

Joint unmixing-deconvolution algorithms for hyperspectral images

Yingying Song, El-Hadi Djermoune, David Brie
 CRAN, CNRS Université de Lorraine
 Vandoeuvre-lès-Nancy, France
 firstname.lastname@univ-lorraine.fr

Cédric Richard
 CNRS, OCA Université Côte d'Azur
 Nice, France
 cedric.richard@unice.fr

Abstract—This paper combines supervised linear unmixing and deconvolution problems to increase the resolution of the abundance maps for industrial imaging systems. The joint unmixing-deconvolution (JUD) algorithm is introduced based on the Tikhonov regularization criterion for offline processing. In order to meet the needs of industrial applications, the proposed JUD algorithm is then extended for online processing by using a block Tikhonov criterion. The performance of JUD is increased by adding a non-negativity constraint which is implemented in a fast way using the quadratic penalty method and fast Fourier transform. The proposed algorithm is then assessed using both simulated and real hyperspectral images.

Index Terms—Hyperspectral image unmixing, hyperspectral image deconvolution, non-negative Tikhonov regularization

I. INTRODUCTION

Due to the limited spatial resolution of hyperspectral imagers, the surface area covered by any pixel of the acquired image may contain different chemical materials characterized by their spectral responses. Each pixel of a hyperspectral image is thus a mixture of several spectral signatures of different pure materials called endmembers. Unmixing hyperspectral data aims at the estimation of the endmembers and their fractional abundances inside each pixel area [1]. Blind unmixing refers to the situation in which both the endmembers and abundances have to be estimated while in supervised unmixing one seeks to estimate the abundances with known endmembers; this is the case considered here. In this context, enforcing non-negativity plays an important role and has motivated the development of a number of methods (see for example [2]).

The most widely used mixture model representation is the linear mixing model which is the one adopted in this work. However, let us mention that in recent years many attention was paid to the so-called non-linear mixing models and a comprehensive treatment of different non-linear mixing models and resulting non-linear unmixing algorithms is presented in [3]. The linear mixing model implicitly assumes that each endmember is perfectly represented by a single spectral signature. This strong assumption often does not hold for real datasets and many works focussed on taking into account the so-called spectral variability (see for example [4], [5]).

Hyperspectral image unmixing may lead to a loss of resolution because of the blurring introduced by the imager. The

idea developed in this paper is to combine unmixing and deconvolution to recover abundance maps with fine spatial resolution. This problem is referred to as the *joint unmixing-deconvolution* (JUD) problem. As far as we know, only a few works are addressing this problem [6]–[8]. In particular, [7] gives a theoretical analysis of the JUD and shows that deconvolution is beneficial for endmembers estimation and that, in the blind unmixing case, it is preferable to first deblur the hyperspectral image and then to estimate the endmembers [8]. In [6], the endmembers being assumed to be known, a JUD method with total variation regularization is proposed. The goal of this paper is to propose fast approaches to the JUD problem which can be implemented online using a sliding block approach similar to the one proposed in [9]. Actually, it should be seen as a preliminary step toward the development of recursive estimators based on LMS (least mean squares).

The remainder of the paper is organized as follows. Section II introduces the observation model. Section III presents the unconstrained JUD problem based on Tikhonov regularization for both offline (batch) and online processing, the latter being based on a sliding block scheme. The non-negativity constrained JUD is then introduced in Section IV and an efficient implementation based on the fast Fourier transform is detailed. In Section V, experiments are conducted on a simulated image and a real hyperspectral image. Finally, the conclusions are drawn in Section VI.

II. OBSERVATION MODEL

Consider a hyperspectral image $\mathbf{X} \in \mathbb{R}^{N \times K \times P}$ where N , K and P are the number of measurements related to across track, along track and spectral dimensions, respectively. The first two dimensions are spatial dimensions. Each column vector $\mathbf{s}_r = [s_r^1, \dots, s_r^P]^\top$, for $p = 1, \dots, P$ in matrix $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_R] \in \mathbb{R}^{P \times R}$ represents an endmember spectrum with R the number of abundances. Denoting $\mathbf{X}^p \in \mathbb{R}^{N \times K}$ the 2D spatial image corresponding to p -th wavelength and \mathbf{x}_k^p the k -th column of \mathbf{X}^p , the hyperspectral image \mathbf{X} can be organized into a single vector \mathbf{x} :

$$\begin{aligned} \mathbf{x}^p &\triangleq \text{col}\{\mathbf{x}_k^p\}_{k=1}^K, \\ \mathbf{x} &\triangleq \text{col}\{\mathbf{x}^p\}_{p=1}^P. \end{aligned} \quad (1)$$

where $\text{col}\{\cdot\}$ stacks its vector arguments on top of each other. We use similar notations for the observed image by

This work has been supported by the FUI AAP 2015 Trispirabois Project and the Conseil Régional de Lorraine.

substituting \mathbf{y} to \mathbf{x} . Similarly, let us denote by \mathbf{A}_r the 2D spatial image of r -th abundance and $\mathbf{a}_{k,r}$ the k -th column of \mathbf{A}_r . Then, the abundances can also be organized into a single vector \mathbf{a} :

$$\begin{aligned}\mathbf{a}_r &\triangleq \text{col}\{\mathbf{a}_{k,r}\}_{k=1}^K, \\ \mathbf{a} &\triangleq \text{col}\{\mathbf{a}_r\}_{r=1}^R.\end{aligned}\quad (2)$$

Thus, the linear mixing model can be rewritten as:

$$\mathbf{x} = (\mathbf{S} \otimes \mathbf{I}_{NK})\mathbf{a} \quad (3)$$

where \otimes stands for the Kronecker product and matrix \mathbf{I}_J denotes the $J \times J$ identity matrix.

Consider now the 2D spatial convolution kernel $\mathbf{H}^{*p} \in \mathbb{R}^{M \times L} = [\mathbf{h}_L^p, \dots, \mathbf{h}_1^p]$, for $p = 1, \dots, P$, with $\mathbf{h}_\ell^p = [h_{M,\ell}^p, \dots, h_{1,\ell}^p]^\top$. Let $\mathbf{H}_{(J)}^p \in \mathbb{R}^{N \times N}$ be the circulant-block-circulant convolution (block-Toeplitz) matrix with first block column and first block row given by $[\mathbf{H}_{(L+1)/2}^{p\top}, \dots, \mathbf{H}_L^{p\top}, \mathbf{0}_{N \times (J-L)N}^\top, \mathbf{H}_1^{p\top}, \dots, \mathbf{H}_{(L-1)/2}^{p\top}]^\top$ and $[\mathbf{H}_{(L+1)/2}^p, \dots, \mathbf{H}_1^p, \mathbf{0}_{N \times (J-L)N}, \mathbf{H}_L^p, \dots, \mathbf{H}_{(L+1)/2+1}^p]$, respectively. Here matrix $\mathbf{H}_\ell^p \in \mathbb{R}^{N \times N}$ is a Toeplitz matrix with first column and first row given by $[h_{(M+1)/2,\ell}^p, \dots, h_{M,\ell}^p, 0, \dots, 0, h_{1,\ell}^p, \dots, h_{(M-1)/2,\ell}^p]^\top$ and $[h_{(M+1)/2,\ell}^p, \dots, h_{1,\ell}^p, 0, \dots, 0, h_{M,\ell}^p, \dots, h_{(M+1)/2+1,\ell}^p]$, respectively. The global convolution matrix is then $\mathbf{H} \triangleq \text{blkdiag}\{\mathbf{H}_{(K)}^p\}_{p=1}^P$ which is a block-diagonal matrix of size $NKP \times NKP$. Combining the linear mixing model in (3) with the convolution model $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{e}$, where \mathbf{e} is a zero-mean iid measurement noise, yields the joint convolution-mixing model:

$$\mathbf{y} = \mathbf{H}(\mathbf{S} \otimes \mathbf{I}_{NK})\mathbf{a} + \mathbf{e}. \quad (4)$$

Matrix $(\mathbf{S} \otimes \mathbf{I}_{NK})$ is of size $NKP \times NKR$. The storage and multiplication of matrices \mathbf{H} and $(\mathbf{S} \otimes \mathbf{I}_{NK})$ require a memory which is way too large. Thus we focus on the case when the convolution kernel is the same along the spectral dimension p , i.e. $\mathbf{H}_{(K)}^1 = \dots = \mathbf{H}_{(K)}^P$. In this case the convolution kernel is acting on each abundance. This leads to the following model requiring smaller memory:

$$\mathbf{y} = (\mathbf{S} \otimes \mathbf{H}_{(K)}^1)\mathbf{a} + \mathbf{e}. \quad (5)$$

III. JOINT UNMIXING-DECONVOLUTION FOR HYPERSPECTRAL IMAGES

A. Offline joint unmixing-deconvolution

Estimating abundance \mathbf{a} from the blurred noisy image \mathbf{y} using the joint unmixing-deconvolution (JUD) method is derived by introducing the following Tikhonov criterion:

$$\mathcal{J}_{\text{Tikh}}(\mathbf{a}) = \frac{1}{2} \|\mathbf{y} - (\mathbf{S} \otimes \mathbf{H}_{(K)}^1)\mathbf{a}\|_2^2 + \frac{\eta_a}{2} \|(\mathbf{I}_R \otimes \mathbf{\Delta}_{(K)})\mathbf{a}\|_2^2 \quad (6)$$

where the regularizer $\|(\mathbf{I}_R \otimes \mathbf{\Delta}_{(K)})\mathbf{a}\|_2^2$, controlled by parameter η_a , promotes the spatial smoothness on the abundances and the matrix $\mathbf{\Delta}_{(K)}$ is a Laplacian operator. First, define the Laplacian matrix:

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} = [\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \boldsymbol{\delta}_3] \quad (7)$$

and assume periodic boundary conditions. Let $\mathbf{\Delta}_i$ be a circulant matrix of size $N \times N$ corresponding to column $\boldsymbol{\delta}_i$. Then, the matrix $\mathbf{\Delta}_{(J)}$ is a circulant-block-circulant matrix of size $NJ \times NJ$ with first block column given by $[\mathbf{\Delta}_2^\top, \mathbf{\Delta}_3^\top, \mathbf{0}_{N \times (J-3)N}^\top, \mathbf{\Delta}_1^\top]^\top$ and first block row given by $[\mathbf{\Delta}_2, \mathbf{\Delta}_1, \mathbf{0}_{N \times (J-3)N}, \mathbf{\Delta}_3]$. The minimizer of (6) can be expressed by:

$$\begin{aligned}\hat{\mathbf{a}} &= \left(\mathbf{S}^\top \mathbf{S} \otimes \mathbf{H}_{(K)}^{1\top} \mathbf{H}_{(K)}^1 + \mathbf{I}_R \otimes \eta_a \mathbf{\Delta}_{(K)}^\top \mathbf{\Delta}_{(K)} \right)^{-1} \\ &\cdot (\mathbf{S}^\top \otimes \mathbf{H}_{(K)}^{1\top})\mathbf{y}.\end{aligned}\quad (8)$$

The computational cost of solution (8) is mainly related to the inversion of a matrix which does not enjoy any Kronecker structure. However, by exploiting the circulant-block-circulant structure of matrix $\mathbf{\Delta}_{(K)}$, an efficient implementation of the estimator can be derived. This will be discussed in section IV. For very large images, batch processing is time-consuming. In the next paragraph, we present an online method based on a sliding block strategy allowing to process data streams with controlled numerical complexity.

B. Online unmixing and deconvolution

Following the same idea introduced in [9] for online deconvolution, the online joint unmixing-deconvolution is addressed as a sliding block processing. Consider a block of size Q , assumed to be odd, and define:

$$\begin{aligned}\mathbf{y}'_k{}^p &\triangleq \text{col} \left\{ \mathbf{y}_{k-Q+q}^p \right\}_{q=1}^Q, & \mathbf{y}'_k &\triangleq \text{col} \left\{ \mathbf{y}'_k{}^p \right\}_{p=1}^P \\ \mathbf{a}'_{k,r} &\triangleq \text{col} \left\{ \mathbf{a}_{k-Q+q,r} \right\}_{q=1}^Q, & \mathbf{a}'_k &\triangleq \text{col} \left\{ \mathbf{a}'_{k,r} \right\}_{r=1}^R.\end{aligned}\quad (9)$$

The sequential blurred noisy mixing model is then given by:

$$\mathbf{y}'_k = (\mathbf{S} \otimes \mathbf{H}_{(Q)}^1)\mathbf{a}'_k + \mathbf{e}'_k \quad (10)$$

where \mathbf{e}'_k is the vectorization of noise in the block.

The block version of criterion (6) is given by:

$$\begin{aligned}\mathcal{J}_{\text{BT}}(\mathbf{a}'_k) &= \frac{1}{2} \left\| \mathbf{y}'_k - (\mathbf{S} \otimes \mathbf{H}_{(Q)}^1)\mathbf{a}'_k \right\|_2^2 \\ &+ \frac{\eta_a}{2} \|(\mathbf{I}_R \otimes \mathbf{\Delta}_{(Q)})\mathbf{a}'_k\|_2^2\end{aligned}\quad (11)$$

for which the minimizer is:

$$\begin{aligned}\hat{\mathbf{a}}'_k &= \left(\mathbf{S}^\top \mathbf{S} \otimes \mathbf{H}_{(Q)}^{1\top} \mathbf{H}_{(Q)}^1 + \eta_a \mathbf{I}_R \otimes \mathbf{\Delta}_{(Q)}^\top \mathbf{\Delta}_{(Q)} \right)^{-1} \\ &\cdot (\mathbf{S}^\top \otimes \mathbf{H}_{(Q)}^{1\top})\mathbf{y}'_k.\end{aligned}\quad (12)$$

The final result corresponds to the central part of $\hat{\mathbf{a}}'_k$, that is:

$$\hat{\mathbf{a}}_{k-(Q-1)/2} = \mathbf{C}\hat{\mathbf{a}}'_k \quad (13)$$

where matrix $\mathbf{C} \triangleq \mathbf{I}_R \otimes [\mathbf{0}_{N \times N(Q-1)/2}, \mathbf{I}_N, \mathbf{0}_{N \times N(Q-1)/2}]$ selects the $(Q-1)/2$ -th block of data $\hat{\mathbf{a}}'_{k,r}$, $\forall r = 1, \dots, R$.

IV. JOINT UNMIXING-DECONVOLUTION UNDER NON-NEGATIVITY CONSTRAINTS

A. Non-negative JUD

Enforcing the non-negativity of abundances is physically sound in hyperspectral image unmixing. In addition, this constraint also enjoys a stabilizing property [10] which explains its benefit in image restoration. We present here a non-negative version of JUD (NN-JUD) based on the block Tikhonov criterion (11). Consider the non-negative block Tikhonov problem:

$$\min_{\mathbf{a}'_k \geq \mathbf{0}} \mathcal{J}_{\text{BT}}(\mathbf{a}'_k) \quad (14)$$

This problem is solved using the quadratic penalty method [11]. It consists in introducing slack variables $\mathbf{z} \geq \mathbf{0}$ and replacing the inequality constraints by equality constraints $\mathbf{a}'_k - \mathbf{z} = \mathbf{0}$. Combined to the augmented Lagrangian method, this transformation leads to the non-negative block Tikhonov (NNBT) problem:

$$\begin{aligned} \min_{\mathbf{a}'_k, \mathbf{z} \geq \mathbf{0}} \mathcal{J}_{\text{NNBT}}(\mathbf{a}_k, \mathbf{z}) &= \frac{1}{2} \left\| \mathbf{y}'_k - (\mathbf{S} \otimes \mathbf{H}_{(Q)}^1) \mathbf{a}'_k \right\|_2^2 \\ &+ \frac{\eta_a}{2} \left\| (\mathbf{I}_R \otimes \mathbf{\Delta}_{(Q)}) \mathbf{a}'_k \right\|_2^2 + \frac{\xi}{2} \left\| \mathbf{a}'_k - \mathbf{z} + \mathbf{u} \right\|_2^2 - \frac{\xi}{2} \left\| \mathbf{u} \right\|_2^2 \end{aligned} \quad (15)$$

where ξ is an increasing barrier and \mathbf{u} is the scaled Lagrange multiplier. The solution is obtained by iterating over i the following calculations for $i = 1, \dots, N_{\text{iter}}$:

$$\hat{\mathbf{a}}_k^{i+1} = \left(\mathbf{S}^\top \mathbf{S} \otimes \mathbf{H}_{(Q)}^{1\top} \mathbf{H}_{(Q)}^1 + \mathbf{I}_R \otimes \eta_a \mathbf{\Delta}_{(Q)}^\top \mathbf{\Delta}_{(Q)} + \xi^i \mathbf{I}_{NQ} \right)^{-1} \left((\mathbf{S}^\top \otimes \mathbf{H}_{(Q)}^{1\top}) \mathbf{y}'_k + \xi^i (\mathbf{z}^i - \mathbf{u}^i) \right) \quad (16)$$

$$\mathbf{z}^{i+1} = \max(\mathbf{0}, \hat{\mathbf{a}}_k^{i+1} + \mathbf{u}^i) \quad (17)$$

$$\mathbf{u}^{i+1} = \mathbf{u}^i + \hat{\mathbf{a}}_k^{i+1} - \mathbf{z}^{i+1} \quad (18)$$

$$\xi^{i+1} = \beta \xi^i, \beta \geq 1. \quad (19)$$

with $\mathbf{z}^0 = \mathbf{0}$, $\mathbf{u}^0 = \mathbf{0}$ and $\xi^0 > 0$.

B. Efficient implementation of the NN-JUD

In this section, we address the problem of implementing efficiently the NN-JUD. First we note that $\hat{\mathbf{a}}_k^{i+1}$ is solution of the following linear equation:

$$\left(\mathbf{S}^\top \mathbf{S} \otimes \mathbf{H}_{(Q)}^{1\top} \mathbf{H}_{(Q)}^1 + \mathbf{I}_R \otimes (\eta_a \mathbf{\Delta}_{(Q)}^\top \mathbf{\Delta}_{(Q)} + \xi^i \mathbf{I}_{NQ}) \right) \mathbf{a}'_k = \left((\mathbf{S}^\top \otimes \mathbf{H}_{(Q)}^{1\top}) \mathbf{y}'_k + \xi^i (\mathbf{z}^i - \mathbf{u}^i) \right) \quad (20)$$

which can be recognized as the Kronecker form of a Sylvester-type equation¹ [12]. The eigenvalue decomposition of the symmetric matrix $\mathbf{S}^\top \mathbf{S}$ reads $\mathbf{S}^\top \mathbf{S} = \mathbf{U} \mathbf{\Sigma}^2 \mathbf{U}^\top$ where $\mathbf{\Sigma}^2$ is a diagonal matrix whose entries are the (real and nonnegative) eigenvalues of $\mathbf{S}^\top \mathbf{S}$. As $\mathbf{H}_{(Q)}^1$ and $\mathbf{\Delta}_{(Q)}$ are circulant block circulant matrices, so are $\mathbf{H}_{(Q)}^{1\top} \mathbf{H}_{(Q)}^1$ and $\eta_a \mathbf{\Delta}_{(Q)}^\top \mathbf{\Delta}_{(Q)} + \xi^i \mathbf{I}_{NQ}$. They are diagonalizable in the Fourier basis which means that $\mathbf{H}_{(Q)}^{1\top} \mathbf{H}_{(Q)}^1 = \mathbf{F}^* \mathbf{\Gamma}^2 \mathbf{F}$ and $\eta_a \mathbf{\Delta}_{(Q)}^\top \mathbf{\Delta}_{(Q)} + \xi^i \mathbf{I}_{NQ} =$

¹The Sylvester matrix equation $\mathbf{A} \mathbf{X} \mathbf{B}^\top + \mathbf{C} \mathbf{X} \mathbf{D}^\top = \mathbf{E}$, where $\mathbf{A}, \mathbf{B}, \mathbf{C}$, and \mathbf{D} are square matrices of proper dimensions, may be transformed using the vectorization operator to $(\mathbf{B} \otimes \mathbf{A} + \mathbf{D} \otimes \mathbf{C}) \text{vec}(\mathbf{X}) = \text{vec}(\mathbf{E})$.

$\mathbf{F}^* \mathbf{\Phi}_i^2 \mathbf{F}$ where matrix \mathbf{F} denotes the 2D discrete Fourier transform matrix and $(\cdot)^*$ stands for the conjugate transpose. Inserting these expressions in (16) yields the following linear equation:

$$\begin{aligned} (\mathbf{U} \otimes \mathbf{F}^*) (\mathbf{\Sigma}^2 \otimes \mathbf{\Gamma}^2 + \mathbf{I}_R \otimes \mathbf{\Phi}_i^2) (\mathbf{U}^* \otimes \mathbf{F}) \mathbf{a}'_k = \\ \left((\mathbf{S}^\top \otimes \mathbf{H}_{(Q)}^{1\top}) \mathbf{y}'_k + \xi^i (\mathbf{z}^i - \mathbf{u}^i) \right) \end{aligned} \quad (21)$$

whose solution is given by:

$$\hat{\mathbf{a}}_k^{i+1} = (\mathbf{U} \otimes \mathbf{F}^*) (\mathbf{\Sigma}^2 \otimes \mathbf{\Gamma}^2 + \mathbf{I}_R \otimes \mathbf{\Phi}_i^2)^{-1} (\mathbf{U}^* \otimes \mathbf{F}) \left((\mathbf{S}^\top \otimes \mathbf{H}_{(Q)}^{1\top}) \mathbf{y}'_k + \xi^i (\mathbf{z}^i - \mathbf{u}^i) \right) \quad (22)$$

To have a fast implementation, all the necessary matrices are pre-calculated and stored in memory. The cost of the eigendecomposition of $\mathbf{S}^\top \mathbf{S}$ is $\mathcal{O}(R^3)$, and the cost of computing Fourier decompositions of $\mathbf{H}_{(Q)}^{1\top} \mathbf{H}_{(Q)}^1$ and $\eta_a \mathbf{\Delta}_{(Q)}^\top \mathbf{\Delta}_{(Q)} + \xi^i \mathbf{I}_{NQ}$ is $\mathcal{O}(N^2 Q^2 (\log_2 NQ))$ each. The calculation of $(\mathbf{S}^\top \otimes \mathbf{H}_{(Q)}^{1\top}) \mathbf{y}'_k$ is $\mathcal{O}(NQ R (\log_2 NQ + P))$ which just has to be made once. Thus, assuming $R \ll NQ$, the pre-calculation cost is dominated by the cost of the Fourier transforms which is $\mathcal{O}(2N^2 Q^2 (\log_2 NQ))$. The computation of the product of $(\mathbf{U} \otimes \mathbf{F}^*)$ (or $(\mathbf{U}^* \otimes \mathbf{F})$) by a vector of size RNQ is $\mathcal{O}(NQ R (\log_2 NQ + R))$ which dominates the cost of the inversion of the diagonal matrix $\mathbf{\Sigma}^2 \otimes \mathbf{\Gamma}^2 + \mathbf{I}_R \otimes \mathbf{\Phi}_i^2$ which is $\mathcal{O}(2NQ + R)$. Thus, at each iteration, the complexity of estimating \mathbf{a}'_k is $\mathcal{O}(2NQ R (\log_2 NQ + R))$. Table I summarizes the computational cost of online JUD algorithms.

TABLE I
APPROXIMATE COMPUTATIONAL COST PER SLICE OF JUD AND NN-JUD

JUD	Pre-calculations	$\mathcal{O}(2N^2 Q^2 (\log_2 NQ) + R^3)$
	Core algorithm	$\mathcal{O}(2NQ R (R + \log_2 NQ))$
NN-JUD	Pre-calculations	$\mathcal{O}(2N_{\text{iter}} N^2 Q^2 (\log_2 NQ) + R^3)$
	Core algorithm	$\mathcal{O}(2N_{\text{iter}} NQ R (R + \log_2 NQ))$

V. EXPERIMENTAL RESULTS

A. Simulated hyperspectral image

We first generate the unblurred and noisy-free image according to the mixture model (3). In this experiment, an instantaneous mixture of 3 sources is considered. The abundance maps are of size 101×101 while the endmembers, which include 32 spectral bands, correspond to NIR spectra of wood samples. To simulate an observed image, the original image is blurred by a Gaussian kernel of size 7×7 with full width at half-maximum (FWHM) set to 3 pixel. A Gaussian noise was added to reach a 5 dB signal-to-noise ratio (SNR). Figure 1 shows abundances estimated by JUD with and without non-negative constraints. The results are compared with those achieved by separated unmixing and deconvolution (SUD) method [8] which consists in performing successively the unmixing and the deconvolution of the abundance maps. Here, the regularization parameter η_a was set to be 5 and the size of the block Q was set to be 7. The number of iterations of the quadratic penalty method N_{iter} was fixed to 10. The initial

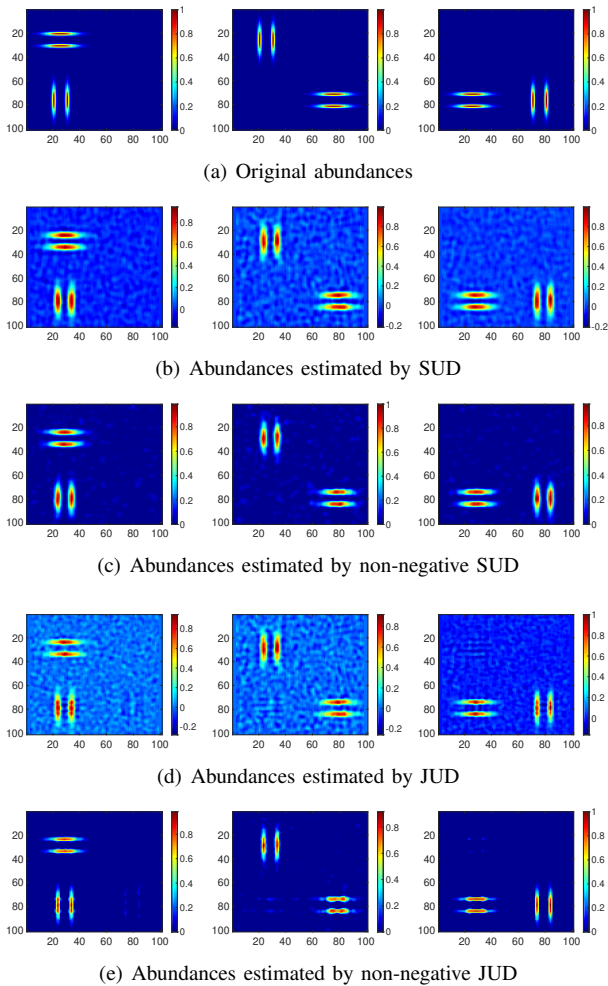


Fig. 1. Online joint unmixing-deconvolution results

value of penalty factor ξ was set to be 1 and $\beta = 10$. Results obtained with non-negativity constraints (Figures 1(e) and 1(c)) exhibit lower noise level. Moreover, one can observe a better deconvolution performance with NN-JUD as compared to NN-SUD.

Figure 2 shows the MSEs of estimated abundances by JUD and SUD with and without non-negative constraints as functions of SNR. Since SUD is less sensitive to the correlation of spectral sources, it yields better performance than JUD for small SNR in the unconstrained case. The non-negativity constraint allows to increase the performance especially for JUD. To conclude, non-negative JUD method should be considered for the online deconvolution and supervised unmixing problem.

B. Application to wood waste sorting

The last experiment aims at validating the performance of the non-negative JUD algorithm on real blurred hyperspectral images of size $211 \times 471 \times 32$ (spatial, time and spectral sizes respectively). The image contains pieces of wood waste of three types: raw wood, plywood and medium density fiber (MDF). Spectra of each type of wood were learned on similar

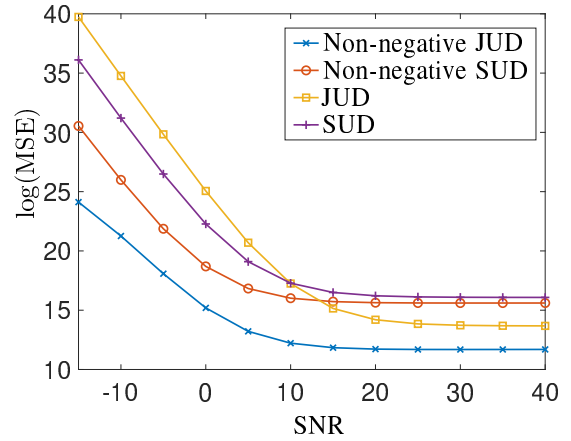


Fig. 2. MSEs of abundances estimated by different unmixing and deconvolution methods as functions of SNR

pieces of wood wastes. The spectral response of the conveyor was estimated from data in an area of size $121 \times 91 \times 32$ and was subtracted from each pixel of the hyperspectral image. By doing so, the spectral response of the conveyor can be assimilated to a zero-mean Gaussian noise. The convolution filter was estimated to be a Gaussian kernel of size 5×5 with FWHM of 2 points. The design parameters were set to: $\eta_a = 1$, $Q = 5$, $N_{iter} = 10$. The initial value of penalty factor ξ was set to 1 and $\beta = 10$. Figure 3(a) shows the spatial image corresponding to one wavelength. Abundances estimated by the online joint unmixing-deconvolution algorithm are shown in Figure 3(b)-3(d). Results show that the JUD has good performance to estimate different abundances. We can observe an improvement of spatial resolution in particular around the plywood object (abundance 2). A simple thresholding of the abundance will allow to classify the different types of wood wastes. It is worth to be mentioned that a careful inspection of the top left piece of raw wood revealed that it was partly covered by some residual glue, which is one of the main components of the plywood. This explains why the right part of the raw wood is shown on the second abundance (Figure 3(c)) which corresponds to the plywood spectral signature.

VI. CONCLUSION

In this paper, we addressed the joint unmixing-deconvolution problem for hyperspectral image processing. The proposed JUD criterion includes an ℓ_2 norm regularizer promoting the spatial smoothness of the abundance maps. It is extended to the online case by using a sliding-block criterion. A non-negativity constraint is also added to improve the unmixing-deconvolution performance. The implementation of NN-JUD algorithm was carried out by using the fast Fourier transform yielding lower computational cost. Experimental results on simulated images showed that NN-JUD method has better performance than unconstrained JUD. An application of the proposed non-negative JUD on a real hyperspectral image is also provided to confirm the performance. Future works will focus on the development of fast LMS-based recursive

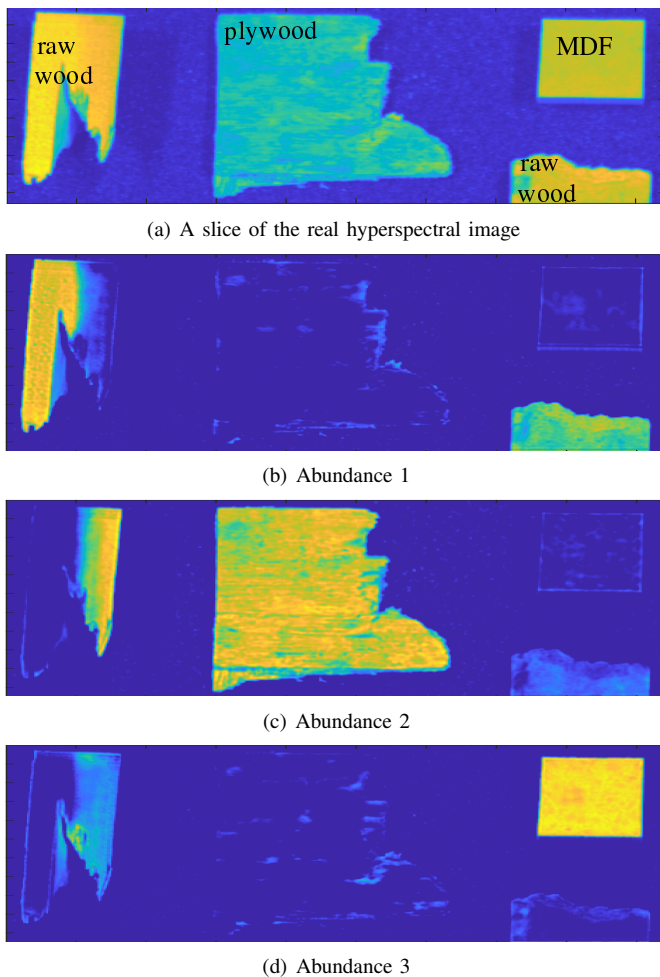


Fig. 3. Results of the online non-negative JUD on a real hyperspectral image of wood wastes.

estimators containing ℓ_1 -norm regularizers for industrial imaging systems. The integration of the proposed JUD in a wood waste classification software is currently under study.

REFERENCES

- [1] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, and J. Chanussot, "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE journal of selected topics in applied earth observations and remote sensing*, vol. 5, no. 2, pp. 354–379, 2012.
- [2] E. Chouzenoux, M. Legendre, S. Moussaoui, and J. Idier, "Fast constrained least squares spectral unmixing using primal-dual interior-point optimization," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 1, pp. 59–69, 2014.
- [3] N. Dobigeon, J.-Y. Tourneret, C. Richard, J. C. M. Bermudez, S. McLaughlin, and A. O. Hero, "Nonlinear unmixing of hyperspectral images: Models and algorithms," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 82–94, 2014.
- [4] A. Zare and K. Ho, "Endmember variability in hyperspectral analysis: Addressing spectral variability during spectral unmixing," *IEEE Signal Processing Magazine*, vol. 31, no. 1, pp. 95–104, 2014.
- [5] L. Drumetz, M.-A. Veganzones, S. Henrot, R. Phlypo, J. Chanussot, and C. Jutten, "Blind hyperspectral unmixing using an extended linear mixing model to address spectral variability," *IEEE Transactions on Image Processing*, vol. 25, no. 8, pp. 3890–3905, 2016.

- [6] X.-L. Zhao, F. Wang, T.-Z. Huang, M. K. Ng, and R. J. Plemmons, "Deblurring and sparse unmixing for hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 7, pp. 4045–4058, 2013.
- [7] S. Henrot, C. Soussen, M. Dossot, and D. Brie, "Does deblurring improve geometrical hyperspectral unmixing?" *IEEE Transactions on Image Processing*, vol. 23, no. 3, pp. 1169–1180, 2014.
- [8] S. Henrot, C. Soussen, and D. Brie, "Sequential deconvolution-unmixing of blurred hyperspectral data," in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 5152–5156.
- [9] Y. Song, E.-H. Djermoune, J. Chen, C. Richard, and D. Brie, "On-line deconvolution for industrial hyperspectral imaging systems," *SIAM Journal on Imaging Sciences*, vol. 12, no. 1, pp. 54–86, 2019.
- [10] J. M. Bardsley, J. K. Merikoski, and R. Vio, "The stabilizing properties of nonnegativity constraints in least-squares image reconstruction," *International Journal of Pure and Applied Mathematics*, vol. 43, no. 1, p. 95, 2008.
- [11] J. Nocedal and S. Wright, "Numerical optimization, series in operations research and financial engineering," *Springer, New York, USA*, 2006.
- [12] J. D. Gardiner, A. J. Laub, J. J. Amato, and M. C. B., "Solution of the Sylvester matrix equation $AXB^T + CXD^T = E$," *ACM Transactions on Mathematical Software*, vol. 18, no. 2, pp. 223–231, 1992.