

Modèle non-linéaire et semi-paramétrique pour le démixage non-supervisé d'images hyperspectrales

Rita AMMANOUIL, André FERRARI, Cédric RICHARD

Laboratoire Lagrange, Université de Nice Sophia-Antipolis, France

rita.ammannouilt@unice.fr, andre.ferrari@unice.fr, cedric.richard@unice.fr

Résumé – Cette communication propose un algorithme non-supervisé pour le démixage non-linéaire de données hyperspectrales. Plus précisément, un spectre est considéré comme étant la somme d'une combinaison linéaire de spectres purs et d'une fonction non linéaire des spectres voisins. L'estimation de la partie non linéaire est non paramétrique et suppose que la fonction non linéaire, à valeurs vectorielles, appartient à un espace de Hilbert à noyau reproduisant. Les spectres purs et leur nombre ne sont pas connus au préalable. Le problème d'optimisation convexe associé est résolu en utilisant l'algorithme des directions alternées (ADMM).

Abstract – This communication proposes an unsupervised algorithm for the nonlinear unmixing of hyperspectral images. A pixel's spectra is described as the sum of a linear combination of the endmembers plus a non linear function of its neighbours. The estimation of the nonlinear contribution is nonparametric and assumes that the nonlinear, vector valued, function belongs to a reproducing kernel Hilbert space with a predefined kernel function. The endmembers and their number are assumed to be unknown. The related convex optimization problem is solved using the Alternating Direction Method of Multipliers (ADMM) algorithm.

1 Introduction

En raison de la faible résolution spatiale des imageurs hyperspectraux, la surface représentée par un pixel peut contenir plusieurs matériaux. La réflexion de la lumière à la surface de chaque matériau crée un spectre *pur* qui contribue au spectre estimé pour ce pixel. Le modèle le plus répandu décrivant ce mélange spectral est le modèle linéaire [1]. Ce dernier suppose que le spectre mesuré est une combinaison linéaire des spectres purs.

Plusieurs modèles ont été développés pour prendre en compte des combinaisons spectrales plus complexes. Par exemple, les modèles bilinéaires [2, 3, 4] ajoutent une somme pondérée des produits des spectres purs pris « deux à deux ». Ou encore, le modèle proposé dans [5] modélise la contribution non linéaire de façon plus générale en utilisant une fonction non paramétrique appartenant à un espace de Hilbert à noyau reproduisant (RKHS). Pour un aperçu sur les différents modèles non linéaires, le lecteur est appelé à lire [6]. De façon analogue à [5], cette communication propose de modéliser la contribution non linéaire en utilisant une fonction non paramétrique appartenant à un RKHS. La différence principale avec [5] et les modèles bilinéaires cités ci-dessus est que la non linéarité est ici une fonction (vectorielle) des spectres voisins et non des spectres purs. Le choix d'incorporer dans le modèle une fonction non linéaire des spectres voisins est justifiée par exemple dans [7] par *l'effet d'adjacence* : le spectre d'un pixel est une fonction de la réflexion de la lumière sur la surface correspondant à ce pixel (partie linéaire) mais aussi de la déviation de la lumière qui a subi une réflexion sur les surfaces adjacentes

(partie non linéaire). Contrairement aux approches pixel-par-pixel [2, 3, 4, 5], l'introduction de cette dépendance spatiale nécessite de traiter toute ou partie de l'image lors de l'estimation de la non linéarité. Une autre différence est l'aspect non supervisé de la méthode proposée. Nous supposons en effet que les spectres purs et leur nombre sont inconnus, mais qu'ils sont présents dans la scène. Une régularisation favorisant la parcimonie des groupes [8] est alors utilisée pour identifier les spectres purs parmi les observations [9].

La section 2 décrit le nouveau modèle de mélange proposé. Après une brève introduction du modèle linéaire dans le cas supervisé classique, 2.1 s'intéresse à la partie non linéaire. Cette section décrit en particulier les RKHS de fonctions à valeurs vectorielles développés dans [10] et montre la capacité du modèle proposé à généraliser des modèles paramétriques de la littérature. La section 3.1 montre comment l'estimation de la partie linéaire peut être adaptée au cas non-supervisé. La section 3 développe l'algorithme d'optimisation du type ADMM permettant d'identifier les spectres purs, d'estimer les abondances des spectres purs pour les autres spectres et d'estimer la fonction non linéaire. La section 4 illustre les performances de la méthode sur des données synthétiques.

2 Modèle de mélange non linéaire

Dans le modèle linéaire, le spectre d'un pixel est une combinaison convexe des spectres purs pondérés par les abondances :

$$\mathbf{s}_n = \sum_{i=1}^M a_{i,n} \mathbf{r}_i + \mathbf{e}_n, \quad \forall n = 1, \dots, N, \quad (1)$$

avec $a_{j,n} \geq 0$ et $\sum_{j=1}^M a_{j,n} = 1$, où \mathbf{s}_n est le spectre du n -ème pixel de dimension L , L est le nombre de bandes de fréquences,

M est le nombre de spectres purs, \mathbf{r}_i est le i -ème spectre pur de dimension L , $a_{i,n}$ est l'abondance de \mathbf{r}_i dans \mathbf{s}_n , \mathbf{e}_n représente du bruit blanc Gaussian, N est le nombre d'observations. Soit \mathbf{A} la matrice des abondances de dimension $M \times N$, la colonne n de \mathbf{A} est notée \mathbf{a}_n , et la i -ème composante de \mathbf{a}_n est notée $a_{i,n}$. Tous les vecteurs sont des vecteurs colonnes.

2.1 Description de la non linéarité

Le modèle (1) a été largement étudié vu sa simplicité. Cependant, il existe de nombreux cas où le modèle linéaire est insuffisant, ce qui explique le développement récent de modèles plus généraux aptes à prendre en compte des mélanges plus complexes [6]. Cette communication propose le modèle suivant :

$$\mathbf{s}_n = \sum_{i=1}^M a_{i,n} \mathbf{r}_i + \mathbf{f}(\mathbf{v}_n) + \mathbf{e}_n, \quad (2)$$

avec $\mathbf{v}_n = [(\mathbf{v}_1^n)^\top, \dots, (\mathbf{v}_c^n)^\top]^\top \in \mathbb{R}^{cL}$ le vecteur contenant les c voisins du pixel n , et \mathbf{f} une fonction non linéaire tel que,

$$\begin{aligned} \mathbf{f} : \mathbb{R}^{cL} &\rightarrow \mathbb{R}^L \\ \mathbf{v}_n &\rightarrow \mathbf{f}(\mathbf{v}_n). \end{aligned} \quad (3)$$

Nous supposons que \mathbf{f} appartient à un RKHS \mathcal{H}_k de fonctions à valeurs vectorielles [10] associé au noyau suivant

$$\begin{aligned} \mathbf{k} : \mathbb{R}^{cL} \times \mathbb{R}^{cL} &\rightarrow \mathbb{R}^{L \times L} \\ (\mathbf{v}_n, \mathbf{v}_m) &\rightarrow \mathbf{k}(\mathbf{v}_n, \mathbf{v}_m). \end{aligned} \quad (4)$$

La matrice de Gram résultante, $\mathbf{K} = \{\mathbf{k}(\mathbf{v}_n, \mathbf{v}_m)\}_{n,m=1}^N$, est une matrice de $N \times N$ blocs tel que chaque bloc est de dimension $L \times L$. En d'autres termes, \mathbf{K} est une matrice de $LN \times LN$ scalaires. Le modèle adopté dans (2) retient la combinaison linéaire des spectres purs, et ajoute une fonction non linéaire et non paramétrique $\mathbf{f}(\mathbf{v}_n)$ agissant sur les voisins de \mathbf{s}_n . Alors que les modèles de la littérature font intervenir une même non linéarité intra-bandes, $\mathbf{f}(\mathbf{v}_n)$, où \mathbf{v}_n contient les c voisins de \mathbf{s}_n , permet de prendre en compte selon le type de noyau des relations inter-bandes et/ou inter-voisins.

Soulignons que la non linéarité $\mathbf{f}(\mathbf{v}_n)$ n'agit pas directement sur les corps purs, comme dans le cas des modèles bilinéaires [6], mais le fait indirectement au travers des \mathbf{v}_n . $\mathbf{f}(\mathbf{v}_n)$ permet ainsi de modéliser l'effet d'adjacence selon lequel la lumière réfléchiée sur une surface voisine au pixel n peut dévier et contribuer à son spectre [7].

Selon le choix de $\mathbf{k}(\cdot, \cdot)$ il est possible d'imposer plusieurs formes de non linéarité. Considérons par exemple

$$\mathbf{k}(\mathbf{v}_n, \mathbf{v}_m) = \text{diag}(\{\mathbf{k}(\psi_i(\mathbf{v}_n), \psi_i(\mathbf{v}_m))\}_{i=1}^L), \quad (5)$$

avec

$$\mathbf{k}(\psi_i(\mathbf{v}_n), \psi_i(\mathbf{v}_m)) = (\psi_i(\mathbf{v}_n)^\top \psi_i(\mathbf{v}_m))^2,$$

et $\psi_i(\mathbf{v}_n) = [v_{1,i}^n, \dots, v_{c,i}^n]^\top$, le vecteur contenant la i -ème composante spectrale de chaque voisin. Ce noyau induit l'espace des polynômes d'ordre deux « par bande », ce qui correspondra à des produits de spectres voisins pris « deux à deux ». La fonction \mathbf{f} correspond ainsi à un modèle bilinéaire sur les spectres voisins, ce qui rapproche (2) du modèle linéaire quadratique proposé dans [4] qui considère une non linéarité de la

forme $\sum_{i=1}^M \sum_{j=1}^M \beta_{i,j,n} \mathbf{r}_i \circ \mathbf{r}_j$, où \circ désigne le produit d'Hadamard. De façon plus générale, l'utilisation d'un noyau diagonal du type (5) se rapproche du modèle non linéaire non paramétrique proposé dans [5].

2.2 Identification des spectres purs

Nous avons supposé jusqu'à présent que dans (1) et (2) les spectres purs et leur nombre sont connus. Dans ce qui suit, nous nous placerons dans le contexte non supervisé proposé dans [9] où ces derniers ne sont pas connus. Toutefois, nous supposons qu'ils sont présents parmi les observations. Dans la mesure où le niveau du bruit est faible, on peut alors approcher (2) par le modèle suivant

$$\mathbf{s}_n = \sum_{i=1}^N x_{i,n} \mathbf{s}_i + \mathbf{f}(\mathbf{v}_n) + \mathbf{e}_n. \quad (6)$$

Soit \mathbf{X} la nouvelle matrice des abondances, avec $X_{i,n} = x_{i,n}$ l'abondance de \mathbf{s}_i dans \mathbf{s}_n . Si \mathbf{s}_i est un spectre pur, la i -ème ligne de \mathbf{X} , notée $\mathbf{x}_{\lambda_i} = [x_{i,1}, \dots, x_{i,N}]^\top$, contient les valeurs des abondances qui lui correspondent. Si \mathbf{s}_i n'est pas un spectre pur, \mathbf{x}_{λ_i} est une ligne nulle. Ainsi, une estimation de \mathbf{X} favorisant la présence de lignes nulles permettra d'identifier les spectres purs parmi les \mathbf{s}_i . L'utilisation de cette approche afin d'identifier les spectres purs a été validée pour le cas linéaire dans [9, 8].

3 Algorithme d'estimation

D'après la section 2, l'estimation de (\mathbf{X}, \mathbf{f}) correspond à l'attache aux données (6), avec des contraintes de positivité et de somme à un sur les colonnes de \mathbf{X} . L'estimation devant promouvoir des lignes nulles dans \mathbf{X} nous proposons de façon analogue à [9] d'utiliser une régularisation sur \mathbf{X} du type groupe Lasso. Le problème d'optimisation s'écrit donc :

$$\begin{aligned} \min_{\mathbf{f} \in \mathcal{H}_k, \mathbf{X}} & \frac{1}{2} \sum_{n=1}^N \|\mathbf{s}_n - \sum_{i=1}^N x_{i,n} \mathbf{s}_i - \mathbf{f}(\mathbf{v}_n)\|^2 \\ & + \frac{\lambda}{2} \|\mathbf{f}\|_{\mathcal{H}_k}^2 + \mu \sum_{i=1}^N \|\mathbf{x}_{\lambda_i}\|_2 \\ \text{subject à} & x_{i,n} \succeq 0 \quad \forall i, n = 1, \dots, N, \\ & \sum_{i=1}^N x_{i,n} = 1 \quad \forall n = 1, \dots, N, \end{aligned} \quad (7)$$

où les deux paramètres libres λ et μ contrôlent le degré de régularité de \mathbf{f} et la parcimonie sur les lignes de \mathbf{X} respectivement. Compte tenu de la non différentiabilité du critère (7), nous proposons d'utiliser l'algorithme des directions alternées (ADMM) [11] pour résoudre ce problème. Pour cela une nouvelle variable \mathbf{Z} est introduite dans (7) tout en ajoutant une contrainte pour assurer le consensus entre \mathbf{Z} et \mathbf{X} :

$$\begin{aligned} \min_{\mathbf{f} \in \mathcal{H}_k, \mathbf{X}, \mathbf{Z}} & \frac{1}{2} \sum_{n=1}^N \|\mathbf{s}_n - \sum_{i=1}^N x_{i,n} \mathbf{s}_i - \mathbf{f}(\mathbf{v}_n)\|^2 \\ & + \frac{\lambda}{2} \|\mathbf{f}\|_{\mathcal{H}_k}^2 + \mu \sum_{i=1}^N \|\mathbf{z}_{\lambda_i}\|_2 + \mathcal{I}_{\mathbb{R}^+}(\mathbf{Z}) \\ \text{subject à} & \mathbf{A}\mathbf{X} + \mathbf{B}\mathbf{Z} = \mathbf{C}, \end{aligned} \quad (8)$$

avec

$$\mathbf{A} = \begin{pmatrix} \mathbf{I} \\ \mathbf{1}^\top \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} -\mathbf{I} \\ \mathbf{0}^\top \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} \mathbf{0} \\ \mathbf{1}^\top \end{pmatrix}.$$

Le Lagrangien augmenté de (8) est alors :

$$\begin{aligned} \mathcal{L}_\rho(\mathbf{f}, \mathbf{X}, \mathbf{Z}, \Lambda) = & \frac{1}{2} \sum_{n=1}^N \|\mathbf{s}_n - \sum_{i=1}^N x_{i,n} \mathbf{s}_i - \mathbf{f}(\mathbf{v}_n)\|^2 \\ & + \frac{\lambda}{2} \|\mathbf{f}\|_{\mathcal{H}_k}^2 + \mathcal{I}_{\mathbb{R}^+}(\mathbf{Z}) + \mu \sum_{i=1}^N \|\mathbf{z}_{\lambda_i}\|_2 + \text{tr}(\Lambda^\top (\mathbf{A}\mathbf{X} \\ & + \mathbf{B}\mathbf{Z} - \mathbf{C})) + \frac{\rho}{2} \|\mathbf{A}\mathbf{X} + \mathbf{B}\mathbf{Z} - \mathbf{C}\|_F^2 \end{aligned} \quad (9)$$

Les différentes étapes d'une itération de l'algorithme ADMM sont développées ci-dessous. Chacune de ces étapes utilise les valeurs les plus récentes des autres variables.

3.1 Minimisation par rapport à \mathbf{f}

Cette étape consiste à minimiser le Lagrangien augmenté par rapport à la fonction \mathbf{f} . Cela revient à résoudre :

$$\min_{\mathbf{f} \in \mathcal{H}_k} \frac{1}{2} \sum_{n=1}^N \|\mathbf{s}_n - \sum_{i=1}^N x_{i,n} \mathbf{s}_i - \mathbf{f}(\mathbf{v}_n)\|^2 + \frac{\lambda}{2} \|\mathbf{f}\|_{\mathcal{H}_k}^2.$$

La fonction \mathbf{f} optimale est obtenue en annulant la dérivée de la fonction objectif calculée en utilisant les propriétés des RKHS :

$$-\sum_{n=1}^N \mathbf{k}(\cdot, \mathbf{v}_n) (\mathbf{s}_n - \sum_{i=1}^N x_{i,n} \mathbf{s}_i - \mathbf{f}(\mathbf{v}_n)) + \lambda \mathbf{f} = 0, \quad (10)$$

ce qui équivaut à

$$\mathbf{f} = \sum_{n=1}^N \mathbf{k}(\cdot, \mathbf{v}_n) \xi_n, \quad (11)$$

avec

$$\xi_n = \frac{1}{\lambda} (\mathbf{s}_n - \sum_{i=1}^N x_{i,n} \mathbf{s}_i - \mathbf{f}(\mathbf{v}_n)). \quad (12)$$

On déduit des équations (11) et (12) que les coefficients de \mathbf{f} sont la solution unique des équations linéaires :

$$\sum_{n=1}^N \mathbf{k}(\mathbf{v}_m, \mathbf{v}_n) \xi_n + \lambda \xi_m = \mathbf{s}_m - \sum_{i=1}^N x_{i,m} \mathbf{s}_i, \quad \forall m = 1, \dots, N.$$

Ces équations correspondent au système linéaire suivant :

$$(\mathbf{K} + \lambda \mathbf{I}) \boldsymbol{\zeta} = \mathbf{d}, \quad (13)$$

avec $\mathbf{K} = \{\mathbf{k}(\mathbf{v}_n, \mathbf{v}_m)\}_{n,m=1}^N$, $\boldsymbol{\zeta} = [\xi_1^\top, \dots, \xi_N^\top]^\top$ et $\mathbf{d} = \text{vec}(\mathbf{S} - \mathbf{S}\mathbf{X})$ où l'opérateur $\text{vec}(\cdot)$ empile les colonnes d'une matrice les unes au dessus des autres.

3.2 Minimisation par rapport à \mathbf{X}

Cette étape consiste à minimiser le Lagrangien augmenté par rapport à \mathbf{X} . Le problème est quadratique en \mathbf{X} , et se ramène à résoudre le système linéaire suivant :

$$(\mathbf{S}^\top \mathbf{S} + \rho \mathbf{A}^\top \mathbf{A}) \mathbf{X} = \mathbf{S}^\top (\mathbf{S} - \mathbf{F}) - \mathbf{A}^\top \Lambda - \rho \mathbf{A}^\top (\mathbf{B}\mathbf{Z} - \mathbf{C}).$$

3.3 Minimisation par rapport à \mathbf{Z}

La minimisation du Lagrangien augmenté par rapport à \mathbf{Z} est séparable par rapport à chaque ligne dans \mathbf{Z} . Cette étape se ramène à résoudre N problèmes de la forme

$$\min_{\mathbf{z}} \frac{1}{2} \|\mathbf{z} - \mathbf{v}\|_2^2 + \alpha \|\mathbf{z}\|_2 + \mathcal{I}_{\mathbb{R}^+}(\mathbf{z}) \quad (14)$$

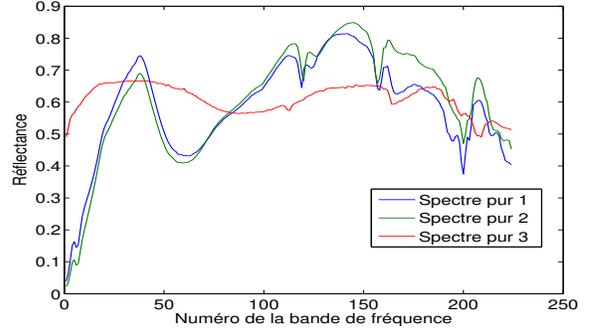


FIGURE 1 – Réflectance des spectres purs utilisés pour générer les données synthétiques.

avec $\mathbf{v} = \mathbf{x} + \rho^{-1} \boldsymbol{\lambda}$, $\alpha = \rho^{-1} \mu$. Les vecteurs $\boldsymbol{\lambda}$, \mathbf{x} and \mathbf{z} correspondent à une ligne dans Λ , \mathbf{X} et \mathbf{Z} , respectivement. Le minimiseur de (14) admet une solution unique donnée par l'opérateur proximal de la fonction $\Psi(\mathbf{z}) = \alpha \|\mathbf{z}\|_2 + \mathcal{I}_{\mathbb{R}^+}(\mathbf{z})$:

$$\begin{cases} \mathbf{z}^* = \mathbf{0} & \text{si } \|(\mathbf{v})_+\|_2 < \alpha \\ \mathbf{z}^* = \left(1 - \frac{\alpha}{\|(\mathbf{v})_+\|_2}\right) (\mathbf{v})_+ & \text{sinon} \end{cases} \quad (15)$$

avec $(\cdot)_+ = \max(\mathbf{0}, \cdot)$.

3.4 Mise à jour des multiplicateurs de Lagrange

La dernière étape d'une itération de l'algorithme ADMM consiste à mettre à jour les multiplicateurs de Lagrange selon l'équation suivante :

$$\Lambda^{k+1} = \Lambda^k + \rho(\mathbf{A}\mathbf{X} + \mathbf{B}\mathbf{Z} - \mathbf{C}). \quad (16)$$

Les étapes sont répétées jusqu'à ce qu'un critère d'arrêt sur le résidu primal ou dual soit satisfait.

4 Résultats expérimentaux

La performance de l'algorithme a été testée avec des données synthétiques. Pour cela nous utilisons 3 spectres réels avec $L = 224$, voir figure 1, pour générer 97 spectres mélangés linéairement. Les spectres purs sont ensuite ajoutés aux données, ce qui fait au total $N = 100$. Une contribution non linéaire selon l'expression $\gamma \mathbf{s}_{n-1} \circ \mathbf{s}_{n+1}$, avec $\gamma = 0.05$, est ensuite ajoutée à 50% des pixels. Du bruit blanc Gaussien est enfin ajouté aux données avec un rapport signal sur bruit de 50 dB.

Les figures 2 (a) et (b) montrent les coordonnées des données ainsi créées dans l'espace engendré par les 3 et 2 premières bandes obtenues par une analyse en composantes principales (ACP) des \mathbf{s}_n . On peut clairement observer que les pixels auxquels une non linéarité a été ajoutée sont « poussés » en dehors du simplexe qui est ici un triangle ($M = 3$). Trois algorithmes permettant d'identifier les spectres purs sont appliqués aux données ainsi générées : NFINDR [12], l'algorithme GLUP (Group Lasso With Unit Sum And Positivity

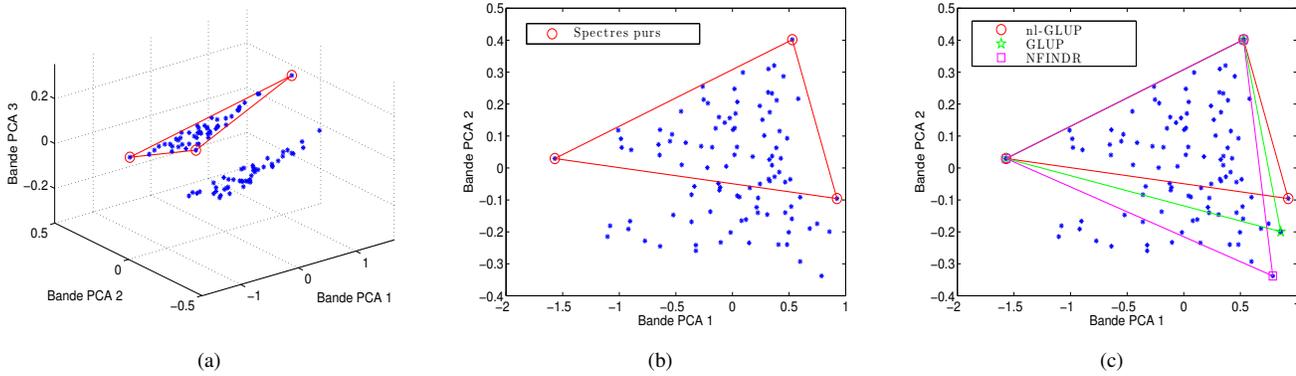


FIGURE 2 – (a) Données synthétiques : ACP de dimension 3 (b). Données synthétiques : ACP de dimension 2. (c) Spectres purs détectés par les 3 méthodes.

TABLE 1 – Temps d’exécution de l’algorithme et racine carrée de l’erreur quadratique moyenne entre \mathbf{X} et $\hat{\mathbf{X}}$.

	nl-GLUP	GLUP	NFINDR
Temps (sec)	10.01	1.61	0.1212
Erreur	0.0328	0.0679	-

Constraints) [9] et l’algorithme proposé dénomé nl-GLUP. Les deux derniers algorithmes nécessitent de repérer les lignes non nulles dans \mathbf{X} correspondants aux spectres purs. Pour cela, la moyenne de chaque ligne est comparée à un seuil. Contrairement aux deux autres algorithmes, NFINDR nécessite de fixer le nombre de spectres purs, ici 3. Les valeurs suivantes sont utilisées pour les paramètres de nl-GLUP : $\mu = 1$, $\lambda = 0.1$, $\rho = 0.05$, et un nombre d’itération maximal de 250. GLUP correspond à nl-GLUP avec $\lambda = 0$ et $\mu = 2$ pour ce cas. Le noyau choisi pour nl-GLUP est Gaussien, du type :

$$k(\psi_i(\mathbf{v}_n), \psi_i(\mathbf{v}_m)) = \exp(-\sigma^{-1} \|\psi_i(\mathbf{v}_n) - \psi_i(\mathbf{v}_m)\|^2),$$

avec $\sigma = 3$. Vue la structure diagonale de chaque bloc $\mathbf{k}(\mathbf{v}_n, \mathbf{v}_m)$ dans \mathbf{K} , (13) peut être divisé en L systèmes de N variables chacun.

Les spectres purs détectés par chaque algorithme sont reportés dans la Figure 2 (c). Contrairement à GLUP et NFINDR qui n’ont pu détecter que 2 parmi les 3 spectres purs, l’approche proposée a correctement indentifié les 3 spectres. Le tableau 1 donne le temps d’exécution de chacun des algorithmes. L’étape la plus exigeante de nl-GLUP en temps de calcul est la résolution de (13). Enfin le tableau reporte la racine carrée de l’erreur quadratique moyenne entre \mathbf{X} et $\hat{\mathbf{X}}$ pour nl-GLUP et GLUP qui est clairement en faveur de nl-GLUP.

5 Perspectives

Cette communication propose un nouveau modèle pour le démixage non supervisé et non linéaire des données hyperspectrales. Les performances ont été testées sur des données synthétiques. Toutefois, l’expérimentation sur des données réelles présentera de nouveaux enjeux compte tenu de la taille du problème et du haut niveau de corrélation entre les spectres.

Références

- [1] N. Keshava and J. F. Mustard, “Spectral unmixing,” *IEEE TRANS. SIG. PROC.*, vol. 19, no. 1, pp. 44–57, 2002.
- [2] J. Nascimento and J. Bioucas-Dias, “Nonlinear mixture model for hyperspectral unmixing,” in *SPIE*, 2009, pp. 74770I–74770I.
- [3] Y. Altmann, A. Halimi, N. Dobigeon, and J-Y Tourneret, “Supervised nonlinear spectral unmixing using a postnonlinear mixing model for hyperspectral imagery,” *IEEE TRANS. IP*, vol. 21, no. 6, pp. 3017–3025, 2012.
- [4] I. Meganem, P. Deliot, X. Briottet, Y. Deville, and S. Hosseini, “Linear–quadratic mixing model for reflectances in urban environments,” *IEEE TGRS*, vol. 52, no. 1, pp. 544–558, 2014.
- [5] J. Chen, C. Richard, and P. Honeine, “Nonlinear estimation of material abundances in hyperspectral images with l1-norm spatial regularization,” *IEEE TGRS*, vol. 52, no. 5, pp. 2654 – 2665, 2014.
- [6] N. Dobigeon, J.-Y. Tourneret, C. Richard, J. C. Bermudez, S. McLaughlin, and A. Hero, “Nonlinear unmixing of hyperspectral images : Models and algorithms,” *IEEE SP Magazine*, vol. 31, no. 1, pp. 82–94, 2014.
- [7] D. Burazerovic, R. Heylen, B. Geens, S. Sterckx, and P. Scheunders, “Detecting the adjacency effect in hyperspectral imagery with spectral unmixing techniques,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 6, no. 3, pp. 1070–1078, 2013.
- [8] D. Iordache, J. Bioucas-Dias, and A. Plaza, “Collaborative sparse regression for hyperspectral unmixing,” *IEEE TGRS*, vol. 52, no. 1, pp. 341–354, February 2013.
- [9] R. Ammanouil, A. Ferrari, C. Richard, and D. Mary, “Blind and fully constrained unmixing of hyperspectral images,” *IEEE TRANS. IP*, vol. 23, no. 12, pp. 5510 – 5518, October 2014.
- [10] C. Micchelli and M. Pontil, “On learning vector-valued functions,” *Neural Computation*, vol. 17, no. 1, pp. 177–204, 2005.
- [11] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [12] M. E. Winter, “N-FINDR : an algorithm for fast autonomous spectral endmember determination in hyperspectral data,” in *Proc. SPIE Imaging Spectrometry*, October 1999.