Contributions to unsupervised and nonlinear unmixing of hyperspectral data

Rita Ammanouil

UNIVERSITE DE NICE-SOPHIA ANTIPOLIS - UFR Sciences

Ecole Doctorale Sciences Fondamentales et Appliquées

THESE

pour obtenir le titre de
Docteur en Sciences
de l'UNIVERSITE de Nice-Sophia Antipolis

Discipline : Sciences de l'Ingénieur

présentée et soutenue par
Rita AMMANOUIL

Contributions au démélange non-supervisé et non linéaire de données hyperspectrales

Contributions to unsupervised and nonlinear unmixing of hyperspectral data

Thèse dirigée par André FERRARI et Cédric RICHARD
soutenue le 13 octobre 2016

Jury :

Rapporteurs :	José BIOUCAS-DIAS	Professeur Associé	-	Université de Lisbonne, Portugal
	Jocelyn CHANUSSOT	Professeur	-	INP de Grenoble, France
Examinateurs :	Sandrine MATHIEU	Chef de projet	-	Thales Alenia Space, France
	Steve MCLAUGHLIN	Professeur	-	Université de Heriot-Watt, Edinburgh
	Paul SCHEUNDERS	Professeur	-	Université de Antwerp, Belgique
	Jean-Yves TOURNERET	Professeur	-	INP de Toulouse, France
Directeurs :	André FERRARI	Professeur	-	Université de Nice Sophia Antipolis, France
	Cedric RICHARD	Professeur	-	Université de Nice Sophia Antipolis, France
Invité:	Paul HONEINE	Professeur	-	Université de Rouen Normandie, France

Abstract

Spectral unmixing has been an active field of research since the earliest days of hyperspectral remote sensing. It is concerned with the case where various materials are found in the spatial extent of a pixel, resulting in a spectrum that is a mixture of the signatures of those materials. Unmixing then reduces to estimating the pure spectral signatures and their corresponding proportions in every pixel. In the hyperspectral unmixing jargon, the pure signatures are known as the endmembers and their proportions as the abundances. This thesis focuses on spectral unmixing of remotely sensed hyperspectral data. In particular, it is aimed at improving the accuracy of the extraction of compositional information from hyperspectral data. This is done through the development of new unmixing techniques in two main contexts, namely in the unsupervised and nonlinear case. In particular, we propose a new technique for blind unmixing, we incorporate spatial information in (linear and nonlinear) unmixing, and we finally propose a new nonlinear mixing model. More precisely, first, an unsupervised unmixing approach based on collaborative sparse regularization is proposed where the library of endmembers candidates is built from the observations themselves. This approach is then extended in order to take into account the presence of noise among the endmembers candidates. Second, within the unsupervised unmixing framework, two graph-based regularizations are used in order to incorporate prior local and nonlocal contextual information. Next, within a supervised nonlinear unmixing framework, a new nonlinear mixing model based on vector-valued functions in reproducing kernel Hilbert space (RKHS) is proposed. The aforementioned model allows to consider different nonlinear functions at different bands, regularize the discrepancies between these functions, and account for neighboring nonlinear contributions. Finally, the vector-valued kernel framework is used in order to promote spatial smoothness of the nonlinear part in a kernel-based nonlinear mixing model. Simulations on synthetic and real data show the effectiveness of all the proposed techniques.

Keywords: Hyperspectral data, unsupervised unmixing, nonlinear unmixing, alternating direction method of multipliers (ADMM), group lasso regularization, Laplacian regularization, vector valued reproducing kernel Hilbert space (RKHS).

Résumé

Le démélange spectral est l'un des problèmes centraux pour l'exploitation des images hyperspectrales. En raison de la faible résolution spatiale des imageurs hyperspectraux en télédetection, la surface représentée par un pixel peut contenir plusieurs matériaux. Dans ce contexte, le démélange consiste à estimer les spectres purs (les endmembers) ainsi que leurs fractions (les abondances) pour chaque pixel de l'image. Le but de cette thèse est de proposer de nouveaux algorithmes de démélange qui visent à améliorer l'estimation des spectres purs et des abondances. En particulier, les algorithmes de démélange proposés s'inscrivent dans le cadre du démélange non-supervisé et non-linéaire. Dans un premier temps, on propose un algorithme de démelange non-supervisé dans lequel une régularisation favorisant la parcimonie des groupes est utilisée pour identifier les spectres purs parmi les observations. Une extension de ce premier algorithme permet de prendre en compte la présence du bruit parmi les observations choisies comme étant les plus pures. Dans un second temps, les connaissances a priori des ressemblances entre les spectres à l'échelle locale et non-locale ainsi que leurs positions dans l'image sont exploitées pour construire un graphe adapté à l'image. Ce graphe est ensuite incorporé dans le problème de démélange non-supervisé par le biais d'une régularisation basée sur le Laplacian du graphe. Enfin, deux algorithmes de démélange non-linéaires sont proposés dans le cas supervisé. Les modèles de mélanges non-linéaires correspondants incorporent des fonctions à valeurs vectorielles appartenant à un espace de Hilbert à noyaux reproduisants. L'intérêt de ces fonctions par rapport aux fonctions à valeurs scalaires est qu'elles permettent d'incorporer un a priori sur la ressemblance entre les différentes fonctions. En particulier, un a priori spectral, dans un premier temps, et un a priori spatial, dans un second temps, sont incorporés pour améliorer la caractérisation du mélange non-linéaire. La validation expérimentale des modèles et des algorithmes proposés sur des données synthétiques et réelles montre une amélioration des performances par rapport aux méthodes de l'état de l'art. Cette amélioration se traduit par une meilleure erreur de reconstruction des données.

Mots clés: Données hyperspectrales, démélange non-supervisé, démélange non-linéaire, algorithme des directions altérnées (ADMM), régularisation de type group lasso, régularisation avec le Laplacian, espace de Hilbert à noyau reproduisant (RKHS).

Dedicated to my beloved family
Pierre, Elige, Rémie, Rami
and to my partner Jean

Acknowledgments

First and foremost my sincerest gratitude goes to my supervisors, André Ferrari and Cédric Richard. I am grateful for their support, their patience, and most importantly for their guidance throughout the last three years. Without their wisdom, and their professionalism this thesis would not have been successfully completed. In particular, a special thanks goes to André for always trusting me.

Second, I would like to thank the members of my thesis committee. I thank José Bioucas-Dias and Jocelyn Chanussot who kindly accepted to review my thesis manuscript and provided me with their comments and suggestions. I also thank Sandrine Mathieu, Steve Mclaughlin, Paul Scheunders, Jean-Yves Tourneret, and Paul Honeine who accepted to be part of my thesis committee and to come all the way to Nice.

I would like to express my gratitude to Paul Honeine and Joumana Farah who were the first to introduce me to the exciting path of research during my engineering final year project. I particularly thank Paul for his support and advice whenever it was needed during the last three years. I would also like to thank Sandrine Mathieu, David Mary, and Jean-Yves Tourneret for their contributions in my research work. In particular, I would like to thank Sandrine Mathieu for kindly providing me with an interesting Meris data set.

I thank all the members of the Lagrange laboratory, Rémy, Anthony, khalil, Céline, Claude, and Henri for their kindness. A special thanks to Raja, Wei and Jie whom I was lucky to meet during the beginning of this journey. My gratitude extends to Delphine "ma Française préférée" for her friendship and for her help. The same gratitude goes to my friend Reine for her fun and pleasant presence. My deepest thank you goes to the greatest friend I have ever had, Roula. Words cannot describe all the adventures we have been through during the last three years. Thank you for simply being here from day one.

Last but not least, my deepest appreciation goes to my family members, and especially my father Pierre and my beautiful mother Elige. Without their support and their prayers, I would not have had the strength to complete this work. Finally, a heartfelt thank you to my beloved partner Jean who despite the long distance, was always by my side. I am lucky for having him by my side, always believing in me and making sure I never give up on my dreams ...

Rita

Contents

List of Figures

List of Tables

Acronyms and Notations

Acronyms

NASA national aeronautics and space administration

ADMM alternating direction method of multipliers

RKHS reproducing kernel Hilbert space

AVIRIS airborne visible/infrared imaging spectrometer

VIS visible

NIR near infrared

SWIR shortwave infrared

CHRIS compact high resolution imaging spectrometer

VD virtual dimensionality

FCLS fully constrained least squares

ML maximum likelihood

JPL jet propulsion laboratory

NMF non negative matrix factorization

CSS column subset selection

SSA single scattering albedo

TV total variation

RMSE root mean square error

MSA maximum spectral angle

ASA average spectral angle

SNR signal-to-noise ratio

MiSTO multidimensional shrinkage thresholding operator

GLUP group lasso, unit sum, and positivity

NGLUP reduced noise GLUP

VCA virtual component analysis

PCA principal component analysis

NDU nonlinear neighbor and band dependent unmixing

CG conjugate gradient

CLC corine land cover

Notations: scalars, vectors, and matrices

a, A	Lowercase and uppercase lightface letters denote scalars
$\boldsymbol{a}$	Lowercase boldface letters denote column vectors
$\boldsymbol{A}$	Uppercase boldface letters denote matrices
A_{ij}	(i,j)-th entry of $\boldsymbol{A}$
a_{ij}	(i,j)-th entry of $\boldsymbol{A}$
$\boldsymbol{a}_i$	Column vector representing the i-th column in $\boldsymbol{A}$
$\boldsymbol{a}_{\lambda_i}$	Column vector representing the i-th row in $\boldsymbol{A}$
$\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N]$	Matrix with columns $\boldsymbol{a}_i$, $i = 1, \ldots, N$
$\boldsymbol{A} = [\boldsymbol{a}_{\lambda_1}, \ldots, \boldsymbol{a}_{\lambda_N}]^\top$	Matrix with rows $\boldsymbol{a}_{\lambda_i}^\top$, $i = 1, \ldots, N$
$\boldsymbol{A}_\omega = [\boldsymbol{a}_{\omega_1}, \ldots, \boldsymbol{a}_{\omega_{N'}}]$	Restriction of $\boldsymbol{A}$ to the columns indexed by ω

Special vectors and matrices

$\boldsymbol{1}_N$	$N \times 1$ unit vector (All entries are 1)
$\boldsymbol{0}_N$	$N \times 1$ null vector (All entries are 0)
$\boldsymbol{1}_{N \times N'}$	$N \times N'$ unit matrix (All entries are 1)
$\boldsymbol{0}_{N \times N'}$	$N \times N'$ null matrix (All entries are 0)
$\boldsymbol{I}_N$	$N \times N$ identity matrix

Some matrix operations

$\boldsymbol{A}^\top$	Transposed matrix of matrix $\boldsymbol{A}$
$\boldsymbol{A}^{-1}$	Inverse matrix of matrix $\boldsymbol{A}$
$\boldsymbol{A}^\dagger$	Pseudo inverse matrix of matrix $\boldsymbol{A}$
$\mathrm{trace}(\boldsymbol{A})$	Trace of matrix $\boldsymbol{A}$
$\mathrm{vec}(\boldsymbol{A})$	Vector version of $\boldsymbol{A}$ (see Sec.)
$\max(\boldsymbol{A}, \boldsymbol{B})$	Maximum values in $\boldsymbol{A}$ and $\boldsymbol{B}$ component-wise
$\lvert\boldsymbol{A}\rvert$	Determinant of $\boldsymbol{A}$
$\boldsymbol{A} \otimes \boldsymbol{B}$	Kronecker product of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$
$\boldsymbol{A} \odot \boldsymbol{B}$	Hadamard (element wise) product of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$
$\boldsymbol{A} \succeq \boldsymbol{0}$	Components of $\boldsymbol{A}$ are non-negative

Matrix norms

$\lVert\boldsymbol{a}\rVert_2$	ℓ_2-norm (or $\ell_{2,1}$-norm) of vector $\boldsymbol{a}$ ($\lVert\boldsymbol{a}\rVert_{\mathrm{F}} = \sqrt{\boldsymbol{a}^\top \boldsymbol{a}}$)
$\lVert\boldsymbol{a}\rVert_1$	ℓ_1-norm of vector $\boldsymbol{a}$ ($\lVert\boldsymbol{a}\rVert_1 = \sum_i \lvert a_i\rvert$)
$\lVert\boldsymbol{a}\rVert_\infty$	ℓ_∞-norm of vector $\boldsymbol{a}$ ($\lVert\boldsymbol{a}\rVert_{1,\infty} = \max_i \lvert a_i\rvert$)
$\lVert\boldsymbol{A}\rVert_{\mathrm{F}}$	Frobenius norm of matrix $\boldsymbol{A}$ ($\lVert\boldsymbol{A}\rVert_{\mathrm{F}} = \sqrt{\mathrm{trace}(\boldsymbol{A}^\top \boldsymbol{A})}$)
$\lVert\boldsymbol{A}\rVert_{1,\infty}$	$\ell_{1,\infty}$-norm of matrix $\boldsymbol{A}$ ($\lVert\boldsymbol{A}\rVert_{1,\infty} = \sum_i \max_j \lvert A_{i,j}\rvert$)

Spectral unmixing notations

N	Number of pixels in the image
L	Number of spectral bands
M	Number of endmembers
$\boldsymbol{S}$	$L \times N$ matrix of observation spectra
$\boldsymbol{R}$	$L \times M$ matrix of endmembers
$\boldsymbol{A}$	$M \times N$ matrix of abundances

Chapter 1

Motivations and organisation

1.1 Motivations

Hyperspectral imaging, also known as imaging spectroscopy, has emerged as one of the most important technologies in the realm of remote sensing. The premise behind hyperspectral imaging is that it provides images with thorough spectral information for each pixel, enabling the identification of various materials and the extraction of various physical parameters in remotely sensed scenes. It is therefore not surprising that hyperspectral imaging has vastly contributed to increasing our understanding of the world.

The earliest successes of hyperspectral imaging began due to its potential and unprecedented applications in remote sensing. Naturally, the development of hyperspectral sensors with hundreds of contiguous spectral channels was a significant step beyond multispectral sensors with only tens of broad spectral channels. Originally, hyperspectral images were valued due to their ability to identify a wide range of surface cover materials that could not be identified using images with lower spectral resolution such as multispectral images. For example, hyperspectral images have been successfully used to identify mineral components and map vegetation species [Goetz et al., 1985, Pieters and Mustard, 1988]. Furthermore, they have been used to study plant canopy [Peterson et al., 1988], measure water vapor in the atmosphere [Gao and Goetz, 1990], estimate chlorophyl content in water [Hamilton et al., 1993], analyze vegetation species [Clark and Swayze, 1995], and perform geologic mapping [Kruse, 1998]. The utility of hyperspectral imaging has been also proven for target detection in the military field, and for studying the composition of stars and planets in astronomy. Although hyperspectral imaging was originally developed for remote sensing applications, it has also spread into non-remote sensing applications such as food quality monitoring, forensic inspections, and disease diagnosis to cite a few. Nowadays,

the amount of hyperspectral data is continually increasing and becoming more available to the public. As a result, it can be expected that its applications to new fields will be soon explored.

Nevertheless, effective applications and future advancements in the hyperspectral imaging domain heavily rely on the development of appropriate signal processing techniques. Indeed, the various hyperspectral imaging applications require the development and adaptation of new or existing signal processing techniques to fully exploit the specificities of this data. Some of the well-known and established signal processing techniques in the area of hyperspectral imaging are dimensionality reduction, feature extraction, target detection, anomaly detection, classification, and unmixing to cite a few [Bioucas-Dias et al., 2012]. Among the previously cited techniques, unmixing has particularly witnessed growing attention and contributions over the past years. This is mainly due to the fact that unmixing allows to analyze the hyperspectral data with very high accuracy. Research dedicated to the unmixing topic has led to the development of blind unmixing techniques, and to the development of new paradigms for jointly exploiting the spatial and spectral dimensions of the data. Moreover, recent research trends in this topic have proposed nonlinear unmixing schemes in order to increase the interpretation and analysis accuracy.

This thesis focuses on unmixing and contributes to the aforementioned research trends in this topic. In particular, we propose a new technique for blind unmixing, we incorporate spatial information in (linear and nonlinear) unmixing, and we finally propose a new nonlinear mixing model. The thesis organization is detailed in the following section.

1.2 Thesis organisation

The thesis is composed of seven chapters as shown in Figure 1.1. The present chapter, chapter 1, gives an overview of the overall work presented in the thesis and its organization. Chapter 2 introduces the hyperspectral unmixing framework. The following four chapters, namely chapters 3 to 6, are the core of the thesis in the sense that they explain the proposed unmixing techniques. The organization of these chapters is detailed in what follows:

In the third Chapter, we introduce an original approach for unsupervised unmixing based on collaborative sparse regularization. This chapter addresses the problem of blind and fully constrained unmixing of hyperspectral images. Unmixing is performed without the use of any dictionary, and assumes that the number of constituent materials in the scene and their spectral signatures are unknown. Two models with increasing complexity are developed to achieve this challenging task, depending on how noise interacts with hyperspectral data. The first one leads

to a convex optimization problem, and is solved with the Alternating Direction Method of Multipliers (ADMM). The second one accounts for signal-dependent noise, and is addressed with a Reweighted Least Squares algorithm.

In the fourth chapter, we propose to incorporate spatial regularization within the unmixing formulation using a graph-based framework. More precisely, this chapter introduces two graph based regularizations in the linear and unsupervised unmixing framework. The proposed regularizations rely upon the construction of a graph representation of the hyperspectral image. In the first case, a quadratic Laplacian regularization is used to promote smoothness in the estimated abundance maps and collaborative estimation between homogeneous areas of the image. In the second case, a graph-based Total Variation regularization is used to promote piece-wise constant reconstructed spectra with respect to the graph structure. The resulting optimization problems are convex and solved using the Alternating Direction Method of Multipliers (ADMM). The performance of the proposed algorithms is demonstrated by comparing them to other well-known algorithms using both simulated and real hyperspectral data.

In the fifth chapter, we propose a new nonlinear mixing model that allows to incorporate spectral prior regarding the nonlinearities at different bands. This chapter presents a kernel based nonlinear mixing model for hyperspectral data where the nonlinear function belongs to a Hilbert space of vector valued functions and the endmembers are assumed to be known. The proposed model extends existing ones by accounting for band dependent and neighboring nonlinear contributions. The key idea is to work under the assumption that nonlinear contributions are dominant in some parts of the spectrum while they are less pronounced in other parts. In addition to this, we motivate the need for taking into account nonlinear contributions originating from the ground covers of neighboring pixels by practical considerations, precisely the adjacency effect. The proposed nonlinear function is associated to a matrix valued kernel that allows to jointly model a wide range of nonlinearities and include prior information regarding band dependencies. Furthermore, the choice of the nonlinear function input allows to incorporate neighboring effects. A particular class of kernels is investigated where the kernel's design relies upon the construction of a graph where each band represents a pixel. The optimization problem is strictly convex and the corresponding iterative algorithm is based on the alternating direction method of multipliers (ADMM). Finally, experiments conducted using synthetic and real data demonstrate the effectiveness of the proposed approach.

In the sixth chapter, we propose to incorporate a spatial regularization within the supervised kernel-based nonlinear unmixing formulation. Using tools from vector-valued functions in a

RKHS, we incorporate a regularization that promotes smooth spatial variations of the nonlinear part. The spatial regularizer and the nonlinear contributions are jointly modelled by a vector-valued function that lies in a reproducing kernel Hilbert space (RKHS). The design of the kernel relies upon the construction of a graph representation of the hyperspectral image where each node represents a pixel. The unmixing problem is strictly convex and reduces to a quadratic programming (QP) problem. Simulations on synthetic data illustrate the effectiveness of the proposed approach.

Finally, the seventh chapter concludes the thesis by summarizing the advantages and main contributions of the methods developed in the previous chapters. It also provides concluding remarks and an outlook on future research directions.

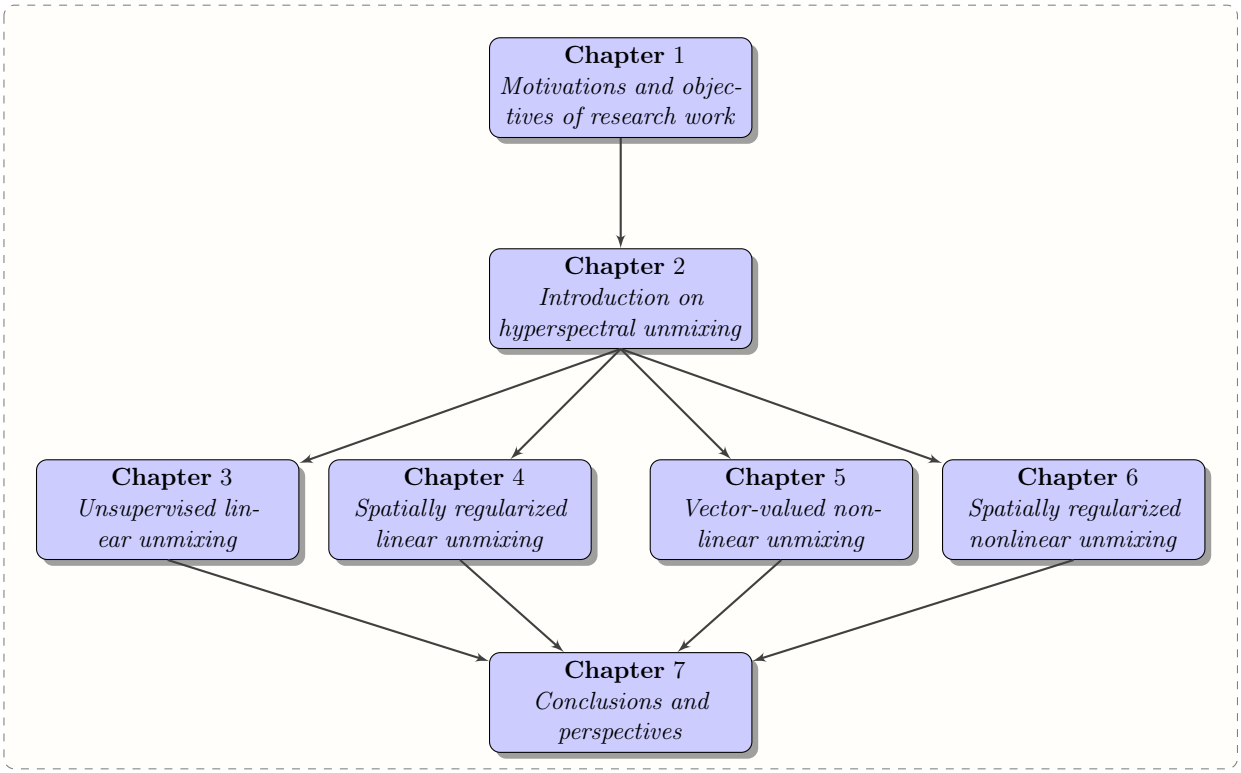


Figure 1.1: Thesis organisation

Chapter 2

Introduction on hyperspectral imaging

In this chapter, we begin with an introduction on hyperspectral imaging. Afterwards, the concept of mixed and pure spectra often encountered in remote sensing applications is explained. Linear and nonlinear spectral mixing models and unmixing techniques are reviewed. Finally, the importance of exploiting spatial and spectral prior information in the unmixing procedure is discussed.

2.1 Hyperspectral imaging concept

Hyperspectral imaging is defined as the simultaneous acquisition of images in hundreds of narrow and adjacent wavelength bands such that for each pixel in the image a reflectance spectrum can be derived [Goetz et al., 1985]. Hyperspectral images are produced using airborne or spaceborne hyperspectral sensors also known as imaging spectrometers.

2.1.1 Hyperspectral sensors

Hyperspectral sensors measure the solar radiation scattered from the surface with relatively high spectral and spatial resolutions. To provide perspective on the spectral and spatial characteristics of these sensors, consider the following sensor examples. The airborne visible/infrared imaging spectrometer (AVIRIS)[1] delivers hyperspectral images in 224 contiguous wavelength bands in the wavelength range $400 - 2500$ nm, and a spatial resolution of 20 m when flown onboard an airborne at approximately 20 km above ground level. Other examples of hyperspectral sensors

[1] http://aviris.jpl.nasa.gov/aviris

Table 2.1: Spatial and spectral characteristics of AVIRIS, CHRIS, and Hyperion.

	AVIRIS	CHRIS	Hyperion
wavelength range (nm)	$400 - 2450$	$415 - 1050$	$400 - 2500$
number of bands	224	63	220
spectral resolution (nm)	9.6	12	10
spatial resolution (m)	20	36	30
altitude (km)	20	600	705

are Hyperion[2] and compact high resolution imaging spectrometer (CHRIS)[3]. In contrast with AVIRIS, these sensors are flown onboard spacebornes. Table 2.1 gives the spectral and spatial characteristics of the three previously cited sensors. It can be noted that CHRIS covers the visible (VIS) and near infrared (NIR) portions of the electromagnetic spectrum, whereas AVIRIS and Hyperion also cover the shortwave infrared (SWIR) portion of the electromagnetic spectrum. The spectral resolution for the three sensors is around 10 nm, and the spatial resolution is between 20 and 36 m. In addition to these three currently operational sensors, several sensors are under construction and will be part of planned missions. The reader is referred to [Staenz et al., 2012] for a summary of current and future spaceborne hyperspectral sensors and missions initiatives planned in different countries around the world.

Historically, a great part of the developments that helped establish and demonstrate the hyperspectral technology emerged at the national aeronautics and space administration (NASA) jet propulsion laboratory (JPL) [Wendisch and Brenguier, 2013, chap. 2]. This was mainly due to the pioneering work of Alexander Goetz and his colleagues who developed imaging spectrometers as well as important image processing software and atmospheric correction algorithms [MacDonald et al., 2009]. In 1983, the driving force for this technology was the deployment of the Airborne imaging spectrometer (AIS) which provided the first hyperspectral images in 128 spectral bands covering the spectral range from 1200 to 2400 nm with a spectral resolution of 9.6 nm. The development and deployment of AIS mainly served as a technology demonstrator and a starting point for hyperspectral sensors design. Soon thereafter, in 1987, AIS was followed by the well-known hyperspectral sensor AVIRIS [Goetz, 1991, Kruse et al., 1999]. The considerations that influenced the design of AVIRIS with the characteristics as described in table 2.1 are reported in [Porter and Enmark, 1987]. As mentioned previously, AVIRIS is currently providing data for scientific use. Nevertheless, AVIRIS will be soon replaced by an AVIRIS - Next Generation (AVIRIS-NG)[4] providing data with higher quality compared to the classic AVIRIS.

[2] http://eo1.usgs.gov/sensors/hyperion
[3] https://earth.esa.int/web/guest/missions/esa-operational-eo-missions/proba
[4] http://aviris-ng.jpl.nasa.gov

2.1.2 Hyperspectral reflectance data

Hyperspectral imaging is sometimes referred to as "imaging spectroscopy", or "imaging spectrometry" as in [Goetz et al., 1985], and described as "spatial spectroscopy from afar" as in [Wendisch and Brenguier, 2013, chap. 2]. These nomenclatures highlight the fact that hyperspectral imaging endows optical imaging with spectroscopic measurements. Spectroscopy is the science concerned with the identification of materials based on their scattering properties [Hapke, 2012]. This science exploits the fact that light is scattered differently by materials depending on their molecular composition, scale and shape. Examining the reflectance values at sufficiently sampled portions of the electromagnetic spectrum or equivalently the overall shape of the reflectance spectrum allows to identify the composition of the object and infer various properties [Shaw and Burke, 2003]. Spectroscopy is concerned with field or laboratory measurements estimated using a spectrometer on a single pixel basis, whereas hyperspectral imaging is concerned with spectral measurements from afar and on an image basis. To sum up, hyperspectral imaging provides the possibility to combine the potential of spectroscopic analysis along with a spatial and contextual analysis in a single system.

The fundamental quantity used to describe hyperspectral images is the spectral reflectance corresponding to the viewed surface. The information related to the composition of the surface being viewed is revealed by analyzing this quantity across the covered spectral range [see Chang, 2007, chap. 2]. By definition, the reflectance is the ratio of incident light reflected by a surface at a specific wavelength band after a part of that light has been absorbed or emitted by the surface [Hapke, 2012]. Being a ratio, the reflectance is a number with no unit ranging from 0 to 1, i.e. when there is no reflection and when there is perfect reflection respectively. In hyperspectral imaging, the reflectance is simultaneously estimated for each pixel at hundreds of narrow and adjacent wavelength bands. The ordered set of reflectance values across the wavelength range is referred to as the reflectance spectrum or simply spectrum. The estimated reflectance values depend on the corresponding wavelength band and usually have smooth variations for a sufficiently small spectral resolution. As a result, the overall hyperspectral image reduces to a stack of spectrally contiguous images often represented as a cube. Figure 2.1 gives a schematic example of such a cube having two spatial dimensions and one spectral dimension. On the left is the hyperspectral data cube which consists of a stack of images at different wavelength bands, in the middle is the spectrum from one pixel in the image providing the reflectance values across the range of wavelength bands, and on the right is a plot of the pixel's spectrum showing the variation of the reflectance values as a function of the wavelength bands.

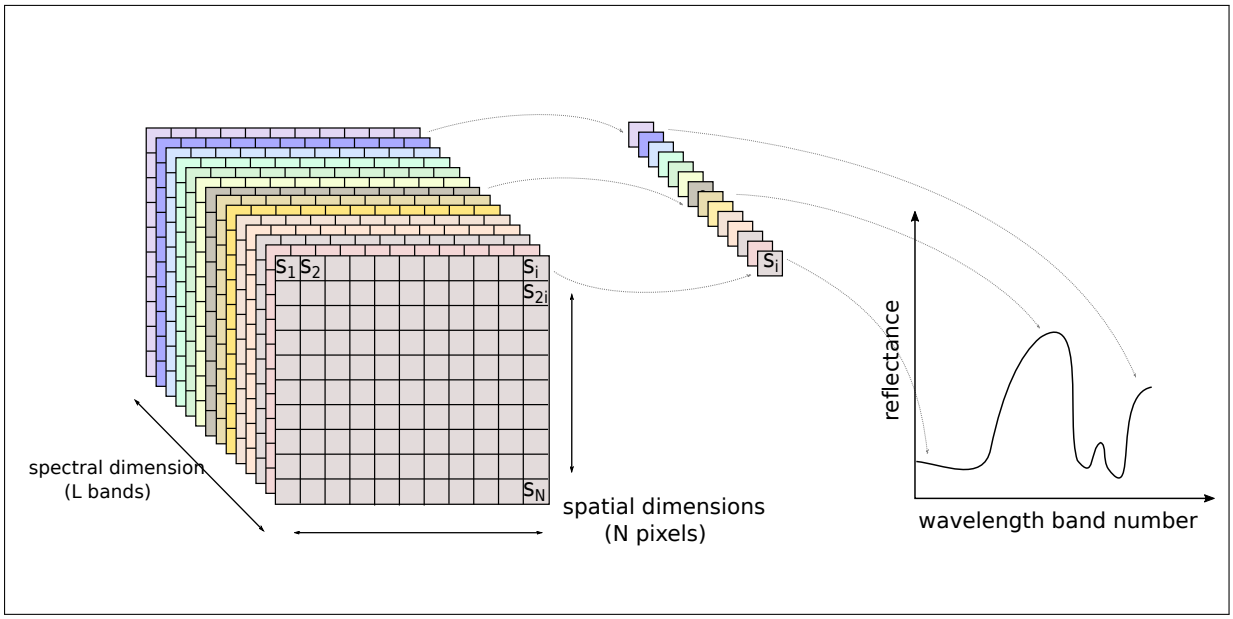


Figure 2.1: Hyperspectral imaging concept showing from left to right the hyperspectral image cube, the reflectance spectrum of a pixel, and the plot of the reflectance values as a function of wavelength.

Finally, note that the estimation of the surface reflectance from remotely sensed measurements is more challenging compared to field and laboratory measurements due to sensor noise, atmospheric effects, imaging artifacts, and illumination geometry [Liang et al., 2001, Richter and Schlapfer, 2005]. The estimation of the reflectance requires accurately removing all these effects. See for example [Gao et al., 2009] for a review of various atmospheric correction and calibration techniques. In this manuscript, all data sets used are in reflectance values and when the term spectrum is used it is meant the reflectance spectrum.

2.2 Spectral mixing models

Rather than assigning to each pixel a specific material or class, spectral unmixing acknowledges the fact that a pixel can have a mixed composition, hence a mixed spectrum. The mixing framework systematically distinguishes between pure and mixed spectra. A pure spectrum, usually referred to as an endmember, is associated with a specific material or class. It is ideally a unique spectral signature representative of the corresponding material. A pure spectrum is estimated for a pixel when light beams reaching the sensor have interacted with only one material. In contrast, a mixed spectrum is estimated for a pixel when light beams reaching the sensor have interacted with more than one material. As the name implies, a mixed spectrum is a mixture

of the pure spectra of the materials within the corresponding pixel. In general, there are two directions for approaching the spectral mixture problem: the linear and nonlinear models.

2.2.1 Linear mixing model

The simultaneous presence of distinct materials in one pixel is due to the fact that the sensor spatial resolution is not high enough to resolve each material in a distinct pixel. In general, the spatial resolution of an hyperspectral image in remote sensing can vary from a few meters up to a hundreds of meters. No matter the spatial resolution in this range, the spatial extent of a pixel is large enough to simultaneously contain different types of materials. According to the linear mixing model, a mixed spectrum is expressed as a weighted sum of the endmembers spectra. This assumption holds provided that the mixing scale is macroscopic, the pixel's surface consists of distinct materials spatially distributed side by side in the pixel's surface, and there is negligible interaction between them. Furthermore, the linear mixing assumption holds when incident light light beams are subject to only one reflection, i.e. each photon interacts with only one material before reaching the sensor. More formally, assume that the hyperspectral image is estimated at L wavelength bands, and contains in total N pixels indexed by $n = 1, \ldots, N$ as shown in Figure 2.1. According to the linear mixing model we have that:

$$\boldsymbol{s}_n \quad = \sum_{i=1}^{M} a_{in}\boldsymbol{r}_i + \boldsymbol{e}_n, \, \forall \, n = 1, \ldots N, \tag{2.1}$$

where $\boldsymbol{s}_n \in \mathbb{R}^L$ is the L-dimensional spectrum for the n-th pixel, M denotes the number of endmembers, $a_{i,n}$ is the abundance of the i-th endmember in the n-th pixel, $\boldsymbol{r}_i$ is the L-dimensional spectrum of the i-th endmember, $\boldsymbol{e}_n$ is a vector of Gaussian white noise accounting for sensor noise and error of the model. In all our notations, vectors are column vectors. The abundances, represent the contribution of each endmember in the mixed spectrum. More precisely, the abundance of an endmember is the surface fraction occupied by the endmember with respect to the pixel surface. Being surface fractions, further constraints are imposed on the abundances. They must be positive and sum to one:

$$\begin{cases} a_{in} \geq 0, \\ \sum_{i=1}^{M} a_{in} = 1. \end{cases} \tag{2.2}$$

The LMM is a simple yet very representative model, which was extensively studied in the literature, see for example the surveys [Bioucas-Dias et al., 2012, Keshava and Mustard, 2002]. This model was used since the earliest days of remote sensing for analysing multispectral images before its use naturally extended to hyperspectral images, see for example [Singer and McCord, 1979,

Adams et al., 1986, Shimabukuro and Smith, 1991]. Nevertheless, some scenes such as containing particulate materials, vegetated and urban areas exhibit strong nonlinear effects [Bioucas-Dias et al., 2012]. This has led to the refinement of the LMM through the development of nonlinear mixing models.

2.2.2 Nonlinear mixing models

The linear mixing model accurately describes the spectral mixture for the situations where the materials are in distinct patches within the pixel, light beams reaching the sensor undergo only one reflection, and hence each light beam interacts with a single material. When this is not the case, and either one of the two assumptions does not hold, the spectral mixture is nonlinear. More precisely, this corresponds to the situations where the materials are intimately mixed, or when light beams undergo multiple reflections before reaching the sensor. In both cases, the same photons interact with more than one material and the mixture is nonlinear. For example, intimate mixtures occur in sandy or particulate mixtures containing different materials, whereas multiple reflections occur in multi-layered scenes where multiple reflections are due to the 3 dimensionality of objects. Naturally, nonlinear models have a complex mathematical relationship between the endmembers and the abundances when compared with the linear mixing model.

In the case of intimate mixtures, the mixing model is derived by Hapke based on radiative transfer theory [Hapke, 2012]. It can be shown that the mixture is nonlinear when expressed in terms of reflectance values, but when expressed in terms of scattering albedo, the mixture becomes linear. However, it is relatively challenging and complex to unmix intimate mixtures based on Hapke's model. This is due to the fact that it is a nonlinear function of various parameters related to the scene that are usually not available or hard to obtain. Another alternative for modeling intimate mixtures is through the use of kernel functions. The authors of [Broadwater and Banerjee, 2009] and [Broadwater and Banerjee, 2010] proposed a model able to take into account intimate mixtures through the use of specifically designed kernels. The kernel based model is a generalization of the linear mixing model in the sense that when the kernel is set to a linear one the underlying model becomes linear.

In the case of multilayered scenes, multiple reflections lead to a different class of nonlinear mixing models known as bilinear models. The mathematical expression established for multiple reflections is the term by term product of two reflectance vectors in the case of two reflections (bilinear model), and more than two reflectance vectors in the case of multiple reflections (multilinear model) [Heylen and Scheunders, 2016]. In particular, bilinear models have received a lot

of attention and various models were developed in this context such as [Nascimento and Bioucas-Dias, 2009, Fan et al., 2009, Halimi et al., 2011, Altmann et al., 2012, Meganem et al., 2014] to cite a few. These models usually adopt the following generic formulation:

$$\boldsymbol{s}_n = \sum_{i=1}^{M} a_{in}\boldsymbol{r}_i + \beta_{ijn}\sum_{i=1}^{M}\sum_{j=1}^{M} a_{in}a_{jn}\boldsymbol{r}_i \odot \boldsymbol{r}_j + \boldsymbol{e}_n, \tag{2.3}$$

where β_{ijn}, the nonlinearity parameter, is in the range $[0\,1]$ and $\odot$ denotes the Hadamard (element wise) product. The constraints imposed on the abundances and the parameters β_{ijn} differ from one model to the other. On the right hand side of equation (2.3), the first term corresponds to the linear mixture and the second term corresponds to the nonlinear (bilinear) one.

Another way for modelling the nonlinear mixtures of the endmembers is through the use of nonlinear functions in reproducing kernel Hilbert spaces [Aronszajn, 1950, Shawe-Taylor and Cristianini, 2004]. For example, the authors of [Chen et al., 2013] propose the following model:

$$\boldsymbol{s}_n = \sum_{i=1}^{M} a_{in}\boldsymbol{r}_i + \boldsymbol{g}_n(\boldsymbol{R}) + \boldsymbol{e}_n, \tag{2.4}$$

where $\boldsymbol{g}_n(\boldsymbol{R}) = [g_n(\boldsymbol{r}_{\lambda_1}) \dots g_n(\boldsymbol{r}_{\lambda_L})]^\top$, $g_n(\cdot)$ is a nonlinear function in a reproducing kernel Hilbert space (RKHS), and $\boldsymbol{r}_{\lambda_i}$ denotes the i-th row in the endmembers matrix $\boldsymbol{R} = [\boldsymbol{r}_1, \dots, \boldsymbol{r}_M]$. The advantage of kernel-based models over models similar to (2.3) is that they are non parametric which means that they do not impose a predetermined form for the nonlinear term. In fact, depending on the kernel choice, the nonlinear function is associated with a nonlinear feature map that determines the nonlinear mapping of the endmembers. For instance, the authors of [Chen et al., 2013] show that model (5.3) is able to incorporate bilinear, multilinear as well as more complex nonlinear interactions between the endmembers for certain choices of the kernel, namely the second order polynomial and Gaussian kernels respectively.

In contrast with all the previously cited models, the authors of [Yokoya et al., 2014, Févotte and Dobigeon, 2015] do not impose any analytical form for the nonlinear term which is merely treated as a positive residual term. Nevertheless, this model-free approach can be limiting since it does not control the nonlinear expression, and it does lead to any physical interpretation. As a result, it can prevent accurate estimations of the nonlinear contribution.

2.3 Spectral unmixing

While mixing models provide the leverage for expressing the composition of mixed spectra in terms of endmembers and the abundances. Hyperspectral unmixing, or more generally spectral

unmixing, is a source separation problem aimed at estimating the abundances and the endmembers spectra given the mixed spectra and eventually the corresponding mixing model. Note that when the abundances and endmembers are jointly estimated, the unmixing is known as unsupervised. Whereas, when the abundances are estimated given that the endmembers have been previously determined, the unmixing is known as supervised. Several approaches and algorithms were proposed for supervised and unsupervised unmixing hyperspectral images in both, the linear and nonlinear case. These methods have been extensively reviewed in the literature, see for example the surveys [Keshava and Mustard, 2002, Bioucas-Dias et al., 2012] on linear unmixing, and the surveys [Heylen et al., 2014a, Dobigeon et al., 2014b] on nonlinear unmixing. Hereafter, we review some of the various approaches adopted in linear unmixing, then in nonlinear unmixing.

2.3.1 Linear unmixing

In general, supervised linear unmixing involves the following consecutive steps: determining the number of endmembers, extracting the spectral signature of the endmembers, and estimating their abundances for every pixel in the scene. Several algorithms have been proposed to perform each stage separately. For example, virtual dimensionality (VD) [Chang and Du, 2004], followed by N-FINDR [Winter, 1999] or VCA [Nascimento and Bioucas-Dias, 2005], followed by fully constrained least squares (FCLS) [Heinz and Chang, 2001] is among the most widely used processing chain for linear unmixing. Alternatively, unsupervised methods jointly perform the endmembers and abundances estimation. Some of the unsupervised methods assume that the number of endmembers is known, whereas some methods perform the joint endmembers and abundances estimation without even knowing a priori the endmembers number. Hereafter, we focus on unsupervised linear unmixing. In particular, we review geometrical, non negative matrix factorization (NMF), and sparse based techniques dedicated for this task.

A geometrical approach for joint estimation of the endmembers and the abundances was proposed in [Honeine and Richard, 2012]. The authors express the abundances in terms of a volume or distance ratio and propose to use these expressions within existing geometrical endmember extraction algorithms that already compute distances and volumes in their iterative procedure. For example, NFINDR [Winter, 1999] and VCA [Nascimento and Bioucas-Dias, 2005] compute volumes and distances respectively while searching for the endmembers. As a result, the abundance estimation can be easily applied and does not require additional cost. However, the geometrical approach proposed in [Honeine and Richard, 2012] does not take into account

the positivity constraint. More precisely, the algorithm can result in negative abundances when a mixed pixel is outside the simplex whose vertices are the selected endmembers.

While geometrical approaches require the presence of the endmembers among the observations, unsupervised algorithms using NMF [Lee and Seung, 2001] overcome this constraint by estimating the endmembers spectra rather than identifying them among the observations. The NMF decomposes the observations matrix into a product of two non negative matrices, the endmembers and the abundances matrix. The corresponding algorithm consists of iterating two update rules that alternately estimate the abundances and the endmembers. Most of the NMF based unmixing algorithms impose additional contraints. For example, the authors of [Yang et al., 2011] incorporate a sparseness measure into the optimization problem to promote sparse abundances. The authors of [Liu et al., 2011] incorporate two additional constraints aimed at promoting abundance separation, and smooth spectral variations throughout the image. Whereas, the authors of [Miao and Qi, 2007] impose a minimum volume constraint on the simplex whose vertices are the endmembers. The NMF is a nonconvex optimization problem, hence adding the constraints introduces prior information that can guide the convergence to a relatively more appropriate local minima. However, a concern when dealing with NMF is that the estimated endmembers may not correspond to real material signatures.

Another class of algorithms for unsupervised unmixing exploits sparse regression regularization. This approach usually requires having a large set of endmember candidates, such as a spectral library of known endmembers, and unmixing is performed by expressing the observations using a small number of the candidates. For instance, one of the well known sparse based unsupervised techniques is SUnSAL [Iordache et al., 2011] which consists of a constrained least squares optimization problem using the ℓ_1-norm regularization, to promote sparse abundances. A collaborative approach was developed in [Iordache et al., 2013] where the $\ell_{2,1}$-norm, also known as the Group lasso [Yuan and Lin, 2006], is used rather than the ℓ_1-norm in order to simultaneously set to zero all the abundances corresponding to the candidates that are not present in the scene. Nevertheless, a disadvantage of using spectral libraries is that there may calibration mismatches since the candidate endmembers spectra and the available observations are acquired in different conditions and using different sensors. To overcome this problem, the same concept can be used while assuming that the endmembers are present in the scene, thus the observations themselves are used in the spectral library. Following this idea, the collaborative sparse regression strategy in [Iordache et al., 2013] was used in [Iordache et al., 2014b] to extract the endmembers from the observations themselves. Similarly, the authors of [Esser et al., 2012]

use an $\ell_{1,\infty}$-norm instead of the $\ell_{2,1}$-norm regularization in order to extract the endmembers from the observations.

Finally, note that using a constrained least squares optimization problem with sparse regularization with the same purpose, namely finding among the observations (not specifically hyperspectral data) those few observations that can best approximate the other observations by linear combinations, have been investigated in other similar problems. For example, this is the case in column subset selection (CSS) [Boutsidis et al., 2009, Tropp, 2006] and sparse subspace clustering [Elhamifar and Vidal, 2009].

2.3.2 Nonlinear unmixing

In section 2.2.2, various nonlinear mixing models have been reviewed, namely, intimate mixtures, bilinear models, and some model-free models. Hereafter, we review nonlinear unmixing algorithm developed in the literature and dedicated for these models. Depending on wether the endmembers are known or not, the proposed unmixing algorithm is supervised or unsupervised respectively.

When dealing with intimate mixtures, Hapke's model can be used to perform linear unmixing after converting reflectance values to single scattering albedo (SSA) of the composing particles. This approach was adopted in [Mustard and Pieters, 1989] and [Nascimento and Bioucas-Dias, 2010]. In particular, the work in [Nascimento and Bioucas-Dias, 2010] proposed a two step unsupervised nonlinear mixing algorithm. The first step consists of converting reflectance data to SSA, afterwards an unsupervised linear unmixing algorithm is used, namely the simplex identification via split augmented Lagrangian (SISAL) [Bioucas-Dias, 2009]. Another alternative for unmixing intimate mixtures is based on kernel models. In this framework, the authors [Broadwater and Banerjee, 2009, 2010] simply propose to replace the product of endmembers in the cost function of the FCLS problem [Heinz and Chang, 2001] by a kernel function acting on the corresponding spectra. Nevertheless, unlike the previous algorithm, the endmembers are assumed to be known a priori.

In the case of bilinear models, several unmixing approaches were developed in the supervised case. For example, in [Nascimento and Bioucas-Dias, 2009], the endmembers matrix is augmented with virtual endmembers consisting of pairwise products of the actual endmembers and unmixing is performed using the FCLS algorithm [Heinz and Chang, 2001]. In [Fan et al., 2009] and [Halimi et al., 2011] a first order taylor approximation is used to transform the problem into a linear one and unmixing is also performed using FCLS. In contrast with the previously cited algorithms, the algorithm proposed in [Altmann et al., 2014] performs unsupervised unmixing in a Bayesian

framework.

In the case of the model-free mixing approaches several unmixing algorithms have been introduced in the literature. For the kernel based model developed in [Chen et al., 2013], the optimization problem reduces to solving a positively constrained quadratic programming problem. The authors of [Heylen et al., 2011b] and [Heylen et al., 2014b] propose a fully geometrical approach for unsupervised and nonlinear unmixing using non-euclidean distances. In [Heylen et al., 2011b], the authors rewrite NFINDR in terms of distance geometry and use geodesic distances rather than Euclidean distances. Whereas in Heylen et al. [2014b], the authors rewrite DMaxD Schott et al. [2003] in terms of distances rather than NFINDR, and various non Euclidean distances are proposed. Subsequently, the abundances are estimated using the distances computed in the previous step and the simplex projection algorithm developed in Heylen et al. [2011a]. Finally, the unsupervised nonlinear models developed in [Févotte and Dobigeon, 2015, Yokoya et al., 2014, Eches and Guillaume, 2014] are unmixed using NMF.

2.4 Incorporating spatial information into unmixing

Several spectral unmixing algorithms have been exposed in the previous section. However, all these algorithms do not exploit the contextual and spatial information present in hyperspectral images. Recently, several algorithms showed that incorporating spatial information and exploiting the complementarity between the spectral and spatial dimensions can improve the unmixing results. However, a drawback of this framework is that pixels are no longer treated individually and the whole image is processed simultaneously which results in an increased computational complexity. Hereafter, we give particular attention to some of the algorithms that incorporate spatial information in order to improve the abundances estimation step. See Shi and Wang [2014] for a review of algorithms that incorporate spatial information in the abundance estimation step in addition to the endmember extraction and selection of endmember combinations steps.

One of the most widely used tools for incorporating spatial information in image processing is the total variation (TV) regularization [Rudin et al., 1992]. TV is well known for recovering piecewise constant signals and for preserving discontinuities. Following this idea, the authors in [Iordache et al., 2012a] use a Total Variation (TV) regularization for the abundances on top of sparse ℓ_1-norm regularized linear unmixing. Similarly, the authors of [Chen et al., 2014] incorporate a TV regularization for the abundances in a nonlinear unmixing algorithm. In both works, the TV regularization promotes piece wise constant abundances estimates. This effect is

particularly interesting when the image contains large blocks of structured objects such as roads and buildings where it can be assumed that the pixels belonging to those structures have the same abundances (i.e. the abundances are constant). However, there are images where this not the case, i.e. the abundances have smooth variations rather than being constant such as in vegetation or soil areas. As a result, it is more appropriate to use the ℓ_2-norm rather than the ℓ_1-norm used in TV. Furthermore, TV regularization is based on a local spatial graph Strong and Chan [1996] in the sense that each pixel is only connected to its four spatial neighbors. As a result, TV does not directly capture long range similarities between pixels in distant structures in the image, and the graph structure is not adapted to the image itself. Following these ideas, several graph regularizations were introduced as a generalization of TV to random and nonlocal graphs. For example, the authors of [Lu et al., 2013] and [Tong et al., 2014] build nonlocal graphs adapted to the image and incorporate a quadratic graph Laplacian regularization within sparse unmixing. The corresponding optimization problems are solved using Non negative Matrix Factorization (NMF).

There are other approaches for incorporating spatial information within the unmixing procedure. For example, the authors of [Zare, 2011] use a regularization term based on Fuzzy local information that encourages the abundances in a pixel to be similar in value to a weighted sum of the abundances in neighboring pixels. In their work [Zare and Gader, 2011] also based on Fuzzy local information, they incorporate within the optimization problem a regularization term that measures the spatial complexity in the image. The aim of this regularization is to promote smooth abundance variations from one pixel to its neighbors. The authors of [Castrodad et al., 2011] incorporate a quadratic regularization term in the unmixing problem accounting for grouping and spectral coherence. Finally, the authors of [Eches et al., 2011] exploit Markov Random fields in order to incorporate spatial information in the unmixing procedure, then an hierarchical Bayesian model is adopted in order to solve the resulting unmixing problem.

Chapter 3

Blind & fully constrained linear unmixing

This chapter has been adapted from the journal paper [Ammanouil et al., 2014].

In this chapter we address the problem of blind and fully constrained unmixing. Unmixing is performed without the use of a spectral library of known materials, and assumes that the number of constituent materials in the scene and their spectral signatures are unknown. The estimated abundances are fully constrained, they satisfy the desired sum-to-one and non negativity constraints. Two models with increasing complexity are developed to achieve this challenging task, depending on how noise interacts with hyperspectral data. The first one leads to a convex optimization problem, and is solved with the alternating direction method of multipliers (ADMM). The second one accounts for signal-dependent noise, and is addressed with a Reweighted Least Squares algorithm. Experiments on synthetic and real data demonstrate the effectiveness of the proposed approach.

3.1 Introduction

Three consecutive tasks are usually required for unmixing: determining the number of endmembers, extracting the spectral signature of the endmembers, and estimating their abundances for every pixel in the scene. Several algorithms have been proposed to perform each stage separately. The pipeline virtual dimensionality (VD) [Chang and Du, 2004], followed by N-FINDR [Winter, 1999] and fully constrained least squares (FCLS) [Heinz and Chang, 2001] is among the most

Table 3.1: Notations for chapter 3

L	1×1	Number of spectral bands
N	1×1	Number of observations
M	1×1	Number of endmembers
N'	1×1	Number of candidate endmembers
$\boldsymbol{S}$	$L \times N$	Matrix of observed spectra
$\boldsymbol{R}$	$L \times M$	Endmembers matrix
$\boldsymbol{A}$	$M \times N$	Abundances (w.r.t. endmembers in $\boldsymbol{R}$)
$\boldsymbol{E}$	$L \times N$	Noise matrix
ω	$N' \times 1$	Subset of N' indexes in $\{1, \ldots, N\}$
$\widetilde{\boldsymbol{S}}$	$L \times N$	Noiseless matrix of observed spectra
$\widetilde{\boldsymbol{S}}_\omega$	$L \times N'$	Matrix of candidate endmembers from $\widetilde{\boldsymbol{S}}$
$\boldsymbol{S}_\omega$	$L \times N'$	Matrix of candidate endmembers from $\boldsymbol{S}$
$\boldsymbol{X}$	$N' \times N$	Abundances (w.r.t. to candidate endmembers in $\widetilde{\boldsymbol{S}}_\omega$)

widely used processing pipelines. Alternative methods jointly perform (part of) these tasks in order to solve the blind source separation problem [Eches et al., 2010, Honeine and Richard, 2012, Miao et al., 2007]. We propose a blind and fully constrained approach which jointly estimates the abundances and the endmembers spectra without prior knowledge of the endmembers number.

In order to introduce our approach, we shall start by describing the noise-free spectral mixing case first. Consider the noise-free linear spectral mixing model where a mixed pixel is expressed as a linear combination of the endmembers weighted by their fractional abundances

$$\tilde{\boldsymbol{s}}_n = \sum_{i=1}^{M} \boldsymbol{r}_i a_{in} \,, \forall\ n = 1, \ldots, N, \tag{3.1}$$

in matrix form, we simply have

$$\widetilde{\boldsymbol{S}} = \boldsymbol{R}\boldsymbol{A}, \tag{3.2}$$

where $\widetilde{\boldsymbol{S}} = [\tilde{\boldsymbol{s}}_1, \ldots, \tilde{\boldsymbol{s}}_N]$, $\boldsymbol{R} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_M]$, $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N]$, and $\tilde{\boldsymbol{s}}_n$ is the L-dimensional (noise-free) spectrum of the n-th pixel, L is the number of frequency bands, N is the number of pixels in the image, $\boldsymbol{r}_i$ is L-dimensional spectrum of the i-th endmember, M is the number of endmembers, $\boldsymbol{a}_n$ is the M-dimensional abundance vector of the n-th pixel, and a_{in} (the (i, n)-th entry of matrix $\boldsymbol{A}$) represents the abundance of the endmember $\boldsymbol{r}_i$ in pixel $\tilde{\boldsymbol{s}}_n$. The tilde over symbols in this chapter refers to noise-free data. All vectors are column vectors. The main

notations used in this chapter are summarized in table 3.1. Furthermore, the abundances obey the non-negativity and sum-to-one constraints

$$\begin{cases} a_{in} \geq 0, \forall\ i \text{ and } n, \\ \sum_{i=1}^{M} a_i = 1, \forall\ n. \end{cases} \tag{3.3}$$

In this study, we shall assume that the endmembers are unknown but present in the scene. Let ω be a subset of N' indexes in $\{1, \ldots, N\}$ that contains at least the column index of each endmember. Under these assumptions, and without loss of generality, we observe that the mixing model (3.2) can be reformulated as follows

$$\widetilde{\boldsymbol{S}} = \widetilde{\boldsymbol{S}}_\omega \boldsymbol{X}, \tag{3.4}$$

where $\widetilde{\boldsymbol{S}}_\omega = [\tilde{\boldsymbol{s}}_{\omega_1}, \ldots, \tilde{\boldsymbol{s}}_{\omega_{N'}}]$ denotes the restriction of $\widetilde{\boldsymbol{S}}$ to its columns indexed by ω, and $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]$ is the $N' \times N$ abundance matrix with respect to $\widetilde{\boldsymbol{S}}_\omega$. Similarly as above, x_{ij} is the abundance of $\tilde{\boldsymbol{s}}_{\omega_i}$ in $\tilde{\boldsymbol{s}}_j$. Let $\boldsymbol{x}_{\lambda_i} = [x_{i1}, \ldots, x_{iN}]^\top$ be the i-th row of $\boldsymbol{X}$, i.e. $\boldsymbol{x}_{\lambda_i}$ contains the abundances of the i-th endmember in all the pixels, which is also known as the abundance map of the i-th endmember. On the one hand, if $\tilde{\boldsymbol{s}}_{\omega_i}$ is an endmember, $\boldsymbol{x}_{\lambda_i}$ has non-zero entries and represents the corresponding abundance map. On the other hand, if $\tilde{\boldsymbol{s}}_{\omega_i}$ is a mixed pixel, $\boldsymbol{x}_{\lambda_i}$ has all its elements equal to zero. As a consequence, $\boldsymbol{X}$ admits $N' - M$ rows of zeros, the other rows being equal to rows of $\boldsymbol{A}$. This means that $\boldsymbol{X}$ allows to identify the endmembers in $\widetilde{\boldsymbol{S}}$ through its non-zero rows, which is an interesting property to be exploited in the case where the endmembers are unknown. Let us now turn to the more realistic situation where some noise corrupts the observations. In this case, model (3.4) becomes

$$\boldsymbol{S} = \widetilde{\boldsymbol{S}} + \boldsymbol{E} = \widetilde{\boldsymbol{S}}_\omega \boldsymbol{X} + \boldsymbol{E}, \tag{3.5}$$

where $\boldsymbol{S}$ denotes the available data, and $\boldsymbol{E}$ is the noise supposed to be additive.

The aim of this chapter is to derive two unmixing approaches with increasing complexity, depending on how noise is to be handled. These methods are blind in the sense that the endmembers and their cardinality are unknown. The first one considers the approximate model

$$\boldsymbol{S} \approx \boldsymbol{S}_\omega \boldsymbol{X} + \boldsymbol{E}, \tag{3.6}$$

compared to (3.5), we thus assume that noise does not dramatically affect the factorization of the mixing process, which is valid for very high signal-to-noise ratio (SNR). With this approach, we shall look for a few columns of $\boldsymbol{S}_\omega$ that can effectively represent the whole scene. In order to estimate the abundance matrix $\boldsymbol{X}$, we use prior information. First, we impose that the estimated

abundances obey the non-negativity and sum-to-one constraints, namely, $x_{ij} \geq 0$ for all i and j, and $\sum_{i=1}^{N'} x_{ij} = 1$ for all j. In addition, as discussed above, the algorithm has to force rows of $\boldsymbol{X}$ to be zero vectors in order to identify the endmembers. Because the locations and the cardinality of the endmembers are unknown, the set of candidates has to be sufficiently large, that is, $N' \gg M$. We thus expect many rows in $\boldsymbol{X}$ to be equal to zero. To promote this effect, the so-called Group Lasso $\ell_{2,1}$-norm regularization can be employed [Yuan and Lin, 2006]. Because model (3.6) is a poor approximation of model (3.5) as the noise power increases, we shall also propose an alternative strategy to solve the unmixing problem based on the exact model (3.5). The first approach leads to a convex optimization problem that can be solved with the ADMM [Boyd et al., 2011]. The second one takes the noise in $\boldsymbol{S}_\omega$ into account, which results in a non-convex and heteroscedastic optimization problem. The latter will be solved with an iteratively reweighted least squares algorithm.

In both cases, the proposed sparse-based regression strategy subserves a blind and self-dependent framework. This is due to the fact that the endmembers and the abundances are jointly estimated using a spectral library derived from the observations themselves. Hence, the resulting image endmembers are derived from the purest pixels in the scene. The advantage of this approach compared to the use of a spectral library built from field campaigns or laboratory measurements is that the endmembers and the available image spectra are estimated using the same sensors and under the same conditions. However, the proposed approach relies on the presence of the endmembers among the observations, which is conditioned by the spatial resolution of the sensor and the spatial distribution organization of the image.

To the best of our knowledge, this work is the first that proposes to solve the noisy problem (3.5). Few models similar to the approximate model (3.6) have been studied in the literature [Esser et al., 2012, Fu et al., 2013, Iordache et al., 2013, 2014b]. These last four works assume that $\boldsymbol{S}_\omega$ is noise-free. Moreover, in [Esser et al., 2012], the authors use an $\ell_{1,\infty}$-norm instead of the $\ell_{2,1}$-norm regularization, and incorporate an additional ℓ_1-norm instead of the unit-sum constraint considered here. In [Fu et al., 2013], the authors derive a Matching Pursuit approach [S. G. Mallat, 1993] in order to estimate the endmembers. With this greedy approach, neither the positivity, nor the sum-to-one constraints, are taken into account. A similar technique is considered in [Iordache et al., 2013], but the authors do not assume that the endmembers are present in the scene and use a predefined dictionary. In their recent work [Iordache et al., 2014b], the authors of [Iordache et al., 2013] apply model (3.6) in order to extract the endmembers from the observations.

The rest of this chapter is organized as follows. Sections 3.2 and 3.3 respectively describe the unmixing models (3.6) and (3.5), and the corresponding estimation methods. Section 3.4 provides experimental results on synthetic and real data. Finally, Section 3.5 concludes this chapter.

3.2 Group lasso, unit sum, and positivity (GLUP)

3.2.1 Optimization problem

The aim of this section is to derive the estimation method for model (3.6). We assume that the noise $\boldsymbol{E}$ is Gaussian independent and identically distributed, with zero mean and possibly unknown variance σ^2, that is, $e_{ki} \sim \mathcal{N}(0, \sigma^2)$. The negative log-likelihood for model (3.6) is given by

$$\mathcal{L}(\boldsymbol{X}) = \frac{NL}{2}\log(2\pi) + \frac{NL}{2}\log(\sigma^2) + \frac{1}{2\sigma^2}\|\boldsymbol{S} - \boldsymbol{S}_\omega \boldsymbol{X}\|_{\mathrm{F}}^2. \tag{3.7}$$

The maximum likelihood (ML) estimate, namely, the minimizer of $\mathcal{L}(\boldsymbol{X})$, is the solution of the Least Squares (LS) approximation problem

$$\text{minimize}_{\boldsymbol{X}} \|\boldsymbol{S} - \boldsymbol{S}_\omega \boldsymbol{X}\|_{\mathrm{F}}^2. \tag{3.8}$$

Since model (3.6) follows from an approximation of model (3.5), the relevance of this LS fidelity term is essentially to ensure that $\boldsymbol{S}_\omega \boldsymbol{X}$ matches $\boldsymbol{S}$. The unmixing problem under investigation, however, requires that $\boldsymbol{X}$ only has a few rows different from zero, in addition to the non-negativity and sum-to-one constraints. This leads to following convex optimization problem

$$\begin{array}{ll} \text{minimize}_{\boldsymbol{X}} & \frac{1}{2}\|\boldsymbol{S} - \boldsymbol{S}_\omega \boldsymbol{X}\|_{\mathrm{F}}^2 + \mu \sum_{k=1}^{N'} \|\boldsymbol{x}_{\lambda_k}\|_2 \\ \text{subject to} & x_{ij} \geq 0 \quad \forall\, i, j \\ & \sum_{i=1}^{N'} x_{ij} = 1 \quad \forall\, j, \end{array} \tag{3.9}$$

with $\mu \geq 0$ a regularization parameter and $\boldsymbol{x}_{\lambda_k}$ the k-th row of $\boldsymbol{X}$. The Group Lasso regularization term induces sparsity in the estimated abundance matrix at the group level [Yuan and Lin, 2006], by possibly driving several rows of $\boldsymbol{X}$ to zero. It is worth noting that when $\mu = 0$ and $\boldsymbol{S}_\omega = \boldsymbol{S}$, the identity matrix is a solution of problem (3.9). This solution may not be unique depending on $\boldsymbol{S}$. It follows that the efficiency of our approach relies on the $\ell_{2,1}$-norm regularization function.

3.2.2 ADMM algorithm

The solution of problem (3.9) can be obtained in a simple and flexible manner using the ADMM algorithm [Boyd et al., 2011]. We consider the canonical form

$$\begin{array}{ll} \text{minimize}_{\boldsymbol{X},\boldsymbol{Z}} & \frac{1}{2}\|\boldsymbol{S}-\boldsymbol{S}_\omega\boldsymbol{X}\|_{\mathrm{F}}^2+\mu\sum_{k=1}^{N'}\|\boldsymbol{z}_{\lambda_k}\|_2+\mathcal{I}_{\mathbb{R}_+^{N'\times N}}(\boldsymbol{Z}) \\ \text{subject to} & \boldsymbol{\mathcal{A}}\boldsymbol{X}+\boldsymbol{\mathcal{B}}\boldsymbol{Z}=\boldsymbol{\mathcal{C}} \end{array} \tag{3.10}$$

with

$$\boldsymbol{\mathcal{A}}=\begin{pmatrix}\boldsymbol{I}_{N'}\\ \mathbf{1}_{N'}^\top\end{pmatrix},\ \boldsymbol{\mathcal{B}}=\begin{pmatrix}-\boldsymbol{I}_{N'}\\ \mathbf{0}_{N'}^\top\end{pmatrix},\ \boldsymbol{\mathcal{C}}=\begin{pmatrix}\mathbf{0}_{N'\times N}\\ \mathbf{1}_N^\top\end{pmatrix},$$

where $\mathcal{I}_{\mathbb{R}_+^{N'\times N}}(\boldsymbol{Z})$ is the indicator of the positive orthant guarantying the positivity constraint, that is, $\mathcal{I}_{\mathbb{R}_+^{N'\times N}}(\boldsymbol{Z})=0$ if $\boldsymbol{Z}\succeq\mathbf{0}$ and $+\infty$ otherwise, $\boldsymbol{I}_{N'}$ is the $N'\times N'$ identity matrix, $\mathbf{0}_{N'\times N}$ is an $N'\times N$ matrix of zeros, $\mathbf{1}_{N'}$ and $\mathbf{0}_{N'}$ are $N'\times 1$ vectors of ones and zeros respectively. The equality constraint $\boldsymbol{\mathcal{A}}\boldsymbol{X}+\boldsymbol{\mathcal{B}}\boldsymbol{Z}=\boldsymbol{\mathcal{C}}$ imposes the consensus $\boldsymbol{X}=\boldsymbol{Z}$ and the sum-to-one constraint. Note that defining the constraint matrices differently, in particular setting $\boldsymbol{\mathcal{A}}=\boldsymbol{I}_{N'}$, $\boldsymbol{\mathcal{B}}=-\boldsymbol{I}_{N'}$, and $\boldsymbol{\mathcal{C}}=\mathbf{0}_{N'\times N}$ allows to drop the sum-to-one constraint. In matrix form, the augmented Lagrangian associated with problem (3.10) is given by [Eckstein and Bertsekas, 1992]

$$\begin{aligned}\mathcal{L}_\rho(\boldsymbol{X},\boldsymbol{Z},\boldsymbol{\Lambda})=\ & \tfrac{1}{2}\|\boldsymbol{S}-\boldsymbol{S}_\omega\boldsymbol{X}\|_{\mathrm{F}}^2+\mu\textstyle\sum_{k=1}^{N'}\|\boldsymbol{z}_{\lambda_k}\|_2+\mathcal{I}_{\mathbb{R}_+^{N'\times N}}(\boldsymbol{Z})+\tfrac{\rho}{2}\|\boldsymbol{\mathcal{A}}\boldsymbol{X}+\boldsymbol{\mathcal{B}}\boldsymbol{Z}-\boldsymbol{\mathcal{C}}\|_{\mathrm{F}}^2\\ & +\mathrm{trace}(\boldsymbol{\Lambda}^\top(\boldsymbol{\mathcal{A}}\boldsymbol{X}+\boldsymbol{\mathcal{B}}\boldsymbol{Z}-\boldsymbol{\mathcal{C}})),\end{aligned} \tag{3.11}$$

where $\boldsymbol{\Lambda}$ is the matrix of Lagrange multipliers, μ and ρ are positive regularization and penalty parameters, respectively. The flexibility of the ADMM lies in the fact that it splits the initial variable $\boldsymbol{X}$ into two variables, $\boldsymbol{X}$ and $\boldsymbol{Z}$, and equivalently the initial problem into two subproblems. At iteration $k+1$, the ADMM algorithm is outlined by three sequential steps. First, the augmented Lagrangian is minimized with respect to the unknown variable $\boldsymbol{X}$ and then with respect to $\boldsymbol{Z}$ while in each minimization keeping the other variables fixed to their previous estimate. Finally, the matrix of Lagrange multipliers is updated. To summarize, the ADMM at iteration $k+1$ consists of the following steps:

$$\begin{aligned}\boldsymbol{X}^{k+1} &= \text{minimize}_{\boldsymbol{X}}\ \mathcal{L}_\rho(\boldsymbol{X},\boldsymbol{Z}^k,\boldsymbol{\Lambda}^k),\\ \boldsymbol{Z}^{k+1} &= \text{minimize}_{\boldsymbol{Z}}\ \mathcal{L}_\rho(\boldsymbol{X}^{k+1},\boldsymbol{Z},\boldsymbol{\Lambda}^k),\\ \boldsymbol{\Lambda}^{k+1} &= \boldsymbol{\Lambda}^k+\rho(\boldsymbol{\mathcal{A}}\boldsymbol{X}^{k+1}+\boldsymbol{\mathcal{B}}\boldsymbol{Z}^{k+1}-\boldsymbol{\mathcal{C}}).\end{aligned} \tag{3.12}$$

The ADMM steps are developed hereafter. To keep the notations simple, we drop the iteration index in the development of the first two steps.

$\boldsymbol{X}$ minimization step

This step consists of minimizing the augmented Lagrangian with respect to $\boldsymbol{X}$ while fixing $\boldsymbol{Z}$ and $\boldsymbol{\Lambda}$ to their previous estimates. The augmented Lagrangian is quadratic in terms of $\boldsymbol{X}$. As a result, the solution has an analytical expression that is obtained by setting the gradient with respect to $\boldsymbol{X}$ of the augmented Lagrangian to zero:

$$\boldsymbol{X} = ({\boldsymbol{S}_\omega}^\top \boldsymbol{S}_\omega + \rho \boldsymbol{\mathcal{A}}^\top \boldsymbol{\mathcal{A}})^{-1} (\boldsymbol{S}_\omega^\top \boldsymbol{S} - \boldsymbol{\mathcal{A}}^\top [\boldsymbol{\Lambda} + \rho\,(\boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}})]). \tag{3.13}$$

$\boldsymbol{Z}$ minimization step

After discarding the terms that are independent of $\boldsymbol{Z}$, the minimization of the augmented Lagrangian with respect to $\boldsymbol{Z}$ reduces to solving the following problem:

$$\begin{array}{ll} \text{minimize}_{\boldsymbol{Z}} & \mu \sum_{k=1}^{N'} \|\boldsymbol{z}_{\lambda_k}\|_2 + \operatorname{trace}(\boldsymbol{\Lambda}^\top \boldsymbol{\mathcal{B}}\boldsymbol{Z}) + \frac{\rho}{2} \|\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}\|_{\mathrm{F}}^2 \\ \text{subject to} & \boldsymbol{Z} \succeq \boldsymbol{0}. \end{array} \tag{3.14}$$

This minimization step can be split into N problems given the structure of matrices $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$, one for each row of $\boldsymbol{Z}$, that is,

$$\text{minimize}_{\boldsymbol{z}} \quad \tfrac{1}{2}\|\boldsymbol{z} - \boldsymbol{v}\|_2^2 + \alpha \|\boldsymbol{z}\|_2 + \mathcal{I}_{\mathbb{R}_+^N}(\boldsymbol{z}), \tag{3.15}$$

where $\boldsymbol{v} = \boldsymbol{x} + \rho^{-1}\boldsymbol{\lambda}$ and $\alpha = \rho^{-1}\mu$. Vectors $\boldsymbol{\lambda}$, $\boldsymbol{x}$ and $\boldsymbol{z}$ correspond to a given row of $\boldsymbol{\Lambda}$, $\boldsymbol{X}$ and $\boldsymbol{Z}$, respectively. The minimization problem (3.15) admits a unique solution given by the proximity operator [Combettes and Pesquet, 2009] of $\mathcal{F}_1(\boldsymbol{z}) = \alpha\|\boldsymbol{z}\|_2 + \mathcal{I}_{\mathbb{R}_+^N}(\boldsymbol{z})$:

$$\begin{cases} \boldsymbol{z}^* = \boldsymbol{0}_N & \text{if } \|(\boldsymbol{v})_+\|_2 < \alpha \\ \boldsymbol{z}^* = \left(1 - \frac{\alpha}{\|(\boldsymbol{v})_+\|_2}\right)(\boldsymbol{v})_+ & \text{otherwise,} \end{cases} \tag{3.16}$$

where $(\cdot)_+ = \max(\boldsymbol{0}, \cdot)$. On the one hand, the proximity operator of $\mathcal{F}_1(\boldsymbol{z}) = \alpha\|\boldsymbol{z}\|_2$ is the multidimensional shrinkage thresholding operator (MiSTO) [Puig et al., 2011]. On the other hand, the proximity operator of the indicator function $\mathcal{F}_2(\boldsymbol{z}) = \mathcal{I}_{\mathbb{R}_+^N}(\boldsymbol{z})$ is the projection onto the positive orthant. The proximity operator of $\mathcal{F}(\boldsymbol{z})$ in (3.16), that we refer to as Positively constrained MiSTO, is an extension of both previous operators. The solution is of the form

$$\operatorname{prox}_{\mathcal{F}} = \operatorname{prox}_{\mathcal{F}_1} \circ \operatorname{prox}_{\mathcal{F}_2},$$

that is, the thresholding of the projection. Operator (3.16) was recently used in [Thiebaut et al., 2013]. The derivation of this operator can be found in Appendix A.

Update of the Lagrange multipliers

The update of the Lagrange multipliers is carried out at the end of each ADMM iteration, i.e. after the $\boldsymbol{X}$ and $\boldsymbol{Z}$ updates. Note that the Lagrange multipliers represent the running sum of residuals. It gives an insight on the convergence of the algorithm. More precisely, as the number of iterations tends to infinity, the primal residual tends to zero and the Lagrange multipliers converge to the dual optimal point. The update of the Lagrange multipliers is given by:

$$\boldsymbol{\Lambda}^{k+1} = \boldsymbol{\Lambda}^{k} + \rho(\boldsymbol{\mathcal{A}}\boldsymbol{X}^{k+1} + \boldsymbol{\mathcal{B}}\boldsymbol{Z}^{k+1} - \boldsymbol{\mathcal{C}}). \tag{3.17}$$

Stopping criteria

As suggested in [Boyd et al., 2011], a reasonable stopping criteria is that the primal and dual residuals must be smaller than some tolerance thresholds, namely,

$$\begin{aligned} &\|\boldsymbol{\mathcal{A}}\boldsymbol{X}^{k+1} + \boldsymbol{\mathcal{B}}\boldsymbol{Z}^{k+1} - \boldsymbol{\mathcal{C}}\|_{\mathrm{F}} \leq \epsilon_{\mathrm{pri}}, \\ &\|\rho\boldsymbol{\mathcal{A}}^{\top}\boldsymbol{\mathcal{B}}(\boldsymbol{Z}^{k+1} - \boldsymbol{Z}^{k})\|_{\mathrm{F}} \leq \epsilon_{\mathrm{dual}}. \end{aligned} \tag{3.18}$$

The pseudocode for the so-called group lasso, unit sum, and positivity (GLUP) method is provided by Algorithm 1. It is worth emphasizing that the main difference between the ADMM steps developed in GLUP and those in [Iordache et al., 2013] arises in the ADMM variable splitting. The global problem in [Iordache et al., 2013] is decomposed into three subproblems: the least squares minimization, the Group Lasso regularization, and projection on the positive orthant. A consequence is that three ADMM variables are used instead of two, leading to additional steps. Furthermore, the sum-to-one constraint is not considered in [Iordache et al., 2013].

3.3 Reduced noise GLUP (NGLUP)

3.3.1 Optimization problem

We now turn to the more realistic model (3.5). Let $\boldsymbol{E}_{\omega}$ and $\boldsymbol{I}_{\omega}$ be the $L \times N'$ and $N \times N'$ restrictions of $\boldsymbol{E}$ and $\boldsymbol{I}_N$ to the columns indexed by ω, respectively. The mixing model given by (3.5) can be rewritten as:

$$\boldsymbol{S} = (\boldsymbol{S}_{\omega} - \boldsymbol{E}_{\omega})\boldsymbol{X} + \boldsymbol{E} = \boldsymbol{S}_{\omega}\boldsymbol{X} + \boldsymbol{E}(\boldsymbol{I}_N - \boldsymbol{I}_{\omega}\boldsymbol{X}). \tag{3.19}$$

Algorithm 1 : $\boldsymbol{X} = \text{GLUP}(\boldsymbol{S}, \boldsymbol{S}_\omega, \rho, \mu)$

Precompute $\boldsymbol{\mathcal{A}}$, $\boldsymbol{\mathcal{B}}$, and $\boldsymbol{\mathcal{C}}$
Initialize $\boldsymbol{Z} = \boldsymbol{0}$ and $\boldsymbol{\Lambda} = \boldsymbol{0}$
$\boldsymbol{Q} = (\boldsymbol{S}_\omega{}^\top \boldsymbol{S}_\omega + \rho \boldsymbol{\mathcal{A}}^\top \boldsymbol{\mathcal{A}})^{-1}$
while $\text{res}_{\text{pri}} \geq \epsilon_{\text{pri}}$ or $\text{res}_{\text{dual}} \geq \epsilon_{\text{dual}}$ **do**
  $\boldsymbol{X} = \boldsymbol{Q}(\boldsymbol{S}_\omega^\top \boldsymbol{S} - \boldsymbol{\mathcal{A}}^\top(\boldsymbol{\Lambda} + \rho\,[\boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}]))$
  $\boldsymbol{Z}^{\text{old}} = \boldsymbol{Z}$
  for $i = 1 \cdots N'$ **do**
    $\boldsymbol{v}_i = ((\boldsymbol{x}_i)^\top + \rho^{-1}\boldsymbol{\lambda}_i)_+$
    if $\|\boldsymbol{v}_i\|_2 < \rho^{-1}\mu$ **then**
      $\boldsymbol{z}_i = \boldsymbol{0}$
    else
      $\boldsymbol{z}_i = \left(1 - \frac{\mu}{\rho\|\boldsymbol{v}_i\|_2}\right)\boldsymbol{v}_i$
    end if
  end for
  $\text{res}_{\text{pri}} = \|\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{C}\|_{\text{F}}$
  $\text{res}_{\text{dual}} = \|\rho\boldsymbol{\mathcal{A}}\boldsymbol{\mathcal{B}}(\boldsymbol{Z} - \boldsymbol{Z}_{\text{old}})\|_{\text{F}}$
  $\boldsymbol{\Lambda} = \boldsymbol{\Lambda} + \rho(\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}})$
end while

Let us define the matrix $\boldsymbol{C}(\boldsymbol{X})$ as

$$\boldsymbol{C}(\boldsymbol{X}) = (\boldsymbol{I}_N - \boldsymbol{I}_\omega \boldsymbol{X})^\top (\boldsymbol{I}_N - \boldsymbol{I}_\omega \boldsymbol{X}). \tag{3.20}$$

It follows that

$$\text{vec}(\boldsymbol{E}(\boldsymbol{I}_N - \boldsymbol{I}_\omega \boldsymbol{X})) \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{C}(\boldsymbol{X}) \otimes \boldsymbol{I}_N), \tag{3.21}$$

where $\otimes$ represents the Kronecker product of matrices, and $\text{vec}(\cdot)$ is the operator that stacks the columns of a matrix on top of each other. The derivation of (3.21) is in Appendix B. Model (3.19) belongs to the family of heteroscedastic models [Hooper, 1993], where the variance of the additive noise depends on the unknown variable $\boldsymbol{X}$. The presence of $\boldsymbol{X}$ in the expression of the noise variance expression has consequences on the negative log-likelihood of model (3.19), which no longer leads to the LS approximation problem. The negative log-likelihood associated with model (3.19) is given by

$$\begin{aligned}
\mathcal{L}(\boldsymbol{X}, \sigma^2) =& \frac{1}{2}\text{vec}(\boldsymbol{S} - \boldsymbol{S}_\omega \boldsymbol{X})^\top (\sigma^2 \boldsymbol{C}(\boldsymbol{X}) \otimes \boldsymbol{I}_N)^{-1} \text{vec}(\boldsymbol{S} - \boldsymbol{S}_\omega \boldsymbol{X}) + \frac{1}{2}\log|\sigma^2 \boldsymbol{C}(\boldsymbol{X}) \otimes \boldsymbol{I}_N| \\
=& \frac{L}{2}\log|\sigma^2 \boldsymbol{C}(\boldsymbol{X})| + \frac{1}{2}\text{trace}((\boldsymbol{S} - \boldsymbol{S}_\omega \boldsymbol{X})(\sigma^2 \boldsymbol{C}(\boldsymbol{X}))^{-1}(\boldsymbol{S} - \boldsymbol{S}_\omega \boldsymbol{X})^\top) \\
=& \frac{L}{2}\log|\sigma^2 \boldsymbol{C}(\boldsymbol{X})| + \frac{1}{2}\|\boldsymbol{S} - \boldsymbol{S}_\omega \boldsymbol{X}\|^2_{(\sigma^2 \boldsymbol{C}(\boldsymbol{X}))^{-1}},
\end{aligned} \tag{3.22}$$

where $|\cdot|$ denotes the determinant of the corresponding matrix. The ML estimate for problem (3.22) with the Group Lasso regularization, non-negativity and sum-to-one constraints yields the

following constrained optimization problem

$$\begin{array}{ll} \text{minimize}_{\boldsymbol{X},\sigma^2} & \frac{L}{2}\log|\sigma^2\boldsymbol{C}(\boldsymbol{X})| + \frac{1}{2}\|\boldsymbol{S}-\boldsymbol{S}_\omega\boldsymbol{X}\|^2_{(\sigma^2\boldsymbol{C}(\boldsymbol{X}))^{-1}} + \mu\sum_{k=1}^{N'}\|\boldsymbol{x}_{\lambda_k}\|_2 \\ \text{subject to} & x_{ij}\geq 0 \quad \forall\, i,j \\ & \sum_{i=1}^{N'} x_{ij} = 1 \quad \forall\, j, \end{array} \tag{3.23}$$

Compared to problem (3.9), the objective function in (3.23) has an additional logarithmic term due to the fact that the noise variance depends on $\boldsymbol{X}$, hence it cannot be dropped. Furthermore, the second term of the objective function in (3.23), namely the Frobenius norm of the difference between the observations and the reconstructed spectra, is weighted by the noise variance and a function of $\boldsymbol{X}$.

3.3.2 ADMM algorithm

Problem (3.23) is not convex and requires the estimation of σ^2. Note that, minimizing the second term in the objective, namely the weighted Frobenius norm, is closely related to the Iteratively Reweighted Least Squares (IRLS) problem that arise in the case of heteroscedastic models [Daubechies et al., 2010]. A difference with IRLS algorithms is that $(\boldsymbol{S}-\boldsymbol{S}_\omega\boldsymbol{X})$ in equation (3.22) is usually substituted by $(\boldsymbol{S}-\boldsymbol{S}_\omega\boldsymbol{X})^\top$. In IRLS, the estimation process is carried out in two steps. The first step consists of updating the entries of the weights matrix, which are usually set to be inversely proportional to the noise variances. The second step is the calculation of a LS estimator using the updated weights. Many strategies can be used to estimate the variances for the weight matrix, see for example [Fuller and Rao, 1978, Carroll and Cline, 1988, Hooper, 1993]. In our case, the resolution of problem (3.23) with respect to σ^2 for fixed $\boldsymbol{X}$ gives a closed form expression for the variance as a function of $\boldsymbol{X}$:

$$\sigma^2(\boldsymbol{X}) = \frac{1}{NL}\text{trace}((\boldsymbol{S}-\boldsymbol{S}_\omega\boldsymbol{X})\,\boldsymbol{C}(\boldsymbol{X})^{-1}(\boldsymbol{S}-\boldsymbol{S}_\omega\boldsymbol{X})^\top). \tag{3.24}$$

Let $\boldsymbol{W}(\boldsymbol{X}) = \sigma^2(\boldsymbol{X})\boldsymbol{C}(\boldsymbol{X})$ denote the weight matrix of the least squares term in (3.23). To solve problem (3.23) with respect to σ^2 and $\boldsymbol{X}$, we propose to proceed iteratively. Let $\boldsymbol{X}^k$ be the solution of the previous iteration. The first step consists of calculating $\boldsymbol{W}(\boldsymbol{X}^k)$ using equations (3.20) and (3.24). In the second step, this updated weight matrix is used to estimate $\boldsymbol{X}^{k+1}$ as follows

$$\begin{array}{ll} \text{minimize}_{\boldsymbol{X}} & \frac{1}{2}\|\boldsymbol{S}-\boldsymbol{S}_\omega\boldsymbol{X}\|^2_{(\boldsymbol{W}^k)^{-1}} + \mu\sum_{k=1}^{N'}\|\boldsymbol{x}_k\|_2 \\ \text{subject to} & x_{ij}\geq 0 \quad \forall\, i,j \\ & \sum_{i=1}^{N'} x_{ij} = 1 \quad \forall\, j, \end{array} \tag{3.25}$$

where $\boldsymbol{W}^k = \boldsymbol{W}(\boldsymbol{X}^k)$. Given $\boldsymbol{W}^k$, problem (3.25) reduces to a weighted version of GLUP (3.9) due to the weighted norm in the first term. The ADMM solution developed in section 3.2 can be adapted to solve the optimization problem (3.25). Minimizing the augmented Lagrangian with respect to $\boldsymbol{Z}$, and updating the Lagrange multipliers, can by carried out exactly as in Section 3.2. For concision, only the $\boldsymbol{X}$ minimization step is described hereafter.

$\boldsymbol{X}$ minimization step

Omitting the terms that do not depend on $\boldsymbol{X}$, the minimization of the augmented Lagrangian with respect to $\boldsymbol{X}$ leads to

$$\text{minimize}_{\boldsymbol{X}} \frac{1}{2}\|\boldsymbol{S} - \boldsymbol{S}_\omega \boldsymbol{X}\|^2_{(\boldsymbol{W}^k)^{-1}} + \text{trace}(\boldsymbol{\Lambda}^\top(\boldsymbol{\mathcal{A}}\boldsymbol{X})) + \frac{\rho}{2}\|\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}\|^2_{\text{F}}. \tag{3.26}$$

Similarly to the previous ADMM algorithm, $\boldsymbol{Z}$ and $\boldsymbol{\Lambda}$ are set to their previous estimates namely $\boldsymbol{Z}^k$ and $\boldsymbol{\Lambda}^k$. However, we drop the iteration index to keep the notations simple. Problem (3.26) is quadratic in $\boldsymbol{X}$ and admits an analytical solution obtained by setting the gradient to zero. This amounts to solving the Sylvester equation [Bartels and Stewart, 1972], which has the following form

$$\boldsymbol{S}_\omega^\top \boldsymbol{S}_\omega \boldsymbol{X}(\boldsymbol{W}^k)^{-1} + \rho \boldsymbol{\mathcal{A}}^\top \boldsymbol{\mathcal{A}}\boldsymbol{X} = \boldsymbol{S}_\omega^\top \boldsymbol{S}(\boldsymbol{W}^k)^{-1} - \rho \boldsymbol{\mathcal{A}}^\top \left(\boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}} + \frac{\boldsymbol{\Lambda}}{\rho}\right). \tag{3.27}$$

In the experiments, we use the Matlab's *dlyap* function to solve this Sylvester equation. Finally, given that problem (3.23) is not convex, an alternating optimization algorithm is more likely to converge to local minima with worse accuracy than the convex version. For this reason, we suggest, as a warm start, to initialize the so-called reduced noise GLUP (NGLUP) algorithm with GLUP's estimate. Algorithm 2 provides the pseudocode for NGLUP. The algorithm contains two main loops. The inner loop aims at finding the solution of problem (3.25), whereas the outer loop updates the weight matrix.

3.4 Experimental results

3.4.1 Synthetic Data

Data generation

The performances of GLUP and NGLUP were evaluated using synthetic data. We used eight endmembers with 420 spectral bands extracted from the USGS spectral library of minerals. These endmembers were previously used in Chen et al. [2012]. Figure 3.1 shows the reflectance spectra of

Algorithm 2 : $\boldsymbol{X} = \text{NGLUP}(\boldsymbol{S}, \boldsymbol{S}_\omega, \rho^\circ, \mu^\circ, \rho, \mu)$

Precompute $\boldsymbol{\mathcal{A}}$, $\boldsymbol{\mathcal{B}}$, and $\boldsymbol{\mathcal{C}}$
Initialize $\boldsymbol{X} = \text{GLUP}(\boldsymbol{S}, \boldsymbol{S}_\omega, \rho^\circ, \mu^\circ)$, $\boldsymbol{Z} = \boldsymbol{X}$, $\quad \boldsymbol{\Lambda} = \boldsymbol{0}$
while $\|\boldsymbol{X} - \boldsymbol{X}_{\text{old}}\|_2 \geq \epsilon_{\text{tol}}$ **do**
  $\boldsymbol{C}(\boldsymbol{X}) = (\boldsymbol{I} - \boldsymbol{I}_\omega \boldsymbol{X})^\top (\boldsymbol{I} - \boldsymbol{I}_\omega \boldsymbol{X})$
  $\sigma^2(\boldsymbol{X}) = \frac{1}{NL}\text{trace}((\boldsymbol{S} - \boldsymbol{S}_\omega \boldsymbol{X})\boldsymbol{C}(\boldsymbol{X})^{-1}(\boldsymbol{S} - \boldsymbol{S}_\omega \boldsymbol{X})^\top)$
  $\boldsymbol{W}(\boldsymbol{X}) = \sigma^2(\boldsymbol{X})\boldsymbol{C}(\boldsymbol{X})$
  $\boldsymbol{X}^{\text{old}} = \boldsymbol{X}$, $J = 1$
  while $(\|\boldsymbol{R}\|_2 \geq \epsilon_{\text{pri}}$ or $\|\boldsymbol{P}\|_2 \geq \epsilon_{\text{dual}})$ and $(J \leq J_{\max})$ **do**
    $\boldsymbol{X}$ = solution of Sylvester equation (3.27)
    $\boldsymbol{Z}_{\text{old}} = \boldsymbol{Z}$
    for $i = 1 \cdots N'$ **do**
      $\boldsymbol{v}_i = ((\boldsymbol{x}_i)^\top + \rho^{-1}\boldsymbol{\lambda}_i)_+$
      if $\|\boldsymbol{v}_i\|_2 < \rho^{-1}\mu$ **then**
        $\boldsymbol{z}_i = \boldsymbol{0}$
      else
        $\boldsymbol{z}_i = \left(1 - \frac{\mu}{\rho\|\boldsymbol{v}_i\|_2}\right)\boldsymbol{v}_i$
      end if
    end for
    $\boldsymbol{R} = \boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}$
    $\boldsymbol{P} = \rho\boldsymbol{\mathcal{A}}\boldsymbol{\mathcal{B}}(\boldsymbol{Z} - \boldsymbol{Z}_{\text{old}})$
    $\boldsymbol{\Lambda} = \boldsymbol{\Lambda} + \rho(\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}})$
    $J = J + 1$
  end while
end while

the endmembers which correspond to the following materials: epidote, kaolinite, buddingtonite, alunite, calcite, almandine, jarosite and lepidolite. The maximum spectral mutual coherence of the eight endmembers was $\Theta_{\max} = 0.9940$, the mutual coherence between two spectra, for example $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$, being the absolute value of the inner product between the two vectors divided by the product of their norms:

$$\Theta_{ij} = \frac{|\boldsymbol{s}_i^\top \boldsymbol{s}_j|}{\|\boldsymbol{s}_i\|_2 \|\boldsymbol{s}_j\|_2}. \tag{3.28}$$

The abundances were generated based on a Dirichlet distribution with unit parameter, as a consequence of which the resulting abundances obeyed the non-negativity and sum-to-one constraints, and were uniformly distributed over the simplex whose vertices are the endmembers. This model is widely used to generate the abundances in simulated data sets, see for example [Nascimento and Bioucas-Dias, 2005, Altmann et al., 2013, Iordache et al., 2014a].

Simulation results

First, we used three endmembers to generate an hyperspectral data set containing $N = 100$ (resp. 500) pixels with a SNR of 50 dB. The pure pixels were indexed by integers 1–3 for simplicity, the

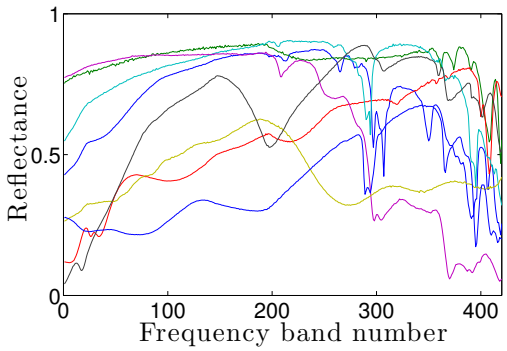


Figure 3.1: Reflectance spectra of the endmembers selected from the USGS spectral library of minerals .

Table 3.2: Probability of detecting $\widehat{M}$ endmembers, using synthetic data generated with $M = 7$ endmembers.

M	2	3	6	7	8	9
GLUP (30 dB)	0	0	0	**1**	0	0
NGLUP (30 dB)	0	0	0	**0.98**	0.02	0
VD (30 dB)	**0.79**	0.21	0	0	0	0
GLUP (20 dB)	0	0	0	**0.71**	0.02	0.27
NGLUP (20 dB)	0	0	0.01	**0.96**	0.03	0
VD (20 dB)	**1**	0	0	0	0	0

mixed pixels being indexed by integers 4–100 (resp. 4–500). We ran GLUP algorithm using all the observations ($\boldsymbol{S}_\omega = \boldsymbol{S}$) with $\mu = 10$ and $\rho = 100$. The primal and dual tolerances, namely ϵ_{pri} and ϵ_{dual}, were set to 10^{-5}. The first row of Figure 3.2 shows the mean of each row of the estimated abundance matrices $\widehat{\boldsymbol{X}}$. We observe that the first three pixels in Figures 3.2 (a) and (b) can be identified as the endmembers since the mean values of the first three rows are clearly different from zero. The second row of Figure 3.2 shows the projection of the data onto the space spanned by the first two principal component analysis (PCA) axes. Blue stars indicate the data points, and red squares indicate the points that had a non-zero row in $\widehat{\boldsymbol{X}}$, namely, those that were identified as the endmembers. We can see from Figures 3.2 (c) and (d) that the red squares correspond to the vertices of the simplex enclosing all the data points. We evaluate the

estimation accuracy using the root mean square error (RMSE) between $\boldsymbol{X}$ and its estimate $\widehat{\boldsymbol{X}}$:

$$\text{RMSE}_{\boldsymbol{X}} = \sqrt{\frac{1}{NN'}\|\boldsymbol{X} - \widehat{\boldsymbol{X}}\|_{\text{F}}^2}. \tag{3.29}$$

In fact, GLUP provided the results in 4.73 (resp. 138.59) seconds[1] with a RMSE equal to 0.0049 (resp., 0.0097) for $N = 100$ (resp., $N = 500$).

We tested NGLUP in less favorable conditions by increasing the number of endmembers and decreasing the SNR. To this end, 7 endmembers were used to generate 93 (resp. 493) mixed pixels. The pure pixels were indexed by integers 1–7. Data points were corrupted with an additive Gaussian noise, corresponding to a SNR of 20 dB. We tested the algorithm for a maximum number of inner iterations $J_{\max} = 1$, 10 and 100. We found that NGLUP converged to the same solution even when the number of inner iterations J was equal to 1. For this reason, only one inner iteration per outer iteration was used for the rest of the experiments. The running time of the algorithm was 77 seconds (resp. 45 min). The first row of Figure 3.3 shows the mean value of each row of the abundance matrix $\widehat{\boldsymbol{X}}$ estimated by GLUP for (a) $N = 100$ and (b) $N = 500$ pixels. The second row of Figure 3.3 shows the mean value of each row of the abundance matrix $\widehat{\boldsymbol{X}}$ estimated by NGLUP for (a) $N = 100$ and (b) $N = 500$ pixels. The 7 largest mean values correspond to the 7 endmembers. As expected, NGLUP converged to a sparser and more accurate solution than GLUP. We observed that similar results can be obtained with approximately sparse mixtures of the endmembers. This scenario can be simulated by setting the Dirichlet distribution's scale parameter to some positive value smaller than one.

We repeated the previous simulation with $M = 7$, and $N = 100$ pixels 100 times. For each realization, we examined the number of mean values of the rows of $\widehat{\boldsymbol{X}}$ that were larger than a predefined threshold empirically set to 0.01. We considered this value, denoted as $\widehat{M}$, as the estimated number of endmembers in the scene. Table 3.2 provides the probability of detecting $\widehat{M}$ endmembers with GLUP and NGLUP, given that the synthetic data was generated with $M = 7$ endmembers. The same task was performed using VD [Chang and Du, 2004]. We compared the results of NGLUP with those of VD, the probability of false alarm of VD being set to 10^{-3}. Table 3.2 shows that NGLUP was able to identify the presence of 7 endmembers in 98% (resp., 96%) of the cases with an SNR of 30 dB (resp. 20 dB). VD only identified 2 endmembers in most cases. Even with higher values of the SNR, VD did not identify the correct number of endmembers. Note that VD has asymptotic convergence, and thus requires a very large number of observations in order to converge. This explains the poor performance of VD compared to GLUP and NGLUP.

[1] Machine specifications: 2.2 GHz Intel Core i7 processor and 8 GB RAM

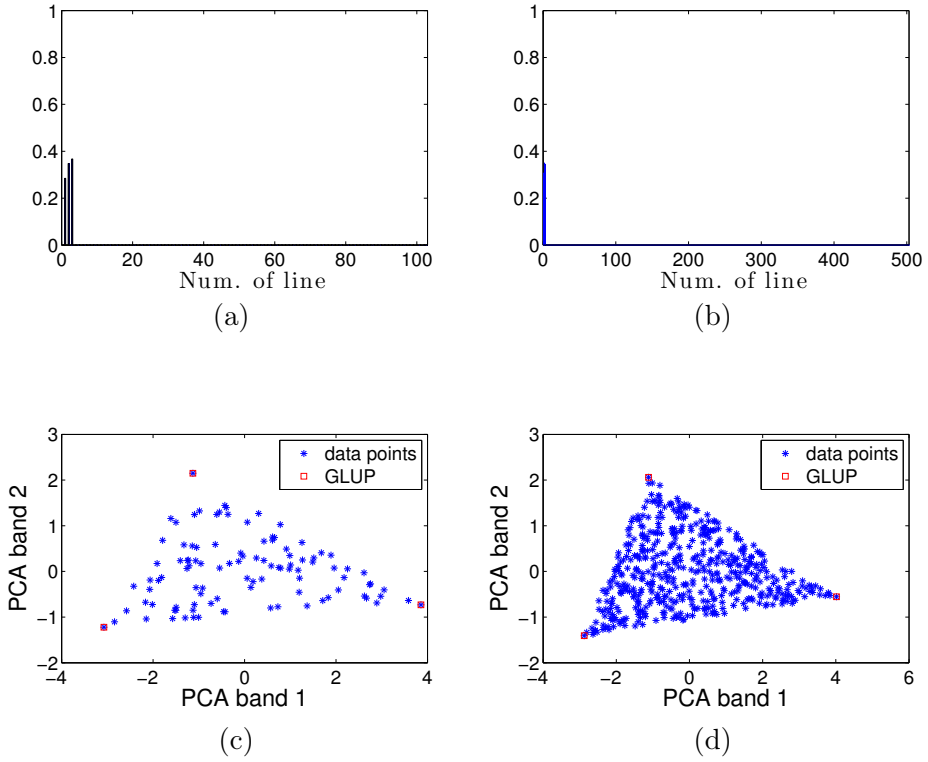


Figure 3.2: First row: Mean value of each row of $\widehat{\boldsymbol{X}}$ estimated with GLUP, obtained with (a) $N = 100$ and (b) $N = 500$ pixels with SNR $= 50$ dB. Second row: 2D data projection and GLUP estimated endmembers obtained with (c) $N = 100$ and (d) $N = 500$ pixels.

Finally, we compared the performance of the proposed approach with 3 endmember extraction algorithms, namely, N-FINDR [Winter, 1999], VCA [Nascimento and Bioucas-Dias, 2005], and SDSOMP [Fu et al., 2013]. In particular, we considered the case where an outlier is present among the observations. In real data, an outlier usually corresponds to bad sensor measurements. To this end, 3 endmembers were used to generate 500 mixed pixels. The 3 endmembers as well as an additional spectra, the outlier, were inserted among the observations. Figure 3.4 shows the 2D data projection after performing 2-dimensional PCA. The outlier can be determined by visual inspection as it is the only point outside the simplex. GLUP found 3 endmembers denoted by green stars. We can see that they correspond to the 3 vertices of the simplex, thus to the true endmembers. On the other hand, with VCA, N-FINDR and SDSOMP, the number of endmembers to find was explicitly set to 3. The three algorithms correctly identified 2 endmembers out of 3, and the outlier instead of the third one. This advantage over geometrical and greedy approaches is related to the formulation of endmember extraction as a penalized optimization problem.

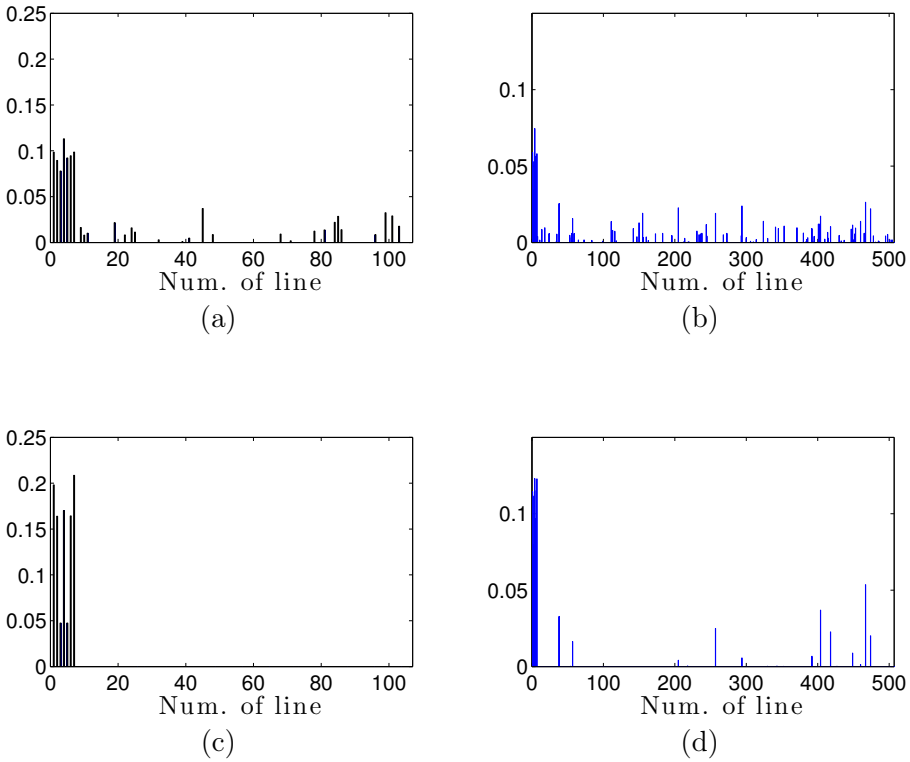


Figure 3.3: First row: Mean value of each row of $\widehat{\boldsymbol{X}}$ estimated with GLUP, obtained with (a) $N = 100$ and (b) $N = 500$ pixels with SNR $= 20$ dB. Second row: Mean value of each row of $\widehat{\boldsymbol{X}}$ estimated with NGLUP, obtained with (a) $N = 100$ and (b) $N = 500$ pixels with SNR $= 20$ dB.

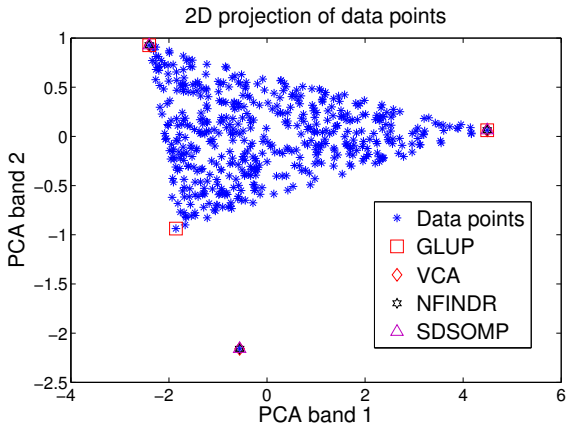


Figure 3.4: Endmembers estimated by GLUP, VCA, N-FINDR, and SDSOMP when the data contains an outlier.

3.4.2 Real data: Cuprite

Data set description

In this section, we shall evaluate the performance of NGLUP using real hyperspectral data. The tests were performed on the so-called Cuprite hyperspectral data provided by NASA's sensor AVIRIS[2]. The hyperspectral data set was captured over a mining district in southern Nevada. Originally, the images were collected with 224 spectral bands over the wavelength interval $400-2500$ nm. After removing the water absorption bands ($1-2$, $105-115$, $150-170$, and $223-224$), 188 bands were left for the analysis. The spatial resolution is about 17 meters. The relatively low spatial resolution of the measurements makes this data particularly interesting to test unmixing algorithms. The image we use for the experiments is a subset of 250×191 pixels, that is, a total of 47750 pixels.

Unmixing results

Typically, $\boldsymbol{S}$ should contain all the available observations, that is, $N = 47750$. In order to alleviate the computational burden, we selected a subset of samples from the original scene. The sampling strategy must guarantee the presence of the endmembers among the selected candidates. To perform this task, we initialized $\boldsymbol{S}$ with the whole observations. Next, we computed the mutual coherence between pairs of candidates. Then, the pair with the largest mutual coherence was identified, and one of the two spectra was randomly discarded. The process was repeated until 300 spectra with the lowest mutual coherences were left in $\boldsymbol{S}$. As shown by the experimental results provided hereafter, this led to a subset of samples sufficiently representative of the original data to identify the endmembers. Following this strategy, the mutual coherence in the case of Cuprite was reduced from 1 (with $N = 47750$) to 0.9996 (with $N = 300$). In addition to alleviating the computational load, an advantage of this strategy is that reducing the coherence of the dictionary, which is the set of available data in our case, improves the performance of the algorithm [Bruckstein et al., 2009, Candes et al., 2006]. Other algorithms can be used to perform this task, for example K-means clustering with an angle constraint as in [Esser et al., 2012], or pruning by subspace projections as in [Iordache et al., 2012b]. However, we found this sampling strategy efficient in our experiments.

We applied GLUP and NGLUP (with $\boldsymbol{S} = \boldsymbol{S}_\omega$) successively using the subset of pixels, the penalty parameter μ being set to 1 and 10000 respectively. With this setting, we obtained 11

[2]available online: http://aviris.jpl.nasa.gov/data/free_data.html

non-zero means in the estimated abundance matrix, that is, 11 endmembers. We also extracted the endmembers using N-FINDR and VCA, the number of endmembers being set to 11. Given that VCA and N-FINDR have lower computational complexity, we applied them on the subset of pixels and on the whole image. Figure 3.5 shows the identified endmembers in each case. It is worth noting that NGLUP was able to identify the endmembers without any prior knowledge on their number. In Figure 3.6, we compare 6 of the spectra estimated by NGLUP with those estimated by N-FINDR when the latter is applied on the whole scene. It can be observed that the 6 spectra correspond to some of the major minerals present in the scene: Sphene, Kaolinite, Muscovite, Alunite, Dumortierite [Nascimento and Bioucas-Dias, 2005]. Figure 3.7 shows the corresponding abundance maps estimated by FCLS, the endmembers being those estimated by NGLUP.

Finally, in table 3.3, we report the root mean square error between the available observations spectra $\boldsymbol{S}$ and the estimated spectra $\widehat{\boldsymbol{S}}$ defined as:

$$\mathrm{RMSE}_{\boldsymbol{S}} = \sqrt{\frac{1}{LN}\|\boldsymbol{S} - \widehat{\boldsymbol{S}}\|_{\mathrm{F}}^2}. \tag{3.30}$$

Furthermore, we report the average and maximum spectral angles. Let θ_i be the spectral angle between the i-th original spectrum $\boldsymbol{s}_i$ and its reconstructed version $\widehat{\boldsymbol{s}}_i$ defined as

$$\theta_i = \mathrm{acos}\left(\frac{\boldsymbol{s}_i^\top \widehat{\boldsymbol{s}}_i}{\|\boldsymbol{s}_i\|_2 \|\widehat{\boldsymbol{s}}_i\|_2}\right). \tag{3.31}$$

The maximum spectral angle (MSA) and average spectral angle (ASA) are defined as

$$\theta_{\max} = \max_{i=1\cdots N}(\theta_i), \tag{3.32}$$

and

$$\theta_{\mathrm{avg}} = \frac{1}{N}\sum_{i=1}^{N}\theta_i, \tag{3.33}$$

respectively. When the three algorithms were applied with the sampled subset, NGLUP always had better scores. When VCA and N-FINDR were applied over the whole scene, NGLUP slightly outperformed N-FINDR and had comparable performance to VCA.

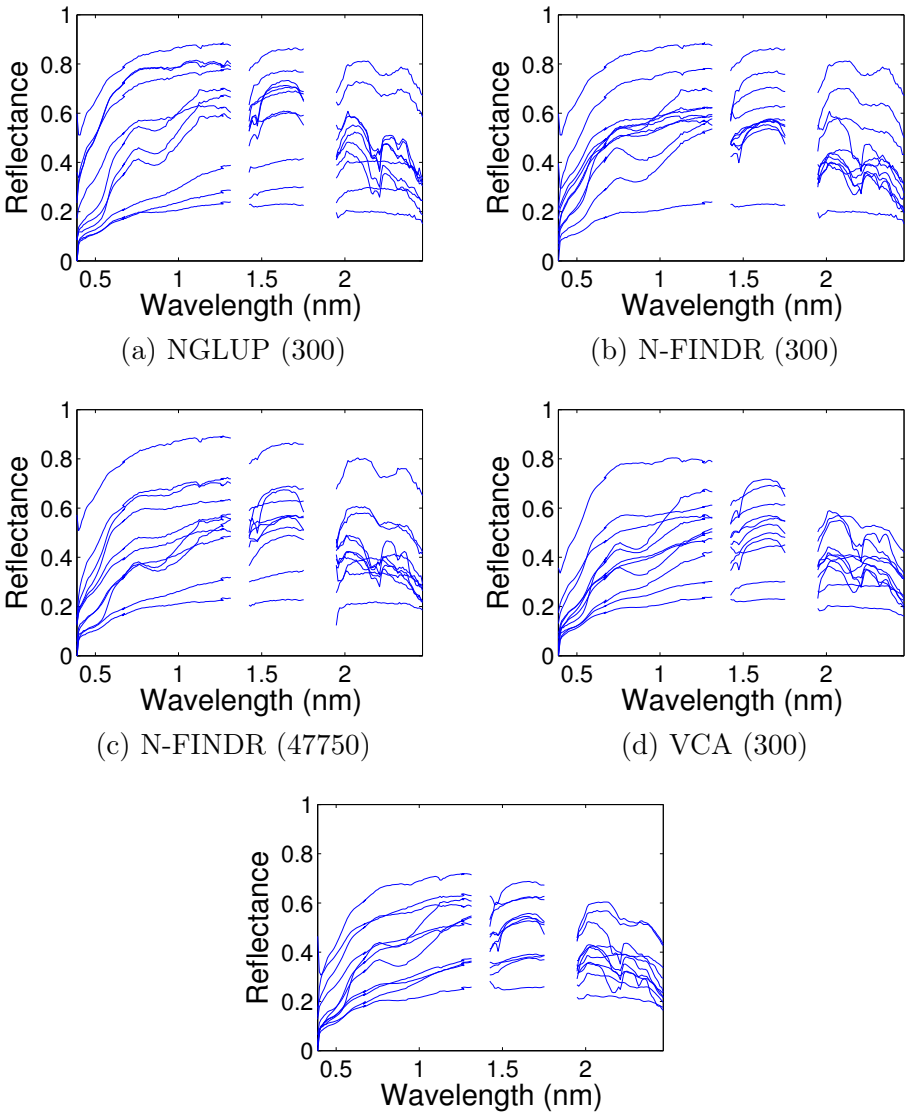
3.5 Conclusion

In this chapter, we presented two approaches for blind and fully constrained unmixing. The two methods are based on mixing models with increasing complexity, and allow to simultaneously determine the endmembers and estimate their abundances in the scene. Compared to the

Table 3.3: Reconstruction quality of Cuprite using NGLUP, N-FINDR, and VCA with FCLS.

Algorithm	$\text{RMSE}_{\boldsymbol{S}}$	ASA	MSA
NGLUP (N=300)	0.0075	1.115°	4.679°
N-FINDR (N=300)	0.0115	1.583°	9.286°
N-FINDR (N=47750)	0.0095	1.400°	7.626°
VCA (N=300)	0.0120	1.289°	7.450°
VCA (N=47750)	0.0062	0.855°	8.020°

first model called GLUP, the second model called NGLUP explicitly considers that endmembers present in the scene are corrupted by noise. The main advantage of the both models was that the endmembers used to characterize the observations were obtained in the same atmospheric conditions as the observations and underwent the same atmospheric corrections as well. Furthermore, since the observations and equivalently the estimated endmembers were corrupted with noise the second model takes into consideration the presence of noise in the endmembers. Experiments on synthetic and real data demonstrated the performance of both approaches. Nevertheless, the limiting factor in NGLUP, is that it is computationally more expensive than GLUP due to the solution of the Sylvester equation. This required sampling the data in order to reduce the computational complexity of this step. Finally, note that both models do not exploit the spatial information inherently present in hyperspectral data. The next chapter studies two graph-based regularizations for unsupervised unmixing that allow to incorporate spatial information in order to improve the unmixing accuracy.



(a) NGLUP (300)

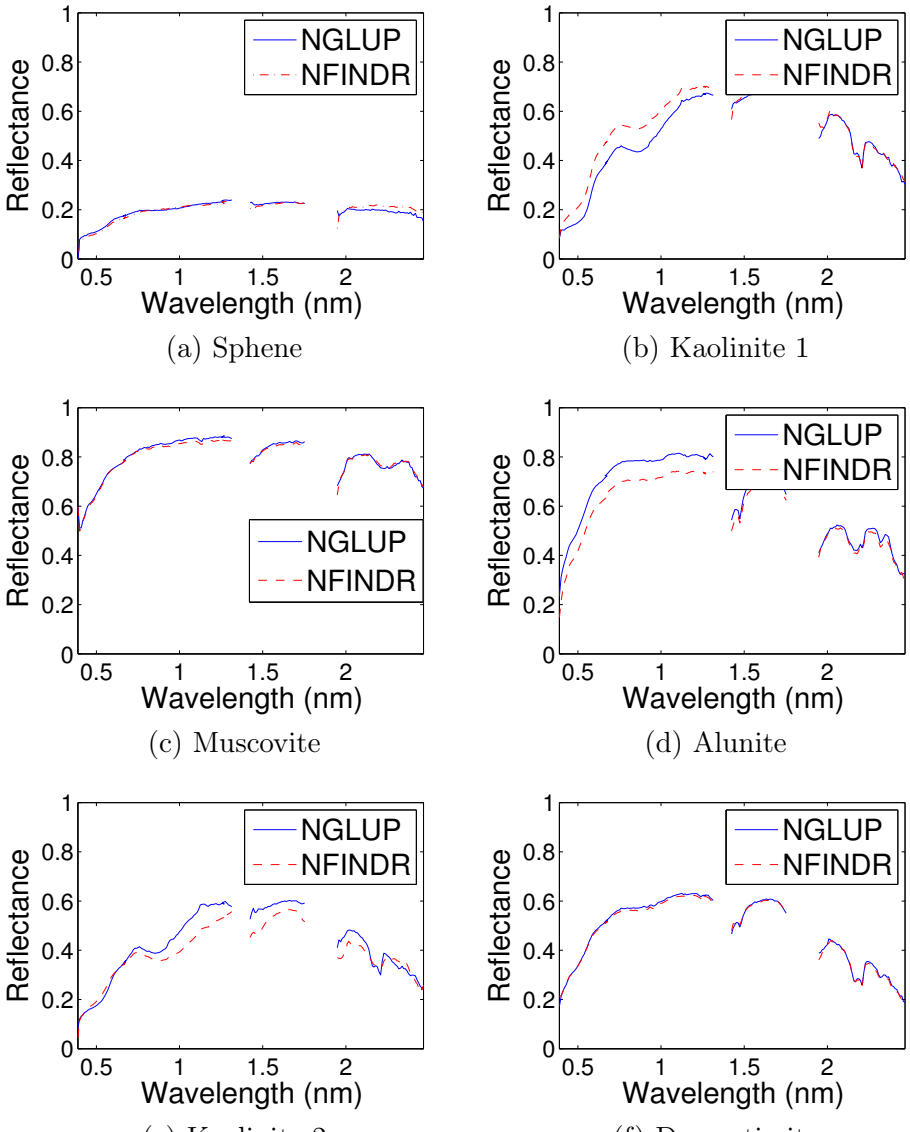
(b) N-FINDR (300)

(c) N-FINDR (47750)

(d) VCA (300)

(e) VCA (47750)

Figure 3.5: Estimated endmembers obtained with (a) NGLUP with 300 samples (b) N-FINDR with 300 samples (c) N-FINDR with all samples (d) VCA with 300 samples (e) VCA with all samples.

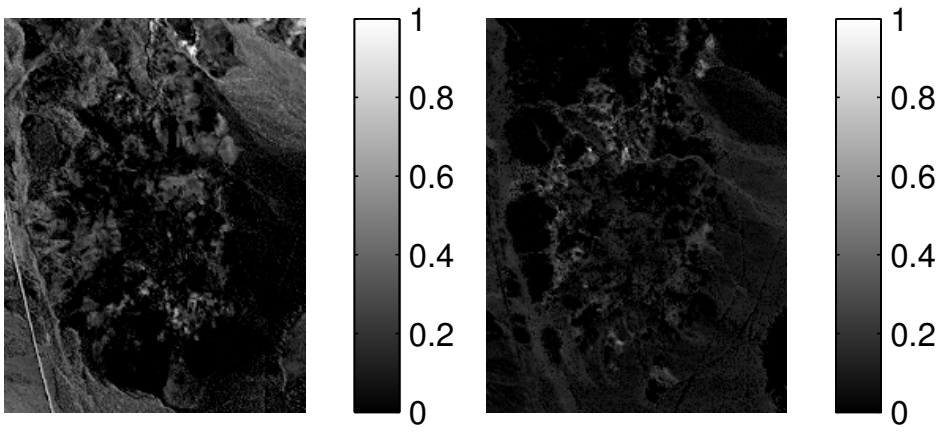


(a) Sphene (b) Kaolinite 1

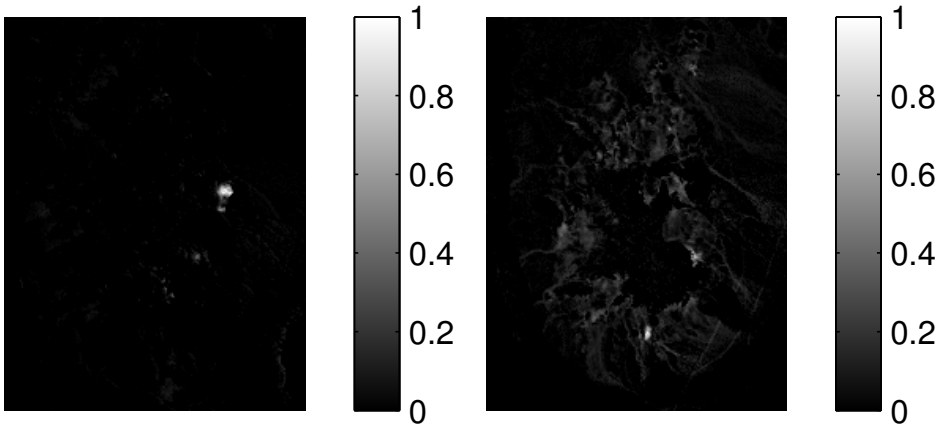
(c) Muscovite (d) Alunite

(e) Kaolinite 2 (f) Dumortierite

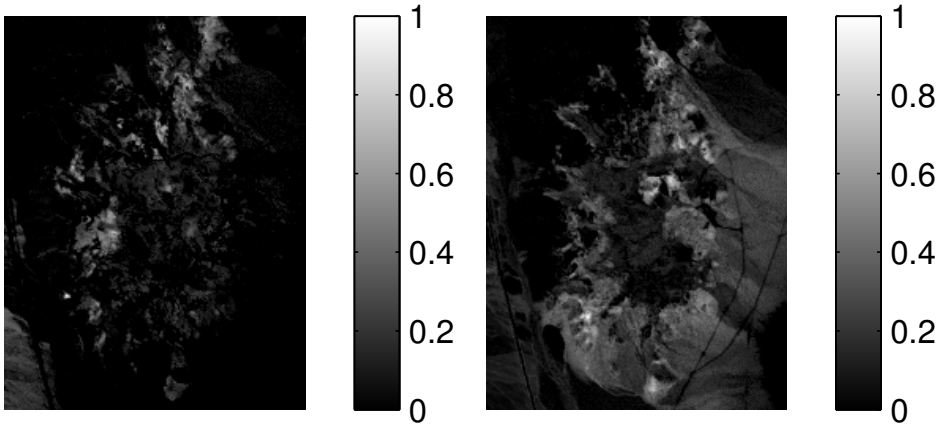
Figure 3.6: Comparison between six endmembers' spectra estimated by NGLUP and by N-FINDR when applied on the whole AVIRIS scene of Cuprite.



(a) Sphene (b) Kaolinite 1



(c) Muscovite (d) Alunite



(e) Kaolinite 2 (f) Dumortierite

Figure 3.7: Abundance maps estimated by FCLS for some of NGLUP endmembers: Sphene, Kaolinite, Muscovite, Alunite, Kaolinite 2, and Dumortierite.

Chapter 4

Graph-based regularized linear unmixing

This chapter has been adapted from the conference papers [Ammanouil et al., 2015a,b].

This chapter introduces two graph-based regularizations in the unsupervised linear unmixing formulation. The proposed regularizations rely upon the construction of a graph representation of the hyperspectral image such that each node in the graph represents a pixel's spectrum, and edges connect *similar* nodes. The first regularization is the quadratic Laplacian regularization imposed on the abundances, and the second regularization is the nonlocal Total variation imposed on the reconstructed spectra. The former regularization is used to promote smooth abundance maps, while the second one is used to promote piece-wise constant reconstructed spectra. The resulting constrained optimization problems are convex and solved using the Alternating Direction Method of Multipliers (ADMM). Finally, simulations conducted on synthetic and real data illustrate the effectiveness of incorporating the proposed regularizations.

4.1 Introduction

In this chapter, we propose to incorporate spatial/spectral prior information regarding the similarities between the different pixels in the unsupervised linear unmixing framework through an additional regularization term. Incorporating this prior information allows to improve the unmixing accuracy. Furthermore, it acknowledges the importance of contextual information inherently present in hyperspectral images. In particular, we perform this task by incorporating a regular-

ization term based on a graph representation of the image. In fact, representing by edges the pairwise spatial and/or spectral similarities between the spectra at different pixels in the image gives rise to a graph, where each node represents a pixel. The resulting graph structure provides additional relational information which can be used to improve the abundance estimation accuracy and complement existing unmixing techniques wether in the supervised or unsupervised case. As we shall see further ahead, the graph-based regularization framework provides an elegant and flexible way to incorporate different types of prior information in the unmixing problem. Graph-based regularizations have been widely used in many fields, and their potential has been demonstrated for many applications including digit recognition and text classification [Zhou et al., 2004], web-page categorization [Zhang et al., 2006], hyperspectral data classification [Camps-Valls et al., 2007], manifold learning [Belkin et al., 2006], and non-local image denoising [Kovac and Smith, 2011, Peyré et al., 2008, Couprie et al., 2013, Zhang and Hancock, 2008] to cite a few.

Similarly to chapter 3, we perform unsupervised unmixing using the group Lasso regularization. Furthermore, a graph-based regularization is incorporated within the optimization problem. More precisely, we consider two different settings. In the first setting, the hyperspectral image is mapped to a complete weighted graph, where the weight of each edge is proportional to a measure of the similarity between the spectra associated with the edge. If two pixels are connected by an edge with a high weight, then it is assumed that their abundances should be smooth. The relevance of using a complete graph representation is that pixels in distant regions of the image can collaborate to improve the abundance estimation. Using tools of discrete calculus on graphs [Shuman et al., 2013], we penalize the squared norm of the discrepancies between the estimated abundances of connected pixels through the use of the graph Laplacian matrix. The corresponding optimization problem is given by:

$$(\mathcal{P}_1) \qquad \begin{array}{ll} \text{minimize}_{\boldsymbol{X}} & \frac{1}{2}\|\boldsymbol{S}-\boldsymbol{\mathcal{D}}\boldsymbol{X}\|_{\mathrm{F}}^2+\mu\sum_{k=1}^{N'}\|\boldsymbol{x}_{\lambda_k}\|_2+\lambda\mathcal{J}_{\mathcal{G}_1}(\boldsymbol{X}), \\ \text{subject to} & x_{ij}\geq 0 \quad \forall\, i,j \\ & \sum_{i=1}^{N'} x_{ij}=1 \quad \forall\, j, \end{array} \tag{4.1}$$

where the first term mainly ensures that the reconstructed spectra matches the observations, the second term is the group Lasso regularization [Yuan and Lin, 2006] that forces all the observations to have the same set of endmembers by promoting zero rows in the estimated abundance matrix (similarly to chapter 3), and the constraints ensure the positivity and the sum-to-one. Note that we have used $\boldsymbol{\mathcal{D}}$ to denote an $L \times N'$ dictionary of spectral materials rather than $\boldsymbol{S}_\omega$ that corresponds to a subset of candidate endmembers chosen from the observations. From

a mathematical point of view, this does not affect the optimization problem. Nevertheless, we use this notation given that a library of spectral materials was used in the experiments. Most importantly, the third term in the objective, namely $\mathcal{J}_{\mathcal{G}_1}(\boldsymbol{X})$ is the quadratic Laplacian regularization. The so called quadratic Laplacian regularization aims at promoting smooth abundances estimates with respect to the graph structure. The expression as well as more details regarding this regularization are given in section 4.3. In the second setting, we opt for a sparse graph by constructing the edge set from the four neighbours of each node and the k nearest neighbours rather than considering a complete graph. Similarly to the first case, the weight of each edge is proportional to a measure of the similarity between the spectra associated with the edge. However, if two pixels are connected by an edge, we assume that their reconstructed spectra rather than their abundances should be piece wise constant. To perform this task, the ℓ_1-norm of the discrepancies between the estimated spectra of connected pixels is penalized through the use of the adjacency matrix of the graph. We refer to this regularization as the nonlocal TV. The corresponding optimization problem is given by:

$$(\mathcal{P}_2) \qquad \begin{array}{ll} \text{minimize}_{\boldsymbol{X}} & \frac{1}{2}\|\boldsymbol{S}-\boldsymbol{\mathcal{D}}\boldsymbol{X}\|_{\mathrm{F}}^2+\mu\sum_{k=1}^{N'}\|\boldsymbol{x}_{\lambda_k}\|_2+\lambda\mathcal{J}_{\mathcal{G}_2}(\boldsymbol{\mathcal{D}}\boldsymbol{X}). \\ \text{subject to} & x_{ij}\geq 0 \quad \forall\, i,j \\ & \sum_{i=1}^{N'} x_{ij}=1 \quad \forall\, j, \end{array} \tag{4.2}$$

where, compared to (4.1), $\mathcal{J}_{\mathcal{G}_2}(\boldsymbol{\mathcal{D}}\boldsymbol{X})$ is the nonlocal TV imposed on the reconstructed spectra. The expression of $\mathcal{J}_{\mathcal{G}_2}(\boldsymbol{\mathcal{D}}\boldsymbol{X})$ is developed in section 4.3. In both cases, the optimization problem is convex and solved using an ADMM algorithm.

Several works in the literature incorporate spatial regularization within the unmixing procedure. For example, in [Iordache et al., 2012a] the authors use a TV regularization for sparse ℓ_1-norm regularized unmixing. The TV regularization promotes piece wise constant abundances estimates while preserving discontinuities. However, the underlying graph in TV is restricted to local connections, and only relates a pixel to its four neighbors. In addition to this, the efficiency of TV in preserving discontinuities depends on the corresponding tuning parameter, i.e., for relatively large values of the tuning parameter much of the detail is lost. In contrast with TV, we use nonlocal graphs which makes it possible for distant but similar pixels to collaborate in order in order to improve the unmixing accuracy. The fact that the graph is weighted proportionally to the similarity between the pixels spectra allows to spatially adapt the corresponding tuning parameter in order to do more effective regularization [Strong and Chan, 1996]. Very recently, the authors of [Lu et al., 2013] and [Tong et al., 2014] used a graph-based regularization, namely the quadratic Laplacian regularization, for sparse $\ell_{1/2}$-norm regularized Non negative Matrix

Factorization (NMF) within the context of blind unmixing. Their algorithm performs alternate minimization in order to simultaneously estimate the endmembers and the abundances. In this work, we use the ADMM algorithm [Boyd et al., 2011] which allows to take into account the sum-to-one and positivity constraints for the abundances, and the Group lasso regularizer which allows the use of large endmember candidates. Several methods in the literature incorporate other spatial or spectral-spatial information in the unmixing problem such as [Jia and Qian, 2007, Castrodad et al., 2011, Eches et al., 2011, Zare, 2011]. Finally, in contrast with all the previous works where the prior is directly imposed on the abundances, the second regularization, namely the nonlocal TV in (4.2) is imposed on the reconstructed spectra rather than the abundances. This choice seemed natural since the weighted graph is built according to the observed spectra themselves. For a detailed review of unmixing methods with spatial information, the reader is referred to [Shi and Wang, 2014].

The rest of this chapter is organized as follows. Section 4.2 describes how to map the hyperspectral image to a weighted graph, section 4.3 introduces the quadratic Laplacian and the nonlocal TV regularizations, section 4.4 develops the ADMM algorithms for solving the graph regularized unmixing problems, section 4.5 discusses the complexity of each algorithm, section 4.6 tests the proposed algorithms on synthetic and real data. Finally, Section 4.7 concludes the chapter.

Table 4.1: Notations for chapter 4

$\boldsymbol{\mathcal{D}}$	$L \times N'$	Spectral library of known materials
$\mathcal{V}$	–	Set of vertices
$\mathcal{E}$	–	Set of edges
$\boldsymbol{\mathcal{W}}$	$N \times N$	Affinity matrix
$\mathcal{L}_{\mathcal{G}}$	$N \times N$	Laplacian matrix of the graph
$\boldsymbol{\Gamma}^\top$	$N \times \lvert\boldsymbol{\Gamma}\rvert$	Incidence matrix ($\boldsymbol{\Gamma}$: edge node matrix of the graph)
$\boldsymbol{D}$	$N \times N$	Degree matrix of the graph

4.2 Image to graph mapping

Before introducing the graph-based regularizations used in (4.1) and (4.2), we first give some notations and describe the strategies used for generating meaningful graphs from the hyperspectral image as done for example in Figure 4.1. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \boldsymbol{\mathcal{W}})$ be the image graph where $\mathcal{V} = \{v_1, \cdots, v_N\}$ is the set of vertices, $\boldsymbol{\mathcal{W}} \in \mathbb{R}^{N \times N}$ is the affinity or adjacency matrix, and

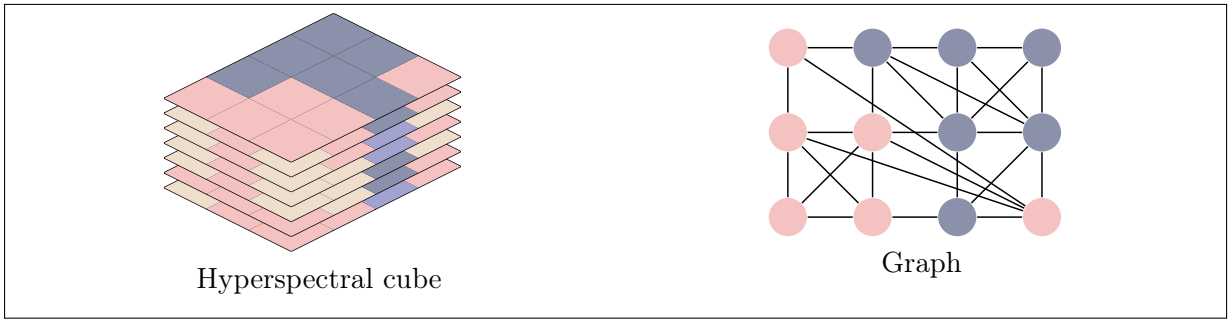


Figure 4.1: Mapping the hyperspectral image to a weighted graph $\mathcal{G}$.

$\mathcal{E} = \{e_{ij} | i \sim j\}$ is the set of (undirected) edges where $i \sim j$ means that vertices v_i and v_j are connected. Every vertex or node in $\mathcal{V}$ represents a pixel in the image, and it is associated with the corresponding pixel's spectrum. There exists different techniques for generating a meaningful set of edges, and assigning a weight to each edge. Hereafter, we briefly describe how to perform these two tasks, for a detailed and comprehensive review see chapter 4 in [Grady and Polimeni, 2010].

4.2.1 Defining the edge set

Two straightforward options for defining the edge set, hence the topology of the graph, are either building a four neighbourhood regular graph, or a fully connected graph. These two cases are respectively depicted in the first and second subfigures of Figure 4.2. The four neighbourhood graph used in classical TV seems like a natural possibility in the case of an image. However, it is not adapted to the underlying image. The fully connected or complete graph is the most thorough choice since it represents all pairwise relationships. However, the number of edges, equal to $\frac{1}{2}N(N-1)$, can become prohibitive for large N. Another alternative is to connect each pixel to its first order spatial neighbours in addition to its k nearest neighbours among the other pixels in the image w.r.t. some spectral distance. This alternative can be seen as a compromise between the four neighbourhood and the complete graph structures. Compared to the four neighbourhood structure, it keeps the connections with the spatially closest neighbour while also adding nonlocal connections with the most spectrally similar pixels. As a result, instead of only having connections with the local spatial neighbourhood of a pixel, spectrally similar pixels provided by a k nearest neighbour are also added. This enhances the topology by allowing the collaboration between pixels located in distant regions across the image [Couprie et al., 2013]. Compared to the complete graph, this alternative allows to remove irrelevant connections and it drastically reduces the number of edges.

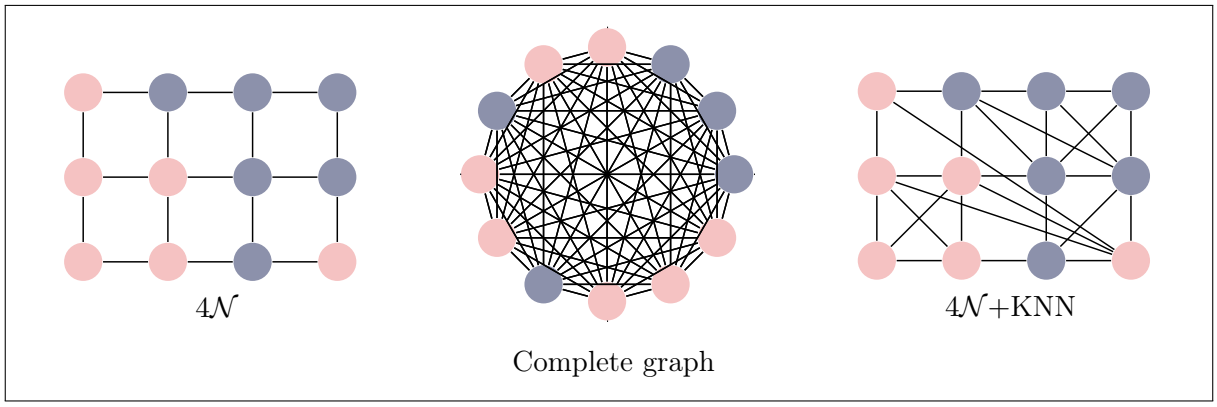


Figure 4.2: Graph representation of the HSI with different edge sets.

4.2.2 Defining the weights

Given the edge set, the weights can be assigned as follows. If two vertices v_i and v_j are not connected by an edge, then the corresponding weight is set to zero ($\mathcal{W}_{ij} = 0$). If two vertices v_i and v_j are connected by an edge e_{ij}, then $\mathcal{W}_{ij}$ is set to some positive value that represents a measure of the similarity between the spectra associated with v_i and v_j. We consider non-oriented graphs, hence $\mathcal{W}_{ij}$ is equal to $\mathcal{W}_{ji}$. Note that assigning a zero weight to an edge is equivalent to removing the corresponding edge.

The entries $\mathcal{W}_{ij}$ of $\boldsymbol{\mathcal{W}}$ satisfy the following conditions. If pixels i and j are similar then $\mathcal{W}_{i,j}$ is set to some positive value proportional to their degree of similarity. If pixels i and j are dissimilar then $\mathcal{W}_{ij}$ tends to zero. There are different heuristics for choosing $\mathcal{W}_{ij}$. In general, a monotonically decreasing function of the spectral distance is used to assign a weight to each edge. For example, this can be done by using a Gaussian kernel:

$$\mathcal{W}_{ij} = \exp\left(-\frac{\|\boldsymbol{s}_i - \boldsymbol{s}_j\|^2}{2\sigma^2}\right), \tag{4.3}$$

where σ is the kernel's bandwidth [Gillis and Bowles, 2012, Zhang et al., 2014]. Or by simply using binary weights and thresholding:

$$\begin{cases} \mathcal{W}_{ij} = 1 & \text{if } \|\boldsymbol{s}_i - \boldsymbol{s}_j\|_2^2 < d_{\text{min}}^2 \\ \mathcal{W}_{ij} = 0 & \text{otherwise,} \end{cases} \tag{4.4}$$

where d_{min}^2 represents the maximum squared spectral distance allowed between connected pixels. In addition to the pixel's spectrum, each pixel can be defined by a vector of spatial features, for instance, the average of its surrounding area, its coordinates in the image. This spatial information leads to a second spatial affinity matrix which can be easily combined with the spectral one [Camps-Valls et al., 2007]. Finally, k-nearest neighbours and thresholding are commonly used in

order to set to zero small weights in $\boldsymbol{\mathcal{W}}$ [Argyriou et al., 2005]. The authors of [Zhang et al., 2014, Gillis and Bowles, 2012, Camps-Valls et al., 2007] propose different strategies for defining an affinity matrix that takes into account both the spatial and the spectral information of a pixel.

4.2.3 Graph operators

In addition to the adjacency matrix, the graph can be characterized by the incidence and Laplacian matrices. In fact the two graph-based regularizations that we introduce in the next section are expressed in terms of these two operators.

- The incidence matrix or the edge-node matrix encodes the incidence relationships between the nodes in $\mathcal{V}$ and the edges in $\mathcal{E}$. Let us denote by $\boldsymbol{\Gamma}^\top \in \mathbb{R}^{N\times|\mathcal{E}|}$ the graph incidence matrix [Grady and Polimeni, 2010], where $|\mathcal{E}|$ represents the cardinality of $\mathcal{E}$ i.e. the number of edges. Each row of $\boldsymbol{\Gamma}$ is indexed by an edge and has two nonzero elements:

$$\begin{cases} \boldsymbol{\Gamma}_{e_{ij},i} = & \mathcal{W}_{ij} \\ \boldsymbol{\Gamma}_{e_{ij},j} = & -\mathcal{W}_{ij}, \end{cases} \tag{4.5}$$

  encoding which vertices are incident at the edge and its corresponding weight.

- Let us denote by $\mathcal{L}_\mathcal{G}$ The Laplacian matrix of the graph. The Laplacian is an $N\times N$ matrix which can be directly defined as:

$$\begin{cases} (\mathcal{L}_\mathcal{G})_{ij} = & -\mathcal{W}_{ij}, & \text{if } i \neq j \\ (\mathcal{L}_\mathcal{G})_{ij} = & \sum_{i=1}^{N} \mathcal{W}_{ij} & \text{otherwise.} \end{cases} \tag{4.6}$$

  Alternatively, the Laplacian matrix can be defined as $\mathcal{L}_\mathcal{G} = \boldsymbol{D} - \boldsymbol{\mathcal{W}}$, where $\boldsymbol{D}$ is the $N\times N$ diagonal matrix degree matrix such as $\boldsymbol{D}_{ii} = \sum_{j=1}^{N} \mathcal{W}_{ij}$ and zero otherwise.

The notations are summarized in Figures (4.4) using the illustrative example in Figure 4.3 of a random Graph $\mathcal{G}$. The graph in Figure 4.3 is characterized by the following node and edge sets $\mathcal{V} = \{v_1, v_2, v_3, v_4\}$, and $\mathcal{E} = \{e_{12}, e_{13}, e_{23}, e_{24}\}$ respectively. The adjacency, degree, Laplacian and incidence matrices are given by Figure 4.4. Note that, in the example of the degree matrix, we have used the shorthand notations $\mathcal{W}_{21+23+24}$ and $\mathcal{W}_{31+32}$ for $\mathcal{W}_{21}+\mathcal{W}_{23}+\mathcal{W}_{24}$ and $\mathcal{W}_{31}+\mathcal{W}_{32}$ respectively.

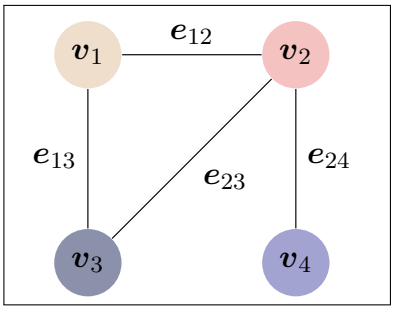


Figure 4.3: Illustrative example of a random graph.

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{pmatrix} 0 & \mathcal{W}_{12} & \mathcal{W}_{13} & 0 \\ \mathcal{W}_{21} & 0 & \mathcal{W}_{23} & \mathcal{W}_{24} \\ \mathcal{W}_{31} & \mathcal{W}_{32} & 0 & 0 \\ 0 & \mathcal{W}_{42} & 0 & 0 \end{pmatrix} \end{array}$$

Adjacency matrix $\boldsymbol{\mathcal{W}}$

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{pmatrix} \mathcal{W}_{12+13} & 0 & 0 & 0 \\ 0 & \mathcal{W}_{21+23+24} & 0 & 0 \\ 0 & & \mathcal{W}_{31+32} & 0 \\ 0 & 0 & 0 & \mathcal{W}_{42} \end{pmatrix} \end{array}$$

Degree matrix $\boldsymbol{D}$

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} e_{12} & e_{12} & e_{23} & e_{24} \end{array} \\ \begin{pmatrix} \mathcal{W}_{12} & \mathcal{W}_{12} & 0 & 0 \\ -\mathcal{W}_{12} & -\mathcal{W}_{12} & \mathcal{W}_{23} & 0 \\ 0 & 0 & -\mathcal{W}_{24} & \mathcal{W}_{24} \\ 0 & 0 & 0 & -\mathcal{W}_{24} \end{pmatrix} \end{array}$$

Incidence matrix $\boldsymbol{\Gamma}^\top$

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{array} \begin{array}{c} \begin{array}{cccc} v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{pmatrix} d_{11} & -\mathcal{W}_{12} & -\mathcal{W}_{13} & 0 \\ -\mathcal{W}_{21} & d_{22} & -\mathcal{W}_{23} & -\mathcal{W}_{24} \\ -\mathcal{W}_{31} & -\mathcal{W}_{32} & d_{33} & 0 \\ 0 & -\mathcal{W}_{42} & 0 & d_{44} \end{pmatrix} \end{array}$$

Laplacian matrix $\mathcal{L}_\mathcal{G}$

Figure 4.4: Adjacency, degree, incidence, and Laplacian matrices of the graph in the illustrative example.

4.3 Graph based regularization

As mentioned previously, we consider two interpretations of the graph. In the first one, we consider that if two nodes are connected by an edge with a relatively high weight, then they are likely to have similar abundances. Whereas in the second one, we consider that if this is the case, then they are likely to have similar reconstructed spectra. We shall now introduce two graph-based regularizations that allow to incorporate this information in the unmixing problem, the quadratic Laplacian regularization and the nonlocal TV.

4.3.1 Quadratic Laplacian regularization

The quadratic Laplacian regularization used in (4.1) is defined as:

$$\mathcal{J}_{\mathcal{G}_1}(\boldsymbol{X}) = \text{trace}(\boldsymbol{X}\mathcal{L}_{\mathcal{G}}\boldsymbol{X}^\top), \tag{4.7}$$

where $\mathcal{L}_{\mathcal{G}}$ is the graph Laplacian matrix. As a result, (4.1) becomes:

$$(\mathcal{P}_1) \quad \begin{array}{ll} \text{minimize}_{\boldsymbol{X}} & \frac{1}{2}\|\boldsymbol{S} - \boldsymbol{\mathcal{D}}\boldsymbol{X}\|_{\text{F}}^2 + \mu \sum_{k=1}^{N'} \|\boldsymbol{x}_{\lambda_k}\|_2 + \lambda \text{trace}(\boldsymbol{X}\mathcal{L}_{\mathcal{G}}\boldsymbol{X}^\top), \\ \text{subject to} & x_{ij} \geq 0 \quad \forall\, i,j \\ & \sum_{i=1}^{N'} x_{ij} = 1 \quad \forall\, j. \end{array} \tag{4.8}$$

To see the relevance of this regularization term in (4.8), we rewrite it as follows [Mohar and Alavi, 1991]:

$$\text{trace}(\boldsymbol{X}\mathcal{L}_{\mathcal{G}}\boldsymbol{X}^\top) \quad = \sum_{i=1}^{N}\sum_{j=1}^{N} \mathcal{W}_{ij}\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2. \tag{4.9}$$

The quantity in (4.9) can be seen as a measure of the discrepancies between all pairs of abundance estimates weighted by their degree of similarity. In the unmixing framework of (4.8), for every row in $\boldsymbol{X}$, the quadratic Laplacian regularization (4.9) penalizes the square of the difference between the abundances of similar pixels proportionally to their degree of similarity. As a result, it promotes smooth abundance estimates with respect to the graph structure. Note that the regularization parameter λ in (4.8) controls the extent at which similar pixels estimate similar abundances, hence it controls the degree of smoothness of the abundance estimates.

4.3.2 Nonlocal TV regularization

The second regularization that we introduce in the unmixing formulation in (4.2) is the nonlocal TV regularization imposed on the reconstructed spectra rather than directly on the abundances. In fact, the graph is intended to capture the features of the image. Thus, it seems appropriate to impose the graph based regularization on the reconstructed image at each spectral band. The nonlocal TV regularization is given by:

$$\mathcal{J}_{\mathcal{G}_2}(\boldsymbol{\mathcal{D}}\boldsymbol{X}) = \|\boldsymbol{\Gamma}(\boldsymbol{\mathcal{D}}\boldsymbol{X})^\top\|_1, \tag{4.10}$$

where $\boldsymbol{\Gamma}^\top \in \mathbb{R}^{N\times|\mathcal{E}|}$ is graph incidence matrix defined in section 4.2.3. As a result, problem (4.2) becomes:

$$(\mathcal{P}_2) \quad \begin{array}{ll} \text{minimize}_{\boldsymbol{X}} & \frac{1}{2}\|\boldsymbol{S} - \boldsymbol{\mathcal{D}}\boldsymbol{X}\|_{\text{F}}^2 + \mu \sum_{k=1}^{N'} \|\boldsymbol{x}_{\lambda_k}\|_2 + \lambda\|\boldsymbol{\Gamma}(\boldsymbol{\mathcal{D}}\boldsymbol{X})^\top\|_1. \\ \text{subject to} & x_{ij} \geq 0 \quad \forall\, i,j \\ & \sum_{i=1}^{N'} x_{ij} = 1 \quad \forall\, j. \end{array} \tag{4.11}$$

To see the relevance of the nonlocal TV regularizer in (4.11), we rewrite it as follows:

$$\mathcal{J}_{\mathcal{G}_2}(\boldsymbol{\mathcal{D}}\boldsymbol{X}) = \sum_{i=1}^{N}\sum_{j=1}^{N}\mathcal{W}_{ij}\|\boldsymbol{\mathcal{D}}\boldsymbol{x}_i - \boldsymbol{\mathcal{D}}\boldsymbol{x}_j\|_1. \tag{4.12}$$

This quantity measures the discrepancies between the estimated spectra at all pairs of nodes weighted by their degree of similarity, where the discrepancies are measured in the ℓ_1-norm sense. The classical TV regularization is well known for promoting piecewise constant estimates while preserving discontinuities in the image. Nevertheless, the efficiency of TV depends upon the regularization parameter, i.e. for large values of the regularization parameter much of the detail can be removed. In contrast with classical TV, the advantage of nonlocal TV (4.10) is that the weights, which are proportional to the similarity between the two corresponding nodes, allow to strengthen the penalization over similar data and weaken it over dissimilar data. Finally, note that problem (4.2) can be interpreted as the combination of unmixing with the image filtering problem:

$$\text{minimize}_{\widehat{\boldsymbol{S}}} \quad \tfrac{1}{2}\|\boldsymbol{S} - \widehat{\boldsymbol{S}}\|_{\text{F}}^2 + \lambda\mathcal{J}_{\mathcal{G}_2}(\widehat{\boldsymbol{S}}), \tag{4.13}$$

which aims at finding a piecewise constant version $\widehat{\boldsymbol{S}}$ of $\boldsymbol{S}$.

4.4 ADMM algorithm

The ADMM is used in order to solve the two graph regularized optimization problems, namely problems (4.8) and (4.11).

4.4.1 Quadratic Laplacian regularization

For the first optimization problem (4.8), we consider the following variable splitting:

$$\begin{array}{ll} \text{minimize}_{\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z}} & \tfrac{1}{2}\|\boldsymbol{S} - \boldsymbol{\mathcal{D}}\boldsymbol{X}\|_{\text{F}}^2 + \lambda\text{trace}(\boldsymbol{Y}\mathcal{L}_{\mathcal{G}}\boldsymbol{Y}^\top) + \mu\sum_{k=1}^{N'}\|\boldsymbol{z}_{\lambda_k}\|_2 + \mathcal{I}_{\mathbb{R}_+^{N'\times N}}(\boldsymbol{Z}) \\ \text{subject to} & \boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} = \boldsymbol{\mathcal{C}} \\ & \boldsymbol{X} = \boldsymbol{Y} \end{array} \tag{4.14}$$

with

$$\boldsymbol{\mathcal{A}} = \begin{pmatrix} \boldsymbol{I}_{N'} \\ \boldsymbol{1}_{N'}^\top \end{pmatrix}, \ \boldsymbol{\mathcal{B}} = \begin{pmatrix} -\boldsymbol{I}_{N'} \\ \boldsymbol{0}_{N'}^\top \end{pmatrix}, \ \boldsymbol{\mathcal{C}} = \begin{pmatrix} \boldsymbol{0}_{N'\times N} \\ \boldsymbol{1}_{N}^\top \end{pmatrix},$$

where $\mathcal{I}_{\mathbb{R}_+^{N'\times N}}(\boldsymbol{Z})$ is the indicator of the positive orthant guarantying the positivity constraint, that is, $\mathcal{I}_{\mathbb{R}_+^{N'\times N}}(\boldsymbol{Z}) = 0$ if $\boldsymbol{Z} \succeq \boldsymbol{0}$ and $+\infty$ otherwise. The constraints impose the consensus $\boldsymbol{X} =$

$\boldsymbol{Y}$, $\boldsymbol{X} = \boldsymbol{Z}$, and the sum-to-one. In matrix form, the augmented Lagrangian for problem (4.14) is given by

$$\begin{aligned}\mathcal{L}_\rho(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{V}, \boldsymbol{\Lambda}) = \; & \tfrac{1}{2}\|\boldsymbol{S} - \boldsymbol{\mathcal{D}}\boldsymbol{X}\|_F^2 + \mu \textstyle\sum_{k=1}^{N'} \|\boldsymbol{z}_{\lambda_k}\|_2 + \mathcal{I}(\boldsymbol{Z}) + \lambda \text{trace}(\boldsymbol{Y}\mathcal{L}_\mathcal{G}\boldsymbol{Y}^\top) \\ & + \text{trace}(\boldsymbol{V}^\top(\boldsymbol{X} - \boldsymbol{Y})) + \tfrac{\rho}{2}\|\boldsymbol{X} - \boldsymbol{Y}\|_\text{F}^2 \\ & + \tfrac{\rho}{2}\|\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}\|_\text{F}^2 + \text{trace}(\boldsymbol{\Lambda}^\top(\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}))\end{aligned} \tag{4.15}$$

where $\boldsymbol{\Lambda}$ and $\boldsymbol{V}$ are the matrices of the Lagrange multipliers, and ρ is the penalty parameter. The flexibility of the ADMM lies in the fact that it splits the initial optimization problem into three subproblems. At iteration $k+1$, the ADMM algorithm is outlined by the following five sequential steps:

$$\begin{aligned}\boldsymbol{X}^{k+1} &= \text{minimize}_{\boldsymbol{X}}\, \mathcal{L}_\rho(\boldsymbol{X}, \boldsymbol{Y}^k, \boldsymbol{Z}^k, \boldsymbol{V}^k, \boldsymbol{\Lambda}^k), \\ \boldsymbol{Y}^{k+1} &= \text{minimize}_{\boldsymbol{Y}}\, \mathcal{L}_\rho(\boldsymbol{X}^{k+1}, \boldsymbol{Y}, \boldsymbol{Z}^k, \boldsymbol{V}^k, \boldsymbol{\Lambda}^k), \\ \boldsymbol{Z}^{k+1} &= \text{minimize}_{\boldsymbol{Z}}\, \mathcal{L}_\rho(\boldsymbol{X}^{k+1}, \boldsymbol{Y}^{k+1}, \boldsymbol{Z}, \boldsymbol{V}^k, \boldsymbol{\Lambda}^k), \\ \boldsymbol{V}^{k+1} &= \boldsymbol{V}^k + \rho(\boldsymbol{X}^{k+1} - \boldsymbol{Y}^{k+1}), \\ \boldsymbol{\Lambda}^{k+1} &= \boldsymbol{\Lambda}^k + \rho(\boldsymbol{\mathcal{A}}\boldsymbol{X}^{k+1} + \boldsymbol{\mathcal{B}}\boldsymbol{Z}^{k+1} - \boldsymbol{\mathcal{C}}).\end{aligned} \tag{4.16}$$

The ADMM steps are developed hereafter. Similarly to all the chapters, we drop the iteration index in the development of the minimization steps in order to keep the notations simple.

$\boldsymbol{X}$ minimization step

The augmented Lagrangian is quadratic with respect to $\boldsymbol{X}$. The minimizer has an analytical expression that is obtained by setting the gradient of the augmented Lagrangian with respect to $\boldsymbol{X}$ to zero:

$$\boldsymbol{X} = (\boldsymbol{\mathcal{D}}^\top\boldsymbol{\mathcal{D}} + \rho\boldsymbol{\mathcal{A}}^\top\boldsymbol{\mathcal{A}} + \rho\boldsymbol{I}_{N'})^{-1}(\boldsymbol{\mathcal{D}}^\top\boldsymbol{S} - \boldsymbol{\mathcal{A}}^\top[\boldsymbol{\Lambda} + \rho(\boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}})] - \boldsymbol{V} + \rho\boldsymbol{Y}). \tag{4.17}$$

$\boldsymbol{Y}$ minimization step

Similarly to the first step, the solution is obtained by setting the gradient of the augmented Lagrangian with respect to $\boldsymbol{Y}$ to zero, which yields:

$$\boldsymbol{Y} = (\boldsymbol{V} + \rho\boldsymbol{X})(2\lambda\mathcal{L}_\mathcal{G} + \rho\boldsymbol{I}_N)^{-1}. \tag{4.18}$$

Assume that we did not use $\boldsymbol{Y}$, and assigned the same ADMM variable $\boldsymbol{X}$ for both the fidelity term and the graph Laplacian regularization. In this case, the $\boldsymbol{X}$ minimization reduces to solving a Sylvester equation [Bartels and Stewart, 1972]. The exact solution of this problem can not be computed efficiently due to the high dimensionality of the problem. In fact it requires the

inversion of a $NN' \times NN'$ matrix where N and N' can be both very large. Iterative methods have been proposed to perform this task [Ding and Chen, 2005]. These iterative methods are similar to the first two steps of our ADMM solution in the sense that the initial variable is split into two variables and alternating updates of these variables are performed.

$\boldsymbol{Z}$ minimization step

After discarding the terms that are independent of $\boldsymbol{Z}$, the minimization of the augmented Lagrangian with respect to $\boldsymbol{Z}$ reduces to solving the following problem:

$$\begin{array}{ll} \text{minimize}_{\boldsymbol{Z}} & \mu \sum_{k=1}^{N'} \|\boldsymbol{z}_{\lambda_k}\|_2 + \text{trace}(\boldsymbol{\Lambda}^\top \boldsymbol{C}\boldsymbol{Z}) + \frac{\rho}{2}\|\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}\|_{\text{F}}^2 \\ \text{subject to} & \boldsymbol{Z} \succeq \boldsymbol{0}. \end{array} \tag{4.19}$$

This minimization step can be split into N' problems given the structure of matrices $\boldsymbol{\mathcal{A}}$, $\boldsymbol{\mathcal{B}}$ and $\boldsymbol{\mathcal{C}}$, one for each row of $\boldsymbol{Z}$, that is,

$$\text{minimize}_{\boldsymbol{z}} \quad \tfrac{1}{2}\|\boldsymbol{z} - \boldsymbol{v}\|_2^2 + \alpha\|\boldsymbol{z}\|_2 + \mathcal{I}(\boldsymbol{z}) \tag{4.20}$$

where $\boldsymbol{v} = \boldsymbol{x} + \rho^{-1}\boldsymbol{\lambda}$, $\alpha = \rho^{-1}\mu$. Vectors $\boldsymbol{\lambda}$, $\boldsymbol{x}$ and $\boldsymbol{z}$ correspond to a given row of $\boldsymbol{\Lambda}$, $\boldsymbol{X}$ and $\boldsymbol{Z}$, respectively. The minimization problem (4.20) admits a unique solution given by the proximity operator of function $f(\boldsymbol{z}) = \alpha\|\boldsymbol{z}\|_2 + \mathcal{I}(\boldsymbol{z})$:

$$\begin{cases} \boldsymbol{z}^* = \boldsymbol{0} & \text{if } \|(\boldsymbol{v})_+\|_2 < \alpha \\ \boldsymbol{z}^* = \left(1 - \frac{\alpha}{\|(\boldsymbol{v})_+\|_2}\right)(\boldsymbol{v})_+ & \text{otherwise,} \end{cases} \tag{4.21}$$

where $(\cdot)_+ = \max(\boldsymbol{0}, \cdot)$. Operator (4.27) was recently used in [Thiebaut et al., 2013, Ammanouil et al., 2014]. The derivation of this operator can be found in Appendix A.

Update of the Lagrange multipliers $\boldsymbol{\Lambda}$ and $\boldsymbol{V}$

The last step consists of updating the Lagrange multipliers $\boldsymbol{\Lambda}$ and $\boldsymbol{V}$ using the following expressions

$$\begin{aligned} \boldsymbol{\Lambda}^{k+1} &= \boldsymbol{\Lambda}^k + \rho(\boldsymbol{\mathcal{A}}\boldsymbol{X}^{k+1} + \boldsymbol{\mathcal{B}}\boldsymbol{Z}^{k+1} - \boldsymbol{\mathcal{C}}), \\ \boldsymbol{V}^{k+1} &= \boldsymbol{V}^k + \rho(\boldsymbol{X}^{k+1} - \boldsymbol{Y}^{k+1}). \end{aligned} \tag{4.22}$$

As suggested in [Boyd et al., 2011], a reasonable stopping criterion for this iterative algorithm is that the primal and dual residuals must be smaller than some tolerance thresholds.

4.4.2 Nonlocal TV regularization

In the case of the nonlocal TV regularized unmixing problem, in order to find the solution using an ADMM algorithm, we adopt the following variable splitting scheme:

$$\begin{array}{ll} \text{minimize}_{\boldsymbol{X},\boldsymbol{Y}_{1\to 2},\boldsymbol{Z}} & \frac{1}{2}\|\boldsymbol{S}-\boldsymbol{\mathcal{D}}\boldsymbol{X}\|_{\mathrm{F}}^2+\mu\sum_{k=1}^{N'}\|\boldsymbol{z}_{\lambda_k}\|_2+\mathcal{I}(\boldsymbol{Z})+\lambda\|\boldsymbol{Y}_2\|_1 \\ \text{subject to} & \boldsymbol{\mathcal{A}}\boldsymbol{X}+\boldsymbol{\mathcal{B}}\boldsymbol{Z}=\boldsymbol{\mathcal{C}} \\ & \boldsymbol{Y}_1=(\boldsymbol{\mathcal{D}}\boldsymbol{X})^\top \\ & \boldsymbol{Y}_2=\boldsymbol{\Gamma}\boldsymbol{Y}_1 \end{array} \tag{4.23}$$

where

$$\boldsymbol{\mathcal{A}}=\begin{pmatrix}\boldsymbol{I}_{N'}\\ \boldsymbol{1}_{N'}^\top\end{pmatrix},\ \boldsymbol{\mathcal{B}}=\begin{pmatrix}-\boldsymbol{I}_{N'}\\ \boldsymbol{0}_{N'}^\top\end{pmatrix},\ \boldsymbol{\mathcal{C}}=\begin{pmatrix}\boldsymbol{0}_{N'\times N}\\ \boldsymbol{1}_{N}^\top\end{pmatrix}.$$

$\boldsymbol{X},\boldsymbol{Y}_1,\boldsymbol{Y}_2,\boldsymbol{Z}$ are the ADMM variables. The constraints are imposed to ensure that problem (4.23) is equivalent to problem (4.11). The augmented Lagrangian for problem (4.23) is given by

$$\begin{aligned} \mathcal{L}_\rho(\boldsymbol{X},\boldsymbol{Y}_{1\to 2},\boldsymbol{Z},\boldsymbol{\Lambda},\boldsymbol{V}_{1\to 2}) = \ & \tfrac{1}{2}\|\boldsymbol{S}-\boldsymbol{\mathcal{D}}\boldsymbol{X}\|_{\mathrm{F}}^2+\mu\textstyle\sum_{k=1}^{N'}\|\boldsymbol{z}_{\lambda_k}\|_2+\mathcal{I}(\boldsymbol{Z})+\lambda\|\boldsymbol{Y}_2\|_1 \\ & +\mathrm{trace}(\boldsymbol{\Lambda}^\top(\boldsymbol{\mathcal{A}}\boldsymbol{X}+\boldsymbol{\mathcal{B}}\boldsymbol{Z}-\boldsymbol{\mathcal{C}}))+\mathrm{trace}(\boldsymbol{V}_1^\top(\boldsymbol{Y}_1-(\boldsymbol{\mathcal{D}}\boldsymbol{X})^\top)) \\ & +\mathrm{trace}(\boldsymbol{V}_2^\top(\boldsymbol{Y}_2-\boldsymbol{\Gamma}\boldsymbol{Y}_1))+\tfrac{\rho}{2}\|\boldsymbol{\mathcal{A}}\boldsymbol{X}+\boldsymbol{\mathcal{B}}\boldsymbol{Z}-\boldsymbol{\mathcal{C}}\|_F^2 \\ & +\tfrac{\rho}{2}\|\boldsymbol{Y}_1-(\boldsymbol{\mathcal{D}}\boldsymbol{X})^\top\|_F^2+\tfrac{\rho}{2}\|\boldsymbol{Y}_2-\boldsymbol{\Gamma}\boldsymbol{Y}_1\|_F^2 \end{aligned} \tag{4.24}$$

where $\boldsymbol{\Lambda},\boldsymbol{V}_1,\boldsymbol{V}_2$ are the Lagrange multipliers and ρ is the penalty parameter. The ADMM steps are developed hereafter:

$\boldsymbol{X}$ minimization step

After discarding the terms independent of $\boldsymbol{X}$ in the augmented Lagrangian, minimizing the augmented Lagrangian w.r.t. $\boldsymbol{X}$ reduces to a Least squares problem. The solution is obtained by solving a set of linear equations:

$$((1+\rho)\boldsymbol{\mathcal{D}}^\top\boldsymbol{\mathcal{D}}+\rho\boldsymbol{\mathcal{A}}^\top\boldsymbol{\mathcal{A}})\boldsymbol{X} = \boldsymbol{\mathcal{D}}^\top\boldsymbol{S}-\boldsymbol{\mathcal{A}}^\top\boldsymbol{\Lambda}+\boldsymbol{\mathcal{D}}^\top\boldsymbol{V}_1^\top-\rho\boldsymbol{\mathcal{A}}^\top(\boldsymbol{\mathcal{B}}\boldsymbol{Z}-\boldsymbol{\mathcal{C}})+\rho\boldsymbol{\mathcal{D}}^\top\boldsymbol{Y}_1^\top.$$

Note that the $\boldsymbol{X}$ minimization step is separable column wise, i.e. a linear system of equations is solved for every column of $\boldsymbol{X}$. However, other variable splitting choices could lead to a more complex Sylvester equation such as for example using one additional variable $\boldsymbol{Y}_1$ and letting $\boldsymbol{Y}_1=\boldsymbol{\Gamma}(\boldsymbol{\mathcal{D}}\boldsymbol{X})^\top$.

$\boldsymbol{Y}_1$ minimization step

Similarly to the first step, the $\boldsymbol{Y}_1$ minimization amounts to solving the following set of linear equations:

$$(\boldsymbol{I}_N + \boldsymbol{\Gamma}^\top\boldsymbol{\Gamma})\boldsymbol{Y}_1 = -\frac{1}{\rho}\boldsymbol{V}_1 + \frac{1}{\rho}\boldsymbol{\Gamma}^\top\boldsymbol{V}_2 + \boldsymbol{X}^\top\boldsymbol{\mathcal{D}}^\top + \boldsymbol{\Gamma}^\top\boldsymbol{Y}_2. \tag{4.25}$$

$\boldsymbol{Y}_2$ minimization step

Minimizing the augmented Lagrangian w.r.t. $\boldsymbol{Y}_2$ reduces to the well known lasso problem. The solution is obtained using soft thresholding:

$$\boldsymbol{Y}_2 = \text{soft}_{\frac{\lambda}{\rho}}(\boldsymbol{\Gamma}\boldsymbol{Y}_1 - \frac{1}{\rho}\boldsymbol{V}_2) \tag{4.26}$$

where $\text{soft}_\alpha(\cdot) = \text{sign}(\cdot)(|\cdot| - \alpha)_+$ is applied element wise and $(\cdot)_+ = \max(\boldsymbol{0}, \cdot)$.

***Z* minimization step**

Minimizing the augmented Lagrangian w.r.t. $\boldsymbol{Z}$ reduces to solving the positively constrained group lasso problem:

$$\begin{cases} \boldsymbol{z}^* = \boldsymbol{0} & \text{if } \|(\boldsymbol{v})_+\|_2 < \alpha \\ \boldsymbol{z}^* = \left(1 - \frac{\alpha}{\|(\boldsymbol{v})_+\|_2}\right)(\boldsymbol{v})_+ & \text{otherwise} \end{cases} \tag{4.27}$$

where $\boldsymbol{v} = \boldsymbol{x} + \rho^{-1}\boldsymbol{\lambda}$, $\alpha = \rho^{-1}\mu$, $\boldsymbol{\lambda}$, $\boldsymbol{x}$ and $\boldsymbol{z}$ correspond to a row in $\boldsymbol{\Lambda}$, $\boldsymbol{X}$ and $\boldsymbol{Z}$ respectively.

Update the Lagrange multipliers

The last step at each ADMM iteration consists of updating the Lagrange multipliers

$$\begin{aligned} \boldsymbol{\Lambda}^{k+1} &= \boldsymbol{\Lambda}^k + \rho(\boldsymbol{\mathcal{A}}\boldsymbol{X}^{k+1} + \boldsymbol{\mathcal{B}}\boldsymbol{Z}^{k+1} - \boldsymbol{\mathcal{C}}) \\ \boldsymbol{V}_1^{k+1} &= \boldsymbol{V}_1^k + \rho(\boldsymbol{Y}_1^{k+1} - (\boldsymbol{X}^{k+1})^\top) \\ \boldsymbol{V}_2^{k+1} &= \boldsymbol{V}_2^k + \rho(\boldsymbol{Y}_2^{k+1} - \boldsymbol{\Gamma}\boldsymbol{Y}_1^{k+1}). \end{aligned} \tag{4.28}$$

4.5 A note on complexity

Finally, we pay particular attention to the computational complexity of the resulting ADMM algorithms. In both algorithms, the most expensive step requires solving a linear system with N variables, N being very large in real images. More precisely, in the first ADMM algorithm (section 4.4.1), the most expensive step corresponds to the $\boldsymbol{Y}$ minimization step (4.18) where the linear system of equations depends on the Laplacian matrix. Whereas in the second ADMM

algorithm (section 4.4.2), this corresponds to the $\boldsymbol{Y}_1$ minimization step (4.25) where the linear system of equations depends on the incidence matrix. In fact, it is computationally expensive to solve the corresponding linear systems which have a computational complexity of $\mathrm{O}(N^3)$, and it is memory intensive to store the Laplacian and the incidence matrices themselves. In the first case, our approach for reducing both the storage and computational complexity is mostly related to the graph formulation of the problem. Whereas in the second case, it simply exploits the sparse structure of graph.

In the first case, recall that we have proposed to build a fully connected weighted graph, hence the Laplacian matrix is an $N \times N$ matrix which can easily outstrip the storage capacity of a computer. A solution is to partition the initial graph into k_c clusters while avoiding to computation and storage of the whole Laplacian matrix. This allows to deal with smaller matrices and solve smaller linear systems of equations with respect to each sub-graph. Following this idea, we propose to use the algorithm of [Ng et al., 2002] in order to partition the N nodes of the graph into k_c clusters of smaller size. This allows to approximately solve the $\boldsymbol{Y}$ minimization step (4.18) by solving k_c smaller linear systems, where the number of variables is now smaller than N. For more details regarding the clustering step see Appendix C. Note that the overall optimization problem (4.8) can not be similarly separated into k_c distinct problems due to the group lasso regularization. The purpose of this step is to reduce the computational complexity of (4.18) while preserving the global knowledge captured by the graph structure. For this reason, the segmentation must be conservative, in the sense that k_c should not be very large. In fact, a segmentation into too many clusters with strong connections between inter-cluster pixels could create undesirable cluster-like artifacts in the abundance maps.

In the second ADMM algorithm, we have proposed to build the graph based on the four neighbors and k nearest spectral neighbors. The same clustering strategy described in the previous section can be used in order to partition the graph into smaller clusters and hence reduce the computational complexity of the $\boldsymbol{Y}_1$ minimization step (4.25). Nevertheless, in this case one can exploit the fact that the graph structure is sparse in order to alleviate the storage and computational complexity especially for relatively small values of k. Being sparse, storing the incidence matrix allows to save memory. Furthermore it is computationally tractable to handle the resulting sparse linear system of equations using iterative methods. In the experiments, we use Matlab which internally solves the sparse linear system using a conjugate gradient based algorithm.

4.6 Experiments

In what follows, we present two sets of experiments in order to show the relevance of the proposed regularizations and the corresponding algorithms. The first set of experiments, in section 4.6.1, is dedicated to the quadratic Laplacian regularization. Whereas the second set of experiments, in section 4.6.2, is dedicated to the nonlocal TV regularization.

4.6.1 Experiments with the Quadratic Laplacian regularization

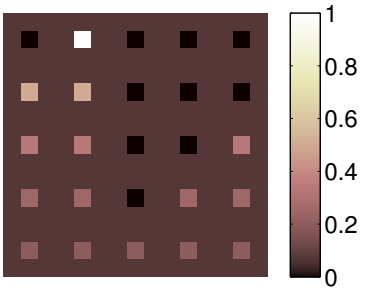
Simulations with synthetic data sets

The performance of the proposed approach was first evaluated using two simulated data sets, namely, Data1 and Data2 designed with different levels of homogeneity. Data1 is the same data set used in the experiments of [Iordache et al., 2012a, Chen et al., 2014]. The image consists of 75×75 pixels generated using 5 endmembers $[\boldsymbol{r}_1, \boldsymbol{r}_2, \cdots \boldsymbol{r}_5]$. The library $\boldsymbol{\mathcal{D}}$ used in all the experiments for this chapter contains in total 230 endmembers with 224 spectral bands extracted from the USGS spectral library of minerals. The background is a mixture of the 5 endmembers with the following abundance values $[0.1149\ 0.0741\ 0.2003\ 0.2055\ 0.4051]^\top$. There are 25 squares in the image disposed in a 5×5 grid fashion (see Figure 4.5). Each square is an homogeneous surface where its pixels have the same abundances. Most of the abundances in Data1 verify the assumption of local consistency. Data2 is generated similarly to Data1, except that it is created using 15 distinct endmembers, and the squares in each row are identical in the sense that they have the same abundances. In addition to local consistency, there exists distant homogeneous surfaces in Data2 that are identical. As a result a pixel has local similar neighbors and distant ones too.

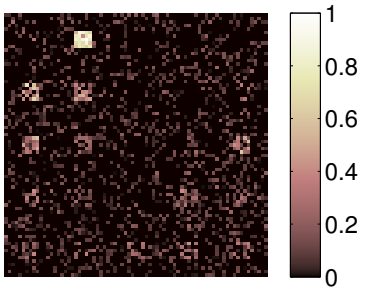
The first step in the proposed approach consists of defining the graph. We test the quadratic Laplacian regularization with a complete graph, i.e. there is an edge between all pairs of pixels. As for the weights in the affinity matrix, we simply threshold the square of the spectral distance and set the weights according to:

$$\begin{cases} \mathcal{W}_{ij} = 1 & \text{if } \|\boldsymbol{s}_i - \boldsymbol{s}_j\|_2^2 < d_{\text{min}}^2 \\ \mathcal{W}_{ij} = 0 & \text{otherwise,} \end{cases} \tag{4.29}$$

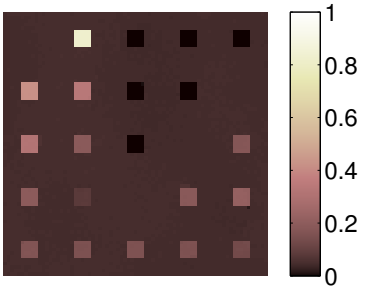
where d_{min}^2 represents the maximum squared spectral distance required in order to consider that two pixels are similar. Note that having a zero weight ($\mathcal{W}_{ij} = 0$) is equivalent to removing the corresponding edge. As previously explained in Section 4.2, there are different heuristics for choosing the weights. Setting the weights according to (4.29) was sufficient in our experiments



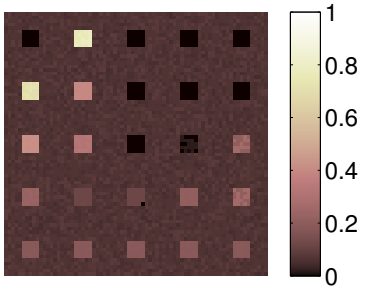
(a) $\boldsymbol{r}_2$: True



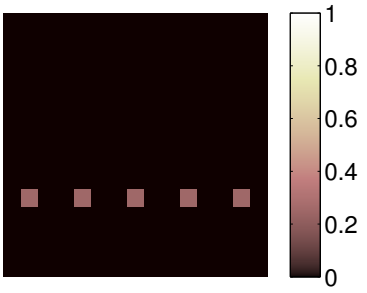
(b) $\boldsymbol{r}_2$: FCLS



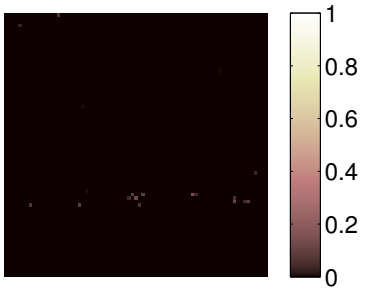
(c) $\boldsymbol{r}_2$: SUnSAL-TV



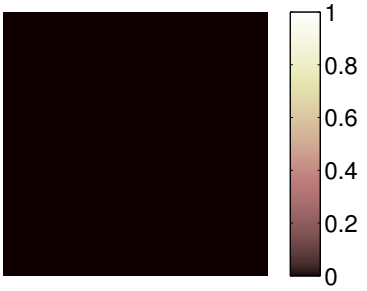
(d) $\boldsymbol{r}_2$: GLUP-Lap



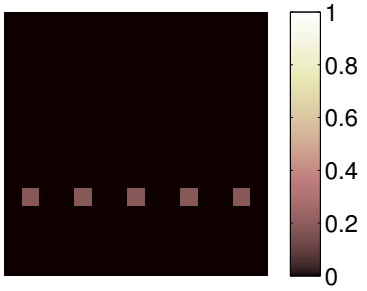
(e) $\boldsymbol{r}_7$: True



(f) $\boldsymbol{r}_7$: FCLS



(g) $\boldsymbol{r}_7$: SUnSAL-TV



(h) $\boldsymbol{r}_7$: GLUP-Lap

Figure 4.5: First and second rows: Abundance maps for endmember 2 in Data1 obtained with SNR = 30 dB. Third and fourth rows: Abundance maps for endmember 7 in Data2 obtained with SNR = 30 dB. The parameters are the reported in Table 1.

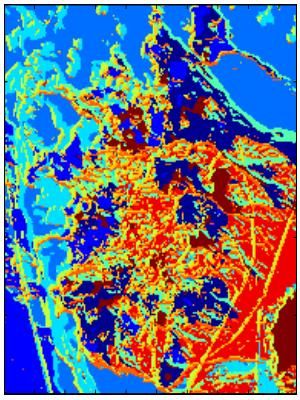


Figure 4.6: Clustering map used for Cuprite obtained with 10 clusters.

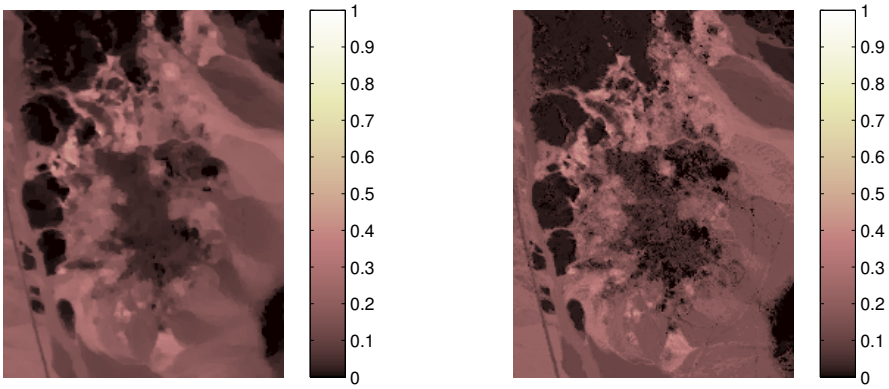


Figure 4.7: Abundance maps for Alunite obtained using from left to right SUnSAL-TV and the proposed GLUP-Lap algorithms.

to demonstrate the effectiveness of the method. The algorithm described in Appendix C is then used to cut the graph into 10 disjoint subgraphs in order to reduce the computational complexity of the $\boldsymbol{Y}$-minimization step as explained in section 4.5.

We compared the performances of FCLS [Heinz and Chang, 2001] and SUnSAL-TV [Iordache et al., 2012a] with the proposed approach denoted by GLUP-Lap (Group Lasso with Unit sum, Positivity constraints and graph Laplacian regularization). We used the Root Mean Square Error $\text{RMSE}_{\boldsymbol{X}} = \sqrt{\frac{1}{NN'} \times \|\widehat{\boldsymbol{X}} - \boldsymbol{X}\|_F^2}$ as the evaluation metric. We tested SUnSAL-TV and GLUP-Lap for different combinations of the sparsity and the spatial tuning parameters μ and λ. Table 4.2 reports the best performance of each algorithm for a given data set and a given SNR with the corresponding optimal pair of regularization parameters. GLUP-Lap requires the tuning of an additional parameter d_{min}^2 which is also reported in the table. Both, SUnSAL-TV and GLUP-Lap, outperformed FCLS. GLUP-Lap had the lowest RMSE for all cases. As the SNR increases the rate at which GLUP-Lap improves with respect to FCLS increases. This is due to the fact

that the observations contain less noise, thus the adjacency matrix becomes more reliable. The simulations performed with Data2 show that this data set is more difficult than the previous since it contains a large number of endmembers: 15 compared to 5 in Data1. As before, GLUP-Lap outperformed FCLS and SUnSAL-TV. It is important to note that GLUP-Lap and SUnSAL-TV were run under the same ADMM conditions. The penalty parameter was set to 0.05, and the maximum number of iterations to 200 in both algorithms. The first fours abundance maps in Figure 4.5 show the true abundance map of the second endmember in Data1, and the estimated maps obtained with FCLS, SUnSAL-TV and GLUP-Lap with SNR=30dB. It can be seen from these maps that both SUnSAL-TV and GLUP-Lap estimated smooth abundance maps compared to FCLS, with SUnSAL-TV having the smoothest map. However, the squares that were not correctly estimated by SUnSAL-TV were better estimated with GLUP-Lap. This is possibly due to the fact that the pixels in these squares were encouraged by the nonlocal graph to have similar estimates and thus appeared as consistent blocks in the GLUP-Lap result. The same observation can be made for the abundance map of $\boldsymbol{r}_7$ in Data2. This endmember is only present in squares of the fourth row. The last four abundance maps in Figure 4.8 (third and fourth row) show that FCLS was not able to correctly estimate the abundance of this endmember. SUnSAL-TV, possibly due to the links a pixel has with its surrounding pixels, also failed to correctly estimate its abundances. Despite the difficulty of this abundance map, GLUP-Lap perfectly recovered the abundances. Even if the 5 squares are separated, their pixels are possibly connected in the graph and collaboratively estimate their abundances.

Simulations real data set

We also tested the proposed approach on a subset of the Cuprite scene [1] provided by the AVIRIS spectrometer. This scene was captured over a mining district in Nevada, the subset we use has 191×250 pixels and 188 spectral bands over the wavelength interval $400 - 2500$ nm. Figure 4.6 shows the clustering map that was used to partition the image into smaller sub-graphs. Figure 4.7 shows the abundance maps of Alunite obtained using SUnSAL-TV and GLUP-Lap respectively. In the former case, μ and η were both set to 10^{-3} and the execution time was 1281 seconds. In the latter case, μ and η were set to 10^{-3} and 0.1 respectively, d^2_{min} was set to 0.005, and the execution time was 5275 seconds. It can be seen from Figure 4.7 that the abundance maps estimated by GLUP-Lap are less smooth than SUnSAL-TV. However, they conserved relatively more details.

[1] available at http://www.ehu.eus/ccwintco

Table 4.2: RMSE obtained with different values of the SNR, with the optimal values of the couple (μ; λ) for SUnSAL-TV and GLUP-Lap, the penalty parameter was set to $\rho = 0.05$ for both algorithms.

	SNR 20 dB	SNR 30 dB	SNR 40 dB
		Data1	
FCLS	0.0262	0.0173	0.0101
SUnSAL-TV	0.0156 (0.05; 0.05)	0.0075 (5 10^{-3}; 0.01)	0.0034 (10^{-3}; 5 10^{-3})
GLUP-Lap	0.0152 (0.01; 0.5) $d^2_{min} = 2.5$	0.0049 (5 10^{-4}; 0.5) $d^2_{min} = 0.3$	0.0012 (5 10^{-5}; 0.5) $d^2_{min} = 0.05$
		Data2	
FCLS	0.0307	0.0240	0.0151
SUnSAL-TV	0.0250 (0.05; 0.3)	0.0132 (10^{-4}; 0.005)	0.0073 (5 10^{-5}; 10^{-3})
GLUP-Lap	0.0174 (0.01; 1) $d^2_{min} = 1.8$	0.0078 (10^{-4}; 1) $d^2_{min} = 0.5$	0.0023 (5 10^{-5}; 1) $d^2_{min} = 0.5$

4.6.2 Experiments with nonlocal TV regularization

Simulations with synthetic data sets

Similarly to the first set of experiments, the performance of the proposed approach was first evaluated using Data1. However, this time, as explained in section 4.2, we define the edge set of the graph by connecting a pixel to its four neighbors and to its 10 nearest neighbors ($k_c = 10$) where the spectral distance is measured with the ℓ_2-norm. We then use binary weights according to (4.29). The proposed approach with the nonlocal TV regularization is denoted by GLUP TV$_{\mathcal{G}}$ in the experiments. Its performance is compared with FCLS, SUnSAL TV [Iordache et al., 2012a] and a TV regularized Collaborative unmixing [Iordache et al., 2013]. The TV regularized Collaborative unmixing is obtained by setting $\boldsymbol{Y}_1 = \boldsymbol{X}^\top$ in the second constraint of (4.23), $k_c = 0$ i.e. no nonlocal neighbors, and $d^2_{\text{min}} = \infty$ in (4.29) which finally amounts to using a regular 4 neighborhood graph. Furthermore, we tested the same graph structure used in GLUP TV$_{\mathcal{G}}$ directly on the abundances rather than on the reconstructed spectra, i.e. using $\mathcal{J}_{\mathcal{G}_2}(\boldsymbol{X})$ rather than $\mathcal{J}_{\mathcal{G}_2}(\boldsymbol{\mathcal{D}X})$. This approach is denoted as GLUP TV$_{\mathcal{G}}$ $(*)$ in the table. The ADMM penalty parameter was set to 0.05, and the maximum number of iterations to 200 in all ADMM based algorithms. Table 4.3 reports the best scores for the RMSE between the true and the estimated

abundance matrix ($\text{RMSE}_{\boldsymbol{X}}$) with the corresponding optimal pairs of regularization parameters. Nonlocal TV requires tuning an additional parameter d^2_{min} which is also reported in the table. All TV approaches outperformed FCLS. Nonlocal TV had the lowest RMSE for all cases. Figure 4.8 shows the true abundance map of endmember $\boldsymbol{e}_1$, and the estimated maps obtained with Collaborative TV, SUnSAL TV and GLUP $\text{TV}_{\mathcal{G}}$ with an SNR equal to 30 dB. It can be seen from these maps that both TV approaches and the proposed nonlocal TV estimated smooth abundance maps. However, the proposed approach was able to better recover the abundances of the squares in the second column. This is possibly due to the fact that the pixels in a square are connected to each other and disconnected from the background. This has prevented smoothing over the squares and making them disappear.

Finally Figure 4.9 shows the RMSE as a function of λ for the optimal μ in the case of the synthetic data set for a SNR of 30 dB. This figure shows that both Graph $\text{TV}_{\mathcal{G}}$ and Graph $\text{TV}_{\mathcal{G}}$ $(*)$ had the best results compared to the other methods showing the advantage of having a nonlocal graph adapted to the image rather than a 4 neighborhood graph as in classical TV. Furthermore, it shows that when the tuning parameter increased, Graph $\text{TV}_{\mathcal{G}}$ $(*)$ maintained a lower RMSE compared to Graph $\text{TV}_{\mathcal{G}}$. In other words, it shows that with high values of the tuning parameter, imposing the TV on the abundances gave better results than in the case where it is imposed on the reconstructed spectra.

Simulations with real data set

We also tested the proposed approach using real hyperspectral data, namely the Cuprite scene provided by NASA AVIRIS imaging spectrometer. Figure 4.10 shows the abundances estimated using SUnSAL TV and the proposed approach for two endmembers. The tuning parameters λ and μ were both set to 10^{-3} for SUnSAL TV [Iordache et al., 2012a] and to 5×10^{-3} for the proposed algorithm, d^2_{min} was set to 2.5. From the two endmember abundance maps, it can be seen that TV provided smoother results. However, the proposed approach was able to preserve relatively more details.

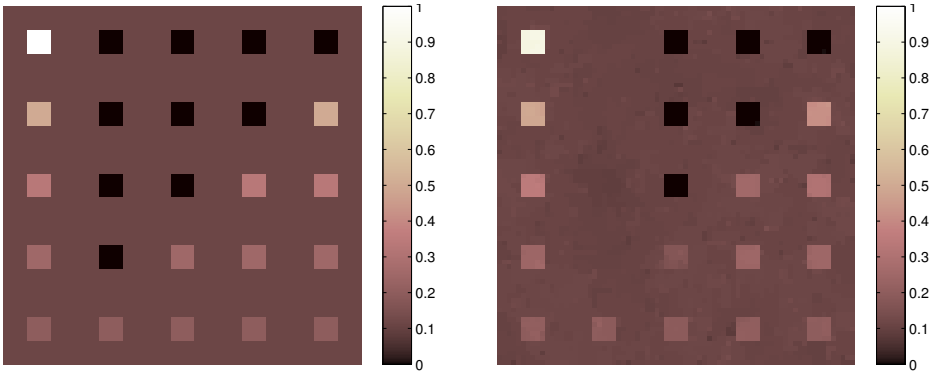
4.7 Conclusion

In this chapter, we incorporated two Graph based regularizations within the unmixing problem. The proposed regularizations take into account the spatial information and the spectral correlation inherently present in hyperspectral images. This prior was expressed through the con-

Table 4.3: RMSE obtained with different values of SNR, with the optimal couples of tuning parameters (μ; λ), ρ being set to 0.05.

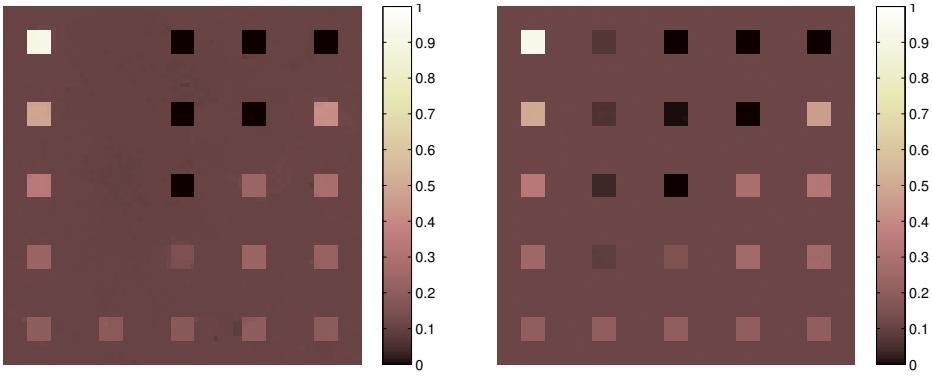
	SNR 20 dB	SNR 30 dB	SNR 40 dB
FCLS	0.0262	0.0173	0.0101
SUnSAL TV	0.0156 (0.05; 0.05)	0.0075 (5×10^{-3}; 0.01)	0.0034 (10^{-3}; 5×10^{-3})
Collaborative TV	0.0151 (0.5; 0.05)	0.0071 (0.1; 0.01)	0.0028 (0.1; 5 10^{-3})
GLUP $\text{TV}_{\mathcal{G}}$	0.0101 (0.3; 0.01) $d^2_{min} = 2.5$	0.0028 (0.1; 0.005) $d^2_{min} = 0.3$	0.0010 (0.05; 10^{-3}) $d^2_{min} = 0.05$
GLUP $\text{TV}_{\mathcal{G}}$ $(*)$	0.0128 (0.3; 0.05) $d^2_{min} = 2.5$	0.0037 (0.1; 0.05) $d^2_{min} = 0.3$	0.0014 (0.05; 5×10^{-3}) $d^2_{min} = 0.05$

struction of a weighted graph adapted to the image under investigation. We tested two graphs, a complete graph and a sparse graph which only considers the 4 neighbors and k_c nearest neighbors in for each pixel. We also tested two regularizations, namely the quadratic Laplacian regularization and the nonlocal TV regularization, that impose smooth and piece wise constant estimates respectively. Furthermore, the regularizations were applied on the abundances and on the reconstructed spectra. The resulting optimization problems were solved using the ADMM algorithm which allowed to split the original problem into smaller problems, and most importantly avoid solving more complex sylvester equations. A special attention was given to the computational complexity of the proposed algorithms. In the experiments, both regularizations provided better estimates for the abundance maps and preserved much of the details due to the adapted structure of the graph which modulates the strength of the regularization. Furthermore, we showed in the experiments that the nonlocal graph structure creates more consistent areas at the local and global level. In the next chapter, we shift to the case of supervised nonlinear unmixing. In particular, we exploit tools from vector-valued reproducing kernel Hilbert spaces (RKHS). We will see that the kernel design also allow to impose structured regularization on the nonlinear contributions at different bands based on a graph representations where each node represents a certain band.



(a) True

(b) Collaborative TV



(c) SUnSAL TV

(d) GLUP $TV_{\mathcal{G}}$

Figure 4.8: Abundance maps for endmember 1 in the synthetic data set obtained with SNR = 30 dB. The optimal parameters are reported in Table 1

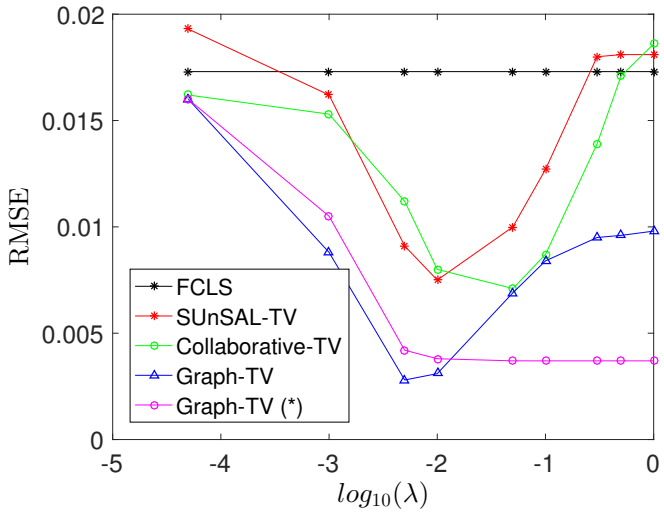
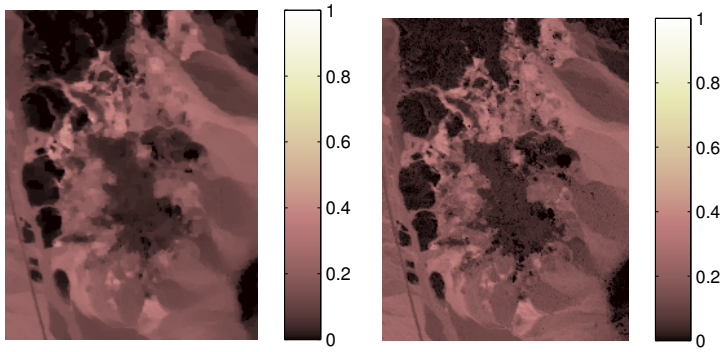
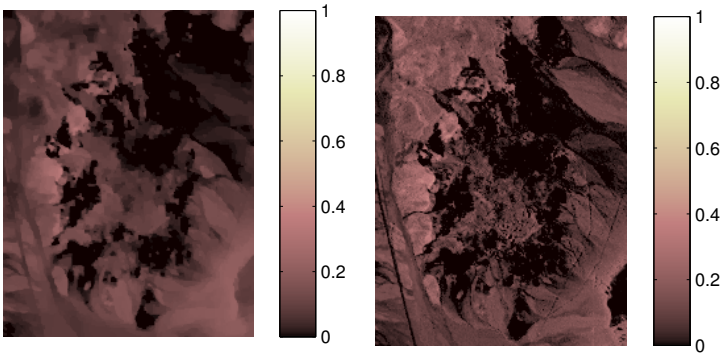


Figure 4.9: RMSE between the true abundances and the estimated abundances for different values of the spatial regularization.



(a) SUnSAL TV (b) GLUP TV$_{\mathcal{G}}$



(c) SUnSAL TV (d) GLUP TV$_{\mathcal{G}}$

Figure 4.10: Abundance maps for two endmembers in Cuprite obtained using SUnSAL TV and GLUP TV$_{\mathcal{G}}$.

Chapter 5

Supervised nonlinear unmixing with vector-valued kernel functions

This chapter has been adapted from the journal paper [Ammanouil et al., 2016a] (under revision).

This chapter presents a kernel based nonlinear mixing model for hyperspectral data where the nonlinear function belongs to a Hilbert space of vector valued functions. The proposed model extends existing ones by accounting for band dependent and neighboring nonlinear contributions. The key idea is to work under the assumption that nonlinear contributions are dominant in some parts of the spectrum while they are less pronounced in other parts. In addition to this, we motivate the need for taking into account nonlinear contributions originating from the ground covers of neighboring pixels by practical considerations, precisely the adjacency effect. The relevance of the proposed model is that the nonlinear function is associated to a matrix valued kernel that allows to jointly model a wide range of nonlinearities and include prior information regarding band dependencies. Furthermore, the choice of the nonlinear function input allows to incorporate neighboring effects. The optimization problem is strictly convex and the corresponding iterative algorithm is based on the alternating direction method of multipliers (ADMM). Finally, experiments conducted using synthetic and real data demonstrate the effectiveness of the proposed approach.

5.1 Introduction

The work presented in chapter 3 was devoted to unsupervised linear unmixing. In this chapter, we shift our attention to supervised nonlinear unmixing. For self containment, we first briefly recall the LMM and the corresponding notations. According to the LMM, the spectrum associated with a mixed pixel is a linear combination of the endmembers spectra [Adams et al., 1986, Keshava and Mustard, 2002, Bioucas-Dias et al., 2012]. More formally, we have:

$$\boldsymbol{s}_n = \sum_{i=1}^{M} a_{i,n}\boldsymbol{r}_i + \boldsymbol{e}_n, \forall\, n = 1, \dots N, \tag{5.1}$$

where $\boldsymbol{s}_n$ is the L-dimensional spectrum of the n-th pixel, L is the number of frequency bands, M denotes the number of endmembers, $a_{i,n}$ is the abundance of the i-th endmember in the n-th pixel, $\boldsymbol{r}_i$ is the L-dimensional spectrum of the i-th endmember, $\boldsymbol{e}_n$ is a vector of Gaussian white noise, and N is the number of observations. All vectors are column vectors. The abundances, which represent the relative contributions of the endmembers [Heinz and Chang, 2001], are positive and usually sum to one: $a_{i,n} \geq 0$ and $\sum_{i=1}^{M} a_{i,n} = 1$ respectively. The LMM (5.1) is a simplified spectral mixing model. It only considers light reaching the sensor that has interacted once with the imaged surface, and neglects complex interactions between light, the imaged surface, neighboring surfaces and the atmosphere.

More recently, there has been a considerable amount of studies devoted to nonlinear mixing models [Dobigeon et al., 2014b, Heylen et al., 2014a]. In particular, bilinear models are among the most widely known to account for nonlinear effects [Nascimento and Bioucas-Dias, 2009, Fan et al., 2009, Halimi et al., 2011, Altmann et al., 2014]. The physical assumption underlying these models is that light beams go through multiple reflections before reaching the sensor, mainly due to the three dimensionality of real scenes and scattering in the atmosphere [Altmann et al., 2012, Dobigeon et al., 2014a, Meganem et al., 2014]. The mathematical expression established for multiple reflections is the term by term product of two reflectance vectors in the case of bilinear models, and more than two reflectance vectors in the case of multilinear models [Heylen and Scheunders, 2016]. For example, the polynomial post nonlinear mixing model (PPNM) introduced in [Altmann et al., 2012] considers bilinear contributions through the following formulation:

$$\boldsymbol{s}_n = \sum_{i=1}^{M} a_{i,n}\boldsymbol{r}_i + u_n\Big(\sum_{i=1}^{M} a_{i,n}\boldsymbol{r}_i\Big) \odot \Big(\sum_{i=1}^{M} a_{i,n}\boldsymbol{r}_i\Big) + \boldsymbol{e}_n, \tag{5.2}$$

where u_n is the nonlinearity parameter, and $\odot$ denotes the Hadamard (element wise) product. On the right hand side of equation (5.2), the first term corresponds to the linear mixture and the second term corresponds to the nonlinear (bilinear) one. Another way for modeling the

nonlinear mixtures of the endmembers is through the use of nonlinear (scalar-valued) functions in reproducing kernel Hilbert spaces [Aronszajn, 1950, Shawe-Taylor and Cristianini, 2004]. The advantage of kernel-based models over models similar to (5.2) is that they are non parametric which means that they do not impose a predetermined form for the nonlinearity. For example, the authors of [Chen et al., 2013] propose the following model known as the Khype model:

$$\boldsymbol{s}_n = \sum_{i=1}^{M} a_{i,n}\boldsymbol{r}_i + \boldsymbol{g}_n(\boldsymbol{R}) + \boldsymbol{e}_n, \tag{5.3}$$

where $\boldsymbol{g}_n(\boldsymbol{R}) = [g_n(\boldsymbol{r}_{\lambda_1}) \dots g_n(\boldsymbol{r}_{\lambda_L})]^\top$, $g_n(\cdot)$ is a nonlinear function in a reproducing kernel Hilbert space (RKHS), and $\boldsymbol{r}_{\lambda_i}$ denotes the i-th row in $\boldsymbol{R}$. The authors of [Chen et al., 2013] show that model (5.3) is able to incorporate bilinear, multilinear as well as more complex nonlinear interactions between the endmembers depending on the kernel choice. The main drawback of both, bilinear and (scalar-valued) kernel based models, is that they impose the same function at all bands, which can be restrictive in practice. More precisely, bilinear models consider the same amount of bilinear contributions at all bands. In the case of the PPNM (5.2), the same weight $u_n a_{i,n} a_{j,n}$ is used to scale the bilinear contribution of $\boldsymbol{r}_i \odot \boldsymbol{r}_j$ at all bands. Similarly, the kernel-based model (5.3) considers the same scalar-valued nonlinear function $g_n(\cdot)$ at all bands. In contrast with the aforementioned models, the authors of [Yokoya et al., 2014, Févotte and Dobigeon, 2015] do not impose any analytical form for the nonlinear term. The nonlinear contribution is merely treated as a positive residual term that is sparsely (rarely) present among the observations [Févotte and Dobigeon, 2015]. Nevertheless, this model-free approach can be limiting itself since it does not control the nonlinear expression, hence it can prevent accurate estimations of the nonlinear contribution.

The nonlinear mixing model proposed in this chapter circumvents the drawbacks of the previously cited models by assuming that the nonlinear function belongs to a reproducing kernel Hilbert space (RKHS) of vector-valued functions. This approach improves upon the case of RKHS of scalar-valued functions by allowing for variable nonlinear contributions at different bands. The key idea is to work under the assumption that nonlinear contributions can be dominant in some parts of the spectrum and less significant in other parts [Somers et al., 2009, Richter et al., 2006, Tanre et al., 1987]. Unlike the scalar valued case where the same function is considered at each band, the proposed model relaxes this constraint, and allows to take into account wavelength dependent nonlinear contributions. In particular, we focus on RKHS associated with a special type of kernels, namely separable kernels. This type of kernels jointly defines the form of the nonlinear contribution, and allows to include prior information regarding the similarity between the nonlinear contributions at different wavelength bands. Similarly to the PPNM model, the

input of the nonlinear vector-valued function includes the linear mixture spectra present in the pixel. We go one step further by also including the spectra of the linear mixtures in neighboring pixels. This choice is motivated by the adjacency effect [Richter et al., 2006] which states that solar radiation reflected off adjacent surfaces can be scattered into the sensor's instantaneous field of view. Figure 5.1 shows two forms of the adjacency effect, and depicts how neighboring surfaces can nonlinearly contribute to the reflectance vector estimated for a pixel. The adjacency effects are usually removed in a pre-processing step known as atmospheric correction. There exists different empirical methods for atmospheric correction [Burazerovic et al., 2013, Tanre et al., 1987, Richter et al., 2006]. Nevertheless, the validity of some of these methods to correct the adjacency effect is still questionnable [Liang et al., 2001], and errors occurred by these methods can damage the quality of information extracted from remote sensing data [Hadjimitsis et al., 2010]. As a result, accounting for potential adjacency effects through the input of the nonlinear function increases the mixing model accuracy.

The chapter is organized as follows. Section 5.2 describes the nonlinear mixing model, and discusses approaches for constructing the matrix valued kernel, section 5.3 develops the optimization problem and the corresponding estimation algorithm. Finally, section 5.4 validates the proposed mixing model using synthetic and real data.

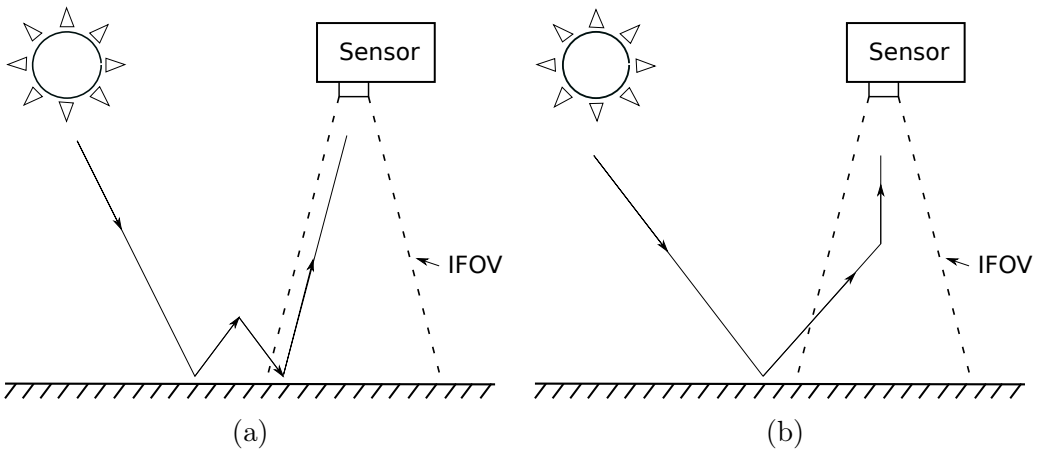


Figure 5.1: Illustration of two forms of the adjacency effect resulting from: (a) multiple reflections involving the targeted surface and an adjacent surface, (b) reflection off an adjacent surface that is then scattered in the atmosphere into the sensor's instantaneous field of view (IFOV).

Table 5.1: Notations for chapter 5

$\boldsymbol{\mathcal{C}}_n$	$c \times 1$	Indexes of the n-th pixel's neighbours including n
$\boldsymbol{s}_n^{\text{lin}}$	$L \times 1$	Spectra of the linear part in $\boldsymbol{s}_n$
$\tilde{\boldsymbol{v}}_n$	$cL \times 1$	Spectra of the linear part in the n-th pixel's neighbours
$\boldsymbol{v}_n$	$cL \times 1$	Observed spectra of the n-th pixel's neighbours
$g_n(\boldsymbol{r}_{\lambda_\ell})$	1×1	Nonlinear contribution in n-th pixel, ℓ-th band (used in Khype)
$\boldsymbol{f}(\boldsymbol{v}_n)$	$L \times 1$	Nonlinear contribution in n-th pixel, all bands (used in prop. model)
$f_\ell(\boldsymbol{v}_n)$	1×1	ℓ-th component of $\boldsymbol{f}(\boldsymbol{v}_n)$
$\widetilde{\boldsymbol{k}}(\boldsymbol{v}_n, \boldsymbol{v}_{n'})$	$L \times L$	Matrix-valued Kernel
$k(\boldsymbol{v}_n, \boldsymbol{v}_{n'})$	1×1	Scalar kernel (used in separable kernel design)
$\widetilde{\boldsymbol{K}}$	$LN \times LN$	Gram matrix associated with $\widetilde{\boldsymbol{k}}(\boldsymbol{v}_n, \boldsymbol{v}_{n'})$
$\boldsymbol{K}$	$N \times N$	Gram matrix associated with $k(\boldsymbol{v}_n, \boldsymbol{v}_{n'})$
$\widetilde{\mathcal{H}}_{\boldsymbol{k}}$	–	RKHS associated with $\widetilde{\boldsymbol{k}}$ ($\boldsymbol{f} \in \widetilde{\mathcal{H}}_{\boldsymbol{k}}$)
$\mathcal{H}_k$	–	RKHS associated with k ($f_\ell \in \mathcal{H}_k$)
$\boldsymbol{T}_\ell$	$cL \times cL$	Operator that extracts the c reflectance values at the ℓ-th band in $\boldsymbol{v}_n$
$\widetilde{\boldsymbol{\mathcal{E}}}$	$L \times L$	Positive semi-definite matrix (used in separable kernel design)
$\widetilde{\boldsymbol{\mathcal{W}}}$	$L \times L$	Adjacency matrix of the graph (representing the spectral bands)

5.2 Nonlinear mixing model

5.2.1 Model Description

First, we assume that the image is partitioned into patches or groups of pixels, and that the pixels in each patch are associated with a vector-valued nonlinear function. For ease of notations, assume that the available observations $\boldsymbol{S} = [\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N]$ belong to the same patch and that they are associated with the function $\boldsymbol{f}$. The proposed nonlinear model decomposes the spectrum of a pixel into the sum of a linear and nonlinear part:

$$\boldsymbol{s}_n = \boldsymbol{s}_n^{\text{lin}} + \boldsymbol{f}(\tilde{\boldsymbol{v}}_n) + \boldsymbol{e}_n, \tag{5.4}$$

where $\boldsymbol{s}_n^{\text{lin}} = \sum_{i=1}^{M} a_{i,n} \boldsymbol{r}_i$, $\tilde{\boldsymbol{v}}_n = \text{col}(\{\boldsymbol{s}_i^{\text{lin}}\}_{i \in \boldsymbol{\mathcal{C}}_n})$, $\text{col}(\cdot)$ is an operator that stacks its arguments on top of each other, and $\boldsymbol{\mathcal{C}}_n$ is a set of c pixels indices including n and $c-1$ of its neighboring pixels indices (for example $\boldsymbol{\mathcal{C}}_n = \{n, n-1, n+1\}$). The nonlinear contribution in (5.4) is expressed in terms of the pixel and its neighbors linear mixtures. This is in accordance with several bilinear models such as the post polynomial nonlinear mixing (PPNM) model [Altmann

et al., 2014] which expresses the nonlinear contribution solely in terms of $\boldsymbol{s}_n^{\text{lin}}$. The inconvenience of model (5.4) is that $\tilde{\boldsymbol{v}}_n$ depends on the unknown abundances. As a result, the corresponding optimization problem is not convex since the function $\boldsymbol{f}$ is unknown itself. In order to have a convex optimization problem, we propose to approximate $\boldsymbol{s}_i^{\text{lin}}$ by $\boldsymbol{s}_i$, hence equation (5.4) becomes:

$$\boldsymbol{s}_n \quad = \boldsymbol{s}_n^{\text{lin}} + \boldsymbol{f}(\boldsymbol{v}_n) + \boldsymbol{e}_n, \tag{5.5}$$

where $\boldsymbol{v}_n = \text{col}(\{\boldsymbol{s}_i\}_{i \in \mathcal{C}_n})$, and it is assumed that $\boldsymbol{f}(\boldsymbol{v}_n)$ and $\boldsymbol{e}_n$ are small compared to $\boldsymbol{s}_n^{\text{lin}}$. The latter assumption holds provided that the signal to noise ratio (SNR) is relatively high and that the linear part dominates the nonlinear one. The vector-valued approach offers an elegant and flexible way to jointly estimate multiple nonlinear functions at all bands since $\boldsymbol{f}(\boldsymbol{v}_n)$ implicitly corresponds to having different scalar-valued functions per band, i.e. $\boldsymbol{f}(\boldsymbol{v}_n) = [f_1(\boldsymbol{v}_n) \dots f_L(\boldsymbol{v}_n)]^\top$. As mentioned previously, the same nonlinear function is associated with all the pixels in the corresponding patch. It is important to note that the patch should contain enough pixels in order to have a good estimation of the nonlinear function. Moreover, it should be small enough to reflect the variability of the nonlinear function in the different regions of the image.

5.2.2 RKHS of vector-valued functions

The nonlinear function $\boldsymbol{f}$ used in (5.5) is a vector-valued function, its evaluation is a vector with L components representing the nonlinear contributions present in $\boldsymbol{s}_n$ at each band:

$$\begin{array}{rccc} \boldsymbol{f}: & \mathbb{R}^{Lc} & \rightarrow & \mathbb{R}^L \\ & \boldsymbol{v}_n & \rightarrow & \boldsymbol{f}(\boldsymbol{v}_n). \end{array} \tag{5.6}$$

As mentioned previously, we assume that $\boldsymbol{f}$ belongs to $\widetilde{\mathcal{H}}_{\boldsymbol{k}}$, a RKHS of vector-valued functions associated with the following kernel function:

$$\begin{array}{rccc} \widetilde{\boldsymbol{k}}: & \mathbb{R}^{Lc} \times \mathbb{R}^{Lc} & \rightarrow & \mathbb{R}^{L \times L} \\ & (\boldsymbol{v}_n, \boldsymbol{v}_{n'}) & \rightarrow & \widetilde{\boldsymbol{k}}(\boldsymbol{v}_n, \boldsymbol{v}_{n'}). \end{array} \tag{5.7}$$

Unlike the scalar-valued case, the kernel is a positive semi-definite matrix in $\mathbb{R}^{L \times L}$ rather than a positive scalar value. The overall Gram matrix $\widetilde{\boldsymbol{K}}$ associated with the function $\boldsymbol{f}$ is the matrix obtained from the evaluation of the kernel function (5.7) at all observation couples. It is a block matrix, such that the block indexed by (n, n') is given by:

$$\widetilde{\boldsymbol{k}}_{n,n'} = \widetilde{\boldsymbol{k}}(\boldsymbol{v}_n, \boldsymbol{v}_{n'}). \tag{5.8}$$

The Gram matrix consists of $N \times N$ blocks, where each block is an $L \times L$ matrix as defined in (5.8). As a result, $\widetilde{\boldsymbol{K}}$ is an $LN \times LN$ matrix. The representer theorem for vector-valued functions parallels the theorem in the scalar valued case. According to [Micchelli and Pontil, 2005], $\boldsymbol{f}(\boldsymbol{v}_n)$ can be expressed as an expansion of the kernel function over all training points:

$$\boldsymbol{f}(\boldsymbol{v}_n) = \sum_{n'=1}^{N} \widetilde{\boldsymbol{k}}(\boldsymbol{v}_n, \boldsymbol{v}_{n'})\boldsymbol{\alpha}_{n'}, \tag{5.9}$$

where $\boldsymbol{\alpha}_{n'} \in \mathbb{R}^L$. As a result, estimating the nonlinear function reduces to estimating the coefficients $\{\boldsymbol{\alpha}_{n'}\}_{n'=1}^{N}$. Finally, the norm of the function $\boldsymbol{f}$ in the RKHS $\widetilde{\mathcal{H}}_{\boldsymbol{k}}$ can be written as:

$$\|\boldsymbol{f}\|^2_{\widetilde{\mathcal{H}}_{\boldsymbol{k}}} = \sum_{n=1}^{N}\sum_{n'=1}^{N} \boldsymbol{\alpha}_n^\top \widetilde{\boldsymbol{k}}(\boldsymbol{v}_n, \boldsymbol{v}_{n'})\boldsymbol{\alpha}_{n'}, \tag{5.10}$$

which gives a natural measure of the complexity of the function [Evgeniou et al., 2000, Micchelli and Pontil, 2005].

5.2.3 Kernel design

The kernel function $\widetilde{\boldsymbol{k}}(\boldsymbol{v}_n, \boldsymbol{v}_{n'})$, as defined in equation (5.7), is an $L \times L$ matrix. The design of the kernel allows to jointly define the nonlinearity and include prior information regarding the similarity between the outputs of the nonlinear function at different bands. The authors of [Micchelli and Pontil, 2005, Alvarez et al., 2012] describe two possible kernel classes known as the transformable and the separable kernels. In what follows we describe these two classes of kernels, and explain their relevance in the nonlinear unmixing context.

Transformable and separable kernels

The first class of matrix valued kernels is known as transformable kernels. In the transformable case, the kernel $\widetilde{\boldsymbol{k}}(\boldsymbol{v}_n, \boldsymbol{v}_{n'})$ is defined in a component-wise fashion through a scalar valued kernel. Each component of the kernel is defined as:

$$\left[\widetilde{\boldsymbol{k}}(\boldsymbol{v}_n, \boldsymbol{v}_{n'})\right]_{\ell,\ell'} = k(\boldsymbol{T}_\ell \boldsymbol{v}_n, \boldsymbol{T}_{\ell'} \boldsymbol{v}_{n'}), \tag{5.11}$$

where k is a scalar valued kernel, and $\boldsymbol{T}_\ell$ is an operator that extracts the c reflectance values in $\boldsymbol{v}_n$ corresponding to the ℓ-th band. More precisely, $\boldsymbol{T}_\ell \boldsymbol{v}_n = \text{col}(\{s_{\ell,i}\}_{i \in \boldsymbol{\mathcal{C}}_n})$ is a vector with c components. The scalar valued kernel acts jointly on pixels and bands indices, (n, n') and (ℓ, ℓ') respectively. The relevance of the transformable kernel is that the Gram matrix encodes the similarity between all pairs of pixels at all bands. Hence, it exploits all these correlations in order to jointly estimate the L components of the nonlinear function.

The second class of matrix valued kernels is known as the separable kernels. This class of kernels allows to incorporate prior information regarding the similarities between the different components of the vector valued function. The separable kernel is defined as the product between a scalar kernel acting on the input and an $L \times L$ positive semi-definite matrix encoding the similarities between the nonlinear contributions at different bands, i.e. between the different components of $\boldsymbol{f}$. For this class, the kernel is defined as follows:

$$\widetilde{\boldsymbol{k}}(\boldsymbol{v}_n, \boldsymbol{v}_{n'}) = k(\boldsymbol{v}_n, \boldsymbol{v}_{n'})\widetilde{\boldsymbol{\mathcal{E}}}, \tag{5.12}$$

where $\widetilde{\boldsymbol{\mathcal{E}}}$ is an $L \times L$ positive semi-definite matrix. The norm of $\boldsymbol{f}$ gives further insight on how $\widetilde{\boldsymbol{\mathcal{E}}}$ encodes the similarities between the nonlinear contributions at different bands. In fact, the norm of $\boldsymbol{f}$ in $\widetilde{\mathcal{H}}_{\boldsymbol{k}}$ [Evgeniou et al., 2005, Baldassarre, 2010] is given by:

$$\|\boldsymbol{f}\|^2_{\widetilde{\mathcal{H}}_{\boldsymbol{k}}} = \sum_{\ell=1}^{N}\sum_{\ell'=1}^{N} \widetilde{\mathcal{E}}^{\dagger}_{\ell,\ell'} \langle f_\ell, f_{\ell'} \rangle_{\mathcal{H}_k}, \tag{5.13}$$

where $\widetilde{\boldsymbol{\mathcal{E}}}^{\dagger}$ is the pseudo inverse of $\widetilde{\boldsymbol{\mathcal{E}}}$, the scalar valued nonlinear functions $f_1, \ldots, f_L$ belong to the RKHS $\mathcal{H}_k$ associated with k, such that $\boldsymbol{f} = [f_1, \ldots, f_L]^\top$. Note that in the case of the separable kernel, given (5.8) and (5.12), the overall Gram matrix can be written in the following form:

$$\widetilde{\boldsymbol{K}} = \boldsymbol{K} \otimes \widetilde{\boldsymbol{\mathcal{E}}}, \tag{5.14}$$

where $\otimes$ is the kronecker product, and $\boldsymbol{K}$ is the $N \times N$ Gram matrix associated with the scalar valued kernel, namely $k_{n,n'} = k(\boldsymbol{v}_n, \boldsymbol{v}_{n'})$. Hereafter, we investigate a special structure for the matrix $\widetilde{\boldsymbol{\mathcal{E}}}$. First, we assume that there exits prior information about the closeness between nonlinear contributions at different bands, i.e. between the functions $f_1, \ldots, f_L$. This prior can be modeled by a graph. We denote by $\widetilde{\boldsymbol{\mathcal{W}}} \in \mathbb{R}^{L \times L}$ the adjacency matrix of this graph [Grady and Polimeni, 2010]. More precisely, when two bands are likely to have similar nonlinear contributions, the corresponding nodes are connected by an edge and associated with a positive similarity weight $\widetilde{\mathcal{W}}_{\ell,\ell'} > 0$, otherwise $\widetilde{\mathcal{W}}_{\ell,\ell'}$ is set to zero. The authors of [Evgeniou et al., 2005] show that when $\widetilde{\boldsymbol{\mathcal{E}}}^{\dagger}$ is related to $\widetilde{\boldsymbol{\mathcal{W}}}$ as follows:

$$\begin{cases} \widetilde{\mathcal{E}}^{\dagger}_{\ell,\ell'} &= -\widetilde{\mathcal{W}}_{\ell,\ell'}, \text{ if } \ell \neq \ell', \\ \widetilde{\mathcal{E}}^{\dagger}_{\ell,\ell} &= \sum_{\ell'=1}^{N} \widetilde{\mathcal{W}}_{\ell,\ell'}, \text{ otherwise,} \end{cases} \tag{5.15}$$

using (5.13), the norm of $\boldsymbol{f}$ in $\widetilde{\mathcal{H}}_{\boldsymbol{k}}$ can be rewritten as:

$$\|\boldsymbol{f}\|^2_{\widetilde{\mathcal{H}}_{\boldsymbol{k}}} = \sum_{\ell=1}^{L} \|f_\ell\|^2_{\mathcal{H}_k} \widetilde{\mathcal{W}}_{\ell,\ell} + \frac{1}{2}\sum_{\ell=1}^{L}\sum_{\ell'=1}^{L} \|f_\ell - f_{\ell'}\|^2_{\mathcal{H}_k} \widetilde{\mathcal{W}}_{\ell,\ell'}. \tag{5.16}$$

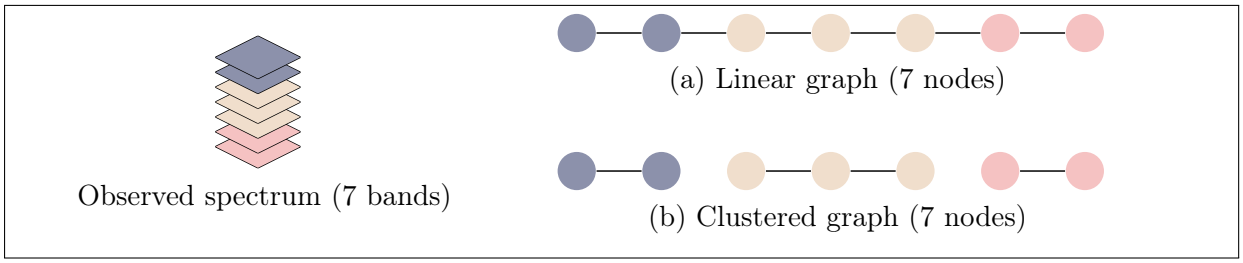


Figure 5.2: Two examples of the possible graph representations of the connections between the spectral bands (with $L = 7$) in the case of a : (a) Linear graph and (b) Clustered graph.

From a regularization point of view, the norm of $\boldsymbol{f}$ as given by equation (5.16) is known as the graph regularizer. It penalizes the norms of the individual functions in addition to the differences between each pair of functions, hence forcing them to be similar. Moreover, the strength of the similarity between each pair of functions is determined by the corresponding weight. More precisely, a high value of $\widetilde{\mathcal{W}}_{\ell,\ell'}$ promotes a strong similarity between f_ℓ and $f_{\ell'}$, and conversely, a low value of $\widetilde{\mathcal{W}}_{\ell,\ell'}$ promotes a weak similarity between the two functions. In other words, the norm of $\boldsymbol{f}$ as given by equation (5.16) promotes similarity between the estimated nonlinearities at different bands in accordance with the prior information reflected through the graph. Finally, note that when $\widetilde{\boldsymbol{\mathcal{E}}} = \boldsymbol{I}_L$, the norm of $\boldsymbol{f}$ reduces to the sum of the individual norms of its components $f_{\ell'}$. This corresponds to the case where there is no prior information between the nonlinearities at different bands.

In general, it is very likely that the nonlinear contributions at consecutive bands have smooth spectral variations. This prior information can be represented by a linear graph as shown in Figure 5.2 (a), where each node is connected to nodes at adjacent bands with unit weight. Nevertheless, more complex similarities can be incorporated using the graph structure. For example, in certain scenes nonlinear contributions can be dominant in certain spectral domains and less dominant in other spectral domains [Somers et al., 2009, Asner and Lobell, 2000]. This prior information can be represented by a clustered graph as shown in Figure 5.2 (b), where only nodes in spectral domains with similar nonlinear behavior are connected to each other.

Scalar kernel choice

The transformable and the separable kernels are both defined respectively in equations (5.11) and (5.12) using a scalar valued kernel k. Similarly to the case of functions in a scalar valued RKHS, the choice of the kernel corresponds to a certain representation of the input data in a higher dimensional space known as the feature space [Shawe-Taylor and Cristianini, 2004].

Hence, looking at the feature space can provide guidelines for choosing an appropriate kernel.

We focus on the polynomial and Gaussian kernels due to their successful application to nonlinear unmixing in the scalar valued case [Chen et al., 2013, 2014]. In particular, the second order homogeneous polynomial kernel:

$$k(\boldsymbol{v}_n, \boldsymbol{v}_{n'}) = (\boldsymbol{v}_n{}^\top \boldsymbol{v}_{n'})^2, \tag{5.17}$$

can be written as the inner product of the feature maps of $\boldsymbol{v}_n$ and $\boldsymbol{v}_{n'}$, where the feature map is defined as follows:

$$\begin{aligned}\phi(\boldsymbol{v}_n) = [&(v_{n,1})^2, \dots (v_{n,Lc})^2, \sqrt{2}(v_{n,1}v_{n,2})^2, \dots \sqrt{2}(v_{n,1}v_{n,Lc})^2, \\ &\sqrt{2}(v_{n,2}v_{n,3})^2, \sqrt{2}(v_{n,2}v_{n,Lc})^2, \dots \sqrt{2}(v_{n,Lc-1}v_{n,Lc})^2].\end{aligned} \tag{5.18}$$

The feature map of the second order homogeneous polynomial kernel maps its input vector to all the possible pairwise products between its components. This can be seen as a representation of all possible second order interactions between the spectral values in the input vector. On the other hand, the Gaussian kernel:

$$k(\boldsymbol{v}_n, \boldsymbol{v}_{n'}) = \exp(-\frac{\|\boldsymbol{v}_n - \boldsymbol{v}_{n'}\|^2}{2\sigma^2}), \tag{5.19}$$

can be expressed as an infinite series of higher order polynomial kernels:

$$k(\boldsymbol{v}_n, \boldsymbol{v}_{n'}) = \sum_{j=0}^{\infty} \frac{(\boldsymbol{v}_n{}^\top \boldsymbol{v}_{n'})^j}{\sigma^{2j} j!} \exp(-\frac{\|\boldsymbol{v}_n\|^2}{2\sigma^2}) \exp(-\frac{\|\boldsymbol{v}_{n'}\|^2}{2\sigma^2}). \tag{5.20}$$

Theoretically, the Gaussian kernel represents the case where an endless number of reflections occurs in the scene since it incorporates all higher order interactions between the input spectra. The drawback of the Gaussian kernel is that its feature map also contains a constant and a linear contribution (for $j = 0$ and $j = 1$ in equation (5.20) respectively) which can hinder the estimation accuracy. Nevertheless, the Gaussian kernel shows satisfying results in practice as will be seen in the experiments.

5.3 Estimation algorithm

5.3.1 Optimization problem

In this section we derive the optimization problem aimed at estimating the abundances and the nonlinear function based on model (5.5). We assume that the endmembers present in the scene are known. Assuming that the noise is white, Gaussian, with zero mean, and a possibly unknown

variance, the maximum likelihood estimation leads to the least square (LS) optimization problem:

$$\text{minimize}_{\boldsymbol{A},\boldsymbol{f}\in\widetilde{\mathcal{H}}_{\boldsymbol{k}}} \quad \tfrac{1}{2}\sum_{n=1}^{N}\|\boldsymbol{s}_n - \boldsymbol{R}\boldsymbol{a}_n - \boldsymbol{f}(\boldsymbol{v}_n)\|^2, \tag{5.21}$$

where $\boldsymbol{R} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_M]$, and $\boldsymbol{a}_n = [a_{1,n}, \ldots, a_{M,n}]^\top$. Problem (5.21) mainly ensures that the estimated model matches the observations. Nevertheless, the estimation of the abundances and the nonlinear function based on (5.21) is an underdetermined problem. As a result, it requires regularization and taking into account additional constraints on the abundances. For these reasons, we shall consider the following optimization problem:

$$\begin{aligned} \text{minimize}_{\boldsymbol{A},\boldsymbol{f}\in\widetilde{\mathcal{H}}_{\boldsymbol{k}}} \quad & \tfrac{1}{2}\sum_{n=1}^{N}\|\boldsymbol{s}_n - \boldsymbol{R}\boldsymbol{a}_n - \boldsymbol{f}(\boldsymbol{v}_n)\|^2 + \tfrac{\lambda}{2}\|\boldsymbol{f}\|^2_{\widetilde{\mathcal{H}}_{\boldsymbol{k}}} + \mu\mathcal{J}(\boldsymbol{A}) \\ \text{subject to} \quad & a_{i,n} \succeq 0 \quad \forall\, i = 1, \cdots M\,, n = 1, \cdots N, \\ & \textstyle\sum_{i=1}^{M} a_{i,n} = 1 \quad \forall\, n = 1, \cdots, N, \end{aligned} \tag{5.22}$$

where $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N]$, λ and μ are tuning parameters that control the tradeoff between the LS term and the two regularization terms. The first regularization, namely the ℓ_2-norm of $\boldsymbol{f}$ in $\widetilde{\mathcal{H}}_{\boldsymbol{k}}$, constrains the complexity of the estimated function [Evgeniou et al., 2000]. Furthermore, in the case of the separable kernel, it corresponds to the graph regularizer (5.16) which promotes certain similarities between the outputs of $\boldsymbol{f}$. The second regularizer, namely $\mathcal{J}(\boldsymbol{A})$, aims at incorporating prior information about the abundances. For example, in the experiments we use the Frobenius norm of the abundances:

$$\mathcal{J}(\boldsymbol{A}) = \frac{1}{2}\|\boldsymbol{A}\|^2_{\text{F}}, \tag{5.23}$$

known for promoting smoothness. As shown in the next section 5.3.2, using another expression for $\mathcal{J}(\boldsymbol{A})$ is not cumbersome and affects one step in the iterative algorithm. Nevertheless, an advantage of using the ℓ_2 norm of the nonlinear function and the Frobenius norm of the abundances is that each regularization is strictly convex with respect to the corresponding unknown variable. Hence, the overall optimization problem is strictly convex with respect to all the unknown variables. Finally, the proposed optimization problem (5.22) imposes the positivity and sum-to-one constraints on the estimated abundances. Some of the nonlinear mixing models in the literature keep the sum-to-one constraint, as for example [Altmann et al., 2014, Chen et al., 2014]. It can be argued that this constraint should be relaxed to $\sum_{i=1}^{M} a_{i,n} \leq 1$ especially when dealing with real hyperspectral data. Even if this constraint is strictly enforced in the proposed optimization problem (5.22), it can be relaxed by introducing a shade endmember in the endmember matrix [Heylen et al., 2011a]. Furthermore, we show in the next section that dropping this constraint requires a simple modification of the iterative algorithm.

5.3.2 Iterative algorithm

In this section, we use the alternating direction method of multipliers (ADMM) [Boyd et al., 2011] to solve the proposed optimization problem (5.22). The ADMM is a primal dual splitting method based on the augmented Lagrangian [Esser, 2010, Eckstein and Bertsekas, 1992]. Following the ADMM strategy, new variables and the corresponding consensus constraints are introduced in (5.22) in order to decouple the various terms in the objective function. We reformulate the optimization problem (5.22) in the following equivalent manner:

$$\begin{array}{ll} \text{minimize}_{\boldsymbol{X},\boldsymbol{Z},\boldsymbol{f}\in\widetilde{\mathcal{H}}_{\boldsymbol{k}}} & \frac{1}{2}\sum_{n=1}^{N}\|\boldsymbol{s}_n-\boldsymbol{R}\boldsymbol{x}_n-\boldsymbol{f}(\boldsymbol{v}_n)\|^2+\frac{\lambda}{2}\|\boldsymbol{f}\|^2_{\widetilde{\mathcal{H}}_{\boldsymbol{k}}}+\mu\mathcal{J}(\boldsymbol{Z})+\mathcal{I}_{\mathbb{R}_+^{M\times N}}(\boldsymbol{Z}) \\ \text{subject to} & \boldsymbol{\mathcal{A}}\boldsymbol{X}+\boldsymbol{\mathcal{B}}\boldsymbol{Z}=\boldsymbol{\mathcal{C}}, \end{array} \tag{5.24}$$

with

$$\boldsymbol{\mathcal{A}}=\begin{pmatrix}\boldsymbol{I}_M\\ \boldsymbol{1}_M^\top\end{pmatrix},\ \boldsymbol{\mathcal{B}}=\begin{pmatrix}-\boldsymbol{I}_M\\ \boldsymbol{0}_M^\top\end{pmatrix},\ \boldsymbol{\mathcal{C}}=\begin{pmatrix}\boldsymbol{0}_{M\times N}\\ \boldsymbol{1}_N^\top\end{pmatrix} \tag{5.25}$$

where $\boldsymbol{X}$ and $\boldsymbol{Z}$ are the ADMM variables, and $\mathcal{I}_{\mathbb{R}_+^{M\times N}}(\boldsymbol{Z})$ is the indicator of $\mathbb{R}_+^{M\times N}$ (i.e., $\mathcal{I}_{\mathbb{R}_+^{M\times N}}(\boldsymbol{Z})=0$ if $\boldsymbol{Z}\in\mathbb{R}_+^{M\times N}$ and $\mathcal{I}_{\mathbb{R}_+^{M\times N}}(\boldsymbol{Z})=\infty$ if $\boldsymbol{Z}\notin\mathbb{R}_+^{M\times N}$). Compared to problem (5.22), $\boldsymbol{A}$ was substituted by $\boldsymbol{X}$ and $\boldsymbol{Z}$, and a consensus constraint between the new variables was introduced. The positivity constraint was moved to the objective function through the indicator function, and the sum-to-one was incorporated within the equality constraint. As mentioned in the previous section, the sum-to-one constraint can be relaxed by adding a shade endmember to $\boldsymbol{R}$. Another alternative for relaxing the sum-to-one is by changing the definition of the matrices in (5.25) to:

$$\boldsymbol{\mathcal{A}}=\boldsymbol{I}_M,\quad \boldsymbol{\mathcal{B}}=-\boldsymbol{I}_M,\quad \boldsymbol{\mathcal{C}}=\boldsymbol{0}_{M\times N}. \tag{5.26}$$

The augmented Lagrangian associated with problem (5.24) is given by:

$$\begin{aligned}\mathcal{L}_\rho(\boldsymbol{X},\boldsymbol{Z},\boldsymbol{f},\boldsymbol{\Lambda}_\rho)= &\ \frac{1}{2}\sum_{n=1}^{N}\|\boldsymbol{s}_n-\boldsymbol{R}\boldsymbol{x}_n-\boldsymbol{f}(\boldsymbol{v}_n)\|^2+\frac{\lambda}{2}\|\boldsymbol{f}\|^2_{\widetilde{\mathcal{H}}_{\boldsymbol{k}}}+\mu\mathcal{J}(\boldsymbol{Z})+\mathcal{I}_{\mathbb{R}_+^{M\times N}}(\boldsymbol{Z})\\ &+\text{trace}(\boldsymbol{\Lambda}_\rho^\top(\boldsymbol{\mathcal{A}}\boldsymbol{X}+\boldsymbol{\mathcal{B}}\boldsymbol{Z}-\boldsymbol{\mathcal{C}}))+\frac{\rho}{2}\|\boldsymbol{\mathcal{A}}\boldsymbol{X}+\boldsymbol{\mathcal{B}}\boldsymbol{Z}-\boldsymbol{\mathcal{C}}\|_{\text{F}}^2,\end{aligned} \tag{5.27}$$

where $\boldsymbol{\Lambda}_\rho$ is the matrix of Lagrange multipliers associated with the linear constraints in (5.24), and ρ is the penalty parameter. At each iteration, the ADMM algorithm consists of minimizing the augmented Lagrangian (5.27) sequentially. First, it is minimized with respect to the unknown variables $\{\boldsymbol{X},\boldsymbol{f}\}$ and then with respect to $\boldsymbol{Z}$ while in each minimization keeping the other variables fixed to their previous estimate. Finally, it consists of updating the Lagrange multipliers matrix $\boldsymbol{\Lambda}_\rho$ associated with the linear constraints. This approach allows to break the optimization

problem into a sequence of smaller and simpler sub-problems. To summarize, the ADMM at iteration $k+1$ consists of the following steps:

$$\begin{aligned} \{\boldsymbol{X}^{k+1}, \boldsymbol{f}^{k+1}\} &= \text{minimize}_{\boldsymbol{X},\boldsymbol{f}\in\widetilde{\mathcal{H}}_{\boldsymbol{k}}} \mathcal{L}_\rho(\boldsymbol{X}, \boldsymbol{Z}^k, \boldsymbol{f}, \boldsymbol{\Lambda}_\rho^k), \\ \boldsymbol{Z}^{k+1} &= \text{minimize}_{\boldsymbol{Z}} \mathcal{L}_\rho(\boldsymbol{X}^{k+1}, \boldsymbol{Z}, \boldsymbol{f}^{k+1}, \boldsymbol{\Lambda}_r ho^k), \\ \boldsymbol{\Lambda}_\rho^{k+1} &= \boldsymbol{\Lambda}_\rho^k + \rho(\boldsymbol{\mathcal{A}}\boldsymbol{X}^{k+1} + \boldsymbol{\mathcal{B}}\boldsymbol{Z}^{k+1} - \boldsymbol{\mathcal{C}}). \end{aligned} \tag{5.28}$$

The ADMM steps, namely the $\{\boldsymbol{X}, \boldsymbol{f}\}$ minimization step, the $\boldsymbol{Z}$ minimization step, and the update of the Lagrange multipliers, are developed hereafter. To keep the notations simple, we drop the iteration index in the development of the first two steps.

$\{\boldsymbol{X}, \boldsymbol{f}\}$ minimization step

This step consists of minimizing the augmented Lagrangian with respect to $\{\boldsymbol{X}, \boldsymbol{f}\}$. After discarding the terms independent of $\{\boldsymbol{X}, \boldsymbol{f}\}$ in (5.27), this step reduces to the following optimization problem:

$$\text{minimize}_{\boldsymbol{X},\boldsymbol{f}\in\widetilde{\mathcal{H}}_{\boldsymbol{k}}} \tfrac{1}{2} \sum_{n=1}^{N} \|\boldsymbol{s}_n - \boldsymbol{R}\boldsymbol{x}_n - \boldsymbol{f}(\boldsymbol{v}_n)\|^2 + \tfrac{\lambda}{2}\|\boldsymbol{f}\|^2_{\widetilde{\mathcal{H}}_{\boldsymbol{k}}} + \text{trace}(\boldsymbol{\Lambda}_\rho^\top \boldsymbol{\mathcal{A}}\boldsymbol{X}) + \tfrac{\rho}{2}\|\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}\|^2_{\text{F}}. \tag{5.29}$$

We rewrite (5.29) in the following equivalent form:

$$\begin{aligned} \text{minimize}_{\boldsymbol{X},\boldsymbol{f}\in\widetilde{\mathcal{H}}_{\boldsymbol{k}},\boldsymbol{E}} \quad & \tfrac{1}{2} \sum_{n=1}^{N} \|\boldsymbol{e}_n\|^2 + \tfrac{\lambda}{2}\|\boldsymbol{f}\|^2_{\widetilde{\mathcal{H}}_{\boldsymbol{k}}} + \text{trace}(\boldsymbol{\Lambda}_\rho^\top \boldsymbol{\mathcal{A}}\boldsymbol{X}) + \tfrac{\rho}{2}\|\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}\|^2_{\text{F}} \\ \text{subject to} \quad & \boldsymbol{e}_n = \boldsymbol{s}_n - \boldsymbol{R}\boldsymbol{x}_n - \boldsymbol{f}(\boldsymbol{v}_n), \\ & \forall\, n = 1, \cdots, N, \end{aligned} \tag{5.30}$$

where $\boldsymbol{E} = [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_N]$, and solve its dual problem. The Lagrangian associated with problem (5.30) is given by:

$$\begin{aligned} \mathcal{L}(\boldsymbol{X}, \boldsymbol{f}, \boldsymbol{E}, \boldsymbol{\Lambda}) = \tfrac{1}{2} \sum_{n=1}^{N} \|\boldsymbol{e}_n\|^2 + \tfrac{\lambda}{2}\|\boldsymbol{f}\|^2_{\widetilde{\mathcal{H}}_{\boldsymbol{k}}} + \textstyle\sum_{n=1}^{N} \boldsymbol{\lambda}_n^\top(\boldsymbol{s}_n - \boldsymbol{R}\boldsymbol{x}_n - \boldsymbol{f}(\boldsymbol{v}_n) - \boldsymbol{e}_n) \\ + \text{trace}(\boldsymbol{\Lambda}_\rho^\top \boldsymbol{\mathcal{A}}\boldsymbol{X}) + \tfrac{\rho}{2}\|\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}\|^2_{\text{F}}, \end{aligned} \tag{5.31}$$

where $\boldsymbol{\Lambda} = [\boldsymbol{\lambda}_1, \cdots, \boldsymbol{\lambda}_N]$ is the matrix of Lagrange multipliers associated with the linear constraints in (5.30). The partial derivatives of the Lagrangian with respect to the primal variables, namely $\boldsymbol{X}, \boldsymbol{f}$ and $\boldsymbol{E}$, are:

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial \boldsymbol{X}} &= \rho\boldsymbol{\mathcal{A}}^\top\boldsymbol{\mathcal{A}}\boldsymbol{X} - \boldsymbol{R}^\top\boldsymbol{\Lambda} + \boldsymbol{\mathcal{A}}^\top\boldsymbol{\Lambda}_\rho + \rho\boldsymbol{\mathcal{A}}^\top(\boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}), \\ \frac{\partial \mathcal{L}}{\partial \boldsymbol{f}} &= \lambda\boldsymbol{f}(\cdot) - \frac{1}{\lambda}\sum_{j=1}^{N} \widetilde{\boldsymbol{k}}(\cdot, \boldsymbol{v}_j)\boldsymbol{\lambda}_j, \\ \frac{\partial \mathcal{L}}{\partial \boldsymbol{E}} &= \boldsymbol{E} - \boldsymbol{\Lambda}. \end{cases} \tag{5.32}$$

Setting the gradient of the partial derivatives in (5.32) to zero gives the primal variables as a function of the Lagrange multipliers:

$$\begin{cases} \boldsymbol{X} & = \frac{(\boldsymbol{\mathcal{A}}^\top \boldsymbol{\mathcal{A}})^{-1}}{\rho}(\boldsymbol{R}^\top \boldsymbol{\Lambda} - \boldsymbol{\mathcal{A}}^\top \boldsymbol{\Lambda}_\rho - \rho \boldsymbol{\mathcal{A}}^\top (\boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}})), \\ \boldsymbol{f}(\cdot) & = \frac{1}{\lambda}\sum_{j=1}^{N} \widetilde{\boldsymbol{k}}(\cdot, \boldsymbol{v}_j)\boldsymbol{\alpha}_j, \\ \boldsymbol{E} & = \boldsymbol{\Lambda}. \end{cases} \tag{5.33}$$

To derive the Lagrange dual problem, the primal variables are substituted in (5.31) by their expressions from (5.33). This results in a quadratic form with respect to the Lagrange multipliers, and yields the following dual problem:

$$\text{maximize}_{\boldsymbol{\Lambda}} \quad -\tfrac{1}{2}\text{vec}(\boldsymbol{\Lambda})^\top \boldsymbol{Q}\text{vec}(\boldsymbol{\Lambda}) + \text{vec}(\boldsymbol{\Lambda})^\top \boldsymbol{p}, \tag{5.34}$$

with

$$\begin{cases} \boldsymbol{Q} & = \boldsymbol{I}_{LN} + \frac{1}{\lambda}\widetilde{\boldsymbol{K}} + \frac{1}{\rho}\boldsymbol{I}_N \otimes \boldsymbol{D}, \\ \boldsymbol{p} & = \text{vec}(\boldsymbol{S} + \frac{1}{\rho}\boldsymbol{R}(\boldsymbol{\mathcal{A}}^\top \boldsymbol{\mathcal{A}})^{-1}\boldsymbol{\mathcal{A}}^\top(\boldsymbol{\Lambda}_\rho + \rho(\boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}))), \end{cases} \tag{5.35}$$

where $\boldsymbol{D} = \boldsymbol{R}(\boldsymbol{\mathcal{A}}^\top \boldsymbol{\mathcal{A}})^{-1}\boldsymbol{R}^\top$, $\text{vec}(\cdot)$ is an operator that stacks the columns of a matrix on top of each other. As a result, the $\{\boldsymbol{X}, \boldsymbol{f}\}$ minimization step reduces to solving the following linear equation system:

$$\boldsymbol{Q}\,\text{vec}(\boldsymbol{\Lambda}) = \boldsymbol{p}, \tag{5.36}$$

with LN unknown variables. Once $\boldsymbol{\Lambda}$ is determined, it is substituted in (5.33) in order to evaluate the updated abundances. Note that the nonlinear function does not need to be evaluated at each iteration. It can be evaluated once the ADMM algorithm has converged.

$\boldsymbol{Z}$ minimization step

This step consists of minimizing the augmented Lagrangian with respect to $\boldsymbol{Z}$. After discarding the terms independent of $\boldsymbol{Z}$ in (5.27) and accounting for the special structure of the matrices $\boldsymbol{\mathcal{A}}$, $\boldsymbol{\mathcal{B}}$, and $\boldsymbol{\mathcal{C}}$ given in (5.25), problem (5.37) reduces to the following optimization problem:

$$\text{minimize}_{\boldsymbol{Z}} \quad \tfrac{\rho}{2}\left\|\boldsymbol{Z} - \boldsymbol{X}\right\|_{\text{F}}^2 - \text{trace}(\boldsymbol{\Lambda}_\rho^\top \boldsymbol{Z}) + \mu\mathcal{J}(\boldsymbol{Z}) + \mathcal{I}_{\mathbb{R}_+^{M\times N}}(\boldsymbol{Z}). \tag{5.37}$$

In particular, when $\mathcal{J}(\boldsymbol{Z})$ is the Frobenius norm (5.23), problem (5.37) reduces to the following positively constrained least squares problem:

$$\text{minimize}_{\boldsymbol{Z}} \quad \tfrac{1}{2}\left\|\boldsymbol{Z} - \tfrac{\rho}{\rho+\mu}\,(\boldsymbol{X} + \tfrac{1}{\rho}\boldsymbol{\Lambda}_\rho)\right\|_{\text{F}}^2 + \mathcal{I}_{\mathbb{R}_+^{M\times N}}(\boldsymbol{Z}). \tag{5.38}$$

The solution of problem (5.38) is obtained by a projection onto the positive orthant:

$$\boldsymbol{Z} = \tfrac{\rho}{\rho+\mu}\,(\boldsymbol{X} + \tfrac{1}{\rho}\boldsymbol{\Lambda}_\rho)_+\,, \tag{5.39}$$

where $(\cdot)_+ = \max(0, \cdot)$ is applied component wise. As mentioned previously, $\mathcal{J}(\boldsymbol{Z})$ can be set to a regularization other than the Frobenius norm. For example, we demonstrate the case of the ℓ_1 norm known for promoting sparse abundances. In this case, problem (5.37) reduces to the following optimization problem:

$$\text{minimize}_{\boldsymbol{Z}} \quad \tfrac{1}{2}\,\|\boldsymbol{Z} - (\boldsymbol{X} + \tfrac{1}{\rho}\boldsymbol{\Lambda}_\rho)\|_{\mathrm{F}}^2 + \tfrac{\mu}{\rho}\|\boldsymbol{Z}\|_1 + \mathcal{I}_{\mathbb{R}_+^{M\times N}}(\boldsymbol{Z}). \tag{5.40}$$

The solution of (5.40) is the well-known soft thresholding [Chen et al., 2001] applied to the projection onto the positive orthant:

$$\boldsymbol{Z} = \text{soft}_{\frac{\mu}{\rho}}((\boldsymbol{X} + \frac{1}{\rho}\boldsymbol{\Lambda}_\rho)_+), \tag{5.41}$$

where $\text{soft}_{\frac{\mu}{\rho}}(\cdot) = \text{sgn}(\cdot)(|\cdot| - \frac{\mu}{\rho})_+$, and the soft thresholding operator is applied component wise. The solution in equation (5.41) can be simplified to:

$$\boldsymbol{Z} = (\boldsymbol{X} + \frac{1}{\rho}\boldsymbol{\Lambda}_\rho - \frac{\mu}{\rho})_+\,. \tag{5.42}$$

It is important to note that the sum to one constraint should be relaxed in (5.24) when the ℓ_1 norm is considered. Otherwise, $\mathcal{J}(\boldsymbol{Z})$ would be a constant in the feasible set, i.e. when the abundances are positive and sum to one [Bioucas-Dias and Figueiredo, 2010].

Update of the Lagrange multipliers

The last step consists of updating the Lagrange multipliers according to the following rule,

$$\boldsymbol{\Lambda}_\rho^{k+1} = \boldsymbol{\Lambda}_\rho^k + \rho(\boldsymbol{\mathcal{A}}\boldsymbol{X}^{k+1} + \boldsymbol{\mathcal{B}}\boldsymbol{Z}^{k+1} - \boldsymbol{\mathcal{C}}), \tag{5.43}$$

This step can be seen as a gradient ascent of the augmented Lagrangian with respect to the Lagrange multiplier. Furthermore, it evaluates the running sum of the constraint residuals.

5.3.3 Implementation details

The ADMM steps described in the previous section are repeated until convergence. As suggested in [Boyd et al., 2011], a reasonable stopping criteria is that the primal and dual residuals must be smaller than some tolerance thresholds, namely,

$$\|\boldsymbol{\mathcal{A}}\boldsymbol{X}^{k+1} + \boldsymbol{\mathcal{B}}\boldsymbol{Z}^{k+1} - \boldsymbol{\mathcal{C}}\|_{\mathrm{F}} \leq \epsilon_{\text{pri}}, \tag{5.44}$$

$$\|\rho \boldsymbol{\mathcal{A}}^\top \boldsymbol{\mathcal{B}}(\boldsymbol{Z}^{k+1} - \boldsymbol{Z}^k)\|_{\mathrm{F}} \leq \epsilon_{\mathrm{dual}}, \tag{5.45}$$

We refer to the proposed algorithm as nonlinear neighbor and band dependent unmixing (NDU). The pseudocode of the proposed algorithm is given in Algorithm 3. The most computationally expensive step in the iterative algorithm is the $\boldsymbol{X}$ minimization step which requires solving an $LN \times LN$ system of linear equations (5.36). Solving the system of linear equations would have a memory complexity $\mathcal{O}(L^2N^2)$ and a runtime complexity $\mathcal{O}(L^3N^3)$. In practice, we do not need to compute the exact solution. The ADMM algorithm will converge even if the $\boldsymbol{X}$ minimization step is carried out only approximately [Boyd et al., 2011, Eckstein and Bertsekas, 1992]. This allows us to solve (5.36) using an iterative algorithm which can reduce the runtime complexity. In the case of the separable kernel, both the memory and the runtime complexity can be reduced furthermore by exploiting the fact that $\boldsymbol{Q}$ is the sum of kronecker products.

The conjugate gradient (CG) [Saad, 2003] is one of the most widely used iterative techniques for solving a large linear system of equations where $\boldsymbol{Q}$ is a positive definite matrix as in (5.36). The CG can yield the exact solution after LN iterations, but in practice a good initialization yields faster convergence [Shewchuk, 1994]. At each iteration of the CG, the dominating operation is a matrix vector multiplication involving $\boldsymbol{Q}$. In general, the number of operations required for multiplying $\boldsymbol{Q}$ by a vector is $\mathcal{O}(L^2N^2)$. However, in the case of the separable kernel, $\boldsymbol{Q}$ is a sparse matrix and can be written as the sum of kronecker products:

$$\boldsymbol{Q} = \boldsymbol{I}_L \otimes \boldsymbol{I}_N + \frac{1}{\lambda}\boldsymbol{K} \otimes \widetilde{\boldsymbol{\mathcal{E}}} + \frac{1}{\rho}\boldsymbol{I}_N \otimes \boldsymbol{D}, \tag{5.46}$$

where we have replaced $\widetilde{\boldsymbol{K}}$ by its expression from (5.14). For an efficient implementation of the product between $\boldsymbol{Q}$ and some vector $\mathrm{vec}(\boldsymbol{\mathcal{J}})$ where $\boldsymbol{\mathcal{J}}$ is an $L \times N$ matrix, the following relationships can be used:

$$\begin{cases} (\boldsymbol{K} \otimes \widetilde{\boldsymbol{\mathcal{E}}})\mathrm{vec}(\boldsymbol{\mathcal{J}}) = \mathrm{vec}(\widetilde{\boldsymbol{\mathcal{E}}}\boldsymbol{\mathcal{J}}\boldsymbol{K}), \\ (\boldsymbol{I}_N \otimes \boldsymbol{D})\mathrm{vec}(\boldsymbol{\mathcal{J}}) = \mathrm{vec}(\boldsymbol{D}\boldsymbol{\mathcal{J}}) \end{cases} \tag{5.47}$$

Given (5.46) and (5.47), the overall product between $\boldsymbol{Q}$ and some vector $\mathrm{vec}(\boldsymbol{\mathcal{J}})$ is given by:

$$\boldsymbol{Q}\mathrm{vec}(\boldsymbol{\mathcal{J}}) = \mathrm{vec}(\boldsymbol{\mathcal{J}}) + \frac{1}{\lambda}\mathrm{vec}(\widetilde{\boldsymbol{\mathcal{E}}}\boldsymbol{\mathcal{J}}\boldsymbol{K}) + \frac{1}{\rho}\mathrm{vec}(\boldsymbol{D}\boldsymbol{\mathcal{J}}). \tag{5.48}$$

Equation (5.48) is expressed in terms of ordinary matrix products, which means that we do not have to compute any kronecker products. Compared to the case where the kronecker product is evaluated, the memory complexity is reduced from $\mathcal{O}(L^2N^2)$ to $\mathcal{O}(L^2 + N^2)$ and the number of operations at each iteration is reduced from $\mathcal{O}(L^2N^2)$ to $\mathcal{O}(\max(L^2N, LN^2))$.

Algorithm 3 : $[\boldsymbol{X}, \boldsymbol{F}] = \mathrm{NDU}(\boldsymbol{S}, \boldsymbol{R}, \lambda, \mu, \rho)$

Precompute $\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}, \boldsymbol{\mathcal{C}}, \widetilde{\boldsymbol{K}}, \boldsymbol{Q}$
Initialize $\boldsymbol{Z}, \boldsymbol{\Lambda}_\rho$
while $\mathrm{res}_{\mathrm{pri}} > \epsilon_{\mathrm{pri}}$ or $\mathrm{res}_{\mathrm{dual}} > \epsilon_{\mathrm{dual}}$ **do**
  $\boldsymbol{p} = \mathrm{vec}(\boldsymbol{S} + \frac{1}{\rho}\boldsymbol{R}(\boldsymbol{\mathcal{A}}^\top\boldsymbol{\mathcal{A}})^{-1}\boldsymbol{\mathcal{A}}^\top(\boldsymbol{\Lambda}_\rho + \rho(\boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}})))$
  $\mathrm{vec}(\boldsymbol{\Lambda}) = \boldsymbol{Q}^{-1}\boldsymbol{p}$ % See section 5.3.3
  $\boldsymbol{X} = \frac{1}{\rho}(\boldsymbol{\mathcal{A}}^\top\boldsymbol{\mathcal{A}})^{-1}(\boldsymbol{R}^\top\boldsymbol{\Lambda} - \boldsymbol{\mathcal{A}}^\top\boldsymbol{\Lambda}_\rho - \rho\boldsymbol{\mathcal{A}}^\top(\boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}))$
  $\boldsymbol{Z}^{\mathrm{old}} = \boldsymbol{Z}$
  if $\mathcal{J}(\boldsymbol{Z}) = \|\boldsymbol{Z}\|_{\mathrm{F}}^2$ **then**
    $\boldsymbol{Z} = \frac{\rho}{\rho+\mu}\,(\boldsymbol{X} + \frac{1}{\rho}\boldsymbol{\Lambda}_\rho)_+$
  else if $\mathcal{J}(\boldsymbol{Z}) = \|\boldsymbol{Z}\|_1$ **then**
    $\boldsymbol{Z} = (\boldsymbol{X} + \frac{1}{\rho}\boldsymbol{\Lambda}_\rho - \frac{\mu}{\rho})_+$
  end if
  $\boldsymbol{\Lambda}_\rho = \boldsymbol{\Lambda}_\rho + \rho(\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}})$
  $\mathrm{res}_{\mathrm{pri}} = \|\boldsymbol{\mathcal{A}}\boldsymbol{X} + \boldsymbol{\mathcal{B}}\boldsymbol{Z} - \boldsymbol{\mathcal{C}}\|_{\mathrm{F}}$
  $\mathrm{res}_{\mathrm{dual}} = \|\rho\boldsymbol{\mathcal{A}}^\top\boldsymbol{\mathcal{B}}(\boldsymbol{Z} - \boldsymbol{Z}^{\mathrm{old}})\|_{\mathrm{F}}$
end while
$\boldsymbol{F} = \frac{1}{\lambda}\widetilde{\boldsymbol{K}}\mathrm{vec}(\boldsymbol{\Lambda})$
$\boldsymbol{F} = \mathrm{reshape}(\boldsymbol{F}, L, N)$

5.4 Experiments

5.4.1 Synthetic data

Data generation

The proposed approach is first illustrated using synthetic data. Several patches were generated according to three mixing models that incorporate the main assumptions underlying the proposed nonlinear mixing model, i.e., bilinear contributions, adjacency effects, and band selectivity. The bilinear contributions are created by adding pairwise products of spectra, the adjacency effect is created by adding bilinear contributions from neighboring pixels, and band selectivity is created by assigning a different weight to the nonlinear contributions at different bands. The three mixing models (MM) are denoted as MM 1, MM 2, and MM 3, and they are defined as follows:

- MM 1 (bilinear contributions):

$$\boldsymbol{s}_n \;\; = \boldsymbol{s}_n^{\mathrm{lin}} + u\,\boldsymbol{s}_n^{\mathrm{lin}} \odot \boldsymbol{s}_n^{\mathrm{lin}} + \boldsymbol{e}_n \tag{5.49}$$

- MM 2 (bilinear contributions + adjacency effects):

$$\boldsymbol{s}_n \;\; = \boldsymbol{s}_n^{\mathrm{lin}} + u\,\textstyle\sum_{i=n-2}^{n+2}\gamma_{n,i}\,\boldsymbol{s}_i^{\mathrm{lin}} \odot \boldsymbol{s}_i^{\mathrm{lin}} + \boldsymbol{e}_n \tag{5.50}$$

- MM 3 (bilinear contributions + adjacency effects + band selectivity):

$$\boldsymbol{s}_n = \boldsymbol{s}_n^{\text{lin}} + u \sum_{i=n-2}^{n+2} \gamma_{n,i}\, \boldsymbol{s}_i^{\text{lin}} \odot \boldsymbol{s}_i^{\text{lin}} \odot \boldsymbol{h} + \boldsymbol{e}_n \tag{5.51}$$

where u in equation (5.49) is an attenuation parameter set to 0.2 in all the simulations, the coefficients $\gamma_{n,i}$ in equation (5.50) assign a different weight to the bilinear contributions coming from neighbors, the coeficients were set to $\gamma_{n,n-2} = \gamma_{n,n+2} = 0.05$, $\gamma_{n,n-1} = \gamma_{n,n+1} = 0.3$ and $\gamma_{n,n} = 0.4$, and $\boldsymbol{h}$ in equation (5.51) is an L dimensional vector where each component assigns a different weight to the nonlinear contribution at the corresponding band. Figure 5.3 (a) shows the entries of $\boldsymbol{h}$ that were used for the experiments. In fact, $\boldsymbol{h}$ was chosen such that it favors nonlinear contributions at the center of the spectrum and attenuates nonlinear contributions at the extremities of the spectrum. Note that $\boldsymbol{h}$ is unknown by all the unmixing methods used in the experiments. Several patches were created with $N = 100$ pixels using different numbers of endmembers and different values of the SNR. The endmembers were selected from the USGS spectral library of minerals. Their frequency bands are in the range $400 - 2560$ nm, and were decimated such as to have $L = 20$ bands. Figure 5.3 (b) shows five endmembers spectra used in the simulations. The abundances were generated using a beta distribution with a unit shape parameter.

Unmixing methods

We tested three nonlinear unmixing algorithms. The first algorithm is the extended endmember matrix method. It considers the linear mixing model where the endmember matrix is extended by adding the pairwise products of the endmembers [Raksuntorn and Du, 2010]. This algorithm is denoted as Ext in the experiments, it consists of solving a positively constrained least squares problem and has no tuning parameters. The second algorithm is the one proposed in [Chen et al., 2013], it is denoted as Khype and based on scalar valued RKHS. Khype was tested with a Gaussian (G) and a second order homogeneous polynomial (P) kernel. The third algorithm is the one proposed in this chapter, it is denoted in what follows as NDU (Nonlinear neighbor and band Dependent Unmixing). Similarly to Khype, NDU was tested with a Gaussian (G) and a second order homogeneous polynomial (P) kernel. Furthermore, given that the kernel in NDU is matrix valued, it was tested using a transformable (Tr.) and a separable (Sp.) structure. In the case of the separable kernel, the graph that represents the similarities between the different bands is linear, i.e. each band is connected to the previous and next band, with unit weights. In order to determine $\boldsymbol{v}_n$, the neighborhood was set to $\mathcal{C}_n = \{n, n-1, n+1\}$.

Note that NDU requires tuning two parameters λ and μ, whereas Khype [Chen et al., 2013] uses the same parameter to penalize the norm of the abundances and the nonlinear function ($\lambda = \mu$). In order to have a fair comparison between the two algorithms, we modified Khype such as to have two distinct parameters λ and μ. For both algorithms, the tuning parameters were tested in the range [10^{-4} 10^{-3} 10^{-2} 10^{-1} 1 10]. The standard deviations of the Gaussian kernels used with Khype and NDU were chosen such that the resulting Gram matrices have their values in the same range. The polynomial Gram matrices were scaled in order to have their values in the range [0 1]. Figure 5.4 shows the Gram matrices used with Khype, and NDU in different settings and obtained with $M = 3$ and $SNR = 40$ dB. The first column in Figure 5.4 shows the $L \times L$ Gram matrices used by Khype. The second and third columns in Figure 5.4 show the Gram matrices used by NDU obtained with a transformable and separable structure respectively. Recall that the NDU Gram matrices are $N \times N$ block matrices, where each block is an $L \times L$ matrix. In the case of the transformable kernel, each block is an $L \times L$ Gram matrix itself. Figure 5.4 shows that a block or sub-Gram matrix in the transformable kernel is similar the corresponding Khype Gram matrix even though it is calculated using the observations themselves rather than the endmember matrix as in Khype. Whereas for the separable kernel, each block is equal to $\widetilde{\boldsymbol{\mathcal{E}}}^{\dagger}$ multiplied by the corresponding scalar valued kernel.

Performance measures

The abundance estimation accuracy was evaluated using the RMSE between the actual abundances and their estimates:

$$\text{RMSE}_{\boldsymbol{A}} = \sqrt{\frac{1}{MN}\|\boldsymbol{A} - \widehat{\boldsymbol{A}}\|_{\text{F}}^2}, \tag{5.52}$$

where $\boldsymbol{A}$ represents the true abundances matrix and $\widehat{\boldsymbol{A}}$ represents the abundances estimated by the unmixing algorithm. In addition to the abundances, each one of the unmixing algorithms estimates the nonlinear contributions. Let $\boldsymbol{F}$ and $\widehat{\boldsymbol{F}}$ denote the true and estimated $L \times N$ matrices of nonlinear contributions. The estimation accuracy of the nonlinear part was also evaluated using the RMSE between $\boldsymbol{F}$ and its estimate $\widehat{\boldsymbol{F}}$:

$$\text{RMSE}_{\boldsymbol{F}} = \sqrt{\frac{1}{LN}\|\boldsymbol{F} - \widehat{\boldsymbol{F}}\|_{\text{F}}^2}. \tag{5.53}$$

Simulation results

Tables 5.2, 5.3 and 5.4 report the results obtained using MM 1, MM 2, and MM 3 respectively. For each case, we report the root mean square errors of the estimated abundances (first term in

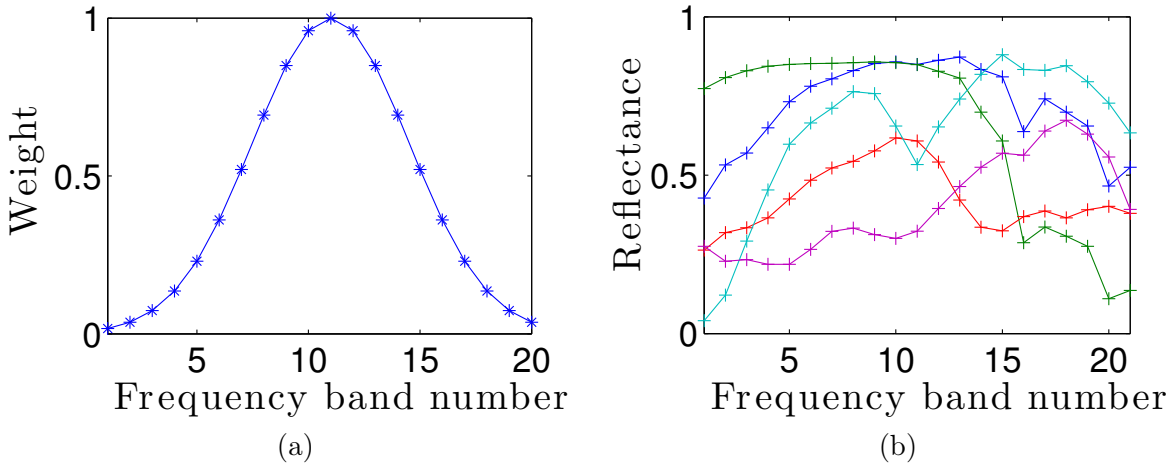


Figure 5.3: From left to right: (a) The components of vector $\boldsymbol{h}$ corresponding to the weight of the nonlinear contribution at each band, (b) Endmembers spectra used to generate the illustrative examples.

brackets) and the estimated nonlinear contributions (second term in brackets). Khype and NDU require tuning two parameters, namely λ and μ, the best parameters are reported in brackets below the RMSE results. More precisely, the first term and second term in brackets correspond to the best values of λ and μ respectively. The estimation accuracy of the three methods became worse when the noise level increased. The same performance was observed for different values of M. In general, Khype outperformed Ext, and NDU outperformed both methods. Note that Ext gave results worse than NDU and Khype even when MM 1 was used. This is most probably due to the fact that the sum to one constraint was not incorporated in the implementation of Ext, unlike the case of Khype and NDU. In the majority of the cases, NDU gave the best RMSE of the abundances and the nonlinear part when used with the separable and polynomial kernel (Sp. + P). The polynomial and the Gaussian kernels usually had comparable results in all scenarios. The transformable kernel and the separable kernel had comparable results only when MM 1 and 2 were used, i.e. in tables 5.2 and 5.3. However, the separable kernel outperformed the transformable kernel when MM 3 was used, i.e. in table 5.4.

Figure 5.5 shows the true and estimated nonlinear contributions for all pixels at all bands obtained with $P = 3$ and SNR=40 dB. The first row in Figure 5.5 shows the true nonlinear contributions. Whereas the following rows show the nonlinear contributions estimated with Ext, Khype, NDU with a transformable kernel, and NDU with a separable kernel. For conciseness, we show the results with either the Gaussian or the polynomial kernel for Khype and NDU. In particular, the kernel that gave the best RMSE was chosen. Figure 5.5 mainly allows to visually compare the estimation of the nonlinear contribution obtained with the various methods and

mixing models. The first column in Figure 5.5 corresponds to the results obtained with MM 1, i.e. with bilinear contributions. It can be seen that Khype and NDU slightly outperformed Ext. The second column in Figure 5.5 corresponds to the results obtained with MM 2, i.e. with bilinear contributions and adjacency effects. The first figure in column 2 shows that the true nonlinear contributions at adjacent pixels (i.e. at adjacent columns in the image) have smooth variations. In this case, NDU with both the transformable and the separable kernel gave the smoothest results compared to Ext and Khype. Recall that the kernels used by NDU account for adjacency effects through their input vector $\boldsymbol{v}_n$. The third column in Figure 5.5 corresponds to the results obtained with MM 3, i.e. with bilinear contributions, adjacency effects, and band selectivity. In accordance with MM 3, the first figure in column 3 shows that the true nonlinear contributions are smooth, they are the most pronounced at the center of the spectrum and they are attenuated (almost zero) at the extremities of the spectrum. In this case, NDU with the separable kernel gave the best results. All the methods estimated the highest nonlinear contributions at the center of the spectrum. However, NDU with the separable kernel handled the attenuation of the bilinear contributions at the extremities of the spectrum better than the other methods. This is probably due to the fact that the prior information on the similarities between the nonlinear contributions at different bands is better in the case of the separable kernel than in the transformable kernel. More precisely, the separable kernel exploits the linear graph structure which promotes smooth variations at adjacent bands (i.e. adjacent rows in the image). Whereas, the transformable kernel exploits the correlations between all bands in order to estimate the nonlinear contributions.

5.4.2 Real data: Gulf of Lion

Data set description and ground truth

The second set of experiments considers real data estimated by the Meris spectrometer and captured over the gulf of Lion in the south east of France. The image has 280×330 pixels, 13 spectral bands in the range $400 - 800$ nm, and a spatial resolution of 300 m. This data set will be referred to as the "Meris" image in the following and is depicted in Figure 5.6 (a). Furthermore, Figure 5.6 (b) shows the corresponding classification map provided by corine land cover (CLC) database[1]. Note that the two images were coregistered, and the classification map was chosen as close as possible to the date of the Meris image in order to have a consistent comparison. The

[1] www.statistiques.developpement-durable.gouv.fr/clc/carte/metropole

Table 5.2: RMSE ($\times 10^{-2}$) and optimal parameters obtained with MM 1. The first two terms in brackets are the RMSE of the estimated abundances (left term) and the nonlinear part (right term), and the second two terms in the lower brackets are the optimal values for λ (left term) and μ (right term).

	$SNR = 40$			$SNR = 30$			$SNR = 20$		
	$M = 3$	$M = 4$	$M = 5$	$M = 3$	$M = 4$	$M = 5$	$M = 3$	$M = 4$	$M = 5$
Ext	$(5.31, 2.35)$	$(4.98, 2.84)$	$(4.51, 2.63)$	$(7.38, 3.75)$	$(8.30, 5.84)$	$(7.19, 5.01)$	$(13.77, 8.06)$	$(15.28, 13.96)$	$(13.84, 11.22)$
Khype	$(2.01, 0.88)$	$(2.72, 1.14)$	$(2.60, 1.12)$	$(3.77, 1.55)$	$(4.68, 1.50)$	$(4.63, 1.84)$	$(9.52, 3.73)$	$(9.55, 3.12)$	$(9.33, 3.10)$
(G)	$(0.1, 0.01)$	$(0.01, 10^{-3})$	$(0.01, 10^{-3})$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(1, 0.1)$	$(1, 0.1)$	$(1, 0.1)$
Khype	$(2.79, 1.03)$	$(2.21, 1.20)$	$(2.08, 1.09)$	$(4.05, 1.55)$	$(4.21, 1.45)$	$(4.47, 1.74)$	$(8.73, 3.34)$	$(9.34, 2.91)$	$(9.01, 2.89)$
(P)	$(0.01, 10^{-3})$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(1, 0.1)$	$(1, 0.1)$	$(1, 0.1)$
NDU.	$(1.22, 0.54)$	$(1.50, 0.35)$	$(1.60, 0.29)$	$(2.82, 0.77)$	$(3.53, 0.95)$	$(3.65, 1.14)$	$(6.07, 2.26)$	$(7.79, 1.90)$	$(7.76, 2.11)$
(Tr. + G)	$(1, 0.01)$	$(1, 0.01)$	$(1, 0.01)$	$(10, 0.1)$	$(1, 10^{-4})$	$(10, 0.01)$	$(10, 10^{-4})$	$(10, 0.01)$	$(10, 0.01)$
NDU	$(1.43, 0.55)$	$(1.27, 0.26)$	$(1.35, 0.34)$	$(2.11, 0.76)$	$(3.12, 0.52)$	$(3.30, 0.67)$	$(5.55, 1.71)$	$(7.54, 1.66)$	$(7.87, 1.78)$
(Tr.+P)	$(10^{-4}, 10^{-4})$	$(0.1, 10^{-4})$	$(0.1, 10^{-4})$	$(1, 10^{-4})$	$(1, 10^{-4})$	$(1, 10^{-4})$	$(10, 10^{-4})$	$(10, 0.01)$	$(10, 0.01)$
NFU	$(1.79, 0.69)$	$(1.53, 0.52)$	$(1.98, 0.68)$	$(2.30, 1.36)$	$(3.81, 0.83)$	$(3.63, 0.89)$	$(6.17, 1.89)$	$(7.51, 2.73)$	$(7.71, 1.16)$
(Sp+G)	$(0.01, 10^{-4})$	$(0.01, 10^{-3})$	$(0.01, 10^{-3})$	$(0.01, 10^{-4})$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(1, 0.01)$	$(0.1, 0.01)$	$(1, 0.1)$
NDU	$(1.17, 0.43)$	$(1.97, 0.72)$	$(2.15, 0.77)$	$(2.02, 0.85)$	$(3.73, 1.27)$	$(3.55, 0.98)$	$(5.79, 1.61)$	$(7.44, 1.65)$	$(7.61, 1.39)$
(Sp+P)	$(10^{-3}, 10^{-4})$	$(10^{-3}, 10^{-4})$	$(10^{-3}, 10^{-4})$	$(0.01, 10^{-3})$	$(0.1, 0.01)$	$(0.01, 10^{-3})$	$(1, 0.1)$	$(1, 0.1)$	$(1, 0.1)$

Table 5.3: RMSE ($\times 10^{-2}$) and optimal parameters obtained with MM 2. The first two terms in brackets are the RMSE of the estimated abundances (left term) and the nonlinear part (right term), and the second two terms in the lower brackets are the optimal values for λ (left term) and μ (right term).

	$SNR = 40$			$SNR = 30$			$SNR = 20$		
	$M = 3$	$M = 4$	$M = 5$	$M = 3$	$M = 4$	$M = 5$	$M = 3$	$M = 4$	$M = 5$
Ext	$(4.27, 2.78)$	$(4.49, 3.32)$	$(3.80, 2.85)$	$(6.72, 3.44)$	$(8.03, 5.19)$	$(7.25, 4.65)$	$(12.98, 6.97)$	$(16.11, 12.57)$	$(13.57, 10.93)$
Khype	$(2.55, 0.99)$	$(2.33, 0.79)$	$(2.18, 0.88)$	$(4.37, 1.67)$	$(4.75, 1.34)$	$(5.07, 1.65)$	$(10.37, 4.18)$	$(9.41, 3.02)$	$(10.12, 2.93)$
(G)	$(0.01, 10^{-3})$	$(0.01, 10^{-3})$	$(0.01, 10^{-3})$	$(0.1, 10^{-3})$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(1, 0.01)$	$(1, 0.1)$	$(1, 0.1)$
Khype	$(2.61, 1.05)$	$(2.26, 0.99)$	$(2.35, 1.14)$	$(3.76, 1.43)$	$(4.42, 1.27)$	$(5.06, 1.68)$	$(9.10, 3.30)$	$(9.16, 2.85)$	$(10.10, 2.85)$
(P)	$(0.1, 0.01)$	$(0.1, 10^{-3})$	$(0.01, 10^{-3})$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(1, 0.1)$	$(1, 0.1)$
NDU	$(0.92, 0.64)$	$(1.54, 0.38)$	$(1.39, 0.33)$	$(1.97, 0.63)$	$(3.51, 0.60)$	$(3.93, 0.79)$	$(6.98, 2.53)$	$(8.51, 2.70)$	$(9.05, 2.71)$
(Tr.+G)	$(10, 0.01)$	$(1, 10^{-4})$	$(1, 10^{-4})$	$(10, 10^{-3})$	$(10, 10^{-3})$	$(10, 0.01)$	$(10, 10^{-4})$	$(10, 0.1)$	$(10, 0.01)$
NDU	$(1.75, 0.74)$	$(1.73, 0.43)$	$(1.62, 0.38)$	$(2.76, 1.04)$	$(3.42, 0.60)$	$(4.08, 0.73)$	$(5.82, 1.72)$	$(8.12, 2.10)$	$(9.14, 2.52)$
(Tr+P)	$(1, 0.01)$	$(1, 10^{-4})$	$(1, 10^{-4})$	$(10, 0.01)$	$(10, 10^{-4})$	$(10, 0.01)$	$(10, 10^{-4})$	$(10, 0.1)$	$(10, 0.01)$
NDU	$(1.92, 0.67)$	$(2.35, 0.84)$	$(1.87, 0.73)$	$(2.46, 1.40)$	$(3.71, 0.92)$	$(4.04, 1.03)$	$(5.67, 1.56)$	$(7.60, 1.38)$	$(8.93, 1.32)$
(Sp.+G)	$(0.01, 10^{-4})$	$(0.01, 10^{-4})$	$(0.01, 10^{-4})$	$(0.01, 10^{-4})$	$(0.1, 10^{-3})$	$(0.1, 10^{-3})$	$(1, 0.01)$	$(1, 0.1)$	$(1, 0.1)$
NDU	$(1.14, 0.39)$	$(2.21, 0.80)$	$(1.70, 0.61)$	$(2.12, 1.42)$	$(3.45, 0.83)$	$(3.91, 0.78)$	$(5.49, 1.85)$	$(7.36, 1.13)$	$(8.80, 1.03)$
(Sp.+P)	$(10^{-3}, 10^{-4})$	$(0.01, 10^{-4})$	$(0.01, 10^{-3})$	$(10^{-3}, 10^{-4})$	$(0.1, 10^{-3})$	$(0.01, 10^{-3})$	$(0.1, 10^{-3})$	$(1, 0.1)$	$(1, 0.1)$

Table 5.4: RMSE ($\times 10^{-2}$) and optimal parameters obtained with MM 3. The first two terms in brackets are the RMSE of the estimated abundances (left term) and the nonlinear part (right term), and the second two terms in the lower brackets are the optimal values for λ (left term) and μ (right term).

	$SNR = 40$			$SNR = 30$			$SNR = 20$		
	$M = 3$	$M = 4$	$M = 5$	$M = 3$	$M = 4$	$M = 5$	$M = 3$	$M = 4$	$M = 5$
Ext	$(9.21, 12.8)$	$(12.1, 15.7)$	$(8.97, 12.6)$	$(10.7, 13.0)$	$(12.7, 15.6)$	$(11.3, 15.5)$	$(17.7, 16.8)$	$(18.1, 21.0)$	$(15.9, 19.3)$
Khype	$(4.33, 2.18)$	$(3.55, 1.99)$	$(3.47, 1.91)$	$(5.80, 2.60)$	$(5.28, 2.20)$	$(5.48, 2.19)$	$(10.56, 5.21)$	$(9.59, 3.08)$	$(8.90, 3.53)$
(G)	$(0.1, 10^{-3})$	$(0.1, 10^{-3})$	$(0.1, 10^{-3})$	$(0.1, 10^{-3})$	$(0.1, 0.01)$	$(0.1, 0.01)$	$(10, 0.1)$	$(1, 0.01)$	$(10, 0.1)$
Khype	$(1.24, 2.63)$	$(4.34, 2.15)$	$(3.85, 2.30)$	$(2.56, 2.71)$	$(5.26, 2.24)$	$(4.85, 2.28)$	$(7.66, 3.31)$	$(9.44, 2.86)$	$(8.74, 3.33)$
(P)	$(1, 10^{-3})$	$(1, 0.01)$	$(1, 10^{-3})$	$(1, 0.01)$	$(1, 10^{-3})$	$(1, 0.01)$	$(1, 0.01)$	$(1, 0.1)$	$(10, 0.1)$
NDU	$(4.09, 1.95)$	$(3.20, 1.68)$	$(3.17, 1.68)$	$(4.39, 1.97)$	$(4.33, 1.78)$	$(4.63, 1.72)$	$(8.72, 3.25)$	$(10.06, 3.22)$	$(8.84, 3.15)$
(Tr.+G)	$(10, 10^{-4})$	$(10, 10^{-4})$	$(10, 10^{-3})$	$(10, 10^{-4})$	$(10, 10^{-4})$	$(10, 10^{-4})$	$(10, 10^{-4})$	$(10, 0.01)$	$(10, 0.01)$
NDU	$(5.96, 2.77)$	$(5.62, 2.30)$	$(5.32, 2.21)$	$(6.33, 2.80)$	$(6.62, 2.41)$	$(5.83, 2.31)$	$(10.15, 3.72)$	$(10.31, 3.38)$	$(8.81, 2.93)$
(Tr+P)	$(10, 0.1)$	$(10, 0.01)$	$(10, 10^{-4})$	$(10, 0.1)$	$(10, 0.01)$	$(10, 0.01)$	$(10, 0.1)$	$(10, 0.1)$	$(10, 0.1)$
NDU	$(1.29, 0.50)$	$(2.62, 1.03)$	$(1.75, 0.72)$	$(1.98, 0.69)$	$(3.40, 0.66)$	$(3.46, 0.68)$	$(6.29, 1.53)$	$(8.43, 1.62)$	$(7.99, 1.95)$
(Sp+G)	$(0.1, 10^{-4})$	$(0.1, 10^{-3})$	$(0.1, 10^{-4})$	$(0.1, 10^{-4})$	$(0.1, 10^{-4})$	$(1, 0.01)$	$(1, 0.01)$	$(10, 0.1)$	$(10, 0.1)$
NDU	$(0.95, 0.34)$	$(1.56, 0.42)$	$(1.51, 0.34)$	$(1.96, 0.41)$	$(3.25, 0.50)$	$(3.46, 0.48)$	$(6.15, 1.77)$	$(8.15, 0.98)$	$(7.64, 1.07)$
(Sp+P)	$(0.01, 10^{-4})$	$(0.01, 10^{-3})$	$(0.01, 10^{-4})$	$(0.1, 10^{-3})$	$(0.1, 10^{-4})$	$(0.1, 10^{-3})$	$(0.1, 0.01)$	$(1, 0.1)$	$(1, 0.1)$

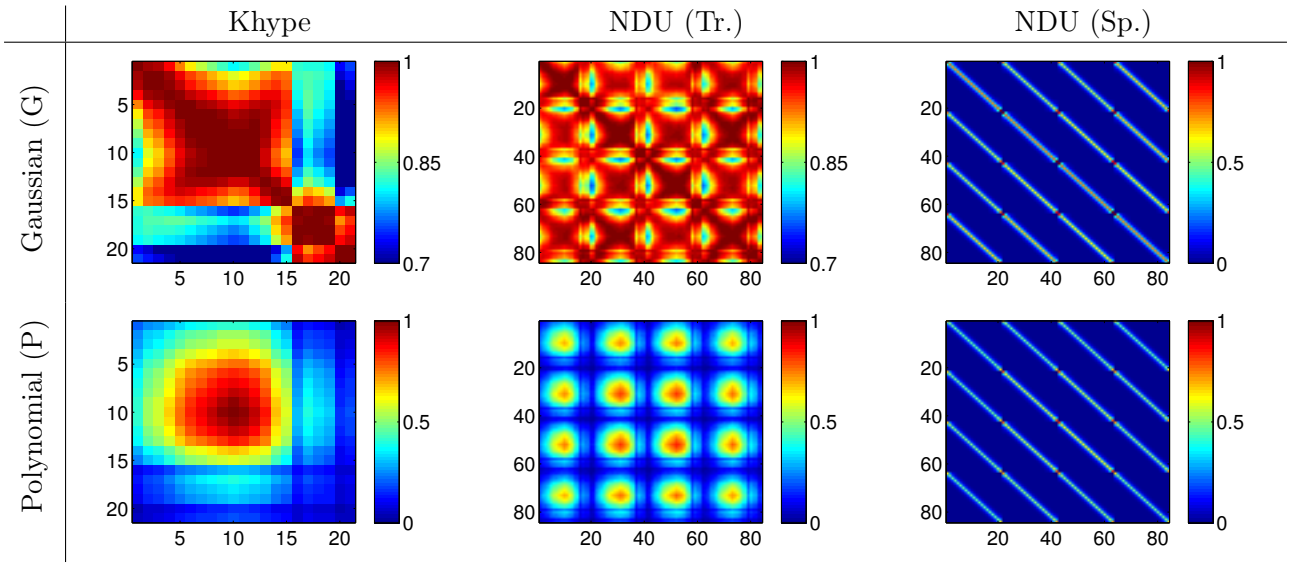


Figure 5.4: Gram matrices obtained with $M = 3$ and SNR $= 40$ dB. First column: Gram matrices used by Khype, second column: left corners of the Gram matrices obtained using a transformable kernel, and third column: left corners of the Gram matrices obtained using a separable kernel. The first and the second row correspond to the Gaussian and polynomial kernels respectively.

classification map will be used for visual evaluation, in order to better evaluate and interpret the unmixing results provided by the various algorithms.

The CLC classification map has a spatial resolution approximately 10 times greater than the Meris data set's spatial resolution. Therefore, it was downscaled in order to obtain spatial abundance maps for the Meris image [Zhukov et al., 1999]. The spatial abundance of a certain class/endmember in a pixel in the estimated low resolution image is the average of the corresponding class occurrences in the corresponding window in the high resolution image. These maps are regarded as a potential visual ground truth that allow to better evaluate and interpret the unmixing results. The first row in Figure 5.8 shows the ground truth obtained from the CLC classification map, which corresponds to the proportions of three classes: water, agricultural areas, and forests and semi natural areas.

Unmixing results

We extracted 3 and 4 endmembers using virtual component analysis (VCA) [Nascimento and Bioucas-Dias, 2005]. We noticed that each time one of the endmembers extracted by VCA was not meaningful in the sense that the corresponding abundance map was not spatially coherent. Nevertheless, in the case with 4 extracted endmembers the three abundance maps corresponding to the meaningful endmembers were relatively in accordance with the estimated ground truth.

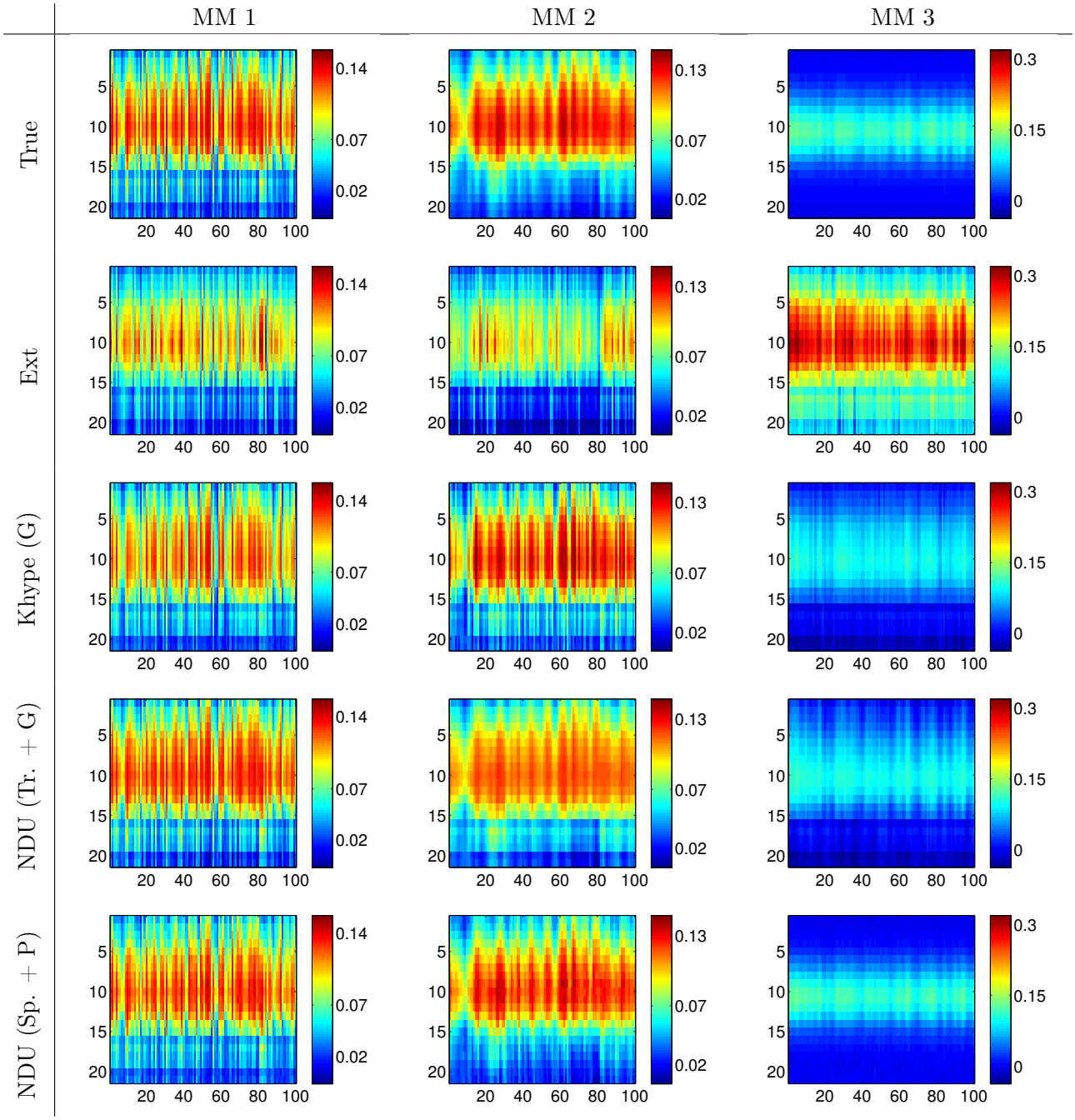
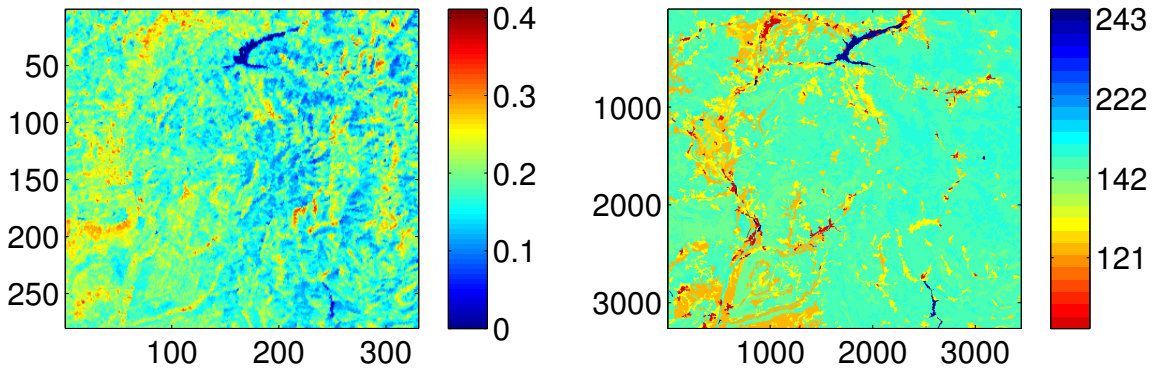


Figure 5.5: True and estimated nonlinear contributions in all pixels at all bands obtained with $M = 3$ and SNR=40 dB, the vertical and horizontal axis in each figure represent the frequency band and the pixel number respectively.

In what follows we only show the results obtained with four endmembers. Figure 5.7 shows the estimated endmembers spectra, note that endmember 4 corresponds to the outlier. The abundance maps for each endmember were estimated using the fully constrained least squares approach (FCLS), Ext, Khype and NDU. The separable kernel was used with NDU with a linear graph as in the experiments with synthetic data. Both Khype and NDU were tested using a Gaussian and a second order homogeneous polynomial kernel. As in the previous section, $\boldsymbol{v}_n$ was defined using the pixels and its neighboring pixels spectra in particular the left and right neighbors were chosen. The tuning parameters λ and μ were set to 10 and 10^{-3} respectively for both Khype and NDU. Unlike Khype and Ext that were applied on each pixel separately, the image was divided into 10×10 patches and NDU was applied on each patch.

Table 5.5 reports the root mean square error and the average spectral angle between the available observations and the reconstructed spectra denoted as $\text{RMSE}_{\boldsymbol{S}}$ and ASA respectively. These two evaluation metrics were previously defined in (3.30) and (3.33) respectively in chapter 3. Table 5.5 shows that NDU scored the best results in terms of both the RMSE and the ASA. The results obtained with Ext slightly improved the ones obtained with FCLS. Khype outperformed both methods, Ext and FCLS, and had results very close to the ones obtained with NDU. Figure 5.8 shows that the abundance maps estimated by the various algorithms are rather similar. As mentioned previously, the fourth endmember corresponds to noise hence its abundance maps are not shown in Figure 5.8. Figure 5.9 shows the nonlinear part estimated by NDU and Khype at band 10. In fact, most of the areas where nonlinear contributions appear are mainly located on the boundaries of agricultural areas (endmember 2) surrounded by forests and semi-natural areas (endmember 3). Note that the nonlinear contributions estimated by NDU are relatively spatially smoother than the ones estimated by Khype. However, NDU results exhibit some artifacts due to the fact that the image was partitioned into square patches.

Finally, Figure 5.10 compares the nonlinear contribution estimated by NDU and Khype at all bands for the pixels delimited by rows 171 and 180 and columns 231 and 240. Both algorithms estimated the highest nonlinear contributions for this particular region. However, the nonlinear contributions have different variations throughout the spectral bands. NDU estimated the highest nonlinear contributions at the higher frequency bands whereas Khype estimated almost the same level of nonlinearity at all frequency bands. Furthermore, NDU estimated smooth nonlinear contributions at adjacent pixels compared to Khype. It can be concluded that NDU captures more spectral variability throughout the spectral bands and that it provides smooth nonlinear contributions at adjacent pixels.



(a) Band 10 of Meris data (b) Classification map

Figure 5.6: Fom left to right: (a) band 10 of the Meris data set, (b) corresponding CLC classification map (See classes names in Appendix A).

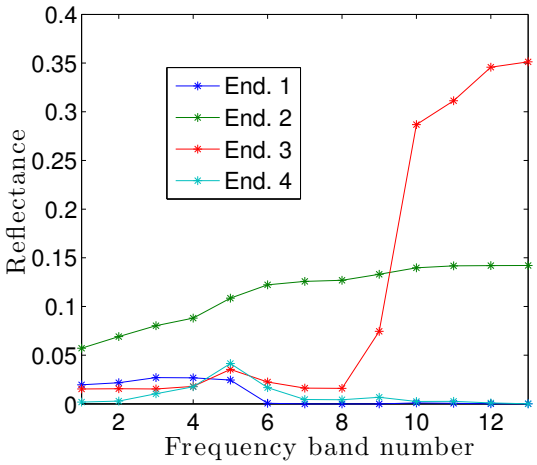


Figure 5.7: Endmembers spectra estimated by VCA for the Meris data set.

5.5 Conclusion

This chapter proposed a new kernel based nonlinear mixing model for hyperspectral data. The proposed vector-valued function is able to account for band dependent and neighboring nonlinear contributions. The proposed framework has several characteristics. It allows to handle in a unified framework several types of nonlinearities depending on the choice of the kernel function. Unlike nonlinear models proposed in the literature, it considers a different nonlinear function for each spectral band. The fact that the nonlinear function acts on the reflectance vectors observed in the corresponding pixel and its neighbors is intended to account for nonlinearities originating from the ground cover of the pixel and its neighbors. Furthermore, the separable kernel design can be used to incorporate prior information regarding the similarities between nonlinear contributions at different bands. In particular, a linear graph was proposed to promote

Table 5.5: Root mean square error $\text{RMSE}_{\boldsymbol{S}}$ ($\times 10^{-2}$) and average spectral angle (ASA) in radian of the reconstructed spectra obtained with the Meris data set.

	FCLS	Ext	Khype (G)	Khype (P)	Sep. (G)	Sep. (P)
$\text{RMSE}_{\boldsymbol{S}}$	1.14	1.13	0.66	1.04	0.41	0.46
ASA	0.0393	0.0343	0.0338	0.0381	0.0153	0.0204

smooth nonlinear variations between adjacent bands. The performance of the proposed approach was validated on synthetic and real data estimated by the Meris spectrometer and captured over the gulf of Lion in the south east of France. Finally, note that the proposed approach requires partitioning the image into patches which can result in artifacts in the estimated nonlinear part. Future work should aim at attenuating those artifacts through an extension of the vector-valued approach. More precisely, the vector-valued framework can be adapted such as to promote smoothness between the estimated nonlinear contributions at adjacent bands and between the nonlinear functions at adjacent patches simultaneously. In the next chapter, we will see how the vector-valued RKHS framework can be used to promote smooth variations of the nonlinear component in a kernel-based nonlinear mixing model.

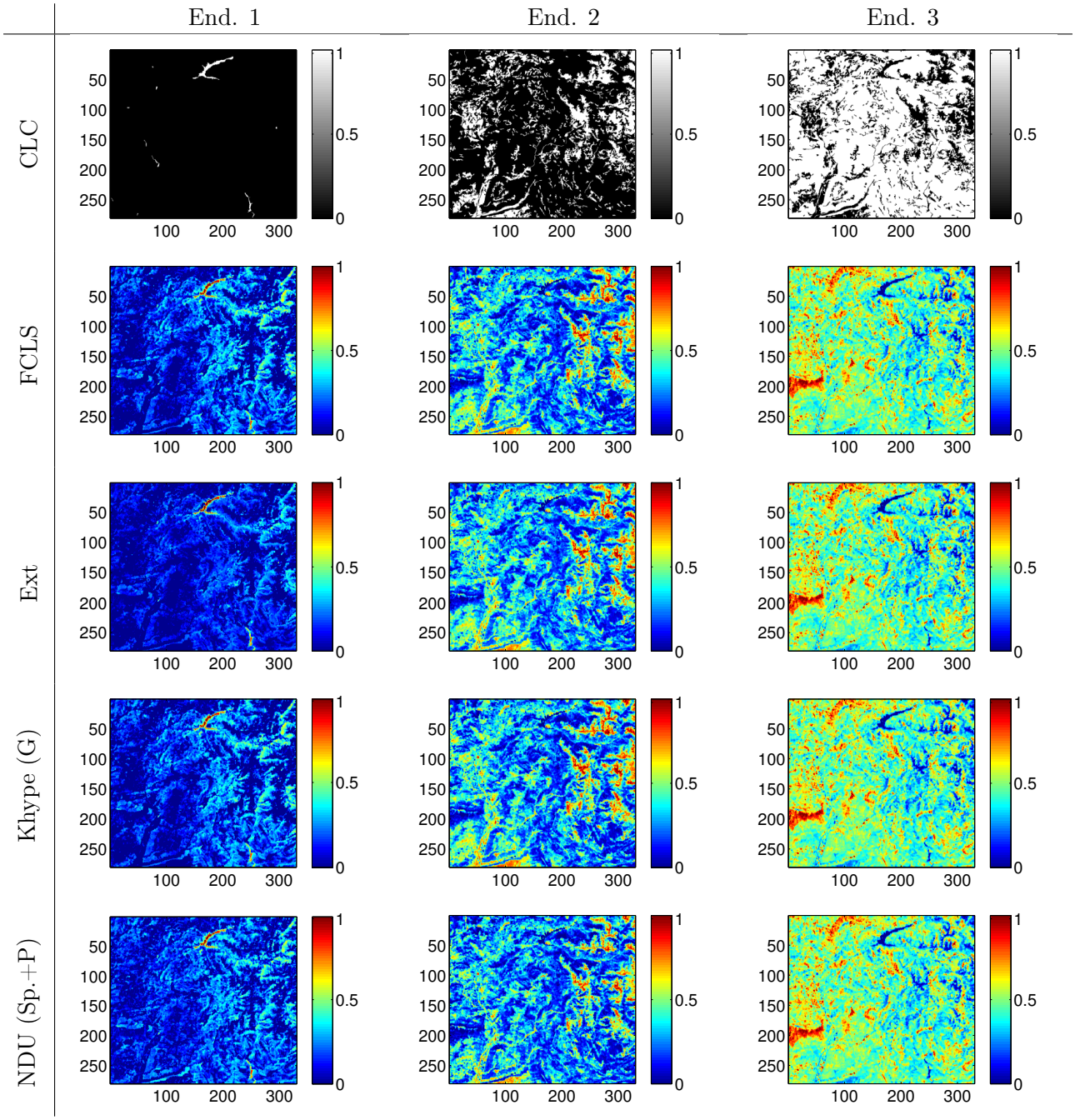
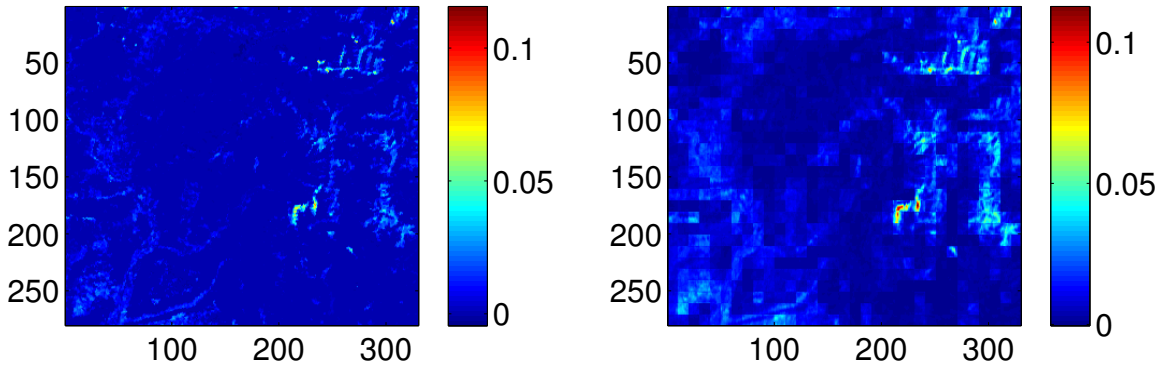
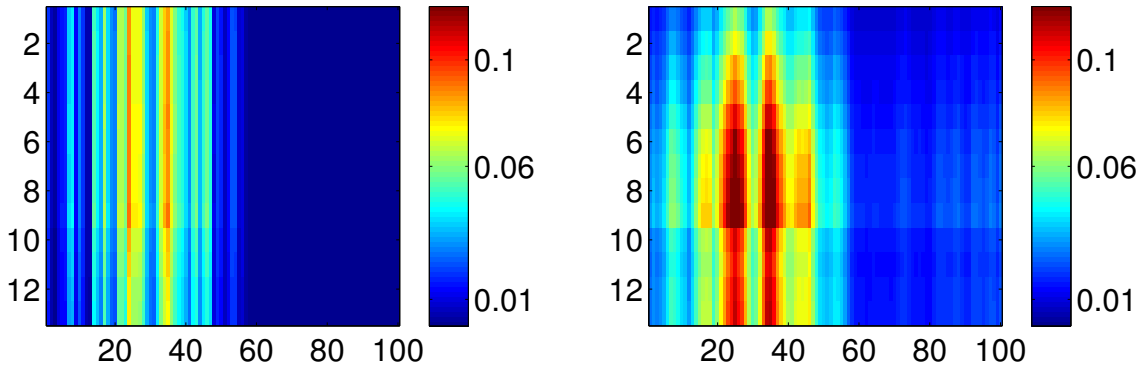


Figure 5.8: Abundance maps of the first three endmembers obtained with VCA and corresponding to the Meris real data set. The abundance maps of End. 1, 2, and 3 correspond to water, agricultural areas, and forests and semi natural areas respectively.



(a) Khype (G) (b) NDU (Sp.+P)

Figure 5.9: Nonlinear contributions at all pixels at band 10 obtained with: (a) Khype used with a Gaussian kernel (G) and (b) NDU used with a separable and polynomial kernel (Sp.+P).



(a) Khype (G) (b) NDU (Sp.+P)

Figure 5.10: Nonlinear contributions at all bands in some of the pixels in the Meris data set. The horizontal and vertical axis correspond to the frequency and pixel index respectively.

Table 5.6: Classes in classification map of Meris data set.

Code	Color	description
1		**Artificial surfaces**
1.1		**Urban fabric**
1.1.1		Continuous urban fabric
1.1.2		Discontinuous urban fabric
1.2		**Industrial, commercial and transport units**
1.2.1		Industrial or commercial units
1.2.2		Road and rail networks and associated land
1.2.4		Airports
1.3		**Mine, dump and construction sites**
1.3.1		Mineral extraction sites
1.4		**Artificial, non-agricultural vegetated areas**
1.4.2		Sport and leisure facilities Not translated
2		**Agricultural areas**
2.1		**Arable land**
2.1.1		Non-irrigated arable land
2.2		**Permanent crops**
2.2.1		Vineyards
2.2.2		Fruit trees and berry plantations
2.3		**Pastures**
2.3.1		Pastures
2.4		**Heterogeneous agricultural areas**
2.4.2		Complex cultivation patterns
2.4.3		Land principally occupied by agriculture
2.4.4		Agro-forestry areas
3		**Forest and seminatural areas**
3.1		**Forests**
3.1.1		Broad-leaved forest
3.1.2		Coniferous forest
3.1.3		Mixed forest
3.2		**and/or herbaceous vegetation associations**
3.2.1		Natural grasslands
3.2.2		Moors and heathland
3.2.3		Sclerophyllous vegetation
3.2.4		Transitional woodland-shrub
3.3		**Open spaces with little or no vegetation**
3.3.1		Beaches, dunes, sands
3.3.2		Bare rocks
3.3.3		Sparsely vegetated areas
3.3.5		Glaciers and perpetual snow
5		**Water bodies**
5.1		**Inland waters**
5.1.1		Water courses
5.1.2		Water bodies

Chapter 6

Spatial regularization for nonlinear unmixing

This chapter has been adapted from the conference paper [Ammanouil et al., 2016b].

This chapter introduces a new framework for incorporating spatial regularization into a nonlinear unmixing procedure dedicated to hyperspectral data. The proposed model promotes smooth spatial variations of the nonlinear component in the mixing model. The spatial regularizer and the nonlinear contributions are jointly modelled by a vector-valued function that lies in a reproducing kernel Hilbert space (RKHS). The unmixing problem is strictly convex and reduces to a quadratic programming (QP) problem. Simulations on synthetic data illustrate the effectiveness of the proposed approach.

6.1 Introduction

In chapter 5, we proposed a nonlinear mixing model that takes into account band dependent and neighboring nonlinear contributions. This was done using tools from the theory of vector-valued functions in reproducing kernel Hilbert spaces (RKHS). Roughly speaking, we regularized the nonlinear contributions at different spectral bands. In this chapter, we propose to use the same tool used in chapter 5, namely vector-valued functions in RKHS, in order to spatially regularize the nonlinear contribution in a kernel based nonlinear mixing model. This is done by introducing a new spatial regularizer acting on the nonlinear contributions at different pixels. Similarly to chapter 5, we depart from the kernel-based nonlinear model proposed in [Chen et al., 2013]

known as Khype. Compared to [Chen et al., 2013], we go one step further by using an RKHS of vector-valued functions rather than scalar-valued functions [Evgeniou et al., 2005, Alvarez et al., 2012]. More precisely, each output of the vector-valued function represents the nonlinear contribution of the mixing model at a given pixel. We consider a special class of kernels, known as separable kernels [Alvarez et al., 2012] that we have introduced in chapter 5. In particular, these kernels are defined as the product of two terms, a scalar-valued kernel acting on the input, and a matrix-valued kernel encoding the closeness between the outputs. The first and the second term play a central role in defining the nonlinearity and the spatial regularization, respectively. The closeness between the outputs of the function, i.e. the nonlinear contributions at different pixels, is defined using a graph. In particular, the proposed spatial regularization consists of penalizing the ℓ_2-norm of the difference between the outputs that appear as connected in the graph and hence promotes smoothness. Being solely defined by an appropriate design of the kernel, the spatial regularization is relatively transparent from the optimization problem point of view, which is then shown to reduce to a quadratic problem.

To the best of our knowledge, there is no nonlinear model in the literature that promotes smooth nonlinear contributions. Nevertheless, several works considered other types of prior information for nonlinear unmixing. The authors of [Chen et al., 2014] incorporated a total variation over the abundances, which promotes piecewise smooth abundances in the scene. The authors of [Févotte and Dobigeon, 2015] introduced a robust nonlinear matrix factorization unmixing algorithm that promotes sparse nonlinear contributions. In contrast with the previously cited models, we promote smooth nonlinear contributions over the scene. This prior is justified by the spatial smoothness inherently present in natural scenes.

This chapter is organized as follows. Section 6.2 introduces the nonlinear mixing model. Section 6.3 details the kernel design and the underlying spatial regularization. Section 6.4 presents the proposed unmixing algorithm. Finally, experimental results investigated in section 6.5 show the effectiveness of the proposed approach.

6.2 Vector-valued formulation

Consider an hyperspectral image with N pixels, estimated over L spectral bands. For self containment of this chapter, we recall the LMM [Heinz and Chang, 2001] where the spectrum of the n-th pixel is modeled as:

$$\boldsymbol{s}_n = \sum_{i=1}^{M} a_{i,n}\boldsymbol{r}_i + \boldsymbol{e}_n, \forall n = 1, \dots N, \tag{6.1}$$

Table 6.1: Notations for chapter 6

$g_n(\boldsymbol{r}_{\lambda_\ell})$	1×1	Nonlinear contribution in n-th pixel, ℓ-th band (used in Khype)
$\boldsymbol{g}(\boldsymbol{r}_{\lambda_\ell})$	$N \times 1$	Nonlinear contribution in all pixels, ℓ-th band (used in prop. model)
$\overline{\boldsymbol{k}}(\boldsymbol{r}_{\lambda_\ell}, \boldsymbol{r}_{\lambda_{\ell'}})$	$N \times N$	Matrix-valued kernel
$k(\boldsymbol{r}_{\lambda_\ell}, \boldsymbol{r}_{\lambda_{\ell'}})$	1×1	Scalar kernel (used in separable kernel design)
$\overline{\boldsymbol{K}}$	$NL \times NL$	Gram matrix associated with $\overline{\boldsymbol{k}}(\boldsymbol{r}_{\lambda_\ell}, \boldsymbol{r}_{\lambda_{\ell'}})$
$\boldsymbol{K}$	$L \times L$	Gram matrix associated with $k(\boldsymbol{r}_{\lambda_\ell}, \boldsymbol{r}_{\lambda_{\ell'}})$
$\overline{\mathcal{H}}_{\boldsymbol{k}}$	–	RKHS associated with $\overline{\boldsymbol{k}}$ ($\boldsymbol{g} \in \overline{\mathcal{H}}_{\boldsymbol{k}}$)
$\mathcal{H}_k$	–	RKHS associated with k ($g_n \in \mathcal{H}_k$)
$\overline{\boldsymbol{\mathcal{E}}}$	$N \times N$	Positive semi-definite matrix (used in separable kernel design)
$\overline{\boldsymbol{\mathcal{W}}}$	$N \times N$	Adjacency matrix of the graph (representing the pixels)

where $\boldsymbol{s}_n = [s_{1,n}, \ldots, s_{L,n}]^\top$ is the L-dimensional spectrum of the n-th pixel, M is the number of endmembers, $a_{i,n}$ is the abundance of the i-th endmember in the n-th pixel, $\boldsymbol{r}_i$ is the L-dimensional spectrum of the i-th endmember, and $\boldsymbol{e}_n$ is a vector of white Gaussian noise. All vectors are column vectors. The abundances, being the relative contributions of the endmembers, are positive and usually sum to one [Heinz and Chang, 2001], namely: $a_{i,n} \geq 0$ and $\sum_{i=1}^{M} a_{i,n} = 1$. As mentioned previously, most nonlinear mixing models incorporate an additional term within the LMM (6.1). We depart from the nonlinear mixing model, known as Khype, that was proposed in [Chen et al., 2013]:

$$s_{\ell,n} = \boldsymbol{r}_{\lambda_\ell}^\top \boldsymbol{a}_n + g_n(\boldsymbol{r}_{\lambda_\ell}) + e_{\ell,n}, \tag{6.2}$$

where $\boldsymbol{R} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_M]$ is the $L \times M$ matrix of endmembers, $\boldsymbol{r}_{\lambda_\ell}$ is an $M \times 1$ vector formed with the elements of the ℓ-th row of $\boldsymbol{R}$, $\boldsymbol{a}_n = [a_{1,n}, \ldots, a_{M,n}]^\top$ is the abundance vector of the n-th pixel, and g_n is a scalar-valued function in an RKHS modeling the nonlinearity at any band. Let $\boldsymbol{g} = [g_1, \ldots, g_N]^\top$ be a vector-valued function. Equation (6.2) can be rewritten as follows:

$$\boldsymbol{s}_{\lambda_\ell} = \boldsymbol{A}^\top \boldsymbol{r}_{\lambda_\ell} + \boldsymbol{g}(\boldsymbol{r}_{\lambda_\ell}) + \boldsymbol{e}_{\lambda_\ell}, \tag{6.3}$$

where $\boldsymbol{s}_{\lambda_\ell}$ and $\boldsymbol{e}_{\lambda_\ell}$ denote the ℓ-th rows of $\boldsymbol{S} = [\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N]$ and $\boldsymbol{E} = [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_N]$ respectively. The aim of the next section is to show the relevance of the vector-valued formulation (6.3). In particular, we demonstrate the ability of vector-valued functions to incorporate prior information about the similarities between $g_1, \ldots, g_N$, the outputs of $\boldsymbol{g}$, through an appropriate kernel design.

6.3 Kernel design and regularization

We will assume that the nonlinear function $\boldsymbol{g}$ in (6.3) lies in an RKHS of vector-valued functions, denoted by $\overline{\mathcal{H}}_{\boldsymbol{k}}$, associated with the following separable kernel function [Alvarez et al., 2012, Evgeniou et al., 2005]:

$$\begin{aligned} \bar{\boldsymbol{k}}: \quad \mathbb{R}^M \times \mathbb{R}^M &\rightarrow \mathbb{R}^{N\times N} \\ (\boldsymbol{r}_{\lambda_\ell}, \boldsymbol{r}_{\lambda_{\ell'}}) &\rightarrow \bar{\boldsymbol{k}}(\boldsymbol{r}_{\lambda_\ell}, \boldsymbol{r}_{\lambda_{\ell'}}), \end{aligned} \tag{6.4}$$

with

$$\overline{\boldsymbol{k}}(\boldsymbol{r}_{\lambda_\ell}, \boldsymbol{r}_{\lambda_{\ell'}}) = k(\boldsymbol{r}_{\lambda_\ell}, \boldsymbol{r}_{\lambda_{\ell'}})\, \overline{\boldsymbol{\mathcal{E}}}. \tag{6.5}$$

The function $k(\cdot,\cdot)$ is a scalar-valued kernel such as the polynomial or Gaussian kernel, and $\overline{\boldsymbol{\mathcal{E}}}$ is an $N \times N$ symmetric nonnegative matrix. Let $\boldsymbol{K}$ be the $L \times L$ Gram matrix associated with the scalar-valued kernel k, namely, $k_{\ell,\ell'} = k(\boldsymbol{r}_{\lambda_\ell}, \boldsymbol{r}_{\lambda_{\ell'}})$, and let $\overline{\boldsymbol{K}}$ be the $NL \times NL$ Gram matrix associated with the matrix-valued kernel $\overline{\boldsymbol{k}}$, namely, $\overline{\boldsymbol{k}}_{\ell,\ell'} = \overline{\boldsymbol{k}}(\boldsymbol{r}_{\lambda_\ell}, \boldsymbol{r}_{\lambda_{\ell'}})$. Given equation (6.5), we have:

$$\overline{\boldsymbol{K}} = \boldsymbol{K} \otimes \overline{\boldsymbol{\mathcal{E}}}. \tag{6.6}$$

Moreover, the norm of $\boldsymbol{g}$ in $\overline{\mathcal{H}}_{\boldsymbol{k}}$ [Evgeniou et al., 2005] is given by:

$$\|\boldsymbol{g}\|^2_{\overline{\mathcal{H}}_{\boldsymbol{k}}} = \sum_{n,n'=1}^{N} \overline{\mathcal{E}}^{\dagger}_{n,n'} \langle g_n, g_{n'} \rangle_{\mathcal{H}_k}, \tag{6.7}$$

where $\overline{\boldsymbol{\mathcal{E}}}^{\dagger}$ is the pseudo inverse of $\overline{\boldsymbol{\mathcal{E}}}$. The above expression shows that the norm of $\boldsymbol{g}$ is equal to the weighted sum of the pairwise inner products between the individual functions. From a regularization point of view, equation (6.7) can be used to promote structured similarities between the different functions through the design of $\overline{\boldsymbol{\mathcal{E}}}$. Hereafter, we investigate the so-called "graph regularizer" [Evgeniou et al., 2005] and provide the corresponding structure for the matrix $\overline{\boldsymbol{\mathcal{E}}}$. Note that the authors of [Evgeniou et al., 2005] provided other examples of regularizers with the corresponding design of $\overline{\boldsymbol{\mathcal{E}}}$.

Due to the inherent spatial correlation present in real images, spatially neighboring pixels usually have similar spectra. We assume that they are also characterized by similar nonlinear contributions. This prior about the closeness between adjacent pixels can be modeled by a graph. We denote by $\overline{\boldsymbol{\mathcal{W}}} \in \mathbb{R}^{N\times N}$ the adjacency matrix of this graph [Grady and Polimeni, 2010]. When two pixels are adjacent, the corresponding nodes are connected by an edge and associated with a positive similarity weight $\overline{\mathcal{W}}_{n,n'} > 0$, otherwise $\overline{\mathcal{W}}_{n,n'}$ is set to zero. In accordance with the prior, the graph regularizer promotes similarity between the estimated nonlinearities at adjacent

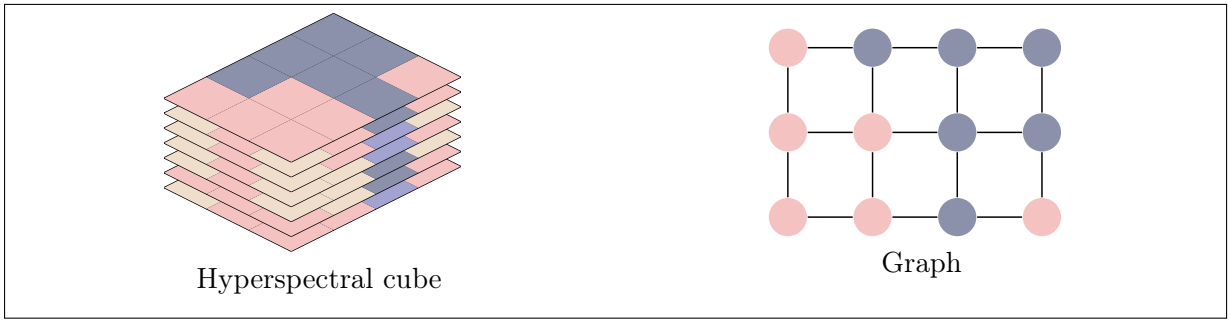


Figure 6.1: Mapping the hyperspectral image to a regular $4\mathcal{N}$ graph $\mathcal{G}$.

pixels in the image, hence connected nodes in the graph. It is defined as:

$$\|\boldsymbol{g}\|^2_{\overline{\mathcal{H}}_{\boldsymbol{k}}} = \sum_{n=1}^{N} \|g_n\|^2_{\mathcal{H}_k} \overline{\mathcal{W}}_{n,n} + \frac{1}{2} \sum_{n=1}^{N} \sum_{n'=1}^{N} \|g_n - g_{n'}\|^2_{\mathcal{H}_k} \overline{\mathcal{W}}_{n,n'}. \tag{6.8}$$

Note that, (6.8) penalizes the norms of the individual functions in addition to the differences between each pair of functions, hence forcing them to be similar. Moreover, the strength of the similarity between each pair of functions is determined by the corresponding weight. More precisely, a high value of $\overline{\mathcal{W}}_{n,n'}$ promotes a strong similarity between g_n and $g_{n'}$, and conversely, a low value of $\overline{\mathcal{W}}_{n,n'}$ promotes a weak similarity between the two functions. Using (6.7) and (6.8), some calculations show that $\overline{\boldsymbol{\mathcal{E}}}^{\dagger}$ is related to $\overline{\boldsymbol{\mathcal{W}}}$ as follows:

$$\begin{cases} \overline{\mathcal{E}}^{\dagger}_{n,n'} & = -\overline{\mathcal{W}}_{n,n'}, \text{ if } n \neq n', \\ \overline{\mathcal{E}}^{\dagger}_{n,n} & = \sum_{n'=1}^{N} \overline{\mathcal{W}}_{n,n'}, \text{ otherwise.} \end{cases} \tag{6.9}$$

Finally, note that when $\overline{\boldsymbol{\mathcal{E}}} = \boldsymbol{I}_N$, the norm of $\boldsymbol{g}$ reduces to the sum of the individual norms of its components g_n. This corresponds to processing all the functions independently without exploiting any regularization between them as in the Khype model [Chen et al., 2013].

6.4 Estimation algorithm

In order to estimate the abundances and the nonlinear function, we propose to consider the following optimization problem:

$$\begin{array}{ll} \text{minimize}_{\{\boldsymbol{e}_{\lambda_\ell}\}_{\ell=1}^{L}, \boldsymbol{g} \in \mathcal{H}_k, \boldsymbol{A}} & \frac{1}{2} \sum_{\ell=1}^{L} \|\boldsymbol{e}_{\lambda_\ell}\|^2 + \frac{\lambda}{2} \|\boldsymbol{g}\|^2_{\overline{\mathcal{H}}_{\boldsymbol{k}}} + \frac{\mu}{2} \|\boldsymbol{A}\|^2_{\mathrm{F}} \\ \text{subject to} & \boldsymbol{e}_{\lambda_\ell} = \boldsymbol{s}_{\lambda_\ell} - \boldsymbol{A}^{\top} \boldsymbol{r}_{\lambda_\ell} - \boldsymbol{g}(\boldsymbol{r}_{\lambda_\ell}) \\ & a_{i,n} \succeq 0 \quad \forall\, i = 1, \cdots M\,, n = 1, \cdots N, \\ & \sum_{i=1}^{M} a_{i,n} = 1 \quad \forall\, n = 1, \cdots, N, \end{array} \tag{6.10}$$

$$G(\widehat{\boldsymbol{v}}, \widehat{\boldsymbol{\Lambda}}, \boldsymbol{u}) =$$

$$-\tfrac{1}{2}\begin{pmatrix} \widehat{\boldsymbol{v}} \\ \widehat{\boldsymbol{\Lambda}} \\ \boldsymbol{u} \end{pmatrix}^{\top} \left(\begin{array}{c|c|c} \boldsymbol{K}_{\widehat{\boldsymbol{v}}} & \frac{1}{\mu}\boldsymbol{R}\otimes\boldsymbol{I}_N & \frac{1}{\mu}(\boldsymbol{R}\mathbf{1}_M)\otimes\boldsymbol{I}_N \\ \hline \frac{1}{\mu}\boldsymbol{R}^{\top}\otimes\boldsymbol{I}_N & \frac{1}{\mu}\boldsymbol{I}_{MN} & \frac{1}{\mu}\mathbf{1}_M\otimes\boldsymbol{I}_N \\ \hline \frac{1}{\mu}(\boldsymbol{R}\mathbf{1}_M)^{\top}\otimes\boldsymbol{I}_N & \frac{1}{\mu}\mathbf{1}_M^{\top}\otimes\boldsymbol{I}_N & \frac{M}{\mu}\boldsymbol{I}_N \end{array}\right) \begin{pmatrix} \widehat{\boldsymbol{v}} \\ \widehat{\boldsymbol{\Lambda}} \\ \boldsymbol{u} \end{pmatrix}^{\top} + \begin{pmatrix} \widehat{\boldsymbol{v}} \\ \widehat{\boldsymbol{\Lambda}} \\ \boldsymbol{u} \end{pmatrix}^{\top} \begin{pmatrix} \widehat{\boldsymbol{s}} \\ \mathbf{0}_{MN} \\ \mathbf{1}_N \end{pmatrix}, \tag{6.13}$$

where

$$\boldsymbol{K}_{\widehat{\boldsymbol{v}}} = (\boldsymbol{I}_{LN} + \frac{1}{\lambda}\overline{\boldsymbol{K}} + \frac{1}{\mu}(\boldsymbol{R}\boldsymbol{R}^{\top})\otimes\boldsymbol{I}_N). \tag{6.14}$$

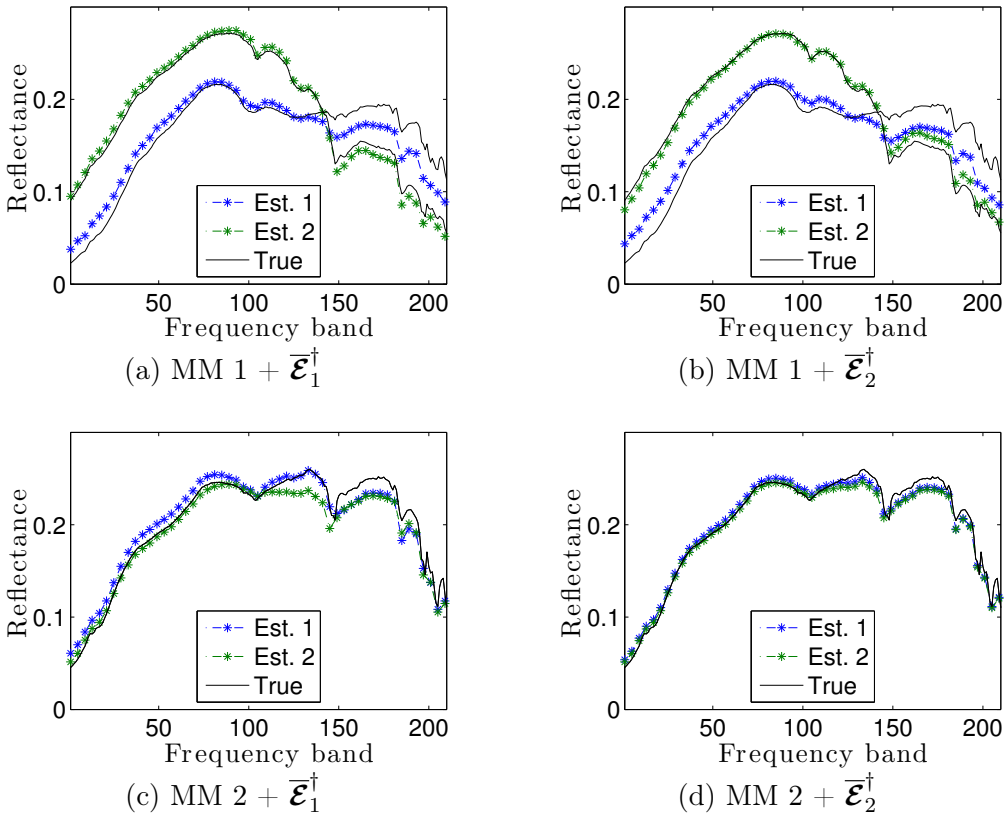
where $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N]$, and λ, μ are tuning parameters. The first term in the objective function (6.10) measures the square error between the observations and the estimated model. The second term in the objective function (6.10) is the ℓ_2-norm of $\boldsymbol{g}$ in $\overline{\mathcal{H}}_{\boldsymbol{k}}$. This term incorporates the norms of its individual outputs in addition to their weighted differences (6.8). As a result, it constrains the regularity of the estimated functions and their pairwise differences depending on the kernel design. The third term in (6.10) is the Frobenius norm of $\boldsymbol{A}$ which constrains the norm of the estimated abundances. The relevance of having simultaneously two strictly convex regularizers is that it ensures the strict convexity of the objective function. The Lagrangian associated with problem (6.10) is:

$$\begin{aligned} \mathcal{L}(\boldsymbol{E}, \boldsymbol{g}, \boldsymbol{A}, \boldsymbol{V}, \boldsymbol{\Lambda}, \boldsymbol{u}) = \; & \tfrac{1}{2}\textstyle\sum_{\ell=1}^{L}\|\boldsymbol{e}_{\lambda_\ell}\|^2 + \tfrac{\lambda}{2}\|\boldsymbol{g}\|^2_{\overline{\mathcal{H}}_{\boldsymbol{k}}} + \tfrac{\mu}{2}\|\boldsymbol{A}\|^2_{\mathrm{F}} - \boldsymbol{u}^{\top}(\boldsymbol{A}^{\top}\mathbf{1}_M - \mathbf{1}_N) \\ & + \textstyle\sum_{\ell=1}^{L}\boldsymbol{v}_{\lambda_\ell}^{\top}(\boldsymbol{s}_{\lambda_\ell} - \boldsymbol{A}^{\top}\boldsymbol{r}_{\lambda_\ell} - \boldsymbol{g}(\boldsymbol{r}_{\lambda_\ell}) - \boldsymbol{e}_{\lambda_\ell}) - \operatorname{trace}(\boldsymbol{\Lambda}^{\top}\boldsymbol{A}) \end{aligned} \tag{6.11}$$

where $\boldsymbol{V} = [\boldsymbol{v}_{\lambda_1}, \ldots, \boldsymbol{v}_{\lambda_L}]^{\top}$, $\boldsymbol{\Lambda}$, and $\boldsymbol{u}$ are the Lagrange multipliers associated with the constraints in (6.10). Given that problem (6.10) is strictly convex, its solution can be found by solving the Lagrange dual problem [Boyd and Vandenberghe, 2008]. Setting the derivatives of the Lagrangian w.r.t. the primal variables to zero yields:

$$\begin{cases} \boldsymbol{E} & = \boldsymbol{V} \\ \boldsymbol{g}(\cdot) & = \sum_{\ell=1}^{L}\boldsymbol{k}(\cdot, \boldsymbol{r}_{\lambda_\ell})\frac{\boldsymbol{v}_{\lambda_\ell}}{\lambda} \\ \boldsymbol{A} & = \frac{1}{\mu}(\boldsymbol{R}^{\top}\boldsymbol{V} + \boldsymbol{\Lambda} + \mathbf{1}_M\boldsymbol{u}^{\top}) \end{cases} \tag{6.12}$$

Replacing the optimal variables in (6.11) by their expressions in (6.12), gives the Lagrangian dual function. Some calculations show that the Lagrange dual function can be written as a quadratic form (see equation (6.13)). The vectors $\widehat{\boldsymbol{v}}$, $\widehat{\boldsymbol{\Lambda}}$ in (6.13) are shorthand notations for $\operatorname{vec}(\boldsymbol{V}^{\top})$ and $\operatorname{vec}(\boldsymbol{\Lambda}^{\top})$, where $\operatorname{vec}(\cdot)$ is an operator that stacks the columns of its input matrix on top of each other. The Lagrange dual problem consists of maximizing the Lagrange dual function (6.13), with the additional constraint $\boldsymbol{\Lambda} \geq 0$ where the inequality is applied element



(a) MM 1 + $\overline{\boldsymbol{\mathcal{E}}}_1^\dagger$ (b) MM 1 + $\overline{\boldsymbol{\mathcal{E}}}_2^\dagger$

(c) MM 2 + $\overline{\boldsymbol{\mathcal{E}}}_1^\dagger$ (d) MM 2 + $\overline{\boldsymbol{\mathcal{E}}}_2^\dagger$

Figure 6.2: True and estimated nonlinear parts in $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ obtained using the various settings for the bilinear coefficients and for $\overline{\boldsymbol{\mathcal{E}}}$.

wise. In other words, the dual problem reduces to solving a positively constrained quadratic problem. In the experiments, problem (6.13) is solved using a quadratic solver. When $\overline{\boldsymbol{\mathcal{E}}} = \boldsymbol{I}_N$, i.e. no proximity between the functions is assumed, problem (6.13) is separable and reduces to N quadratic subproblems.

6.5 Experiments on synthetic data

6.5.1 Illustrative example

The proposed approach is first illustrated using an illustrative example. Eight endmembers were randomly selected from the ENVI software library. The endmembers spectra consist of $L = 210$ spectral bands uniformly sampled in the range from 395 to 2560 nm. Two nonlinearly mixed spectra, namely, $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$, were generated such that:

$$\begin{cases} \boldsymbol{s}_1 = & \boldsymbol{R}\boldsymbol{a}_1 + u\left(\alpha_{1,1}\,\boldsymbol{R}\boldsymbol{a}_1 \odot \boldsymbol{R}\boldsymbol{a}_1 + \alpha_{1,2}\,\boldsymbol{R}\boldsymbol{a}_2 \odot \boldsymbol{R}\boldsymbol{a}_2\right) + \boldsymbol{e}_1, \\ \boldsymbol{s}_2 = & \boldsymbol{R}\boldsymbol{a}_2 + u\left(\alpha_{2,1}\,\boldsymbol{R}\boldsymbol{a}_1 \odot \boldsymbol{R}\boldsymbol{a}_1 + \alpha_{2,2}\,\boldsymbol{R}\boldsymbol{a}_2 \odot \boldsymbol{R}\boldsymbol{a}_2\right) + \boldsymbol{e}_2, \end{cases} \tag{6.15}$$

where "$\odot$" is the element wise product between two vectors, u is an attenuation coefficient set to 0.5 in the experiments, and $\alpha_{i,j} \in [0\ 1]$ is the contribution of the bilinear term depending on $\boldsymbol{a}_j$ in $\boldsymbol{s}_i$. Note that the second term on the right hand side of the first equation in (6.15) corresponds to the nonlinear contribution $[g_1(\boldsymbol{r}_{\lambda_1}) \dots g_1(\boldsymbol{r}_{\lambda_L})]^\top$, and similarly for the second equation. Two cases are considered:

- MM 1: $\alpha_{1,1} = \alpha_{2,2} = 1 \quad \text{and} \quad \alpha_{1,2} = \alpha_{2,1} = 0$
- MM 2: $\alpha_{1,1} = \alpha_{2,2} = 0.5 \ \text{and} \ \alpha_{1,2} = \alpha_{2,1} = 0.5$

MM1 corresponds to the well-known polynomial post nonlinear mixing model (PPNM) [Altmann et al., 2012]. MM2 corresponds to a bilinear model where the bilinear contributions simultaneously depend on both abundance vectors. In particular, setting all bilinear coefficients to 0.5 yields the same nonlinear contribution for $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$. Finally, the two following cases were considered for the matrix $\overline{\boldsymbol{\mathcal{E}}}^\dagger$:

- $\overline{\boldsymbol{\mathcal{E}}}_1^\dagger$: $\overline{\mathcal{W}}_{1,1} = \overline{\mathcal{W}}_{2,2} = 1$, $\overline{\mathcal{W}}_{1,2} = \overline{\mathcal{W}}_{2,1} = 0$,
- $\overline{\boldsymbol{\mathcal{E}}}_2^\dagger$: $\overline{\mathcal{W}}_{1,1} = \overline{\mathcal{W}}_{2,2} = 1$, $\overline{\mathcal{W}}_{1,2} = \overline{\mathcal{W}}_{2,1} = 10$.

The first case ($\overline{\boldsymbol{\mathcal{E}}}^\dagger = \overline{\boldsymbol{\mathcal{E}}}_1^\dagger$) does not promote any a priori similarity. The resulting norm of $\boldsymbol{g}$ given by (6.8) reduces in this case to the sum of the norms of the individual functions. The second case ($\overline{\boldsymbol{\mathcal{E}}}^\dagger = \overline{\boldsymbol{\mathcal{E}}}_2^\dagger$) promotes similarity between g_1 and g_2. In addition to the sum of norms of the individual functions g_1 and g_2, the norm of $\boldsymbol{g}$ incorporates the difference between g_1 and g_2 weighted by $\overline{\mathcal{W}}_{2,1}$. As for the scalar kernel, a second order polynomial kernel is used:

$$k(\boldsymbol{r}_{\lambda_\ell}, \boldsymbol{r}_{\lambda_{\ell'}}) = (\boldsymbol{r}_{\lambda_\ell}^\top \boldsymbol{r}_{\lambda_{\ell'}})^2. \tag{6.16}$$

The feature map of the kernel as defined by (6.16) incorporates the pairwise products between the endmembers, which motivates its use with the bilinear model. Gaussian noise was added in order to reach the desired signal to noise ratio (SNR). For each case, 100 Monte Carlo runs were performed. The performance of the proposed approach is evaluated using the root mean square error (RMSE) between the true and the estimated abundances ($\text{RMSE}_{\boldsymbol{A}} = \sqrt{\frac{1}{MN}\|\boldsymbol{A} - \widehat{\boldsymbol{A}}\|_{\text{F}}^2}$). The parameters λ and μ were tested among the values $[10^{-3}, 5 \times 10^{-3}, 10^{-2}, 10^{-1}, 1, 10]$. The endmembers are assumed to be known in the experiments. Table 6.2 reports the RMSEs of the estimated abundances and the nonlinear parts obtained in each scenario with several values of the SNR and M. Table 6.3 reports the optimal tuning parameters for each case. As expected, penalizing the discrepancy between g_1 and g_2, that is, using $\overline{\boldsymbol{\mathcal{E}}}_2^\dagger$, gives better results

Table 6.2: RMSEs ($\times 10^{-2}$) for the abundances and the nonlinear part (left and right term in brackets respectively) obtained with the illustrative example.

		$\overline{\boldsymbol{\varepsilon}}_1^{\dagger}$		
		$SNR = 40$	$SNR = 30$	$SNR = 20$
M=3	MM 1	$(1.28, 0.78)$	$(2.07, 1.05)$	$(4.55, 1.97)$
	MM 2	$(2.16, 0.78)$	$(2.54, 1.09)$	$(5.33, 2.17)$
M=5	MM 1	$(3.07, 1.46)$	$(2.60, 1.74)$	$(5.30, 3.34)$
	MM 2	$(2.32, 0.91)$	$(2.84, 1.25)$	$(4.72, 1.93)$
M=8	MM 1	$(2.12, 1.51)$	$(2.73, 1.81)$	$(5.59, 3.34)$
	MM 2	$(2.09, 0.80)$	$(3.21, 1.56)$	$(6.82, 3.43)$
		$\overline{\boldsymbol{\varepsilon}}_2^{\dagger}$		
		$SNR = 40$	$SNR = 30$	$SNR = 20$
M=3	MM 1	$(2.36, 0.85)$	$(4.24, 1.97)$	$(5.42, 2.40)$
	MM 2	$(1.12, 0.63)$	$(1.68, 0.81)$	$(4.27, 1.65)$
M=5	MM 1	$(3.07, 1.46)$	$(2.60, 1.74)$	$(5.30, 3.34)$
	MM 2	$(1.53, 0.67)$	$(2.10, 1.00)$	$(4.48, 1.73)$
M=8	MM 1	$(4.74, 2.33)$	$(3.67, 2.11)$	$(5.97, 3.10)$
	MM 2	$(1.37, 0.77)$	$(2.86, 1.21)$	$(5.99, 2.85)$

when the nonlinear parts are equal. On the other hand, using $\overline{\boldsymbol{\varepsilon}}_2^{\dagger}$ for different functions g_1 and g_2 deteriorates the results. More importantly, improving the estimation of the nonlinear part also yields an improved estimation of the abundances. This shows the importance of the estimation of $\boldsymbol{g}$, and the use of a correct prior. Figure 6.2 shows the true and estimated functions g_1 and g_2 for each case. Figures 6.2 (a) and (b) correspond to the case where g_1 and g_2 are different, they are represented by two solid lines. Whereas Figures 6.2 (c) and (d) correspond to the case where g_1 and g_2 are equal, hence they are both represented by one solid line. The estimated nonlinear contributions are represented using blue and green dashed lines respectively. Figures 6.2 (b) and (d) show how penalizing the difference between g_1 and g_2 yields closer estimations compared to Figures 6.2 (a) and (c).

6.5.2 Spatial data set

The proposed approach was tested on a synthetic image known as the spatial image [Chen et al., 2014]. The abundance maps for this image are the same as those used for the image IM2 in [Chen et al., 2014]. The image has 100×100 pixels, and is composed of 8 endmembers. As in the previous experiment, a bilinear mixing model was used where the bilinear coefficients depend on neighboring abundances. The attenuation parameter was set to $u = 0.5$. The endmembers spectra used in the previous experiment were used in this experiment. A white Gaussian noise

Table 6.3: Optimal parameters (λ, μ) used with each algorithm in the ilustrative example.

		$\overline{\boldsymbol{\varepsilon}}_1^\dagger$		
		$SNR = 40$	$SNR = 30$	$SNR = 20$
M=3	MM 1	$(1, 0.1)$	$(1, 0.1)$	$(1, 0.1)$
	MM 2	$(0.1, 0.01)$	$(1, 0.1)$	$(1, 0.1)$
M=5	MM 1	$(1, 0.1)$	$(1, 0.1)$	$(10, 1)$
	MM 2	$(0.1, 0.01)$	$(1, 0.1)$	$(10, 1)$
M=8	MM 1	$(1, 0.1)$	$(1, 0.1)$	$(1, 0.1)$
	MM 2	$(1, 0.01)$	$(1, 0.01)$	$(1, 0.1)$
		$\overline{\boldsymbol{\varepsilon}}_2^\dagger$		
		$SNR = 40$	$SNR = 30$	$SNR = 20$
M=3	MM 1	$(0.005, 0.005)$	$(1, 0.1)$	$(1, 0.1)$
	MM 2	$(1, 0.1)$	$(1, 0.1)$	$(1, 0.1)$
M=5	MM 1	$(0.01, 0.001)$	$(1, 0.1)$	$(10, 1)$
	MM 2	$(0.1, 0.01)$	$(1, 0.1)$	$(10, 1)$
M=8	MM 1	$(0.1, 0.01)$	$(1, 0.1)$	$(1, 0.1)$
	MM 2	$(0.1, 0.01)$	$(1, 0.01)$	$(1, 0.1)$

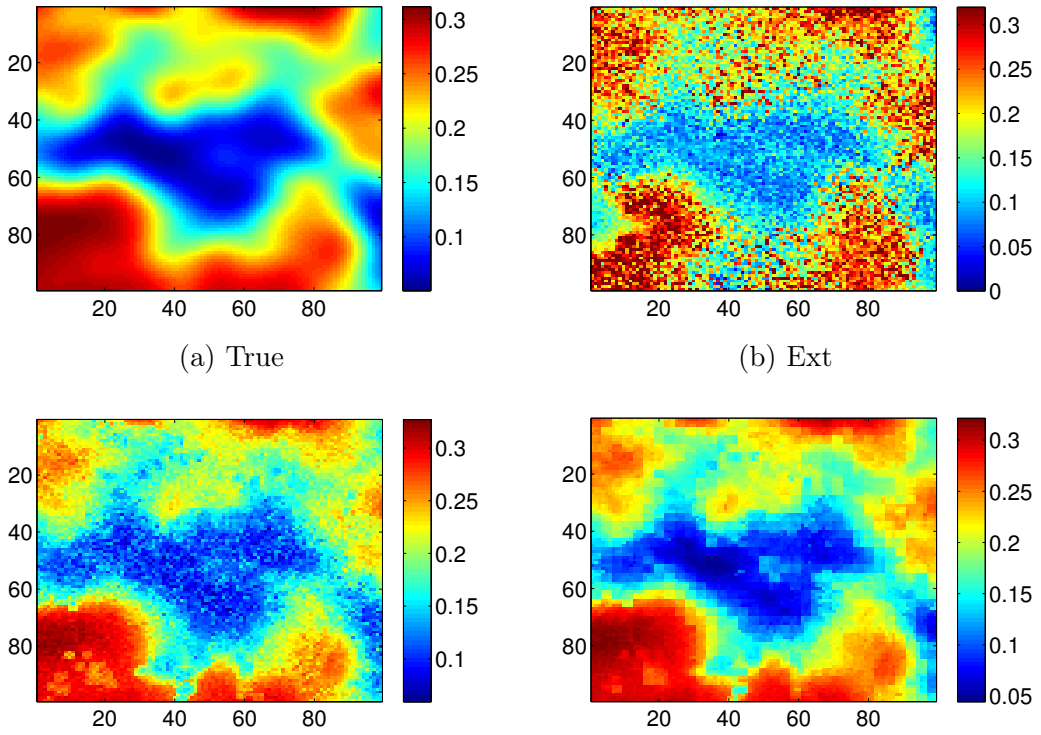
was added to the observations in order to get an SNR of 30 dB. The image was unmixed using three methods. The first method is the extended endmember matrix method (ExtM) [Raksuntorn and Du, 2010]. It consists of extending the endmember matrix artificially with cross-spectra of pure materials. The second method is Khype [Chen et al., 2013]. It was obtained by simply setting $\overline{\boldsymbol{\varepsilon}}$ to the identity matrix in our algorithm. The third method is the proposed approach used with $\overline{\boldsymbol{\varepsilon}} \neq \boldsymbol{I}$, i.e., with prior information regarding the similarity between the nonlinearities. For the latter method, the image was decomposed into 3×3 disjoint patches in order to reduce the computational complexity. In each patch, nonlinear parts at adjacent pixels are assumed to be similar. The similarity weights were tested among the values [10, 50, 100]. After preliminary tests, they were set to 50 in all experiments. Table 6.4 reports the RMSEs of the abundances and the nonlinear contributions for the three methods. The best scores in terms of the RMSEs are obtained with the proposed approach. Figure 6.3 shows the true and estimated nonlinear contributions at band #100 obtained with each method. Figure 6.3 (d) shows that incorporating the spatial prior resulted in visually smoother variations.

6.6 Conclusion

This chapter proposed a new framework for incorporating spatial regularization in nonlinear unmixing. The proposed model promotes smooth spatial variations of the nonlinear components in the mixing model. The optimization problem reduces to a QP problem. In particular, the

Table 6.4: Unmixing performance and optimal tuning parameters obtained with the spatial data set.

	Ext	Khype	Prop.
RMSE$(A, A^\star)$	0.0507	0.0380	0.0276
RMSE$(F, F^\star)$	0.0507	0.0213	0.0138
(λ, μ)	–	$(1, 0.01)$	$(1, 0.01)$



(a) True (b) Ext

(c) Khype (d) Proposed

Figure 6.3: True and estimated nonlinear contributions at band #100 obtained with the spatial data set using the extended endmember method (Ext), Khype , and the proposed approach.

proposed model exploits the separable kernel design and the resulting graph regularization which allows to incorporate the prior about the similarities between the nonlinear part at different pixels using a graph. Finally, the performance of the proposed approach was validated on synthetic data.

Chapter 7

Concluding remarks

This manuscript addressed several problems in hyperspectral unmixing. First, an unsupervised unmixing approach based on collaborative sparse regularization was proposed where the library of endmembers candidates was built from the observations themselves. This approach was then extended in order to take into account the presence of noise among the endmembers candidates. Second, within the unsupervised unmixing framework, two graph-based regularizations were used in order to incorporate prior local and nonlocal contextual information. Next, within a supervised nonlinear unmixing framework, a new nonlinear mixing model based on vector-valued functions in RKHS was proposed. The aforementioned model allowed us to consider different nonlinear functions at different bands and to account for neighboring nonlinear contributions. Finally, in the last chapter of this thesis, the vector-valued kernel framework was used in order to promote spatial smoothness of the nonlinear part in a kernel-based nonlinear mixing model. In this concluding chapter, the methods proposed in each chapter are briefly recalled and some future research directions are discussed.

7.1 Summary of objectives and contributions

This thesis focused on the problem of mixed spectra through the development of new unmixing techniques in various interrelated contexts throughout its four core chapters, namely chapters 3 to 7. In particular, the objectives and contributions are described in what follows:

1. The first objective was to perform linear unmixing given that the number of endmembers and their spectral signatures are not known, i.e. in a blind and unsupervised setting. It was assumed that the endmembers are present among the observations, and they were sought as the purest pixels in the scene. To perform this task, the observations were

modelled as a linear combination of a few spectra from the observations themselves. The unmixing problem was formulated using sparse regularization, namely the Group Lasso regularization. The main advantage of the proposed model was that the endmembers used to characterize the observations were obtained in the same atmospheric conditions as the observations and underwent the same atmospheric corrections as well. Nevertheless, since the observations and equivalently the estimated endmembers were corrupted with noise two models were considered depending on how the presence of noise in the endmembers was handled.

2. The second objective was to improve the unmixing accuracy by taking into account the spatial information and the spectral correlation inherently present in hyperspectral images. To perform this task, we proposed to construct a weighted graph, such that each pixel corresponds to a node in the graph and the weights represent to the degree of similarity between the corresponding pixels. The graph structure was able to easily combine the spatial and spectral prior information through its edge set and weights. The prior information was then incorporated in the optimization problem through the definition of an appropriate regularization term depending on the graph structure. We considered two different similarity priors, hence two regularizations, that we incorporated in an unsupervised linear mixing model. The first prior assumed that the abundances have smooth variations at similar pixels, whereas the second prior assumed that the reconstructed spectra have piece-wise smooth variations at similar pixels. Furthermore, we proposed two strategies for building a meaningful graph from the observations.

3. The third objective was to improve the unmixing accuracy in the case of a nonlinear mixing model. Roughly speaking, we performed this task by spectrally (chapter 5) and spatially (chapter 6) regularizing the nonlinear component in a kernel-based nonlinear mixing model. To perform these two tasks, we proposed a new framework based on vector-valued RKHS. In particular, the proposed models exploit a special type of kernels known as the separable kernels which allows to incorporate the prior about the similarities between the pixels (chapter 5) and the bands (chapter 6) using a graph representation.

4. Finally, the last objective was to solve the various proposed models using a relatively unified and flexible framework. In fact, the majority of the proposed optimization problems are convex, constrained, and usually involve a non-smooth regularization term in the objective function. In order to solve these optimization problems, the proposed algorithms

were based on the ADMM. This choice allowed to effectively handle the various convex, constrained, and non-smooth optimization problems by decomposing the original problem into a sequence of smaller ones. In particular, we exploited the ADMM framework to avoid solving sylvester equations in chapter 4, and to handle an optimization problem which includes an unknown function in a RKHS.

7.2 Future research directions

This thesis provided several contributions in unsupervised and nonlinear unmixing of hyperspectral data. As for future research, it would be interesting to investigate the following ideas which are aimed at improving and exploring extensions of the proposed models.

Short term perspectives

- ***Graph lasso and group lasso with overlap***: In chapter 3, the proposed unsupervised unmixing model is based on the group lasso regularization which enables the selection of the endmembers present in the scene. The group lasso regularization imposes that the same group of endmembers explains the observed spectra. Interestingly, the work in [Jacob et al., 2009] proposes two new collaborative sparse penalties, namely, the graph lasso and the group lasso with overlap. If used within the unmixing framework, these regularizations would allow to incorporate prior information about the relations between the endmembers. For example, in the case of the group lasso with overlap, it would be possible to select the endmembers such that they are a union of groups of endmembers defined a priori with possible overlaps. Whereas in the case of the graph lasso, it would be possible to select the endmembers such that the selected endmembers tend to be connected to each other in a given graph where each candidate endmember is represented by a node. Both penalizations are interesting to examine in the unmixing framework where one can incorporate additional information about the coexistence of endmembers, i.e. if a specific endmember is present in the scene then the other endmembers in the corresponding group are present too. Nevertheless, the application of these two regularizations in the case of libraries of known spectral signatures or image-based candidate endmembers needs further investigation with regards to how the group partitions or graph topology should be chosen.

- ***Distributed optimization over graphs***: In chapter 4, we have introduced two graph based regularizations in the unmixing framework for incorporating contextual information.

The resulting optimization problems do not scale well due to the fact that the abundances are no longer estimated in a pixel-by-pixel manner. Instead, all the abundances in the image have to be simultaneously estimated. To reduce the computational complexity of the resulting algorithms, we performed clustering and imposed a sparse graph structure. It would be interesting to investigate the potential computational gain, memory wise and time wise, if distributed optimization techniques over graphs are used. For example, extending the work in [Nassif et al., 2016] to the graph regularized unmixing framework would allow for each node (pixel) to estimate its own abundance and share its current estimate with its neighbors in order to solve the overall problem iteratively.

Long term perspectives

- ***Learning the graph structure for the graph-based regularizations***: Another future research direction includes the influence of the graph topology configuration. The efficiency of the two regularizations proposed in chapter 4 relies on the construction of a meaningful graph representation. We have tested two simple strategies mainly based on the pairwise correlation between the available observations and their spatial configuration. Nevertheless, the observations can be corrupted with noise which hinders the accuracy of the correlation metric derived from the observations when the SNR increases. Furthermore, the assumption that the similarities between the observations reflect the similarities between the estimated abundances may not be sufficient or accurate in practice. An alternative for defining the graph is to learn it simultaneously while unmixing. For example, this problem has been studied in [Dong et al., 2015] where the authors jointly estimate the unknown variables and the corresponding Laplacian matrix. It would be interesting to investigate the extension of the aforementioned work to the framework adopted in chapter 4. Furthermore, learning the Laplacian matrix hence the topology of the graph would provide new insights on the interactions between the pixels in the image and would allow to analyze the relative importance of local and nonlocal connections.

- ***Learning the graph structure for the separable kernel***: Similarly to the previous point, the graph structure required for the design of the separable kernel in chapters 5 and 6 can be simultaneously learned while estimating the unknown variables. The work in [Sindhwani et al., 2012] attempts to perform this task in a scalable way. In the unmixing framework, this would imply a simultaneous estimation of the graph structure, the nonlinear function, and the abundances. Furthermore, research in this direction can

benefit the proposed approaches in chapters 5 and 6 computationally since the authors of [Sindhwani et al., 2012] propose a scalable method based. Finally, in parallel to what was mentioned in the previous point, learning the graph structure allows us to analyze the nonlinear interactions between the bands (chapter 5) or the pixels (chapter 6).

Appendix A

Proof of positively constrained MiSTO

Proof. Since problem (3.16) is convex, we simply have to check the validity of the solution in the two cases $\|(\boldsymbol{v})_+\|_2 > \alpha$ and $\|(\boldsymbol{v})_+\|_2 < \alpha$. Let $f_0(\boldsymbol{z}) = \frac{1}{2}\|\boldsymbol{z} - \boldsymbol{v}\|_2^2 + \alpha\|\boldsymbol{z}\|_2$. For $\|(\boldsymbol{v})_+\|_2 > \alpha$, the gradient of f_0 is given by

$$\nabla f_0(\boldsymbol{z}^*) = \left(1 + \frac{\alpha}{\|\boldsymbol{z}^*\|_2}\right)\boldsymbol{z}^* - \boldsymbol{v}. \tag{A.1}$$

Replacing by the appropriate expression from (3.16) yields

$$\nabla f_0(\boldsymbol{z}^*) = (\boldsymbol{v})_+ - \boldsymbol{v} \geq 0 \tag{A.2}$$

$$\boldsymbol{z}_i^* \cdot \nabla f_0(\boldsymbol{z}^*)_i \propto ((\boldsymbol{v})_+)_i \cdot ((\boldsymbol{v})_+ - \boldsymbol{v})_i = 0. \tag{A.3}$$

These two conditions correspond the optimality conditions, which means that $\boldsymbol{z} \succeq \boldsymbol{0}$ is a solution for the constrained problem. For more details, refer to section 4.2.3 in [Boyd and Vandenberghe, 2008].

For the second case, note that for every $\boldsymbol{z} \succeq \boldsymbol{0}$, we have

$$\sum_i \boldsymbol{z}_i \boldsymbol{v}_i \leq \sum_i \boldsymbol{z}_i (\boldsymbol{v}_i)_+ \leq \|\boldsymbol{z}\|_2 \cdot \|(\boldsymbol{v})_+\|_2. \tag{A.4}$$

It follows that

$$\begin{aligned} f_0(\boldsymbol{z}) - f_0(\boldsymbol{0}) &= \frac{1}{2}\sum_i \boldsymbol{z}_i^2 - \sum_i \boldsymbol{z}_i \boldsymbol{v}_i + \alpha\|\boldsymbol{z}\|_2 \\ &\geq \frac{1}{2}\|\boldsymbol{z}\|_2^2 - \|\boldsymbol{z}\|_2 \cdot \|(\boldsymbol{v})_+\|_2 + \alpha\|\boldsymbol{z}\|_2 \\ &\geq \frac{1}{2}\|\boldsymbol{z}\|_2^2 + \|\boldsymbol{z}\|_2(\alpha - \|(\boldsymbol{v})_+\|_2). \end{aligned} \tag{A.5}$$

This proves that for $\|(\boldsymbol{v})_+\|_2 \leq \alpha$, the minimum is reached for $\boldsymbol{z}^* = \boldsymbol{0}$. □

Appendix B

Probability distribution of observations (NGLUP)

We return to model (3.5) that was considered in chapter 3, section 3.3, which expresses the mixing model as:

$$\boldsymbol{S} = \boldsymbol{S}_\omega \boldsymbol{X} + \boldsymbol{E}(\boldsymbol{I}_N - \boldsymbol{I}_\omega \boldsymbol{X}). \tag{B.1}$$

The aim of this section is to determine the probability distribution of the observations $\{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_N\}$ based on model (B.1) needed in order to derive the corresponding negative log likelihood. To lighten the notations, we consider that $\omega = \{1, \ldots, N\}$. In this case equation (B.1) becomes:

$$\boldsymbol{S} = \boldsymbol{S}\boldsymbol{X} + \boldsymbol{E}(\boldsymbol{I}_N - \boldsymbol{X}). \tag{B.2}$$

For each column in $\boldsymbol{S}$ we have

$$\boldsymbol{s}_k = \boldsymbol{S}\boldsymbol{x}_k + \boldsymbol{b}_k, \tag{B.3}$$

where $\boldsymbol{x}_k$ denotes the k-th column of $\boldsymbol{X}$ and

$$\boldsymbol{b}_k = \boldsymbol{e}_k - \boldsymbol{E}\boldsymbol{x}_k. \tag{B.4}$$

We rewrite $\boldsymbol{b}_k$ in the following equivalent form:

$$\boldsymbol{b}_k = \sum_{\ell=1}^{N} (i_{\ell k} - x_{\ell k})\boldsymbol{e}_\ell, \tag{B.5}$$

where $i_{\ell k}$ is the (ℓk)-th entry of $\boldsymbol{I}$, i.e. $i_{\ell k} = 0$ if $k \neq \ell$ and $i_{\ell k} = 1$ if $k = \ell$. Recall that the noise is assumed to be Gaussian independent and identically distributed, with zero mean and a possibly unknown variance σ^2, that is,

$$e_{ki} \sim \mathcal{N}(0, \sigma^2), \tag{B.6}$$

and equivalently, for each column $\boldsymbol{e}_k$, we have

$$\boldsymbol{e}_k \sim \mathcal{N}(\boldsymbol{0}_L, \sigma^2 \boldsymbol{I}_L). \tag{B.7}$$

Based on (B.5) and (B.7), $\boldsymbol{b}_k$ follows a Gaussian distribution with the following mean and variance, and the covariance between $\boldsymbol{b}_k$ and $\boldsymbol{b}_m$ is derived as follows:

$$\begin{aligned}
\mathrm{E}[\boldsymbol{b}_k] &= \mathrm{E}\left[\textstyle\sum_{\ell=1}^{N}(i_{\ell k} - x_{\ell k})\boldsymbol{e}_\ell\right] \\
&= \textstyle\sum_{\ell=1}^{N}(i_{\ell k} - x_{\ell k})\mathrm{E}\left[\boldsymbol{e}_\ell\right] \\
&= \boldsymbol{0}_L,
\end{aligned}$$

$$\begin{aligned}
\mathrm{Var}(\boldsymbol{b}_k) &= \mathrm{Var}\left(\textstyle\sum_{\ell=1}^{N}(i_{\ell k} - x_{\ell k})\boldsymbol{e}_\ell\right) \\
&= \mathrm{Var}\left(\textstyle\sum_{\ell=1}^{N}(i_{\ell k} - x_{\ell k})\boldsymbol{e}_\ell\right) \\
&= \textstyle\sum_{\ell=1}^{N}(i_{\ell k} - x_{\ell k})^2\mathrm{Var}\left(\boldsymbol{e}_\ell\right) \\
&\quad +2\textstyle\sum_{1\le\ell<m\le N}(i_{\ell k} - x_{\ell k})(i_{k,m} - x_{m,k})\mathrm{Cov}\left(\boldsymbol{e}_\ell, \boldsymbol{e}_m\right) \\
&= \textstyle\sum_{\ell=1}^{N}(i_{\ell k} - x_{\ell k})^2\sigma^2\boldsymbol{I}_L \\
&= \sigma^2\|\boldsymbol{i}_k - \boldsymbol{x}_k\|^2\boldsymbol{I}_L,
\end{aligned} \tag{B.8}$$

$$\begin{aligned}
\mathrm{Cov}(\boldsymbol{b}_k, \boldsymbol{b}_m) &= \mathrm{Cov}\left(\textstyle\sum_{\ell=1}^{N}(i_{\ell k} - x_{\ell k})\boldsymbol{e}_\ell, \sum_{\ell'=1}^{N}(i_{\ell' m} - x_{\ell' m})\boldsymbol{e}_\ell\right) \\
&= \textstyle\sum_{\ell=1}^{N}\sum_{\ell'=1}^{N}(i_{\ell k} - x_{\ell k})(i_{\ell' m} - x_{\ell' m})\mathrm{Cov}\left(\boldsymbol{e}_\ell, \boldsymbol{e}'_\ell\right) \\
&= \textstyle\sum_{\ell=1}^{N}(i_{\ell k} - x_{\ell k})(i_{\ell,m} - x_{\ell,m})\mathrm{Cov}\left(\boldsymbol{e}_\ell, \boldsymbol{e}_\ell\right) \\
&= (\boldsymbol{i}_k - \boldsymbol{x}_k)^\top(\boldsymbol{i}_m - \boldsymbol{x}_m)\mathrm{Var}\left(\boldsymbol{e}_\ell\right) \\
&= \sigma^2(\boldsymbol{i}_k - \boldsymbol{x}_k)^\top(\boldsymbol{i}_m - \boldsymbol{x}_m)\boldsymbol{I}_L.
\end{aligned}$$

Let $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_N]$, it can be concluded from (B.8) that we have the following mean and variance for the vector $\mathrm{vec}(\boldsymbol{B}) = [\boldsymbol{b}_1^\top, \ldots, \boldsymbol{b}_N^\top]^\top$:

$$\mathrm{E}([\boldsymbol{b}_1^\top, \ldots, \boldsymbol{b}_N^\top]^\top) = \boldsymbol{0}_{LN}, \tag{B.9}$$

$$\mathrm{Var}([\boldsymbol{b}_1^\top, \ldots, \boldsymbol{b}_N^\top]^\top) = \sigma^2\left[(\boldsymbol{I}_N - \boldsymbol{X})^\top(\boldsymbol{I}_N - \boldsymbol{X})\right] \otimes \boldsymbol{I}_L, \tag{B.10}$$

where we have used that:

$$\mathrm{Var}([\boldsymbol{b}_1^\top, \ldots, \boldsymbol{b}_N^\top]^\top) = \begin{bmatrix}
\mathrm{Var}(\boldsymbol{b}_1) & \mathrm{Cov}(\boldsymbol{b}_1, \boldsymbol{b}_2) & \cdots & \mathrm{Cov}(\boldsymbol{b}_1, \boldsymbol{b}_N) \\
\mathrm{Cov}(\boldsymbol{b}_2, \boldsymbol{b}_1) & \mathrm{Var}(\boldsymbol{b}_2) & \cdots & \mathrm{Cov}(\boldsymbol{b}_2, \boldsymbol{b}_N) \\
\vdots & \vdots & \ddots & \vdots \\
\mathrm{Cov}(\boldsymbol{b}_N, \boldsymbol{b}_1) & \mathrm{Cov}(\boldsymbol{b}_N, \boldsymbol{b}_2) & \cdots & \mathrm{Var}(\boldsymbol{b}_N)
\end{bmatrix}. \tag{B.11}$$

Given the relationship between $\boldsymbol{b}_k$ and $\boldsymbol{s}_k$ in (B.3), the vector $[\boldsymbol{s}_1^\top, \ldots, \boldsymbol{s}_N^\top]^\top$ follows the Gaussian distribution with the following mean and variance:

$$\mathrm{E}([\boldsymbol{s}_1^\top, \ldots, \boldsymbol{s}_N^\top]^\top) = [(\boldsymbol{S}\boldsymbol{x}_1)^\top, \ldots, (\boldsymbol{S}\boldsymbol{x}_N)^\top]^\top, \tag{B.12}$$

$$\mathrm{Var}([\boldsymbol{s}_1^\top, \ldots, \boldsymbol{s}_N^\top]^\top) = \sigma^2 \left[(\boldsymbol{I}_N - \boldsymbol{X})^\top (\boldsymbol{I}_N - \boldsymbol{X}) \right] \otimes \boldsymbol{I}_L. \tag{B.13}$$

Appendix C

Graph clustering

The purpose of the graph clustering is to reduce the computational complexity of the (C.1)

$$\boldsymbol{Y}^{k+1} = (\boldsymbol{V}^k + \rho \boldsymbol{X}^{k+1})(2\lambda \mathcal{L}_{\mathcal{G}} + \rho \boldsymbol{I}_N)^{-1}, \tag{C.1}$$

using spectral clustering. Assume that the pixels can be grouped into k disjoint clusters or subgraphs of size $N_1, N_2, \cdots, N_k$ denoted by $\mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_k$ such that $N = N_1 + N_2 + \cdots + N_k$. Without any loss of generality, assume that the pixels' spectra in $\mathcal{G}$ are ordered such that the first N_1 are in $\mathcal{G}_1$, the next N_2 are in $\mathcal{G}_2$, and so on. This yields the following new block diagonal affinity matrix

$$\widehat{\boldsymbol{\mathcal{W}}} = \begin{bmatrix} \boldsymbol{\mathcal{W}}^{(11)} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\mathcal{W}}^{(22)} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \boldsymbol{\mathcal{W}}^{(kk)} \end{bmatrix} \tag{C.2}$$

where parenthesized superscripts are used to index a sub block in the corresponding matrix. The diagonal blocks in $\widehat{\boldsymbol{\mathcal{W}}}$, namely, $\boldsymbol{\mathcal{W}}^{(ii)}$ are the $N_i \times N_i$ matrices of inter-cluster affinities. Compared to the affinity matrix $\boldsymbol{\mathcal{W}}$ of the original graph $\mathcal{G}$, $\widehat{\boldsymbol{\mathcal{W}}}$ is equal to $\boldsymbol{\mathcal{W}}$ with the difference that the intra-cluster affinities are set to zero. This corresponds to zeroing W_{ij} in $\boldsymbol{\mathcal{W}}$ if $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$ belong to different clusters, thus moving the clusters "infinitly" far apart [Ng et al., 2002]. The advantage of approximating $\boldsymbol{\mathcal{W}}$ by $\widehat{\boldsymbol{\mathcal{W}}}$ is two fold. First, storing $\widehat{\boldsymbol{\mathcal{W}}}$ requires less memory compared to $\boldsymbol{W}$. It is no longer necessary to store the intra-cluster affinities. Second, using $\widehat{\boldsymbol{\mathcal{W}}}$ instead of $\boldsymbol{\mathcal{W}}$ in (C.1) allows to reduce the $\boldsymbol{Y}$ minimization step to k separate problems, one for

each cluster, which yields

$$\text{minimize}_{\boldsymbol{Y}^{(i)}} \quad \tfrac{\rho}{2} \|\boldsymbol{X}^{(i)} - \boldsymbol{Y}^{(i)}\|_{\mathrm{F}}^2 - \text{trace}({\boldsymbol{V}^{(i)}}^{\top}(\boldsymbol{Y}^{(i)})) + \lambda \text{trace}(\boldsymbol{Y}^{(i)} \widehat{\mathcal{L}}_{\mathcal{G}}^{(ii)} {\boldsymbol{Y}^{(i)}}^{\top}) \tag{C.3}$$

where $\widehat{\mathcal{L}}_{\mathcal{G}}^{(ii)}$ corresponds to the i-th diagonal block of the Laplacian matrix derived from $\widehat{\boldsymbol{\mathcal{W}}}$. The solution for each sub-problem,

$${\boldsymbol{Y}^{(i)}}^{k+1} = ({\boldsymbol{V}^{(i)}}^{k} + \rho {\boldsymbol{X}^{(i)}}^{k+1})(2\lambda \widehat{\mathcal{L}}_{\mathcal{G}}^{(ii)} + \rho \boldsymbol{I})^{-1}, \tag{C.4}$$

requires solving a smaller linear system with only N_i unknown variables, N_i being smaller than N. The storage and computational complexity of the overall $\boldsymbol{Y}$ minimization step depend on the size of the largest sub-graph

$$N_{\max} = \max(N_i)_{i=1}^{k}. \tag{C.5}$$

To summarize, partitioning the graph into k clusters allows to approximate the affinity matrix $\boldsymbol{\mathcal{W}}$ by a sparse and block-diagonal affinity matrix $\widehat{\boldsymbol{\mathcal{W}}}$. Furthermore, it allows to solve k smaller linear systems instead of solving one linear system of size N resulting in memory and speed gains.

In order to estimate the optimal $\widehat{\boldsymbol{\mathcal{W}}}$, the graph $\mathcal{G}$ needs to be partitioned into k disjoint clusters. To perform this task, we use normalized cuts as described in [Fowlkes et al., 2004]. The authors of [Fowlkes et al., 2004] propose an approximation technique for evaluating the first n_e eigenvectors of the Laplacian matrix. The eigenvectors are then used to find the optimal partition of the graph into two clusters. This approach to clustering is known as spectral clustering, some of the most used spectral clustering algorithms are described in [Weiss, 1999]. We use the strategy proposed in [Ng et al., 2002] to simultaneously find the k clusters rather than two clusters. In fact the leading eigenvectors of the normalized affinity matrix induce an embedding of the pixels in a low dimensional subspace. The approach in [Ng et al., 2002] consists of using a simple clustering method such as k-means in order to cluster the rows of the leading eigenvectors. The resulting clustering algorithm is described in algorithm 4.

Algorithm 4 Spectral clustering into k clusters

Compute $\boldsymbol{\mathcal{W}}^{(11)}$ and $\boldsymbol{\mathcal{W}}^{(12)}$
$\boldsymbol{d}_1 = \text{sum}([\boldsymbol{\mathcal{W}}^{(11)}, \boldsymbol{\mathcal{W}}^{(12)}], 2)$
$\boldsymbol{d}_2 = \text{sum}(\boldsymbol{\mathcal{W}}^{(21)}, 2) + \text{sum}(\boldsymbol{\mathcal{W}}^{(21)} \times \text{pinv}(\boldsymbol{\mathcal{W}}^{(11)}) \times \boldsymbol{\mathcal{W}}^{(12)}, 2)$
$\boldsymbol{d}1 = \mathbf{1}./\sqrt{\boldsymbol{d}_1}$
$\boldsymbol{d}2 = \mathbf{1}./\sqrt{\boldsymbol{d}_2}$
$\boldsymbol{\mathcal{W}}^{(11)} = \boldsymbol{\mathcal{W}}^{(11)} \times \boldsymbol{d}_1 \times \boldsymbol{d}_1^\top$
$\boldsymbol{\mathcal{W}}^{(12)} = \boldsymbol{\mathcal{W}}^{(12)} \times \boldsymbol{d}_2 \times \boldsymbol{d}_2^\top$
$[\boldsymbol{U}, \boldsymbol{L}] = \text{eigs}(\boldsymbol{\mathcal{W}}^{(11)}, k)$
$\boldsymbol{V} = [\boldsymbol{U}; \boldsymbol{\mathcal{W}}^{(21)} \times \boldsymbol{U} \times \boldsymbol{L}^{-1}]$
Normalize each row in $\boldsymbol{V}$ to have unity norm
Treating each row in $\boldsymbol{V}$ as a point, use k-means to cluster points into k groups

Appendix D

List of publications

The research work during the thesis resulted in publications in several international journals, and peer-reviewed international and national conferences. More precisely, it lead to 1 published journal paper, 1 journal paper under revision, 5 international peer-reviewed conference papers, and 1 national peer-reviewed conference paper. The thesis work was partly supported by the Agence Nationale pour la Recherche, France, through the Hypanema project (ANR-12-BS03-003). Furthermore, it was supported by the regional council of Provence-Alpes-Côte d'Azur through the regional doctoral grant program, and the socio-economic Partnership of THALES Alenia Space. Hereafter is a list of the publications developed during the thesis.

- **Peer-reviewed international journal articles (1+1)**

  - R. Ammanouil, A. Ferrari, C. Richard, and S. Mathieu, "Nonlinear unmixing of hyperspectral data with vector-valued kernel functions," Article submitted to *IEEE Trans. Image Process.* in May 2016, currently under revision.

  - R. Ammanouil, A. Ferrari, C. Richard and D. Mary, "Blind and fully constrained unmixing of hyperspectral images," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5510–5518, Dec. 2014.

- **Peer-reviewed international conference articles (5)**

  - R. Ammanouil, A. Ferrari, C. Richard and J.-Y. Tourneret, "Spatial regularization for nonlinear unmixing of hyperspectral data with vector-valued kernel functions," in *Proc. 19-th IEEE Workshop on Statistical Signal Processing (SSP)*, Palma de Mallorca, Spain, June, 2016.

- R. Ammanouil, A. Ferrari and C. Richard, "Unsupervised neighbor dependent nonlinear unmixing," in *Proc. 41-st IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai, China, Mar. 2016, pp. 1437 – 1441.
- R. Ammanouil, A. Ferrari and C. Richard, "A graph Laplacian regularization for hyperspectral data unmixing," in *Proc. 40-th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brisbane, Australia, Apr. 2015, pp. 1637–1641.
- R. Ammanouil, A. Ferrari and C. Richard, "Hyperspectral data unmixing with graph-based regularization," in *Proc. 7-th IEEE GRSS Workshop Hyperspectral Image Signal Process.: Evolution in Remote Sens. (WHISPERS)*, Tokyo, Japan, June 2015, pp. 1–4.
- R. Ammanouil, A. Ferrari, C. Richard and D. Mary, "GLUP: Yet another algorithm for blind unmixing of hyperspectral data," in *Proc. 6-th IEEE GRSS Workshop Hyperspectral Image Signal Process.: Evolution in Remote Sens. (WHISPERS)*, Lausanne, Switzerland, June 2014, pp. 1–4.

- **Peer-reviewed national conference article (1)**
  - R. Ammanouil, A. Ferrari and C. Richard, "Modèle non-linéaire et semi - paramétrique pour le demélange non-supervisé d'images hyperspectrales," *Acte du 25 ème colloque GRETSI*, Lyon, France, sept. 2015.

Appendix E

Résumé en Français

Résumé

Le démélange spectral est l'un des problèmes centraux pour l'exploitation des images hyperspectrales. En raison de la faible résolution spatiale des imageurs hyperspectraux en télédetection, la surface représentée par un pixel peut contenir plusieurs matériaux. Dans ce contexte, le démélange consiste à estimer les spectres purs (les endmembers) ainsi que leurs fractions (les abondances) pour chaque pixel de l'image. Le but de cette thèse est de proposer de nouveaux algorithmes de démélange qui visent à améliorer l'estimation des spectres purs et des abondances. En particulier, les algorithmes de démélange proposés s'inscrivent dans le cadre du démélange non-supervisé et non-linéaire. Dans un premier temps, on propose un algorithme de démelange non-supervisé dans lequel une régularisation favorisant la parcimonie des groupes est utilisée pour identifier les spectres purs parmi les observations. Une extension de ce premier algorithme permet de prendre en compte la présence du bruit parmi les observations choisies comme étant les plus pures. Dans un second temps, les connaissances a priori des ressemblances entre les spectres à l'échelle locale et non-locale ainsi que leurs positions dans l'image sont exploitées pour construire un graphe adapté à l'image. Ce graphe est ensuite incorporé dans le problème de démélange non-supervisé par le biais d'une régularisation basée sur le Laplacian du graphe. Enfin, deux algorithmes de démélange non-linéaires sont proposés dans le cas supervisé. Les modèles de mélanges non-linéaires correspondants incorporent des fonctions à valeurs vectorielles appartenant à un espace de Hilbert à noyaux reproduisants. L'intérêt de ces fonctions par rapport aux fonctions à valeurs scalaires est qu'elles permettent d'incorporer un a priori sur la ressemblance entre les différentes fonctions. En particulier, un a priori spectral, dans un premier temps, et un a priori spatial, dans un second temps, sont incorporés pour améliorer la caractérisation du mélange non-linéaire. La validation expérimentale des modèles et des algorithmes proposés sur des données synthétiques et réelles montre une amélioration des performances par rapport aux méthodes de l'état de l'art. Cette amélioration se traduit par une meilleure erreur de reconstruction des données.

Mots clés: Données hyperspectrales, démélange non-supervisé, démélange non-linéaire, algorithme des directions altérnées (ADMM), régularisation de type group lasso, régularisation avec le Laplacian, espace de Hilbert à noyau reproduisant (RKHS).

E.1 Introduction

L'analyse des images hyperspectrales, également connue comme la spectroscopie, est l'une des technologies les plus importantes et les plus connues dans le domaine de la télédétection. L'avantage principale des images hyperspectrales est qu'elles fournissent la représentation d'une scène suivant un très grand nombre de bandes spectrales avec une grande résolution. Cela permet l'identification des matériaux présents dans la scènes et l'extraction de plusieurs paramètres physiques.

Le démélange spectral est l'un des outils les plus importants pour analyser les données hyperspectrales. Au lieu d'attribuer un matériau ou une classe spécifique à chaque pixel, comme dans le cas de la classification, le démélange spectral suppose qu'un pixel peut avoir une composition mixte, et par la suite un spectre mixte. Dans ce contexte, on distingue systématiquement entre les spectres purs et les spectres mixtes. Un spectre pur, aussi connu comme un endmember, est un spectre associé à un matériau ou à une classe spécifique. Idéalement, il s'agit d'une signature spectrale qui représente la matière correspondante. En revanche, un spectre mixte est estimé dans le cas où plusieurs faisceaux lumineux ayant interagis avec plus qu'un matériau atteignent le capteur: un spectre mixte est un mélange des spectres purs des matériaux présents dans le pixel correspondant. Dans le cas non-supervisé, le démélange consiste à estimer les spectres purs (les endmembers) et leurs fractions (les abondances) pour tous les pixels de l'image. Dans le cas non-supervisé, les spectres purs sont connus a priori, et le démélange consiste à estimer les abondances. Deux approches existent pour décrire le mécanisme de mélange spectral: les modèles linéaires et les modèles non-linéaires. D'après le modèle de mélange linéaire, un spectre mixte est exprimé comme une somme pondérée des spectres purs. Cette modélisation est relativement précise dans le cas où les matériaux sont repartis sur des zones distinctes dans le pixel, les faisceaux lumineux arrivant au capteur subissent une seule réflexion, et par la suite chaque faisceau lumineux a interagi avec un seul matériau. Si cela n'est pas le cas, c.a.d. si l'une des conditions précédentes n'est pas vérifiée, le mélange spectral devient non-linéaire. Plus précisément, cela correspond au cas où les matériaux sont mélangés de façon intime (intimate mixture), ou lorsque les faisceaux lumineux subissent des réflexions multiples avant d'atteindre le capteur. Dans ces deux cas, les photons interagissent avec plus qu'un matériau et le mélange est non-linéaire. L'estimation des endmembers et des abondances est évidemment plus complexe dans le cas non-linéaire.

E.2 Résumé de la thèse

Cette thèse s'inscrit dans le cadre du démélange des données hyperspectrales, en particulier le démélange non-supervisé et non-linéaires. Les trois contributions principales de cette thèse sont: le développement d'une nouvelle approche pour le démélange non-supervisé, la modélisation de l'information spatiale par un graphe pour le démélange (linéaire et non-linéaire), et l'introduction d'un nouveau modèle de démélange non-linéaire. La thèse se compose de sept chapitres. Le premier chapitre commence avec un aperçu des travaux présentés dans la thèse et présente son organisation. Chapitre 2 explique le contexte du démélange hyperspectral. Les quatre chapitres suivants, c.a.d. les chapitres 3 à 6 , présentent les contribution de la thèse, en particulier ils expliquent les techniques de démélange proposées. L'organisation de ces chapitres est présentée dans ce qui suit.

Dans le troisième chapitre, nous introduisons une approche originale pour le démélange non-supervisé utilisant une régularisation favorisant la parcimonie des groupes. Le démélange est effectué sans un dictionnaire de spectres purs, et suppose que le nombre de matériaux présents dans la scène et que leurs signatures spectrales sont inconnues. Deux modèles sont développés pour réaliser cette tâche, selon la façon dont le bruit interagit avec les données hyperspectrales. Le premier modèle conduit à un problème d'optimisation convexe, et le problème d'optimization correspondant est résolu avec la méthode des directions alternées (ADMM). Alors que le second modèle prend en compte la présence du bruit dans les observations, et le problème d'optimisation correspondant est résolu avec un algorithme de moindres carrés pondérés.

Dans le quatrième chapitre, nous proposons d'intégrer une régularisation spatiale dans la formulation du démélange en utilisant des graphes. Plus précisément, ce chapitre présente deux régularisations basées sur les graphes qui sont utilisées dans un problème de démélange linéaire et non-supervisé. Les régularisations proposées reposent sur la construction d'une correspondance entre l'image hyperspectrale et un graphe. Dans le premier cas, une régularisation quadratique basée sur le Laplacian du graphe est proposée pour promouvoir des variations douces dans les cartes d'abondances et une collaboration entre les zones homogènes de l'image. Dans le second cas, une régularisation de type variation totale basée sur un graphe est utilisée pour promouvoir l'estimation de spectres constants par morceau par rapport à la structure du graphe. Les problèmes d'optimisation résultants sont convexes et résolus en utilisant la méthode des directions alternées (ADMM). La performance des algorithmes proposés est démontrée en les comparant à d'autres algorithmes de l'état de l'art avec des données hyperspectrales réelles et

simulées.

Le cinquième chapitre propose un nouveau modèle de mélange non-linéaire qui permet de incorporer un a priori sur la variation spectrale de la contribution non-linéaire. Pour cela il modélise la partie non-linéaire en utilisant une fonction non-linéaire à valeur vectorielle appartenant à un espace de Hilbert à noyau reproduisant. En comparaison avec les modèles non-linéaires existants, le modèle proposé permet de prendre en compte des fonctions non-linéaires différentes par bande. La motivation principale de ce modèle est que les contributions non-linéaires peuvent être dominantes dans certaines parties du spectre alors qu'elles sont moins prononcées dans d'autres parties. En plus la fonction non-linéaire agit sur les spectres voisins. Cela est motivé par le fait que les contributions non-linéaires peuvent être issues des pixels voisins, par exemple à cause de l'effet d'adjacence. La fonction non-linéaire proposée est associée à un noyau à valeur matricielle permettant de modéliser conjointement plusieurs types de non-linéarités et d'introduire des informations a priori concernant les dépendances entre les différentes bandes spectrales. En outre, le fait que la fonction agisse sur les spectres voisins permet d'incorporer des effets d'adjacence. Un type particulier de noyaux dont la conception repose sur la construction d'un graphe où chaque bande est représentée par un noeud dans le graphe est étudié. Le problème d'optimisation est strictement convexe et l'algorithme itératif correspondant est basé sur la méthode des directions alternées (ADMM). Enfin, des expériences menées sur des données synthétiques et réelles montrent l'efficacité de l'approche proposée.

Dans le sixième chapitre, nous proposons d'intégrer une régularisation spatiale dans un modèle de mélange non-linéaire aussi basé sur les noyaux. De façon similaire au chapitre précédent, le modèle proposé utilise les fonctions à valeur vectorielle appartenant à un espace de Hilbert à noyau reproduisant (RKHS). Cette fonction est utilisée pour introduire une régularisation sur la partie non-linéaire promouvant des variations lisses spatialement de la partie non-linéaire. La régularisation spatiale et les contributions non-linéaires sont modélisées conjointement par la fonction à valeur vectorielle. La conception du noyau repose sur la construction d'un graphe correspondant à l'image hyperspectrale où chaque noeud représente un pixel. Le problème d'optimisation correspondant est un problème quadratique strictement convexe. Des simulations sur des données synthétiques illustrent l'efficacité de l'approche proposée.

Le septième et dernier chapitre conclut la thèse et donne un résumé des avantages et les principales contributions des méthodes développées dans les chapitres précédents. Il fournit également des perspectives pour des futures travaux.

E.3 Conclusion

Cette thèse a abordé plusieurs problèmes liés au démélange des données hyperspectrales. Dans cette section, nous résumons les contributions principales de ce travail et nous proposons quelques perspectives pour des futurs travaux.

1. Le premier objectif de la thèse était de résoudre le problème de démélange dans un cadre non-supervisé, i.e. lorsque le nombre de endmembers et leurs spectres sont inconnus. Pour résoudre ce problème, nous avons supposé que les spectres purs sont présents parmi les observations, et que chaque spectre mesuré est une combinaison linéaire de ces spectres purs. Le problème d'optimisation correspondant est alors basé sur une régularisation parcimonieuse qui favorise la parcimonie des groupes. Un des intérêts de ce modèle est que les corps purs ainsi estimés ont subis les mêmes corrections atmosphériques que les observations. Toutefois, en choisissant les spectres purs parmi les observations, il ne tient pas compte du bruit additif dont celles-ci sont entachées. Afin d'améliorer la performance de la méthode proposée, un deuxième modèle qui prend en compte la présence du bruit parmi les observations est considéré.

2. Le deuxième objectif de la thèse était d'améliorer la performance du démélange dans le cas non-supervisé en prenant en compte l'information spatiale présente dans les images hyperspectrales. Pour cela, nous avons proposé de construire un graphe pondéré où chaque noeud correspond à un pixel et où les poids attribués aux liens entre les noeuds représentent le degré de similarité entre les pixels correspondants. La structure du graphe a permis de représenter l'a priori spatial et spectral à travers l'ensemble des liens et des poids associés. Dans un premier temps, cet a priori a été incorporé dans le problème d'optimisation grace à une régularisation définie par le Laplacian du graphe. Dans un second temps, il a été incorporé dans le problème d'optimisation grace à une régularisation définie par la matrice d'adjacence du graphe. Dans le premier problème d'optimisation nous avons considéré que les abondances ont des variations lisses sur le graphe. Dans le deuxième problème d'optimisation, nous avons considéré que les spectres reconstruits ont des variations lisses sur le graphe.

3. Le troisième objectif était d'améliorer la performance dans le cas d'un mélange non-linéaire. Pour cela, nous avons utilisé des fonctions non-linéaires à valeur vectorielle dans un espace de Hilbert à noyau reproduisant. Nous proposons pour cela deux modèles de démélange

dans lesquels la contribution non-linéaire est régularisée spectralement (dans le chapitre 5), et spatialement (dans le chapitre 6). En particulier, la fonction noyau utilisée dans les deux modèles proposés permet d'incorporer un a priori sur les ressemblances entre les différentes sorties de la fonction grace à un graphe.

La majorité des problèmes d'optimization rencontrés dans ce travail sont convexes, contraints, et contiennent un terme non differentiable dans la fonction objective. Pour résoudre ces problèmes, nous avons utilisé la méthode des directions alternées. Cette méthode permet de décomposer le problème initial en une séquence de sous-problèmes relativement plus simples. Par exemple, cette approche nous a permis de contourner la résolution d'une équation de Sylvester dans le chapitre 4, et de résoudre des problèmes d'optimisation avec des fonctions dans un espace de Hilbert à noyau reproduisant.

Concernant les futurs travaux de recherche, il serait intéressant d'examiner les pistes suivantes qui permettront d'améliorer et d'étendre les méthodes déjà proposées. A court terme, nous envisageons d'utiliser les régularisations de type Graph lasso et group lasso avec chevauchement [Jacob et al., 2009] pour améliorer l'extraction des corps purs parmi les observations. Ces deux types de régularisations permettront d'incorporer un a priori sur les relations entre les endmembers. En plus, il pourrait être intéressant de rendre l'estimation des abondances proposée dans le chapitre 4 distribuée comme par exemple dans [Nassif et al., 2016]. A plus long terme, une des directions de recherche serait d'apprendre la structure du graphe simultanément avec le problème d'estimation. Cela est en contrast avec l'approche adoptée dans le chapitre 4 et 5 où le graphe est fixé a priori.

Bibliography

J. Adams, M. Smith, and P. Johnson. Spectral mixture modeling: a new analysis of rock and soil types at the viking lander 1 site. *Journal of Geophysical Research: Solid Earth*, 91(B8): 8098–8112, 1986.

Y. Altmann, A. Halimi, N. Dobigeon, and J.-Y. Tourneret. Supervised nonlinear spectral unmixing using a postnonlinear mixing model for hyperspectral imagery. *IEEE Transactions on Image Processing*, 21(6):3017–3025, 2012.

Y. Altmann, N. Dobigeon, S. McLaughlin, and J.-Y. Tourneret. Nonlinear spectral unmixing of hyperspectral images using gaussian processes. *IEEE Transactions on Signal Processing*, 61 (10):242–245, 2013.

Y. Altmann, N. Dobigeon, and J.-Y. Tourneret. Unsupervised post-nonlinear unmixing of hyperspectral images using a hamiltonian monte carlo algorithm. *IEEE Transactions on Image Processing*, 23(6):2663–2675, 2014.

M. Alvarez, L. Rosasco, and N. Lawrence. Kernels for vector-valued functions: A review. *Foundations and Trends in Machine Learning*, 4(3):195–266, 2012.

R. Ammanouil, A. Ferrari, C. Richard, and D. Mary. Blind and fully constrained unmixing of hyperspectral images. *IEEE Transactions on Image Processing*, 23(12):5510 – 5518, October 2014.

R. Ammanouil, A. Ferrari, and C. Richard. A graph laplacian regularization for hyperspectral data unmixing. In *Proc. of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1637–1641. IEEE, 2015a.

R. Ammanouil, A. Ferrari, and C. Richard. Hyperspectral data unmixing with graph-based regularization. In *Proc. IEEE GRSS Workshop Hyperspectral Image Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pages 1–4. IEEE, 2015b.

R. Ammanouil, A. Ferrari, C. Richard, and S. Mathieu. Nonlinear unmixing of hyperspectral data with vector-valued kernel functions. *submitted to IEEE Transactions on Image Processing*, May 2016a.

R. Ammanouil, A. Ferrari, C. Richard, and J.-Y. Tourneret. Spatial regularization for nonlinear unmixing of hyperspectral data with vector-valued kernel functions. In *Proc. IEEE Workshop on Statistical Signal Processing*, pages 1–4. IEEE, 2016b.

A. Argyriou, M. Herbster, and M. Pontil. Combining graph laplacians for semi–supervised learning. *NIPS*, 2005.

N. Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, pages 337–404, 1950.

G. Asner and D. Lobell. A biogeophysical approach for automated swir unmixing of soils and vegetation. *Remote sensing of environment*, 74(1):99–112, 2000.

L. Baldassarre. *Multi-output learning with spectral filters*. PhD thesis, Ph. D. dissertation, Dept. of Physics-University of Genoa, 2010.

R. H. Bartels and G. W. Stewart. Solution of the matrix equation $AX + XB = C$. *Communications of the ACM*, 15(9):820–826, 1972.

M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *The Journal of Machine Learning Research*, 7: 2399–2434, 2006.

J. Bioucas-Dias and M. Figueiredo. Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing. In *Proc. of the 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pages 1–4. IEEE, 2010.

J. 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*, 5(2): 354–379, 2012.

J. M. Bioucas-Dias. A variable splitting augmented lagrangian approach to linear spectral unmixing. In *First Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, pages 1–4. IEEE, 2009.

C. Boutsidis, M. Mahoney, and P. Drineas. An improved approximation algorithm for the column subset selection problem. In *Proc. of the ACM-SIAM Symposium on Discrete Algorithms*, pages 968–977. Society for Industrial and Applied Mathematics, 2009.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2008.

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*, 3(1):1–122, 2011.

J. Broadwater and A. Banerjee. A comparison of kernel functions for intimate mixture models. In *First Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, pages 1–4. IEEE, 2009.

J. Broadwater and A. Banerjee. A generalized kernel for areal and intimate mixtures. In *2010 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, pages 1–4. IEEE, 2010.

A. Bruckstein, D. Donoho, and M. Elad. From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM Review*, 51:34–81, 2009.

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*, 6(3):1070–1078, 2013.

G. Camps-Valls, T. B. Marsheva, and D. Zhou. Semi-supervised graph-based hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 45(10):3044–3054, 2007.

E. Candes, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207, 2006.

R. J. Carroll and D. B. H. Cline. An asymptotic theory for weighted least-squares with weights estimated by replication. *Biometrika*, 75(1):35–43, March 1988.

A. Castrodad, Z. Xing, J. Greer, E. Bosch, L. Carin, and G. Sapiro. Learning discriminative sparse representations for modeling, source separation, and mapping of hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 49(11):4263–4281, 2011.

C. Chang. *Hyperspectral data exploitation: theory and applications*. John Wiley and Sons, 2007.

C. Chang and Q. Du. Estimation of number of spectrally distinct signal sources in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 42(3):608–619, March 2004.

J. Chen, C. Richard, and P. Honeine. Nonlinear unmixing of hyperspectral images based on multi-kernel learning. In *IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing*, 2012.

J. Chen, C. Richard, and P. Honeine. Nonlinear unmixing of hyperspectral data based on a linear-mixture/nonlinear-fluctuation model. *IEEE Transactions on Signal Processing*, 61(2): 480–492, 2013.

J. Chen, C. Richard, and P. Honeine. Nonlinear estimation of material abundances in hyperspectral images with l1-norm spatial regularization. *IEEE Transactions on Geoscience and Remote Sensing*, 52(5):2654 – 2665, 2014.

S. Chen, D. Donoho, and M. Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43(1):129–159, 2001.

R. Clark and G. A. Swayze. Mapping minerals, amorphous materials, environmental materials, vegetation, water, ice and snow, and other materials: the usgs tricorder algorithm. In *In summaries of the fifth annual JPL airborne earth science workshop, JPL publication*, 1995.

P. L. Combettes and J. C. Pesquet. Proximal splitting methods in signal processing. In *arXiv:0912.3522*, 2009.

C. Couprie, L. Grady, L. Najman, J. C. Pesquet, and H. Talbot. Dual constrained tv-based regularization on graphs. *SIAM Journal on Imaging Sciences*, 6(3):1246–1273, 2013.

I. Daubechies, R. De Vore, M. Fornasier, and C. S. Güntürk. Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics*, 63(1): 1–38, 2010.

F. Ding and T. Chen. Gradient based iterative algorithms for solving a class of matrix equations. *IEEE Transactions on Automatic Control*, 50(8):1216–1221, 2005.

N. Dobigeon, L. Tits, B. Somers, Y. Altmann, and P. Coppin. A comparison of nonlinear mixing models for vegetated areas using simulated and real hyperspectral data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6):1869–1878, 2014a.

N. Dobigeon, J.-Y. Tourneret, C. Richard, J. C. Bermudez, S. McLaughlin, and A. O. Hero. Nonlinear unmixing of hyperspectral images: Models and algorithms. *IEEE Signal Processing Magazine*, 31(1):82–94, 2014b.

X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst. Laplacian matrix learning for smooth graph signal representation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3736–3740. IEEE, 2015.

O. Eches and M. Guillaume. A bilinear–bilinear nonnegative matrix factorization method for hyperspectral unmixing. *IEEE Geoscience and Remote Sensing Letters*, 11(4):778–782, 2014.

O. Eches, N. Dobigeon, C. Mailhes, and J.-Y. Tourneret. Bayesian estimation of linear mixtures using the normal compositional model. *IEEE Transactions on Image Processing*, 19(6):1403–1413, June 2010.

O. Eches, N. Dobigeon, and J.-Y. Tourneret. Enhancing hyperspectral image unmixing with spatial correlations. *IEEE Transactions on Geoscience and Remote Sensing*, 49(11):4239–4247, 2011.

J. Eckstein and D. P. Bertsekas. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55(1):293–318, 1992.

E. Elhamifar and R. Vidal. Sparse subspace clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2790–2797. IEEE, 2009.

E. Esser, M. Moller, S. Osher, G. Sapiro, and J. Xin. A convex model for nonnegative matrix factorization and dimensionality reduction on physical space. *IEEE Transactions on Image Processing*, 21(7):3239–3252, July 2012.

J. Esser. *Primal dual algorithms for convex models and applications to image restoration, registration and nonlocal inpainting.* PhD thesis, University of California Los Angeles, 2010.

T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. *Advances in computational mathematics*, 13(1):1–50, 2000.

T. Evgeniou, C. Micchelli, and M. Pontil. Learning multiple tasks with kernel methods. *Journal of Machine Learning Research*, pages 615–637, 2005.

W. Fan, B. Hu, J. Miller, and M. Li. Comparative study between a new nonlinear model and common linear model for analysing laboratory simulated-forest hyperspectral data. *International Journal of Remote Sensing*, 30(11):2951–2962, 2009.

C. Févotte and N. Dobigeon. Nonlinear hyperspectral unmixing with robust nonnegative matrix factorization. *IEEE Transactions on Image Processing*, 24(12):4810–4819, 2015.

C. Fowlkes, S. Belongie, F. Chung, and J. Malik. Spectral grouping using the nystrom method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):214–225, 2004.

X. Fu, W. Ma, T. Chan, J. Bioucas-Dias, and M.-D. Iordache. Greedy algorithms for pure pixels identification in hyperspectral unmixing: A multiple-measurement vector viewpoint. In *Proc. of the European Signal Processing Conference (EUSIPCO)*, pages 1–5. IEEE, 2013.

W. A. Fuller and J. N. K. Rao. Estimation for a linear regression model with unknown diagonal covariance matrix. *The Annals of Statistics*, 6(5):1149–1158, september 1978.

B. C. Gao and A. Goetz. Column atmospheric water vapor and vegetation liquid water retrievals from airborne imaging spectrometer data. *Journal of Geophysical Research: Atmospheres*, 95 (D4):3549–3564, 1990.

B. C. Gao, M. Montes, C. Davis, and A. Goetz. Atmospheric correction algorithms for hyperspectral remote sensing data of land and ocean. *Remote Sensing of Environment*, 113:S17–S24, 2009.

D. Gillis and J. Bowles. Hyperspectral image segmentation using spatial-spectral graphs. In *Proc. of the SPIE Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery*, pages 83901Q–83901Q. International Society for Optics and Photonics, 2012.

A. Goetz. Imaging spectrometry for studying earth, air, fire and water. *EARSeL Advances in Remote Sensing*, 1(1):3–15, 1991.

A. Goetz, G. Vane, J. Solomon, and B. Rock. Imaging spectrometry for earth remote sensing. *Science*, 228(4704):1147–1152, 1985.

L. J. Grady and J. R. Polimeni. *Discrete calculus*. Springer, 2010.

D. Hadjimitsis, G. Papadavid, A. Agapiou, K. Themistocleous, M. Hadjimitsis, A. Retalis, S. Michaelides, N. Chrysoulakis, L. Toulios, and C. Clayton. Atmospheric correction for satel-

lite remotely sensed data intended for agricultural applications: impact on vegetation indices. *Natural Hazards and Earth System Science*, 10(1):89–95, 2010.

A. Halimi, Y. Altmann, N. Dobigeon, and J.-Y. Tourneret. Nonlinear unmixing of hyperspectral images using a generalized bilinear model. *IEEE Transactions on Geoscience and Remote Sensing*, 49(11):4153–4162, 2011.

M. K. Hamilton, C. O. Davis, W. J. Rhea, S. H. Pilorz, and K. L. Carder. Estimating chlorophyll content and bathymetry of lake tahoe using aviris data. *Remote Sensing of Environment*, 44 (2):217–230, 1993.

B. Hapke. *Theory of reflectance and emittance spectroscopy*. Cambridge University Press, 2012.

D. C. Heinz and C. I. Chang. Fully constrained least squares linear spectral mixture analysis method for material quantification in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 39(3):529–545, March 2001.

R. Heylen and P. Scheunders. A multilinear mixing model for nonlinear spectral unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 2016.

R. Heylen, D. Burazerovic, and P. Scheunders. Fully constrained least squares spectral unmixing by simplex projection. *IEEE Transactions on Geoscience and Remote Sensing*, 49(11):4112–4122, 2011a.

R. Heylen, D. Burazerovic, and P. Scheunders. Non-linear spectral unmixing by geodesic simplex volume maximization. *IEEE Journal of Selected Topics in Signal Processing*, 5(3):534–542, 2011b.

R. Heylen, M. Parente, and P. Gader. A review of nonlinear hyperspectral unmixing methods. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6): 1844 –1868, 2014a.

R. Heylen, P. Scheunders A. Rangarajan, and P. Gader. Nonlinear unmixing by using different metrics in a linear unmixing chain. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):2655–2664, June 2014b.

P. Honeine and C. Richard. Geometric unmixing of large hyperspectral images: A barycentric coordinate approach. *IEEE Transactions on Geoscience and Remote Sensing*, 50(6):2185–2195, June 2012.

P. M. Hooper. Iterative weighted least squares estimation in heteroscedastic linear models. *Journal of the American Statistical Association*, 88(421):179–184, 1993.

M. D. Iordache, J. M. Bioucas-Dias, and A. Plaza. Sparse unmixing of hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing*, 49(6):2014–2039, 2011.

M.-D. Iordache, J. Bioucas-Dias, and A. Plaza. Total variation spatial regularization for sparse hyperspectral unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 50(11):4484–4502, 2012a.

M. D. Iordache, J. M. Bioucas-Dias, and A. Plaza. Dictionary pruning in sparse unmixing of hyperspectral data. In *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pages 1–4, June 2012b.

M.-D. Iordache, J. Bioucas-Dias, and A. Plaza. Collaborative sparse regression for hyperspectral unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 52(1):341–354, February 2013.

M.-D. Iordache, J. Bioucas-Dias, A. Plaza, and B. Somers. MUSIC-CSR: Hyperspectral unmixing via multiple signal classification and collaborative sparse regression. *IEEE Transactions on Geoscience and Remote Sensing*, 52:4364–4382, 2014a.

M.-D. Iordache, A. Okujeni, S. van der Linden, J. Bioucas-Dias, A. Plaza, and B. Somers. A multi-measurement vector approach for endmember extraction in urban environments. In *Image Information Mining Conference: The Sentinels Era*, Bucharest, Romania, 2014b.

L. Jacob, G. Obozinski, and J. P. Vert. Group lasso with overlap and graph lasso. In *Proc. of the annual international conference on machine learning*, pages 433–440. ACM, 2009.

S. Jia and Y. Qian. Spectral and spatial complexity-based hyperspectral unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 45(12):3867–3879, 2007.

N. Keshava and J. F. Mustard. Spectral unmixing. *IEEE Signal Processing Magazine*, 19(1): 44–57, 2002.

A. Kovac and A. Smith. Nonparametric regression on a graph. *Journal of Computational and Graphical Statistics*, 20(2):432–447, 2011.

A. F. Kruse. Advances in hyperspectral remote sensing for geologic mapping and exploration. In *Proceedings 9th Australasian remote sensing conference, Sydney, Australia*, 1998.

F. A. Kruse, J. W. Boardman, and J. F. Huntington. Fifteen years of hyperspectral data: northern grapevine mountains, nevada. In *Proceedings of the 8th JPL Airborne Earth Science Workshop: Jet Propulsion Laboratory Publication*, pages 99–17, 1999.

D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562, 2001.

S. Liang, H. Fang, and M. Chen. Atmospheric correction of landsat etm+ land surface imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 39(11):2490–2498, 2001.

X. Liu, W. Xia, B. Wang, and L. Zhang. An approach based on constrained nonnegative matrix factorization to unmix hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing*, 49(2):757–772, 2011.

X. Lu, H. Wu, Y. Yuan, P. Yan, and X. Li. Manifold regularized sparse NMF for hyperspectral unmixing. *IEEE Transactions on Geoscience and Remote Sensing*, 51(5):2815–2826, May 2013.

J. S. MacDonald, S. L. Ustin, and M. E. Schaepman. The contributions of Dr. Alexander FH Goetz to imaging spectrometry. *Remote Sensing of Environment*, 113:S2–S4, 2009.

I. Meganem, P. Deliot, X. Briottet, Y. Deville, and S. Hosseini. Linear–quadratic mixing model for reflectances in urban environments. *IEEE Transactions on Geoscience and Remote Sensing*, 52(1):544–558, 2014.

L. Miao and H. Qi. Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization. *IEEE Transactions on Geoscience and Remote Sensing*, 45(3):765–777, 2007.

L. Miao, H. Qi, and H. Szu. A maximum entropy approach to unsupervised mixed-pixel decomposition. *IEEE Transactions on Image Processing*, 16(4):1008–1021, April 2007.

C. Micchelli and M. Pontil. On learning vector-valued functions. *Neural Computation*, 17(1): 177–204, 2005.

B. Mohar and Y. Alavi. The laplacian spectrum of graphs. *Graph theory, combinatorics, and applications*, 2:871–898, 1991.

J. F. Mustard and C. M. Pieters. Photometric phase functions of common geologic minerals and applications to quantitative analysis of mineral mixture reflectance spectra. *Journal of Geophysical Research: Solid Earth*, 94(B10):13619–13634, 1989.

J. M. P. Nascimento and J. Bioucas-Dias. Vertex component analysis: A fast algorithm to unmix hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing*, 43(4):898–910, 2005.

J. M. P. Nascimento and J. Bioucas-Dias. Nonlinear mixture model for hyperspectral unmixing. In *Proc. of the SPIE Image and Signal Processing for Remote Sensing*, pages 74770I–74770I, 2009.

J. P. Nascimento and J. M. Bioucas-Dias. Unmixing hyperspectral intimate mixtures. In *Remote Sensing*, pages 78300C–78300C. International Society for Optics and Photonics, 2010.

R. Nassif, C. Richard, A. Ferrari, and A. Sayed. Proximal multitask learning over networks with sparsity-inducing coregularization. *IEEE Transactions on Signal Processing*, (99), 2016.

A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2:849–856, 2002.

D. L. Peterson, J. D. Aber, P. A. Matson, D. H. Card, N. Swanberg, C. Wessman, and M. Spanner. Remote sensing of forest canopy and leaf biochemical contents. *Remote Sensing of Environment*, 24(1):85–108, 1988.

G. Peyré, S. Bougleux, and L. Cohen. Non-local regularization of inverse problems. In *Computer Vision–ECCV*, pages 57–68. Springer, 2008.

C. Pieters and J. F. Mustard. Exploration of crustal/mantle material for the earth and moon using reflectance spectroscopy. *Remote Sensing of Environment*, 24(1):151–178, 1988.

W. M. Porter and H. T. Enmark. A system overview of the airborne visible/infrared imaging spectrometer (aviris). In *31st Annual Technical Symposium*, pages 22–31. International Society for Optics and Photonics, 1987.

A. T. Puig, A. Wiesel, G. Fleury, and A. O. Hero. Multidimensional shrinkage-thresholding operator and group lasso penalties. *IEEE Signal Processing Letters*, 18(6):363–366, April 2011.

N. Raksuntorn and Q. Du. Nonlinear spectral mixture analysis for hyperspectral imagery in an unknown environment. *IEEE Geoscience and Remote Sensing Letters*, 7(4):836–840, 2010.

R. Richter and D. Schlapfer. Atmospheric/topographic correction for satellite imagery. *DLR report DLR-IB*, pages 565–01, 2005.

R. Richter, M. Bachmann, W. Dorigo, and A. Muller. Influence of the adjacency effect on ground reflectance measurements. *IEEE Geoscience and Remote Sensing Letters*, 3(4):565–569, 2006.

L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

Z. Zhang S. G. Mallat. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, december 1993.

Y. Saad. *Iterative methods for sparse linear systems*. Siam, 2003.

J. Schott, K. Lee, R. Raqueno, G. Hoffmann, and G. Healey. A subpixel target detection technique based on the invariance approach. In *AVIRIS Airborne Geoscience workshop Proceedings*, 2003.

G. Shaw and H. Burke. Spectral imaging for remote sensing. *Lincoln Laboratory Journal*, 14(1): 3–28, 2003.

J. Shawe-Taylor and N. Cristianini. *Kernel methods for pattern analysis*. Cambridge university press, 2004.

J. R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain, 1994.

C. Shi and L. Wang. Incorporating spatial information in spectral unmixing: A review. *Remote Sensing of Environment*, 149:70–87, 2014.

Y. E. Shimabukuro and J. A. Smith. The least-squares mixing models to generate fraction images derived from remote sensing multispectral data. *IEEE Transactions on Geoscience and Remote sensing*, 29(1):16–20, 1991.

D. I. Shuman, S. k. Naran, P. Frossard, A. Ortega, and P. Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Transactions on Signal Processing*, 30(3):83–98, march 2013.

V. Sindhwani, M. Quang, and A. Lozano. Scalable matrix-valued kernel learning for high-dimensional nonlinear multivariate regression and granger causality. *available as arXiv preprint arXiv:1210.4792*, 2012.

R. B. Singer and T. B. McCord. Mars-large scale mixing of bright and dark surface materials and implications for analysis of spectral reflectance. In *Lunar and Planetary Science Conference Proceedings*, volume 10, pages 1835–1848, 1979.

B. Somers, K. Cools, S. Delalieux, J. Stuckens, D. Van der Zande, W. Verstraeten, and P. Coppin. Nonlinear hyperspectral mixture analysis for tree cover estimates in orchards. *Remote Sensing of Environment*, 113(6):1183–1193, 2009.

K. Staenz, A. Mueller, A. Held, and U. Heiden. Technical committees corner: International spaceborne imaging spectroscopy (isis) technical committee. *IEEE Geosciscience Remote Sensing Newsletter*, 165:38–42, 2012.

D. Strong and T. Chan. Exact solutions to total variation regularization problems. In *UCLA CAM Report*, 1996.

D. Tanre, P. Deschamps, P. Duhaut, and M. Herman. Adjacency effect produced by the atmospheric scattering in thematic mapper data. *Journal of Geophysical Research: Atmospheres*, 92(D10):12000–12006, 1987.

E. Thiebaut, F. Soulez, and L. Denis. Exploiting spatial sparsity for multiwavelength imaging in optical interferometry. *Journal of the Optical Society of America A*, 30(2):160–170, February 2013.

L. Tong, J. Zhou, X. Bai, and Y. Gao. Dual graph regularized NMF for hyperspectral unmixing. In *Proc. of the International Conference on Digital lmage Computing: Techniques and Applications (DICTA)*, Nov. 2014.

J. Tropp. Just relax: Convex programming methods for identifying sparse signals in noise. *IEEE Transactions on Information Theory*, 52(3):1030–1051, 2006.

Y. Weiss. Segmentation using eigenvectors: a unifying view. In *Proc. of the ICCV*, volume 2, pages 975–982. IEEE, 1999.

M. Wendisch and J.-L. Brenguier. *Airborne measurements for environmental research: methods and instruments*. John Wiley & Sons, 2013.

M. E. Winter. N-FINDR: an algorithm for fast autonomous spectral endmember determination in hyperspectral data. In *Proc. of the SPIE Imaging Spectrometry*, October 1999.

Z. Yang, G. Zhou, S. Xie, S. Ding, J.-M. Yang, and J. Zhang. Blind spectral unmixing based on sparse nonnegative matrix factorization. *IEEE Transactions on Image Processing*, 20(4): 1112–1125, April 2011.

N. Yokoya, J. Chanussot, and A. Iwasaki. Nonlinear unmixing of hyperspectral data using semi-nonnegative matrix factorization. *IEEE Transactions on Geoscience and Remote Sensing*, 52 (2):1430–1437, 2014.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical methodology)*, 68(1):49–67, February 2006.

A. Zare. Spatial-spectral unmixing using fuzzy local information. In *Proc. of the International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1139–1142. IEEE, 2011.

A. Zare and P. Gader. Piece-wise convex spatial-spectral unmixing of hyperspectral imagery using possibilistic and fuzzy clustering. In *Fuzzy Systems (FUZZ), 2011 IEEE International Conference on*, pages 741–746. IEEE, 2011.

F. Zhang and E. R. Hancock. Graph spectral image smoothing using the heat kernel. *Pattern Recognition*, 41(11):3328–3342, 2008.

T. Zhang, A. Popescul, and B. Dom. Linear prediction models with graph regularization for web-page categorization. In *Proc. of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 821–826, 2006.

Y. Zhang, H. L. Yang, D. Lunga, S. Prasad, and M. Crawford. Spatial context driven manifold learning for hyperpsectral image classification. In *Proc. of the Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*. IEEE, 2014.

D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. *NIPS*, 16(16):321–328, 2004.

B. Zhukov, D. Oertel, F. Lanzl, and G. Reinhackel. Unmixing-based multisensor multiresolution image fusion. *IEEE Transactions on Geoscience and Remote Sensing*, 37(3):1212–1226, 1999.