}_i\}_{i=1}^N$, one for each pixel, via the following objective function:

$$\min_{\{\boldsymbol{\alpha}_i\}_{i=1}^N} \sum_{i=1}^{N} \|\mathbf{S}\boldsymbol{\alpha}_i - \mathbf{x}_i\|^2. \tag{2}$$

The performance of SU can be further improved by introducing physical constraints in solving (2). Two abundance constraints generally imposed on (2) are the abundance sum-to-one constraints specified by $\|\boldsymbol{\alpha}_i\|_1 = 1$ for $1 \le i \le N$ and the abundance nonnegativity constraints specified by $\boldsymbol{\alpha}_i \ge 0$ for $1 \le i \le N$. In other words, the LSMA unmixes each pixel $\mathbf{x}_i$ by seeking the most plausible abundance vector $\boldsymbol{\alpha}_i$ via p signatures with or without the abundance constraints.

In the development of LSMA, the representative of the unconstrained methods is OSP (LSOSP) [2], [8]. With the above abundance constraints imposed, NCLS [2], [11] and FCLS [2], [12] were successively proposed. They have been widely used for unmixing remote sensed images.

III. Kernel Method and KLSMA

A. Kernel Method

Suppose $D = \{\mathbf{x}_i \in \mathcal{X}\}_{i=1}^N$ is a given data set. Let $\phi : \mathcal{X} \to \mathcal{F}$ denote the implicit feature map that transforms data from input domain $\mathcal{X}$ to a reproduced kernel Hilbert space (RKHS) $\mathcal{F}$ by

$$\mathbf{x}_i \mapsto \phi(\mathbf{x}_i), \qquad \text{for } i = 1, 2, \ldots, N. \tag{3}$$

The corresponding kernel function $k(\cdot, \cdot)$ is employed to efficiently compute the inner products of data in $\mathcal{F}$, without explicitly mapping them to $\mathcal{F}$, i.e.,

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle, \qquad \text{for } 1 \le i; \quad j \le N. \tag{4}$$

Commonly adopted kernel functions include radial basis kernel, polynomial kernel, and sigmoid kernels [21].

In the sequel, we extend the notation of the implicit mapping $\phi(\cdot)$ and kernel $k(\cdot, \cdot)$ by allowing its arguments to be matrices for the ease of introducing MKL-SMA. For matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\phi(\mathbf{A})$ is defined as $[\phi(\mathbf{a}_1)\phi(\mathbf{a}_2) \ldots \phi(\mathbf{a}_n)]$, where $\mathbf{a}_i$ is the ith column of $\mathbf{A}$. For two matrices $\mathbf{A}$ and $\mathbf{B}$, $\mathbf{k}(\mathbf{A}, \mathbf{B})$ is defined as $\phi^\top(\mathbf{A})\phi(\mathbf{B})$. Note that we use $\mathbf{k}(\cdot, \cdot)$ to denote the kernel function with at least one argument to be a matrix.

B. KLSMA

The kernel trick provides the feasibility to implement LSMA in a higher dimensional feature space. KLSMA aims to find the abundance vectors $\{\boldsymbol{\alpha}_i\}_{i=1}^N$ via the objective function

$$\min_{\{\boldsymbol{\alpha}_i\}_{i=1}^N} \sum_{i=1}^{N} \|\phi(\mathbf{S})\boldsymbol{\alpha}_i - \phi(\mathbf{x}_i)\|^2. \tag{5}$$

Three KLSMA approaches, i.e., KOSP (KLSOSP), KNCLS, and KFCLS, were developed accordingly to solve nonlinear problems for remote sensing images. KOSP was first derived in [24] using singular value decomposition. A simpler approach was later proposed in [28] by directly applying the kernel trick to the formulation of OSP. The abundance vectors derived by KOSP/KLSOSP are presented by

$$\begin{aligned} \boldsymbol{\alpha}_i^{\mathrm{KOSP}}(\mathbf{x}_i) &= k(\mathbf{d}, \mathbf{x}_i) - \mathbf{k}(\mathbf{d}, \mathbf{U})\mathbf{k}(\mathbf{U}, \mathbf{U})^{-1}\mathbf{k}(\mathbf{U}, \mathbf{d}) \\ \boldsymbol{\alpha}_i^{\mathrm{KLSOSP}}(\mathbf{x}_i) &= \frac{\boldsymbol{\alpha}_i^{\mathrm{KOSP}}(\mathbf{x}_i)}{k(\mathbf{d}, \mathbf{d}) - \mathbf{k}(\mathbf{d}, \mathbf{U})\mathbf{k}(\mathbf{U}, \mathbf{U})^{-1}\mathbf{k}(\mathbf{U}, \mathbf{d})} \end{aligned} \tag{6}$$

where $\mathbf{d}$ is the target signature of interest, and $\mathbf{U}$ is composed of undesired signatures. Refer to [24] and [28] for the details. KNCLS and KFCLS were developed in [23], [27], and [29]. Since NCLS does not have analytic forms, it needs two iterative equations to find the solutions. To realize KNCLS, the two kernelized equations are

$$\boldsymbol{\alpha}_i^{\mathrm{KNCLS}}(\mathbf{x}_i) = \mathbf{k}(\mathbf{S}, \mathbf{S})^{-1}\mathbf{k}(\mathbf{S}, \mathbf{x}_i) - \mathbf{k}(\mathbf{S}, \mathbf{S})^{-1}\boldsymbol{\lambda} \tag{7}$$

$$\boldsymbol{\lambda} = \mathbf{k}(\mathbf{S}, \mathbf{x}_i) - \mathbf{k}(\mathbf{S}, \mathbf{S})\boldsymbol{\alpha}_i^{\mathrm{KNCLS}}(\mathbf{x}_i). \tag{8}$$

The KNCLS solution can be obtained by substituting (7) and (8) into the NCLS algorithm. As for KFCLS, a new signature matrix $\mathbf{S}' = \begin{bmatrix} \delta\mathbf{S} \\ \mathbf{1}^\top \end{bmatrix}$ and an auxiliary vector $\mathbf{x}_i' = \begin{bmatrix} \delta\mathbf{x}_i \\ 1 \end{bmatrix}$ are employed to replace $\mathbf{S}$ and $\mathbf{x}_i$, respectively, where $\mathbf{1} = [1, 1, \ldots, 1]^\top \in \mathbb{R}^p$, and δ controls the rate of convergence. The implementation details can be found in [27] and [29].

IV. Proposed Approach MKL-SMA

Here, we first give a brief introduction to MKL. Then, our approach to MKL and its integration with KLSMA are respectively described.

A. MKL

Recent advances in MKL, such as [31], have shown that learning SVMs with multiple kernels often increases the accuracy. In contrast to many kernel methods, MKL can offer very nice *a posteriori* interpretation about data characteristics. In most MKL algorithms, an ensemble kernel, which is referred to as a convex combination of the input basis kernels, is derived to fuse the information carried by the basis kernels. Specifically, suppose there are a total of M kernel matrices $\{\mathbf{K}_m\}_{m=1}^M$ available, corresponding to the induced feature maps $\{\phi_m : \mathcal{X} \to \mathcal{F}_m\}_{m=1}^M$. The ensemble kernel $\mathbf{K}$, which is parameterized by kernel weights $\{\gamma_m\}_{m=1}^M$, is defined as

$$\mathbf{K} = \sum_{m=1}^{M} \gamma_m \mathbf{K}_m, \quad \text{s.t. } \gamma_m \ge 0. \tag{9}$$

The task of MKL is to learn a kernel machine and derive the kernel weights $\{\gamma_m\}_{m=1}^M$. Compared with traditional methods using a single kernel, MKL allows us to make more appropriate use of the available data. We could include the prior knowledge to design a set of kernel functions (matrices) to better extract information from the data, such as adopting complementary feature descriptors or employing different types of kernels

with various hyperparameters. Therefore, MKL enhances the interpretability of the decision function and theoretically provides a higher generalization capability in both hard and soft classifications. This paper aims to generalize LSMA/KLSMA to MKL-SMA and leverage the flexibility of MKL to tackle the difficulties of hyperspectral image unmixing.

B. Interpretation of RKHS With Multiple Kernels

Before deriving MKL-SMA, we discover the nonlinear structure of the feature spaces spanned by multiple implicit mappings and the relationship between the implicit mappings and Mercer kernels. Given a data set $D = \{\mathbf{x}_i \in \mathcal{X}\}_{i=1}^N$, we consider to adopt implicit mappings, i.e., $\Phi = \{\phi_m : \mathcal{X} \to \mathcal{F}_m\}_{m=1}^M$, where each mapping ϕ_m projects the data sample $\mathbf{x}_i$ to an RKHS $\mathcal{F}_m$ of dimension L_m by

$$\mathbf{x}_i \mapsto \phi_m(\mathbf{x}_i), \qquad \text{for } i = 1, 2, \ldots, N. \tag{10}$$

According to Mercer's theorem, there exists a kernel function $k_m(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ corresponding to each ϕ_m. That is, for each pair of data $\mathbf{x}_i, \mathbf{x}_j \in D$, we have

$$k_m(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi_m(\mathbf{x}_i), \phi_m(\mathbf{x}_j) \rangle . \tag{11}$$

Since these implicit mappings project data into different spaces, we aim at leveraging diverse data characteristics captured in these spaces for better data analysis.

To combine these kernels for information fusion, we would consider a nonnegative combination of these feature maps $\phi'(\mathbf{x})$, i.e.,

$$\phi'(\mathbf{x}) = \sum_{m=1}^{M} \kappa_m \phi(\mathbf{x}), \qquad \text{with } \kappa_m \geq 0. \tag{12}$$

However, the implicit mappings $\{\phi_m(\cdot)\}_{m=1}^M$ may have different dimensionalities, and thus, such a direct linear combination is infeasible. To address this problem, we consider the Cartesian product space spanned by $\{\mathcal{F}_m\}_{m=1}^M$. More specifically, we construct a new set of implicit mappings $\Psi = \{\psi_m\}_{m=1}^M$ from the original mappings $\Phi = \{\phi_m\}_{m=1}^M$, which maps the data sample $\mathbf{x}$ to the space of a higher dimension L, i.e.,

$$\psi_1 = \begin{bmatrix} \phi_1 \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}, \ \psi_2 = \begin{bmatrix} \mathbf{0} \\ \phi_2 \\ \vdots \\ \mathbf{0} \end{bmatrix}, \ldots, \psi_M = \begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \phi_M \end{bmatrix} \tag{13}$$

where $L = \sum_{m=1}^M L_m$. The new feature maps $\{\psi_m\}_{m=1}^M$ then forms a set of orthogonal bases; they are of the same dimension and have the following property:

$$\langle \psi_m(\mathbf{x}_i), \psi_{m'}(\mathbf{x}_j) \rangle = \begin{cases} k_m(\mathbf{x}_i, \mathbf{x}_j), & \text{if } m = m' \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

Remark: There would be the case where some feature spaces $\phi_m(\cdot)$ have infinite dimensionalities. In this case, we can always interlace the dimensions of $\{\phi_m(\cdot)\}_{m=1}^M$ to form the associated Cartesian product space, so that the resulted bases still fulfill the orthogonal property (14).

We then consider fusing information by seeking a convex combination of these new feature maps, i.e.,

$$\psi(\mathbf{x}) = \beta_1 \psi_1(\mathbf{x}) + \beta_2 \psi_2(\mathbf{x}) + \cdots + \beta_M \psi_M(\mathbf{x})$$
$$\text{s.t.} \sum_{m=1}^{M} \beta_m = 1, \quad \beta_m \geq 0, \quad \text{for } m = 1, 2, \ldots, M. \tag{15}$$

It can be proven that, by considering an arbitrary pair of data, i.e., $\mathbf{x}_i, \mathbf{x}_j \in D$, we have

$$\begin{aligned} \langle \psi(\mathbf{x}_i), \psi(\mathbf{x}_j) \rangle &= \left\langle \sum_{m=1}^{M} \beta_m \psi_m(\mathbf{x}_i), \sum_{m'=1}^{M} \beta_{m'} \psi_{m'}(\mathbf{x}_j) \right\rangle \\ &= \sum_{m=1}^{M} \sum_{m'=1}^{M} \beta_m \beta_{m'} \langle \psi_m(\mathbf{x}_i), \psi_{m'}(\mathbf{x}_j) \rangle \\ &= \sum_{m=1}^{M} \beta_m^2 k_m(\mathbf{x}_i, \mathbf{x}_j). \end{aligned} \tag{16}$$

The $\langle \psi(\mathbf{x}_i), \psi(\mathbf{x}_j) \rangle$ in (16) is equivalent to element (i, j) of the ensemble kernel matrix $\mathbf{K}$, i.e., $\mathbf{K}(i, j)$. Thus, the composite feature map in (15), which is parameterized by the nonnegative kernel weight vector $\boldsymbol{\beta} = [\beta_1 \ \beta_2 \ \ldots \ \beta_M]^\top$, has the corresponding ensemble kernel matrix

$$\mathbf{K} = \beta_1^2 \mathbf{K}_1 + \beta_2^2 \mathbf{K}_2 + \cdots + \beta_M^2 \mathbf{K}_M \tag{17}$$

where $\mathbf{K}_m$ denotes the kernel matrix produced by feature mapping ψ_m.

It follows that our formulation of MKL can exploit the kernel trick to efficiently compute the inner product of data in the space induced by the composite feature map in (15). A simple illustration of the induced spaces of MKL-SMA is shown in Fig. 1(a).

C. MKL-SMA and Its Optimization

We here generalize the LSMA/KLSMA techniques in Sections II and III to deal with multiple kernels simultaneously, which is referred to as MKL-SMA. This framework consists of two stages, namely, the training and testing stages. Fig. 1(b) shows the flowchart of the proposed MKL-SMA.

1) Training Stage: Given a set of training samples $D_t = \{\mathbf{x}_i\}_{i=1}^N$ and a set of feature mappings $\Psi = \{\psi_m\}_{m=1}^M$, our goal in the training stage is to determine the optimal kernel weight vector $\boldsymbol{\beta}$ in (15) and the abundance vectors $\{\boldsymbol{\alpha}_i\}_{i=1}^N$. By integrating (15) into the objective function of LSMA in (2), it leads to the following constrained optimization problem:

$$\begin{aligned} \min_{\{\boldsymbol{\alpha}_i\}_{i=1}^N, \boldsymbol{\beta}} \quad & \sum_{i=1}^{N} \|(\psi(\mathbf{S})\boldsymbol{\alpha}_i - \psi(\mathbf{x}_i))\|^2 \\ \text{s.t.} \quad & \psi(\mathbf{x}_i) = \sum_{m=1}^{M} \beta_m \psi_m(\mathbf{x}_i), \qquad \text{for } i = 1, 2, \ldots, N \\ & \sum_{m=1}^{M} \beta_m = 1, \quad \beta_m \geq 0, \qquad \text{for } m = 1, 2, \ldots, M. \end{aligned} \tag{18}$$

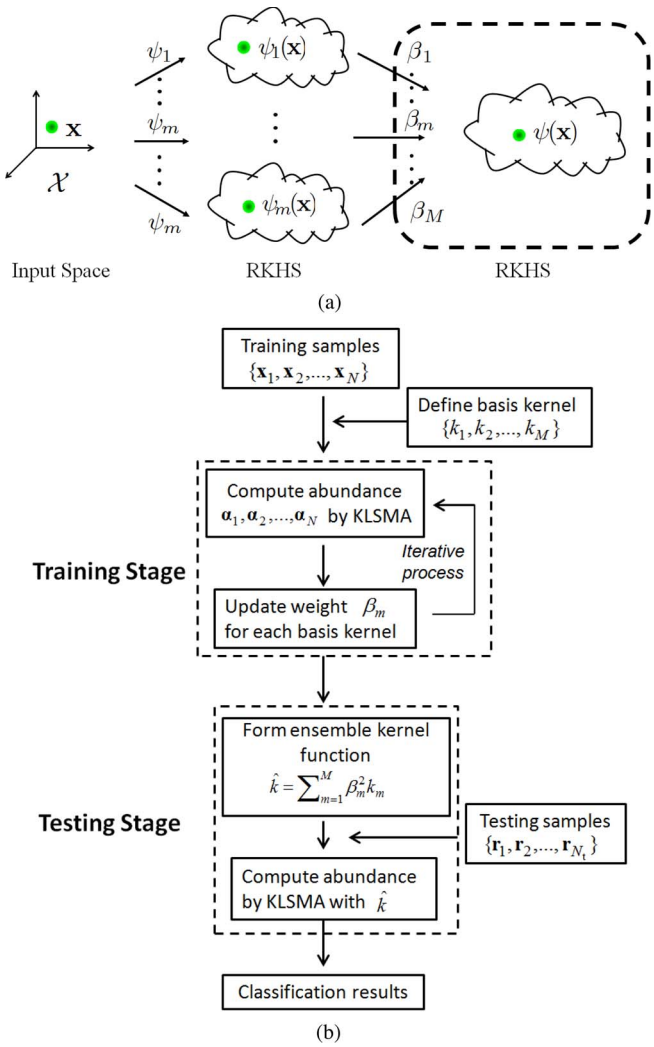


Fig. 1. (a) Three kinds of spaces in MKL-SMA: the input space of spectral pixels, the RKHS induced by each basis kernel, and the new RKHS induced by the ensemble kernel. (b) Flowchart of the MKL-SMA algorithm.

Since direct optimization of (18) is difficult, we instead adopt an iterative two-step strategy to alternately optimize the abundance vectors $\{\boldsymbol{\alpha}_i\}_{i=1}^N$ and the kernel weight vector $\boldsymbol{\beta}$. At each iteration, one of $\{\boldsymbol{\alpha}_i\}_{i=1}^N$ and $\boldsymbol{\beta}$ is optimized, whereas the other is fixed, and then, their roles are switched. Iterations are repeated until the convergence of the objective values.

On optimizing abundance vectors $\{\boldsymbol{\alpha}_i\}_{i=1}^N$*:* By fixing $\boldsymbol{\beta}$, the feature map ψ and its corresponding kernel are then determined as well, namely, $\mathbf{K} = \sum_{m=1}^M \beta_m^2 \mathbf{K}_m$, conducted in (17). Furthermore, all the constraints in (18) are irrelevant to the abundance vectors $\{\boldsymbol{\alpha}_i\}_{i=1}^N$. Thus, the optimization problem in (18) is reduced to the single-kernel LSMA problem and can be effectively solved by KLSMA.

Since the prior knowledge about the weights of basis kernels is generally unavailable, in the beginning of the training stage, we simply assume that they are uniformly distributed, i.e., $\beta_1 = \beta_2 = \cdots = \beta_M = 1/M$. It gives stable performance, and the optimization procedure converges after only a few iterations in our experiments.

On optimizing kernel weight vector $\boldsymbol{\beta}$*:* In the other case of optimizing $\boldsymbol{\beta}$ with fixed $\{\boldsymbol{\alpha}_i\}_{i=1}^N$, it is shown that a closed-form solution can be derived and efficiently computed. By expanding composite feature map ψ in (18), the objective function can be represented as

$$\left\| \sum_{m=1}^{M} \beta_m \mathbf{v}_m \right\|^2 \tag{19}$$

where

$$\mathbf{v}_m = \sum_{i=1}^{N} \left(\psi_m(\mathbf{S})\boldsymbol{\alpha}_i - \psi_m(\mathbf{x}_i) \right). \tag{20}$$

Since $\langle \mathbf{v}_m, \mathbf{v}'_m \rangle = 0$ if $m \neq m'$ by jointly considering (14) and (20), it follows that

$$\left\| \sum_{m=1}^{M} \beta_m \mathbf{v}_m \right\|^2 = \sum_{m=1}^{M} \|\beta_m \mathbf{v}_m\|^2. \tag{21}$$

By substituting the derived objective function (20) into the constrained optimization problem (18), it results in

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & c_1\beta_1^2 + c_2\beta_2^2 + \cdots + c_M\beta_M^2 \\ \text{s.t.} \quad & \sum_{m=1}^{M} \beta_m = 1, \\ & \beta_m \geq 0, \qquad \text{for } m = 1, 2, \ldots, M \end{aligned} \tag{22}$$

where

$$\begin{aligned} c_m &= \langle \mathbf{v}_m, \mathbf{v}_m \rangle \\ &= \sum_{j=1}^{p} \alpha_j^* \left\{ \sum_{j'=1}^{p} \alpha_{j'}^* k_m(\mathbf{s}_j, \mathbf{s}_{j'}) - 2 \sum_{i=1}^{N} k_m(\mathbf{s}_j, \mathbf{x}_i) \right\} \\ &\quad + \sum_{i=1}^{N} \sum_{j=1}^{N} k_m(\mathbf{x}_i, \mathbf{x}_j), \quad \text{for } m = 1, 2, \ldots, M. \end{aligned} \tag{23}$$

In (23), $\alpha_j^* = \sum_{i=1}^N \alpha_{ij}$, and α_{ij} is the jth element of $\boldsymbol{\alpha}_i$.

The resulting optimization problem (22) is an instance of the quadratic programming (QP) problem. Although there are a number of off-the-shelf QP solvers, we can derive the optimal closed-form solution to (22) by exploiting Cauchy–Schwarz inequality. Thus, the proposed approach does not suffer from the high computational cost or numerical approximation of the QP solvers.

Consider two vectors $\mathbf{a} = [\sqrt{c_1}\beta_1 \ \sqrt{c_2}\beta_2 \ \ldots \ \sqrt{c_M}\beta_M]^\top$ and $\mathbf{b} = [(1/\sqrt{c_1}) \ (1/\sqrt{c_2}) \ \ldots \ (1/\sqrt{c_M})]^\top$. By introducing the Cauchy–Schwarz inequality $\|\mathbf{a}\|^2\|\mathbf{b}\|^2 \geq (\mathbf{a}^\top\mathbf{b})^2$, we have

$$\begin{aligned} &\left(c_1\beta_1^2 + c_2\beta_2^2 + \cdots + c_M\beta_M^2\right)\left(\frac{1}{c_1} + \frac{1}{c_2} + \cdots + \frac{1}{c_M}\right) \\ &\qquad \geq (\beta_1 + \beta_2 + \cdots + \beta_M)^2 = 1. \end{aligned} \tag{24}$$

Since the term $((1/c_1)+(1/c_2)+\cdots+(1/c_M))$ is fixed, the lower bound of the objective function $(c_1\beta_1^2+c_2\beta_2^2+\cdots+c_M\beta_M^2)$ is $1/((1/c_1)+(1/c_2)+\cdots+(1/c_M))$. It is clear that the equality in (24) holds if

$$c_1\beta_1 = c_2\beta_2 = \cdots = c_M\beta_M. \tag{25}$$

By taking (25) and the constraint $\sum_{m=1}^{M}\beta_m = 1$ into account jointly, the optimal solution to (22) is

$$\beta_m = \frac{\frac{1}{c_m}}{\frac{1}{c_1}+\frac{1}{c_2}+\cdots+\frac{1}{c_M}}, \qquad \text{for } m = 1, 2, \ldots, M. \tag{26}$$

That is, the optimal β_m is the harmonic mean of $\{c_i\}_{m=1}^{M}$. In addition, β_m is nonnegative because c_m defined in (23) is nonnegative. $\boldsymbol{\beta} = [\beta_1\ \beta_2\ \ldots\ \beta_M]^\top$ in (26) is thus the optimal solution to (22).

2) Testing Stage: After completing the training stage, it is ready to apply the learned model to predict the unseen testing samples. We first construct the ensemble kernel function based on the learned kernel weight vector $\boldsymbol{\beta} = [\beta_1\ \beta_2\ \ldots\ \beta_M]^\top$, i.e., $\hat{k}(\cdot,\cdot) = \sum_{m=1}^{M}\hat{\beta}_m^2 k_m(\cdot,\cdot)$. For each testing sample, the SU is performed by any abundance estimator with the ensemble kernel $\hat{k}(\cdot,\cdot)$. In this paper, we use KLSOSP [29], KNCLS [23], [29], and KFCLS [23], [29] introduced in Section III to serve as the abundance estimators in the experiments, respectively.

We summarize the training and testing procedures for the proposed MKL-SMA in Algorithms 1 and 2, respectively.

Algorithm 1 Training Procedure of MKL-SMA

Input: Basis kernels $\{k_m(\cdot,\cdot)\}_{m=1}^{M}$ and training set $\{\mathbf{x}_i\}_{i=1}^{N}$
1: **Initialization**: $\beta_1 = \beta_2 = \cdots = \beta_M = 1/M$.
2: Construct ensemble kernel $k(\cdot,\cdot) = \sum_{m=1}^{M}\beta_m^2 k_m(\cdot,\cdot)$;
3: Compute abundance vectors $\{\boldsymbol{\alpha}_i\}_{i=1}^{N}$ by using KLSMA with $k(\cdot,\cdot)$ and $\{\mathbf{x}_i\}_{i=1}^{N}$;
4: Update kernel weights $\{\beta_m\}_{m=1}^{M}$ via (26);
5: Check convergence. If the value of the objective function does not change, denote $\{\hat{\beta}_m = \beta_m\}_{m=1}^{M}$. Otherwise, go to step 2;
Output: Optimized kernel weights $\{\hat{\beta}\}_{m=1}^{M}$

Algorithm 2 Testing Procedure of MKL-SMA

Input: Basis kernels $\{k_m(\cdot,\cdot)\}_{m=1}^{M}$ with weights $\{\hat{\beta}\}_{m=1}^{M}$ and testing set $\{\mathbf{r}_i\}_{i=1}^{N_t}$
1: Construct the optimized ensemble kernel $\hat{k}(\cdot,\cdot) = \sum_{m=1}^{M}\hat{\beta}_m^2 k_m(\cdot,\cdot)$;
2: Compute abundance vectors $\{\boldsymbol{\alpha}_i^t\}_{i=1}^{N_t}$ by using KLSMA with $\hat{k}(\cdot,\cdot)$ and $\{\mathbf{r}_i\}_{i=1}^{N_t}$;
Output: Abundance fractions $\{\boldsymbol{\alpha}_i^t\}_{i=1}^{N_t}$

V. Basis Kernel Construction

The performance of conventional kernel methods crucially relies on the features extracted from the data. We often put emphasis on the design of feature extractors and use all the features to build a better kernel matrix. In the cases of MKL, we need not only to design the features but also to construct the basis kernels according to our prior knowledge about the problem. Although MKL methods can automatically figure out appropriate kernel combinations, their accuracies and generalization capabilities still depend on whether informative and complementary basis kernels are provided. In addition, the basis kernels with larger learned weights are usually more important. Such a property can be further exploited for feature redesigning and ranking. In what follows, three ways for kernel construction are proposed and discussed. We build the basis kernels with different hyperparameter values in the first way and design the basis kernels by investigating the characteristics of the hyperspectral images in the last two ways.

A. DHV Kernels

MKL algorithms were originally designed to work on the basis kernels constructed with different values of the hyperparameter. In this paper, we use radial basis function (RBF) kernels in our implementation, but other types of kernels can be used in our framework as well.

Let $\mathbf{x}_i$ and $\mathbf{x}_j$ be two arbitrary hyperspectral data samples. The resulting different hyperparameter valued (DHV) basis kernels are

$$k_m^{\mathrm{DHV}}(\mathbf{x}_i,\mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|^2}{\sigma_m^2}\right), \qquad \text{for } m = 1, 2, \ldots, M \tag{27}$$

where $\{\sigma_m\}_{m=1}^{M}$ are M different hyperparameter values. Like previous MKL approaches, the whole data features are used in the construction of all the basis kernels. Except for supporting multiscale data similarities, one main advantage of using DHV kernels is that model selection is integrated into training.

However, the DHV basis kernels tend to induce similar feature spaces since they take identical data features into account with only the hyperparameters varying. The large redundancy among basis kernels would result in less performance improvement. To avoid this situation, the selected basis kernels should be more informative and complementary.

B. SS Kernels

In the above, the whole spectral features (bands) are used to build basis kernels, and we referred them to as spectral kernels. Since hyperspectral sensors are probably interfered by many extrinsic environmental factors, the collected spectral pixels usually contain a certain level of random noise that may reduce the accuracy of SU, particularly for the pixel-based classification techniques without denoising. To alleviate such a problem, we consider the spatial continuity of class labels in hyperspectral images and leverage the information of spatial correlation to filter out the unfavorable random noise.

Inspired by the idea in [18], [25], and [42], where a single *stacked* kernel was established by concatenating all spatial contextual information, we instead compile a set of spectral–spatial (SS) kernels, each of which carries distinct spatial information. By so doing, we can work with different types of spatial information without predefining their relatively importance nor dealing with the scaling issue in feature concatenation.

Specifically, we use the union of two sets of kernels, i.e., the spectral and spatial kernels, as the bases, i.e.,

$$\left\{k_m^{\mathrm{SS}}(\mathbf{x}_i,\mathbf{x}_j)\right\}_{m=1}^{M_\mu+M_s} = \left\{k_m(\mathbf{x}_i,\mathbf{x}_j)\right\}_{m=1}^{M_\mu} \bigcup \left\{k_m^s\left(\mathbf{x}_i^s,\mathbf{x}_j^s\right)\right\}_{m=M_\mu+1}^{M_\mu+M_s}. \quad (28)$$

While the former set takes the spectral pixels to be predicted into account, the latter set considers their surrounding neighbors. Suppose $\mathbf{x}_i$ is the feature representation of pixel i. $\mathbf{x}_i^s$ is the mean of its spatial neighbors. M_μ and M_s are the numbers of basis kernels (obtained by varying the parameter values or the neighborhood sizes) in the two sets, respectively. The main virtue of SS kernels is that MKL algorithms would balance the contributions provided by spectral and spatial statistics, and the trained classifier is expected to be more resistant to noise.

C. PSR Kernels

It was mentioned in [33] and [39] that each basis kernel may either use the full set or subsets of variables describing the sample vector. To construct a set of basis kernels that are complementary to each other, we further assume that different spectral bands capture distinct information for SU. In practice, we consider that each basis kernel occupies one particular spectral dimensionality so that the information carried by the basis kernels is independent.

Let $\mathbf{x}_i^l$ and $\mathbf{x}_j^l$ be the responses of $\mathbf{x}_i$ and $\mathbf{x}_j$ in band l, respectively. We then establish a kernel bank with L basis kernels by

$$\left\{k_m^{\mathrm{PSR}}\left(\mathbf{x}_i^l,\mathbf{x}_j^l\right)\right\}_{l=1}^{L}. \quad (29)$$

The kernel weights learned by partial spectral range (PSR) kernel can be used for the studies about the importance of spectral features in SU. Moreover, the usage of PSR kernels also can be extended to feature selection. The analogous idea was also mentioned in [39].

VI. Data Sets and Experimental Settings

A. Hyperspectral Data Sets Used for Experiments

The first data set[1] used for experiments is the real hyperspectral image collected by the ROSIS optical sensor over an urban area of the University of Pavia, Italy, on July 8, 2002. The Pavia image is of the size 610 × 340 with very high spatial resolution about 1.3 m per ground pixel. The original data contain 115 spectral bands ranging from 0.43 to 0.86 μm. After removing the noisy bands, the remaining 103 bands are used for the experiments. Fig. 2(a) and (b) shows the image of band 80 and the ground truth map of all target classes, respectively.

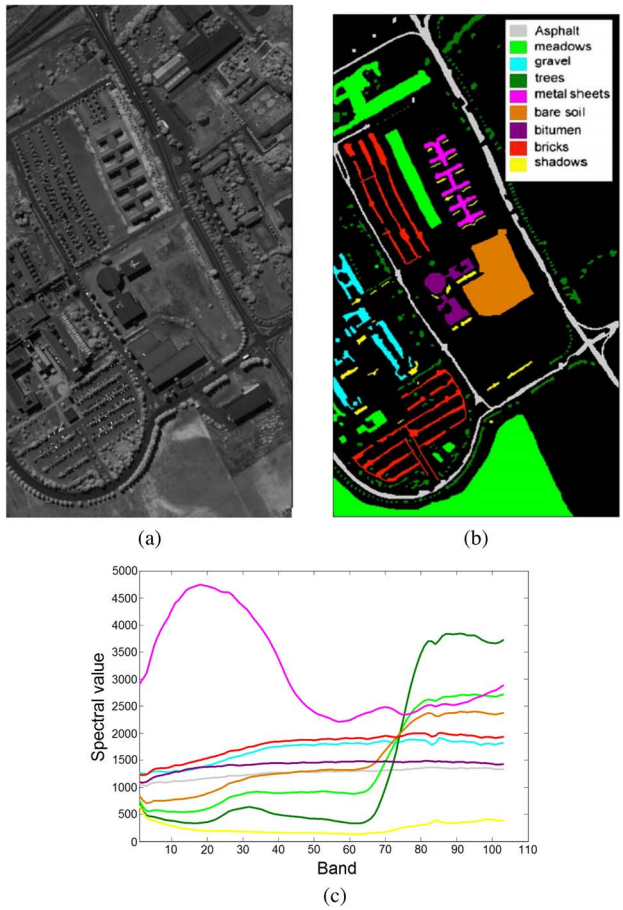


Fig. 2. ROSIS image scene: University of Pavia. (a) Band 80. (b) Ground truth map for nine classes. (c) Spectral signatures of ten classes.

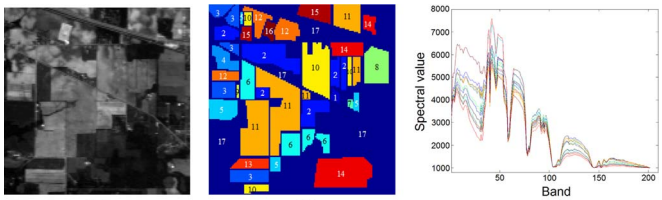
In Fig. 2(b), there are nine classes in this image scene, consisting of several urban targets such as vegetation, soil, and roads. The corresponding spectral signatures are shown in Fig. 2(c). The sample sizes of all the nine classes are 6631 (asphalt), 18 649 (meadows), 2099 (gravel), 3064 (trees), 1345 (painted metal sheets), 5029 (bare soil), 1330 (bitumen), 3682 (self-blocking bricks), and 947 (shadows), respectively.

The second data set[2] used in our experiments is the real AVIRIS image data, Purdue's Indiana Indian Pine test site, which has been extensively studied in the literature and provides a good candidate for those who are interested in algorithm design and analysis. It has 20-m spatial resolution and 10-nm spectral resolution in the range of 0.4–2.5 μm with size 145 × 145 pixel vectors taken from an area of mixed agriculture and forestry in Northwestern Indiana, USA. It was recorded in June 1992 with 220 bands, among which bands 104–108 and 150–162 were removed, whereas the remaining 202 bands were retained. Fig. 3(a) and (b) shows the image of band 20 and the ground truth map, respectively.

There are 17 classes in this image scene, including the background labeled by class 17, which has a wide variety of

[1] http://www.ehu.es/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes

[2] https://engineering.purdue.edu/biehl/MultiSpec/documentation.html



(a) (b) (c)

Fig. 3. AVIRIS image scene: Purdue Indiana Pine test site. (a) Band 20. (b) Ground truth map. (c) Spectral signatures of 17 classes.

TABLE I
LABELS OF 17 CLASSES

class 1 alfalfa (54)	class 10 soybeans-notill (986)
class 2 corn-notill (1434)	class 11 soybeans-min (2468)
class 3 corn-min (834)	class 12 soybeans-clean (614)
class 4 corn (234)	class 13 wheat (212)
class 5 grass/pasture (497)	class 14 woods (1294)
class 6 grass/trees (747)	class 15 bldg-grass-green-drives (380)
class 7 grass/pasture-mowed (26)	class 16 stone-steel towers (95)
class 8 hay-windrowed (489)	class 17 Background (10659)
class 9 oats (20)	

targets such as highways, railroad, houses/buildings, and vegetation that may not be of interest in agriculture applications. The spectral signatures of all classes of interest are shown in Fig. 3(c). The total number of data samples in the scene is $145 \times 145 = 21\,025$. Table I lists the labels of these 17 classes, where the numeral in the brackets denotes the number of data samples in that class. Compared with Pavia data, the Purdue data set is more heavily mixed because of its lower spatial resolution, less precise class annotation, and the higher similarity of the spectral signatures. Thus, processing Purdue data is more challenging.

B. Acquisition of Training and Testing Samples

Since the training samples should reflect the class proportion of the images for SU, the training data set D is obtained by randomly selecting a certain percent of data samples from all target classes. Thus, the resulting training data are unbalanced, but the class proportion is kept. For unmixing, the endmember $\mathbf{s}_i$ is calculated by taking the mean of the training pixels belonging to class i, $1 \le i \le p$.

Once the training samples are selected from the image, we use the rest of image pixels as the testing samples for performance evaluation. In the experiments, the acquisition of training and testing samples is repeated ten times, and the averaged results are reported.

C. Experimental Setup for MKL-SMA

As mentioned in Section V, three ways for kernel construction can be used. In the construction of the DHV kernels, we empirically set $M = 5$ by employing the RBF kernels with different values of the hyperparameter, i.e., $[\sigma/4,\ \sigma/2,\ \sigma,\ 2\sigma,\ 4\sigma]$. As suggested in [21], σ in (27) is set as the average pairwise distance among training data. That is

$$\sigma = E\left(\|\mathbf{x}_i - \mathbf{x}_j\|\right) \qquad \forall i, j = 1, 2, \ldots, N. \tag{30}$$

In the construction of the SS kernels, we use one RBF kernel to represent the spectral features and four RBF kernels to represent the spatial features of different spatial mixing degrees, i.e., $M^{\mu} = 1$ and $M^{s} = 4$ in (28). For the Pavia data set, we set the neighborhood area as a $w \times w$ window with $w \in \{5, 10, 15, 20\}$. The spatial features are the mean vectors of those pixels inside the windows. For the Purdue data set, we set $w \in \{3, 5, 8, 10\}$ since its image size is relatively smaller. The suggested RBF parameter of each basis kernel is determined by (30).

In the construction of the PSR kernels, an RBF kernel is established for each band. Similarly, the RBF parameter is also determined by (30).

D. Performance Metrics and Evaluation

Unlike hard classification that makes decisions by assigning each data sample to one specific class, the SU gives real-valued abundance fractions representing the mixing degrees of the spectral signatures. As a result, the abundance fractions cannot be simply evaluated by the hard-decision-based method.

Because the *unmixing ground truth* is unavailable in most public data sets, research studies in SU cannot but use the data sets with only hard classification labels for performance evaluation. We adopt area under the receiver operating characteristic (ROC) curve (AUC) as the main measure to evaluate the classification performance for unmixed abundance vector set $\hat{\boldsymbol{\alpha}}(\mathbf{r}_i) = (\hat{\alpha_1}(\mathbf{r}_i), \hat{\alpha_2}(\mathbf{r}_i), \ldots, \hat{\alpha_p}(\mathbf{r}_i))^{\top}$ of testing sample $\mathbf{r}_i$. First, we normalize $\hat{\boldsymbol{\alpha}}(\mathbf{r}_i)$. We then utilize abundance percentage mixed-to-pure pixel converter (MPCV) [2] with a threshold $\tau \in [0, 1]$ to convert the normalized abundance to pure class. If the estimated abundance fraction of a signature, for example, $\mathbf{s}_j$, exceeds threshold τ, sample $\mathbf{r}_i$ is then assigned to class j. The preceding procedure allows us to perform ROC analysis on the testing pixels. Since there are multiple signals of interest specified by $\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_p$, to extend a single-signal detection-based ROC analysis to a multiple-signal detection model, we first compute the detection rate $R_D(\mathbf{s}_j)$ and the false alarm rate $R_F(\mathbf{s}_j)$ for the jth signal source $\mathbf{s}_j, \forall j = 1, 2, \ldots, p$, where

$$R_D(\mathbf{s}_j) = \frac{N_D(\mathbf{s}_j)}{N(\mathbf{s}_j)} \quad R_F(\mathbf{s}_j) = \frac{N_F(\mathbf{s}_j)}{N_t - N(\mathbf{s}_j)}. \tag{31}$$

In (31), $N_D(\mathbf{s}_j)$ is the number of the pixels that belong to class j and are correctly detected. $N_F(\mathbf{s}_j)$ is the number of the pixels that are not of class j but detected as class j. $N(\mathbf{s}_j)$ is the number of the pixels of class j, and N_t is the total number of test pixels. For the multiple signal sources, we calculate the mean detection rate $\bar{R}_D$ and the mean false alarm rate $\bar{R}_F$ by

$$\bar{R}_D = \sum_{j=1}^{p} h(\mathbf{s}_j) R_D(\mathbf{s}_j) \quad \bar{R}_F = \sum_{j=1}^{p} h(\mathbf{s}_j) R_F(\mathbf{s}_j) \tag{32}$$

where $h(\mathbf{s}_j)$, i.e., the weighting factor of class j, is defined as

$$h(\mathbf{s}_j) = \frac{N(\mathbf{s}_j)}{\sum_{j=1}^{p} N(\mathbf{s}_j)}. \tag{33}$$

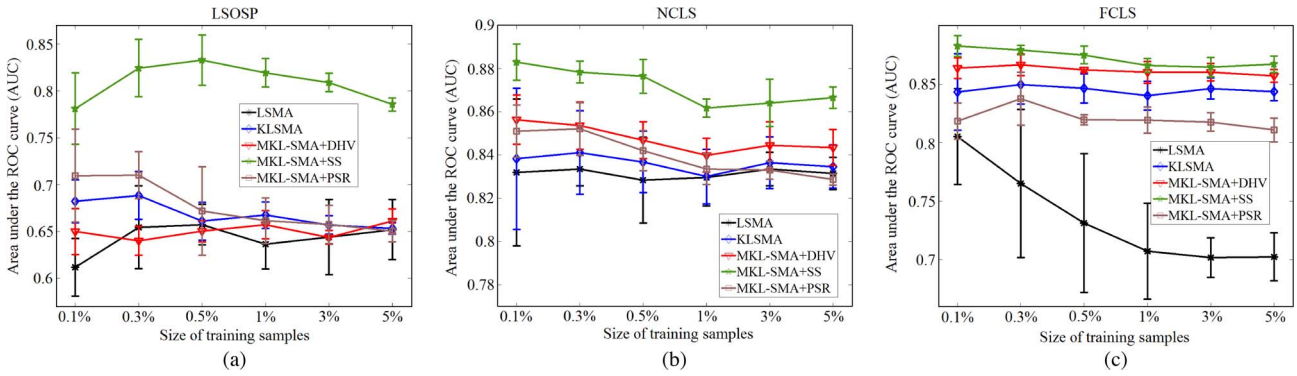


Fig. 4. AUC± standard deviation by applying LSMA, KLSMA, and MKL-SMA to Pavia data with abundance estimator. (a) LSOSP. (b) NCLS. (c) FCLS.

Finally, we can calculate the AUC of $\bar{R}_D$ versus $\bar{R}_F$ for quantitative performance analysis by varying threshold τ.

AUC provides a quantitative analysis between the detection power and the false alarm probability and is commonly used in performance assessment for soft classification or unmixing [2, 24, 30]. We also consider the SU as a hard classification problem in which three common measures, namely, average accuracy (AAC), overall accuracy (OAC), and Kappa coefficient, are used to evaluate the performance. As mentioned previously, SU does not classify a data sample to a certain class, but estimates the abundance fraction of each signature contained in the data sample. Here, we make assumption that an ideal abundance estimation holds the following property: the signature class of the maximum value in an abundance vector will be consistent with the ground truth class. Hence, we assign each estimated abundance vector to a dominant class by MPCV via the winner-take-all rule. That is, for the unmixed abundance vector $\hat{\boldsymbol{\alpha}}(\mathbf{r}_i) = [\hat{\alpha}_1(\mathbf{r}_i)\ \hat{\alpha}_2(\mathbf{r}_i)\ \ldots\ \hat{\alpha}_p(\mathbf{r}_i)]^\top$ of testing sample $\mathbf{r}_i$, the j^*th class is predicted if

$$j^* = \arg\max_{1\le j\le p} \hat{\alpha}_j(\mathbf{r}_i). \tag{34}$$

In the experiments, the primary comparison of interest is to check whether the proposed MKL-SMA can effectively leverage the abundant information carried by multiple kernels and outperforms LSMA and KLSMA. The SVM is also selected as the reference if we convert SU results to hard classification results by MPCV.

VII. Experimental Results

We conduct a comprehensive and comparative study of the performance evaluation and analysis for LSMA, KLSMA, and the proposed MKL-SMA.

A. ROSIS (Pavia) Data

To evaluate the unmixing performance of our approach and the adopted baselines with different numbers of training data, we randomly selected 0.1%, 0.3%, 0.5%, 1%, 3%, and 5% of data of each class as the training samples in the Pavia scene. The unmixing results in AUC by LSMA, KLSMA, and MKL-SMA with three types of basis kernels, i.e., DHV, SS, and PSR, are shown in Fig. 4, where the three subfigures correspond to the results of using LSOSP, NCLS, and FCLS, as abundance estimators, respectively.

As shown in Fig. 4(a), KLSMA is superior to LSMA in all cases. It does imply that kernel methods can effectively solve the linear nonseparability problem in Pavia data. Furthermore, using multiple kernels provides the opportunity of further improvement. MKL-SMA with the PSR kernels (denoted by MKL-SMA+PSR) slightly outperforms KLSMA in lower training sizes. Such improvement may owe to the flexibility of assigning different weights to the PSR kernels, each of which corresponds to a single spectral band. With the aid of spatial information, MKL-SMA using the SS kernels, i.e., MKL-SMA+SS, significantly outperforms KLSMA no matter how many training samples are used. MKL-SMA+SS achieves significant performance gains around 0.1–0.17 over KLSMA in AUC. It points out that MKL-SMA+SS can effectively make use of the additional spatial information to assist the classifier in making more accurate decisions so that the weak unmixing capability of LSOSP could be improved. In addition, we observed that MKL-SMA with DHV kernels, i.e., MKL-SMA+DHV, does not provide any improvement over LSMA and KLSMA. It may result from information redundance among the DHV basis kernels.

Fig. 4(b) shows the performance of various unmixing approaches when NCLS is the abundance estimator. Compared with LSOSP, NCLS employs the additional nonnegative constraint, which is helpful for abundance estimation. Hence, most approaches coupled with NCLS get better outcomes. As shown in the figure, MKL-SMA with any type of basis kernels outperforms KLSMA. MKL-SMA+DHV slightly surpasses KLSMA in this case. As aforementioned, DHV basis kernels tend to carry redundant information and do not complement each other. However, using DHV kernels in MKL sometimes works, particularly when the optimal hyperparameters are difficult to be determined. MKL-SMA+PSR also surpasses KLSMA a little bit since it can better interpret the data in terms of the independence of the bands. Again, MKL-SMA+SS provides significant improvement over KLSMA. The AUC values of KLSMA are around 0.83, whereas those of MKL-SMA+SS are 0.86–0.88. It implies that the NCLS estimator can well utilize the additional spatial information introduced by the SS kernels.

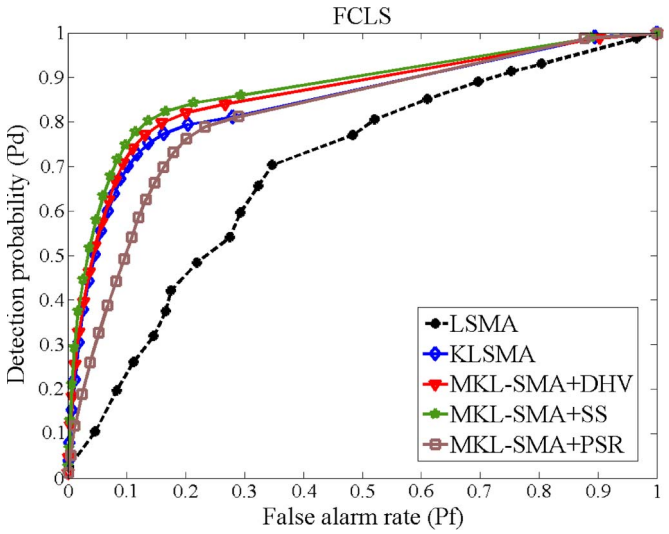


Fig. 5. Representative ROC curves of Pd versus Pf by applying LSMA, KLSMA, and MKL-SMA to Pavia data with abundance estimator FCLS. The experiment is conducted with 0.5% randomly selected training samples.

Fig. 4(c) shows the performance of SU methods when FCLS is adopted as the abundance estimator. Similar to the observations found in Fig. 4(b), we obtained the following ranking in AUC: 1) MKL-SMA; 2) KLSMA; and 3) LSMA, except for MKL-SMA+PSR. FCLS performs poorly because the imposed sum-to-one constraint suppresses its unmixing capability on Pavia data. Fortunately, this issue can be solved by carrying out SU in the feature space expanded by kernels. Interestingly, MKL-SMA+PSR could not surpass KLSMA. It probably results from that the two abundance constraints imposed by FCLS suppress the capability of the PSR kernels in the feature space. In conclusion, using MKL-SMA with appropriate abundance estimators and basis kernels can provide remarkable improvements.

Fig. 5 shows the ROC analysis of all the methods with FCLS estimator when 0.5% randomly selected training samples are used. It can be found that the ROC performance (Pd versus Pf) can be improved by using kernels, as shown in Fig. 5. It again indicates that the PSR kernels cannot afford obvious benefits for unmixing Pavia data. The possible reason is that the importance of the spectral bands in Pavia data is almost equal so that there is no room for improving the performance by reweighting the bands. However, we will show that MKL-SMA+PSR achieves unmixing results of high quality on Purdue data and further discuss the reasons later. Thus, the effectiveness of the PSR kernels is data dependent.

Note that the AUC performance shown in Fig. 4 does not increase as the training size grows. The main reason is that LSMA (KLSMA and MKL-SMA) is an approximate model for SU. Approaches based on LSMA compile the endmembers by computing only the first-order statistics, i.e., mean, from training samples. The strength is that only few training data are sufficient to achieve satisfactory results. However, they ignore the high-order statistics, which can be reliably extracted when more training data are available.

To gain insight into this phenomenon, we computed the AUC values of each class by varying the sizes of training data. An interesting observation was found: As the training size goes larger, the AUC values of some *large* classes, those having more data, slightly decrease, whereas the AUC values of *small* classes significantly improve. Because the overall AUC is the weighted sum of the AUC values of all classes with the weights proportional to their numbers of data, the unmixing performance in terms of overall AUC does not benefit from large training set.

In addition, LSMA and its variants are not biased models in the sense that one endmember is computed for each class no matter how many training data it has. In the cases of fewer training data, the endmembers of the small classes are more likely to be badly estimated. However, a high AUC can still be obtained as long as good performance are achieved in larger classes. In the cases of more training data, the endmembers of all classes are accurately estimated. The accurately estimated endmember of a small class may degrade the performance of a larger class if their endmembers are close to each other. These circumstances usually happen. As large classes dominate the AUC measure, it therefore causes the degradation of the overall performance.

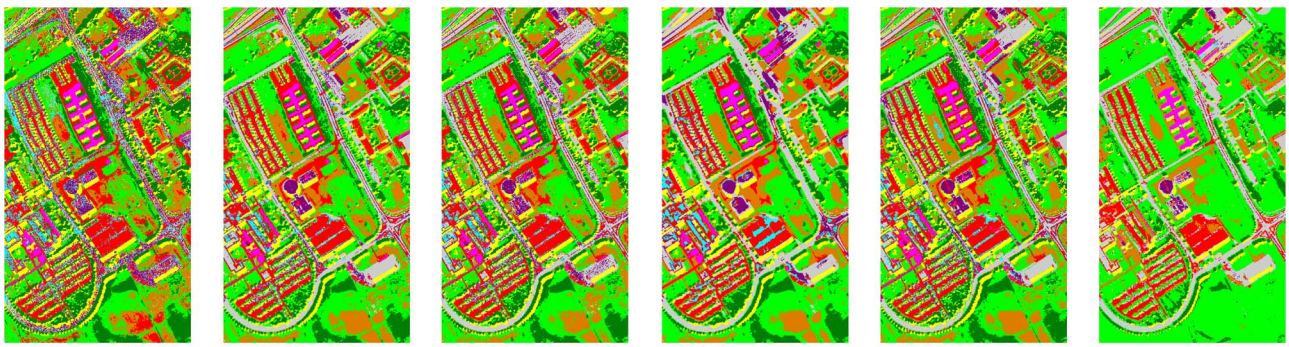
The classification performance converted from unmixing results via MPCV is also concerned. Table II tabulates the AAC, OAC, and Kappa coefficient (in brackets) values of LSMA, KLSMA, and the proposed MKL-SMA with three types of basis kernels, respectively. Several findings in the table can be concluded as follows.

1) KLSMA outperforms LSMA in most cases, and MKL-SMA+SS further increases the classification accuracy over KLSMA. It points out that using multiple kernels also helps make more accurate decisions for classification.
2) Compared with the DHV and PSR kernels, the SS kernels accomplish better classification outcomes in all the measures because SS kernels carry spatial information that can alleviate the problem of misclassification in the noisy regions. Similar to the conclusions drawn in the literature of MKL, using the DHV kernels is not very effective for classification, because MKL with the DHV kernels is reduced to *model selection* instead of *feature fusion*.
3) The abundance estimator plays an important role. NCLS and FCLS work more effectively and stably. LSOSP performs poorly due to its unconstrained nature. Fortunately, this dilemma is mitigated when multiple kernels are taken into account, particularly the SS and PSR kernels.
4) In most cases, SVM achieves superior performance over LSMA/KLSMA/MKL-SMA in OAC/Kappa coefficient due to the fact that SVM is designed for hard classification, whereas MKL-SMA, which is developed for unmixing, achieves the best results in AAC. Nevertheless, the proposed MKL-SMA with properly chosen kernels is still comparable with SVM, particularly in the cases where lower training sizes are adopted.

To visualize the classification results, Fig. 6 displays the hard-decision classification maps of LSMA, KLSMA, MKL-SMA+DHV, MKL-SMA+SS, and MKL-SMA+PSR, respectively. Those maps are created by assigning each pixel a specific color according to which class it belongs to. The classification

TABLE II
PERFORMANCE, IN THE FORM OF [AAC/OAC (KAPPA COEFFICIENT)], BY APPLYING LSMA, KLSMA, AND THE PROPOSED MKL-SMA TO PAVIA DATA

Size of training samples	Method	LSMA	KLSMA	MKL-SMA+DHV	MKL-SMA+SS	MKL-SMA+PSR	SVM
0.1%	LSOSP	42.1 / 52.3 (38.8) ±3 / 5.8 (6.7)	51.2 / 54.6 (42.7) ±5 / 4.9 (5.1)	47.8 / 56.2 (43.8) ±3.9 / 5.1 (5.9)	58.8 / 64.7 (54) ±3.1 / 1.5 (1.5)	49.4 / 54.5 (42.5) ±5.1 / 5.4 (5.4)	36.6 / 63.9 / (48) ±7.5 / 3.4 (5.5)
	NCLS	63 / 55.7 (46.5) ±4.5 / 6.7 (6.3)	67.8 / 63 (53.7) ±4.5 / 8.1 (7.9)	68.8 / 67 (57.5) ±4.3 / 3.7 (3.5)	69.8 / 69.2 (60.1) ±2.4 / 2 (2.1)	68.4 / 65.6 (56.1) ±2.4 / 4 (3.6)	
	FCLS	65.5 / 60 (50) ±3.2 / 4.2 (4.5)	67.5 / 63.6 (54.3) ±4.3 / 8.3 (8.1)	68.4 / 67.1 (57.7) ±3.3 / 4.1 (4.2)	69.7 / 69 (60) ±2.1 / 2.5 (2.4)	59.8 / 57.3 (46.3) ±6.6 / 2.7 (2.9)	
0.3%	LSOSP	38.7 /44.3 (31.5) ±2.8 / 2.1 (2.3)	51.3 / 49.8 (38.1) ±1.3 / 3 (2.7)	47.3 / 50.9 (39.1) ±2.7 / 5.7 (6)	60.8 / 58.4 (48.3) ±3.6 / 3.3 (3.3)	53.8 / 55 (43.7) ±4.1 / 2.8 (3.1)	56.9 / 72.8 (61.5) ±2.5 / 1.6 (2.6)
	NCLS	64.4 /55 (44.9) ±1.9 / 2.4 (2.2)	69.6 / 64.2 (54.7) ±1.3 / 3.3 (3.5)	70.5 / 65 (55.6) ±1.4 / 1.9 (1.9)	73.1 / 68 (59.3) ±1.4 / 1.6 (1.6)	68 / 63.6 (54) ±1.8 / 1.3 (1.1)	
	FCLS	66.2 / 57.9 (47.9) ±1.8 / 2.7 (2.7)	69.1 / 64.7 (55.1) ±1 / 3.1 (3.3)	70.1 / 66.6 (54.6) ±1.8 / 2.9 (3.3)	72.8 / 68.3 (59.5) ±1.4 / 1.5 (1.7)	65.5 / 60.5 (50.3) ±1.9 / 5.5 (5.7)	
0.5%	LSOSP	38.3 / 44.2 (31.6) ±1.6 / 1.3 (1.6)	53.5 / 53.3 (41.8) ±2 / 2.2 (2.4)	49 / 52.6 (37.4) ±1.9 / 2.2 (2.4)	62.4 / 62.3 (52.7) ±1.5 / 1.7 (2)	55.3 / 52.6 (41.7) ±5.1 / 4.4 (5)	60.9 / 75 (65.1) ±2.1 / 1.1 (1.4)
	NCLS	65.3 / 52.9 (43.2) ±2.1 / 2 (2)	70.1 / 63.2 (53.9) ±1.7 / 1.9 (1.8)	70.6 / 63.7 (54.2) ±1.6 / 1.3 (1.2)	73.8 / 66.8 (58.2) ±1.7 / 1.5 (1.7)	68.9 / 62.3 (52.9) ±1.5 / 2 (1.9)	
	FCLS	66.6 / 56.5 (46.7) ±2.1 / 2.2 (2.2)	69.7 / 63.6 (54.4) ±1.8 / 1.7 (1.7)	69.4 / 63.9 (56.5) ±1.7 / 1.6 (1.6)	74.2 / 67.9 (59.3) ±1.5 / 1.4 (1.6)	63.8 / 56.7 (45.2) ±5.1 / 3.1 (4.6)	
1%	LSOSP	35.4 / 39.4 (24.1) ±3.7 / 4.4 (4.1)	53.7 / 52.5 (41) ±2.4 / 2.4 (2.4)	47.2 / 48.7 (31.3) ±3.8 / 4 (3.8)	63.6 / 59.2 (49.5) ±2.5 / 2 (2.4)	58.9 / 56.7 (46.1) ±2.4 / 0.9 (0.9)	66 / 76.5 (67.5) ±3.3 / 1.3 (1.7)
	NCLS	67.1 / 54.1 (44.5) ±0.8 / 2.2 (2)	71.8 / 62.6 (53.3) ±1.6 / 1.9 (2)	72 / 63.5 (54.6) ±1.7 / 1 (1.1)	75.3 / 66 (57.3) ±1.4 / 0.9 (1.1)	70 / 62.2 (52.7) ±2.1 / 1.7 (1.7)	
	FCLS	68.9 / 57.2 (47.7) ±1 / 2.6 (2.6)	71.9 / 63.4 (54.2) ±1.5 / 1.9 (2)	72.5 / 65.5 (56) ±1.3 / 1.6 (1.7)	75.2 / 66.4 (57.8) ±1.7 / 1.3 (1.4)	62.8 / 55 (43.8) ±4.5 / 4.5 (5.6)	
3%	LSOSP	30.9 / 30.1 (18.8) ±3.1 / 2.4 (2.2)	53.6 / 51.2 (39.8) ±0.9 / 0.8 (0.9)	41.2 / 42.4 (31.8) ±2.3 / 1.2 (1.3)	61.9 / 58.3 (48.2) ±0.3 / 1.3 (1.3)	58.1 / 54 (43.5) ±0.5 / 0.7 (0.7)	72.9 / 79.8 (72.2) ±1.4 / 1.1 (1.4)
	NCLS	67.9 / 54 (44.5) ±1.4 / 1.3 (1.4)	71.5 / 63.7 (54.7) ±0.6 / 1.7 (1.8)	72.1 / 63.8 (54.1) ±0.2 / 0.8 (0.8)	74.6 / 66.6 (58) ±0.7 / 1.7 (1.9)	69.4 / 61.7 (52.2) ±0.8 / 0.6 (0.5)	
	FCLS	67.1 / 55.4 (45.5) ±1.6 / 1.8 (2)	71.2 / 64 (54.7) ±0.7 / 1.1 (1.1)	71.5 / 65 (56) ±1.2 / 1.2 (1.3)	74.2 / 66.5 (57.7) ±0.5 / 1.2 (1.3)	64.3 / 56 (44.9) ±4.2 / 4.1 (5.1)	
5%	LSOSP	29.1 / 27.5 (16.3) ±1.6 / 2 (1.8)	53.8 / 52 (40.5) ±1.1 / 1.4 (1.3)	41.5 / 42.8 (31.8) ±1.7 / 2.9 (2.7)	61.2 / 57.4 (47.4) ±0.8 / 0.3 (0.4)	58.7 / 54 (43.7) ±0.6 / 0.7 (0.6)	75.8 / 82 (75.4) ±0.8 / 0.6 (0.9)
	NCLS	68.5 / 54.4 (44.9) ±0.5 / 1.1 (1)	71.2 / 63.4 (54.1) ±0.8 / 1.3 (1.3)	72 / 63.3 (54.1) ±0.2 / 0.6 (0.6)	74.7 / 66.7 (58.2) ±0.4 / 1.4 (1.5)	69.6 / 61.7 (52.2) ±0.9 / 1.2 (1.2)	
	FCLS	67.7 / 55.9 (46.2) ±1 / 1.3 (1.5)	70.7 / 63.6 (54.2) ±0.9 / 0.7 (0.8)	71./ 64.7 (55.6) ±0.7 / 0.7 (0.8)	74.2 / 66.5 (57.9) ±0.2 / 0.8 (0.8)	60.7 / 52 (40.1) ±3.1 / 2.9 (3.7)	



(a) (b) (c) (d) (e) (f)

Fig. 6. Classification maps converted by applying MPCV to the abundance fractions estimated by (a) LSMA, (b) KLSMA, (c) MKL-SMA+DHV, (d) MKL-SMA+SS, and (e) MKL-SMA+PSR when FCLS is used as the abundance estimator. (f) Map by SVM for additional reference. The results are generated from a single run of using 3% randomly selected training samples. The values shown in brackets are the corresponding AAC. (a) LSMA (66.3%). (b) KLSMA (71.4%). (c) MKL-SMA+DHV (72%). (d) MKL-SMA+SS (74.6%). (e) MKL-SMA+PSR (66.1%). (f) SVM (73.5%).

map of LSMA shown in Fig. 6(a) is noisy, whereas all the other kernel-based methods Fig. 6(b)–(e) produce cleaner maps. Among all of them, MKL-SMA+SS seems to give the best results. This can be verified by comparing with the ground truth shown in Fig. 2(b). For instance, the bitumen signature located in the center of the image was better classified by MKL-SMA+SS, whereas the other methods produced the classification maps that contain more noises in the regions of the bitumen.

B. AVIRIS (Purdue) Data

Since the image size of Purdue data is relatively smaller, i.e., 145×145, the fractions of training samples are set as 1%, 3%, 5%, 8%, 10%, and 20%, respectively, in the experiments, so that sufficient training samples for each target class can be acquired.

Fig. 7 shows the AUC of the unmixing results produced by LSMA, KLSMA, and three types of MKL-SMA, respectively, where LSOSP, NCLS, and FCLS are used as abundance

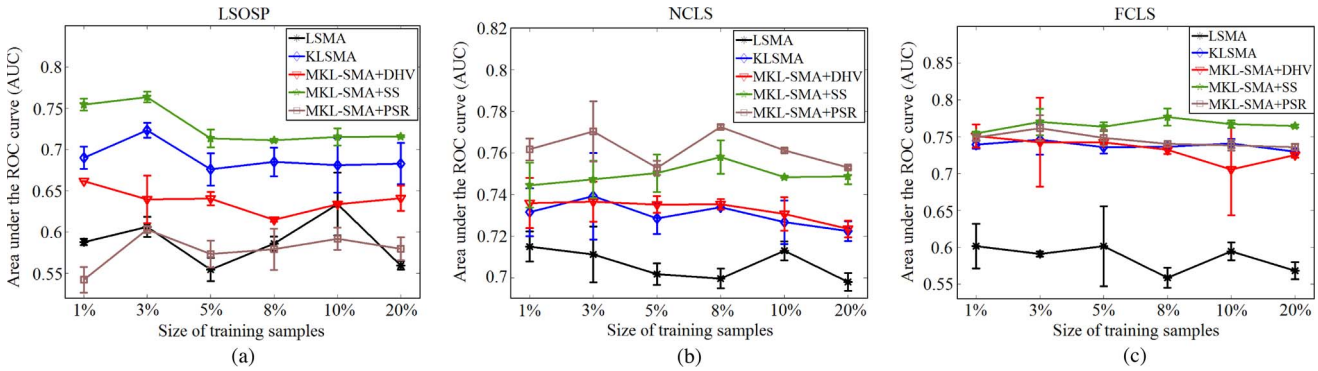


Fig. 7. AUC with standard deviations by applying LSMA, KLSMA, and MKL-SMA to Purdue data with abundance estimator. (a) LSOSP. (b) NCLS. (c) FCLS.

TABLE III
PERFORMANCE, IN THE FORM OF [AAC/OAC (KAPPA COEFFICIENT)], BY APPLYING LSMA, KLSMA, AND THE PROPOSED MKL-SMA TO PURDUE DATA

Size of training samples	Method	**LSMA**	**KLSMA**	**MKL-SMA+DHV**	**MKL-SMA+SS**	**MKL-SMA+PSR**	SVM
1%	FCLS	45.8 / 38.1 (31.8) ±2.5 / 2.6 (3)	48.1 / 39.5 (33.2) ±2.4 / 0.3 (0.6)	49.5 / 40.6 (34.7) ±1.29 / 1.4 (1.6)	54.4 / 42 (36.1) ±1.8 / 0.8 (1)	49.4 / 41.1 (34.6) ±0.3 / 1.1 (1.3)	31.4 / 50.5 (40.5) ±1.1 / 1.6 (3.1)
3%		46.9 / 42 (35.2) ±2.4 / 3.3 (3.2)	48.7 / 43.2 (36.7) ±0.2 / 1.5 (1.5)	50.6 / 44.2 (37.7) ±1.2 / 2.1 (2)	52.3 / 46.7 (40.4) ±2.2 / 1.2 (1.3)	46.6 / 45.3 (39) ±1.37 / 2.3 (2.4)	42.4 / 56.5 (48.5) ±0.8 / 0.4 (0.6)
5%		49.1 / 40 (33.2) ±0.2 / 1.1 (0.8)	50.9 / 41.2 (34.7) ±0.1 / 1 (0.7)	53.1 / 42.2 (35.7) ±0.2 / 0.6 (0.3)	54.3 / 45.2 (38.8) ±0.2 / 1.1 (1.1)	49 / 43.2 (36.8) ±2.2 / 1.4 (1.5)	43.9 / 58.4 (50.4) ±2 / 0.8 (1.1)
8%		49 / 40.7 (34) ±0.5 / 1.4 (1.3)	52.1 / 41.6 (35.1) ±0.5 / 0.6 (0.5)	51.4 / 41.4 (35) ±0.4 / 0.9 (0.9)	56.5 / 46 (39.8) ±2.2 / 0.2 (0.1)	50.2 / 41.1 (35.2) ±0.8 / 0.6 (0.7)	49.9 / 65.2 (59.3) ±1.4 / 0.7 (0.9)
10%		47.8 / 37.7 (31.6) ±1.2 / 2.1 (1.8)	50.3 / 39.8 (33.7) ±0.4 / 2.4 (2.1)	49.7 / 40.3 (34.3) ±1.7 / 1.5 (1.2)	58.6 / 45.2 (39.3) ±2.3 / 0.6 (0.8)	46.8 / 40.8 (34.4) ±0.8 / 2 (1.9)	50.5 / 63.8 (57.8) ±0.6 / 1.2 (1.5)
20%		48.6 / 38.6 (32.3) ±1.3 / 0.5 (0.4)	51.6 / 39 (32.7) ±1 / 0.8 (0.7)	51.6 / 39.5 (33.4) ±1.2 / 0.5 (0.3)	58.7 / 46.1 (39.9) ±0.9 / 0.3 (0.3)	50 / 40.2 (33.8) ±1.7 / 0.3 (0.3)	58.2 / 67.1 (61.9) ±2 / 0.4 (0.5)

estimators. Table III further tabulates the AAC, the OAC, and the Kappa coefficient for the unmixing results when FCLS is used. From Fig. 7 and Table III, several findings are summarized in the following.

1) All the kernel methods, including KLSMA and MKL-SMA, consistently outperform LSMA in most cases of Purdue experiments.
2) MKL-SMA+DHV could not provide improvement over KLSMA in AUC, particularly when LSOSP is adopted. Thus, the performance of MKL-SMA+DHV that captures highly overlapping spectral information is still restricted.
3) MKL-SMA+SS with either LSOSP or FCLS achieves the best performance in AUC. As mentioned, using the SS kernels in the MKL-SMA can reduce the intraclass spectral variations for those images whose ground classes are spatially dependent and reduce the unmixing errors caused by the interference of the noise. The outcomes of applying MKL-SMA+SS to Purdue data again validate the effectiveness of using the SS kernels.
4) MKL-SMA+PSR performs diversely with different abundance estimators. For instance, it reaches the highest AUC over all compared methods in NCLS case, but performs worse than LSMA in LSOSP case in Fig. 7. This phenomenon did not appear in the Pavia data set. It implies that the PSR kernels seem to be very sensitive to the properties of data and the type of abundance estimators.

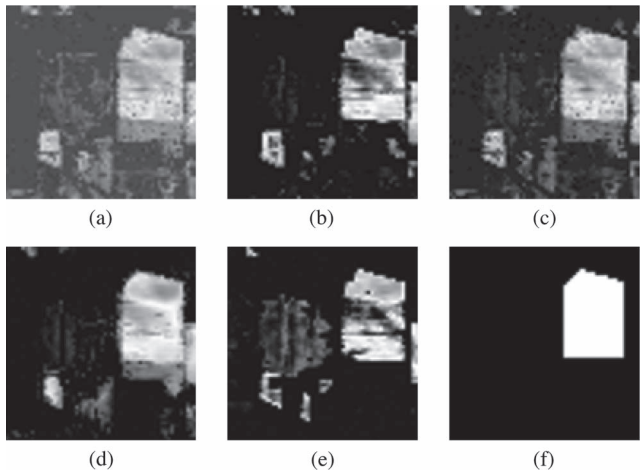


Fig. 8. Abundance maps of class 8 in the Purdue scene estimated by (a) LSMA, (b) KLSMA, (c) MKL-SMA+DHV, (d) MKL-SMA+SS, and (e) MKL-SMA+PSR, respectively. (f) Ground truth. The experiment was conducted by implementing unmixing with 10% randomly selected training samples.

To analyze the unmixing results in a visual manner, Figs. 8 and 9 show the cropped abundance maps corresponding to target classes 8 and 6 in the Purdue scene, respectively. Those maps in each figure were respectively compiled by using LSMA, KLSMA, MKL-SMA+DHV, MKL-SMA+SS, and MKL-SMA+PSR, with abundance estimator FCLS. In Fig. 8, it can be seen that LSMA produced low-contrast results in Fig. 8(a), since more false alarms were induced. KLSMA

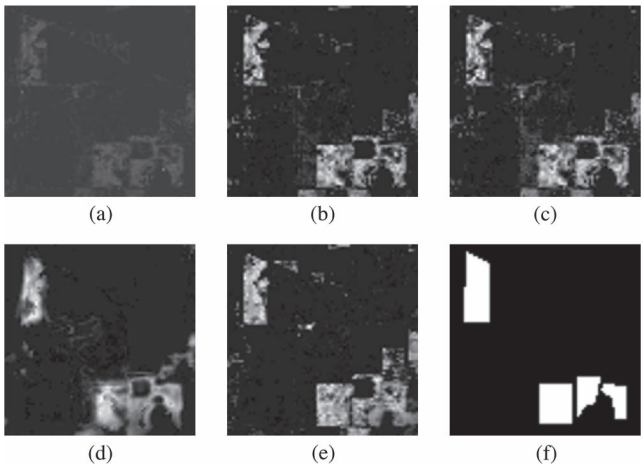


Fig. 9. Abundance maps of class 6 in the Purdue scene estimated by (a) LSMA, (b) KLSMA, (c) MKL-SMA+DHV, (d) MKL-SMA+SS, and (e) MKL-SMA+PSR, respectively. (f) Ground truth. The experiment was conducted by implementing unmixing with 10% randomly selected training samples.

TABLE IV
OPTIMIZED KERNEL WEIGHTS BY MKL-SMA+SS ON PAVIA DATA, WHEN FCLS IS USED AS THE ABUNDANCE ESTIMATOR

Size of training samples	Spectral kernel	Spatial kernel ($w = 5$)	Spatial kernel ($w = 10$)	Spatial kernel ($w = 15$)	Spatial kernel ($w = 20$)
0.1%	**0.4910**	0.2566	0.0952	0.0767	0.0805
0.3%	**0.3399**	0.3360	0.1299	0.0833	0.0579
0.5%	**0.4633**	0.3298	0.1022	0.0610	0.0437
1%	**0.4344**	0.3467	0.1117	0.0635	0.0437
3%	**0.4861**	0.3234	0.0920	0.0582	0.0404
5%	**0.4719**	0.3375	0.0935	0.0561	0.0411

in Fig. 8(b) improves a lot in the background suppression, but it fails to detect the central regions of class 8. MKL-SMA+DHV generated moderate results in Fig. 8(c), since it considered different scales of RBF features at the same time. In Fig. 8(e), MKL-SMA+PSR produced an overenhanced map that is dissimilar to the maps in Fig. 8(a)–(c). Much satisfactory outcomes were accomplished by MKL-SMA+SS in Fig. 8(d). The SS kernels can alleviate the problem caused by large intraclass variations by additional spatial information, and thus, a smoother and more accurate abundance map was yielded. Analogous observations can be found for Fig. 9. Both MKL-SMA+SS and MKL-SMA+PSR perform well for class 6 and preserve the area shapes of this class more accurately.

C. Observations and Discussion

The use of multiple kernels not only increases the classification performance but also reveals the inherent characteristics of the data. In the following, several intriguing observations found in the experiments of MKL-SMA are discussed.

First, it has been mentioned in Section V that the SS kernels provide both spectral and spatial information, whereas MKL-SMA balances their weights through an optimization process. To look into the relative importance of the two kinds of information, Tables IV and V record the optimized kernel weights β by MKL-SMA+SS coupled with FCLS on the Pavia and Purdue scenes, respectively. It can be seen that MKL-SMA assigned almost equal weights on the spectral and spatial kernels for Pavia data, whereas it puts higher weights on the spatial kernels ($w = 5$ and 8) for Purdue data. It suggests that unmixing Pavia data equally relies on both spectral and spatial information, whereas unmixing Purdue data relies more on spatial (contextual) information. Such an observation accords with our expectation since the ground spatial resolution of Purdue data is relatively lower and the spectral similarity of the target signatures is higher [see Fig. 3(c)], and pixels located near the boundaries of two nearby classes are heavily mixed. In addition, Purdue data are much noisier than Pavia data. The noisy pixels are frequently misclassified even if are they located in the pure regions. With the auxiliary spatial information, we can effectively alleviate the misclassifications by unmixing the data with low spatial resolution or with noisy appearance.

TABLE V
OPTIMIZED KERNEL WEIGHTS BY MKL-SMA+SS ON PURDUE DATA, WHEN FCLS IS USED AS THE ABUNDANCE ESTIMATOR

Size of training samples	Spectral kernel	Spatial kernel ($w = 3$)	Spatial kernel ($w = 5$)	Spatial kernel ($w = 8$)	Spatial kernel ($w = 10$)
1%	0.1414	0.2013	**0.2475**	0.2337	0.1762
3%	0.0953	0.1636	0.2383	**0.2868**	0.2160
5%	0.1305	0.2031	0.2239	**0.2375**	0.2049
8%	0.1085	0.2122	**0.2909**	0.2226	0.1657
10%	0.1260	0.2073	**0.2448**	0.2286	0.1993
20%	0.1131	0.1990	**0.2557**	0.2349	0.1973

Second, the optimized kernel weights by MK-LSMA+PSR represent the degrees of the importance of the corresponding spectral bands for SU. That is, those spectral bands that cannot well distinguish signature characteristics are given with lower weights. For instance, the spectral signatures of Purdue data at bands 1–5, 40, 57–59, 76–82, 102–107, and 143–148 shown in Fig. 3(c) are nearly the same so that they are supposed to be less beneficial for unmixing. To verify it, Fig. 10(a) shows the optimized kernel weights by MKL-SMA+PSR in Purdue data. We found that the optimized weights of these less powerful bands are obviously lower than those of the other bands. It indicates that the proposed MKL-SMA can effectively put higher emphasis on the discriminative bands, while reducing the influence of the bands with less spectral discrepancy. Such a finding was also observable in the experiments on Pavia data, in which a spectral region with closer signature values appears at bands ranged between 70 and 80. The optimized kernel weights by MKL-SMA+PSR are shown in Fig. 10(b), where the kernel weights significantly drop in this range.

Removing irrelevant features (or bands here) and reducing data dimensionality are feasible ways to improve classification accuracy in the literature. Since a few spectral bands are associated with very low weights in Fig. 10(a), e.g., bands 35–105, the unmixing results of removing those insignificant bands deserve further investigation. In the following, we show that the performance of KLSMA can be further improved if the insignificant bands are removed. To this end, we prioritize all the bands according to their optimized kernel weights in a decreasing order and then perform KLSMA while progressively removing those bands with higher ranking orders. The obtained results are shown in Fig. 11(a), in which the x-axis denotes the number of the removed bands, and the y-axis denotes the

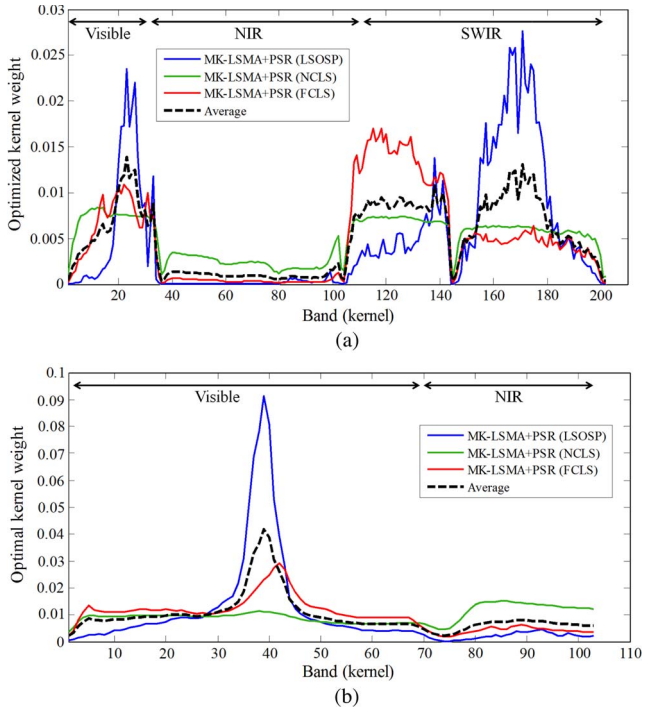


Fig. 10. Plots of the optimized kernel weights β, when the three abundance estimators LSOSP, NCLS, and FCLS are used, respectively. (a) Purdue data. (b) Pavia data. The experiments were conducted with 10% and 3% randomly selected training samples, respectively.

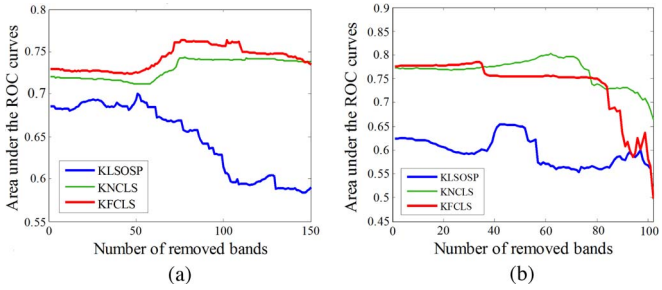


Fig. 11. Performance, in AUC, of the three abundance estimators in the cases where different numbers of the less significant bands are removed on (a) Purdue data and (b) Pavia data.

AUC. It can be seen that the performance, in AUC, of NCLS and FCLS dramatically increases when 70 less significant bands are removed. This suggests that it is unnecessary to exploit full bands to achieve effective SU. Using less significant bands may even hurt the performance of unmixing. It is worth mentioning that the number of the bands with low weights in Fig. 10(a) is about 70. This result manifests that MKL-SMA+PSR can identify those bands that do not really contribute to SU. Such a mechanism can be further utilized to seek redundant or useless bands for other applications.

Third, the convergence issue of the proposed MKL-SMA is investigated. As summarized in Algorithm 1, two key components are performed iteratively in our approach: One is to optimize the kernel weight vector $\boldsymbol{\beta}$ in (15), whereas the other is to solve the conventional KLSMA problem in (5). We have proven that the resulting $\boldsymbol{\beta}$ is globally optimal. However, the off-the-shelf techniques used to solve KLSMA, such as

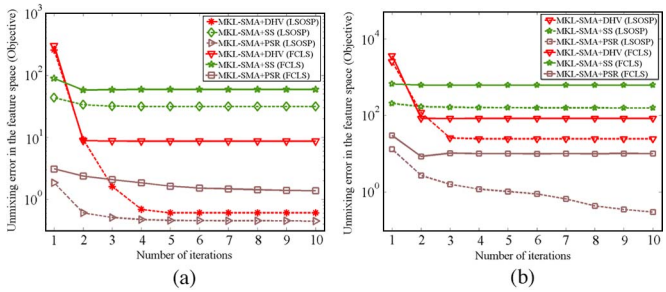


Fig. 12. Objective values along the optimization iterations of the proposed MKL-SMA algorithm on (a) Pavia data and (b) Purdue data.

KNCLS and KFCLS, induce additional constraints to better optimize KLSMA, and thus, they do not guarantee that the global optimum can always be found. Thus, the convergence of the proposed approach is not guaranteed, although it converges in all our experiments. Fig. 12 shows the convergence curves of using MKL-SMA with LSOSP and FCLS estimators, where the x-axis denotes the number of iterations, and the y-axis represents the objective value of (18) at each iteration. It can be seen that the proposed algorithm generally reaches convergence within five iterations. MKL-SMA+PSR requires ten more iterations since it adopts more basis kernels. It is also observed that the use of the constrained estimator, e.g., FCLS, requires fewer number of iterations than the unconstrained estimator, e.g., LSOSP.

Fourth, the computational efficiency of our approach is also discussed. Most existing MKL approaches suffer from the problem of excessive computational cost in optimizing the combination of the basis kernels. However, MKL-SMA shows its advantage over most of the off-the-shelf MKL approaches in the sense that it offers a closed-form solution to determining the optimal convex combination of the given basis kernels.

Fig. 13 shows the comparison of the average computation time in the training and testing phases required by LSMA, KLSMA, and the proposed MKL-SMA, via the FCLS estimator. Compared with LSMA and KLSMA, MKL-SMA unmixes the training data by iteratively optimizing over all the basis kernels and hence induces higher computational cost. However, except for PSR cases where 202 basis kernels are involved, the training time is within a few minutes. It demonstrates the advantage of the closed-form solution in the proposed MKL-SMA. As for the testing stage, the computational costs among different basis kernels are similar, except for those with PSR basis kernels. Therefore, the main computational issue of MKL-SMA locates in the training process. Fortunately, MKL-SMA scales well with the large set of training data.

Finally, the usage of the three types of the basis kernels and the selection of abundance estimators are discussed. The DHV kernels are designed for model selection. In general, the optimal kernel used for KLSMA is unknown in advance. Using DHV kernels with a wide range of parameter values can provide acceptable results. If the additional spatial information is available, the SS kernels can carry both spectral and spatial information. The SS kernels are very helpful for unmixing hyperspectral image scenes that are collected from different altitudes. The PSR kernels are particularly suitable for the

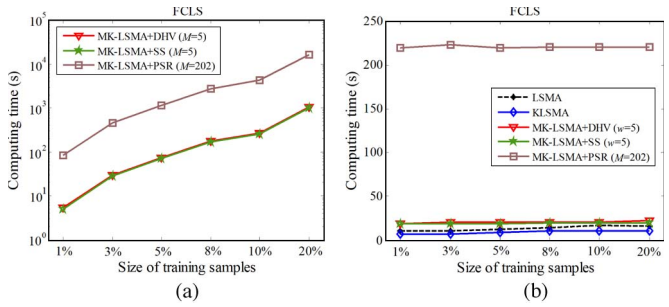


Fig. 13. In the experiments on Purdue data, the computational costs of LSMA, KLSMA, and MKL-SMA, in the (a) training stage and the (b) testing stage. The computing times were calculated under the computer environment with Intel core i7 2.50-GHz central processing unit, 12-GB random access memory, and MATLAB2011b.

images containing useless bands for SU. The produced results can be used for further feature analysis or band selection. On the other hand, the goodness of the abundance estimators for MKL-SMA depends on the data and applications. In general, using FCLS and NCLS are recommended since they commonly produce better unmixing performance than OSP. Other LSMA methods can be used in MKL-SMA as long as its kernel counterpart exists.

VIII. Conclusion

This paper has presented a new framework, which is called MKL-SMA, which integrates MKL into the training process of LSMA and fulfills better unmixing capability for hyperspectral image classification. The proposed MKL-SMA is developed with theoretic merits and boosts the performance of SU in practice. We derived a closed-form solution to optimizing the convex combination of the given basis kernels. Compared with the off-the-shelf MKL algorithms, MKL-SMA scales well with the large number of training data. On the other hand, MKL-SMA utilizes a set of basis kernels to precisely characterize the data and accomplishes much better performance. To demonstrate the flexibility in MKL-SMA, three ways of basis kernel construction were introduced, in which the spectral, spatial, or partial spectral information can be selected as the input feature and combined in the domain of kernel matrices. The experiments conducted on two real hyperspectral images manifest that the proposed MKL-SMA can effectively exploit the rich information carried by the basis kernels and achieve higher classification performance than traditional LSMA and KLSMA in both AUC and classification accuracy. Furthermore, the learned kernel weights by MKL-SMA reveal the intrinsic properties of the data. The knowledge is helpful in designing new feature descriptors for hyperspectral images. The unmixing performance of MKL-SMA critically relies on the basis kernels, which are compiled prior to the unmixing task. How to establish a kernel bank to achieve the best results should be the future work.

Acknowledgment

The authors would like to thank the anonymous reviewers for their comments.

References

[1] D. Landgrebe, "Hyperspectral image data analysis," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 17–28, Jan. 2002.
[2] C.-I. Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*. New York, NY, USA: Academic, 2003.
[3] J. J. Settle and N. A. Drake, "Linear mixing and the estimation of ground cover proportions," *Int. J. Remote Sens.*, vol. 14, no. 6, pp. 1159–1177, Apr. 1993.
[4] N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 44–57, Jan. 2002.
[5] Y. E. Shimabukuro and A. Smith, "The least-squares mixing models to generate fraction images derived from remote sensing multispectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 29, no. 1, pp. 16–20, Jan. 1991.
[6] C.-I. Chang, X.-L. Zhao, M. L. G. Althouse, and J. J. Pan, "Least squares subspace projection approach to mixed pixel classification for hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 36, no. 3, pp. 898–912, May 1998.
[7] C.-I. Chang, S.-S. Chiang, A. Smith, and I. W. Ginsberg, "Linear spectral random mixture analysis for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 2, pp. 375–392, Feb. 2002.
[8] J. C. Harsanyi and C.-I. Chang, "Hyperspectral image classification and dimensionality reduction: An orthogonal subspace projection approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 4, pp. 779–785, Jul. 1994.
[9] C.-I. Chang, "Orthogonal subspace projection (OSP) revisited: A comprehensive study and analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 502–518, Mar. 2005.
[10] C.-I. Chang and B. Ji, "Weighted abundance-constrained linear spectral mixture analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 2, pp. 378–388, Feb. 2006.
[11] C.-I. Chang and D. C. Heinz, "Constrained subpixel target detection for remotely sensed imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 3, pp. 1144–1159, May 2000.
[12] D. C. Heinz and C.-I. Chang, "Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 3, pp. 529–545, Mar. 2001.
[13] J. M. Bioucas-Dias *et al.*, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 354–379, Apr. 2012.
[14] B. Hapke, "Bidirectional reflectance spectroscopy: 1. Theory," *J. Geophys. Res.*, vol. 86, no. B4, pp. 3039–3054, Apr. 1981.
[15] K. J. Guilfoyle, M. L. Althouse, and C.-I. Chang, "A quantitative and comparative analysis of linear and nonlinear spectral mixture models using radial basis function neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 10, pp. 2314–2318, Oct. 2001.
[16] U. Kumar, S. K. Raja, C. Mukhopadhyay, and T. V. Ramachandra, "A multi-layer perceptron based non-linear mixture model to estimate class abundance from mixed pixels," in *Proc. IEEE Students' TechSym*, 2011, pp. 148–153.
[17] G. Camps-Valls and L. Bruzzone, "Kernel-based methods for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 6, pp. 1351–1362, Jun. 2005.
[18] G. Camps-Valls, L. Gomez-Chova, J. Munoz-Mari, J. Vila-Frances, and J. Calpe-Maravilla, "Composite kernels for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 93–97, Jan. 2006.
[19] L. Capobianco, A. Garzelli, and G. Camps-Valls, "Target detection with semisupervised kernel orthogonal subspace projection," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 11, pp. 3822–3833, Nov. 2009.
[20] M. Fauvel, J. A. Benediktsson, J. Chanussot, and J. R. Sveisson, "Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 11, pp. 3804–3814, Nov. 2008.
[21] B. Schölkopf and A. J. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2002.
[22] T. Hofmann, B. Schölkopf, and A. J. Smola, "Kernel methods in machine learning," *Ann. Stat.*, vol. 36, no. 3, pp. 1171–1220, Jun. 2008.
[23] G. Camps-Valls and L. Bruzzone, *Kernel Methods for Remote Sensing Data Analysis*. New York, NY, USA: Wiley, 2009.
[24] H. Kwon and N. M. Nasrabadi, "Kernel orthogonal subspace projection for hyperspectral signal classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 12, pp. 2952–2962, Dec. 2005.
[25] G. Camps-Valls, L. Gomez-Chova, J. Munoz-Mari, J. L. Rojo-Alvarez, and M. Martinez-Ramon, "Kernel-based framework for multitemporal

and multisource remote sensing data classification and change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 6, pp. 1822–1835, Jun. 2008.
[26] B. Guo, S. R. Gunn, R. I. Damper, and J. D. B. Nelson, "Customizing kernel functions for SVM-based hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 17, no. 4, pp. 622–629, Apr. 2008.
[27] J. Broadwater, R. Chellappa, A. Banerjee, and P. Burlina, "Kernel fully constrained least squares abundance estimates," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2007, pp. 4041–4044.
[28] K.-H. Liu, E. Wong, and C.-I. Chang, "Kernel-based linear spectral mixture analysis for hyperspectral image classification," in *Proc. Workshop Hyperspectr. Image Signal Process.—Evol. Remote Sens.*, 2009, pp. 1–4.
[29] K.-H. Liu, E. Wong, E. Y. Du, C. C.-C. Chen, and C.-I. Chang, "Kernel-based linear spectral mixture analysis," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 1, pp. 129–133, Jan. 2012.
[30] M. E. Wong, "Partial volume estimation of magnetic resonance image using linear spectral mixture analysis," M.S. thesis, Dept. Comput. Sci. Elect. Eng., Univ. Maryland, Baltimore, MD, USA, 2011.
[31] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan, "Multiple kernel learning, conic duality, and the SMO algorithm," in *Proc. Int. Conf. Mach. Learn.*, 2004, pp. 1–8.
[32] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "More efficiency in multiple kernel learning," in *Proc. Int. Conf. Mach. Learn.*, 2007, pp. 775–782.
[33] A. Rakotomamonjy, F. R. Bach, S. Canu, and Y. Grandvalet, "SimpleMKL," *J. Mach. Learn. Res.*, vol. 9, pp. 2491–2521, Nov. 2008.
[34] M. Kloft, U. Rückert, and P. L. Bartlett, "A unifying view of multiple kernel learning," in *Proc. Eur. Conf. Mach. Learn. Knowl. Discov. Database*, 2010, pp. 66–81.
[35] M. Gönen and E. Alpaydın, "Multiple kernel learning algorithms," *J. Mach. Learn. Res.*, vol. 12, pp. 2211–2268, Jul. 2011.
[36] Y.-Y. Lin, T.-L. Liu, and C.-S. Fuh, "Multiple kernel learning for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 6, pp. 1147–1160, Jun. 2011.
[37] H.-C. Huang, Y.-Y. Chuang, and C.-S. Chen, "Multiple kernel fuzzy clustering," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 1, pp. 120–134, Feb. 2012.
[38] Y. Gu *et al.*, "Representative multiple kernel learning for classification in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 7, pp. 2582–2865, Jul. 2012.
[39] D. Tuia, G. Camps-Valls, G. Matasci, and M. Kanevski, "Learning relevant image features with multiple-kernel classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 10, pp. 3780–3791, Oct. 2010.
[40] Y. Gu *et al.*, "Multiple-kernel learning-based unmixing algorithm for estimation of cloud fractions with MODIS and CloudSat data," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2012, pp. 1785–1788.
[41] Y. Gu, S. Wang, and X. Jia, "Spectral unmixing in multiple-kernel Hilbert space for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 7, pp. 3968–3981, Jul. 2013.
[42] J. Li, P. R. Marpu, A. Plaza, J. M. Bioucas-Dias, and J. A. Benediktsson, "Generalized composite kernel framework for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 9, pp. 4816–4829, Sep. 2013.

Keng-Hao Liu (M'11) received the B.S. degree in mathematical sciences from National Chengchi University, Taipei, Taiwan, and the M.S. and Ph.D. degrees in electrical engineering from the University of Maryland, Baltimore County, Baltimore, MD, USA, in 2009 and 2011, respectively.

He is currently an Assistant Professor with the Department of Mechanical and Electromechanical Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan. His current research interests include multispectral/hyperspectral image processing, pattern recognition, computer vision, and machine learning.

Yen-Yu Lin (M'12) received the B.S. degree in information management and the M.S. and Ph.D. degrees in computer science and information engineering from National Taiwan University, Taipei, Taiwan, in 2001, 2003, and 2010, respectively.

He is currently an Assistant Research Fellow with the Research Center for Information Technology Innovation, Academia Sinica, Taipei. His current research interests include computer vision, pattern recognition, and machine learning.

Chu-Song Chen (M'96) received the B.S. degree in control engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1989, and the M.S. and Ph.D. degrees from National Taiwan University, Taipei, Taiwan, in 1991 and 1996, respectively.

He is currently a Deputy Director of the Research Center for Information Technology Innovation and a Research Fellow of the Institute of Information Science with Academia Sinica, Taipei. His research interests include pattern recognition, computer vision, signal/image processing, and multimedia analysis.

Dr. Chen is on the Editorial Board of the *Journal of Multimedia* (Academy Publisher), *Machine Vision and Applications* (Springer), and the *Journal of Information Science and Engineering*.