New Concepts in Imaging: Optical and Statistical Models
D. Mary, C. Theys and C. Aime (eds)
EAS Publications Series, **59** *(2013) 417–437*

SUPERVISED NONLINEAR UNMIXING OF HYPERSPECTRAL IMAGES USING A PRE-IMAGE METHODS

N.H. Nguyen[1], J. Chen[1,2], C. Richard[1], P. Honeine[2] and C. Theys[1]

Abstract. Spectral unmixing is an important issue to analyze remotely sensed hyperspectral data. This involves the decomposition of each mixed pixel into its pure endmember spectra, and the estimation of the abundance value for each endmember. Although linear mixture models are often considered because of their simplicity, there are many situations in which they can be advantageously replaced by nonlinear mixture models. In this chapter, we derive a supervised kernel-based unmixing method that relies on a pre-image problem-solving technique. The kernel selection problem is also briefly considered. We show that partially-linear kernels can serve as an appropriate solution, and the nonlinear part of the kernel can be advantageously designed with manifold-learning-based techniques. Finally, we incorporate spatial information into our method in order to improve unmixing performance.

1 Introduction

Pixel-vectors in hyperspectral images are usually mixtures of spectral components associated with a number of pure materials present in the scene (Keshava & Mustard 2002). In order to reveal embedded information, one needs to identify the endmembers present in each pixel and derive the relative proportions of different materials. Under the assumption that the endmembers have been determined *a priori* using some appropriate extraction approaches, see *e.g.*, (Boardman 1993; Nascimento & Bioucas-Dias 2005; Winter 1999), unmixing of hyperspectral images then consists of estimating the fractional abundances.

The abundance estimation problem has most often been solved based on the linear mixing model. Some examples are described in (Dobigeon *et al.* 2009; Heinz & Chang 2001; Honeine & Richard 2012; Theys *et al.* 2009). For instance, the

[1] Université de Nice Sophia-Antipolis, CNRS, Observatoire de la Côte d'Azur, France
[2] Université de Technologie de Troyes, CNRS, France

DOI: 10.1051/eas/1359019

FCLS method presented in (Heinz & Chang 2001) estimates the abundances by minimizing a mean-square-error criterion subject to linear equality and inequality constraints. The geometric strategy described in (Honeine & Richard 2012) reduces to calculate ratios of polyhedra volumes in the space spanned by the hyperspectral pixel-vectors. The main advantage of the former is the convexity of the optimization problem. A very low computational cost characterizes the latter.

In real-world scenes, the interaction between materials can generate nonlinear effects that influence the precision in abundance calculation, and can cause the abundance vectors to violate the non-negativity and the sum-to-one constraints. Nonlinear models can then be introduced to account for these effects, *e.g.*, the generalized bilinear model (Halimi *et al.* 2011), the post non-linear mixing model Jutten & Karhunen (2003), and the intimate model (Hapke 1981). Nonlinear unmixing methods attempt to invert these models and estimate the abundances. In (Halimi *et al.* 2011), a nonlinear unmixing algorithm for general bilinear mixture model was proposed. Based on Bayesian inference, this method however has a high computational complexity and is dedicated to the bilinear model. In (Nascimento & Bioucas-Dias 2009; Raksuntorn & Du 2010), the authors extended the collection of endmembers by adding artificial cross-terms of pure signatures to model light scattering effects on different materials. However, it is not easy to identify which cross-terms should be selected and added to the endmember dictionary. If all the possible cross-terms were considered, the set of endmembers would expand dramatically. Another possible strategy is to use manifold learning approaches such as Isomap (Tenenbaum *et al.* 2000), and LLE (Roweis & Saul 2000), which allow the use of linear methods in a linear space of non-linearly mapped data. Finally, in (Chen *et al.* 2013b), the authors formulated a new kernel-based paradigm that relies on the assumption that the mixing mechanism can be described by a linear mixture of endmember spectra, with additive nonlinear fluctuations defined in a reproducing kernel Hilbert space. This family of models has a clear physical interpretation, and allows to take complex interactions of endmembers into account.

The abundance estimation stage can be accomplished within the context where the abundances of the endmembers are known for some pixels, called training data. A learning process is then applied to estimate the abundances for the remaining pixels. See, *e.g.*, (Altmann *et al.* 2011b; Themelis *et al.* 2010; Tourneret *et al.* 2008). In (Altmann *et al.* 2011b), the map that approximates the abundances for any pixel-vector is a linear combination of radial basis functions. Its weights are estimated based on training samples. An orthogonal least-squares algorithm is then applied to reduce the number of radial basis functions in the model. In this chapter, we show that the learning process for abundance estimation based on training data can be viewed as a pre-image problem (Honeine & Richard 2011). While the mapping from input space to feature space is of primary importance in kernel methods, the reverse mapping from feature space back to input space can be also useful. Solving the pre-image problem within the context of our application consists of approximating the reverse mapping from the high-dimensional space of hyperspectral pixel-vectors to the low-dimensional space of abundance vectors.

We also consider the problem of kernel selection. As in (Chen *et al.* 2013b), we show that partially-linear kernels can serve as an appropriate solution. In this case, the nonlinear part of the kernel can be advantageously designed with manifold-learning-based techniques. We also investigate how to incorporate spatial correlation into the abundance estimation process. Total-variation regularization was introduced with success in (Iordache *et al.* 2011) to perform this task within the context of linear unmixing, and used in (Chen *et al.* 2013a) to extend the kernel-based framework presented in (Chen *et al.* 2013b). In the spirit of these recent results, a pre-image method for nonlinear spectral unmixing coupled with a ℓ_1-type spatial regularization is derived in this chapter.

This chapter is organized as follows. Section 2 describes the problem of nonlinear unmixing of hyperspectral data. It also introduces the pre-image problem within the context of kernel-based data processing. Section 3 solves the pre-image problem with kernel matrix regression in order to perform nonlinear unmixing of hyperspectral data. Section 4 addresses the question of kernel selection. Section 5 aims at solving the same problem with spatial regularization. Section 6 shows experimental results. Finally, Section 7 concludes the chapter.

2 Hyperspectral data unmixing formulated as a pre-image problem

2.1 Hyperspectral image mixing model

Let $\boldsymbol{r} = [r_1, r_2, \ldots, r_L]^\top$ be an observed hyperspectral pixel-vector, with L the number of spectral bands. We shall assume that $\boldsymbol{r}$ is a mixture of R endmember spectra $\boldsymbol{m}_i$. Let us denote by $\boldsymbol{M} = [\boldsymbol{m}_1, \boldsymbol{m}_2, \ldots, \boldsymbol{m}_R]$ the L-by-R endmember matrix, and by $\boldsymbol{\alpha}$ the R-dimensional abundance vector associated with $\boldsymbol{r}$.

We first consider the linear mixing model where any observed pixel is a linear combination of the endmembers, weighted by the fractional abundances, that is,

$$\boldsymbol{r} = \boldsymbol{M}\boldsymbol{\alpha} + \boldsymbol{v} \tag{2.1}$$

where $\boldsymbol{v}$ is a noise vector. The abundance vector $\boldsymbol{\alpha}$ is usually determined by minimizing a cost function, *e.g.*, the mean-square reconstruction error, under the non-negativity and sum-to-one constraints

$$\begin{aligned} &\alpha_i \geq 0, \qquad \forall i \in 1, \ldots, R \\ &\sum_{i=1}^{R} \alpha_i = 1. \end{aligned} \tag{2.2}$$

The above model assumes that abundance vector $\boldsymbol{\alpha}$ lies on a simplex of R vertices. A direct consequence is that pixel-vectors $\boldsymbol{r}$ also lie in a simplex with vertices the R endmember spectra. There are many situations, involving multiple scattering effects, in which model (2.1) may be inappropriate and could be advantageously replaced by a nonlinear one. Consider the general mixing mechanism

$$\boldsymbol{r} = \boldsymbol{\Psi}(\boldsymbol{\alpha}, \boldsymbol{M}) + \boldsymbol{v} \tag{2.3}$$

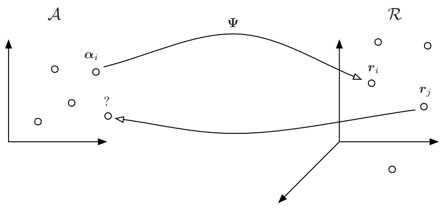

Fig. 1. The basic pre-image problem.

with $\boldsymbol{\Psi}$ an unknown function that defines the interactions between the endmembers in matrix $\boldsymbol{M}$ subject to conditions (2.2).

As illustrated in Figure 1, models (2.1) and (2.3) both rely on a mapping from the low-dimensional input space $\mathcal{A}$ of abundance vectors $\boldsymbol{\alpha}$ into the high-dimensional output space $\mathcal{R}$ of hyperspectral data $\boldsymbol{r}$. In this paper, we consider the problem of estimating abundances as a pre-image problem (Honeine & Richard 2011). Solving the pre-image problem, within a supervised learning context, consists of approximating the reverse mapping that allows to recover the abundance vector $\boldsymbol{\alpha}$ given any pixel-vector $\boldsymbol{r}$, based on training data.

2.2 Estimating a pre-image

This section introduces an original framework, based on the pre-image problem, for supervised unmixing of hyperspectral data. See Figure 2. In order to allow the model to better capture some complex mixing phenomena, we use a reproducing kernel Hilbert space (RKHS) framework in place of $\mathcal{R}$. We shall now review the main definitions and properties related to reproducing kernel Hilbert spaces (Aronszajn 1950).

Let $\mathcal{H}$ denote a Hilbert space of real-valued functions ψ on $\mathcal{R}$, and let $\langle\cdot\,,\cdot\rangle_{\mathcal{H}}$ be the inner product in $\mathcal{H}$. Suppose that the evaluation functional δ_r defined by $\delta_r[\psi] = \psi(\boldsymbol{r})$ is linear with respect to ψ and bounded, for all $\boldsymbol{r}$ in $\mathcal{R}$. By virtue of the Riesz representation theorem, there exists a unique positive definite function $\boldsymbol{r} \mapsto \kappa(\boldsymbol{r}, \boldsymbol{r}')$ in $\mathcal{H}$, denoted by $\kappa(\cdot, \boldsymbol{r}')$ and called *representer of evaluation* at $\boldsymbol{r}'$, which satisfies (Aronszajn 1950)

$$\psi(\boldsymbol{r}') = \langle \psi, \kappa(\cdot, \boldsymbol{r}') \rangle_{\mathcal{H}}, \quad \forall \psi \in \mathcal{H} \tag{2.4}$$

for every fixed $\boldsymbol{r}' \in \mathcal{R}$. A proof of this may be found in (Aronszajn 1950). Replacing ψ by $\kappa(\cdot, \boldsymbol{r})$ in (2.4) yields

$$\kappa(\boldsymbol{r}, \boldsymbol{r}') = \langle \kappa(\cdot, \boldsymbol{r}), \kappa(\cdot, \boldsymbol{r}') \rangle_{\mathcal{H}} \tag{2.5}$$

for all $\boldsymbol{r}$, $\boldsymbol{r}' \in \mathcal{R}$. Equation (2.5) is the origin of the generic term *reproducing kernel* to refer to κ. Denoting by $\boldsymbol{\Phi}$ the map that assigns the kernel function $\kappa(\cdot, \boldsymbol{r})$ to each input data $\boldsymbol{r}$, Equation (2.5) implies that

$$\kappa(\boldsymbol{r}, \boldsymbol{r}') = \langle \boldsymbol{\Phi}(\boldsymbol{r}), \boldsymbol{\Phi}(\boldsymbol{r}') \rangle_{\mathcal{H}}. \tag{2.6}$$

The kernel thus evaluates the inner product of any pair of elements of $\mathcal{R}$ mapped to the space $\mathcal{H}$ without any explicit knowledge of $\boldsymbol{\Phi}$ and $\mathcal{H}$. Within the machine learning area, this key idea is known as the *kernel trick*.

As shown in Figure 2, mapping back to the space $\mathcal{A}$ in order to recover $\boldsymbol{\alpha}$, given any $\kappa(\cdot, \boldsymbol{r})$ in $\mathcal{H}$, is a critical task. Generally, most of the features in $\mathcal{H}$ have no exact pre-image in $\mathcal{A}$. The pre-image problem in kernel-based machine learning has attracted a considerable interest in the last fifteen years. See (Honeine & Richard 2011) for an overview. In (Mika *et al.* 1999), Mika *et al.* introduced the problem and its ill-posedness. They also derived a fixed-point iteration strategy, potentially unstable, to find a solution without any guarantee of optimality. In (Kwok & Tsang 2003), Kwok *et al.* suggested a relationship between the distances in the feature space $\mathcal{H}$ and in the input space $\mathcal{A}$. Applying a multidimensional scaling technique yields an inverse map estimate, and thus a pre-image. This approach has opened the way to a range of other techniques that use training data in both spaces as prior information, such as manifold learning (Roweis & Saul 2000; Tenenbaum *et al.* 2000) and out-of-sample methods (Arias *et al.* 2007; Bengio *et al.* 2003).

In this chapter, we shall use an efficient method for solving the pre-image problem that was recently proposed in (Honeine & Richard 2011). It consists of deriving a transformation that preserves the inner products between training data, in the input space $\mathcal{A}$ and, with some abuse of notation, in the feature space $\mathcal{H}$. Given any $\boldsymbol{r}$, it thus allows to estimate $\boldsymbol{\alpha}$ from $\kappa(\cdot, \boldsymbol{r})$. The next section is dedicated to this approach, and its application to supervised unmixing.

3 Supervised unmixing

Given a set of training data $\{(\boldsymbol{\alpha}_1, \boldsymbol{r}_1), \ldots, (\boldsymbol{\alpha}_n, \boldsymbol{r}_n)\}$, we seek the pre-image $\boldsymbol{\alpha}$ in $\mathcal{A}$ of some arbitrary $\kappa(\cdot, \boldsymbol{r})$ of $\mathcal{H}$. The proposed approach consists of two stages: First, learning the reverse map; Then, estimating the pre-image.

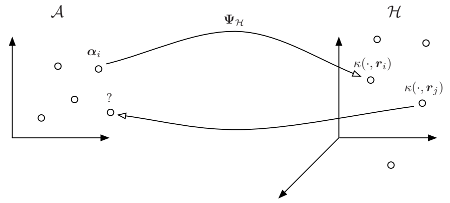
$\mathcal{A}$
$\boldsymbol{\Psi}_{\mathcal{H}}$
$\mathcal{H}$
$\boldsymbol{\alpha}_i$
$\kappa(\cdot, \boldsymbol{r}_i)$
$\kappa(\cdot, \boldsymbol{r}_j)$
?

Fig. 2. The pre-image problem.

3.1 Stage 1: Learning the reverse map

By virtue of the Representer Theorem (Schölkopf *et al.* 2000), we know that we can limit our investigation to the space spanned by the n kernel functions

$\{\kappa(\cdot,\boldsymbol{r}_1),\ldots,\kappa(\cdot,\boldsymbol{r}_n)\}$. Let us focus on only a subspace spanned by ℓ functions to be determined, denoted by $\{\psi_1,\ldots,\psi_\ell\}$ with $\ell \leq n$, of the form

$$\psi_k = \sum_{i=1}^{n} \lambda_{ki}\,\kappa(\cdot,\boldsymbol{r}_i), \quad k = 1,\ldots,\ell. \tag{3.1}$$

We consider the analysis operator C: $\mathcal{H} \rightarrow \mathbb{R}^\ell$ defined as

$$C\varphi = [\langle\varphi,\psi_1\rangle_{\mathcal{H}} \ldots \langle\varphi,\psi_\ell\rangle_{\mathcal{H}}]^\top. \tag{3.2}$$

Note that the k-th entry of the representation of any kernel function $\kappa(\cdot,\boldsymbol{r})$ is given by

$$\langle\kappa(\cdot,\boldsymbol{r}),\psi_k\rangle_{\mathcal{H}} = \sum_{i=1}^{n} \lambda_{ki}\,\kappa(\boldsymbol{r},\boldsymbol{r}_i). \tag{3.3}$$

It is interesting to note that $\langle\kappa(\cdot,\boldsymbol{r}),\psi_k\rangle_{\mathcal{H}} = \psi_k(\boldsymbol{r})$ by the reproducing property of the space $\mathcal{H}$. The kernel function $\kappa(\cdot,\boldsymbol{r})$ is thus represented by the ℓ-length vector

$$\boldsymbol{\psi}_r = [\psi_1(\boldsymbol{r})\,\psi_2(\boldsymbol{r})\,\ldots\,\psi_\ell(\boldsymbol{r})]^\top \tag{3.4}$$

with $\psi_k(\boldsymbol{r})$ defined in (3.3). In order to fully define the analysis operator C, that is, to estimate the λ_{ki}, we suggest to consider the following relationship between any inner product in the input space $\mathcal{A}$ and, with some abuse of notation, with its counterpart in the feature space $\mathcal{H}$

$$\boldsymbol{\alpha}_i^\top\boldsymbol{\alpha}_j = \boldsymbol{\psi}_{r_i}^\top\boldsymbol{\psi}_{r_j} + \epsilon_{ij}, \quad \forall\, i,j = 1,\ldots,n \tag{3.5}$$

where ϵ_{ij} denotes the lack-of-fit of the above model. Note that there is no constraint on the analysis functions ψ_k, except their form (3.1) and the goodness-of-fit constraint (3.5), because reconstruction from expansion coefficients is not considered. Let us now estimate the λ_{ki} in (3.3) so that the empirical variance of ϵ_{ij} is minimal, that is,

$$\min_{\lambda_{11},\ldots,\lambda_{\ell n}} \frac{1}{2}\sum_{i,j=1}^{n}(\boldsymbol{\alpha}_i^\top\boldsymbol{\alpha}_j - \boldsymbol{\psi}_{r_i}^\top\boldsymbol{\psi}_{r_j})^2 + \eta\,P(\psi_1,\ldots,\psi_\ell) \tag{3.6}$$

where P is a regularization function, and η a tunable parameter used to control the tradeoff between fitting the data and smoothness of the solution. We shall use ℓ_2-norm penalization in this paper, defined as

$$P(\psi_1,\ldots,\psi_\ell) = \sum_{k=1}^{\ell}\|\psi_k\|_{\mathcal{H}}^2. \tag{3.7}$$

The optimization problem can be expressed in matrix form as

$$\min_{\boldsymbol{L}} \frac{1}{2}\|\boldsymbol{A} - \boldsymbol{K}\boldsymbol{L}^\top\boldsymbol{L}\boldsymbol{K}\|_F^2 + \eta\,\mathrm{trace}(\boldsymbol{L}^\top\boldsymbol{L}\boldsymbol{K}) \tag{3.8}$$

where $\boldsymbol{A}$ and $\boldsymbol{K}$ are the Gram matrices with (i, j)-th entries defined as $\boldsymbol{\alpha}_i^\top \boldsymbol{\alpha}_j$ and $\kappa(\boldsymbol{r}_i, \boldsymbol{r}_j)$, respectively, and $\boldsymbol{L}$ is the matrix with (i, j)-th entry given by λ_{ij}.

Taking the derivative of this cost with respect to $\boldsymbol{L}^\top \boldsymbol{L}$, rather than $\boldsymbol{L}$, we get

$$\hat{\boldsymbol{L}}^\top \hat{\boldsymbol{L}} = \boldsymbol{K}^{-1}(\boldsymbol{A} - \eta \boldsymbol{K}^{-1})\boldsymbol{K}^{-1}. \tag{3.9}$$

In the following, we shall show that only $\hat{\boldsymbol{L}}^\top \hat{\boldsymbol{L}}$ is needed to calculate the pre-image.

3.2 Stage 2: Estimate the pre-image

Let us first consider the case of any function φ of $\mathcal{H}$, which can be written as follows

$$\varphi = \sum_{i=1}^{n} \phi_i \, \kappa(\cdot, \boldsymbol{r}_i) + \varphi^\perp \tag{3.10}$$

with $\varphi^\perp$ an element of the orthogonal complement to the subspace spanned by the kernel functions $\kappa(\cdot, \boldsymbol{r}_i)$. Note at this point that the parameters ϕ_i are supposed to be known. In any case, they can be evaluated by projecting φ onto the subspace spanned the n kernel functions $\kappa(\cdot, \boldsymbol{r}_i)$, that is, by solving

$$\min_{\boldsymbol{\phi}} \|\varphi - \sum_{i=1}^{n} \phi_i \, \kappa(\cdot, \boldsymbol{r}_i)\|_{\mathcal{H}}^2. \tag{3.11}$$

This yields the n-by-n linear system of equations $\boldsymbol{K}\boldsymbol{\phi} = \boldsymbol{\varphi}_0$, where $\boldsymbol{\varphi}_0$ is the vector with i-th entry $\varphi(\boldsymbol{r}_i)$, and $\boldsymbol{\phi}$ stands for the vector with i-th entry ϕ_i, for $i = 1, \ldots, n$. Referring back to Equation (3.10), the k-th entry of the representation of φ by the analysis operator C, denoted by $\boldsymbol{\varphi}$, is given by

$$\langle \varphi, \psi_k \rangle_{\mathcal{H}} = \sum_{i,j=1}^{n} \phi_i \, \hat{\lambda}_{kj} \, \kappa(\boldsymbol{r}_i, \boldsymbol{r}_j), \tag{3.12}$$

where $\hat{\lambda}_{kj}$ is the (k, j)-th entry of the matrix $\hat{\boldsymbol{L}}$ estimated during Stage 1. This directly implies that $\boldsymbol{\varphi} = \hat{\boldsymbol{L}}\boldsymbol{K}\boldsymbol{\phi}$. Minimizing now the lack-of-fit (3.5), with respect to the pre-image $\boldsymbol{\alpha}$ given $\boldsymbol{\varphi}$, between $\boldsymbol{\alpha}^\top \boldsymbol{\alpha}_i$ and $\boldsymbol{\varphi}^\top \boldsymbol{\psi}_{r_i}$ for $i = 1, \ldots, n$, leads to the optimization problem

$$\begin{aligned} \hat{\boldsymbol{\alpha}} &= \arg\min_{\boldsymbol{\alpha}} \frac{1}{2} \|\boldsymbol{\Lambda}^\top \boldsymbol{\alpha} - \boldsymbol{K}\hat{\boldsymbol{L}}^\top \hat{\boldsymbol{L}} \boldsymbol{K}\boldsymbol{\phi}\|^2 \\ &= \arg\min_{\boldsymbol{\alpha}} \frac{1}{2} \|\boldsymbol{\Lambda}^\top \boldsymbol{\alpha} - (\boldsymbol{A} - \eta \boldsymbol{K}^{-1})\boldsymbol{\phi}\|^2 \end{aligned} \tag{3.13}$$

subject to the non-negativity and sum-to-one constraints (2.2). Here $\boldsymbol{\Lambda}$ is the matrix with i-th column the vector $\boldsymbol{\alpha}_i$.

Let us now consider the particular case where one seeks the pre-image $\boldsymbol{\alpha}$ of some kernel function $\kappa(\cdot, \boldsymbol{r}_0)$. Substituting φ by $\kappa(\cdot, \boldsymbol{r}_0)$ in Equation (3.11) leads us to the system $\boldsymbol{K}\boldsymbol{\phi} = \boldsymbol{\kappa}_0$, where $\boldsymbol{\kappa}_0$ is the vector with i-th entry $\kappa(\boldsymbol{r}_i, \boldsymbol{r}_0)$.

Minimizing the appropriate lack-of-fit (3.5) with respect to the pre-image $\boldsymbol{\alpha}$ leads us to the optimization problem

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \frac{1}{2}\|\boldsymbol{\Lambda}^\top\boldsymbol{\alpha} - (\boldsymbol{A} - \eta\boldsymbol{K}^{-1})\boldsymbol{K}^{-1}\boldsymbol{\kappa}_0\|^2 \tag{3.14}$$

subject to the non-negativity and sum-to-one constraints (2.2). This convex optimization problem can be solved using the FCLS strategy to deal with the equality constraints (Heinz & Chang 2001), associated with a nonnegative least-mean-square algorithm. See, *e.g.*, (Chen *et al.* 2011) for an overview.

4 Kernel selection

The kernel function $\kappa(\cdot, \boldsymbol{r})$ maps the measurements $\boldsymbol{r}$ into a very high, even infinite, dimensional space $\mathcal{H}$. It characterizes the solution space for the possible nonlinear relationships between input data $\boldsymbol{\alpha}$ and output data $\boldsymbol{r}$. Classic examples of kernels are the Gaussian kernel $\kappa(\boldsymbol{r}_i, \boldsymbol{r}_j) = \exp\big(-\|\boldsymbol{r}_i - \boldsymbol{r}_j\|^2/2\sigma^2\big)$, with σ the kernel bandwidth, and the q-th degree non-homogeneous polynomial kernel $\kappa(\boldsymbol{r}_i, \boldsymbol{r}_j) = (1 + \boldsymbol{r}_i^\top\boldsymbol{r}_j)^q$, with $q \in \mathbb{N}^*$. We shall now make some suggestions for selecting specific kernels, before testing it in the next section. On the one hand, we shall briefly propose to design the kernel directly from data by using manifold learning techniques. On the other hand, we shall present a partially-linear kernel that has proved its efficiency for nonlinear unmixing (Chen *et al.* 2013b).

4.1 Kernel selection based on manifold learning techniques

In (Ham *et al.* 2003), the manifold learning problem is treated within the context of kernel PCA. The process of revealing the underlying structure of data is viewed as a nonlinear dimensionality reduction method, based on local information with LLE (Roweis & Saul 2000), or geodesic distance with Isomap (Tenenbaum *et al.* 2000). These techniques can be used to design kernels that preserve some aspects of the manifold structure of the space $\mathcal{R}$ to which the vectors $\boldsymbol{r}_i$ belong, in the feature space $\mathcal{H}$ of the functions $\kappa(\cdot, \boldsymbol{r}_i)$. We used such techniques in (Nguyen *et al.* 2012) for unmixing of hyperspectral data.

As an example, we consider radial basis kernels of the form $\kappa(\boldsymbol{r}_i, \boldsymbol{r}_j) = f(\|\boldsymbol{r}_i - \boldsymbol{r}_j\|)$ with $f \in \mathcal{C}_\infty$. A sufficient condition for this class of kernels to be positive-definite, and thus valid, is the complete monotonicity of the function f, which can be expressed as follows,

$$(-1)^k f^{(k)}(r) \geq 0, \quad \forall r \geq 0 \tag{4.1}$$

where $f^{(k)}$ denotes the k-th order derivative of f (Cucker & Smale 2002). Instead of using the euclidean distance $d_{ij} = \|\boldsymbol{r}_i - \boldsymbol{r}_j\|$ with f, we can use pairwise distances $d_{\text{iso},ij} = \|\boldsymbol{r}_i - \boldsymbol{r}_j\|_{\text{iso}}$ provided by Isomap. This approach consists of constructing a symmetric adjacency graph using a nearest neighborhood based criterion, and applying Dijkstra algorithm to compute the shortest path along edges of this graph, between each pair of data. Unfortunately, the Gram matrix $\boldsymbol{K}_{\text{iso}}$ constructed in such a way has no guarantee of being positive definite. This difficulty can be overcome by using multidimensional scaling, which maps the data into a low-dimensional euclidean subspace where edge lengths are best preserved. An alternative is to force matrix $\boldsymbol{K}_{\text{iso}}$ to be positive definite using one of the approaches describes in (Muñoz & Diego 2006).

4.2 Partially-linear Kernel

Model (2.1)-(2.2) assumes that the relationship between the abundance vectors $\boldsymbol{\alpha}_i$ and the hyperspectral pixel-vectors $\boldsymbol{r}_i$ is linear. There are however many situations, involving multiple scattering effects, in which this model may be inappropriate and could be advantageously replaced by a nonlinear one. In (Chen *et al.* 2013a,b), we studied mixing models defined by a linear trend parameterized by the abundance vector, combined with a nonlinear fluctuation term. Extensive experiments, both with synthetic and real scenes, illustrated the flexibility and and the effectiveness of this class of models. In the spirit of these derivations, we suggest to consider kernels of the form

$$\kappa(\boldsymbol{r}_i, \boldsymbol{r}_j) = (1-\gamma)\, \boldsymbol{r}_i^\top \boldsymbol{\Sigma}\, \boldsymbol{r}_j + \gamma\, \kappa'(\boldsymbol{r}_i, \boldsymbol{r}_j) \tag{4.2}$$

with $\kappa'(\boldsymbol{r}_i, \boldsymbol{r}_j)$ a reproducing kernel, $\boldsymbol{\Sigma}$ a non-negative matrix, and γ a parameter in $[0, 1]$ to adjust the balance between the linear and the nonlinear kernels.

In all the experiments, we shall use the above kernel with $\boldsymbol{\Sigma} = (\boldsymbol{M}\boldsymbol{M}^\top)^\dagger$

$$\kappa(\boldsymbol{r}_i, \boldsymbol{r}_j) = (1-\gamma)\, \boldsymbol{r}_i^\top (\boldsymbol{M}\boldsymbol{M}^\top)^\dagger\, \boldsymbol{r}_j + \gamma\, \kappa'(\boldsymbol{r}_i, \boldsymbol{r}_j) \tag{4.3}$$

where $(\cdot)^\dagger$ stands for the pseudo-inverse. Indeed, for $\gamma = 0$, it can be shown that this kernel leads to the least-mean-square estimate of the abundance vector in the case of a linear mixing scenario.

5 Spatial regularization applied to supervised unmixing

5.1 Formulation

In the previous section, we showed how to estimate the abundances by learning a reverse mapping. This approach consisted of considering pixel vectors as if they were independent from their neighboring pixels. However, a fundamental property of remotely sensed data is that they convey multivariate information into a 2D pictorial representation. Hyperspectral analysis techniques can thus benefit from the inherent spatial-spectral duality in hyperspectral scenes. Following this idea, researchers exploited spatial information for endmember estimation (Martin & Plaza 2011; Rogge *et al.* 2007; Zortea & Plaza 2009) and pixel vectors classification (Fauvel *et al.* 2012, to appear; Li *et al.* 2011). Recently, spatial processing methods were also derived for semi-supervised unmixing (Chen *et al.* 2013a). In this section, we aim at improving the pre-image method by incorporating such information.

Following (Iordache *et al.* 2011), an optimization method based on split variable iteration is proposed to deal with this problem that suffers the non-smoothness of the regularization term.

Let us denote by $\boldsymbol{\Delta}$ the matrix of the abundance vectors, that is, $\boldsymbol{\Delta} = [\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_n]$. In order to take the spatial relationship among pixels into consideration, we suggest to consider a general cost function of the form

$$J(\boldsymbol{\Delta}) = J_{\text{err}}(\boldsymbol{\Delta}) + \nu J_{\text{sp}}(\boldsymbol{\Delta}) \tag{5.1}$$

subject to the non-negativity constraint imposed on each entry of $\boldsymbol{\Delta}$, and the sum-to-one constraint imposed on each column of the matrix $\boldsymbol{\Delta}$, namely, on each $\boldsymbol{\alpha}_i$. For ease of notation, these two physical constraints will be expressed by

$$\begin{aligned} &\boldsymbol{\Delta} \succeq \mathbf{0} \\ &\boldsymbol{\Delta}^\top \mathbf{1}_R = \mathbf{1}_N. \end{aligned} \tag{5.2}$$

The function $J_{\text{err}}(\boldsymbol{\Delta})$ represents the modeling error, and $J_{\text{sp}}(\boldsymbol{\Delta})$ is a regularization term to promote similarity of the fractional abundances within neighboring pixels. The non-negative parameter ν is used to control the trade-off between data fidelity and pixel similarity.

To take spatial relationships among pixels into consideration, let us consider the following regularization function

$$J_{\text{sp}}(\boldsymbol{\Delta}) = \sum_{i=1}^{n} \sum_{j \in \mathcal{N}(i)} \|\boldsymbol{\alpha}_i - \boldsymbol{\alpha}_j\|_1 \tag{5.3}$$

where $\| \, \|_1$ denotes the vector ℓ_1-norm, and $\mathcal{N}(i)$ is the set of neighbors of the pixel i. This regularization term promotes spatial homogeneity as neighboring pixels may be characterized by similar abundances for most materials. Without any loss of generality, in this paper, we restrict the neighborhood of the pixel i by taking the 4 nearest pixels $i-1$ and $i+1$ (row adjacency), $i-w$ and $i+w$ (column adjacency). In this case, let us define the $(n \times n)$ matrices $\boldsymbol{H}_{\leftarrow}$ and $\boldsymbol{H}_{\rightarrow}$ as the two linear operators that compute the difference between any abundance vector and its left-hand neighbor, and right-hand neighbor, respectively. Similarly, let $\boldsymbol{H}_{\uparrow}$ and $\boldsymbol{H}_{\downarrow}$ be the linear operators that compute that difference with the top neighbor and the down neighbor, respectively. With these notations, the regularization function (5.3) can be rewritten in matrix form as

$$J_{\text{sp}}(\boldsymbol{\Delta}) = \|\boldsymbol{\Delta}\boldsymbol{H}\|_{1,1} \tag{5.4}$$

with $\boldsymbol{H}$ the $(n \times 4n)$ matrix $\left(\boldsymbol{H}_{\leftarrow} \, \boldsymbol{H}_{\rightarrow} \, \boldsymbol{H}_{\uparrow} \, \boldsymbol{H}_{\downarrow}\right)$ and $\| \, \|_{1,1}$ the sum of the ℓ_1-norms of the columns of a matrix. Note that this regularization function is convex but non-smooth.

Considering both the modeling error and the regularization term, the optimization problem becomes

$$\min_{\boldsymbol{\Delta}} \sum_{i=1}^{n} \frac{1}{2}\|\boldsymbol{\Lambda}^\top \boldsymbol{\alpha}_i - (\boldsymbol{A} - \eta \boldsymbol{K}^{-1})\boldsymbol{K}^{-1}\boldsymbol{\kappa}_i\|^2 + \nu\|\boldsymbol{\Delta H}\|_{1,1}$$
$$\text{subject to} \quad \boldsymbol{\Delta} \succeq 0 \quad \text{and} \quad \boldsymbol{\Delta}^\top \mathbf{1}_R = \mathbf{1}_N \tag{5.5}$$

where ν controls the trade-off between model fitting in each pixel and similarity among neighboring pixels. For ease of notation, in the following, we shall write $\boldsymbol{\Delta} \in \mathcal{S}_{+1}$ to denote the non-negativity and sum-to-one constraints.

5.2 Solution

Even though the optimization problem (5.5) is convex, it cannot be solved easily because of the non-smooth regularization term. In order to overcome this drawback, we rewrite it in the following equivalent form

$$\min_{\boldsymbol{\Delta}\in\mathcal{S}_{+1}} \sum_{i=1}^{n} \frac{1}{2}\|\boldsymbol{\Lambda}^\top \boldsymbol{\alpha}_i - (\boldsymbol{A} - \eta \boldsymbol{K}^{-1})\boldsymbol{K}^{-1}\boldsymbol{\kappa}_i\|^2 + \nu\|\boldsymbol{U}\|_{1,1}$$
$$\text{subject to} \quad \boldsymbol{V} = \boldsymbol{\Delta} \quad \text{and} \quad \boldsymbol{U} = \boldsymbol{VH} \tag{5.6}$$

where we have introduced two new matrices $\boldsymbol{U}$ and $\boldsymbol{V}$, and two additional constraints. The matrix $\boldsymbol{U}$ will allow us to decouple the non-smooth ℓ_1-norm regularization functional from the main quadratic problem. The matrix $\boldsymbol{V}$ will relax connections between pixels. This variable-splitting approach was initially introduced in (Goldstein & Osher 2009).

As studied in (Goldstein & Osher 2009), the split Bregman iteration algorithm is an efficient method to deal with a broad class of ℓ_1-regularized problems. By applying this framework to (5.5), the following formulation is obtained

$$\begin{aligned}\boldsymbol{\Delta}^{(k+1)}, \boldsymbol{V}^{(k+1)}, \boldsymbol{U}^{(k+1)} = \underset{\boldsymbol{\Delta}\in\mathcal{S}_{+1},\boldsymbol{V},\boldsymbol{U}}{\arg\min} & \sum_{i=1}^{n} \frac{1}{2}\|\boldsymbol{\Lambda}^\top \boldsymbol{\alpha}_i - (\boldsymbol{A} - \eta \boldsymbol{K}^{-1})\boldsymbol{K}^{-1}\boldsymbol{\kappa}_i\|^2 \\ & + \nu\|\boldsymbol{U}\|_{1,1} + \frac{\zeta}{2}\|\boldsymbol{\Delta} - \boldsymbol{V} - \boldsymbol{D}_1^{(k)}\|_F^2 + \frac{\zeta}{2}\|\boldsymbol{U} - \boldsymbol{VH} - \boldsymbol{D}_2^{(k)}\|_F^2\end{aligned} \tag{5.7}$$

with

$$\begin{aligned}\boldsymbol{D}_1^{(k+1)} &= \boldsymbol{D}_1^{(k)} + \left(\boldsymbol{V}^{(k+1)} - \boldsymbol{\Delta}^{(k+1)}\right) \\ \boldsymbol{D}_2^{(k+1)} &= \boldsymbol{D}_2^{(k)} + \left(\boldsymbol{V}^{(k+1)}\boldsymbol{H} - \boldsymbol{U}^{(k+1)}\right)\end{aligned} \tag{5.8}$$

where $\| \, \|_F^2$ denotes the matrix Frobenius norm, and ζ is a positive parameter. Because we have split the components of the cost function, we can now solve the above minimization problem efficiently by iteratively minimizing the cost function with respect to $\boldsymbol{\Delta}$, $\boldsymbol{V}$ and $\boldsymbol{U}$ separately. We shall now describe the three steps that have to be performed.

5.2.1 Step 1: Optimization with respect to $\boldsymbol{\Delta}$

The optimization problem (5.7) reduces to

$$\boldsymbol{\Delta}^{(k+1)} = \arg\min_{\boldsymbol{\Delta}\in\mathcal{S}_{+1}} \sum_{i=1}^{n} \frac{1}{2}\left(\|\boldsymbol{\Lambda}^\top\boldsymbol{\alpha}_i - (\boldsymbol{A}-\eta\boldsymbol{K}^{-1})\boldsymbol{K}^{-1}\boldsymbol{\kappa}_i\|^2 + \zeta\|\boldsymbol{\alpha}_i - \boldsymbol{\xi}_i^{(k)}\|^2\right) \tag{5.9}$$

where $\boldsymbol{\xi}_i^{(k)} = \boldsymbol{V}_i^{(k)} + \boldsymbol{D}_{1,i}^{(k)}$. Here, $\boldsymbol{V}_i^{(k)}$ and $\boldsymbol{D}_{1,i}^{(k)}$ denote the i-th column of $\boldsymbol{V}^{(k)}$ and $\boldsymbol{D}_1^{(k)}$, respectively. It can be observed that this problem can be decomposed into subproblems, each one involving an abundance vector $\boldsymbol{\alpha}_i$. This results from the use of the matrix $\boldsymbol{V}$ in the split iteration algorithm (5.7).

Let us now solve the local optimization problem

$$\begin{aligned}\boldsymbol{\alpha}_i^{(k+1)} = \arg\min_{\boldsymbol{\alpha}_i} &\ \frac{1}{2}\|\boldsymbol{\Lambda}^\top\boldsymbol{\alpha}_i - (\boldsymbol{A}-\eta\boldsymbol{K}^{-1})\boldsymbol{K}^{-1}\boldsymbol{\kappa}_i\|^2 + \zeta\|\boldsymbol{\alpha}_i - \boldsymbol{\xi}_i^{(k)}\|^2\\ \text{subject to} &\ \ \boldsymbol{\alpha}_i \succeq 0\\ &\ \ \boldsymbol{\alpha}_i^\top\boldsymbol{1}_R = 1.\end{aligned} \tag{5.10}$$

Estimating $\boldsymbol{\alpha}_i$ reduces to a quadratic optimization problem with linear equality and inequality constraints, which can be efficiently solved by off-the-shelf methods. This process has to be repeated for $i = 1,\ldots,n$ in order to get $\boldsymbol{\Delta}^{(k+1)}$.

5.2.2 Step 2: Optimization with respect to $\boldsymbol{V}$

The optimization problem (5.7) now reduces to

$$\boldsymbol{V}^{(k+1)} = \arg\min_{\boldsymbol{V}} \|\boldsymbol{\Delta}^{(k+1)} - \boldsymbol{V} - \boldsymbol{D}_1^{(k)}\|_F^2 + \|\boldsymbol{U}^{(k)} - \boldsymbol{V}\boldsymbol{H} - \boldsymbol{D}_2^{(k)}\|_F^2. \tag{5.11}$$

Equating to zero the derivative of (5.11) with respect to $\boldsymbol{V}$ leads to

$$\left(\boldsymbol{\Delta}^{(k+1)} - \boldsymbol{V} - \boldsymbol{D}_1^{(k)}\right) + \left(\boldsymbol{U}^{(k)} - \boldsymbol{V}\boldsymbol{H} - \boldsymbol{D}_2^{(k)}\right)\boldsymbol{H}^\top = 0 \tag{5.12}$$

whose solution is then given by

$$\boldsymbol{V}^{(k+1)} = \left(\boldsymbol{\Delta}^{(k+1)} - \boldsymbol{D}_1^{(k)} + (\boldsymbol{U}^{(k)} - \boldsymbol{D}_2^{(k)})\boldsymbol{H}^\top\right)(\boldsymbol{I} + \boldsymbol{H}\boldsymbol{H}^\top)^{-1}. \tag{5.13}$$

As a conclusion, this subproblem has an explicit solution that involves the inverse of the matrix $(\boldsymbol{I} + \boldsymbol{H}\boldsymbol{H}^\top)$. The latter can be evaluated once the neighborhood relationship is defined.

5.2.3 Step 3: Optimization with respect to $\boldsymbol{U}$

The last optimization problem we have to consider is as follows

$$\boldsymbol{U}^{(k+1)} = \arg\min_{\boldsymbol{U}} \nu\|\boldsymbol{U}\|_{1,1} + \frac{\zeta}{2}\|\boldsymbol{U} - \boldsymbol{V}^{(k+1)}\boldsymbol{H} - \boldsymbol{D}_2^{(k)}\|_F^2. \tag{5.14}$$

Its solution can be expressed via the well-known soft threshold function

$$\boldsymbol{U}^{(k+1)} = \mathrm{Thresh}\left(\boldsymbol{V}^{(k+1)}\boldsymbol{H} + \boldsymbol{D}_2^{(k)}, \frac{\nu}{\zeta}\right) \tag{5.15}$$

where Thresh$(\cdot, \tau)$ denotes the component-wise application of the soft threshold function defined as

$$\mathrm{Thresh}(x, \tau) = \mathrm{sign}(x)\ \max(|x| - \tau, 0). \tag{5.16}$$

As in Step 2, the third subproblem has an explicit solution. The computational time is also almost negligible.

To conclude, the problem (5.6) is solved by iteratively applying (5.7) and (5.8), where the optimization of (5.7) can be performed by applying Steps 1 to 3. These iterations continue until some stopping criterion is satisfied. It can be shown that, if the problem (5.7) has a solution $\boldsymbol{\Delta}^*$ given any $\zeta > 0$, then the generated sequence $\boldsymbol{\Delta}^{(k)}$ converges to $\boldsymbol{\Delta}^*$ (Eckstein & Bertsekas 1992).

6 Simulation results

In this section, we shall experiment the pre-image method with and without spatial regularization in order to evaluate the benefit of using the latter. We shall compare it with state-of-the-art methods.

6.1 Experiments with the pre-image method

Spatial regularization is not addressed in this subsection. Two synthetic scenes were generated with real material spectra, on the one hand from abundance vectors uniformly distributed in the simplex defined by the non-negativity and the sum-to-one constraints, and on the other hand from abundance vectors lying on a manifold.

6.1.1 Experiments on synthetic images with uniformly-distributed abundances

We shall first report some experimental results on synthetic images, which were generated by linear and nonlinear mixing of several endmember signatures. The materials that were considered are alunite, calcite, epidote, kaolinite, and buddingtonite. There spectra were extracted from the ENVI software library, and consisted of 420 contiguous bands, covering wavelength ranging from 0.3951 to 2.56 micrometers. They were used to synthesize 50 × 50 images with different mixture models, each providing $n = 2500$ pixels for evaluating and comparing several unmixing algorithms. These three models were: the linear model, the bilinear mixture model with attenuation factors $\gamma_{ij} = 1$ (Halimi *et al.* 2011), and the post-nonlinear mixing model (PNMM) defined by (Jutten & Karhunen 2003)

$$\boldsymbol{r} = (\boldsymbol{M}\boldsymbol{\alpha})^{\xi} + \boldsymbol{v} \tag{6.1}$$

where $(\cdot)^{\xi}$ denotes the exponential value ξ applied to each entry of the input vector. This parameter was set to 0.7. The abundance vectors $\boldsymbol{\alpha}_i$, with $i = 1, \ldots, 2500$, were uniformly generated in the simplex defined by the non-negativity and the sum-to-one constraints. In the first scene, only three materials were selected to generate images: epidote, kaolinite, buddingtonite. In the second scene, five materials were used: alunite, calcite, epidote, kaolinite, buddingtonite. These scenes were corrupted with an additive white Gaussian noise $\boldsymbol{v}$ with two levels of SNR, 15 dB and 30 dB.

The following algorithms were considered in our experiments.

- **The Fully Constrained Least Square method (FCLS)** (Heinz & Chang 2001): This algorithm relies on a semi-supervised learning setting in the sense that unmixing is performed using endmember spectra as prior information. It is based on a linear mixture model, and provides the optimal solution in the least-mean-square sense subject to the non-negativity and the sum-to-one constraints.
- **The Kernel Fully Constrained Least Square method (KFCLS)** (Broadwater *et al.* 2007): This semi-supervised nonlinear algorithm is the kernel-based counterpart of FCLS, obtained by replacing all the inner products in FCLS by kernel functions. In the experiments, as for our pre-image algorithm, we used the Gaussian kernel with kernel bandwidth $\sigma = 4$.
- **The Bayesian algorithm derived for generalized bilinear model (BilBay)** (Halimi *et al.* 2011): This semi-supervised method is based on appropriate prior distributions for the unknown abundances, which must satisfy the non-negativity and sum-to-one constraints, and then derives joint posterior distribution of these parameters. A Metropolis-within-Gibbs algorithm is used to estimate the unknown model parameters.
- **The RBF-with-OLS method (RBF-OLS)** (Altmann *et al.* 2011a): As our pre-image method, this supervised algorithm aims at learning a nonlinear reverse mapping from $\mathcal{R}$ to $\mathcal{A}$. The estimator is a linear combination of radial basis functions with centers chosen from the training data through an OLS procedure.
- **The pre-image algorithm proposed in this paper**: The inhomogeneous polynomial kernel (P) of degree $d = 2$, the Gaussian kernel (G) with kernel bandwidth $\sigma = 4$, and the partially-linear kernel (PL) associating a linear kernel and a Gaussian kernel with $\sigma = 4$. The parameter γ combining these two kernels, and the regularization coefficient η, were set to 10^{-1} and 10^{-3}.

The cardinality of the training data set was fixed to 200 in order to reach an appropriate compromise between the computational cost and the performance. The root mean square error (RMSE) between the true and the estimated abundance vectors $\boldsymbol{\alpha}_i$ and $\hat{\boldsymbol{\alpha}}_i$ was used to compare the performance of the five algorithms. Results for Scene 1 and Scene 2 unmixing, with three and five endmember materials, are reported in Table 1 and Table 2, respectively.

Table 1. Scene 1 (three materials): RMSE comparison.

	SNR = 30 dB			SNR = 15 dB		
	linear	bilinear	PNMM	linear	bilinear	PNMM
FCLS	0.0037	0.0758	0.0604	0.0212	0.0960	0.0886
KFCLS	0.0054	0.2711	0.2371	0.0296	0.2694	0.2372
BilBay	0.0384	0.0285	0.1158	0.1135	0.1059	0.1191
RBF-OLS	0.0144	0.0181	0.0170	0.0561	0.0695	0.0730
Pre-image method (P)	0.0139	0.0221	0.0129	0.0592	0.0601	0.0764
Pre-image method (G)	0.0086	0.0104	0.0103	0.0422	0.0561	0.0597
Pre-image method (PL)	0.0072	0.0096	0.0098	0.0372	0.0395	0.0514

Table 2. Scene 2 (five materials): RMSE comparison.

	SNR = 30 dB			SNR = 15 dB		
	linear	bilinear	PNMM	linear	bilinear	PNMM
FCLS	0.0134	0.1137	0.1428	0.0657	0.1444	0.1611
KFCLS	0.0200	0.2051	0.1955	0.0890	0.1884	0.1572
BilBay	0.0585	0.0441	0.1741	0.1465	0.1007	0.1609
RBF-OLS	0.0200	0.0236	0.0259	0.0777	0.0805	0.0839
Pre-image method (P)	0.025	0.0267	0.0348	0.0905	0.0903	0.1000
Pre-image method (G)	0.0186	0.0233	0.0245	0.0775	0.0778	0.0875
Pre-image method (PL)	0.0148	0.0184	0.0203	0.0636	0.0616	0.0763

Consider first the semi-supervised algorithms. The FCLS method achieves a very low RMSE for linearly-mixed images because it was initially derived for the linear mixing model. As a consequence, it produces a large RMSE with nonlinearly-mixed images. The KFCLS should have overcome this drawback. It however performs worse than FCLS, even with nonlinearly-mixed images as it does not clearly investigate nonlinear interactions between materials (Chen *et al.* 2013b). BilBay algorithm was derived for the bilinear mixing model, and thus achieves very good performance with bilinearly-mixed images. Nevertheless, its performance severely degrades when dealing with a nonlinear mixing model for which it was not originally designed. Consider now the supervised algorithms. The pre-image method and RBF-OLS outperforms all the semi-supervised algorithms when dealing with non-linearly mixed images. Of course, they make use of more information to achieve this performance. Our approach is however much more flexible than RBF-OLS since it can be associated with any reproducing kernel. In particular, as already observed in (Chen *et al.* 2013b), the experiments demonstrate the benefit of using a partially-linear kernel.

6.1.2 Experiment on synthetic images: Test with swiss-roll data

In order to highlight the flexibility of our approach with respect to kernel selection, we shall now show that kernels designed with manifold learning techniques can be advantageously used. Let us consider the well-known swiss-role artificial data set for illustration purpose. It consists of random samples in a two-dimensional

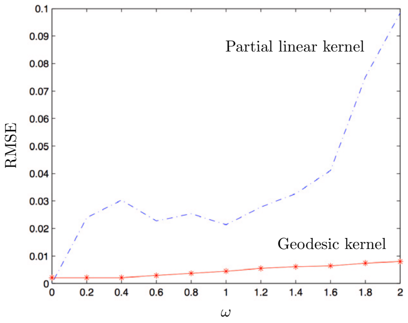

Fig. 3. Geodesic kernel *vs.* partially-linear kernel in the case where data lie in a manifold.

simplex, transformed into a three-dimensional nonlinear manifold by projection on a swiss-roll structure. The non-linearity of the swiss-roll data is parameterized by a variable ω. The coordinate of a data point $\boldsymbol{r}_i$ as a function of the local abundance $\boldsymbol{\alpha}_i$ are expressed by

$$\begin{cases} r_{i1} = \alpha_{i1}\sin(\omega\alpha_{i1}) + 1 \\ r_{i2} = \alpha_{i1}\cos(\omega\alpha_{i1}) + 1 \\ r_{i3} = \alpha_{i2} + 1. \end{cases} \tag{6.2}$$

Following the sum-to-one constraint, the abundance of the third endmembers can be generated by $\alpha_{i3} = 1 - (\alpha_{i1} + \alpha_{i2})$. By setting a single abundance equal to one, and the two others to zero, we obtain the endmember spectra

$$\begin{cases} \boldsymbol{m}_1 = [\sin(\omega) + 1,\ \cos(\omega) + 1,\ 1]^\top \\ \boldsymbol{m}_2 = [1,\ 1,\ 2]^\top \\ \boldsymbol{m}_3 = [1,\ 1,\ 1]^\top. \end{cases} \tag{6.3}$$

Swiss-roll data unmixing was performed with our pre-image algorithm, based on 100-sample training sets, for ω values in the interval $[0, 2]$. The partially-linear kernel with Gaussian kernel whose bandwidth was set to $\sigma = 4$, and the kernel based on geodesic distances provided by Isomap, were considered. The geodesic kernel was constructed using the geodesic distance matrix provided by Isomap and Djisktra algorithms. Note that this matrix was converted into a positive definite matrix using a technique described in (Muñoz & Diego 2006). Figure 3 clearly shows that the geodesic kernel is much more appropriate than the partially-linear kernel in the case where the data lie in a manifold, and the performance of the algorithm is quite steady even for large ω values.

6.2 *Experiments with the spatially-regularized pre-image method*

Two spatially correlated abundance maps were generated for the following experiments. The endmembers were randomly selected from the spectral library ASTER (Baldridge *et al.* 2009). Each signature of this library has reflectance values measured over 224 spectral bands, uniformly distributed in the interval $3 - 12$ micrometers. Two synthetic abundance maps identical to (Iordache *et al.* 2011) were used.

The first data cube, denoted by IM1, and containing 50×50 pixels, was generated using five signatures randomly selected from the ASTER library. Pure regions and mixed regions involving between 2 and 5 endmembers, distributed spatially in the form of square regions, were generated. The background pixels were defined as mixtures of the same 5 endmembers with the abundance vector $[0.1149, 0.0741, 0.2003, 0.2055, 0.4051]^\top$. The first row in Figure 4 shows the true fractional abundances for each endmember. The reflectance samples were generated with the bilinear mixing model, based on the 5 endmembers, and corrupted by a zero-mean white Gaussian noise $\boldsymbol{v}_i$ with a SNR of 20 dB, namely,

$$\boldsymbol{r}_i = \boldsymbol{M}\boldsymbol{\alpha}_i + \sum_{p=1}^{R}\sum_{q=p+1}^{R} \alpha_{n,p}\,\alpha_{n,q}\,\boldsymbol{m}_p \otimes \boldsymbol{m}_q + \boldsymbol{v}_i \tag{6.4}$$

with $\otimes$ the Hadamard product.

The second data cube, denoted by IM2 and containing 100×100 mixed pixels, was generated using 5 endmember signatures. The abundance maps of the endmembers are the same as for the image DC2 in (Iordache *et al.* 2011). The first row of Figure 5 depicts the true distribution of these 5 materials. Spatially homogeneous areas with sharp transitions can be clearly observed. Based on these abundance maps, an hyperspectral data cube was generated with the bilinear model (6.4) applied to the 5 endmember spectral signatures. The scene was also corrupted by a zero-mean white Gaussian noise $\boldsymbol{v}_i$ with a SNR of 20 dB.

Algorithms with and without spatial regularization were compared in order to demonstrate the effectiveness of adding this type of information. Unsupervised algorithms that do not use spatial information, were also considered for comparison purpose. The tuning parameters of the algorithms were set using preliminary experiments on independent data, via a simple search over predefined grids.

1. The linear unmixing method FCLS (Heinz & Chang 2001): The regularization parameter λ was varied in $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$ in order to determine the best configuration.

2. The pre-image algorithm without spatial regularization: The partially-linear kernel with $\gamma = 0.1$ was considered. It was associated with the Gaussian kernel. The bandwidth of the latter was varied in [0.5, 5], and finally set to 4. The regularization parameter η of the pre-image algorithm was varied in $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$, and was finally set to 10^{-3}. The size of the training set was set to 200.

3. The pre-image algorithm with spatial regularization: The same parameter values as above were considered for this algorithm in order to clearly evaluate the interest of taking spatial information into account. The parameters ζ and ν, which are specifically related to the spatial regularization, were tuned as explained below.

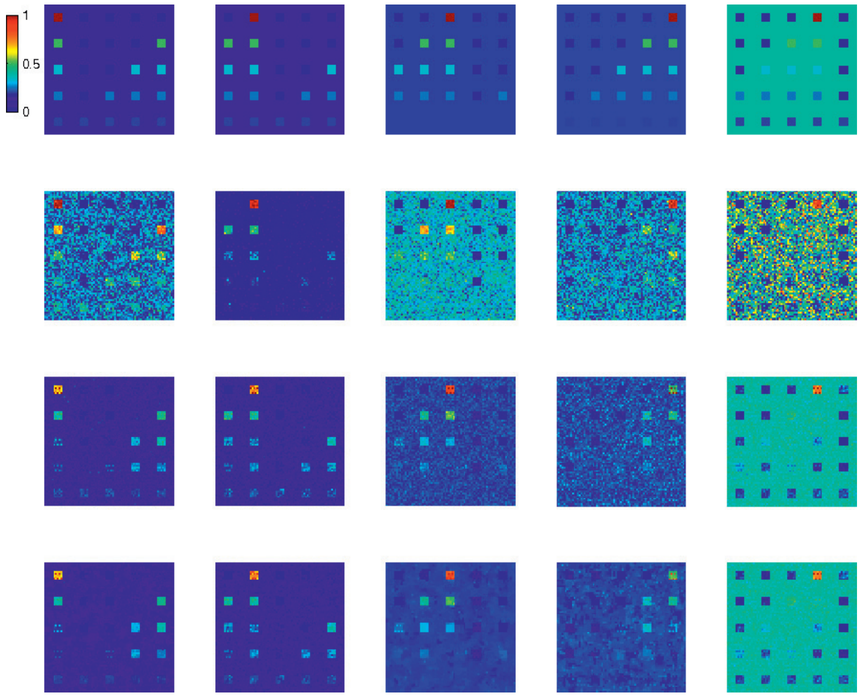

Fig. 4. Estimated abundance maps for IM1. From *top* to *bottom*: true abundance map, FCLS, pre-image method, pre-image method with spatial regularization.

With the image IM1, the preliminary tests led us to $\lambda = 10^{-2}$ for FCLS, and $\zeta = 1$, $\nu = 0.1$ for the proposed algorithm. With the image IM2, these tests led to $\lambda = 0.01$ for FCLS, and $\zeta = 20$, $\nu = 0.5$ for the proposed algorithm.

The estimated abundances are presented in Figures 4 and 5. The reconstruction errors (RMSE) are reported in Table 3. For both images IM1 and IM2, it can be observed that when applied on nonlinearly mixed data, the linear unmixing method FCLS has large reconstruction errors. The proposed pre-image method allows to notably reduce this error in the mean sense, but the estimated abundance maps are corrupted by a noise that partially masks spatial structures of the materials. Finally, the proposed spatially-regularized method has lower reconstruction error and clearer abundance maps. Using spatial information obviously brings advantages to the nonlinear unmixing process.

Table 3. Comparison of the RMSE for IM1 and IM2.

Algorithms	IM1	IM2
FCLS	0.1426	0.0984
pre-image	0.0546	0.0712
pre-image with reg.	0.0454	0.0603

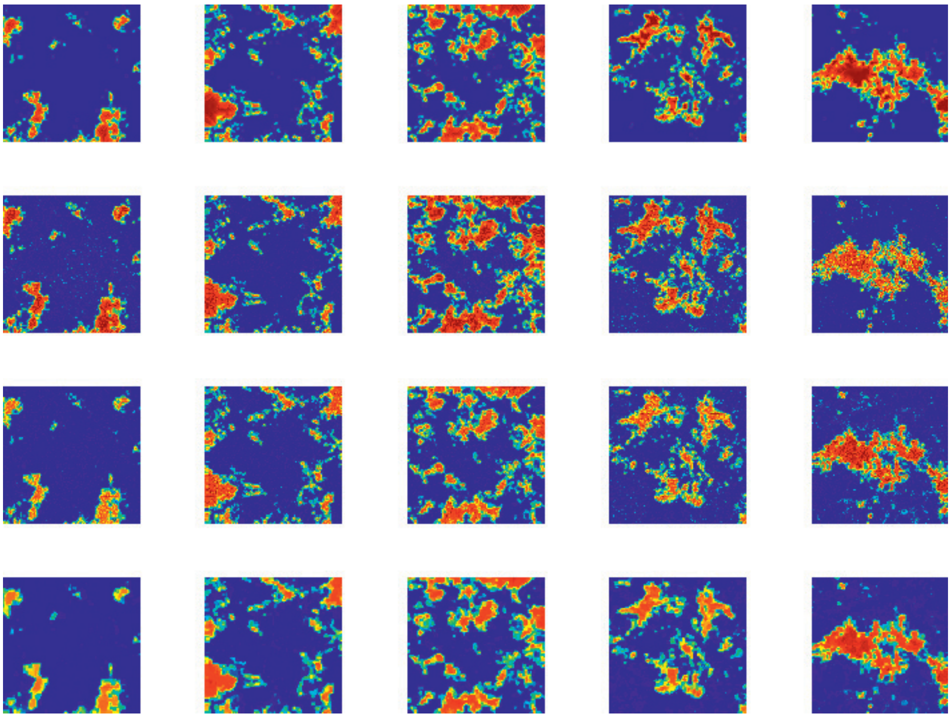

Fig. 5. Estimated abundance maps for IM2. From *top* to *bottom*: true abundance map, FCLS, pre-image method, pre-image method with spatial regularization.

7 Conclusion

In this chapter, we introduced an hyperspectral unmixing algorithm based on the pre-image principle, which is usually addressed by the community of machine learning. Our contribution is two-fold in the sense that the pre-image algorithm described here, and its spatially-regularized counterpart, are both original. We showed that these techniques can be advantageously applied for supervised unmixing provided that labeled pixel-vectors are available.

References

Altmann, Y., Dobigeon, N., McLaughlin, S., & Tourneret, J.-Y., 2011a, in Proc. IEEE IGARSS

Altmann, Y., Halimi, A., Dobigeon, N., & Tourneret, J.-Y., 2011b, in Proc. IEEE IGARSS

Arias, P., Randall, G., & Sapiro, G., 2007, in Proc. IEEE CVPR

Aronszajn, N., 1950, Trans. Amer. Math. Soc., 68, 337

Baldridge, A.M., Hook, S.J., Grove, C.I., & Rivera, G., 2009, Remote Sensing Env., 113, 711

Bengio, Y., Paiement, J.-F., Vincent, P., *et al.*, 2003, in Proc. NIPS

Boardman, J., 1993, in Proc. AVIRIS, 1, 11

Broadwater, J., Chellappa, R., Banerjee, A., & Burlina, P., 2007, in Proc. IEEE IGARSS, 4041

Chen, J., Richard, C., Bermudez, J.-C.M., & Honeine, P., 2011, IEEE Trans. Sig. Proc., 59, 5225

Chen, J., Richard, C., & Honeine, P., 2013a, IEEE Trans. Geosci. Remote Sens.

Chen, J., Richard, C., & Honeine, P., 2013b, IEEE Trans. Sig. Proc., 61, 480

Cucker, F., & Smale, S., 2002, Bull. Am. Math. Soc., 39, 1

Dobigeon, N., Moussaoui, S., Coulon, M., Tourneret, J.-Y., & Hero, A.O., 2009, IEEE Trans. Sig. Proc., 57, 4355

Eckstein, J., & Bertsekas, D., 1992, Math. Prog., 55, 293

Fauvel, M., Tarabalka, Y., Benediktsson, J.A., Chanussot, J., & Tilton, J., 2012, Proc. IEEE, to appear

Goldstein, T., & Osher, S., 2009, SIAM J. Imaging Sci., 2, 323

Halimi, A., Altmann, Y., Dobigeon, N., & Tourneret, J.-Y., 2011, IEEE Trans. Geosci. Remote Sens., 49, 4153

Ham, J., Lee, D. D., Mika, S., & Schölkopf, B., 2003, A kernel view of the dimensionality reduction of manifolds, Tech. Rep. TR-110 (Max-Planck-Institut für biologische Kybernetik)

Hapke, B., 1981, J. Geophys. Res., 86, 3039

Heinz, D.C., & Chang, C.-I., 2001, IEEE Trans. Geosci. Remote Sens., 39, 529

Honeine, P., & Richard, C., 2011, IEEE Signal Proc. Mag., 28, 77

Honeine, P., & Richard, C., 2012, IEEE Trans. Geosci. Remote Sens., 50, 2185

Iordache, M.-D., Bioucas-Dias, J.-M., & Plaza, A., 2011, in Proc. IEEE WHISPERS

Jutten, C., & Karhunen, J., 2003, in Proc. ICA, 245

Keshava, N., & Mustard, J.F., 2002, IEEE Signal Proc. Mag., 19, 44

Kwok, J., & Tsang, I., 2003, in Proc. ICML

Li, J., Bioucas-Dias, J.-M., & Plaza, A., 2011, IEEE Trans. Geosci. Remote Sens., 50, 809

Martin, G., & Plaza, A., 2011, IEEE Geosci. Remote Sens. Lett., 8, 745

Mika, S., Schölkopf, B., Smola, A., *et al.*, 1999, in Proc. NIPS

Muñoz, A., & Diego, I.M., 2006, in Lecture Notes in Computer Science, Structural, Syntactic, and Statistical Pattern Recognition, Vol. 4109, ed. D.-Y. Yeung, J. Kwok, A. Fred, F. Roli & D. Ridder (Springer), 764

Nascimento, J.M.P., & Bioucas-Dias, J.M., 2005, IEEE Trans. Geosci. Remote Sens., 43, 898

Nascimento, J.M.P., & Bioucas-Dias, J.-M., 2009, in Proc. SPIE, 7477

Nguyen, N.H., Richard, C., Honeine, P., & Theys, C., 2012, in Proc. IEEE IGARSS

Raksuntorn, N., & Du, Q., 2010, IEEE Geosci. Remote Sens. Lett., 7, 836

Rogge, D.M., Rivard, B., Zhang, J., *et al.*, 2007, Remote Sensing Env., 110, 287

Roweis, S., & Saul, L., 2000, Science, 2323

Schölkopf, B., Herbrich, R., & Williamson, R., 2000, A generalized representer theorem, Tech. Rep. NC2-TR-2000-81, NeuroCOLT, Royal Holloway College (University of London, UK)

Tenenbaum, J.B., de Silva, V., & Langford, J.C., 2000, Science, 290, 2319

Themelis, K., Rontogiannis, A.A., & Khoutroumbas, K., 2010, in Proc. IEEE ICASSP, 1194

Theys, C., Dobigeon, N., Tourneret, J.-Y., & Lanteri, H., 2009, in Proc. IEEE SSP

Tourneret, J.-Y., Dobigeon, N., & Chang, C.-I., 2008, IEEE Trans. Sig. Proc., 5, 2684

Winter, M.E., 1999, Proc. SPIE Spectrometry V, 3753, 266

Zortea, M., & Plaza, A., 2009, IEEE Trans. Geosci. Remote Sens., 47, 2679