Subspace methods and bio-inspired optimization algorithms for denoising of multidimensionals signals and applications

Abir Zidi

AIX-MARSEILLE UNIVERSITE

ÉCOLE CENTRALE MARSEILLE

Ecole Doctorale Physique et Sciences de la Matière

Fraunhofer IIS

Institut Fresnel

Thèse présentée pour obtenir le grade universitaire de docteur

Discipline : Optique, Photonique et Traitement d'Image

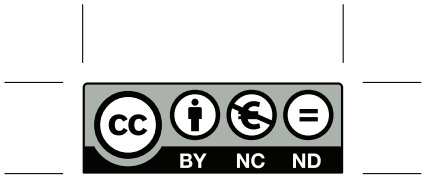
Abir ZIDI

Méthodes par sous-espaces et algorithmes d'optimisation bio-inspirés pour le débruitage de signaux multidimensionnels et applications

Soutenue le 12/06/2017 devant le jury composé de :

M. Camel TANOUGAST	Université de Lorraine	Rapporteur
M. Yide WANG	Université de de Nantes	Rapporteur
M. Rachid OUTBIB	Aix Marseille Université	Examinateur
Mme Véronique SERFATY	Direction générale de l'armement	Examinateur
M. Klaus Spinnler	Fraunhofer Institute IIS	Examinateur
M. Salah BOURENNANE	École Centrale de Marseille	Directeur de thèse
M. Julien MAROT	Aix Marseille Université	Co-Directeur de thèse

Numéro national de thèse/suffixe local : 2017AIXM0001/001ED352



Résumé étendu

Avec le développement des technologies, les scientifiques ont souhaité observer la surface terrestre du ciel ou de l'espace afin qu'ils puissent mieux gérer les ressources de la Terre. Néanmoins, pour identifier différents matériaux sur le terrain par des moyens d'image directe, il faut une caméra de résolution spatiale très élevée, ce qui est assez difficile et coûteux à construire et à fonctionner. En outre, comme l'ordre de la résolution spatiale n'est que des centimètres, cela rend le volume de données d'image extrêmement grand, de sorte que ce soit aussi un gros problème pour stocker et analyser les données d'image dans l'ordinateur. En conséquence, les scientifiques ont développé le mode d'imagerie hyperspectrale pour réduire la résolution spatiale [70]. Les images hyper-spectrales (HSI) sont acquises par les instruments de télédétection, nommés capteurs hyperspectraux, généralement transportés par des avions ou des satellites, mais aussi pour imager des scènes à une échelle plus réduite en laboratoire. Les capteurs hyperspectraux collectent non seulement des informations spatiales, mais aussi un rayonnement spectral dans des centaines de bandes étroites et contiguës d'environ 10 nm, couvrant le visible (de 400nm à 700 nm), l'infrarouge proche (de 700 nm à 1100 nm) et la longueur d'onde infrarouge à ondes courtes de 1100 nm à 2500 nm [83, 139].

L'image hyperspectrale acquise est toujours décrite par une structure tridimensionnelle. Elle peut être modélisée en cube de données 3D, avec deux dimensions spatiales (domaine spatial) et une dimension spectrale (domaine spectral), comme le montre la figure 1. Ainsi, chaque bande du domaine spectral est une image 2D couverte par le champ de vision du capteur, tandis que chaque pixel du domaine spatial est un vecteur constitué d'un grand nombre de longueurs d'onde. En outre, tous les matériaux peuvent être identifiés en analysant les spectres de rayonnement, également appelés signatures de rayonnement, car ils absorbent, reflètent et émettent des rayonnements d'une manière unique [125]. Par conséquent, en analysant les signatures spectrales, l'image hyperspectrale peut distinguer différents matériaux au sol même avec une résolution spatiale relativement faible. Pour cette raison, les

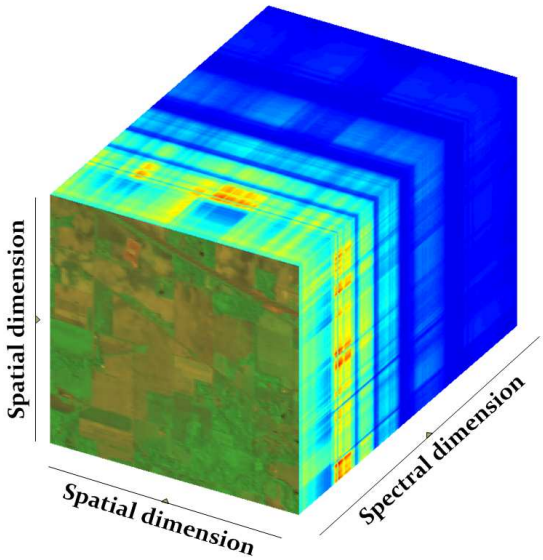


FIGURE 1 – Illustration : un cube de données

images hyperspectrales attirent de plus en plus d'intérêt ces dernières années dans différents domaines, comme la géographie, ou l'agriculture [63, 81, 134]. On utilise l'imagerie hyperspectrales pour effectuer la détection de cible [142] ou la classification [115] pour détecter les objets ou les matériaux intéressants sur le sol ou dans certains matériaux. Dans ce qui suit, nous énumérons certains champs où les HSI sont habituellement utilisés :

- **Mines et géologie :** car l'imagerie hyperspectrale peut identifier les matériaux [45] ou l'huile [34] du ciel ou de l'espace, c'est une façon plus appropriée d'explorer les minéraux que les observations sur le terrain d'un endroit à l'autre.

- **Agriculture et surveillance :** l'imagerie aérienne ou satellite à haute résolution est utile pour évaluer la croissance des cultures [15] et la surveillance à des fins militaires, telles que la surveillance des cibles terrestres militaires [129].

- **Surveillance de l'environnement :** en analysant les spectres recueillis par imagerie hyperspectrale, on trouve les substances toxiques laissées dans l'environnement [149].

- **Imagerie chimique :** l'imagerie hyperspectrale a également été appliquée

dans le domaine chimique pour déterminer les agents chimiques [149].

Le débruitage est donc une étape cruciale pour le traitement des données dans ces spécialités tres pointues : On se doit d'être rigoureux. Notamment, il a été montré que la formalisme de la transformée en ondelettes et les méthodes par sous-espaces insérées dans le contexte du traitement du signal tensoriel sont d'un intérêt certain pour le débruitage des données multidimensionnelles. Cependant il existe de nombreux paramètres à régler si l'on souhaite appliquer de telles méthodes, notamment les valeurs de rang ou dimension du sous espace signal pour tous les modes des coefficients issus de la transformée en ondelettes.

Ainsi on a consacrée cette thèse à la notion d'estimation de la dimension du sous-espace signal. Nous montrons d'abord l'intérêt du choix des valeurs correctes de rang que ce soit pour tronquer une décomposition de Parafac, ou de Tucker, des données. Ensuite, nous nous sommes intéressés à l'estimation automatique des valeurs optimales de rang, par des méthodes d'optimisation bio-inspirées, qui s'inspire du comportement des groupes d'animaux tels les oiseaux en recherche de nourriture.

On a développée des méthodes d'estimation de ces rangs dans le cadre de la transformée en ondelettes. Pour cette étude, nous avons eu recours à la décomposition en paquets d'ondelettes et à l'algèbre multilinéaire. Nous avons appliqué les méthodes de débruitage proposées à diverses images multidimensionnelles :

- Images RGB.
- images multispectrales extraites d'images hyperspectrales de pièces métalliques.
- Images par fluorescence des plantes.
- Images RX multispectrales.

Finalement, une étude comparative a été réalisée avec trois principaux types d'algorithmes : d'une part, la méthode de Perona-Malik basée sur la diffusion ; deuxièmement, la troncature de la HOSVD (higher order singular value decomposition) et MWF (multiway Wiener filtering) et troisièmement, un procédé basé sur la décomposition en paquets d'ondelettes et MWF, où les dimensions du sous-espace de signal sont estimées par un critère statistique plutôt que par une méthode d'optimisation. Les résultats sont prometteurs en termes de débruitage en réalité terrain.

En définitive, dans un contexte appliqué, nous aboutissons un gain de temps avantageux durant l'acquisition des images hyperspectrales. En effet, en diminuant le temps d'exposition pour chaque ligne de l'image acquise, la valeur du rapport signal sur bruit diminue, ce que nous compensons par la méthode de débruitage proposée.

Acknowledgement

First of all I would like to thank in a special way my thesis advisors, Pr. Salah BOURENNANE and Dr. Julien MAROT for providing an excellent research environment and for the precious freedom they always gave me to try out my own ideas. And, especially, for their help and support during all these years. I am also very grateful Mr. Klaus Spinnler. Without them him thesis would have not been possible. They proposed the subject of the thesis and offered me everything I needed to complete this work, but, more importantly, they gave me also the strength to believe in this work. It has been both an honor and a pleasure to work with them. I would also like to thank the jury members ; in particular Pr. Camel TANOUGAST and Pr. Yide WANG for their interest in my work by accepting the tedious task of reviewers and for the relevance of their observations and the accuracy of their assessments, and Pr. Rachid OUTBIB for giving me the honor to preside it.

This work was held at the Fraunhofer institute(Furth,Germany),based on a collaboration with Fresnel institute (Marseille, France). I express my gratitude to the Bavarian research foundation (BFS : Bayerische Forschungsstiftung) for supporting this researches. And, of course, my deepest gratitude to all colleagues of the Fraunhofer IIS, and GSM teams, particularly to my colleagues in the PMO department, for providing a very pleasant environment and a stimulating atmosphere.

Finally, I would like to extend my thanks to all my family and friends. To my father and mother, my husband and my sisters. Thanks for putting up with me all these years ! You have been my strength during the elaboration of this thesis. Thanks for your words of encouragement, your understanding, your loyal support. Thanks for all the moments we shared together. Thanks for your precious friendship.

Résumé

Cette thèse est consacrée à l'étude des rangs matriciels et tensoriels des données multidimensionnelles, et au développement de méthodes d'estimation de ces rangs dans le cadre de la transformée en ondelettes. Pour cette étude, nous avons eu recours à la décomposition en paquets d'ondelettes et à l'algèbre multilinéaire. Une méthode d'optimisation stochastique bio-inspirée a été adaptée, avec pour objectif final de supprimer le bruit dans des images multidimensionnelles. Pour cela nous avons estimé les différentes valeurs des dimensions du sous-espace de tenseur pour tous les modes des coefficients des paquets d'ondelettes. Nous avons appliqué les méthodes de débruitage proposées à diverses images multidimensionnelles : images RGB, images multispectrales extraites d'images hyperspectrales de pièces métalliques, images par fluorescence des plantes, et images RX multispectrales. Finalement, une étude comparative a été réalisée avec trois principaux types d'algorithmes : d'une part, la méthode de Perona-Malik basée sur la diffusion ; deuxièmement, la troncature de HOSVD (Higher-Order Singular Value Decomposition) et MWF (Multiway Wiener Filtering) et troisièmement, un procédé basé sur la décomposition en paquets d'ondelettes et MWF (Multiway Wiener Filtering), où les dimensions du sous-espace de signal sont estimées par un critère statistique plutôt que par une méthode d'optimisation. Les résultats sont prometteurs en termes de débruitage en réalité terrain. En définitive, nous aboutissons à un gain de temps avantageux durant le traitement des images hyperspectrales.

Mots clés : débruitage, imagerie hyperspectrale, optimisation, ondelette, méthodes par sous-espaces, rang tensoriel.

Abstract

This thesis is devoted to study matrix and tensor ranks of multidimensional signals and to the development of methods for estimating these ranks in the framework of the wavelet transform. For this study, we used the wavelet packet decomposition and the multilinear algebra. A bio-inspired stochastic optimization method has been adapted, with the ultimate objective of suppressing noise in multidimensional images. In order to ensure this, we have estimated the different values of the dimensions of the tensor subspace for all the modes of the coefficients of the wavelet packets.We have applied the proposed denoising methods to various multidimensional images: RGB images, multispectral images extracted from hyperspectral images of metal parts, plant fluorescence images, and multispectral RX images. Finally, a comparative study was carried out with three main types of algorithms: on the one hand, the Perona-Malik method based on diffusion; Second, the truncation of HOSVD and MWF, and thirdly, a method based on wavelet packet decomposition and MWF, where the dimensions of the signal subspace are estimated by a statistical criterion rather than by an optimization method. The results are promising in terms of denoising in grund truth. Ultimately, we achieve an advantageous time saving during the acquisition of hyperspectral images.

Keywords: denoising, hyperspectral imaging, optimization, wavelet, subspace-based methods, tensor rank.

Contents

List of Tables

List of Figures

List of Publications

- **Journal paper**

  1. **A. Zidi**, J. Marot, S. Bourennane, and K. Spinnler, *Bio-inspired optimization algorithms for automatic estimation of multiple subspace dimensions in a tensor-wavelet denoising algorithm*, Journal of Remote Sensing Technology, vol. 4, no 1, pp. 90-114, dec. 2016.

- **International Conferences and Workshops**

  1. **A. Zidi**, J. Marot, S. Bourennane, K. Spinnler, "*Automatic rank estimation of parafac decomposition and application to multispectral imagewavelet denoising.*", IEEE International Conference on Image Processing (ICIP), Arizona, USA, Sept 2016.
  2. **A. Zidi**, J. Marot, S. Bourennane, K. Spinnler, "*Multispectral image denoising in wavelet domain with unsupervised tensor based method.*", In Visual Information Processing (EUVIP), Marseille, France, 2016 6 th European Workshop on. IEEE, 2016.
  3. **A. Zidi**, J. Marot, S. Bourennane, K. Spinnler, "*Unmixing of Hyperspectral Images with Pure Prior Spectral Pixels*", In VISAPP, Berlin, Germany, 2015.
  4. **A. Zidi**, J. Marot, S. Bourennane, "*Nonnegative matrix factorization with spatial prior and reference spectra application to remote hyperspectral image understanding*", EUVIP, Paris, France, 2014.

Chapter 1

Introduction

Multidimensional digital signal processing can be defined as the theoretical underpinnings for processing multidimensional signals independent of a specific application [14]. The operation of filtering, sampling, spectrum analysis, and signal representation are basic to all of signal processing. There are many applications of multidimensional signal processing, for instance image and video processing [23, 122, 128], medical imaging [104, 152], array processing [108, 110] and so on.

Algorithms of multidimensional signal processing can be grouped into four categories [14]:

1. Separable algorithms using one-dimensional (1-D) operators to deal with the rows and columns of a multidimensional array.
2. Non-separable algorithms that borrow their derivation from their 1-D counterparts.
3. Multidimensional algorithms significantly different from their 1-D counterparts.
4. Multidimensional algorithms without 1-D counterparts. The former three categories are listed in increasing order of their mathematical and computational complexity and in decreasing order of their popularity.

Separable algorithms deal with the rows and columns of a multidimensional signal sequentially and they have been widely used for image processing since the 1960s due to lower computational complexity than older nonseparable algorithms.

Algorithms of the second category are uniquely multidimensional since they cannot be decomposed into a repetition of 1-D procedures. Some familiar 1-D methodologies do not extend to the multidimensional case or cannot extend because of the absence of a factorization theorem for multidimensional polynomials and the assumption of causality not appropriate for multidimensional signal processing applications.

To process multidimensional data, the algorithms in the former three categories above have some common drawbacks. They can not deal with multidimensional data as a whole. Processing the rows or columns of a multidimensional signal is one solution. Another way requires rearranging data from multidimensional to two dimensional. But these algorithms cannot make full use of or even ignore the relationships among different dimensions because the intuitive representation of a multidimensional signal is a multidimensional array.

1.1 General considerations about tensor decompositions

In the fourth categories, there are some algorithms that perform tensor decompositions, where a tensor is defined as a multidimensional array [60, 62, 82]. Tensor decompositions originated with Hitchcock in 1927 [46, 47] and the idea of a multi-way model is attributed to Cattell in 1944 [7]. These concepts received scant attention until the work of Tucker in the 1960s [138] and Carroll and Chang [5] and Harshman [42] in 1970, all of which appeared in psychometrics literature. In the last ten years, interest in tensor decompositions has expanded to other fields. Examples include image processing [105], numerical linear algebra [71], communication [109] and so on.

Two main decomposition models for multidimensional arrays have been developed: Tucker3 decomposition [138] and PARAFAC/CANDECOMP (Canonical Decomposition / Parallel Factor Analysis) decomposition [5] [42]. Tucker3 decomposition is a form of higher-order principal component analysis and it decomposes a tensor into a core tensor multiplied by a matrix along each mode. Thus it is necessary to estimate multiple ranks for Tucker3 decomposition, which leads to the non-uniqueness of this decomposition. Parallel factor analysis extends the ideas and methods of standard two-way factor analysis to multi-way data. A key motivation

for this is the possibility that 'simultaneous factor analyses' of several matrices in parallel can sometimes lead to a unique set of factors for which any rotation of axes will cause a loss in total fit. So low-rank PARAFAC decomposition can be unique for rank higher than one, and this is the distinguishing characteristic of PARAFAC.

1.2 General state-of-the-art about swarm intelligence

Swarm intelligence algorithms, together with evolutionary, and physics-based algorithms, compose the meta-heuristic algorithms. Meta-heuristics may be classified into three main classes: evolutionary, physics-based, and SI algorithms. The most popular among evolutionary algorithms are the genetic algorithms (GA). This algorithm was proposed by Holland in 1992 [49] and simulates Darwnian evolution concepts. In their seminal work concerning particle swarm optimization, Kennedy and Eberhart [31, 55] got inspired by [116] where the term 'particle swarm' was chosen to define the members of a population or test set. In their paradigm, the population members are mass-less and volume-less. Their evolution is described through position, speed, and acceleration parameters [94]. An often cited, now well-known reference [100] introduces genetic algorithms in the context of evolutionary computation which implies the evolution of a population of candidates which is inspired by Darwin's natural selection theory. Another largely cited reference presents basics about genetics, the hierarchical genetic algorithm, and applications to H∞ control, neural network, and speech recognition [93].

Mainly, the rest of the swarm intelligence (SI) techniques which mimic the behavior of groups of animals are cited in [99].

We now focus on methods which get inspired by the Human behavior, at the individual scale through emotion modeling and at a larger scale through the modeling of tribes.

In [150, 151] a social-emotional PSO (SEPSO) algorithm is proposed. To the best of our knowledge, it is a seminal work in the sense that no swarm intelligence algorithm is inspired by a Human behavior. A social emotional model is proposed to rule the velocity of each particle across the iterations: firstly, each particle is rather depressed and does not learn from the society, secondly it is calm to think and learn from the society, and thirdly passionate to learn. The personal best influences the

velocity in the first and second cases, and the global best influences the velocity in the second and third cases. In a problem involving four parameters to estimate, SEPSO has superior global optimization ability and rate of convergence than PSO. This algorithm is then inspired by the behavior of persons at the individual scale (their emotion).
Tribes PSO has first been proposed in [18] and further developed in [13]. Its performance has been analyzed in [20]. The principles of tribes PSO is to reduce the number of parameters in PSO, through an adaptive process which modifies autonomously the number of particles.

1.3 Objectives of the thesis

The objective of this thesis is to evaluate the interest of multidimensional images in various contexts. Especially, we focus on multispectral and hyperspectral images. But, the noise in HSIs degrades the performances of the target detection and the classification. In order to alleviate this problem, it is necessary to reduce the noise in HSIs before the target detection or classification. Indeed, the prospects of this thesis aims at the detection and classification of very small defects on industrial metal parts. Firstly, multilinear algebra based methods are developed to HSI processing, for instance, for denoising applications, a tensor is defined a multidimensional array, and HSI is modeled as tensor data. Conversely to the conventional methods, they consider the spectral relationship among bands and process tensor data as a whole entity. After studying different algorithms, we observed that subspace-based methods is adapted for removing noise in images. The subspace-based approach can be extended to multidimensional data. Multiway Wiener filtering is an extension to subspace-based approach to tensorial case. It was then placed in a wavelet framework. A subspace-based method requiring the estimation of ranks, usually performed with the statistical Akaike information criterion (AIC) [119], working best with a very high number of signal realizations.

So, we aim at estimating the signal subspace dimensions for all modes of wavelet packet coefficients by minimizing the least squares criterion with the best possible optimization strategy.

Optimization is a commonly encountered mathematical problem in all engineering disciplines. It literally means finding the best possible and desirable solution. Optimization problems are wide ranging and numerous. But former methods to solve optimization problems require enormous computational efforts, particularly

in the field of multispectral, which tend to fail as the problem size increases. To resolve this disadvantage, the nature is of course a great and immense source of inspiration for solving hard and complex problems in computer science since it exhibits extremely diverse, dynamic, robust, complex and fascinating phenomenon. Bio-inspired optimization algorithms are computationally efficient alternatives to deterministic approaches. Meta-heuristics are based on the iterative improvement of either a population of solutions (as in Evolutionary algorithms, Swarm based algorithms) or a single solution (eg. Tabu Search) and mostly employ randomization and local search to solve a given optimization problem. So in this approach A bio-inspired stochastic optimization method has been adapted, with the ultimate objective of suppressing noise in multidimensional images. In order to ensure this, we have estimated the different values of the dimensions of the tensor subspace for all the modes of the coefficients of the wavelet packets. We have applied the proposed denoising methods to various multidimensional images: RGB images, plant fluorescence images, and multispectral RX images. To inspect the effectiveness of our algorithms, a comparative study was carried out with three main types of algorithms: on the one hand, the Perona-Malik method based on diffusion; second, the truncation of HOSVD and MWF, and thirdly, a method based on wavelet packet decomposition and MWF, where the dimensions of the signal subspace are estimated by a statistical criterion rather than by an optimization method. The results are promising in terms of denoising in ground truth. Ultimately, we achieve an advantageous time saving during the acquisition of hyperspectral images.

After finishing the theoretical part. We decided to build a setup to handle hyperspectral images. We encountered a problem of illumination for metal surfaces. To solve this problem, we have constructed a test setup in Fresnel institut. We reproduced this setup in Fraunhofer institut.

1.4 Document organization

This manuscript includes three chapters and a conclusion. The below paragraphs describe the content of each chapter.

1. Chapter 2 presents the state of the art. Firstly, it presents the fundamental concepts of HSIs. This chapter also presents the main tools of multilinear algebra which are conventionally used for the processing of hyperspectral images. More particularly it recalls some definitions and properties of matrix

and tensors, Finally, it presents two commonly used tensor decompositions, including theoretical examples.

2. Chapter 3 is devoted to the denoising of multidimensional data by a tensor subspace-based method. A bio-inspired stochastic optimization method has been adapted, with the ultimate objective of suppressing noise in multidimensional images. In order to ensure this, we have estimated the different values of the dimensions of the tensor subspace for all the modes of the coefficients of the wavelet packets. Finally, a comparative study was carried out with three main types of algorithms.

3. Chapter 4 solves illumination problems caused by the reflection of light on the metal surfaces. It begins by presenting the setup used in Fresnel institute to test his utility then we do the same setup in Fraunhofer institute in order to acquire hyperspectral images of metal surfaces and to test our algorithm.

4. Appendix present different setups. The first is an experimental setup which is a hyperspectral fluorescence setup in the IRSTEA Laboratory. The second setup is a multispectral X-Ray setup used in CPPM Laboratory. Precisely, it is meant for X-ray spectral tomography.

We finalize the manuscript by a general conclusion, as well as further prospects.

1.5 Publications related to this thesis

This PhD thesis follows some advances performed during research work on master thesis was were performed on the field of non-negative matrix factorization. In [153] nonnegative matrix factorization with spatial prior and reference spectra application to improve hyperspectral image understanding is considered. In [156] the unmixing of hyperspectral images with Pure Prior Spectral Pixels is considered. The contents of the manuscript and its relationship with the papers published during this thesis are as follows:
Chapter 3 investigates the automatic estimation of either Tucker or Parafac ranks in a wavelet framework, with bio-inspired optimization methods.
In [157], we investigated Multispectral image denoising in wavelet domain with unsupervised tensor subspace-based method: a short comparison of a genetic algorithm, particle swarm optimization, and the statistical criterion "Akaike Information Criterion" is provided with a view to estimate subspace ranks in a wavelet framework. In [155] the topic of Bio-Inspired Optimization Algorithms for Automatic Es-

timation of Multiple Subspace Dimensions in a Tensor-Wavelet Denoising Algorithm is deeply investigated. It deals with the estimation of triplets of ranks, precisely the ranks of the Tucker decomposition, in the frame of wavelet packet decomposition. In [154] this work is extended to the Parafac decomposition, with an application to multispectral image denoising.

Chapter 2

Brief state of the art

2.1 Introduction

Hyperspectral sensors allow simultaneous acquisition of several informations, especially with the quasi-continuous acquisition of spectral data for all pixels of the scene, from ultraviolet to near infrared, which corresponds to several hundred images associated with different spectral bands for the same scene. This very large number of data inevitably increases the complexity of the treatment, making less effective traditional methods of image processing. In order to exploit this information, it was necessary to develop new methods of hyperspectral image processing. The multi-linear algebra tools are particularly well suited for analyzing this type of multidimensional data. Indeed, numerically, a data that is depending of several parameters can be modeled as a multi-input array. Each input corresponds to one parameter. For example, a hyperspectral image can be represented by an array of three inputs: two inputs for the spatial information (indexes of the spatial coordinates of the pixels) and one input for the spectral information (index of the spectral band). Each input in the array is associated with a sampled physical quantity. These multidimensional arrays allow a unique and global representation of the spectral data.

The treatment of these arrays requires the use of multi-linear algebra tools. Actually, several methods based on linear algebra tools are proposed in the literature [3], [4] for the analysis of hyperspectral images. The most common applications are:

- Detection, for researching and identifying specific objects.
- Classification, for characterizing and grouping objects according to a criterion.

- Unmixing, for estimating the spectra and/or the abundance of each material in the scene.

All these applications usually need multiple preprocessing of hyperspectral data in order to enhance their performance, such as spectral dimension reduction or denoising the data, which are corrupted by noise from various phenomena, such as the atmospheric disturbance, sensors' noise, etc.

2.2 Hyperspectral imagery

Hyperspectral Imaging (HSI) is a technique that combines digital imaging with spectroscopy. HSI collects and processes information from across the electromagnetic spectrum as a function of the wavelength, and produces hyperspectral images with instruments called imaging spectrometers. Spectral imagers form and analyze the spectral radiances at each pixel in the scene, where the objects are characterized by their spatial shape and their spectral radiance. The spectral radiance of an object is its reflected light intensity as a function of wavelength, which is indicative of the material forming the object. Although the human eye can distinguish spatial shape very well, it does not discern spectral radiance characteristics nearly as precisely. Instead, the human eye perceives only a dominant portion of the spectral radiance, which is perceived as the color of the object [10].

The spectral reflectance of an object is obtained by the ratio between the radiance and the scene illumination. In principle most objects can be identified by their spectral reflectance alone. The choice of the relevant spectral bands is crucial for the analysis. Indeed, the deflections of the spectral curves mark the wavelength ranges for which the material selectively absorbs the incident energy. These features are commonly called absorption bands. The overall shape of a spectral curve and the position and strength of absorption bands in many cases can be used to identify and discriminate different materials. For example, vegetation has higher reflectance in the near infrared range and lower reflectance of red light than soils. However, because the human eye is a relatively unsophisticated observer, it is not always possible to distinguish objects based upon their perceived colors.

Multispectral remote sensors such as the Landsat Thematic Mapper and SPOT XS produce images with a few relatively broad wavelength bands. Hyperspectral remote sensors, on the other hand, collect image data simultaneously in hundreds or thousands of narrow, adjacent spectral bands. These measurements make it possible to derive a continuous spectrum for each image cell, as shown in the Fig 2.1.

Hyperspectral images contain a wealth of data, but interpreting them requires an understanding of exactly what properties of materials we are trying to measure, and how they are related to the measurements actually made by the hyperspectral sensor.

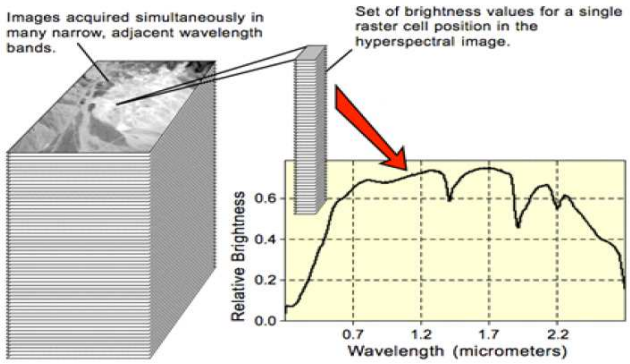


Figure 2.1: a plot of the brightness values versus wavelength shows the continuous spectrum for the image cell, which can be used to identify surface materials.

The development of these complex sensors has involved the convergence of two related but distinct technologies: spectroscopy and the remote imaging of Earth and planetary surfaces [17].

Spectroscopy is the study of light that is emitted by or reflected from materials and its variation in energy with wavelength. As applied to the field of optical remote sensing, spectroscopy deals with the spectrum of sunlight that is diffusely reflected (scattered) by materials at the Earth's surface. Instruments called spectrometers (or spectroradiometers) are used to make ground-based or laboratory measurements of the light reflected from a test material. An optical dispersing element such as a grating or prism in the spectrometer splits this light into many narrow, adjacent wavelength bands and the energy in each band is measured by a separate detector. By using hundreds or even thousands of detectors, spectrometers can make spectral measurements of bands as narrow as $0.01\ \mu m$ over a wide wavelength range, typically at least 0.4 to $2.4\ \mu m$ (visible through middle infrared wavelength ranges) in remote-sensing applications. For other applications, such as astronomy, different wavelength ranges can be used (e.g. UV, visible, infrared and radio wavelength ranges).

Remote imagers are designed to focus and measure the light reflected from many adjacent areas on the Earth's surface. Recent advances have allowed the design

of imagers that have spectral ranges and resolutions comparable to ground-based spectrometers.

2.2.1 Spectral reflectance

In reflected-light spectroscopy the fundamental property that we want to obtain is spectral reflectance: the ratio of reflected energy to incident energy as a function of wavelength. Reflectance varies with wavelength for most materials because energy at certain wavelengths is scattered or absorbed to different degrees. These reflectance variations are clear when we compare spectral reflectance curves (plots of reflectance versus wavelength) for different materials, as in Fig 2.2. The overall shape of a spectral curve and the position and strength of absorption bands in many cases can be used to identify and discriminate different materials. For example, vegetation has higher reflectance in the near infrared range and lower reflectance in the red range than soils.

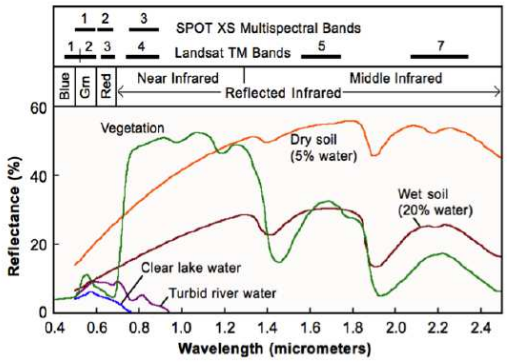


Figure 2.2: representative spectral reflectance curves for several common Earth surface materials over the visible light to reflected infrared spectral range.

2.2.2 Spectral libraries

Several libraries of reflectance spectra of natural and man-made materials are available for public use. These libraries provide a source of reference spectra that can aid the interpretation of hyperspectral and multispectral images.

- ASTER Spectral Library: this library has been made available by NASA as part of the advanced spaceborne thermal emission and reflection radiometer (ASTER) imaging instrument program. It includes spectral compilations from

NASA's Jet Propulsion Laboratory, Johns Hopkins University, and the United States Geological Survey (Reston). The ASTER spectral library currently contains nearly 2000 spectra, including minerals, rocks, soils, man-made materials, water, and snow. Many of the spectra cover the entire wavelength region from 0.4 to 14 μm. The library is accessible interactively via the Worldwide Web at http://speclib.jpl.nasa.gov.

- USGS Spectral Library: the united states geological survey spectroscopy lab in Denver, Colorado has compiled a library of about 500 reflectance spectra of minerals and a few plants over the wavelength range from 0.2 to 3.0 μm.
  This library is accessible online at :
  http://speclab.cr.usgs.gov/spectral.lib04/spectral-lib04.html.

2.2.3 Hyperspectral images

Color images have three layers (or bands) that each has different information. It is possible to make even more layers by using smaller wavelength bands, say 20 nm wide between 400 nm and 800 nm. Then each pixel would be a spectrum of 21 wavelength bands. This is the multivariate image. The 21 wavelength bands in the example are called the image variables and in general there are K variables. An $I \times J$ image in K variables would form a three-way array of size $I \times J \times K$.

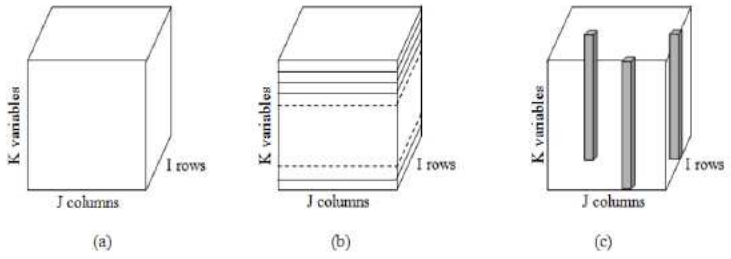


Figure 2.3: (a) an $I \times J$ image in K variables is an $I \times J \times K$ array of data. (b) the $I \times J \times K$ image can be presented as K slices where each slice is a grey-value image. (c) the $I \times J \times K$ images can be presented as an image of vectors. In special cases, the vectors can be shown and interpreted as spectra.

With more than three wavelength bands, simple color representation is not pos-

sible, but some artificial color images may be made by combining any three bands. In that case the colors are not real and are called pseudo-colors.

Many imaging techniques make it possible to make multivariate images and their number is constantly growing. Also, the number of variables available is constantly growing. From about 100 variables upwards the name hyperspectral images was coined in the field of satellite and airborne imaging, but hyperspectral imaging is also available in laboratories and hospitals.

Images as in Fig 2.3 with $K = 2$ or more are multivariate images. Multivariate images can also be mixed mode, e.g. an UV wavelength image, a near infrared (NIR) image and a polarization image in white light. In this case, the vector of three variables is not really a spectrum.

So, the hyperspectral images characterize:

- many wavelength or other variables bands, often more than 100;
- the possibility to express a pixel as a spectrum with spectral interpretation, spectral transformation, spectral data analysis, etc.

Many principles from physics can be used to generate multivariate and hyperspectral images [12]. Other variables, e.g. time, can be used to generate sequencies of images and construct multivariate images. Examples of making NIR optical images are used to illustrate some general principles.

A classical spectrophotometer consists of a light source, a monochromator or filter system to disperse the light into wavelength bands, a sample presentation unit and a detection system including both a detector and digitalization/storage hardware and software. The most common sources for broad spectral NIR radiation are tungsten halogen or xenon gas plasma lamps. Light emitting diodes and tunable lasers may also be used for illumination with less broad wavelength bands. In this case, more diodes or more lasers are needed to cover the whole NIR spectral range ($780 - 2500\ nm$). For broad spectral sources, selection of wavelength bands can be based on specific bandpass filters based on simple interference filters, liquid crystal tunable filters (LCTFs), or acousto-optic tunable filters (AOTFs), or the spectral energy may be dispersed by a grating device or a prism-grating-prism (PGP) filter. Scanning interferometers can also be used to acquire NIR spectra from a single spot.

A spectrometer camera designed for hyperspectral imaging has the hardware components listed above for acquisition of spectral information plus additional hardware necessary for the acquisition of spatial information. The spatial information comes from measurement directly through the spectrometer optics or by controlled

positioning of the sample, or by a combination of both. Three basic camera configurations are used based on the type of spatial information acquired; they are called point scan, line scan or plane scan.

2.2.4 Advantages and disadvantages of HSI

The primary advantage to hyperspectral imaging is that, because an entire spectrum is acquired at each point, the operator holds all available information from the dataset to be mined. HSI can also take advantage of the spatial relationships among the different spectra in a neighborhood, allowing more elaborate spectral-spatial models for a more accurate segmentation and classification of the image.

The primary disadvantages are cost and complexity. Fast computers, sensitive detectors, and large data storage capacities are needed for analyzing hyperspectral data. Moreover, in high dimensionality, it comes difficult to perform accurate parameters estimation, for example in the Bayesian context, and the distance measures lose their efficiency to distinguish between different vectors (see section 2.4).

2.3 Representation of hyperspectral images

By the nature of the hyperspectral data, in which each pixel is a vector, the data are typically represented by a hyperspectral cube. Because of this cubic representation of the data, it is natural to consider the use of tensors of order 3 as a mathematical model for the analysis of hyperspectral images. Typically, the spatial dimensions are respectively associated to 1-mode and 2-mode of the tensor and the spectral dimension is associated to the 3-mode of the tensor, see Fig 2.4.

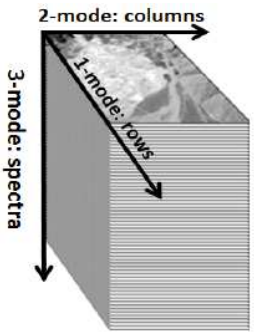


Figure 2.4: illustration of the tensor representation of a hyperspectral image.

The folding matrix in the spectral mode (3-mode) of a tensor is of major interest in the study of data compared to the folding matrices of the spatial modes. Indeed, the folding matrix in the spectral mode allows a concrete physical representation of the spectral data where each column of the folding matrix represents the spectrum of a pixel, unlike the two folding matrices of spatial modes that are more difficult to interpret, see Fig 2.5.

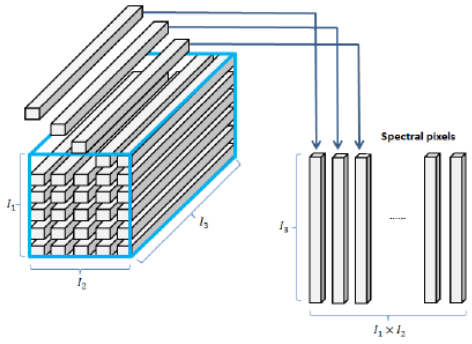


Figure 2.5: folding matrix of the spectral mode corresponding to the column matrix of all the spectral pixels of a hyperspectral image.

Indeed, the folding 3-mode matrix is currently used for spectral analysis. However, introducing spatial information often allows to increase the performances [124].

2.4 Denoising and dimensionality reduction

In the case of HSI, the number of the acquired images can reach hundreds or even exceed thousand images. It is necessary to have a good spectral resolution and a sufficiently large band width for accurate analysis of the information. Indeed, this large number of images involves manipulating vector spaces of high dimension. In more challenging algorithms, the large dimension of the vector spaces, time consumption and significant storage involve a decrease in the accuracy of the statistical estimation for a fixed number of samples. This phenomenon is well known in hyperspectral image processing under the names of "Hughes phenomenon" and "the curses of the high dimensionality" [2], [29]. More often, the acquisition step is followed by a signal space dimensionality reduction, where a subspace signal is estimated. The identification of this subspace enables a correct dimensionality reduction, yielding gains in algorithm performance and complexity, and in data storage. This is a cru-

cial first step in many hyperspectral processing algorithms such as target detection, change detection, classification, and unmixing.

The noise add to the raw data is not only specific to hyperspectral data but is a well-known image and signal-processing problem. Indeed, during the acquisition step of multivariate images from a sensor (HSI camera, IR camera or other), there are still several phenomena disturbing the acquisition process. The disturbances can be directly related to the quality of the sensor (such as electronic noise, the photon noise, aberrations of the optical system) or related to the environment in which the acquisition takes place (such as atmospheric disturbances) [1], [11], [39] [112]. Even low power, this acquisition noise affects the efficiency of the detection and classification algorithms based on spectral identification. In order to reduce the nuisance of these phenomena on the detection and classification methods, a pre-processing data denoising is commonly used to limit the influence on the results of detection and classification [111].

The steps of denoising and spectral dimensionality reduction are crucial stages of preprocessing for the analysis of hyperspectral images. Indeed, they directly affect the efficiency of the detection and classification methods. In recent years, numerous works have shown the interest of the denoising and spectral dimensionality reduction as pre-processing steps for the classification of hyperspectral images [10], [11] [111].

2.4.1 Denoising and dimensionality reduction algorithms

In hyperspectral images processing, the spectral dimensionality reduction and denoising methods are two processes with distinct goals, however, they are based on a similar approach, which consists to use the signal subspace. Usually the denoising methods tend to separate the signal subspace from the noise subspace, while the spectral dimension reduction methods seek to estimate the signal subspace in order to work on this reduced space with smaller dimension than the original data space. Whatever, these two methods can be combined in case where a denoising and data dimensionality reduction are both needed. The most popular used denoising and/or dimensionality reductions methods are: the singular value decomposition (SVD), maximum noise fraction (MNF) [39] and hyperspectral signal identification by minimum error (HySime) [4].

Traditionally, SVD and MNF are used for dimension reduction, but they are also useful for visually identifying the dominant image components. HySime is a combination algorithm of dimensionality reduction and denoising tools, which also esti-

mates automatically the virtual dimension (VD) of the reduced data space.

2.5 Overview of Singular Value Decomposition

In the following chapters, we will often employ the notion of Singular Value Decomposition (SVD) for n-modal matrices (see the definition section 2.7.2) of tensor.
We remind in this paragraph the SVD process for a matrix. SVD is a method commonly used to analyze data in large domains such as signal and image processing (image recognition for face, for classification), economic forecasting, to study the evolution of animal and vegetal in biology. Singular value decomposition is extremely powerful and useful in a lot of contexts.

2.5.1 Singular Value Decomposition

The SVD of a matrix is a linear algebra method for matrix factorization, with many useful applications in signal processing and statistics. This factorization consists in finding in each of the two spaces, corresponding to the two dimensions associated with the matrix, an orthonormal basis.

Let $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2}$ rectangular matrix, $\mathbf{u} \in \mathbb{R}^{I_1 \times 1}$ a vector, $\mathbf{v} \in \mathbb{R}^{I_2 \times 1}$ a vector, and λ a scalar. If the relationship in Eq. (2.1) holds:

$$\mathbf{X}\mathbf{v} = \lambda \mathbf{u} \tag{2.1}$$

The vectors $\mathbf{u}$ and $\mathbf{v}$ are, respectively, left- and right- singular vectors of $\mathbf{X}$, and λ is a singular value of $\mathbf{X}$.

There exists a factorization of a matrix $\mathbf{X}$ into the form:

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \tag{2.2}$$

where:

$$\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_{I_1}] \in \mathbb{R}^{I_1 \times I_1} \tag{2.3}$$

$\mathbf{U}$ is an orthogonal matrix containing the left singular vectors.
The vectors $\{\mathbf{u}_k\}_{k=1,\cdots,I_1} \in \mathbb{R}^{I_1}$ constitute an orthonormal basis of the space $\mathbb{E}^{(1)}$, of dimension I_1, associated with the matrix, and are the eigenvectors of the symmetric

matrix $\mathbf{X}\mathbf{X}^T$.

$$\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_{I_2}] \in \mathbb{R}^{I_2 \times I_2} \tag{2.4}$$

In a similar way, $\mathbf{V}$ is an orthogonal matrix containing the right singular vectors. The vectors $\{\mathbf{v}_k\}_{k=1,\cdots,I_2} \in \mathbb{R}^{I_2}$ constitute an orthonormal basis of the space $\mathbb{E}^{(2)}$, of dimension I_2, associated with the matrix, and are the eigenvectors of the symmetric matrix $\mathbf{X}^T\mathbf{X}$.

Let I denote the minimum value between I_1 and I_2: $I = \min\{I_1, I_2\}$. Then $\mathbf{\Sigma}$ is a pseudo-diagonal matrix: if we denote $\tilde{\mathbf{\Sigma}} = \mathbf{diag}\,(\lambda_1, \cdots, \lambda_I)$, then $\mathbf{\Sigma} \in \mathbb{R}^{I_1 \times I_2}$ is a zero-padded version of $\tilde{\mathbf{\Sigma}}$

$$\mathbf{\Sigma} = \begin{pmatrix} \tilde{\mathbf{\Sigma}} & \mathbf{0} & \cdots & \mathbf{0} \end{pmatrix} \tag{2.5}$$

or

$$\mathbf{\Sigma} = \begin{pmatrix} \tilde{\mathbf{\Sigma}} \\ \mathbf{0} \\ \cdots \\ \mathbf{0} \end{pmatrix} \tag{2.6}$$

$\mathbf{\Sigma}$ is a quasi-diagonal matrix containing I ordered singular values, $\lambda_1 \geq \cdots \geq \lambda_I$, of the matrix $\mathbf{X}$. The singular values λ_k correspond to the square root of the eigenvalues β_k of the symmetric matrix $\mathbf{X}\mathbf{X}^T$.

In the case where $I_1 < I_2$, the decomposition of the matrix $\mathbf{X}$, Eq. (2.2) can be written as:

$$\mathbf{X} = \sum_{k=1}^{I_1} \sqrt{\lambda_k}\mathbf{u}_k\mathbf{v}_k^T = \sum_{k=1}^{I_1} \beta_k \mathbf{X}_k \tag{2.7}$$

where $\lambda_k = \beta_k^2$ is the k^{th} singular value and $\mathbf{X}_k$ is the corresponding proper matrix.

In the case of HSI, a matrix $\mathbf{X}$ is obtained through flattening. Thereafter, SVD decomposes the data space into a set of I components, arranged in a descending order of the corresponding singular values, where the first components contain the maximum variance and represent the most characteristic variability of the data. The spatial and spectral information is extracted from the original matrix in a compact manner. SVD is close to principal component analysis (PCA) with the difference that SVD simultaneously provides the PCAs in both row and column spaces.

For a particular flattened version of the HSI, the columns of matrix $\mathbf{U}$ consist of the left singular vectors that represent the spectral variation of the data set. The columns of matrix $\mathbf{V}$ are the right singular vectors that represent the spatial variation of the data set. Usually, original data can be adequately represented with only a few components. The reduced space $\mathbf{X}_R$ is obtained as:

$$\mathbf{X}_R = \sum_{k=1}^{K} \sqrt{\lambda_k} \mathbf{u}_k \mathbf{v}_k^T \tag{2.8}$$

where K is the desired dimension of the reduced data space.

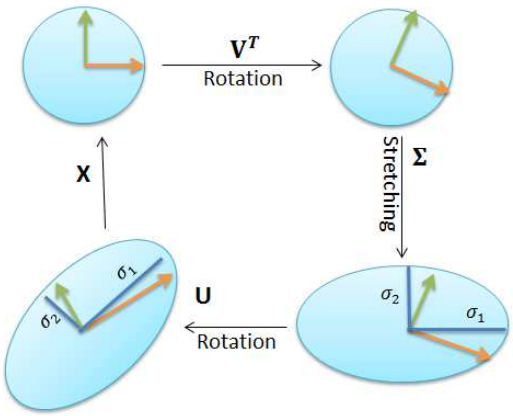


Figure 2.6: Geometry of the SVD

Intuitive interpretations

Geometric description: In the special, yet common case where $\mathbf{X}$ is an $m \times m$ real square matrix with positive determinant, $\mathbf{U}$, $\mathbf{V}^T$, and $\mathbf{\Sigma}$ are real $m \times m$ matrices as well, $\mathbf{\Sigma}$ can be regarded as a scaling matrix, and $\mathbf{U}$, $\mathbf{V}^T$ can be viewed as rotation matrices. Thus the expression $\mathbf{U\Sigma V}^T$ can be intuitively interpreted as a composition of three geometrical transformations: a rotation or reflection, a scaling, and another rotation or reflection. For instance, fig 2.6 above explains how a shear matrix can be described as such a sequence.

Singular values as semiaxes of an ellipse or ellipsoid: As shown in fig 2.6 , the singular values can be interpreted as the semiaxes of an ellipse in $2D$. This concept can be generalized to n-dimensional Euclidean space, with the singular values of any $n \times n$ square matrix being viewed as the semiaxes of an n-dimensional ellipsoid. See below for further details.

The columns of U and V are orthonormal bases: Since $\mathbf{U}$ and $\mathbf{V}^T$ are unitary, the columns of each of them form a set of orthonormal vectors, which can be regarded as basis vectors. The matrix $\mathbf{X}$ maps the basis vector $\mathbf{V_i}$ to the stretched unit vector $\lambda_i \mathbf{U_i}$. By the definition of a unitary matrix, the same is true for their conjugate transposes $\mathbf{U}^T$ and $\mathbf{V}$, except the geometric interpretation of the singular values as stretches is lost. In short, the columns of $\mathbf{U}$, $\mathbf{U}^T$, $\mathbf{V}$, and $\mathbf{V}^T$ are orthonormal bases.

2.5.2 Rank of a matrix

The rank of a matrix is defined as the maximum number of linearly independent column vectors in the matrix or the maximum number of linearly independent row vectors in the matrix. Both definitions are equivalent.

In the remainder of the manuscript, the rank of a matrix $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2}$ is defined as the number of nonzero singular values:

$$\mathbf{rank}(\mathbf{X}) = \arg\min_k(\lambda_k), \forall\ \lambda_k > 0 \tag{2.9}$$

where λ_k are the singular values of $\mathbf{X}$.
The rank of a matrix can also be defined as the dimension of the subspace vector generated by the row vectors (or columns) of $\mathbf{X}$ [73]

2.5.3 Matrix approximation - Eckart Young theorem

We consider a matrix $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2}$ of rank k and find a matrix $\mathbf{X_p}$ in the set of matrices $\mathbf{B} \in \mathbb{R}^{I_1 \times I_2}$ of rank $p < k$ which is the best possible approximation of the initial matrix $\mathbf{X}$ in the least squares sense by considering the Frobenius norm $\|.\|$:

$$\mathbf{X}_p = \underset{\mathbf{B} \in \mathbb{R}^{I_1 \times I_2}}{\arg\min} \left(\|\mathbf{X} - \mathbf{B}\|^2\right) \tag{2.10}$$

where the norm $\|\mathbf{X}-\mathbf{B}\|^2$ is obtained by:

$$\|\mathbf{X}-\mathbf{B}\|^2 = \text{trace}(\left[(\mathbf{X}-\mathbf{B})(\mathbf{X}-\mathbf{B})^T\right]) \tag{2.11}$$

According to the Eckart-Young theorem [32], It has been shown that the best approximation of the matrix $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2}$ by a matrix of the same dimension and of lower rank $p < k$ is the matrix $\mathbf{X}_p \in \mathbb{R}^{I_1 \times I_2}$ obtained from the p first singular vectors and singular values derived from the SVD of matrix $\mathbf{X}$ such as:

$$\mathbf{X}_p = \underset{\mathbf{B}\in\mathbb{R}^{I_1\times I_2}}{\arg\min} (\|\mathbf{X}-\mathbf{B}\|^2) = \mathbf{U}_p\mathbf{\Sigma}_p\mathbf{V}_p^T \tag{2.12}$$

in which:
the matrix $\mathbf{U}_p \in \mathbb{R}^{I_1 \times p}$ is obtained with the p first left singular vectors of the SVD of $\mathbf{X}$:

$$\mathbf{U}_p = [\mathbf{u}_1, \ldots, \mathbf{u}_p] \tag{2.13}$$

$\mathbf{V}_p \in \mathbb{R}^{I_2 \times p}$ is obtained with the p first right singular vectors of the SVD of $\mathbf{X}$:

$$\mathbf{V}_p = [\mathbf{v}_1, \ldots, \mathbf{v}_p] \tag{2.14}$$

and $\mathbf{\Sigma}_p \in \mathbb{R}^{p \times p}$ is the square diagonal matrix composed by p first singular values of the SVD of $\mathbf{X}$:

$$\mathbf{\Sigma}_p = \text{diag}(\lambda_1, \ldots, \lambda_p) \tag{2.15}$$

The matrix $\mathbf{X}_p$ thus obtained is commonly called « truncation » of rank p of the SVD of the matrix $\mathbf{X}$.

Another way of obtaining the matrix $\mathbf{X}_p$ of rank p is to project the column vectors and row vectors of the matrix $\mathbf{X}$ onto the subspaces vector $E_p^{(1)} \subset E^{(1)}$ and $E_p^{(2)} \subset E^{(2)}$ generated by the p left singular vectors and the p right singular vectors associated to the p largest eigenvalues of the matrix $\mathbf{X}$:

$$\mathbf{X}_p = \mathbf{P}^{\perp}_{\mathbf{U}_p}\mathbf{X}\mathbf{P}^{\perp}_{\mathbf{V}_p} \tag{2.16}$$

in which:
The projector $\mathbf{P}^{\perp}_{\mathbf{U}_p} \in \mathbb{R}^{I_1 \times I_1}$ is defined by:

$$\mathbf{P}^{\perp}_{\mathbf{U}_p} = \mathbf{U}_p\mathbf{U}_p^T \tag{2.17}$$

The projector $\mathbf{P}^{\perp}_{\mathbf{V}_p} \in \mathbb{R}^{I_2 \times I_2}$ is defined by:

$$\mathbf{P}^{\perp}_{\mathbf{V}_p} = \mathbf{V}_p\mathbf{V}_p^T \tag{2.18}$$

Further in the manuscript, we will see what is the relationship between singular value decomposition, and its generalization to multidimensional data, that is, Tucker3 decomposition and Parafac decomposition.

2.6 Application examples on simple matrices and images

This section provides examples of matrix rank estimation. It shows how to translate the rank of a matrix in terms of images and describes the effect of truncation on pure images or noisy images.

2.6.1 Example of lower triangular matrix nd singular value decomposition

We deal here with a lower triangular matrix

$$\mathbf{X} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \tag{2.19}$$

This matrix can easily be expressed through:

$$\mathbf{X} = \begin{pmatrix} & & \\ \mathbf{u}_1 & \mathbf{u}_2 & \\ & & \end{pmatrix} \begin{pmatrix} \lambda_1 & \\ & \lambda_2 \end{pmatrix} \begin{pmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \end{pmatrix} \tag{2.20}$$

with:

$\diamond$ $\lambda_1 = \frac{\sqrt{5}+1}{2}$, $\lambda_2 = \frac{\sqrt{5}-1}{2}$,

$\diamond$ $\mathbf{u}_1 = [-1,\ -\lambda_1]^T$, $\mathbf{u}_2 = [-\lambda_1,\ 1]^T$,

$\diamond$ $\mathbf{v}_1 = [-\lambda_1,\ -1]^T$, $\mathbf{v}_2 = [-1,\ \lambda_1]^T$,

all divided by $\sqrt{1+\lambda_1^2}$.

A numerical application yields $\lambda_1 = 1.6180$, $\lambda_2 = 0.6180$,

$\mathbf{u}_1 = [-0.5257,\ -0.8507]^T$, $\mathbf{u}_2 = [-0.8507,\ 0.5257]^T$,

$\mathbf{v}_1 = [-0.8507,\ -0.5257]^T$ and $\mathbf{v}_2 = [-0.5257,\ 0.8507]^T$.

Which give:

$$\mathbf{U} = \begin{pmatrix} -0.5257 & -0.8507 \\ -0.8507 & 0.5257 \end{pmatrix} \tag{2.21}$$

$$\mathbf{V} = \begin{pmatrix} -0.8507 & -0.5257 \\ -0.5257 & 0.8507 \end{pmatrix} \tag{2.22}$$

$$\lambda = \begin{pmatrix} 1.6180 & 0 \\ 0 & 0.6180 \end{pmatrix} \tag{2.23}$$

We can also write:

$$\mathbf{X} \begin{pmatrix} \mathbf{v}_1 & \mathbf{v}_2 \end{pmatrix} = \begin{pmatrix} \lambda_1 \mathbf{u}_1 & \lambda_2 \mathbf{u}_2 \end{pmatrix} \tag{2.24}$$

or:

$$\mathbf{X} = \lambda_1 \mathbf{u}_1 \mathbf{v}_1^T + \lambda_2 \mathbf{u}_2 \mathbf{v}_2^T \tag{2.25}$$

In Eq. (2.25) $\mathbf{u}_1$, $\mathbf{u}_2$ are the left singular vectors and $\mathbf{v}_1$, $\mathbf{v}_2$ are the right singular vectors of $\mathbf{X}$. Hence, the best rank-1 approximation of $\mathbf{X}$ in a least squares sense is:

$$\mathbf{X}_1 = \lambda_1 \mathbf{u}_1 \mathbf{v}_1^T \tag{2.26}$$

or equivalently:

$$\mathbf{X}_1 = \mathbf{u}_1 \mathbf{u}_1^T \mathbf{X} \mathbf{v}_1 \mathbf{v}_1^T. \tag{2.27}$$

On can notice that the scaling term λ_1 in Eq. (2.26) represents the 'magnitude' of the first term of the singular value decomposition, retained in the truncation. in Eq. (2.27), this saling term is not present, but this piece of information is contained in matrix $\mathbf{X}$. From the Eckart-Young theorem presented in section 2.5.3, we can assert that we obtained in Eqs. (2.26) and (2.27) the best rank-one approximation of $\mathbf{X}$ in least squares sense. A numerical application yields, as best rank-1 estimate in least squares sense of $\mathbf{X}$ is:

$$\mathbf{X}_1 = \begin{pmatrix} 0.7236 & 0.4472 \\ 1.1708 & 0.7236 \end{pmatrix} \tag{2.28}$$

In this case, $\|\mathbf{X} - \mathbf{X}_1\|^2 = 0.6180$.

2.6.1.1 'Flag' matrices

We deal here with the French and German flags

(French flag) (German flag)

Figure 2.7: Example of used flags.

Here is a matrix which reminds the French flag:

$$\mathbf{F} = \begin{pmatrix} a & a & b & b & c & c \\ a & a & b & b & c & c \\ a & a & b & b & c & c \\ a & a & b & b & c & c \\ a & a & b & b & c & c \\ a & a & b & b & c & c \end{pmatrix} \tag{2.29}$$

Here is a matrix which reminds the German flag:

$$\mathbf{G} = \begin{pmatrix} a & a & a & a & a & a \\ a & a & a & a & a & a \\ b & b & b & b & b & b \\ b & b & b & b & b & b \\ c & c & c & c & c & c \\ c & c & c & c & c & c \end{pmatrix} \tag{2.30}$$

We notice that $\mathbf{G} = \mathbf{F}^T$.

The matrix of Eq. (2.29) is easily expressed as: $\mathbf{F} = \mathbf{u}\mathbf{v}^T$ with:

$$\mathbf{u} = (\ 1\ 1\ 1\ 1\ 1\ 1\ )^T \tag{2.31}$$

$$\mathbf{v} = (\ a\ a\ b\ b\ c\ c\ )^T \tag{2.32}$$

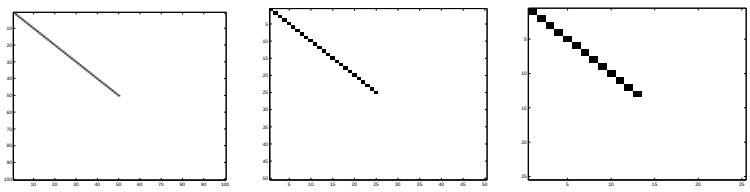
The matrix of Eq. (2.30) is easily expressed as: $\mathbf{G} = \mathbf{F}^T = \mathbf{v}\mathbf{u}^T$. We show thereby that $\mathbf{F}$ and $\mathbf{G}$ are rank-one matrices. However, in this case $\mathbf{u}$ and $\mathbf{v}$ may not be the singular vectors of $\mathbf{F}$ and $\mathbf{G}$. Let's consider for instance a numerical application with $a = 1$, $b = 2$, $c = 3$. The unique non-zero singular value of the matrix in Eq. (2.29) or in Eq. (2.30) is $\lambda = 12.9615 \neq 1$.
Because $\mathbf{F} = 1\ \mathbf{u}\ \mathbf{v}^T \neq \lambda\ \mathbf{u}\ \mathbf{v}^T$, $\mathbf{u}$ and $\mathbf{v}$ of Eqs. (2.31) and (2.32) are not the singular vectors of $\mathbf{F}$, only proportional to them.
The singular vectors associated with $\lambda = 12.9615$ are respectively -0.4082 $\mathbf{u}$ and -0.1890 $\mathbf{v}$.
Notice that $\frac{1}{(-0.4082)(-0.1890)} = 12.9615 = \lambda$.

If we consider the summation $\mathbf{X} = \mathbf{F} + \mathbf{G}$, we can show that $\mathbf{X}$ is a rank-2 matrix. With the same numerical application, we get that the singular values of $\mathbf{X}$ are 1.9827 and 0.0173. As $\mathbf{X} = 1\ \mathbf{u}\ \mathbf{v}^T + 1\ \mathbf{v}\ \mathbf{u}^T$, we can assert that $\mathbf{u}, \mathbf{v}^T$ and $\mathbf{v}, \mathbf{u}^T$ cannot be pairs of singular vectors of $\mathbf{X}$.

2.6.2 Influence of subsampling on matrix rank

We denote by S the subsampling factor applied to both rows and columns, and study the influence of image subsampling on the rank value, for various types of images.

2.6.2.1 Exemplification with straight line segments



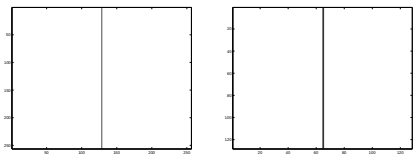
(a) (b) (c)

Figure 2.8: Diagonal segment, for values of S: (a) 1; (b) 2; (c) 4.

Fig. 2.8 illustrates the effect of subsampling on the matrix rank on an image containing a straight line segment, with offset 0 and orientation $45°$. This segment crosses the first half of the image, so that only the first half of the columns contain 1 pixel with value 1 (on the segment). All other pixels are 0-valued. The original image, displayed on Fig. 2.8 (a) has size 100×100, and rank 50. The first subsampled image, by a factor 2 along rows and columns, is displayed on Fig. 2.8 (b). It has

size 50×50, and rank 25. The second subsampled image, by a factor 4 along rows and columns, is displayed on Fig. 2.8 (c). It has size 25×25, and rank 13.

In the example of Fig. 2.8, the rank value is obtained from the original best resolved image by dividing its original rank by the subsampling factor, and rounding it to the nearest integer value. Therefore, in this case the rank of the original image with maximum resolution can be deduced from the rank of the subsampled image through a multiplication by the subsampling factor.

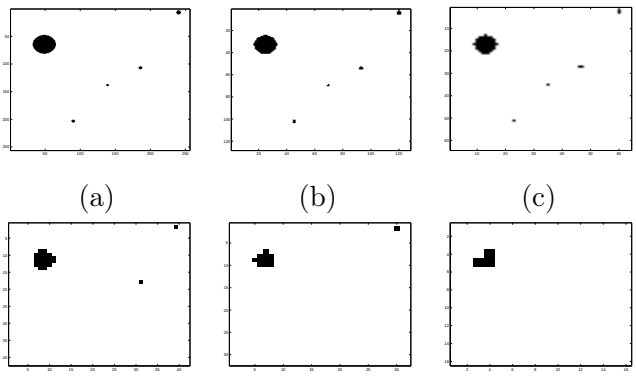


(a) (b)

Figure 2.9: Vertical segment, for values of S: (a) 1; (b) 2.

In the example of Fig. 2.9, the original image, displayed on Fig. 2.8 (a) has size 256×256, and rank 1: it contains a vertical line crossing the image. The image presented in Fig. 2.9 (b) results from a subsampling by a factor 2 along rows and columns. Dividing the original rank by 2 and ceiling to the nearest integer yields 1/2 and eventually 1: the argument presented above is still valid. It is no longer valid when the subsampling factor is 4 or more because the vertical line is removed from the image through subsampling.

2.6.2.2 Exemplification with disks



(a) (b) (c)

(d) (e) (f)

Figure 2.10: Disks, for values of S: (a) 1; (b) 2; (c) 4; (d) 6; (e) 8; (f) 16 .

In Table 2.1 we present the size and rank values for the images in Fig. 2.10.

Image Properties \ S	1	2	4	6	8	16
Size	256×256	128×128	64×64	42×42	32×32	16×16
Rank	21	14	9	5	4	2
Nb objects	5	5	5	3	2	1

Table 2.1: 'Disks' image: Size and ranks values for various instances of subsampling factor.

2.7 Tensor model

Tensor data modeling and tensor analysis have been improved and used in several application fields such as quantum physics, economy, chemometrics [126], psychology, data analysis, etc. Nevertheless, only recent studies focus their interest on tensor methods in signal processing applications. Tensor formulation in signal processing has received great attention since the recent development of multicomponent sensors, especially in imagery (color or multispectral images, video, etc.) and seismic fields (antenna of sensors recording waves with polarization properties). Indeed, the digital data obtained from these sensors are fundamentally higher order tensor objects, that is, multiway arrays whose elements are accessed via more than two indexes. Each index is associated with a dimension of the tensor generally called "nth-mode" [25, 26, 65, 67].

Tensor models were adopted in chemometrics [126], for DS-CDMA system characterization [127], and to perform facial expression classification, by multilinear independent component analysis [140].

Tensor models are also in adequation with a medical imaging modality referred to as magnetic resonance imaging, and the need to process and visualize such images [27]. Tensor data generalize the classical vector and matrix data to entities with more than two dimensions [106, 126]. In signal processing, there was a recent development of multicomponent sensors, especially in imagery (color or multispectral images, video, etc.) and seismic fields (antenna of sensors recording waves with polarization properties). The digital data obtained from these sensors are fundamentally multiway arrays, which are called, generally, higher order tensor objects, or tensors. Each multiway array entry corresponds to any quantity. The elements of a multiway array are accessed via several indexes. Each index is associated with a dimension of the tensor generally called "nth-mode" [25, 26, 66, 67].

2.7.1 Definition of a tensor

A N-th order tensor is a N-dimensional array, $\mathcal{X} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$, in which $\mathbb{R}$ indicates the real manifold, and N is the number of dimensions. The elements in tensor $\mathcal{X}$ can be expressed as $x_{i_1 \ldots i_N}$, with $i_1 = 1, \ldots, I_1; \cdots;\ i_N = 1, \ldots, I_N$. The n-th dimension of tensor $\mathcal{X}$ is called mode-n.

Tensors are generalizations of matrices, for instance, a zero-order tensor $x \in \mathbb{R}$ is a scalar, a first-order tensor $\mathbf{x} \in \mathbb{R}^{I_1}$ a vector and a second-order tensor $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2}$ a matrix. The relationship of scalar, vector, matrix, and tensor is illustrated in Fig. 2.11.

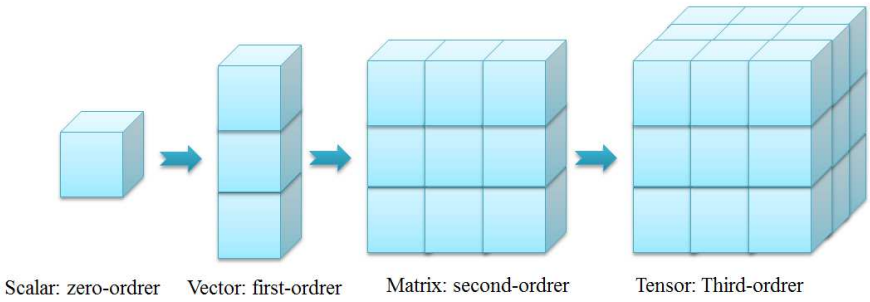


Figure 2.11: The relationship of scalar, vector, matrix and tensor

2.7.2 Tensor unfolding

Unfolding [62] is also known as matricization [80] [57] or flattening [77]. It can convert a tensor into its matrix form for the convenient of using the matrix-based data analyzing methods. A tensor can be unfolded in different ways according to the mode in which the unfolding is performed. The unfolding in mode-n is called mode-n unfolding.

Fig. 2.12 illustrates the principle of unfolding graphically for a third-order tensor (for example HSI) and shows different possible unfolding:

Let $\mathbf{X}_n \in \mathbb{R}^{I_n \times M_n}$ denote the mode-n unfolding matrix of a tensor $\mathbf{X} \in \mathbb{R}^{I_n \times I_1 I_N}$, where $M_n = I_1 \ldots I_{n-1} I_{n+1} \ldots I_N$.

The columns of $\mathbf{X}_n$ are the I_n-dimensional vectors obtained from $\mathcal{X}$ by varying index i_n while keeping the other indices fixed.

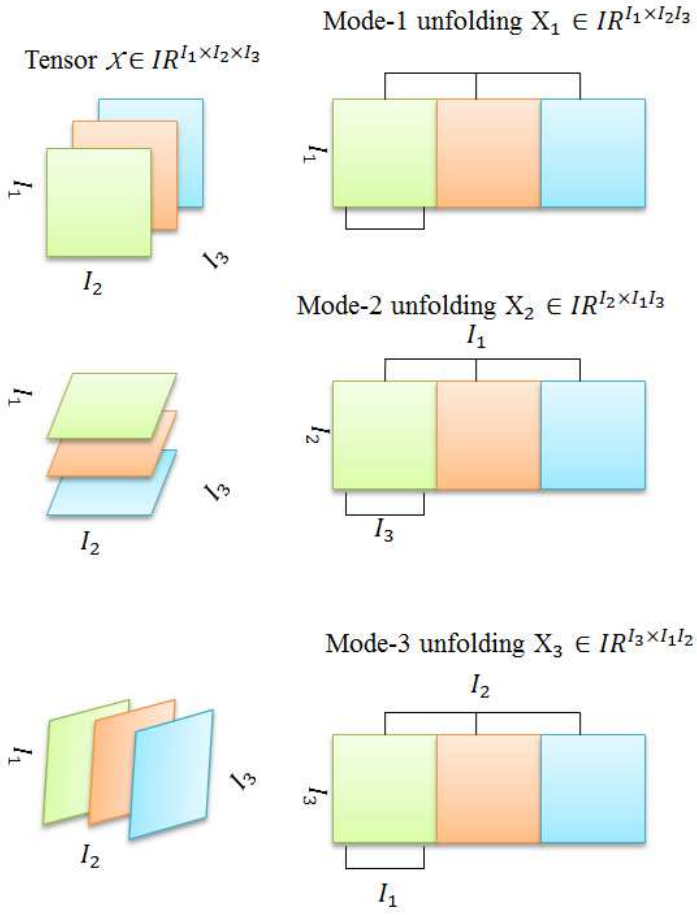


Figure 2.12: Illustration of tensor unfolding

2.7.3 Some tools for tensor processing

Mode-n tensor product $\times_n$

The mode-n product is defined as the product between a data tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$ and a matrix $\mathbf{B} \in \mathbb{R}^{J \times I_n}$ in mode n. This mode-n product is denoted by $\mathcal{C} = \mathcal{X} \times_n \mathbf{B}$, whose entries are given by

$$c_{i_1 \ldots i_{n-1} j i_{n+1} \ldots i_N} \triangleq \sum_{i_n=1}^{I_n} x_{i_1 \ldots i_{n-1} i_n i_{n+1} \ldots i_N} b_{j i_n}, \tag{2.33}$$

Outer product $\circ$

The outer product is defined as the product between two tensors: $\mathcal{X} \in \mathbb{R}^{I_1 \times I_N}$ and $\mathcal{B} \in \mathbb{R}^{J_1 \times J_N}$. The outer product is denoted by $\mathcal{C} = \mathcal{X} \circ \mathcal{B}$, whose entries are given by

$$c_{i_1 \ldots i_N j_1 \ldots j_N} \triangleq x_{i_1 \ldots i_N} b_{j_1 \ldots j_N}, \tag{2.34}$$

where $\mathcal{C} \in \mathbb{R}^{I_1 \times I_N \times J_1 \times J_N}$.

Kronecker product $\otimes$

The Kronecker product is defined as the product between two matrices: $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2}$ and $\mathbf{B} \in \mathbb{R}^{J_1 \times J_2}$. The Kronecker product is denoted by $\mathbf{C} = \mathbf{X} \otimes \mathbf{B}$, which can be calculated by:

$$\mathbf{C} = \begin{bmatrix} x_{11}\mathbf{B} & \cdots & x_{1I_2}\mathbf{B} \\ \vdots & \vdots & \vdots \\ x_{I_1 1}\mathbf{B} & \cdots & x_{I_1 I_2}\mathbf{B} \end{bmatrix}, \tag{2.35}$$

where $\mathbf{C} \in \mathbb{R}^{I_1 J_1 \times I_2 J_2}$.

Khatri-Rao product $\odot$

The Khatri-Rao product is defined as the product between two matrices: $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2}$ and $\mathbf{B} \in \mathbb{R}^{J_1 \times I_2}$. The Khatri-Rao product is denoted by $\mathbf{C} = \mathbf{X} \odot \mathbf{B}$, which can be calculated by:

$$\mathbf{C} = \begin{bmatrix} \mathbf{x}_1 \otimes \mathbf{b}_1 & \cdots & \mathbf{x}_{I_2} \otimes \mathbf{b}_{I_2} \end{bmatrix}, \tag{2.36}$$

where $\mathbf{C} \in \mathbb{R}^{I_1 J_1 \times I_2}$, and $\mathbf{x}_n$ and $\mathbf{b}_n$ are the n-th column vectors of $\mathbf{X}$ and $\mathbf{B}$ respectively.

Frobenius norm $\|\cdot\|$

The Frobenius norm of tensor $\mathcal{X}$ is denoted by $\|X\|$ in this thesis. It can be calculated by:

$$\|X\|^2 = \sum_{i_1=1}^{I_1} \cdots \sum_{i_N=1}^{I_N} x_{i_1 \ldots i_N}^2. \tag{2.37}$$

The quadratic distance between two tensors $\mathcal{X}$ and $\mathcal{B}$ can be calculated by $\|\mathcal{X} - \mathcal{B}\|^2$.

2.7.4 Tensor Rank

In the case of order two, the rank of a matrix have a fundamental role in its SVD, in its canonical decomposition and in its approximation of lower rank. In this section we recall the different definitions of the rank of a tensor of order greater than two [61, 75]. These different definitions make it possible to generalize the matrix SVD to the tensors.

Usual rank

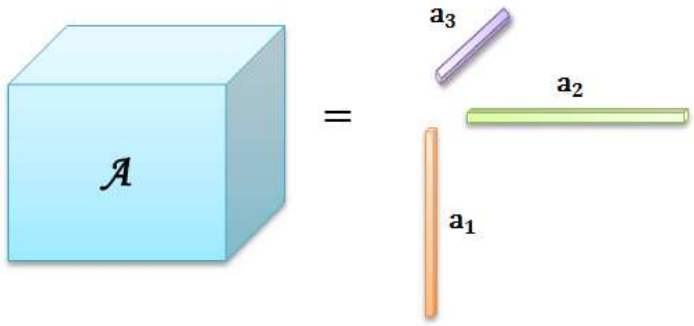


Figure 2.13: A rank-1 third-order tensor equals to the outer product of three vectors

Rank-one tensor let an N-th order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$ has rank 1 when it equals the outer product of N vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N$, which can be expressed as [59]:

$$\mathcal{X} = \mathbf{x}_1 \circ \ldots \circ \mathbf{x}_N. \tag{2.38}$$

A rank-1 third-order tensor is presented in 2.13. Rank-1 tensor is the fundamental concept for defining the rank of a tensor.

Tensor of rank K The concept of rank of a matrix can be extended to higher-order tensors [73, 98]. The definition of a tensor of rank K generalizes the matrix definition. Indeed, $\mathcal{X}$ is a tensor of rank K, where K is the minimum number of rank-1 tensors which, by addition, gives $\mathcal{X}$. We denote the rank of the tensor $\mathcal{X}$ by:

$$Rank(\mathcal{X}) = K \tag{2.39}$$

Therefore, $\mathcal{X} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$ is a rank K tensor if, $\forall n = 1 \ldots N$, There exists K n-modal normalized vectors $u_k^{(n)} \in E^{(n)}$, and K scalars λ_k, k=1 ..., K such as :

$$\mathcal{X} = \sum_{k=1}^{K} \lambda_k u_k^{(1)} \circ \ldots u_k^{(N)} = \sum_{k=1}^{K} \lambda_k \mathcal{X}_k \tag{2.40}$$

The definition (2.40)of the rank of a tensor permits to extend, the canonical decomposition of a matrix, to the tensors. Several studies known as «Parallel Factor Analysis» (Parafac) [43] and «Canonical Decomposition» (CANDECOMP) [6] were carried out on the determination of the rank of a tensor and its decomposition into a minimal sum of rank-1 tensors. These two methods are known as Parafac / CANDECOMP tensor decomposition.

Remark 1: Note that this definition of rank don't impose constraint on the orthogonality of the rank-1 tensors as is imposed naturally in the SVD matrix.

Remark 2: The rank of a matrix $\mathbf{A} \in \mathbb{R}^{I_1 \times I_2}$ is always less than or equal to $min(I_1, I_2)$:

$$Rank(\mathbf{A}) \leq min(I_1, I_2) \tag{2.41}$$

This property is not verified in the case of a tensor [25, 61, 73]. Indeed, the rank of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$ can be such that:

$$Rank(\mathcal{X}) > min(I_1 \ldots I_2) \tag{2.42}$$

Mode-n rank of a tensor The term mode-n rank K_n of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$, denoted by $K_n = \text{rank}_n(\mathcal{X})$, is the dimension of the column space of the mode-n unfolding matrix $\mathbf{X}_n$ [73], *i.e.*,

$$K_n = \text{rank}_n(\mathcal{X}) = \text{rank}(\mathbf{X}_n). \tag{2.43}$$

Like the term rank of a tensor, the term mode-n rank is another way to extend the notion of the matrix rank to the tensor case. The mode-n rank actually is a rank of a matrix, therefore it can be analyzed by means of matrix techniques. Nevertheless, for a matrix $\mathbf{X}$, the ranks of its column space and row space are the same, rank($\mathbf{X}$) = rank($\mathbf{X}^T$), but for the tensor $\mathcal{X}$, the different mode-n ranks are not necessarily the same, which means that the mode-n rank of tensor $\mathcal{X}$ is not a unique value but a set of N values $\{K_1, \ldots, K_N\}$. Since the rank and the mode-n rank both are the extensions of the matrix rank concept, there is a relationship between them, which can be described as [73]:

$$K_n \leq K. \tag{2.44}$$

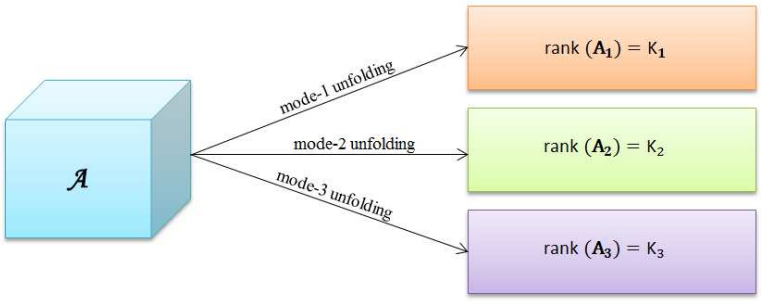


Figure 2.14: A rank-(K_1, K_2, K_3) third-order tensor means that the rank of its mode-n unfolding matrix $\mathbf{A}_n$ is K_n, $n = 1, 2, 3$

It's often useful to be able to approximate a tensor using decompositions. This makes it possible to generalize the matrix decompositions to the higher-order tensors in order to solve filtering problems for example. The main idea is to find a good approximation of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$ by another tensor $\hat{\mathcal{A}}$ to minimize a specific criterion. In the matrix case, the Eckart-Young theorem [33] indicates that the best approximation of Low-rank-K of a matrix $\mathbf{A}$ is obtained by truncating SVD itself,in the sense of the least squares. That is, by keeping only the K first singular vectors associated with the K largest singular values of $\mathbf{A}$, we obtain an optimal approximation of $\mathbf{A}$. In the case of tensors of order higher than 2, this theorem must be extended. Studies have been carried out to overcome the problem of optimal decomposition.

2.8 Tensor decomposition

2.8.1 Tucker3 Model

Tucker3 model was first introduced by TUCKER [136] and refined in [137, 138]. Tucker3 decomposition was originally developed to perform a decomposition for a tensor to find singular values of all modes which are independent from each other, so it was also named as higher-order SVD (HOSVD) [82]. This model is also one source of the solution of lower rank approximation of a tensor [76]. Kroonenberg and Leeuw [68] suggested an alternating least squares algorithm (ALS) to solve the minimization of the approximation error.

2.8.1.1 Tucker3 decomposition

The Tucker3 decomposition [62] can be seen as a form of higher-order principal component analysis, which decomposes a tensor into a core tensor multiplied by a matrix along each mode. Thus, $\mathcal{X} \in \mathbb{R}^{I_1 \times \ldots \times I_N}$ can be expressed as:

$$\begin{aligned} \hat{\mathcal{X}} &= \mathcal{C} \times_1 \mathbf{X}^{(1)} \times_2 \mathbf{X}^{(2)} \times_3 \cdots \times_N \mathbf{X}^{(N)} \\ &= \sum_{k_1=1}^{K_1} \sum_{k_2=1}^{K_2} \cdots \sum_{k_N=1}^{K_N} c_{k_1 k_2 \cdots k_N} \boldsymbol{x}^{k_1} \circ \boldsymbol{x}^{k_2} \circ \cdots \circ \boldsymbol{x}^{k_N} \end{aligned} \tag{2.45}$$

where, the factor matrix $\mathbf{X}^{(n)} \in \mathbb{R}^{I_n \times K_n}$ $(n = 1, 2, \cdots, N)$ is usually orthogonal and can be thought of as the principal components in each mode. It holds the K_n eigenvectors associated with the K_n largest eigenvalues of the unfolding matrix $\mathbf{X}_n$; $\mathcal{C} \in \mathbb{R}^{K_1 \times \cdots \times K_N}$ is the core tensor whose entries show the level of interaction between the different components. Therefore, given a real-valued third-order tensor $\mathcal{X}$, the Tucker3 decomposition provides the lower rank-(K_1, K_2, K_3) multi-mode data $\hat{\mathcal{X}}$ by minimizing $\mathbf{E}[\|\mathcal{X} - \hat{\mathcal{X}}\|^2]$ [118]. An example of Tucker3 decomposition of a three-way array is displayed in Fig. 2.15.

2.8.1.2 Multiple ranks of Tucker3 decomposition

The n-rank of tensor $\mathcal{X}$, denoted by $\mathrm{rank}_n(\mathcal{X})$, is the column rank of the unfolding matrix $\mathbf{X}_n$ of $\mathcal{X}$, that is to say that the n-rank is the dimension of the vector space spanned by the n-mode vectors [62, 74]. By defining $K_n = \mathrm{rank}_n(\mathcal{X})$, $\mathcal{X}$ could be said as a $\mathrm{rank} - (K_1, K_2, \cdots, K_N)$ tensor with $K_n \leq I_n$ for all $n = 1, \cdots N$. Therefore, an approximation of a tensor $\mathcal{X}$ could be calculated by Tucker3 decomposition of $\mathrm{rank} - (K_1, K_2, \cdots, K_N)$ with $K_n \leq \mathrm{rank}_n(\mathcal{X})$.

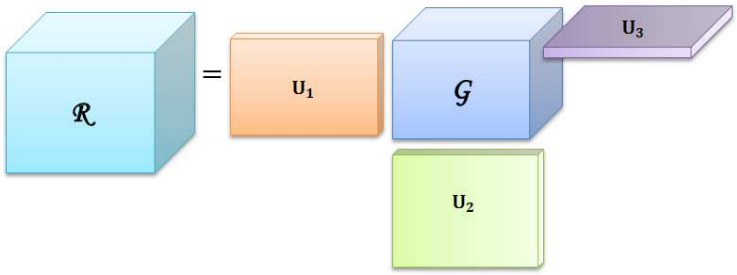


Figure 2.15: Tucker3 decomposition of a three-way array

But Tucker3 decomposition is not unique because the core $\mathcal{C}$ can be modified without affecting the fit as long as the factor matrices are changed inversely [62], such as: making

$$\mathcal{C} \leftarrow \mathcal{C} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \cdots \times_N \mathbf{W}_N$$

and

$$\begin{aligned} \mathbf{X}^{(1)} &\leftarrow \mathbf{X}^{(1)} \times \mathbf{W}_1^{-1} \\ &\vdots \\ \mathbf{X}^{(N)} &\leftarrow \mathbf{X}^{(N)} \times \mathbf{W}_N^{-1} \end{aligned}$$

where $\mathbf{W}_1 \in \mathbb{R}^{K_1 \times K_1}, \cdots, \mathbf{W}_N \in \mathbb{R}^{K_N \times K_N}$, the result is still the same. Since the synchronous estimation of the values of all $K_1, K_2, \cdots, K_N$ is hard, one solution consists in estimating K_N with fixed $K_1, \cdots, K_{N-1}$ [120] but it is time-consuming due to the multiple combinations of different K_n. Another method is Akaike information criterion (AIC) to estimate each K_n value [78, 89, 117]. However the assumption that the rank values are independent one from the others may not always be true. The optimal estimation of the rank values is the purpose of the Chapter 3 of this manuscript.

2.8.1.3 Application of Tucker3

Tucker3 decomposition finds its utility in chemical analysis [44]. Many examples of applications in psychometrics were provided by Kiers and Van Mechelen [58]. De Lathauwer *et al.* used Tucker3 decomposition in signal processing [24]. Muti and Bourennane extended Tucker3 decomposition to multidimensional Wiener filter which was applied to image processing [105]. Vasilescu and Terzopoulos pioneered the use of Tucker decompositions in computer vision with Tensor Faces [141].

Khoromskij and Khoromskaia [56] applied the Tucker decomposition to approximations of classical potentials while adapting optimized algorithms. J.-T. Sun *et al*. [131] used Tucker to analyze web site click-through data. Liu *et al*. [88] applied Tucker to create a tensor space model, analogous to the well-known vector space model in text analysis. J. Sun *et al*. [130] explained how to dynamically update a Tucker approximation, with applications ranging from text analysis to environmental and network modelling. Very recently, through a prewhitening approach, Tucker3 decomposition was extended to reduce non-white noise in HSIs [91].

2.8.2 Parafac model

The idea of the polyadic form of a tensor was proposed by Hitchcock in 1927 [46, 47], *i.e.*, expressing a tensor as the sum of a finite number of rank-one tensors. Later Cattell proposed ideas for parallel proportional analysis and the idea of multiple axes for analysis (circumstances, objects, and features) [7,8]. The concept finally became popular after its third introduction,to the psychometrics community, in the form of CANDECOMP (canonical decomposition) by Carroll and Chang [5] and Parafac by Harshman [42]. Kiers refers to the CANDECOMP/Parafac decomposition as CP in [57]. Mocks [101] independently discovered Parafac in the context of brain imaging and called it the Topographic Components Model.

2.8.3 Parafac decomposition

The Parafac decomposition factorizes a tensor into a sum of rank-1 tensors [62]. For instance, a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ can be expressed as

$$\begin{aligned} \mathcal{X} \approx \hat{\mathcal{X}} &= \sum_{k=1}^{K} \mathcal{X}_k \\ &= \sum_{k=1}^{K} \tilde{\boldsymbol{x}}_k^{(1)} \circ \tilde{\boldsymbol{x}}_k^{(2)} \circ \cdots \circ \tilde{\boldsymbol{x}}_k^{(N)} \end{aligned} \tag{2.46}$$

where $\hat{\mathcal{X}}$ is the rank-K approximation of $\mathcal{X}$, $\mathcal{X}_k \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is rank-1 tensor. We define the factor λ_k as:

$$\lambda_k = \|\tilde{\boldsymbol{x}}_k^{(1)}\| \cdot \|\tilde{\boldsymbol{x}}_k^{(2)}\| \cdot \cdots \cdot \|\tilde{\boldsymbol{x}}_k^{(N)}\|, \quad \text{for } k = 1, 2, \cdots, K$$

and

$$\tilde{\boldsymbol{x}}_k^{(n)} = [\tilde{x}_{1k}^{(n)}, \tilde{x}_{2k}^{(n)}, \cdots, \tilde{x}_{i_n k}^{(n)}]^T, \quad \text{for } n = 1, 2, \cdots, N$$

By normalizing, the unit-norm vectors of the n-mode space of $\mathcal{X}$ could be defined as:

$$\boldsymbol{x}_k^{(n)} = \frac{\tilde{\boldsymbol{x}}_k^{(n)}}{\|\tilde{\boldsymbol{x}}_k^{(n)}\|}.$$

Then Eq. (2.46) becomes:

$$\begin{aligned} \mathcal{X} \approx \hat{\mathcal{X}} &= \sum_{k=1}^{K} \mathcal{X}_k \\ &= \sum_{k=1}^{K} \lambda_k \boldsymbol{x}_k^{(1)} \circ \boldsymbol{x}_k^{(2)} \circ \cdots \circ \boldsymbol{x}_k^{(N)}. \end{aligned} \tag{2.47}$$

Elementwise, Eq. (2.47) is written as

$$\hat{x}_{i_1 i_2 \cdots i_N} = \sum_{k=1}^{K} \lambda_k x_{i_1 k}^{(1)} x_{i_2 k}^{(2)} \cdots x_{i_N k}^{(N)}, \qquad \text{for} \quad i_n = 1, 2, \cdots, I_n \quad \text{and} \quad n = 1, 2, \cdots N \tag{2.48}$$

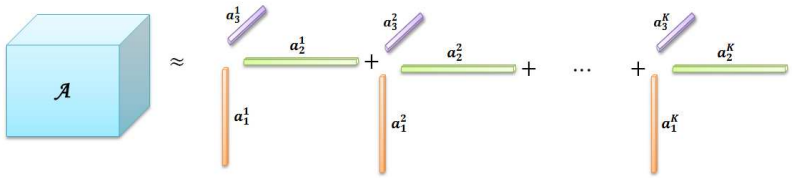


Figure 2.16: Parafac decomposition of a three-way array.

Fig. 2.16 illustrates an example of Parafac decomposition of a three-way array. Parafac decomposition is used to compute $\hat{\mathcal{X}}$ with K components that approximate the best value of $\mathcal{X}$ by minimizing the square error

$$e = \|\mathcal{X} - \hat{\mathcal{X}}\|^2.$$

Using Khatri-Rao product, the n-mode unfolding matrix of $\mathcal{X}$ is given by [62]:

$$\hat{\mathbf{X}}_n = \mathbf{X}^{(n)} \boldsymbol{\Lambda} (\mathbf{X}^{(N)} \odot \cdots \odot \mathbf{X}^{(n+1)} \mathbf{X}^{(n-1)} \odot \cdots \odot \mathbf{X}^{(1)})^T \tag{2.49}$$

where $\mathbf{X}^{(n)} = [\boldsymbol{x}_1^{(n)}, \ldots, \boldsymbol{x}_K^{(n)}]$, $n = 1, 2, \cdots, N$ and

$$\boldsymbol{\Lambda} = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_K \end{bmatrix}.$$

2.8.4 An example of Tucker3 decomposition

This section provides examples of tensor rank estimation. It shows how to translate the rank of a tensor in terms of images and describes the effect of truncation on pure images or noisy images.

We consider a $2 \times 2 \times 2$ tensor where each band is as follows:

$$\mathcal{X}(:,:,1) = \mathcal{X}(:,:,2) = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \tag{2.50}$$

The properties of tensor $\mathcal{X}$ are as follows: We notice from Eq. (2.50) that the two bans of $\mathcal{X}$ are equal. From this, we infer easily that, for the first and second modes ($n = 1$ and $n = 2$), the rank values K_1 and K_2 in the Tucker3 decomposition are equal to 2, the matrix rank of the first band. A possible way to unfold $\mathcal{X}$ along the third mode yields:

$$\mathbf{X}_3 = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix} \tag{2.51}$$

Obviously, from Eq. (2.51) for the third mode, the rank value K_3 is 1.

Matrices $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, $\mathbf{U}^{(3)}$ in the Tucker3 model of Eq. (2.45) are as follows: Also, $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$ are the same as the matrices $\mathbf{U}$, $\mathbf{V}$ in the example of subsection 2.6.1. A numerical application yields:

$$\mathbf{U}^{(1)} = \begin{pmatrix} -0.5257 & -0.8507 \\ -0.8507 & 0.5257 \end{pmatrix} \tag{2.52}$$

$$\mathbf{U}^{(2)} = \begin{pmatrix} -0.8507 & -0.5257 \\ -0.5257 & 0.8507 \end{pmatrix} \tag{2.53}$$

Matrix $\mathbf{U}^{(3)}$ is as follows:

$$\mathbf{U}^{(3)} = \begin{pmatrix} -0.7071 & -0.7071 \\ -0.7071 & 0.7071 \end{pmatrix} \tag{2.54}$$

The core tensor is obtained by inverting Eq. (2.45):

$$\mathcal{S} = \mathcal{X} \times_1 \mathbf{U}^{(1)T} \times_2 \mathbf{U}^{(2)T} \times_3 \mathbf{U}^{(3)T} \tag{2.55}$$

A numerical application yields:

$$\mathcal{S}(:,:,1) = \begin{pmatrix} 2.2882 & 0 \\ 0 & 0.8740 \end{pmatrix} \tag{2.56}$$

$$\mathcal{S}(:,:,2) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{2.57}$$

Firstly, we notice that the core tensor $\mathcal{S}$ exhibits the same n^{th}-mode rank values as tensor $\mathcal{X}$. Also, we notice that the core tensor $\mathcal{S}$ is not diagonal. Hence, the Parafac decomposition of $\mathcal{X}$ is not obvious. To illustrate the Parafac decomposition in a simple case, we prefer further to choose another example of multidimensional array.

Noisy case

We add a random tensor to the tensor defined above. We get a tensor $\mathcal{R}$:

$$\mathcal{R}(:,:,1) = \begin{pmatrix} 1.0219 & 0.0071 \\ 1.0033 & 1.0032 \end{pmatrix} \tag{2.58}$$

$$\mathcal{R}(:,:,2) = \begin{pmatrix} 1.0041 & 0.0014 \\ 1.0058 & 1.0164 \end{pmatrix} \tag{2.59}$$

Let's define the square error SE= $\|\mathcal{X} - \hat{\mathcal{X}}\|^2$ between raw tensor $\mathcal{X}$ and estimate $\hat{\mathcal{X}}$.

With ranks $\mathbf{K} = (2, 2, 1)$, the square error between truncated and noise-free tensor is 0.1805. With this triplet corresponding to the dimension of the signal subspace and other rank triplets, here are the estimation error values:

K	(2, 2, 2)	(2, 2, 1)	(2, 1, 2)	(2, 1, 1)
SE	0.0296	0.0247	0.8899	0.8880
K	(1, 2, 2)	(1, 2, 1)	(1, 1, 2)	(1, 1, 1)
SE	0.8893	0.8880	0.8882	0.8880

Notice that the error value for the triplet $\mathbf{K} = (2, 2, 2)$ is the error between the noise-free matrix and the impaired matrix. Obviously, the best rank triplet is $\mathbf{K} = (2, 2, 1)$, containing the dimension of the signal subspace for each mode. Only with this triplet, the error value between estimated matrix and noise-free matrix is lower than the error between impaired and noise-free matrices. With the best rank triplet, we get the following estimate:

$$\hat{\mathcal{X}}(:,:,1) = \begin{pmatrix} 1.0134 & 0.0043 \\ 1.0049 & 1.0102 \end{pmatrix} \tag{2.60}$$

$$\hat{\mathcal{X}}(:,:,2) = \begin{pmatrix} 1.0127 & 0.0043 \\ 1.0042 & 1.0094 \end{pmatrix} \tag{2.61}$$

This illustrates with a simple example the interest of retrieving the appropriate rank value for the truncation of a tensor decomposition, here the HOSVD.

2.8.5 Examples for the Parafac decomposition

In this study we compute the decomposition with a fixed rank. This is not the same as computing the decomposition with an elevated number of ranks, and then truncating the decomposition.

For the sake of the computational load, we prefer this solution because the maximum Parafac rank of a tensor is a number of elements inside, possibly very elevated.

2.8.5.1 Extension of the 'Flag' matrix to 3D data

We remind the matrix representing the the 'French flag':

$$\mathbf{F} = \begin{pmatrix} a & a & b & b & c & c \\ a & a & b & b & c & c \\ a & a & b & b & c & c \\ a & a & b & b & c & c \\ a & a & b & b & c & c \\ a & a & b & b & c & c \end{pmatrix} \tag{2.62}$$

Let $\mathcal{F}$ be a tensor such that:

$$\mathcal{F}(:,:,1) = \mathbf{F} \tag{2.63}$$

$$\mathcal{F}(:,:,2) = 2\mathbf{F} \tag{2.64}$$

and

$$\mathcal{F}(:,:,3) = 3\mathbf{F} \tag{2.65}$$

The tensor $\mathcal{F}$ is easily expressed as the outer product of three vectors: $\mathcal{F} = \mathbf{u} \circ \mathbf{v} \circ \mathbf{w}$ with:

$$\mathbf{u} = (\ 1\ 1\ 1\ 1\ 1\ 1\ )^T \tag{2.66}$$

$$\mathbf{v} = (\ a\ a\ b\ b\ c\ c\ )^T \tag{2.67}$$

$$\mathbf{w} = (\ 1\ 2\ 3\ )^T \tag{2.68}$$

Hence, the Parafac rank of $\mathcal{F}$ is 1.

We remind the matrix representing of the 'German flag' is denoted by $\mathbf{G}$.

Let $\mathcal{G}$ be a tensor such that:

$$\mathcal{G}(:,:,1) = \mathbf{G} \tag{2.69}$$

$$\mathcal{G}(:,:,2) = 2\mathbf{G} \tag{2.70}$$

and

$$\mathcal{G}(:,:,3) = 3\mathbf{G} \tag{2.71}$$

We notice that $\mathcal{G} = \mathbf{v} \circ \mathbf{u} \circ \mathbf{w}$.

Now let $\mathbf{H}$ be the following matrix:

$$\mathbf{H} = \begin{pmatrix} c & c & c & c & c & c \\ a & a & a & a & a & a \\ b & b & b & b & b & b \\ c & c & c & c & c & c \\ a & a & a & a & a & a \\ b & b & b & b & b & b \end{pmatrix} \tag{2.72}$$

Let $\mathcal{H}$ be a tensor such that:

$$\mathcal{H}(:,:,1) = \mathbf{H} \tag{2.73}$$

$$\mathcal{H}(:,:,2) = 2\mathbf{H} \tag{2.74}$$

and

$$\mathcal{H}(:,:,3) = 3\mathbf{H} \tag{2.75}$$

We notice that $\mathcal{H} = \mathbf{y} \circ \mathbf{u} \circ \mathbf{w}$, with

$$\mathbf{y} = (\ c\ a\ b\ c\ a\ b\ )^T \tag{2.76}$$

.

Let $\mathcal{X}$ be the sum of tensors $\mathcal{F}$, $\mathcal{G}$, $\mathcal{H}$: $\mathcal{X} = \mathcal{F} + \mathcal{G} + \mathcal{H}$. Obviously, because there is no tensor which is proportional to any other in the tensor triplet $(\mathcal{F}, \mathcal{G}, \mathcal{H})$, $\mathcal{X}$ follows the Parafac decomposition model of Eq. (2.48) with rank value $T = 3$.

Let $\mathcal{R}$ be a noisy tensor version of $\mathcal{X}$: $\mathcal{R} = \mathcal{X} + \mathcal{N}$ where $\mathcal{N}$ is a noise tensor. Let $\hat{\mathcal{X}}$ be the estimate of $\mathcal{X}$ obtained by truncation of the Parafac decomposition of $\mathcal{R}$ to rank K (see Eq. (3.9)).

Let's define: the square error $SE_{\mathcal{R},\mathcal{X}} = \|\mathcal{R} - \mathcal{X}\|^2$ between noisy tensor $\mathcal{R}$ and noise-free tensor $\mathcal{X}$; the square error $SE_{\mathcal{X},\hat{\mathcal{X}}} = \|\mathcal{X} - \hat{\mathcal{X}}\|^2$ between noise-free tensor $\mathcal{X}$ and estimate $\hat{\mathcal{X}}$; the square error $SE_{\mathcal{R},\hat{\mathcal{X}}} = \|\mathcal{R} - \hat{\mathcal{X}}\|^2$ between noisy tensor $\mathcal{R}$ and estimate $\hat{\mathcal{X}}$.

When the SNR level is such that $SE_{\mathcal{R},\mathcal{X}} = 0.00022969\ 10^{-5}$, we get the following values, depending on the rank for the truncation K:

For rank values 2 to 8: all error values are multiplied by 10^-5.

2.8.5.2 Noise free

In this case $\mathcal{R} = \mathcal{X}$.

K	2	3	4	5
$SE_{\mathcal{X},\hat{\mathcal{X}}}$	1574528.5345	$5.72\ 10^{-8}$	$4.3876\ 10^{-9}$	$9.9136\ 10^{-12}$
$SE_{\mathcal{R},\hat{\mathcal{X}}}$	1574528.5345	5.72^{-8}	$4.3876\ 10^{-9}$	$9.9136\ 10^{-12}$

K	6	7	8
$SE_{\mathcal{X},\hat{\mathcal{X}}}$	$1.1217\ 10^{-13}$	1.6093	2.11
$SE_{\mathcal{R},\hat{\mathcal{X}}}$	$1.1217\ 10^{-13}$	1.6093	2.11

In this case, a result obtained for K=6 for both errors.

2.8.5.3 Medium noise level

In this case $SE_{\mathcal{X},\mathcal{R}} = \|\mathcal{X} - \mathcal{R}\|^2 = 0.0071983\ 10^{-5}$

K	2	3	4	5
$SE_{\mathcal{X},\hat{\mathcal{X}}}$	1574528.5415	0.0087455	0.011689	0.011591
$SE_{\mathcal{R},\hat{\mathcal{X}}}$	1574528.5345	$3.041\ 10^{-8}$	$1.1258\ 10^{-8}$	$3.0989\ 10^{-11}$

K	6	7	8
$SE_{\mathcal{X},\hat{\mathcal{X}}}$	0.0086459	4.3138	2.5672
$SE_{\mathcal{R},\hat{\mathcal{X}}}$	$6.2363\ 10^{-9}$	4.1999	2.4625

For $SE_{\mathcal{X},\hat{\mathcal{X}}}$, the best value of rank is $K = 6$.

For $SE_{\mathcal{R},\hat{\mathcal{X}}}$, the best value of rank is $K = 5$.

2.8.5.4 High noise level

In this case, $SE_{\mathcal{X},\mathcal{R}} = 7502.212510^{-5}$

K	2	3	4	5
$SE_{\mathcal{X},\hat{\mathcal{X}}}$	1581301.2857	10083.552	6925.7019	9580.2287
$SE_{\mathcal{R},\hat{\mathcal{X}}}$	1569378.5109	$6.5162\ 10^{-8}$	$1.1346\ 10^{-8}$	$1.7301\ 10^{-13}$

K	6	7	8
$SE_{\mathcal{X},\hat{\mathcal{X}}}$	9600.9624	10441.7867	3613261.6752
$SE_{\mathcal{R},\hat{\mathcal{X}}}$	$1.2059\ 10^{-11}$	0.76749	3373016.5814

For $SE_{\mathcal{X},\hat{\mathcal{X}}}$, the best value of rank is $K = 4$.

For $SE_{\mathcal{R},\hat{\mathcal{X}}}$, the best value of rank is $K = 5$.

2.8.5.5 Very high noise level

In this case, $SE_{\mathcal{X},\mathcal{R}} = 59417.4057\ 10^{-5}$

K	2	3	4	5
$SE_{\mathcal{X},\hat{\mathcal{X}}}$	1629141.7144	43745.8148	54329.2849	52030.279
$SE_{\mathcal{R},\hat{\mathcal{X}}}$	1594723.701	$5.7681\ 10^{-8}$	$2.0652\ 10^{-9}$	$1.0281\ 10^{-13}$

K	6	7	8
$SE_{\mathcal{X},\hat{\mathcal{X}}}$	57337.7937	65011.2606	58920.5592
$SE_{\mathcal{R},\hat{\mathcal{X}}}$	$9.3314\ 10^{-14}$	24.493	0.098594

For $SE_{\mathcal{X},\hat{\mathcal{X}}}$, the best value of rank is $K = 3$.

For $SE_{\mathcal{R},\hat{\mathcal{X}}}$, the best value of rank is $K = 6$.

2.8.6 observation

The results obtained in the preceding examples note that the estimated rank differ for each case.

In this case, to solve optimization problems require enormous computational efforts, particularly (in the field of multispectral), which tend to fail as the problem size increases. To resolve this disadvantage, the nature is great source of inspiration for solving hard and complex problems in computer science. Indeed, the use of bio inspired stochastic optimization algorithms as computationally efficient alternatives to deterministic approach.

2.9 Conclusion

In this chapter, we have introduced the definition of a tensor and its corresponding algebra and properties. Particularly, we have discussed two tensor decompositions, Parafac and TUCKER, which are the cornerstones of the tensor-based denoising methods: MWF and Parafac filters.

MWF extends the classical Wiener filter to the multidimensional case by using TUCKER decomposition with minimizing the MSE between the desired signal tensor and the estimated signal tensor. As the filter in one mode relies on the filters in the other modes, the ALS algorithm is used to jointly calculate the MWF filters. In the filtering process, the rank of signal subspace in mode-n needs to be known in order to remove the noise in the orthogonal complement subspace of the signal subspace. For this reason, the AIC algorithm is taken to estimate the rank in mode-n, which makes MWF can reduce noise automatically.

Parafac filtering method was proposed to reduce the rank numbers to be estimated. As aforementioned, the rank in each mode needs to be estimated in MWF, while only one rank needs to be estimated in Parafac filtering, therefore it reduces the rank numbers significantly. Moreover, low-rank Parafac decomposition is unique for rank values higher than one, whereas TUCKER decomposition is not. However, there is not an efficient way to estimate the Parafac rank automatically. We have discussed a rank estimation method in this chapter, however, it is a time-consuming brute force searching way.

The tensor-based denoising methods jointly filter each mode of HSI by employing the multilinear algebra, therefore they can utilize the underlying structure of HSI.

Chapter 3

Bio-inspired optimization algorithms for automatic estimation of multiple subspace dimensions in a tensor-wavelet denoising algorithm

3.1 Introduction

The accurate estimation of signal subspace ranks for the purpose of multidimensional data is still pending and has become a required step since multiway Wiener filtering has been inserted in a wavelet framework. To solve this issue, we propose a criterion to be minimized with a global optimization method.

3.2 Rank estimation issue in a multidimensionnal context

A noisy multidimensional signal will be considered, also called tensor: a signal X impaired by a multidimensional additive, zero-mean white Gaussian noise N [21]. For instance, the additive case holds for hyperspectral images [113, 121]. As concerns the white noise assumption, it is also generally adopted for multidimensional images [21], and permits to focus on the main issue of this paper. In the case where the noise is not white, a prewhitening process could be applied as proposed in [40].

Thus, this tensor can be a model for an HSI, expressed as: $\mathcal{R} = \mathcal{X} + \mathcal{N}$. Tensors $\mathcal{R}$, $\mathcal{X}$, and $\mathcal{N}$ are of size $I_1 \times I_2 \times I_3$. For each spectral band indexed by $i = 1, ..., I_3$, the noise $\mathcal{N}$(:, :, i) is assumed additive, zero-mean, white and Gaussian. The objective is to denoise the tensor R with a subspace-based method. Subspace-based methods have been shown to exhibit good denoising results when applied to data with main salient orientations in the image [79]. They provide an estimated signal tensor which, generally in the literature and the remainder of this paper, is denoted by $\hat{\mathcal{X}}$. This estimate depends on the so-called signal subspace dimensions' or signal subspace ranks' $K_1\times, K_2\times, K_3$ which must be estimated.

3.3 Multiway Wiener filtering in the wavelet framework

In [54], MWF has been inserted into a wavelet framework to denoise images while preserving details. In tensor form, 2-dimensional wavelet packet transform (WPT) is applied for each mode to get the multidimensional wavelet packet transform (MWPT):

$$\mathcal{C}_{\mathbf{l}}^{\mathcal{R}} = \mathcal{R} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3, \tag{3.1}$$

In Eq. (3.1) $\mathcal{C}_{\mathbf{l}}^{\mathcal{R}}$ is the wavelet packet coefficient tensor for levels in $\mathbf{l} = [l_1, l_2, l_3]^T$; for each value of n between 1 and 3, $\mathbf{W}_n \in \mathbb{R}^{I_n \times I_n}$ indicate the WPT matrices, and $\times_n$ denotes the n-mode product [84]. To select the 'frequency' components out of the wavelet packet coefficient tensor, a 'frequency' index is defined for the mode 1, 2 or 3. A set of indices forms the index vector $\mathbf{m} = [m_1, m_2, m_3]^T$. For n equal to 1, 2 or 3 the index values m_n are such that $0 \leq m_n \leq 2^{l_n} - 1$. The coefficient subtensor extracted from $\mathcal{C}_{\mathbf{l}}^{\mathcal{R}}$ and containing the frequency components for a given index vector $\mathbf{m}$ is denoted by $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}}$. It is worth noticing that the size of each coefficient $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}}$ is $\frac{I_n}{2^{l_n}}$ along each mode n, whatever the index vector $\mathbf{m}$ [84].

The principles of multidimensional wavelet packet transform-multiway Wiener filtering (MWPT-MWF) proposed in [84] are to apply MWF to each coefficient subtentor $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}}$. Following the same notation as for the noisy tensor $\mathcal{R}$, let $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}}$ and $\hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}}$ be the coefficient subtensors for the expected tensor $\mathcal{X}$ and its estimate $\hat{\mathcal{X}}$ respectively.

From the Parseval theorem, minimizing the MSE between $\mathcal{X}$ and its estimate $\hat{\mathcal{X}}$ is equivalent to minimizing the MSE between $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}}$ and $\hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}}$ for each $\mathbf{m}$:

$$\|\mathcal{X} - \hat{\mathcal{X}}\|^2 = \|\mathcal{C}_{\mathbf{l}}^{\mathcal{X}} - \hat{\mathcal{C}}_{\mathbf{l}}^{\mathcal{X}}\|^2 = \sum_{\mathbf{m}} \|\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}} - \hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}}\|^2 \tag{3.2}$$

To minimize the expression $\|\mathcal{C}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}} - \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}\|^2$ for all instances of vector $\mathbf{m}$, multiway Wiener filtering is applied. This requires, for each vector $\mathbf{m}$, the knowledge of three rank values.

3.4 Proposed criterion for optimal rank selection

In the literature, the Akaike Information Criterion (AIC) [84] is commonly used to estimate the subspace ranks. AIC estimates the number of sources correctly in an array processing problem [145]. In an array processing context, there are many realizations of the same random signal, and the data are rarely noisy, so AIC exhibits a good behavior. Usually, in the frame of HSI processing, through a stationarity hypothesis, a covariance matrix is computed from the column vectors of the unfolded matrix obtained from the HSI, which are considered as realizations of the same random signal. AIC is applied to the eigenvalues of the covariance matrix obtained for each mode of the HSI [121]. However, it has been shown empirically that there is no clear domination of a subset of eigenvalues with high magnitude with respect to the others [121], and that AIC tends to overestimate the rank values when the data are noisy. In [96], the minimization of the least squares criterion is proposed to estimate the rank values, but its relevance and optimality are not proven. We look for a method which is robust to noise and takes into account the assumptions about our noise model. Maximum likelihood restoration methods have been developed during the last decades for blurred, Poisson noise or additive Gaussian noise models [51]. In the case of Gaussian noise, they yield to the minimization of the least squares criterion [72]. Among other methods [72], mean square error minimization yielded the Wiener filter [41], and MWF [21], which is the core of the methods presented in this work. For these reasons, and because our model for noise is white Gaussian, we propose to minimize a least square error (LSE) criterion to estimate the signal subspace dimensions:

$$J_{\mathbf{m}}(K_1, K_2, K_3) = ||\mathcal{C}^{\mathcal{X}_1}_{\mathbf{l},\mathbf{m}} - \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}(K_1, K_2, K_3)||^2 \tag{3.3}$$

where $||.||$ represents the Frobenius norm, $\mathcal{X}_1$ is a gross estimate of the expected tensor $\mathcal{X}$, and $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}$ is a coefficient subtensor of the final estimate $\hat{\mathcal{X}}$. $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}$ depends on the expected rank values $(K1, K2, K3)$. It is obtained with a fast version of MWF including fixed point algorithm [97].

The criterion $J_{\mathbf{m}}(K_1, K_2, K_3)$ is a nonlinear function of the parameters K_1, K_2, K_3,

hence the need for an adequate optimization method, which must be global. In [95], a single optimization method, particle swarm optimization, is proposed. To the best of our knowledge,no thorough investigation was performed on the comparative performance of optimization methods to estimate the signal subspace ranks in a wavelet framework.

3.5 Brief review on global optimization methods

The considered optimization problem is highly non-linear, and we have no insurance that some constraints on the minimized function are respected. Therefore, we left aside the deterministic optimization methods such as Dividing Rectangles (DIRECT) [52]which assumes that the minimized function is a k-Lipschitzian one. Moreover, as the data considered in this work are impaired with random noise, we focused on stochastic optimization methods. These methods introduce randomness in the initialization and iterative process of the optimization algorithm. Contrary to well-known deterministic methods such as Gradient descent which may converge to a local minimum, this randomness prevents the stochastic optimization method from focusing on a local optimum, but also turns it less sensitive to modeling errors. Further in this work, we provide some details on two bio-inspired optimization methods: swarm optimization [94]and genetic algorithms [100]. First, we present Nelder-Mead [69]. The Nelder- Mead Simplex method [69] is a global optimization method, which is meant to minimize a scalar-valued nonlinear function of several real variables, without any derivative information. It is known to yield a rapid decrease in cost function values [69]. It has been shown that, in dimension two or more, the Nelder-Mead method may converge to a non-critical point of the minimized function [36]. In their seminal work concerning particle swarm optimization, Kennedy and Eberhart [31,55] got inspired by [116]where the term 'particle swarm' was chosen to define the members of a population or test set. In their paradigm, the population members are mass-less and volume-less. Their evolution is described through position, speed, and acceleration parameters [94]. An often cited, now well-known reference [100] introduces genetic algorithms in the context of evolutionary computation which implies the evolution of a population of candidates which is inspired by Darwin's natural selection theory. Another largely cited reference presents basics about genetics, the hierarchical genetic algorithm, and applications to H∞ control, neural network, and speech recognition [93]. For applications which concern specifically image processing, genetic algorithms are combined

with mathematical morphology methods [87], watermarking [102], medical imaging [37], or 3D reconstruction [147]. As GA requires a large number of function evaluations for convergence, and because PSO may prematurely converge to a local minimum, hybrid algorithms were proposed which combine the advantages of both types of methods: genetic algorithms were mixed with particle swarm optimization [53, 103][37-41], or an ant colony algorithm [28]. Recently, such a hybrid algorithm was proposed [16], where the evolutionary natures and social interactions of both algorithms are combined in the frame of a multibiometric system. Nelder-Mead can be a valuable comparative method. However, three parameters should be estimated in the considered optimization problem, and Nelder-Mead method could be stacked in a local minimum of the criterion in Eq. (3.10). Hence, and as a balance to section 3.6., we will then focus on two types of stochastic bio-inspired optimization methods: genetic algorithm (GA) and particle swarm optimization (PSO). Both types of methods provide the global minimum of a scalar function of several variables and are gradient-free.

3.6 Genetic algorithm and particle swarm optimization for tensor rank estimation

GA and PSO are population-based iterative algorithms. They start with an initial random set of Q rank triplets called population. In the following, this population is denoted by $\mathbf{y}_q^{K_1,K_2,K_3}$, $q = 1, \ldots, Q$. Each vector $\mathbf{y}_q = [K_1, K_2, K_3]^T$ is modified throughout the iterations. Vector $\mathbf{y}_q$ is called a chromosome while dealing with a genetic algorithm, and a particle while dealing with swarm optimization.

3.6.1 Principles of Genetic algorithms

The genetic algorithm is a method for solving both constrained and unconstrained optimization problems that is based on natural selection, the process that drives biological evolution. It is based on 'survival of the fittest' theory of Charles Darwin [123]. It was introduced by Holland in [48].

This algorithm is interesting because it's very robust in nature and capable of optimize complex problems, including when a great number of unknown values are expected.

Each individual $\mathbf{y}_q^{K_1,K_2,K_3}$, $q = 1, \ldots, Q$ in the population is called a chromosome, representing a solution to problem at hand. The chromosomes evolve through successive iterations, called generations. During each generation, the chromosomes

$\mathbf{y}_q^{K_1,K_2,K_3}$ are evaluated, using the fitness function $J_{\mathbf{m}}(\mathbf{y}_q^{K_1,K_2,K_3})$ (directly adapted from Eq. (3.10)):

$$J_{\mathbf{m}}(\mathbf{y}_q^{K_1,K_2,K_3}) = ||\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_1} - \hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}}(\mathbf{y}_q^{K_1,K_2,K_3})||^2 \tag{3.4}$$

The genetic algorithm uses three main types of rules at each step to create the next generation from the current population [135]:

• Selection rules select the individuals, called parents, that are best adapted to their environment and contribute to the population at the next generation.

• Crossover is a genetic operator that recombines two chromosomes (a pair of individuals) to produce a new chromosome (called an offspring). The new chromosome may be better than both of the parents if it takes the best characteristics from each of the parents. Crossover occurs during evolution according to a user-definable crossover probability.

• Mutation promotes diversity in population characteristics, to prevent the algorithm from being trapped in local minima [16].

The five main steps of a GA algorithm [100] are described in Algorithm 1.

Algorithme 1 Pseudo-code: Genetic Algorithm for multiple rank estimation

Inputs: fitness function of Eq. (3.4), tensor gross estimate $\mathcal{X}_1$, small factor ε set by the user, to stop the algorithm.

1. Set iteration number iter $= 1$, and maximum number of iterations maxiter.
2. Create an initial population composed of Q random chromosomes with all required rank values $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter})$, $q = 1, \ldots, Q$. This initial population takes the form of a matrix with Q rows and 3 columns.
3. Evaluate fitness function value of Eq.(3.10) for each each chromosome $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter})$ in the population.
4. Generate the new population:
   (a) Selection: choose new parents through fitness function value.
   (b) Crossover: pair probabilistically the parents to create the offsprings with crossover fraction X_f.
   (c) Mutation: modify slightly some components, chosen randomly, of each offspring. We now afford the population $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter}+1)$, $q = 1, \ldots, Q'$
5. Replace the current population with the new one, obtained at step 4.
6. Increase iter ; if iter $<$ maxiter, or $||\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter}) - \mathbf{y}_q^{K_1,K_2,K_3}(\text{iter}-1)|| > \epsilon$ go to step 3.

Output: estimated rank values $\hat{K1}, \hat{K2}, \hat{K3}$

Here are some details for GA algorithm:
At each iteration iter, $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter})$ is the current chromosome q.
At step 2, the initial population is stored in a $Q \times 3$ matrix. Each row of the matrix is 1×3 vector with three random values. The n^{th} component of this vector is a random value between 1 and $\frac{I_n}{2^{l_n}}$. Therefore, the q^{th} row of the matrix is $\mathbf{y}_q^{K_1,K_2,K_3}$ (1).

In the following, we exemplify the step 4. of Algorithm 1, with two chromosomes with index q and q':
At step 4, after selection, we assume that we afford, among the selected chromosomes, $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter}) = [K_1, K_2, K_3]$ and $\mathbf{y}_{q'}^{K_1,K_2,K_3}(\text{iter}+1) = [K_1', K_2', K_3']$
At the iteration iter+1, crossover provides the new offsprings. Whatever n, either

K_n or K'_n is chosen, randomly, as the n^{th} component of $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter}+1)$ and $\mathbf{y}_{q'}^{K_1,K_2,K_3}(\text{iter}+1)$. For instance: $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter}+1) = [K_1, K'_2, K'_3]$ and $\mathbf{y}_{q'}^{K_1,K_2,K_3}(\text{iter}+1) = [K'_1, K_2, K'_3]$.

Then after mutation, these offsprings are slightly modified: $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter}+1) = [K_1+\kappa_1, K'_2+\kappa_2, K'_3+\kappa_3]$ and modified: $\mathbf{y}_{q'}^{K_1,K_2,K_3}(\text{iter}+1) = [K'_1+\kappa'_1, K_2+\kappa'_2, K'_3+\kappa'_3]$, where, whatever n, κ_n and κ'_n are random values which are small compared K_n and K'_n.

For the practical implementation of Algorithm 1, we have selected a globally convergent Lagrangian algorithm which permits to define lower and upper bounds on the estimated values $\hat{K_1}$, $\hat{K_2}$,$\hat{K_3}$ [19]. Our motivation is the following: in the fitness function of Eq. 3.4, the values of K_1, K_2, K_3 cannot be less than 1, and greater than the size of the wavelet packet coefficient $\hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_1}$ along each mode $n : 0 < K_n \leq \frac{I_n}{2^{l_n}}$, where n is equal to either 1,2, or 3.

3.6.2 Principles of particle swarm optimization algorithm

The basic PSO algorithm [55] is inspired by the behavior of fish or bird swarms. The basic principles of PSO algorithm are to simulate the communication between animals of a swarm, which aim at locating food for instance. PSO updates the behavior of such individuals of the swarm, called particles, through time. As shown in Algorithm 2, it is implemented as an iterative algorithm which, for the current iteration number iter, computes two characteristics of the particles: velocity, and position.

Here are some detail for PSO algorithm:

At each iteration iter, $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter})$ is the current position of particle q, and $\mathbf{V}_q^{K_1,K_2,K_3}(\text{iter})$ is its velocity.

At step 1, the initial population is stored in a $Q \times 3$ matrix. Each row of the matrix is a 1×3 vector with three random values. The n^{th} coefficient of this vector is a random values between 1 and $\frac{I_n}{2^{l_n}}$ Therefore, at iteration 1, the q^{th} row of the matrix is $\mathbf{y}_q^{K_1,K_2,K_3}$ (1).

In the following, we exemplify the step 4. of Algorithm 2, with one particle with index q. This particle is expressed as $\mathbf{y}_q^{K_1,K_2,K_3}(iter) = [K_1, K_2, K_3]$. In Eqs. (3.5)

Algorithme 2 Pseudo-code: Particle Swarm Optimization for multiple rank estimation

Inputs: fitness function of Eq. (3.4), tensor gross estimate $\mathcal{X}_1$, small factor ϵ set by the user, to stop the algorithm.

1. Set iteration number iter $= 1$, create an initial population composed of Q random particles with all required rank values $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter})$, $q = 1, \ldots, Q$. This initial population takes the form of a matrix with Q rows and 3 columns.
2. Evaluate fitness function value of each particle $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter})$, $q = 1, \ldots, Q$.
3. Update the local best particles $\mathbf{p}_q^{K_1,K_2,K_3}$ $(q = 1, \ldots, Q)$, and the global best particle $\mathbf{G}^{K_1,K_2,K_3}$,
4. Repeat steps for each particle q, $q = 1, \ldots, Q$:
   (a) Compute velocity $\mathbf{V}_q^{K_1,K_2,K_3}(\text{iter} + 1)$
   (b) Compute position $\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter} + 1)$
5. Exchange current population from new one, obtained at step 4.
6. If iter $<$ maxiter or $\left\| \mathbf{y}_q^{K_1,K_2,K_3}(\text{iter}) - \mathbf{y}_q^{K_1,K_2,K_3}(\text{iter} - 1) \right\| > \epsilon$, and go to step 2.

Output: estimated rank values $\hat{K_1}, \hat{K_2}, \hat{K_3}$

and (3.6), we show how to get an updated particle, at iteration *iter +1*.

At step 4a, the velocity is computed as follows:

$$\mathbf{V}_q^{K_1,K_2,K_3}(\text{iter}+1) = W\ \mathbf{V}_q^{K_1,K_2,K_3}(\text{iter}) + \gamma_{1q}\ r_{1q}(\mathbf{p}_q^{K_1,K_2,K_3} - \mathbf{y}_q^{K_1,K_2,K_3}(\text{iter}))$$

$$\ldots + \gamma_{2q}\ r_{2q}(\mathbf{G}^{K_1,K_2,K_3} - \mathbf{y}_q^{K_1,K_2,K_3}(\text{iter})) \tag{3.5}$$

At step 4b, the position at iteration *iter +1* is computed as the summation of the position at iteration *iter* and the velocity at iteration *iter+1*.

$$\mathbf{y}_q^{K_1,K_2,K_3}(\text{iter}+1) = \mathbf{y}_q^{K_1,K_2,K_3}(\text{iter}) + \mathbf{V}_q^{K_1,K_2,K_3}(\text{iter}+1) \tag{3.6}$$

In Eqs. (3.5) and (3.6), the arithmetic operations are computed componentwise. For a given particle q, the velocity is influenced by two contributions: the cognitive one and the social one. The cognitive contribution is due to personal best location of the particle over all iterations. It is denoted by $\mathbf{p}_q^{K_1,K_2,K_3}$ in Eq. (3.5). Social contribution represents global best location over all particles in the swarm and all iterations. It is denoted by $\mathbf{G}^{K_1,K_2,K_3}$ in Eq. (3.5).
In Eq. (3.5), W is the inertia weight, γ_{1q} and γ_{2q} are the acceleration constants encouraging a local and a global search respectively. The probabilistic aspect of the transition between two iterations of Algorithm 2 is due to r_{1q} and r_{2q}, which, for each particle q and each iteration *iter*, are two numbers generated randomly between 0 and 1. A large inertia weight(W) facilitates a global search while a small inertia weight facilitates a local search. We look forward to encourage a global search for the first iterations, and a local search for the last iterations. Hence, we fix an initial value W_{Init} and a final value W_{Final} for the weighting coefficient. At the iteration iter, the weighting coefficient is computed as: $W = W_{Init} - \frac{(W_{Init}-W_{Final})*\text{iter}}{\text{maxiter}}$, where maxiter is the final iteration number. When this last iteration number is attained, or when the Frobenius norm of the difference between estimated vectors at iteration iter and $\text{iter}-1$ is less than a fixed threshold ϵ (see step 6 of Algorithm 2), the position vector $\mathbf{G}^{K_1,K_2,K_3}$ is a row vector containing 3 components which are the final estimated values $\hat{K1}, \hat{K2}, \hat{K3}$, of the signal subspace ranks.

3.6.3 Comparative discussion: GA and PSO

From Algorithms 1 and 2, we can learn that GA and PSO share many common points: they search a population of points in parallel, not a single point; they do not require derivative information or other auxiliary knowledge, only the objective function. They use probabilistic transition rules between one iteration to the next,

not deterministic ones. This probabilistic aspect is due to the probabilistic crossover and mutation at steps 4(b) and 4(c) in Algorithm 1 (GA). The role of mutation is often seen as providing a guarantee that the probability of searching any given triplet of rank values will never be zero. The crossover routines recombine pairs of individuals with given probability to produce offspring. In Algorithm 2 (PSO), the probabilistic aspect is due to random parameters r_{1q} and r_{2q} (see Eq. (3.5) and step 4(b) of Algorithm 2). The role of these parameters is to enable the scan of the whole research space and to avoid the algorithm to be trapped in a local minimum.

Different terms are used for similar features in both types of algorithms: both methods start with a population of randomly generated candidate solutions, called individuals or phenotypes in GA, and particles in PSO. The population is evolved toward better solutions, regarding a fitness function. Each candidate solution has a set of properties, called chromosomes or genotype in GA, position, and velocity in particle swarm optimization.

Both update the population and search for the optimum with random techniques. Both systems do not guarantee success. GA and PSO are different mainly in the way the information is shared between candidate solutions. In a nutshell, advantages and drawbacks of both methods are as follows: In GA, candidate solutions with much diversity are created. Every chromosome shares information with at least another chromosome, or mutates. The reproductive success varies with fitness, but the best solution at a given iteration is not preserved. Hence, a large number of iterations is generally required for convergence [16].

In PSO, particles do not undergo crossover and mutation: they update themselves with the internal velocity, which depends on local $\mathbf{p}_q^{K_1,K_2,K_3}$ and global $\mathbf{G}^{K_1,K_2,K_3}$ best candidates selected at the previous iteration (see Eqs. (3.5) and (3.6)). There exists a single particle $\mathbf{G}^{K_1,K_2,K_3}$which has an influence on the behavior of the other candidate solutions. Hence, PSO includes a 'memory' feature: over all iterations, the personal best of each particle and the overall best are taken into account to compute velocity. So, the evolution only looks for the best solution at a given iteration. In PSO, there is no selection operator, and the higher the number of particles, the higher the computational load. However, compared with GA, and even if the number of particles remains the same throughout the evolution, the particles tend to converge faster to the final solution.

In the considered problem, we look for only three parameters which are the signal subspace ranks. Hence, and as a balance to section 3.6, we infer that PSO should be faster than GA or any hybrid algorithm. In the next section, we insert the optimization algorithm in a tensor-wavelet framework.

3.7 Unsupervised multiway Wiener filtering in wavelet framework

In this section, we show how to obtain an unsupervised MWF-MWPT algorithm. We estimate the multiple signal subspace dimensions automatically. We also propose a method to set the number of wavelet packet decomposition levels for each mode, and we refine the reference tensor with a recursive algorithm.

3.7.1 Automatic estimation of multiple ranks in wavelet framework

It is necessary to adapt a global optimization method for multiple rank estimation when MWF is inserted in the wavelet framework [84]. Indeed, even if we restrict the study to third-order tensors such as color, multispectral or hyperspectral images, the number of required rank values may be so elevated that they cannot be fixed manually: One triplet of rank values must be estimated for each wavelet packet coefficient. There exists 2^{l_n} coefficients for mode n, $n = 1, 2, 3$. Each wavelet packet coefficient such as $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$ is a tensor of order 3 [84], so 3 rank values must be estimated for each wavelet packet coefficient. When the decomposition is performed, the number of coefficients is equal to $\prod_{n=1}^{3} 2^{l_n}$. In total, the number of rank values to be estimated in the whole algorithm is given by $3\prod_{n=1}^{3} 2^{l_n}$. This number is possibly elevated. Therefore, it is of great interest to estimate the rank values automatically.
We wish to minimize all terms of the summation in Eq. (3.2), knowing that the noise-free tensor X is not available. For it, we propose Algorithm 3, *multidimensional wavelet packet transform and multiway Wiener filtering with rank estimation* (MWPT-MWF-RE).

In Algorithm 3, $\mathbf{H}_{1,m}$, $\mathbf{H}_{2,m}$ and $\mathbf{H}_{3,m}$ denote the n-mode filters of MWF, which depend on rank values K_1, K_2, K_3 [21]: $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$ denote the wavelet packet coefficients of R. Note that, for computational load purposes, different versions of MWF may be used in the algorithm. Here are some details about the key steps of Algorithm 3: At

Algorithme 3 MWPT-MWF-RE

Input: noisy tensor $\mathcal{R}$, gross estimate $\mathcal{X}_1$ of $\mathcal{R}$ obtained by MWF with fixed rank values.

1. Compute the wavelet decomposition of the noisy tensor $\mathcal{R}$ and of the gross estimate $\mathcal{X}_1$: $\mathcal{C}_{\mathbf{l}}^{\mathcal{R}} = \mathcal{R} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3$, $\mathcal{C}_{\mathbf{l}}^{\mathcal{X}_1} = \mathcal{X}_1 \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3$.

2. Extract the wavelet coefficients [84]:
   $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}} = \mathcal{C}_{\mathbf{l}}^{\mathcal{R}} \times_1 \mathbf{E}_{m_1} \times_2 \mathbf{E}_{m_2} \times_3 \mathbf{E}_{m_3}$, $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_1} = \mathcal{C}_{\mathbf{l}}^{\mathcal{X}_1} \times_1 \mathbf{E}_{m_1} \times_2 \mathbf{E}_{m_2} \times_3 \mathbf{E}_{m_3}$.

3. Perform rank estimation and denoising for each wavelet packet coefficient $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}}$ associated with vector $\mathbf{m}$:

   (a) Subsample $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}}$ and $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_1}$ by factors S_1, S_2, and S_3 along the first, second and third mode respectively to get a smaller version $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}_S}$ and $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_{1S}}$. Notice that $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_{1S}}$ is a gross estimate of $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}_S}$.

   (b) Estimate with a global optimization method the optimal rank values $\hat{K_1}, \hat{K_2}, \hat{K_3}$ in terms of the criterion:

   $$J_{\mathbf{m}}(K_1, K_2, K_3) = ||\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_{1S}} - \hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_S}(K_1, K_2, K_3)||^2$$

   In the equation above, $\hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_S}$ is obtained from $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}_S}$ with MWF: $\hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_S} = \mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}_S} \times_1 \mathbf{H}_{1,\mathbf{m}} \times_2 \mathbf{H}_{2,\mathbf{m}} \times_3 \mathbf{H}_{3,\mathbf{m}}$, where $\mathbf{H}_{1,\mathrm{m}}$, $\mathbf{H}_{2,\mathrm{m}}$ and $\mathbf{H}_{3,\mathrm{m}}$ depend on K_1,K_2 and K_3 respectively.

   (c) Multiply the values obtained at step (b) by S_1, S_2 and S_3 respectively: $\hat{K_1} = S_1 K_1$, $\hat{K_2} = S_2 K_2$, and $\hat{K_3} = S_3 K_3$.

   (d) Apply MWF to each coefficient subtensor $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}}$, with the optimal rank values:
   $\hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}} = \mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}} \times_1 \mathbf{H}_{1,\mathbf{m}} \times_2 \mathbf{H}_{2,\mathbf{m}} \times_3 \mathbf{H}_{3,\mathbf{m}}$ where $\mathbf{H}_{1,\mathrm{m}}$, $\mathbf{H}_{2,\mathrm{m}}$ and $\mathbf{H}_{3,\mathrm{m}}$ depend on $\hat{K_1}$, $\hat{K_2}$ and $\hat{K_3}$ respectively.

4. Concatenate all coefficients $\hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}}$ to afford the wavelet packet decomposition $\hat{\mathcal{C}}_{\mathbf{l}}^{\mathcal{X}}$ as a whole [84].

5. Retrieve the final estimated tensor by inverse wavelet packet transform:
   $\hat{\mathcal{X}} = \hat{\mathcal{C}}_{\mathbf{l}}^{\mathcal{X}} \times_1 \mathbf{W}_1^T \times_2 \mathbf{W}_2^T \times_3 \mathbf{W}_3^T$.

Output: denoised tensor $\hat{\mathcal{X}}$.

step 1, we compute wavelet packet decomposition of the processed $\mathcal{R}$, to get $\mathcal{C}_{\mathbf{l}}^{\mathcal{R}}$, and the wavelet decomposition of the gross estimate $\mathcal{X}_1$ to get $\mathcal{C}_{\mathbf{l}}^{\mathcal{X}_1}$.
At step 2, we extract all the coefficients out of the whole decomposition: we get coefficients $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}}$ and $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_1}$. This process is described in detail in [84].
At step 3, we firstly estimate the best rank values, and secondly use these rank values to denoise the coefficients with MWF.
To reduce the computational load of step 3(b), the estimation of the best rank values is performed on subsampled versions of $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}}$ and $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_1}$. The subsampling is made at step 3(a). In step 3(b), we seek for the best rank values, without expecting the best estimates of the singular vectors used in MWF. That is why, still to reduce the computational load, a fast version using fixed-point algorithm and inverse power method [97] may be used if significant data are processed with low SNR values. At step 3(c) we multiply the rank values which were obtained by the subsampling factors. Then, we perform denoising at step 3(d). At step 3(d), the singular vectors used in the Wiener filters along each mode are computed through singular value decomposition, and are then optimal in the least squares sense [33]. At step 4 a simple concatenation of the wavelet packet coefficients is performed. The whole set of coefficients is expressed in a synthetic way as $\hat{\mathcal{C}}_{\mathbf{l}}^{\mathcal{X}}$, which is by definition the wavelet packet decomposition of the denoised tensor. At step 5 the denoised tensor is provided by inverse wavelet packet transform. With a global optimization method, Algorithm 3 is supposed to converge asymptotically towards the best set of rank values, and therefore the best possible estimate $\hat{\mathcal{X}}$ at step 5 and as the final output of the algorithm. In practice, the total number of iterations, that is, the parameter *maxiter*, is fixed automatically: whatever the optimization algorithm is considered, the algorithm stops if the maximum *a priori* fixed number of iteration is attained or if the estimated values do not vary from one iteration to the next.

3.7.2 Estimation of the number of decomposition levels

At step 1 of Algorithm 3, the number of wavelet decomposition levels for each mode, contained in vector $\mathbf{l} = [l_1, l_2, l_3]^T$, is a required input for the computation of the wavelet coefficients. In [84], these values are chosen empirically. We propose an algorithm to estimate vector $\mathbf{l}$. To reduce the range of acceptable values, we first notice that, up to now, the interest of performing decomposition along the wavelength mode has not been emphasized when multidimensional images are processed with MWPT-MWF. That is, l_3 is always fixed to 0. Secondly, without loss of generality, there is no prior assumption concerning the structure of the images. That is, we

assume there is no feature which is expressly related to rows or columns. So, the row and column dimensions are processed in the same manner, and the number of decomposition levels is equal for each dimension: $l_1 = l_2$. Thirdly, it has been shown empirically in [84] that there is no interest to perform wavelet packet decomposition with more than three levels. From the considerations above, we restrict the set of candidate instances for $\mathbf{l}$ to $[1,1,0]^T$, $[2,2,0]^T$, and $[3,3,0]^T$.

To make our choice between these three possibilities, we apply Algorithm 4 below. . In Algorithm 4, at step 1, subsampling is applied to yield a faster process. At step 2(a), we notice that, for each mode n where n is equal to 1,2, or 3, the value of the signal subspace dimension is $K_n = \frac{I_n}{S_n 2^{1+l_n}}$. It is fixed to half the size of the processed wavelet packet coefficient. Notice that we do not choose optimal values of K_n in any sense at step 2(a). Indeed, we do not seek for the best denoised tensor at this step, but for the best number of decomposition levels.

Algorithme 4 Estimation of the number of decomposition levels

Input: noisy tensor $\mathcal{R}$ of size $I_1 \times I_2 \times I_3$, gross estimate $\mathcal{X}_1$ of $\mathcal{R}$ obtained by MWF with fixed rank values $K_1 = \frac{I_1}{2}$, $K_2 = \frac{I_2}{2}$, $K_3 = \frac{I_3}{2}$, set of candidate vectors $\mathbf{l}$.

1. Apply subsampling by factors S_1, S_2, and S_3 along modes 1, 2 and 3, to tensor $\mathcal{R}$ and $\mathcal{X}_1$,
2. For each candidate vector $\mathbf{l} = [l_1, l_2, l_3]^T$:
   (a) Apply algorithm 3 with fixed values of the signal subspace dimensions for each decomposition level:

   $K_1 = \frac{I_1}{S_1\ 2^{1+l_1}}$, $K_2 = \frac{I_2}{S_2\ 2^{1+l_2}}$, $K_3 = \frac{I_3}{S_3\ 2^{1+l_3}}$, that is, half of the size of the coefficients.

   (b) Retrieve the error between the estimate obtained with Algorithm 3 at step 2. (a) and the subsampled version of $\mathcal{X}_1$.
3. Select the vector $\mathbf{l}$ for which the error is minimum.

Output: optimal vector $\mathbf{l}$ in terms of least squares error.

Still, the reference tensor which is used in Eq.(3.10) is obtained in a supervised way: with MWF with manually fixed ranks. In the next subsection, we propose a method to reduce the dependence of the algorithm, and to get a better reference

tensor.

3.7.3 Iterative algorithm for the estimation of the reference tensor

In Eqs.(3.10) and (3.4), a reference tensor is required, which should ideally be the expected tensor itself. As we do not afford it, we propose an iterative algorithm: we first provide a gross estimate obtained with MWF, and refine in an iterative manner this reference tensor. At iteration 1 the reference tensor is the gross estimate $\mathcal{X}_1$. At iteration $r > 1$ the reference tensor is $\hat{\mathcal{X}}_{r-1}$, the result of denoising obtained at the previous iteration $r-1$. This process is repeated iteratively. The algorithm stops when the estimate at iteration r differs from the estimate at iteration $r-1$ by a small factor δ.

In Fig. 3.1, we present the flowchart of the overall algorithm, with iterative estimation of the reference tensor and estimation of the ranks with a global optimization method. This overall algorithm is described in detail in Algorithm 5 below.

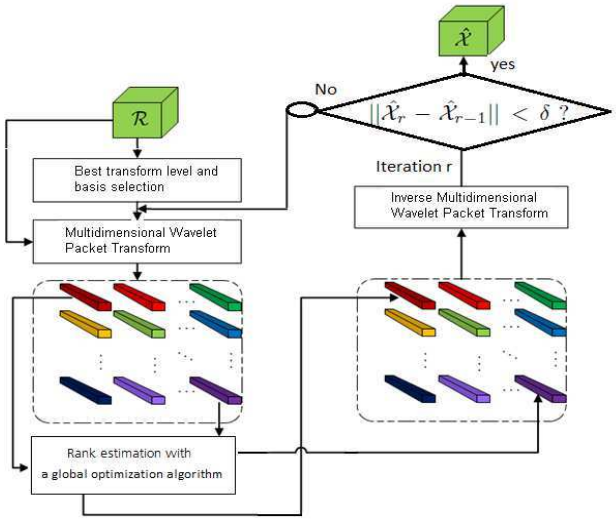


Figure 3.1: Rank estimation with global optimization method in wavelet packet transform domain: iterative process

The chart in Fig. 3.1 corresponds to the algorithms presented above in the work as follows:

If Algorithm 5 stops at iteration $r = 1$, that is, at step 3.a), it is equivalent to selection of the best vector $\mathbf{l}$ followed by Algorithm 3. In the rest of the work, we denote Algorithm 3 as MM-GA, or MM-PSO depending on the optimization method. We denote Algorithm 5 as MM-Iterative. MM stands for MWPT-MWF.

Algorithme 5 Overall algorithm

Input: noisy tensor $\mathcal{R}$ of size $I_1 \times I_2 \times I_3$,

1. Obtain the gross estimate $\mathcal{X}_1$ of $\mathcal{R}$ by MWF with fixed rank values. Apply Algorithm 4 to select the best vector $\mathbf{l}$ with values of the number of decomposition levels for each mode.

2. Set iteration $r = 1$; choose a small factor δ.

   (a) Apply Algorithm 3 (MWPT-MWF-RE) with $\mathcal{X}_1$ obtained at step 1, as gross estimate. Algorithm 3 provides $\hat{\mathcal{X}}_r$ using the optimization method described in Algorithm 1 or Algorithm 2 for instance.

   (b) Increase the iteration index: $r = r + 1$; Set the gross estimate as $\hat{\mathcal{X}}_{r-1}$, the tensor estimated at step 3. a)

   (c) Apply Algorithm 3 (MWPT-MWF-RE) with $\hat{\mathcal{X}}_{r-1}$ obtained at step 3. a), as gross estimate. Algorithm 3 (MWPT-MWF-RE) provides $\hat{\mathcal{X}}_r$.

   (d) Compute the error $e = ||\hat{\mathcal{X}}_r - \hat{\mathcal{X}}_{r-1}||^2$.

   (e) If $e > \delta$, go to step 3. b). Otherwise, set $\hat{\mathcal{X}} = \hat{\mathcal{X}}_r$.

Output: denoised tensor $\hat{\mathcal{X}}$.

3.8 Results

3.8.1 Experimental setup

In this section, we evaluate the performances of the proposed methods on RGB and multispectral images. We consider images where the relevant features are localized in some regions of the image. From the presence of such small local features we expect methods based on wavelet packet decomposition to provide good results, as wavelet packet decomposition permits to separate the processing of high frequency and low frequency features.

In the recent state-of-the-art about HSI denoising [78, 90], the authors emphasize the quality of the recent HSI sensors: since in most bands noise is nearly invisible for human eyes, the authors impair the images with synthetic data. In this paper, we focus on a specific application fluorescence imaging of plant leaves [35]. In the context of fluorescence imagery, the noise magnitude is inherently rather elevated, compared to the applications considered elsewhere. Indeed, the emitted fluorescence light has a much lower magnitude than the excitation light provided by a monochromator.

Consequently, we will firstly present some quantitative results obtained from artificially impaired images. These results are at first obtained from the RGB Baboon image. Secondly, they are obtained from multispectral aerial images. These multispectral images are extracted from hyperspectral images that were obtained with the ROSIS acquisition sensor over the urban area of Pavia, northern Italy. Thirdly, we present results obtained from fluorescence images which are inherently noisy. We remind that $\mathcal{X}$ is the noise-free tensor, $\mathcal{R}$ is the noisy tensor and $\hat{\mathcal{X}}$ is the estimated tensor. The processed noisy images follow the tensor model and are expressed as:$\mathcal{R} = \mathcal{X} + \mathcal{N}$.

The results obtained are evaluated in terms of SNR and perceptual image quality. The perceptual image quality is measured through mean structural similarity (SSIM) [143] over all spectral bands.

We remind that $SNR = 10\ log(\frac{||\mathcal{X}||^2}{||\mathcal{X}-\hat{\mathcal{X}}||^2})$ and $SSIM(\hat{\mathcal{X}}, \mathcal{X})$ $= lu(\mathcal{X}, \hat{\mathcal{X}})co(\mathcal{X}, \hat{\mathcal{X}})st(\mathcal{X}, \hat{\mathcal{X}})$, where lu, co, and st are three terms aiming at image comparison: luminance, contrast, and structure.

Programmes were written in $Matlab^{®}$, and executed on a PC running Windows, with a 3GHz double core and 3GB RAM.

Unless for the last considered application to fluorescence imaging, the images are artificially impaired with white, identically distributed random noise with the following input SNR (SNR_{in}) values (in dB): 5, 7.5, 10, 12.5, and 15. We present both examples for a specific value of SNR_{in} such as 5 dB or 12.5 dB, and statistical results obtained with the values of SNR_{in} cited above and 20 noise realizations.

Values of input SNR between 5 and 15 dB are realistic for the considered fluorescence images, hence our choice. The results are evaluated in terms of output SNR (SNR_{out}) and output SSIM ($SSIM_{out}$). In the wavelet decomposition, following the recommendations in [84] we choose Coiflets as wavelet functions.

As to the number of decomposition levels along each mode, we set the number l_3 for the third mode through the following heuristic process: with fixed number of decomposition levels along the first and two modes to 2, and tried several values for l_3. We noticed that performing decomposition along the wavelength mode does improve the denoising results in terms of output SNR, at least for the considered data. Hence, we set $l_3 = 0$. We then referred to Algorithm 4 to estimate l_1 and l_2 automatically. To run Algorithm 4, we set the subsampling factor S to 4. Unless specified, its yields $l_1 = l_2 = 2$ as optimal values. An example of the number of signal subspace dimension values to be estimated is as follows: if we obtain $\mathbf{l} = [2, 2, 0]^T$, the decomposition yields 16 wavelet packet coefficients, which are 3^{rd}-order tensors. The total number of rank values to be estimated is $3 \prod_{n=1}^{3} 2^{l_n} = 3\ 2^2\ 2^2\ 2^0 = 48$

We test the proposed methods in Algorithm (3) with either genetic algorithm, particle swarm, or Nelder-Mead optimization methods. We also test the iterative method presented in Algorithm 5. These methods are denoted as MM-GA, MM-PSO, or MM-Nelder, for Algorithm 3, and MM-Iterative for Algorithm 5. We selected state-of-the-art comparative methods, based on data projection on a signal subspace: AIC estimating the subspace ranks in the wavelet framework denoted by MM-AIC, truncation of higher order decomposition denoted by THOSVD [26] and multiway Wiener filtering denoted by MWF [106]. We omitted some other methods, based on band-by-band wavelet processing such as FoRWaRD, or classical linear smoothing

filters because they provide either poor SSIM or output SNR values [95]. We propose Perona-Malik [114] as a comparative method. When Perona Malik is applied, the image is smoothed through a diffusion process, in such a way as to encourage intraregion smoothing rather than interregion smoothing. The Perona-Malik method is denoted as 'Per-Mal' in the tables presented further in the paper. Following the recommendations in [114], the parameters are as follows: the number of iterations is 15, the constant integration δt is fixed to its maximum value, $\frac{1}{7}$; the gradient modulus threshold that controls the conduction is $k = 30$; and the conduction coefficient function privileges high contrast edges. The tested algorithms require a few parameters which are set once for all processed images: following the recommendations in [55] PSO is run with the swarm size is $Q = 10$ (about three times the number of unknowns which is three) and $\epsilon = 10^{-6}$. This yields $maxiter = 100$ iterations. The acceleration constants γ_{1i} and γ_{2i} are set to 2 and 3 respectively. We run a version of GA using a Lagrangian algorithm [19,38], with Q = 300 individuals in the initial population. We use the fitness function to provide a measure of how individuals have performed in the problem domain. When THOSVD or MWF are run, we set the values of signal subspace dimension equal to 60$\%$ of the data size along each mode. Unless specified, to create the reference tensor we set the values of signal subspace dimension equal to 60$\%$ of the data size along each mode. When the iterative process described in subsection 3.7.3. is used, 3 iterations are used to refine the gross tensor estimate. Unless specified, the subsampling factors used to obtain a smaller version of the wavelet packet coefficient tensor are set to $S_1 = S_2 = 4$, and $S_3 = 2$ in algorithms 3 and 4. For the RGB display of the multispectral images throughout the section, we select 3 representative bands. Unless specified, they are in the red ($690nm$), green ($550nm$), and blue ($450nm$) wavelength domains respectively.

3.8.2 Baboon RGB image 256×256×3

To process this RGB image, the values of signal subspace dimension in the reference tensor are set to 80$\%$ along the spatial modes, and 100 $\%$ along the wavelength mode, that is, 3 because only three bands compose this image. The subsampling factors are set to $S_1 = S_2 = 4$, and $S_3 = 1$. This image is processed with 2 wavelet packet decomposition levels along the spatial modes, leading to wavelet packet coefficients of size 64×64×3.

3.8.2.1 Exemplification with $SNR_{in} = 5\ dB$

Table 3.1 provides the estimated values for the first computed detail coefficient, and for the approximation coefficient. Notice that the spatial ranks are multiples of 4, because $S_1 = S_2 = 4$. MM-GA and MM-PSO yield similar spatial rank values, whereas AIC tends to underestimate the value of the rank along the first mode (rows), for both approximation and detail coefficients. Otherwise, considering the detail coefficient, the maximum value is obtained for rows and columns. This can be explained as follows: In the case where salient structures are present in an image, such as horizontal lines or columns, there exists a set of columns or rows which are mutually proportional, and the rank of such images is rather low. When we consider the estimated rank values for detail coefficient (see Table 3.1), we notice that the rank values are elevated (64 for all but one value). This is because there is no salient structure in the detail coefficients in the case of this image: it rather contains noise.

Method / Coefficient	MM-GA	MM-PSO	MM-AIC
Approximation	(52, 64, 2)	(52, 56, 3)	(12, 64, 3)
Detail	(64, 64,2)	(64, 64,2)	(16, 64, 3)

Table 3.1: Baboon RGB image 256×256×3: Estimated ranks values

Fig. 3.2 presents the results obtained with the proposed and comparative methods on the Baboon image. It can be noticed that Perona-Malik yields a blurred image, that the image provided by MM-AIC is rather blurred along rows. When MM-Iterative is applied, the result image is slightly more noisy than the result provided by MM-PSO, but the contrast and the contours are better preserved. This is done to the cost of a higher computational load (see Table 3.2).

Per-Mal	MM-GA	MM-Nelder	MM-PSO
0.931	49.03	11.59	8.931
MM-AIC	MM-Iterative	THOSVD	MWF
2.398	24.59	0.143	0.369

Table 3.2: Baboon RGB image 256×256×3: Computational load all methods (in sec.)

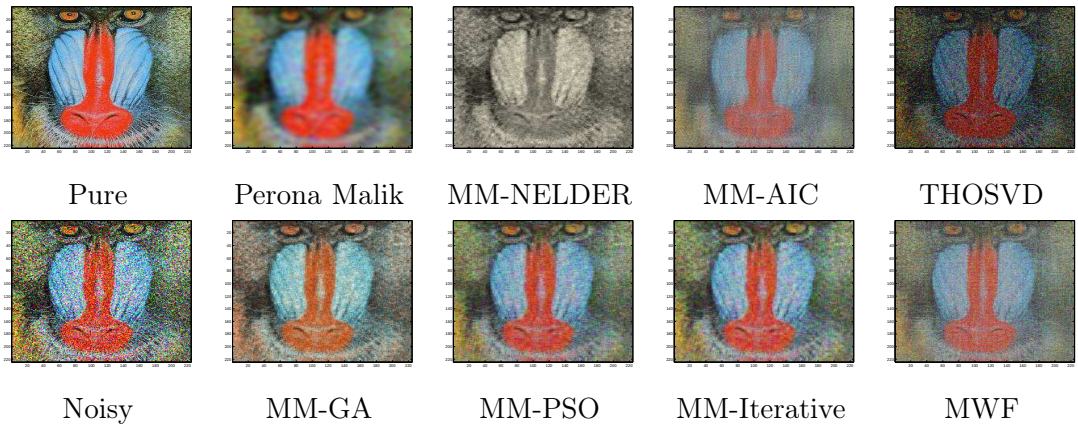


Figure 3.2: Baboon RGB image 256×256×3: pure, noisy 5 dB, denoising results.

3.8.2.2 Statistical results

The statistical results presented in Table 3.3, and Table 3.4 show that, for all input SNR values but 5 dB, the proposed methods MM-GA, MM-PSO, and MM-Iterative perform better than Perona-Malik, MM-AIC, MWF or truncation of HOSVD, regarding output SNR and $SSIM$. Firstly, in the case where $SNR_{in} = 5dB$, we notice that the output SNR values provided by MM-AIC are significantly lower than those obtained with MM-GA or MM-PSO due to the blurred aspect of the result obtained by AIC in Fig. 3.2, and to the underestimation of the rank along rows. For such an image with 3 bands, the comparative Perona-Malik method is valuable for the low SNR value of 5dB but the output SNR is almost the same whatever the input SNR. Secondly, if we consider the proposed methods, and for any value of input SNR, when the iterative method is used, the output SNR and $SSIM$ are slightly better than when PSO is used, at the expense of a larger computational load: $8.931s$. for MM-PSO, $24.59s$. for MM-Iterative, and $2.398s$. for MM-AIC. When this image is considered, MM-GA is significantly slower than MM-PSO. That is why PSO is used in the iterative method, and not GA. We wish to confirm this with multispectral images with more bands.

SNR_{in}	5 db	7,5 db	10 db	12,5 db	15 db
SNR_{out} Per-Ma	13.24	13.36	13.40	13.45	13.46
SNR_{out} MM-GA	12.57	13.87	15.05	16.44	17.64
SNR_{out} MM-Nelder	9.57	9.92	10.25	10.55	10.78
SNR_{out} MM-PSO	12.84	14.065	16.43	17.92	19.52
SNR_{out} MM-AIC	8.81	10.29	12.01	13.96	15.50
SNR_{out} MM-Iterative	13.30	14.52	16.48	17.98	19.61
SNR_{out} THOSVD	6.48	8.67	10.96	13.33	15.78
SNR_{out} MWF	10	12.07	14.14	16.04	17.93

Table 3.3: Baboon RGB image 256×256×3: SNR_{out} all methods

$SSIM_{in}$	0.2530	0.344	0.451	0.561	0.665
$SSIM_{out}$ Per-Ma	0.2500	0.3000	0.3000	0.3100	0.3100
$SSIM_{out}$ MM-GA	0.4671	0.5588	0.6433	0.7313	0.8038
$SSIM_{out}$ MM-Nelder	0.4359	0.5342	0.6200	0.6926	0.7507
$SSIM_{out}$ MM-PSO	0.4161	0.5357	0.6523	0.7330	0.7380
$SSIM_{out}$ MM-AIC	0.2642	0.3544	0.4516	0.5416	0.6082
$SSIM_{out}$ MM-Iterative	0.4302	0.5394	0.6575	0.7332	0.7900
$SSIM_{out}$ THOSVD	0.2946	0.3926	0.4996	0.6058	0.7039
$SSIM_{out}$ MWF	0.3488	0.4469	0.5435	0.6294	0.7116

Table 3.4: Baboon RGB image 256×256×3: $SSIM_{out}$ all methods

3.8.3 Multispectral image PaviaU Image 512×512×16

In this subsection we consider a multispectral image, of size 512×512×16. This image is processed with 2 wavelet packet decomposition levels along the spatial modes, leading to wavelet packet coefficients of size 128×128×16. Firstly, we exemplify the proposed and comparative methods with the case where the input $SNR_{in} = 5dB$. Secondly, we propose an exemplification with an input SNR 12.5 dB. Thirdly, we provide statistical results obtained with various values of input SNR.

3.8.3.1 Exemplification with $SNR_{in} = 5\ dB$

Some examples of result images obtained when processing the multispectral image PaviaU of size 512×512×16 are presented in Figs. 3.3, and 3.4 for the case where

the input SNR is 5 dB. Fig. 3.4 is a zoom on the upper right region of the image.

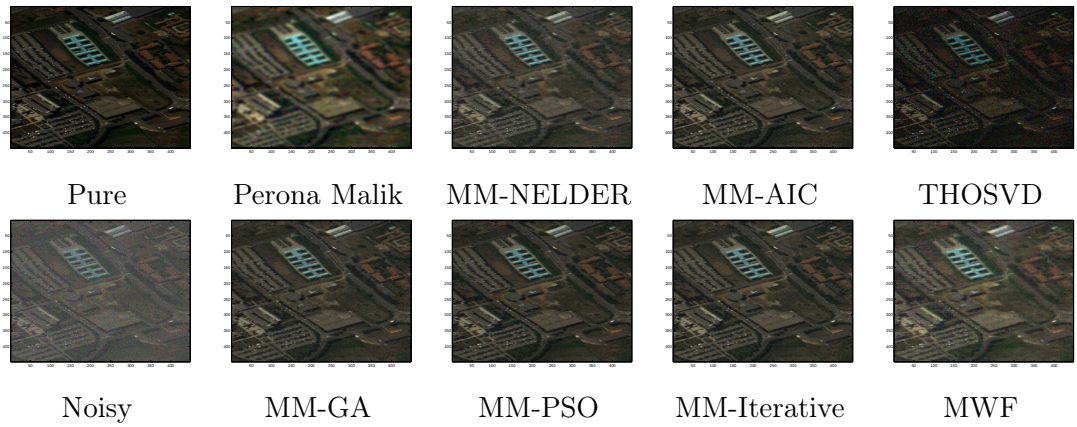


Figure 3.3: PaviaU image $512 \times 512 \times 16$: free noise, noisy 5 dB, denoising results

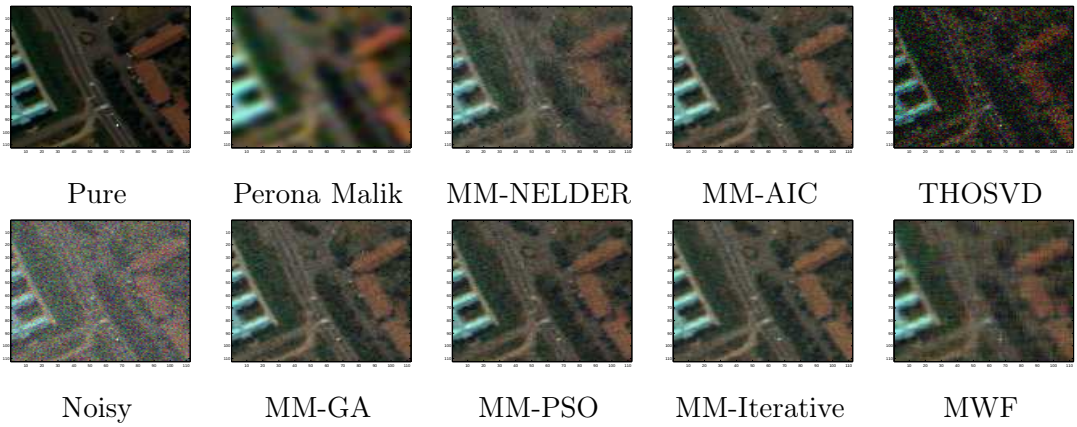


Figure 3.4: PaviaU image $512 \times 512 \times 16$, zoom: free noise, noisy 5 dB, denoising results

Table 3.5, which provides the processing time with an image containing 16 bands, shows that the proposed methods MM-GA, MM-PSO, and MM-Iterative are slower than MM-AIC.

Per-Mal	MM-GA	MM-Nelder	MM-PSO
17.83	196.5	52.03	48.12
MM-AIC	MM-Iterative	THOSVD	MWF
16.22	138.5	2.66	7.16

Table 3.5: PaviaU image $512 \times 512 \times 16$: Computational time for each method.

Table 3.6 presents results obtained regarding output SNR from the zoom image in Fig. 3.4, when a subset of bands is selected out of the 16 bands which compose the image. The number of selected bands varies from 12 to 3. The image is still impaired with an input SNR of 5 dB.

Nb Bands	SNR_{out} MM-PSO	SNR_{out} MM-AIC	SNR_{out} MM-Iterative
12	14.42	13.21	14.22
10	14.09	11.99	14.61
8	13.40	12.03	13.73
6	12.84	9.95	12.95
4	12.66	7.85	13.16
3	13.46	7.56	13.47

Table 3.6: PaviaU image, zoom , $SNR_{in} = 5dB$: SNR_{out} for various number of bands.

The numerical results in Table 3.6 show that the performances of MM-PSO and MM-Iterative hardly change when the number of bands is reduced. Conversely, the output SNR provided by MM-AIC sinks, when the number of bands is reduced from 12 to 3.

3.8.3.2 Exemplification with $SNR_{in} = 12,5\ dB$

The example in Fig. 3.5 concerns a case where the input SNR is 12.5 dB.

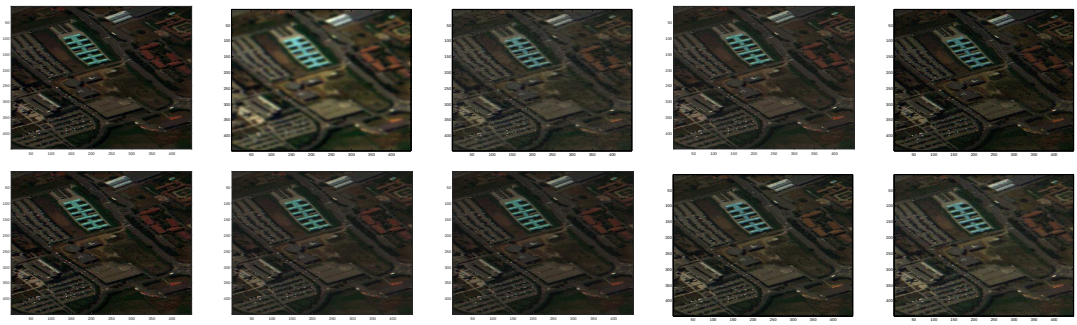


Figure 3.5: PaviaU image $512 \times 512 \times 16$: noise-free, noisy 12.5 dB, denoising results

3.8.3.3 Statistical results

These result images presented in this subsection yield the following comments: we notice that the result image provided by the Perona-Malik method is more blurred than all other result images. As for the methods which are not based on wavelet packet decomposition, the truncation of the HOSVD yields an image which is still rather noisy, and the MWF yields a column and row artifact and a poor detail preservation. As for the methods which include wavelet packet decomposition, the details are slightly better preserved when MM-GA, MM-PSO, and MM-Iterative are used than in the case where MM-AIC is used.

It is worth noticing that the performance of MM-GA, and MM-PSO are barely the same when the number of bands is reduced from 16 to 3. On the contrary, the performance of MM-AIC sinks when the number of bands decreases. This is coherent with the result presented in Table 3.3, for an input SNR of 5 dB, with the Baboon image which is composed of only 3 bands: MM-GA, and MM-PSO outperform MM-AIC in this case as well.

The numerical results in Tables 3.7 and 3.8 are by the visual aspect of the images obtained with input SNR values 5 and 12.5 dB: the output SNR and $SSIM$ values are higher when the MWPT-MWF-RE algorithm is used, at the expense of a higher computational load.

SNR_{in}	5 db	7,5 db	10 db	12,5 db	15 db
SNR_{out} Per-Ma	11.97	12.05	12.10	12.13	12.14
SNR_{out} MM-GA	15.38	17.41	18.91	20.54	21.82
SNR_{out} MM-Nelder	13.89	14.77	15.39	15.84	16.17
SNR_{out} MM-PSO	16.48	17.86	19.10	20.58	22.24
SNR_{out} MM-AIC	16.23	17.26	18.92	20.52	22.15
SNR_{out} MM-Iterative	16.40	18.00	19.24	20.79	22.29
SNR_{out} THOSVD	7.17	9.42	11.73	14.09	16.49
SNR_{out} MWF	13.47	15.00	16.57	18.18	19.84

Table 3.7: PaviaU $512 \times 512 \times 16$: SNR_{out} all methods

From these remarks, we infer that the proposed methods MM-PSO or MM-Iterative are preferable in applications where the processed multispectral image is composed of a low number of bands and the input SNR are rather low.

$SSIM_{in}$	0.48	0.55	0.65	0.74	0.80
$SSIM_{out}$ Per-Ma	0.6539	0.6713	0.6918	0.7040	0.7076
$SSIM_{out}$ MM-GA	0.7772	0.8219	0.8823	0.9253	0.9424
$SSIM_{out}$ MM-Nelder	0.7149	0.7449	0.7554	0.7664	0.7820
$SSIM_{out}$ MM-PSO	0.8149	0.8602	0.9047	0.9264	0.9499
$SSIM_{out}$ MM-AIC	0.8021	0.8687	0.8970	0.9396	0.9530
$SSIM_{out}$ MM-Iterative	0.8280	0.8398	0.9073	0.9360	0.9439
$SSIM_{out}$ THOSVD	0.6477	0.7406	0.8179	0.8774	0.9215
$SSIM_{out}$ MWF	0.6642	0.7397	0.7915	0.8637	0.8988

Table 3.8: PaviaU $512 \times 512 \times 16$: $SSIM_{out}$ all methods

3.8.4 Multispectral image PaviaU Image 128×128×16

In this subsection we consider a multispectral image, of size 128×128×16. This image is processed with 2 wavelet decomposition levels along the spatial modes, leading to wavelet coefficients of size 32×32×16. This image is processed with sub-sampling parameters $S_1 = S_2 = S_3 = 1$.

3.8.4.1 Exemplification with $SNR_{in} = 12.5\ dB$

The results obtained are exemplified in Fig. 3.6, where the image is impaired with $SNR_{in} = 12.5\ dB$.

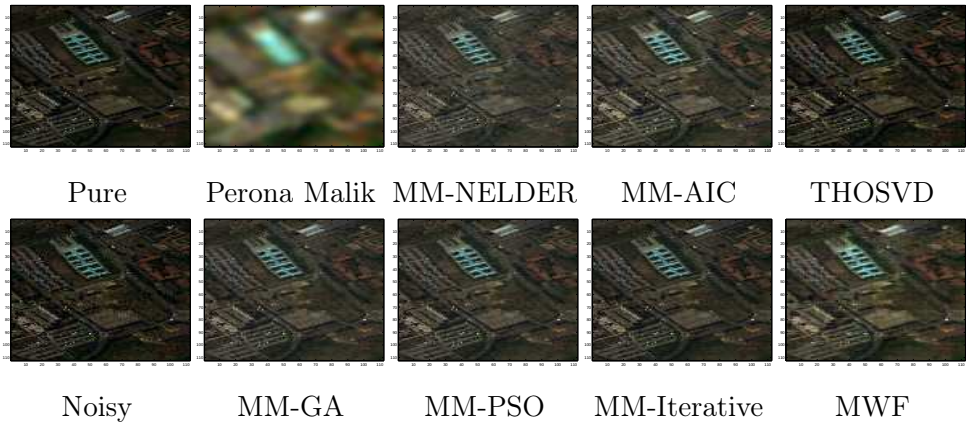


Figure 3.6: PaviaU image $128 \times 128 \times 16$: free noise, noisy 12.5 dB, denoising results.

3.8.4.2 Statistical results

As shown in Tables 3.9 and 3.10, for all input SNR values, the iterative method provides the best values of output SNR and output $SSIM$. As shown in Table 3.11, its computational load is about $20sec.$, higher than methods which are not based on multiway Wiener filtering of wavelet packet coefficients, but still 2 times lower than in the case where MM-GA is used. The interest of using PSO in the iterative method is to reach a good compromise between the improvement of the image quality, and the computational load. The MM-Nelder method yields output SNR and $SSIM$ values which are lower than MM-GA or MM-PSO. This can be due to the fact that, as three unknowns are expected, Nelder optimization method converges towards non-critical points. In the following, we will focus on MM-GA, MM-PSO, MM-AIC, and MM-Iterative methods.

SNR_{in}	5 db	7,5 db	10 db	12,5 db	15 db
SNR_{out} Per-Ma	8,66	8,69	8,72	8,73	8,74
SNR_{out} MM-GA	12,24	14,29	15,51	16,71	18,97
SNR_{out} MM-Nelder	10,60	11,20	11,75	12,03	12,24
SNR_{out} MM-PSO	12,65	14,36	15,40	17,44	18,95
SNR_{out} MM-AIC	13,40	15,13	17,01	18,92	20,91
SNR_{out} MM-Iterative	13,77	15,45	17,33	18,99	20,94
SNR_{out} THOSVD	6,10	8,32	10,61	12,93	15,31
SNR_{out} MWF	5,10	7,59	10,10	12,59	15,10

Table 3.9: PaviaU image $128 \times 128 \times 16$: SNR_{out} all methods

3.8.5 Subsampling and rank estimation

The purpose of this subsection is to evaluate the effect of subsampling on the estimation of the rank values. The multispectral image in Fig. 3.6 is a smaller version of the image in Fig. 3.5, with the same SNR of 12.5 dB. It has been obtained by subsampling along spatial modes by a factor 4, and keeping the same number of bands. We expect to obtain subspace rank values which are four times smaller than for the large image. Table 3.12 presents the values of signal subspace dimensions obtained from the image in Fig. 3.5, and Table 3.13 presents the values of signal subspace dimensions obtained from the image in Fig. 3.6.

$SSIM_{in}$	0,35	0,47	0,58	0,66	0,76
$SSIM_{out}$ Per-Ma	0,3015	0,3017	0,3171	0,3223	0,3237
$SSIM_{out}$ MM-GA	0,5959	0,7171	0,7741	0,8321	0,8891
$SSIM_{out}$ MM-Nelder	0,5179	0,5770	0,5988	0,6108	0,6351
$SSIM_{out}$ MM-PSO	0,6201	0,7233	0,7835	0,8391	0,8803
$SSIM_{out}$ MM-AIC	0,5773	0,7206	0,7612	0,8578	0,9077
$SSIM_{out}$ MM-Iterative	0,6603	0,7524	0,7978	0,8591	0,9150
$SSIM_{out}$ THOSVD	0,4508	0,5702	0,6697	0,7656	0,8389
$SSIM_{out}$ MWF	0,3554	0,4707	0,5822	0,6660	0,7621

Table 3.10: PaviaU image $128 \times 128 \times 16$: $SSIM_{out}$ all methods

Per-Mal	MM-GA	MM-Nelder	MM-PSO
1.446	39.94	10.11	6.880
MM-AIC	MM-Iterative	THOSVD	MWF
1.530	20.54	$8.99\ 10^{-2}$	$1.648\ 10^{-1}$

Table 3.11: PaviaU image $128 \times 128 \times 16$: Computational load all methods (in sec.)

Coefficient \ Method	MM-GA	MM-PSO	MM-AIC
Approximation	(116,119,14)	(113,114,14)	(43,43,6)
Detail	(78,68,12)	(73,72,12)	(1,128,16)

Table 3.12: PaviaU $512 \times 512 \times 16$: Estimated ranks values all methods

Table 3.12 presents the values of signal subspace dimensions obtained from the image in Fig. 3.5 with subsampling parameters $S1 = S2 = S3 = 1$.

Coefficient \ Method	MM-GA	MM-PSO	MM-AIC
Approximation	(29,29,15)	(28,30,14)	(32,32,13)
Detail	(15,20,9)	(20,20,2)	(32,32,9)

Table 3.13: PaviaU image $128 \times 128 \times 16$: Estimated ranks values all methods

Firstly, here is a comparative study of the rank values obtained with the large and with the smaller image.

When GA or PSO are used, comparing the values in Tables 3.13 and 3.12 yields the following comments: when considering the approximation coefficients, the values along rows and columns of signal subspace dimension are 3.8 to 4.1 times higher. For the detail coefficients, the ratio between dimension values is slightly farther from 4. This is coherent with the subsampling factor between the large and the small image, equal to 4. The slight bias with respect to the subsampling factor may be due to the fact that a different noise realization impairs the image. Additionally, we performed similar tests with the iterative method. When it is used, the estimated rank values increase throughout the iterations and reach almost always the size of the wavelet packet coefficient along each mode at the last iteration. Indeed, the noise magnitude of the processed image shrinks throughout the iterations, hence the increase in the values of signal subspace dimension. These results assess, at least for the considered images and subsampling factors, the relevance of step 3a in Algorithm 3.

Secondly, here are comments which aim at comparing the rank values in either approximation or detail coefficients: When referring to the results presented in 3.12 and 3.13, we notice that AIC always provides the maximum possible value, except once: the least possible value, that is, 1, is provided for a detail coefficient of the image in Fig. 3.4. The interest of this method is therefore limited in this case. When GA or PSO are used, the rank values for the approximation coefficient is more elevated than the rank values for the detail coefficient. This is due to the ability of wavelet packet decomposition to concentrate the high frequency components, that is, the noise, in the detail coefficients. The noise magnitude is less elevated in the approximation coefficient, and a larger proportion of singular vectors are considered as forming part of the signal subspace. So the value of the signal subspace dimension is more elevated for the approximation coefficient than for the detail coefficient.

3.8.6 Multispectral image PaviaU Image 128×128×32

The results presented above have shown the interest of using MM-PSO or MM-Iterative to reach elevated output SNR values when the number of bands in the processed multispectral image is rather low. As their computational load is more elevated than MM-AIC, we wish to provide an alternative for the case where the multispectral image is composed of a high number of bands. We aim at demonstrating the interest of fixed point algorithm when the processed image is large, and impaired with a low input SNR of 5 dB. The visual results obtained are presented in Fig. 3.7: we display the noise-free image, the noisy image impaired with a 5 dB white Gaussian noise, and the denoising results obtained with MM-PSO and MWF using either singular value decomposition (SVD) or fixed point (FP) algorithm.

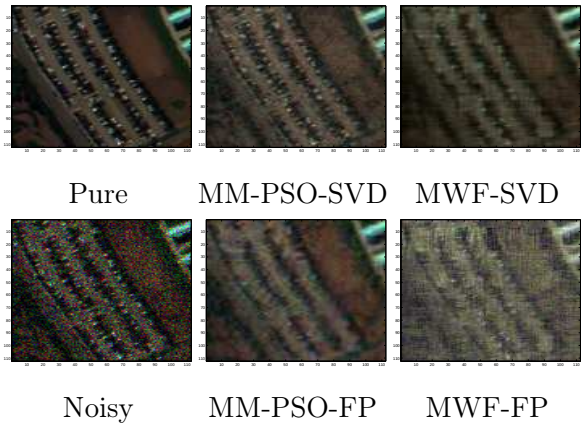


Figure 3.7: PaviaU image 128 × 128 × 32: pure, noisy 5 dB, results with PSO and MWF with either SVD or Fixed Point.

Table 3.14 provides the rank values estimated by GA, PSO and AIC. It shows that, compared to the case when the input SNR is 12.5 dB (see Table 3.13), the estimated rank values are smaller, even much smaller for the detail coefficients. Table 3.15 presents the computational loads obtained with GA, PSO, the iterative method, and MWF when either SVD, or FP, are used. The computational time required by AIC in the same conditions is 2.77 s.

Coefficient \ Method	MM-GA	MM-PSO	MM-AIC
Approximation	(26,26,18)	(16,17,11)	(24,32,15)
Detail	(4,4,4)	(13,8,5)	(1,32,12)

Table 3.14: PaviaU image 128 × 128 × 32: Estimated ranks values all methods

	MM-GA	MM-PSO	MM-Iterative	MWF
svd	262.16	54.47	259.48	0.29
Fixed point	258.92	29.55	129.76	0.23

Table 3.15: 'PaviaU' multispectral image 128×128×32: Computational time for each method with either svd or fixed point.

Table 3.16 presents numerical results obtained with MM-PSO when either SVD or FP are used, and with MM-AIC.

	MM-PSO-SVD	MM-PSO-FP	MM-AIC
SNR_{out}	13.14	12.58	12.17
SSIM_{out}	0.62	0.56	0.58

Table 3.16: PaviaU image $128 \times 128 \times 16$, noisy 5 dB: SNR_{out} and SSIM_{out} obtained by MM-PSO, with SVD or Fixed Point

In the conditions presented above, we notice that for this image fixed point algorithm divides the computational load by almost 2 when MM-PSO is used (see Table 3.15), at the expense of slightly smaller output SNR and $SSIM$ values (see Table 3.16). From Fig. 3.7, we infer that using FP instead of SVD degrades the details. The output SNR value is still higher than when MM-AIC is used.

3.8.7 Performance and number of bands

In this section, we study the performance of MM-AIC and MM-PSO for multispectral images with various values for the number of bands. For this, we have selected CAVE database of multispectral images [53]. This database is composed of images of size 512×512×31. That is, they contain 31 bands, regularly obtained from the wavelength range 400 to 700 nm, with a 10 nm step between each band.
We considered the image called "jelly beans" because it contains lots of small features and details to be preserved. We exemplify the MM-AIC and MM-PSO with two images impaired with random noise in such a way that $SNR_{in} = 2dB$. The first (resp. second) image contains 5 (resp. 31) bands regularly spaced between 400 and 700 nm.

3.8.7.1 Exemplification with $SNR_{in} = 2dB$ and 5 Bands

The image presented in Fig. 3.8 contains 5 bands. The MM-AIC method provided $SNR_{out} = 5,0dB$, and the MM-PSO method provided $SNR_{out} = 8,8dB$.

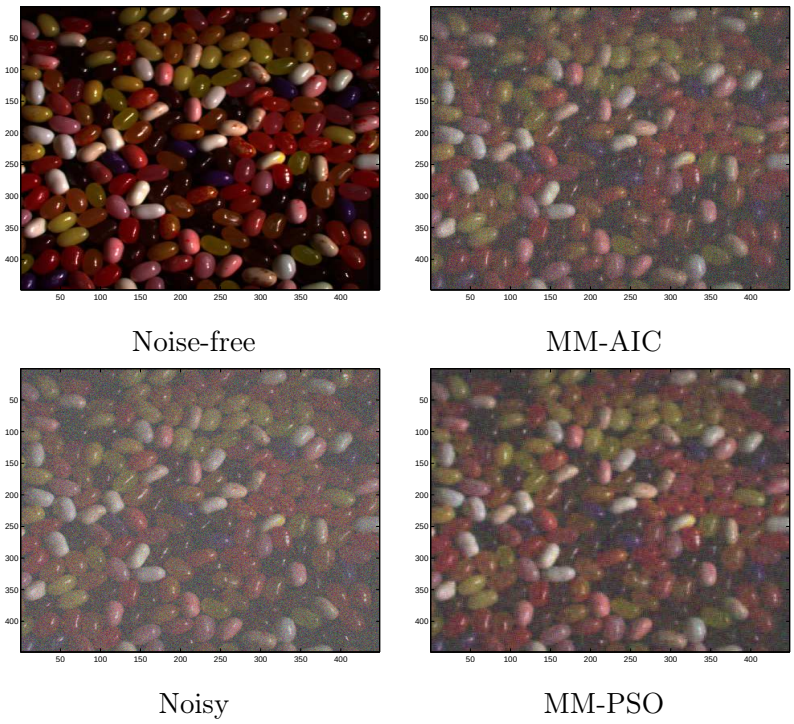


Figure 3.8: "Jelly bean" multispectral image, 5 bands: noise-free and noisy images; result obtained by MM-AIC and MM-PSO

3.8.7.2 Exemplification with $SNR_{in} = 2dB$ and 31 Bands

The image presented in Fig. 3.9 contains 31 bands. The MM-AIC method provided $SNR_{out} = 21,7dB$, and the MM-PSO method provided $SNR_{out} = 24,5dB$.

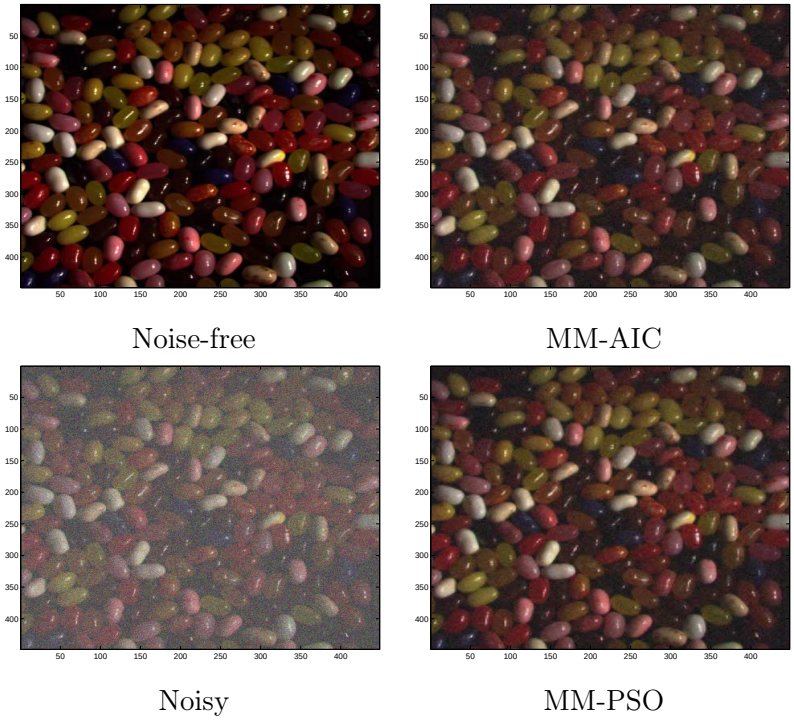


Figure 3.9: "Jelly bean" multispectral image, 31 bands: noise-free and noisy images; result obtained by MM-AIC and MM-PSO

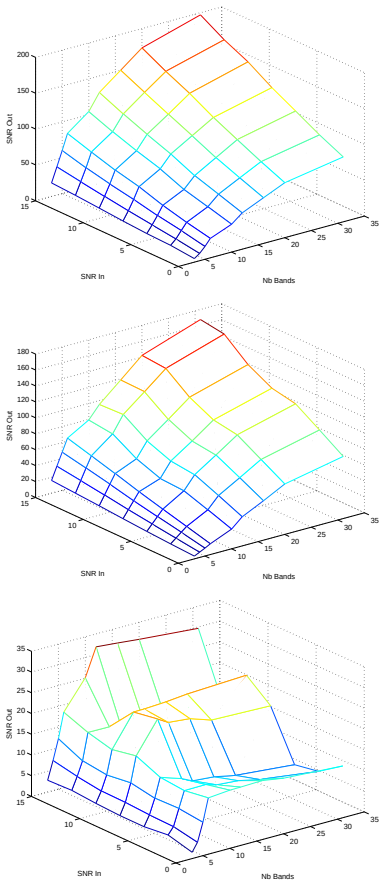
As comments to Figs. 3.8 and 3.9, we first can notice that the higher the number of bands, the better the aspect of the denoised image, with either MM-AIC and MM-PSO. Secondly, we notice that the result obtained by MM-PSO in terms of output SNR is better than the result obtained with MM-AIC. The difference is slightly higher (3,8 dB) when a low number of bands (5) is processed. When the whole "jelly beans" image is processed, with all 31 bands, the difference is 2,8 dB. As stated in 3.7.3 for another multispectral image, the proposed MM-PSO method behaves better than MM-AIC for multispectral images with a low number of bands. Indeed, AIC is a statistical criterion whose performances are undoubtedly improved when a large amount of data are available.

3.8.7.3 Results for various values of input SNR and number of bands

We now present results obtained with the following values of input SNR: 0, 2.5, 5, 7.5, 10, 12.5, and 15 dB, and the following values for the number of bands: 3, 4, 5,

6, 10, 12, 16, 20, and 31. These bands are selected out of the 31 bands of the "Jelly bean" multispectral image [53], and regularly spaced between 400 and 700 nm. For instance, when 3 bands are selected, they correspond to 400 nm, 550 nm, and 700 nm. When 31 bands are selected, all bands between 400 and 700 nm with a 10 nm spacing are included in the processed image.

Fig. 3.10 presents the results obtained in terms of output SNR for all couples of input SNR and number of bands: Fig. 3.10 a) shows the results obtained with MM-PSO, Fig. 3.10 b) the results obtained with MM-AIC, and Fig. 3.10 c) shows the difference between the values obtained with MM-PSO and MM-AIC. For both methods, the larger the number of bands, the better the result obtained.



a) b) c)

Figure 3.10: "Output SNR for various values of input SNR and number of bands: a) MM-PSO, b)MM-AIC; c) Difference MM-PSO - MM-AIC

We notice that the difference is always positive, which means that the performance of MM-PSO is better in all cases. The largest difference is obtained for input SNR values between 7.5 and 15 dB, and for the number of bands between 6 and 25.

3.8.8 Plant leaf fluorescence image

We now consider plant leaf images. While illuminated with a 400 nm light source, a plant leaf emits fluorescence light of little energy, in a wavelength range between 600 and 800 nm. Because of the low brightness of the light emitted through the fluorescence phenomenon, the gain of the imaging sensors are pushed to a high value, and the exposure time must be elevated to capture this phenomenon. The electronic noise is then of large magnitude, and the acquired images are noisy.
The image acquisition setup that was used is as follows: a hyperspectral camera HySpex VNIR-1600 (Norsk Elektro Optikk, Norway) acquired successive lines of 1600 pixels and 160 spectral bands ranging from 415 to 994 nm with a 3.7 nm spectral sampling interval. The lighting was provided by halogen source positioned close to the cameras. The illumination zenith angles were set to $\theta = 20°$. A monochromator is placed in the incoming halogen irradiance between the leaf and the Halogen. This monochromator permits to select the wavelength which induces the fluorescence phenomenon. Fig. 3.11 presents, for a relevant pixel of the acquired scene, the spectrum acquired by the camera. It focuses on the relevant part of the spectrum, corresponding to the fluorescence light, located between 600 and 800 nm.

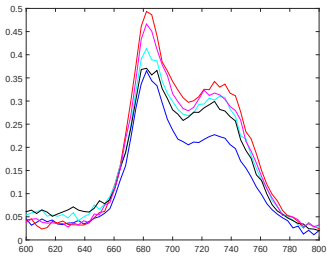


Figure 3.11: Leaf fluorescence, wavelength range of interest

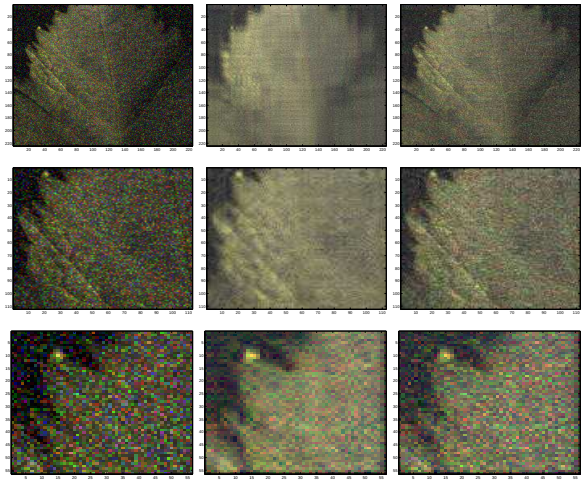
If the time allowed for the acquisition of each frame is lowered, the electronic noise increases with respect to the relevant part of the data. The signal to noise ratio sinks, hence the interest of denoising.

Fig. 3.12 shows the image obtained in the best conditions, with an elevated power for the incident light, and 50 ms of exposure time. We display a subimage of size 256×256 in Fig. 3.12 a), and two other subimages of size 12×128 and 64×64 in Figs. 3.12 b) and 3.12 c).

a) b) c)

Figure 3.12: Leaf fluorescence image: a) large image; b) zoom 1; c) zoom 2.

Figs. 3.13, 3.14, 3.15, and 3.16 correspond to different values of exposure time between 50 ms and 250 ms, and incident power of the excitation light, for the same leaf. For a lower power of the excitation light, the setup is cheaper, but the noise level is higher.

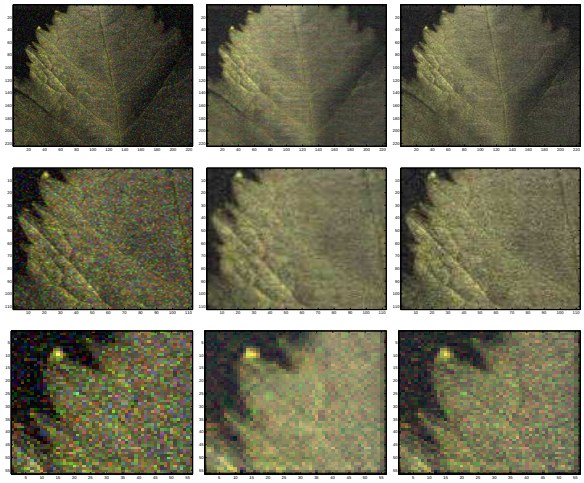


a) b) c)

Figure 3.13: Leaf fluorescence image: a) raw acquisition; b) MM-PSO; c) MM-Iterative.

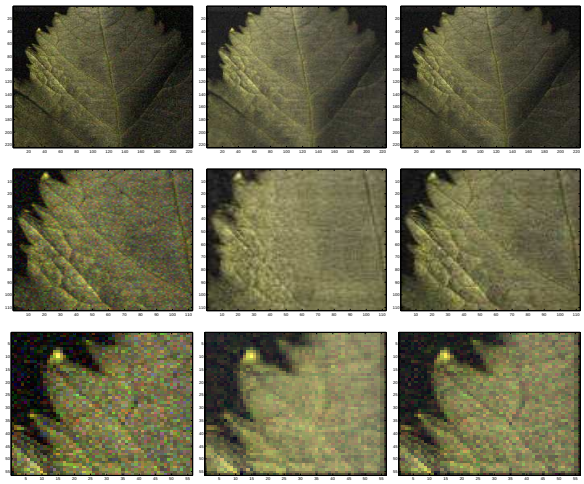
To process these images, we select 16 wavelengths between 677 nm and 740 nm. For the representation of the acquired and processed images, we select 677, 681, and 726 nm for the R, G, and B bands.

To evaluate the denoising results quantitatively, we consider the images in Fig. 3.12 as noise-free. With respect to these images, the SNR of the raw images in Figs. 3.13, 3.14, 3.15, and 3.16 is 0.5, 5.6, 10.2, and 15.5 dB. The MM-PSO method yields 4.0, 8.5, 12.5, and 15.1 dB as output SNR. The MM-iterative method yields 3.6, 8.3, 12.6, and 16.3 as output SNR. The computational time for the largest image of size $256 \times 256 \times 16$ is of 13.8 sec for MM-PSO, and 33.6 sec for MM-Iterative. As for comparison, with an exposure time of 50 ms, the acquisition time required for such an image is of several minutes. We can notice that the results obtained with MM-PSO are more blurred and noisy than when MM-Iterative is applied. Indeed, MWF



a) b) c)

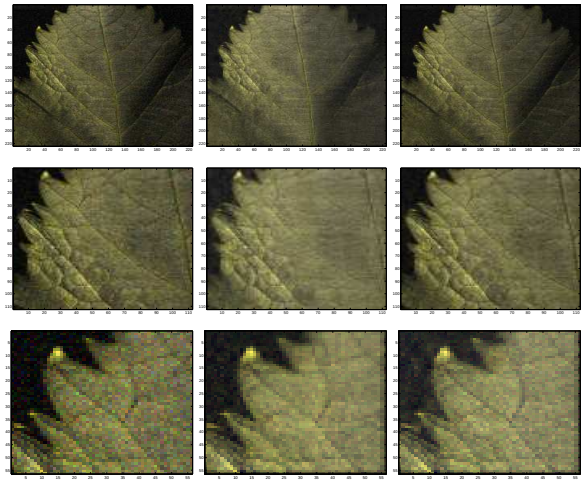
Figure 3.14: Leaf fluorescence image: a) raw acquisition; b) MM-PSO; c) MM-Iterative.



a) b) c)

Figure 3.15: Leaf fluorescence image: a) raw acquisition; b) MM-PSO; c) MM-Iterative.

provides the first gross estimate. It is blurred, with low noise magnitude. When MM-PSO is run once, a denoised image is obtained where the contours are better preserved. It is used a refined reference in MM-Iterative. To get close to this new reference, the PSO algorithm converges to triplet of rank values K_1, K_2, and K_3 which are more elevated than when MM-PSO is used. That is why the result of MM-Iterative appears more noisy but less blurred than the result of MM-PSO. From a qualitative point of view, our methods for denoising and detail preservation remove part of the noise while preserving the contours, that is, the frontier between ribs and limb of the leaf.



a) b) c)

Figure 3.16: Leaf fluorescence image: a) raw acquisition; b) MM-PSO; c) MM-Iterative.

3.9 Automatic rank estimation of Parafac decomposition and application

3.9.1 Introduction

In the frame of multidimensional data denoising, tensor methods were proposed to take into account the relationships between dimensions, as opposed to slice-by-slice methods [21]. For instance, Multiway Wiener Filtering (MWF) has been derived from the minimization of a mean square error criterion between the expected noise-free tensor and the denoised tensor [21]. Since this seminal work, some extensions have been proposed, for the minimum rank approximation of matrices [146] or tensors, based either on Tucker decomposition [85] or on Parafac decomposition [92]. It has been shown that the truncation of the Parafac decomposition permits to minimize the square error (SE) between the 'raw-data' tensor and the estimated tensor [92]. Another field of applied mathematics, Wavelets [22], has provided efficient algorithms for image denoising. For instance, ForWaRD method [107] performs image deconvolution and denoising in the Fourier domain, and refined image denoising in the wavelet domain.

Relation with previous work in the field Owing to the ability of the wavelet transform to separate high frequency components from low frequency ones, wavelets are privileged tools for image denoising [132]. That is why, recently, the wavelets and tensor framework got recently closer to each other: the multiway Wiener filtering, based on the Tucker decomposition, has been introduced in a wavelet frame-

work to preserve small targets while denoising hyperspectral images [85]. This method, called MWF-MWPT (multiway Wiener filtering-multidimensional wavelet packet transform), preserves well the spatial details, but exhibits a drawback: the elevated number of parameters to estimate, that is, for each wavelet decomposition level, and for each tensor mode, the adequate signal subspace rank value. The truncation of the Parafac decomposition exhibits good performances in reducing low amplitude, possibly signal dependent noise [92], and there is only one parameter to estimate: a single rank for the truncation. However, it has never been introduced in a wavelet framework. For rank estimation, DIFFIT [133], Convex Hull [9], and Minimum Description Length [86] have much merit, especially for SNR (signal to noise ratio) values higher than 10 dB, but results seem to be restricted to artificially generated data.

Goal and contributions In this work, our objectives are as follows: reduce the low-amplitude noise in multispectral images, with a low number of parameters and their automatic tuning. For this, we focus on the truncation of the Parafac decomposition. The originality of our work relies on the automatic estimation of the rank for the truncation of the Parafac decomposition, and on the introduction of the Parafac decomposition in the wavelet framework.

Outline Section 3.9.2 sets the problem of the Parafac rank estimation. In section 3.9.3 we propose a criterion to estimate the rank in an optimal sense, and the Nelder-Mead method [69] to optimize it; in section 3.9.4, the truncation of Parafac decomposition is set in a wavelet framework. Section 3.9.5 presents a flowchart of the proposed method; in section 3.9.6, multispectral image denoising results are obtained with the proposed method and comparative ones [55, 107].

3.9.2 Problem setting

3.9.2.1 Truncation of the parafac decomposition: overview

We model a noisy multispectral image as a tensor resulting from a multidimensional signal $\mathcal{X}$ impaired by an additive white noise $\mathcal{N}$ [21, 113]. The multispectral image can be expressed as :

$$\mathcal{R} = \mathcal{X} + \mathcal{N}. \tag{3.7}$$

Tensors $\mathcal{R}$, $\mathcal{X}$, and $\mathcal{N}$ are of size $I_1 \times I_2 \times I_3$. For each spectral band indexed by $i = 1, \ldots, I_3$, the noise $\mathbf{N}(:,:,i)$ is assumed stationary zero-mean. We aim at estimating the desired tensor signal $\mathcal{X}$ from data tensor $\mathcal{R}$. The PARAFAC model factorizes a tensor into a sum of rank-1 tensors [62]: a tensor $\mathcal{R} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ can be

expressed as

$$\mathcal{R} = \sum_{k=1}^{T} \mathcal{R}_k = \sum_{k=1}^{T} \lambda_k \mathbf{a}_k^{(1)} \circ \mathbf{a}_k^{(2)} \circ \mathbf{a}_k^{(3)} \tag{3.8}$$

$\mathcal{R}_k \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is rank-1 tensor; $\mathbf{a}_k^{(1)}, \mathbf{a}_k^{(2)}, \mathbf{a}_k^{(3)} \in \mathbb{R}^{I_n}$ are normalized vectors of the n-mode space of $\mathcal{R}$ normalized by $\mathbf{a}_k^{(n)} = \mathbf{a}_k^{(n)} / \|\mathbf{a}_k^{(n)}\|$, $n = 1, 2, 3$; and $\lambda_k = \|\mathbf{a}_k^{(1)}\| \cdot \|\mathbf{a}_k^{(2)}\| \cdot \|\mathbf{a}_k^{(3)}\|$, $k = 1, 2, \cdots, T$.

It has been proved in [91] that truncating the PARAFAC decomposition to the K terms weighted by the largest values of λ_k yields the minimum square error SE$=$ $\|\mathcal{R} - \hat{\mathcal{X}}\|^2$ between raw tensor $\mathcal{R}$ and estimate $\hat{\mathcal{X}}$. Assuming without loss of generality that the values λ_k are correctly ordered, the truncation of the Parafac decomposition consists in selecting the K first terms in Eq. (3.9), so that we obtain the estimate:

$$\hat{\mathcal{X}} = \sum_{k=1}^{K} \mathcal{R}_k = \sum_{k=1}^{K} \lambda_k \mathbf{a}_k^{(1)} \circ \mathbf{a}_k^{(2)} \circ \mathbf{a}_k^{(3)} \tag{3.9}$$

where K is the rank, $\hat{\mathcal{X}}$ is the truncation of the Parafac decomposition of $\mathcal{R}$, also called rank-K approximation of $\mathcal{R}$. As shown in [85], $\hat{\mathcal{X}}$ is a close estimate of the noise-free tensor $\mathcal{X}$, especially when the noise magnitude is low.

3.9.2.2 The rank estimation issue

In [92], an iterative 'brute-force' method is proposed to obtain a guess on a convenient Parafac truncation rank. the covariance matrix of the residual noise is computed, and the non-diagonal coefficients are summated. As it should tend to 0 when only decorrelated noise is present, a threshold is set empirically on this summation. Unfortunately, the optimality of this method as a function of any criterion is not proved. For these reasons, we propose a criterion to minimize, and an adequate optimization algorithm, to estimate the rank of the truncation which is optimal in the sense of this criterion.

3.9.3 Nelder-Mead optimization method for the estimation of the Parafac truncation rank

For the first time, we propose a method to estimate an optimal rank for the truncation of the Parafac decomposition. For this, firstly, we propose a squared error criterion; secondly, we justify the use of Nelder-Mead optimization method to minimize this criterion.

3.9.3.1 Proposed criterion

The proposed criterion is as follows:

$$J(K) = ||\mathcal{X}_1 - \hat{\mathcal{X}}||^2, \tag{3.10}$$

where $||.||$ represents the Frobenius norm, $\mathcal{X}_1$ is a first, gross estimate of the expected tensor $\mathcal{X}$ or the noisy tensor $\mathcal{R}$ itself, and $\hat{\mathcal{X}}$ its final estimate. It is worth noticing that we remain close to the framework of the Parafac decomposition, while choosing the SE as a criterion. Now we wish to solve:

$$\hat{K} = \underset{K}{argmin}(J(K)) \tag{3.11}$$

It is rather complex to minimize the criterion J as it is a nonlinear function of the parameter K. We wish to estimate a single value -the rank for the truncation of the Parafac decomposition-, but the optimization method used for this purpose must be global. Moreover, the multispectral image considered in this work exhibit a large number of voxels. Therefore, the proposed optimization method must be fast.

3.9.3.2 Why the Nelder-Mead method ?

Some methods from the literature may be available to perform the minimization in Eq. (3.11), but they exhibit some drawbacks: the DIRECT method [52], for instance, requires the objective function to be Lipschitzian. Particle swarm optimization (PSO) [55] provides the global minimum of a scalar function of several variables but is not assumed, to the best of our knowledge, to be particularly fast.
The Nelder-Mead Simplex method [69] is a global optimization method, which is meant to minimize a scalar-valued nonlinear function of several real variables, without any derivative information. It is known to yield a rapid decrease in cost function values [69]. It has been shown that, in dimension two, the Nelder-Mead method may converge to a non-critical point of the minimized function [36]. However, as specified in [69], the global convergence of the Nelder-Mead method is ensured in a one-dimensional problem, which is the case in the current work. For these reasons, we select the Nelder-Mead Simplex Method.
Nelder-Mead algorithm falls into five steps: ordering, reflection, expansion, contraction, and shrinkage. At each iteration, a simplex is modified, following these five steps. Four scalar parameters must be specified to run the Nelder-Mead method: coefficients of reflection (ρ), expansion (χ), contraction (γ), and shrinkage (σ). They are such that $\rho > 0$, $\chi > 1$, $\chi > \rho$, $0 < \gamma < 1$, $0 < \sigma < 1$. The proposed improved

truncation of the Parafac decomposition, with an automatic estimation of the rank is now introduced in a wavelet framework.

3.9.4 Parafac truncation of the multidimensional wavelet packet transform coefficients

After the MWPT, abundant and rare features can be separated into different components, therefore the 'useful part' of each component can be estimated more accurately than using the entire dataset [85]. Furthermore, a better estimation of the 'useful part' can improve the performance of the Parafac rank-K truncation of each component. We propose a combination, named Parafac-MWPT. This subsection proves the ability of Parafac-MWPT to minimize the SE between $\mathcal{R}$ and $\hat{\mathcal{X}}$. By performing multidimensional wavelet transform to tensor $\mathcal{R}$, $\mathcal{X}$ and $\mathcal{N}$, we obtain:

$$\begin{aligned}
&\mathcal{R} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \\
&= (\mathcal{X} + \mathcal{N}) \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \\
&= \mathcal{X} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \\
&+ \mathcal{N} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3
\end{aligned} \tag{3.12}$$

where $\times_n$ denotes the n-mode product for each mode n: wavelet filtering is performed successively after flattening tensor $\mathcal{R}$ along each mode. The coefficient tensor of each part:

$$\mathcal{C}_{\mathbf{l}}^{\mathcal{R}} = \mathcal{R} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \tag{3.13}$$

$$\mathcal{C}_{\mathbf{l}}^{\mathcal{X}} = \mathcal{X} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \tag{3.14}$$

$$\mathcal{C}_{\mathbf{l}}^{\mathcal{N}} = \mathcal{N} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \tag{3.15}$$

and the coefficient tensor of the estimate $\hat{\mathcal{X}}$:

$$\hat{\mathcal{C}}_{\mathbf{l}}^{\mathcal{X}} = \hat{\mathcal{X}} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3 \tag{3.16}$$

With the extraction process proposed in [85], we obtain the components of each frequency $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}}$, $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}}$ and $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{N}}$ from $\mathcal{C}_{\mathbf{l}}^{\mathcal{R}}$, $\mathcal{C}_{\mathbf{l}}^{\mathcal{X}}$ and $\mathcal{C}_{\mathbf{l}}^{\mathcal{N}}$ respectively. We obtain:

$$\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}} = \mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}} + \mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{N}} \tag{3.17}$$

From Parseval's theorem, the following expression can be obtained:

$$\|\mathcal{R} - \hat{\mathcal{X}}\|^2 = \|\mathcal{C}_{\mathbf{l}}^{\mathcal{R}} - \hat{\mathcal{C}}_{\mathbf{l}}^{\mathcal{X}}\|^2 = \sum_{\mathbf{m}} \|\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}} - \hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}}\|^2 \tag{3.18}$$

which means that minimizing the SE between $\mathcal{R}$ and its estimate $\hat{\mathcal{X}}$ is equivalent to minimizing the SE between $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}}$ and $\hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}}$ for each $\mathbf{m}$. If we apply Nelder-Mead

optimization method while performing the truncation of the Parafac decomposition (see section 3.9.3) to estimate the rank K, we get the optimal rank for the Parafac truncation of $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$:

$$\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}} = \sum_{k=1}^{K} \mathcal{C}^{\mathcal{R},k}_{\mathbf{l},\mathbf{m}} \tag{3.19}$$

After estimating $\hat{\mathcal{C}}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$ for each $\mathbf{m}$, we obtain $\hat{\mathcal{C}}^{\mathcal{R}}_{\mathbf{l}}$ by concatenating $\hat{\mathcal{C}}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$. Furthermore, the estimate $\hat{\mathcal{X}}$ can be obtained by inverse MWPT:

$$\hat{\mathcal{X}} = \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l}} \times_1 \mathbf{W}_1^T \times_2 \mathbf{W}_2^T \times_3 \mathbf{W}_3^T \tag{3.20}$$

The process composed of wavelet transform, Parafac truncation, and inverse wavelet transform yields an estimate $\hat{\mathcal{X}}$ of the expected tensor. In Eq. (3.10), we notice that the first reference tensor $\mathcal{X}_1$ which is used in the criterion $J(K)$ is the noisy tensor $\mathcal{R}$ itself. The wavelet coefficient tensor $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$ is supposed to be noisy, which is not convenient for a reference. Therefore, we introduce the denoising process in a loop: Let $\hat{\mathcal{X}}_r$ be the estimated tensor at iteration 'r'. At iteration 'r+1', the new reference becomes tensor $\hat{\mathcal{X}}_r$. The loop is stopped at a given iteration 'r' when $||\hat{\mathcal{X}}_r - \hat{\mathcal{X}}_{r-1}|| < \delta$ where δ is an *a priori* set very low threshold.

3.9.5 Flowchart of the algorithm

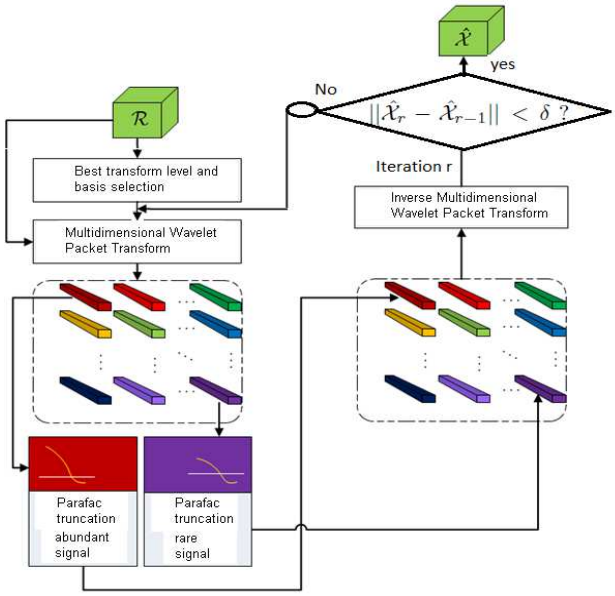


Figure 3.17: Truncation of the Parafac decomposition in wavelet packet transform domain: iterative process

Method \ Criterion	SNR	PSNR	SSIM
Noisy image	10.0	23.7	0.68
ForWaRD	$2.4\ 10^{-2}$	13.6	0.75
Parafac-MWPT:			
• Nelder-Mead	12.8	26.4	0.74
• PSO	13.1	26.7	0.73

Table 3.17: Numerical criteria obtained with the proposed method based on Parafac decomposition, with a rank estimation by Nelder-Mead or PSO optimization algorithms; and ForWaRD algorithm.

3.9.6 Results

In this section, we apply the proposed method based on the truncation of the Parafac decomposition with automatic estimation of the rank by Nelder-Mead and multidimensional wavelet packet to multispectral image denoising. We compare the results obtained with the comparative ForWaRD method [107] on a real-world multispectral image. Programmes were written in Matlab®, and executed on a PC computer running Windows, with a 3GHz double core and 3GB RAM. The multispectral image is artificially impaired with white, identically distributed random noise. This image is of size $128 \times 128 \times 16$: it includes 128 columns, 128 rows and 16 spectral bands. Before impairing the image, we artificially fix the Parafac decomposition rank in the original noise-free image by applying the truncation of the Parafac decomposition with a known rank, $K = 40$. In the proposed method, multidimensional wavelet packet is implemented with Daubechies wavelets and two decomposition. This has been empirically shown to yield the best results for this type of data [85]. This holds also for the comparative ForWaRD algorithm [107], applied successively slice-by-slice to the image. Nelder-Mead and PSO require respectively 200 and 2000 iterations for convergence; the parameters in Nelder-Mead are chosen as $\rho = 1$, $\chi = 2$, $\gamma = 1/2$, $\sigma = 1/2$ [69]. We initialize the rank value K to $\min(I_1,\ I_2,\ I_3)$. The numerical results are computed from images truncated to the size $120 \times 120 \times 16$ to avoid the border issues. The parameter δ is set to 10^{-6}. Table 3.17 presents the results obtained in terms of SNR (signal to noise ratio), PSNR (peak signal to noise ratio), and slice-by-slice SSIM (structural similarity) [144] with the proposed and ForWaRD methods. The rank of the Parafac decomposition is estimated either with Nelder-Mead or with PSO. The number of particles in PSO is such that the results in

terms of SNR are approximately the same than when Nelder-Mead is used. Examples of estimated rank values are as follows: for the first decomposition level PSO provides $K = 41$, and Nelder-Mead provides $K = 39$. Table 3.17 shows that the rank value provided by PSO and Nealder-Mead yield close SNR and PSNR values, which are higher than the comparative ForWaRD method. However, one iteration of PSO lasts $8\ 10^{-2}$ sec. to get this estimation, while Nelder-Mead lasts $4\ 10^{-2}$ sec. Figure 3.18 displays visual results. When the truncation of the Parafac decomposition is used, we display the Nelder-Mead result: the visual aspect of the PSO result is very similar.

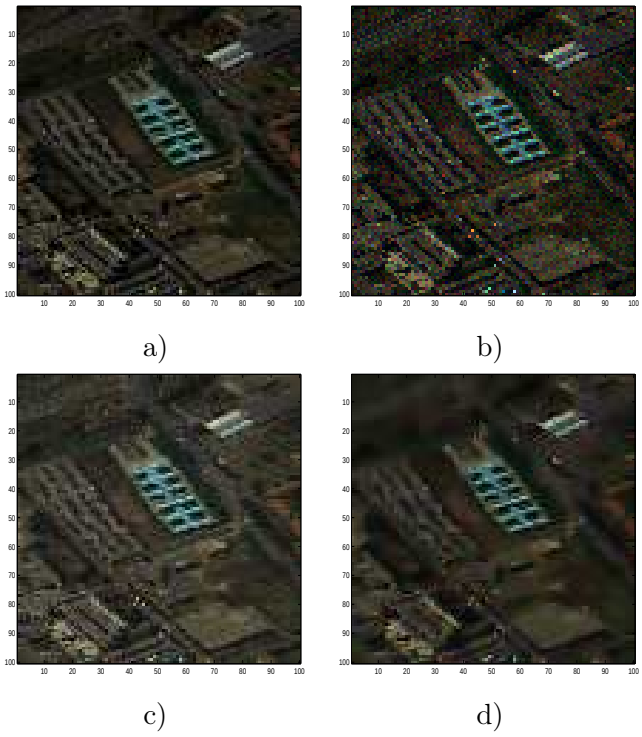


Figure 3.18: a) noise-free multispectral image; b) noisy multispectral image; c) proposed denoising method, using Nelder-Mead; d) ForWaRD.

From the results displayed in Fig. 3.18, we can assert that when the truncation of the Parafac decomposition and ForWaRD are used, the overall visual aspect of the image is well preserved, which is confirmed by the SSIM values in Table 3.17. However, the details are better preserved when the truncation of the Parafac decomposition is used, which is confirmed by the SNR and PSNR values in Table 3.17.

We propose a least squares criterion to get an optimal evaluation of the optimal rank, with respect to this criterion, of the truncation of the Parafac decomposition; secondly, we insert the truncation of the Parafac decomposition in a wavelet framework, for the purpose of multidimensional data denoising. We illustrate the ability

of the proposed method to remove low-magnitude noise in an application to multispectral image denoising. Our method based on Parafac decomposition and wavelet decomposition performs better than the wavelet-based ForWaRD method in terms of SNR and $PSNR$. It preserves the details, while preserving the visual aspect evaluated by the SSIM criterion. Also, the Nelder-Mead method is relevant for the truncation of the Parafac decomposition: in the considered application case, it is two times faster than particle swarm optimization though yielding similar results. Future works could consist in testing the behavior of the proposed method in the presence of signal-dependent noise.

3.10 Conclusion

In this work, an algorithm named MWPT-MWF-RE has been proposed to estimate multiple values of signal subspace dimension, to apply multiway Wiener filtering in a wavelet packet framework. The final purpose of this method is to remove noise from multidimensional images. The main advances brought by our thesis are as follows: Firstly, we propose a criterion depending on signal subspace ranks, and adapt an optimization algorithm to minimize this criterion. Mainly, two bio-inspired optimization algorithms are compared: a genetic algorithm (GA) and particle swarm optimization (PSO). As PSO offers the best compromise between computational load and denoising quality, we insert it in an iterative algorithm. In the iterative algorithm, the raw estimate required in the criterion to be minimized is the estimate provided by MM-PSO obtained at the previous step. Still, the number of decomposition levels should be estimated, and the computational load of the overall algorithm should be reduced. That is why, secondly, we propose to estimate the number of decomposition levels and the adequate values of signal subspace ranks on a subsampled version of the data: we performed tests on various types of images and concluded that, at least for the considered data, the values of the optimal signal subspace dimension in terms of denoising accuracy are divided by the subsampling factor when such a subsampling is applied to the image. Thirdly, we evaluate the performance of the proposed MWPT-MWF-RE algorithm on various multidimensional images: the RGB Baboon image, and various multispectral images extracted from the well-known Pavia University hyperspectral image. We considered an application of plant fluorescence imaging. The emitted light in this context is of very low intensity, yielding images which are impaired with rather low SNR values. A comparative study was performed involving three main types of algorithms: firstly,

the diffusion-based Perona-Malik method; secondly, the truncation of HOSVD and MWF; thirdly, a method based on wavelet packet decomposition and MWF, where the dimensions of signal subspace are estimated with AIC. From the analysis and the comparative study of other similar methods, it can be concluded that an iterative algorithm with automatic rank estimation with particle swarm optimization offers a good compromise between computational load, SNR, and SSIM improvement. The proposed method performs best, compared to AIC, when the noise magnitude is elevated and the number of bands is between 6 and 25. As well as the truncation of the Tucker3 decomposition, on which the MWF is based, we have considered the truncation of the Parafac decomposition in the wavelet framework. Contrary to the case of MWF, where three rank values are required for each wavelet packet coefficient, a single rank value is required in the Parafac decomposition. Hence, we could adapt the Nelder-Mead simplex method, whose global convergence is ensured in a one-dimensional problem As further works, we could adopt the proposed optimization strategy to any other denoising methods which require the tuning of parameters, but also to classification methods, such as for the choice of optimal kernel parameters for support vector machine.

Chapter 4

Image acquisition systems

4.1 Introduction

Illumination is a particularly important component of the image acquisition system for both human and automatic surface inspection. The choice of the illumination technique is a crucial step in the designing process of any machine vision system. The goal of illumination in machine vision is to make the important features of the object to be inspected visible and reduce undesired features [148]. The quality of these important features is related to the illumination concept, as they need to be presented with a maximum of contrast. The challenge of illumination is to increase the signal to noise ratio, and to emphasize and explore these features to maximize the contrast. The means to increase the contrast are the direction of the light, the choice of the light spectral band and the effect of polarization [50].

The HSI algorithms are mainly used in remote sensing applications, which treat threedimensional data (spatial and spectral information) generated from hyperspectral sensors. HSI are non-destructive technologies that represent an attractive solution for characterization, classification and quality control of different materials in several industrial sectors. The studies based on the application of HSI techniques to material classification and inspection are increasing every day [36], demonstrating that this technique represents a very smart and promising analytical tool for quality control. However, despite these advantages, HSI is still difficult to be systematically applied, especially in real-time industrial applications, because of the huge amount of data constituting a spectral image. The long computation time sometimes represents a big constraint at industrial level, where often, a fast processing of the collected information is required.

4.2 Setup X-ray Tomography

The simulated object is a plastic cylinder with holes along its height: there are six cylindric holes where various liquid mixtures have been inserted: iodine, silver nitrate, and bone 6.2.2. The tomographic images have been simulated for 8 viewing angles, and for 3 spectral bands. The data are independent along the viewing angles, but the 3 bands are correlated.

The PIXSCAN is installed in a lead box 6.4 whose wall thickness is dimensioned to attenuate the radiation to a threshold lower than the regulatory threshold of work. You can see the two doors on the left panel as well as the baffle allowing the passage of the cables and the ventilation [64].

The front doors provide access to the imaging system itself, as shown in 6.5.

The PIXSCAN is made up of three main blocks: the detector block that supports the XPAD3 camera, the object block consisting of a motorized rotation table 6.2.2 to allow translations in three directions. It allows to acquire images of the set of projections necessary for the tomographic reconstruction.

4.2.1 Results

In this section, we evaluate the performances of our proposed methods on multi-spectral $X - ray$ image. we will firstly present some quantitative results obtained from artificially impaired images with white, identically distributed random noise with the following input SNR (SNR_{in}) values (in dB): 10 and 30.
Secondly, we will present results obtained from image which have inherently poisson nois.e

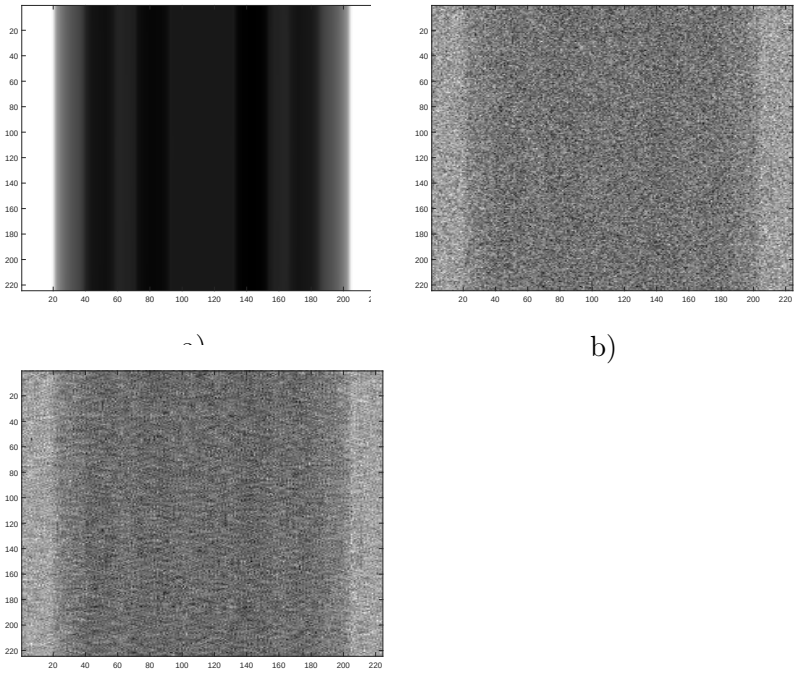
4.2.1.1 Noisy RX image 256×256×3

This image is processed with 2 wavelet packet decomposition levels along the spatial modes, leading to $2^2 \times 2^2 \times 1 = 16$ wavelet packet coefficients of size $64 \times 64 \times 3$. The subsampling factors are set to $S_1 = S_2 = 4$, and $S_3 = 1$.

Exemplification with $SNR_{in} = 10\ dB$ and reference tensor at 60%

To process this image, the values of signal subspace dimension in the reference tensor are set to 60% along the spatial modes, and 100 % along the wavelength mode.

Fig. 4.1 presents the results obtained with the proposed methods on the RX image. It can be noticed that the result image provided by MM-PSO is less noisy.



b)

c)

Figure 4.1: X-Ray image: a) Noise-free ; b) Noisy; c) MM-PSO

Table 4.1 provides the 16 estimated values for the tensor rank in each mode. We notice that the rank values of tensor reference is (64 64 3).

(36 36 3)	(64 64 2)	(38 34 3)	(40 36 3)
(34 36 3)	(64 64 2)	(36 32 3)	(64 64 2)
(64 64 2)	(38 36 3)	(64 64 2)	(36 34 3)
(40 38 3)	(64 64 2)	(38 32 3)	(34 38 3)

Table 4.1: X-Ray image 256×256×3: Rank estimation for all subtensor

Exemplification with $SNR_{in} = 30\ dB$ and reference tensor at 60%

The example in Fig. 4.2 concerns a case where the input SNR is 30 dB.

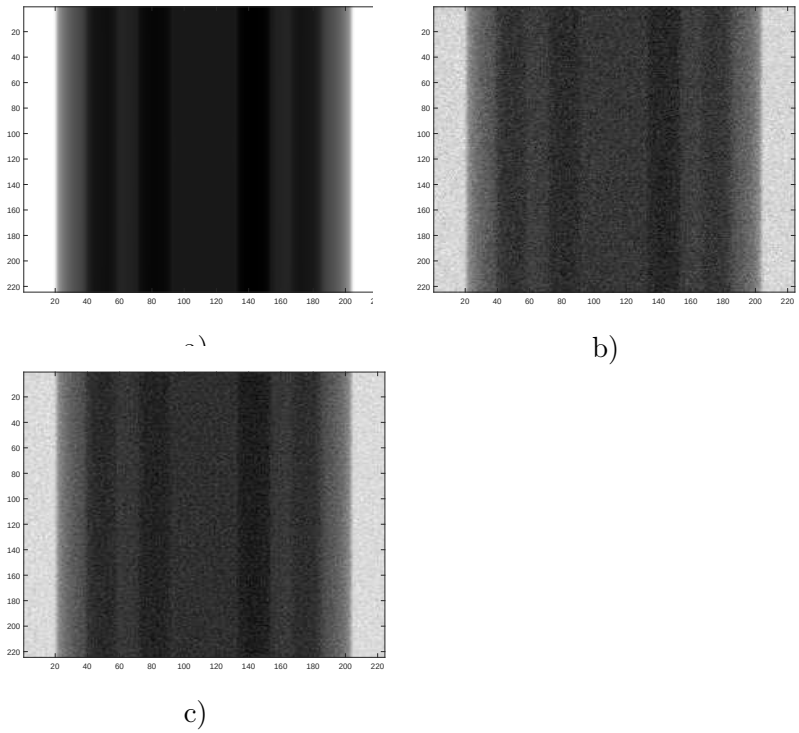


Figure 4.2: X-Ray image: a) Noise-free ; b) Noisy; c) MM-PSO

(36 36 3)	(34 40 3)	(42 38 3)	(38 38 3)
(34 40 3)	(36 38 3)	(64 64 2)	(38 32 3)
(64 64 2)	(40 40 3)	(64 64 2)	(34 34 3)
(42 46 3)	(34 36 3)	(64 64 2)	(38 32 3)

Table 4.2: X-Ray image 256×256×3: Rank estimation for all subtensor

In the following cases, we fix the reference tensor at 20%

Exemplification with $SNR_{in} = 10\ dB$ and reference tensor at 20%

The example in Fig. 4.3 concerns a case where the input SNR is 10 dB.

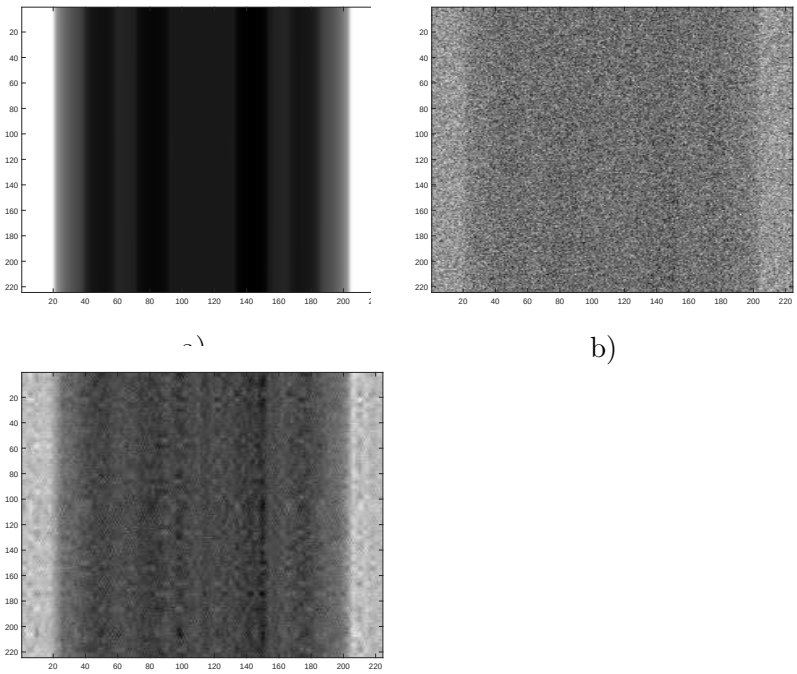


Figure 4.3: X-Ray image: a) Noise-free ; b) Noisy; c) MM-PSO

(8 8 3)	(50 62 2)	(64 64 2)	(64 64 2)
(64 64 2)	(64 64 2)	(64 64 2)	(64 64 2)
(64 62 2)	(60 64 2)	(60 64 2)	(64 64 2)
(64 60 2)	(62 64 2)	(60 64 2)	(64 64 2)

Table 4.3: X-Ray image 256×256×3: Rank estimation for all subtensor

Exemplification with $SNR_{in} = 30\ dB$ and reference tensor at 20%

The example in Fig. 4.4 concerns a case where the input SNR is 30 dB.

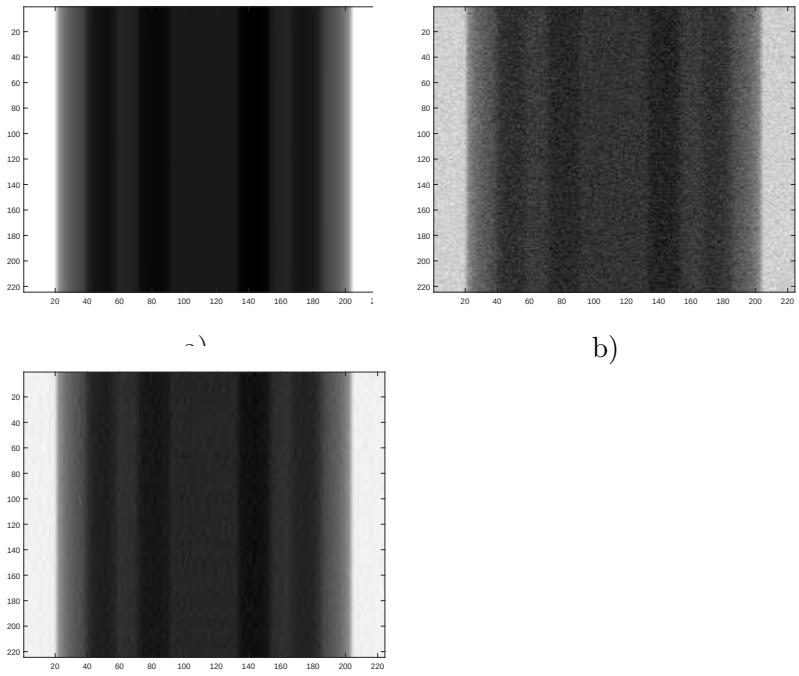


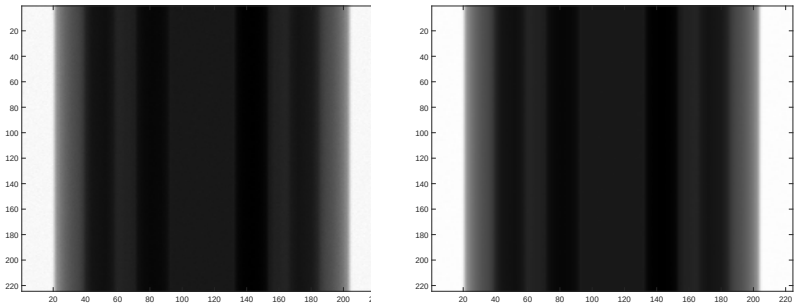
Figure 4.4: X-Ray image: a) : Noise-free ; b) Noisy; c) MM-PSO

(8 8 3)	(8 16 2)	(64 62 3)	(8 20 3)
(64 54 2)	(64 64 2)	(62 64 2)	(62 64 2)
(64 64 2)	(64 62 2)	(64 64 2)	(68 60 2)
(62 64 2)	(64 64 2)	(64 64 2)	(54 64 2)

Table 4.4: X-Ray image 256×256×3: Rank estimation for all subtensor

4.2.1.2 RX image with poisson noise

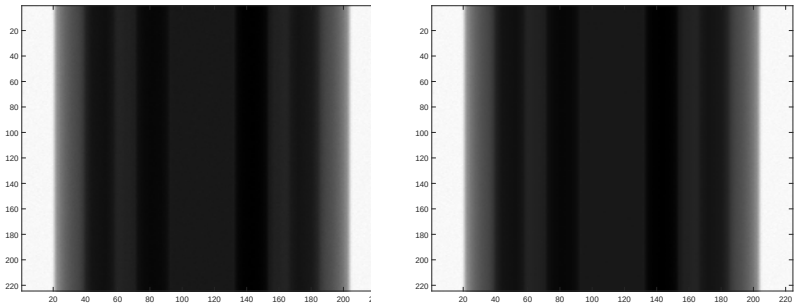
Exemplification with reference tensor at 20%



a) b)

Figure 4.5: X-Ray image: a) Original ; b) MM-PSO

Exemplification with reference tensor at 60%



a) b)

Figure 4.6: X-Ray image: a) Original ; b) MM-PSO

4.2.1.3 Conclusion

This study permits to evaluate the influence of the noise level and the ranks in the reference tensor on the estimated rank values:

	High noise	Low noise
Elevated reference rank 60%	Elevated rank for approximation coefficient,6 pairs of full spatial ranks in the detail coefficients	Elevated rank for approximation coefficient,4 pairs of full spatial ranks in the detail coefficients
Low reference rank 20%	Low rank for approximation coefficient, 14 pairs of full or nearly full spatial ranks in the detail coefficients	Low rank for approximation coefficient,12 pairs of full or nearly full spatial ranks in the detail coefficients

The ranks estimated for the approximation coefficient are mostly influenced by the ranks in the reference tensor. Surprisingly, the ranks estimated for the detail coefficients are more elevated when the noise is high. This means that the most important part of the denoising is performed on the approximation coefficient.

4.3 Experimental setup horizontal lighting

The hyperspectral image acquisition system includes a 2 Kgs sCMOS VIS-NIR camera (see Fig. 6.11 in the chapter 6), consisting of a linear scanning spectrograph which is theoretically sensitive at wavelengths of 100 to 1400 nm, but actually reliable for wavelengths of about 400 to 1000 nm.

4.3.1 Plan of prototype

This type of prototype makes it a shedding light. Indeed, the shining lighting emphasizes the relief and the texture of what is brought to light, which plunges into the shadow everything that is not illuminated. This prototype must be made in a dark room to avoid the reflection of the rays of the white light. For lighting, we use a Khöler Lighting which allows us to obtain a quality image.

You can see the experimental schema in the Fig. 6.8 (in the chapter 6)

4.3.2 Results

In this part, we evaluate the performances of the proposed denoising methods with various values of exposure time used during the acquisition. We consider a scene which is generally used for camera calibration. This image is processed with 2 wavelet packet decomposition levels along the spatial modes and the values of signal subspace dimension in the reference tensor are set to 60% along the spatial modes, and 100 $\%$ along the spectral mode.

We performed experiments with the following values of exposure time: 1ms, 5ms, and 10ms. Fig. 4.7 displays in the first column the raw images acquired with 1, 5, or 10 ms as exposure time respectively; and in the second column the denoising result.

We notice that regardless of the value of the exposure time we obtain the same quality of denoised image: we thus gain up to a factor 10 in terms of acquisition time, saved while acquiring hyperspectral images. For the rest of study, we applied the same method for the setup in Germany. Indeed we wished to check if this setup could improve the quality of acquisitions of metallic surfaces.

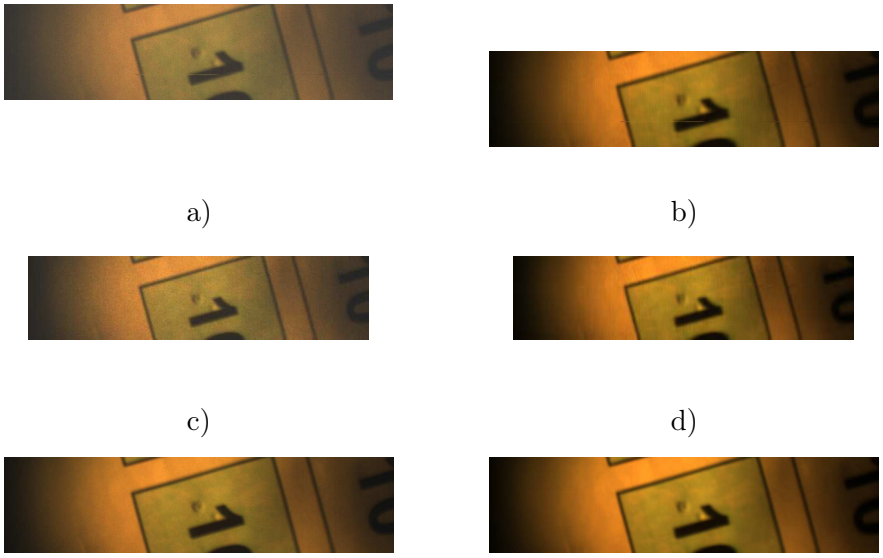


Figure 4.7: a) raw, 1 ms ; b) MM-MWF, 1 ms ; c) raw, 5 ms; d) MM-MWF, 5 ms ; e) raw, 10 ms; f) MM-MWF, 10 ms

4.4 Setup of metallic surface

In this section, we present the acquisition system and some examples of acquisitions performed in Fraunhofer institute IIS.

The hyperspectral image acquisition system consists of a Pike 145F camera with 1.45 megapixel (1388 $\times$ 1038) (see Fig. 4.9), and a linear scanning spectrograph.

Our goal is to test the ability of this acquisition system based on non-direct lighting of the target, to avoid light saturation.

4.4.1 Description of metallic surface setup

Our acquisition system is placed in a dark room: It consists of the hyperspectral camera, a source of white light and a moving plane. The target is placed on the moving plane. The movable plane is controlled by a motor.
In order to perform the acquisition correctly, a collimator is placed in front of the lamp. Indeed the lamp provides a white light, which is not homogeneous. A collimator is a device that narrows a beam of particles or waves. To narrow can mean either to cause the directions of motion to become more aligned in a specific direction (i.e., make collimated light or parallel rays), or to cause the spatial cross section of the beam to become smaller (beam limiting device).

Then a concentrator lens is placed in front of the collimator to target a particular reflection angle.
The whole of this lighting system produces a lighting angle of thirteen degrees with the horizontal.
Thus, according to the laws of geometrical optics, the majority of the energy is reflected and the camera captures only the refracted ray.

In this way a grazing illumination is obtained which emphasizes the relief and the texture of what is brought to light, which plunges into the shadow everything that is not illuminated.

This system allowed us to avoid the saturation of the images produced by the camera.

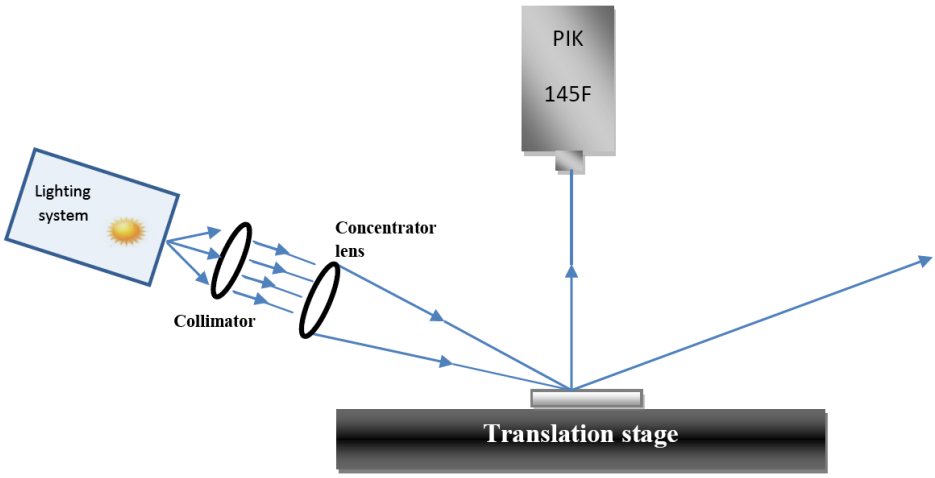


Figure 4.8: Schematic of the experimental setup

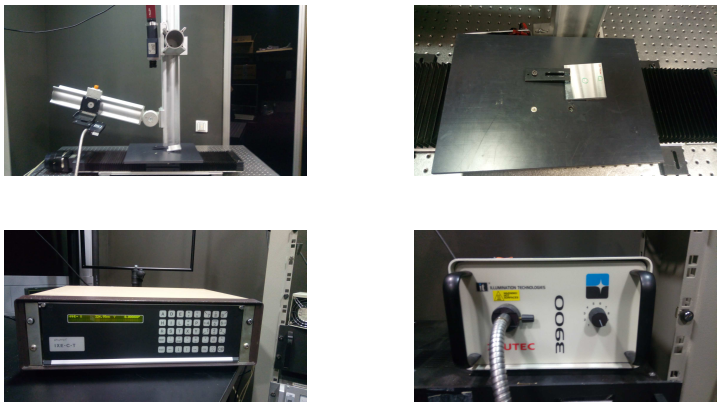


Figure 4.9: The hyperspectral image acquisition system

4.4.2 Acquisitions with metallic surface setup

In Fig. 4.10, we display the acquisition performed with a classical acquisition system with direct illumination.

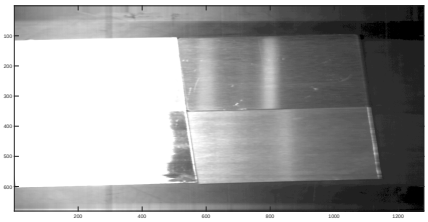


Figure 4.10: Metallic surface imaged with a direct illumination system

In Fig. 4.11, we display the acquisition performed with the proposed system.

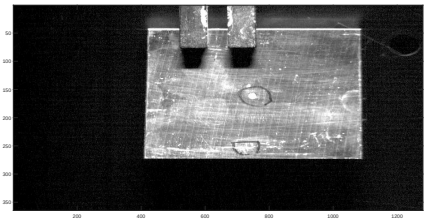


Figure 4.11: Metallic surface imaged with the proposed setup

We can notice, while comparing Figs. 4.10 and 4.11, that the proposed novel system offers a superior image quality, without saturation of light.

Chapter 5

Concluding remarks

In this general conclusion we outline the main outcomes of this thesis, and draw some prospects concerning both algorithmic and applicative aspects.

5.1 Balance

1. Chapter 2 presented the state of the art. Primarily, it presented the fundamental concepts of HSIs. It explained how the HSI images are obtained, how they are represented and the reason why HSI technology is a suitable tool in remote sensing. We recalled the "Hughes phenomenon" related to the high dimensionality of the HSI images and we explained the need of the data space reduction step. This chapter also presented the main tools of multilinear algebra which are conventionally used for the processing of hyperspectral images. More particularly it recalled some definitions and properties of matrix and tensors. Lastly, it presented two commonly used tensor decompositions, including theoretical examples.

2. Chapter 3 devoted to the development of algorithms for denoising of multidimensional.

   An algorithm named MWPT-MWF-RE has been proposed to estimate multiple values of signal subspace dimension, to apply multiway Wiener filtering in a wavelet packet framework.

   To judge the effectiveness of our algorithms,a comparative study was carried out with three main types of algorithms: on the one hand, the Perona-Malik method based on diffusion; Second, the truncation of HOSVD and MWF, and thirdly, a method based on wavelet packet decomposition and MWF, where

the dimensions of the signal subspace are estimated by a statistical criterion rather than by an optimization method.

The results are promising in terms of denoising in grund truth. Ultimately, we achieve an advantageous time saving during the acquisition of hyperspectral images.

3. Chapter 4 presented the different montages used for his research.. It begins by presenting the setup of the fluorescence. Now more than ever, Fluorescence Microscopy is is a major tool in the biology thanks to the work of the rewarded scientists who bypassed the problem of the undulatory nature of light, which allowed an expansion in hyperspectral imaging. Therefore, in this part we make a brief theoretical description of the scheme, then we describe the system used to take fluorescent hyperspectral images. The experimental setup is a hyperspectral fluorescence setup in the IRSTEA (Institut national de recherche en sciences et technologies pour l'environnement et l'agriculture) Laboratory in Montpellier. The second setup is a multispectral X-Ray setup used in CPPM (Centre de Physique des Particules de Marseille) in Marseille. Precisely, it is meant for X-ray spectral tomography. The third setup is the one that I have built in Fraunhofer Institue IIS in Furth, Germany, in order to acquired hyperspectral images of metal surfaces.

5.2 Prospects of the Thesis

The prospects of the thesis two-fold: they concern both the algorithms, and the applicative aspects.

Firstly, as concerns the algorithms:

1. Investigate additional swarm intelligence methods such as Grey Wolf Optimization, glow worm optimization, ..., and their interest for parameter estimation in multidimensional image denoising.

2. In the field of image characterization by hyperspectral images, it is relevant to select the most relevant spectral bands for a given application. An automatic method could be adapted for this: We could propose a relevant criterion to estimate either the number of -regularly spaced- bands, or determine the most relevant band indices.

3. Investigate the interest of denoising only the most relevant coefficients of the wavelet packet decomposition: the exhaustive filtering of all coefficients is

time consuming, and may not be always required to reach a significant improvement in terms of signal to noise ratio.

4. Consider the estimation of the rank of the Parafac decomposition of the wavelet packet decomposition coefficients as a whole. In this thesis, we estimate the best possible Parafac rank independently for each coefficient. We could investigate the inter-dependence of these rank values.

Secondly, as concerns the applicative aspects:

1. Consider other applications involving multidimensional data acquisition systems: we infer that for several applications yielding data of high dimensionality, it is of great importance to reduce the time dedicated to the acquisition. This will be done at the expense of a degradation of the signal to noise ratio, because the lower the exposure time, the higher the gain of the sensors. The denoising methods proposed in this thesis have been applied to fields which are as various as the characterization of plant leaf by fluorescence, liquid mixtures by X-Ray scanning, aerial images... Hence, we can hope that could be applied to any noisy multidimensional data.

2. Create application-oriented denoising methods. For instance, the noise which impairs the fluorescence data and the X-Ray data we have considered is rather modeled as Poisson noise than as Gaussian noise. The proposed methods based on multiway Wiener filtering are optimal in the sense of the maximum likelihood only when Gaussian noise is considered. We could extend this method to other types of noise such as Poisson noise.

3. Investigate the interest of the truncation of the Parafac decomposition inserted in a wavelet framework on real-world data which are impaired with signal-dependent noise.

4. We wish to go further into the matter as concerns the non-destructive testing of metal surfaces: for this we should build a setup including calibration with one image, and testing with all but the first image. The calibration would include not only the classical calibration of the optical system, but also the estimation of the best possible parameters for the denoising method, such as the number of decomposition levels in the wavelet packet decomposition, and the values of the ranks in each coefficient."

Chapter 6

Appendix

6.1 Setup Chlorophyll fluorescence

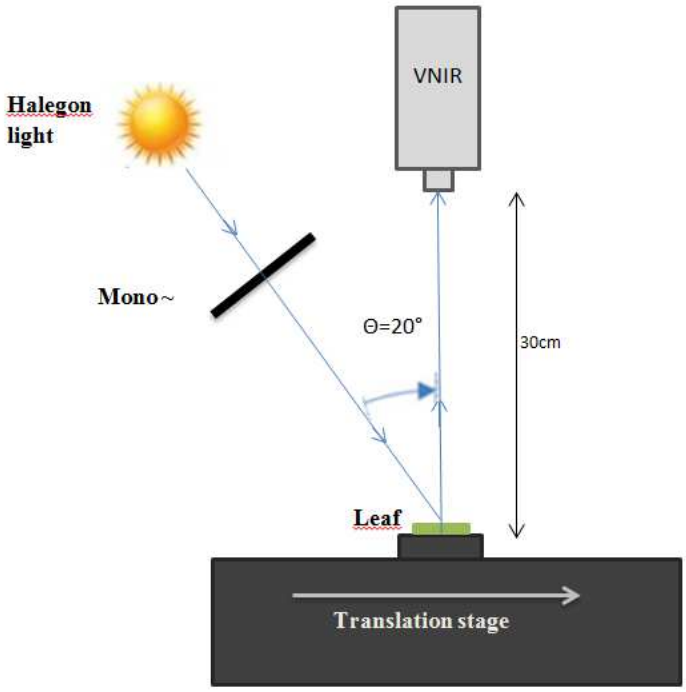


Figure 6.1: Schematic of the experimental setup

Through a collaboration with IRSTEA Laboratory in Montpellier, and the kind help of Dr. Ryad Bendoula, we could afford to make some hyperspectral acquisitions of leaves, to emphasize chlorophyll fluorescence. The hyperspectral camera which is

used is a HySpex VNIR-1600 camera (Norsk Elektro Optikk, Norway) which acquires successive lines of 1600 pixels and 160 spectral bands ranging from 415 to 994 nm with a 3.7 nm spectral sampling interval. As shown in Fig. 6.3, the camera was placed at 30 cm above the imaged leaf. The lighting was provided by a halogen source positioned close to the cameras. The illumination zenith angles were set to $\theta = 20°$ for the VNIR camera. A monochromator is placed in the incoming halogen irradiance between the leaf and the halogen. This monochromator transmits only the wavelength which induces leaf fluorescence. Fig. 6.2 provides photographies of the fluorescence HSI acquisition system.

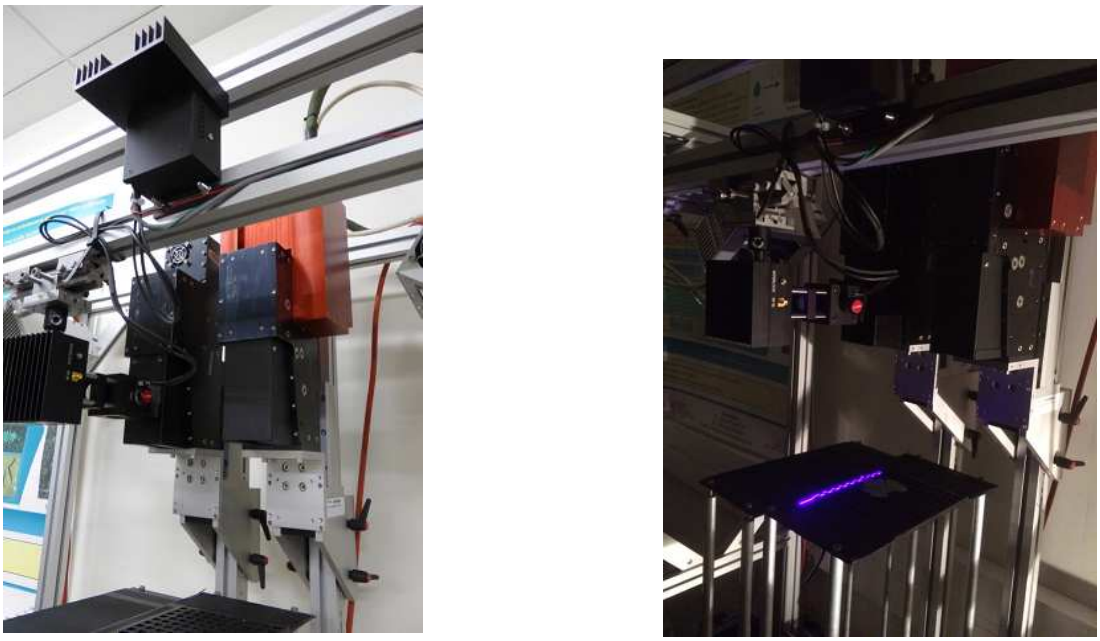


Figure 6.2: HSI camera in a chlorophyll fluorescence setup

We inferred from these conditions of acquisition that it could be interesting in the frame of this work, to test our proposed denoising methods on such images: a fluorescence phenomenon consists of light of a very low intensity. Hence, the gain of the sensors is pushed to a high value, which increases the noise magnitude.

6.2 Setup X-ray Tomography

6.2.1 Plan of the PIXSCAN prototype

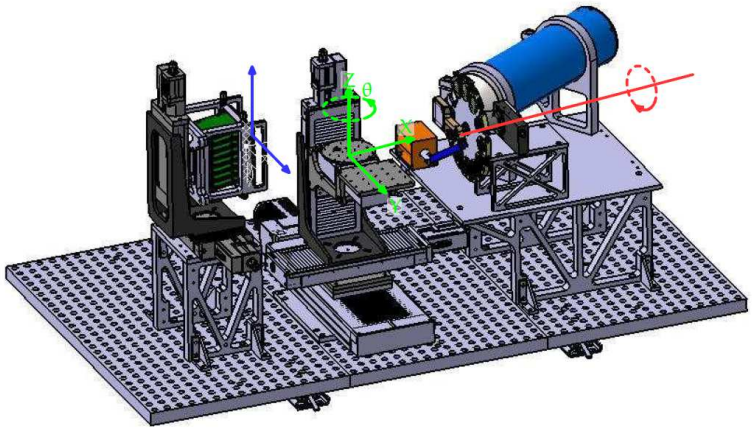


Figure 6.3: Plan of the prototype PIXSCAN

6.2.2 Photography of the PIXSCAN micro-CT prototype

Figure 6.4: Photo of the box where the PIXSCAN is installed.

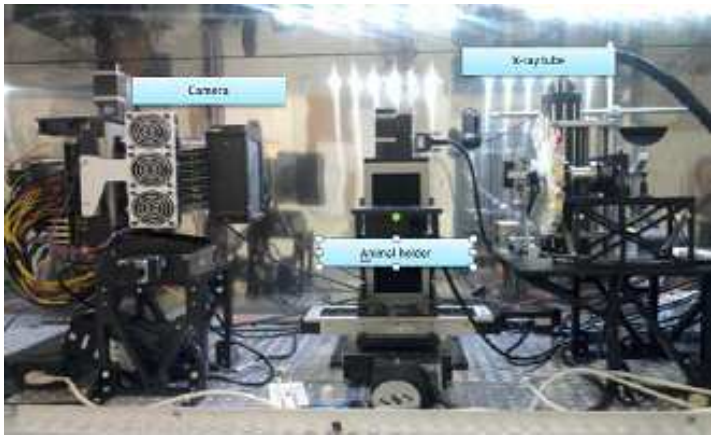


Figure 6.5: PIXSCAN picture

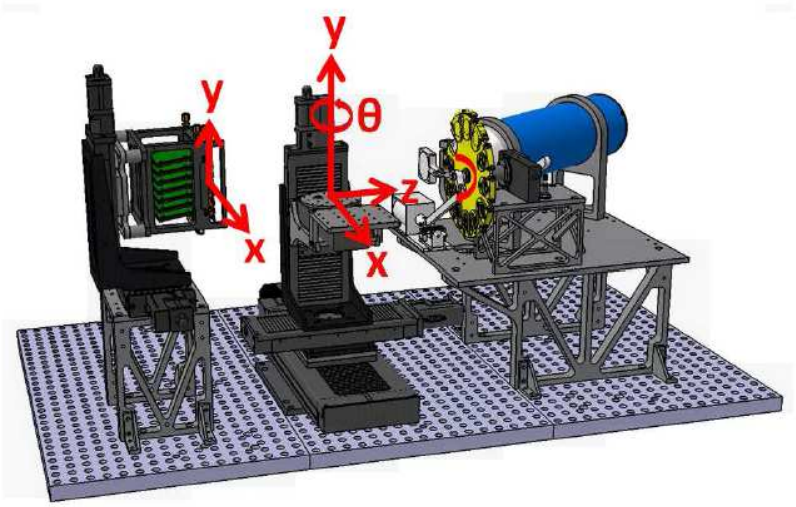


Figure 6.6: Diagram showing the orientation convention of the axes used in the PIXSCAN prototype [64]

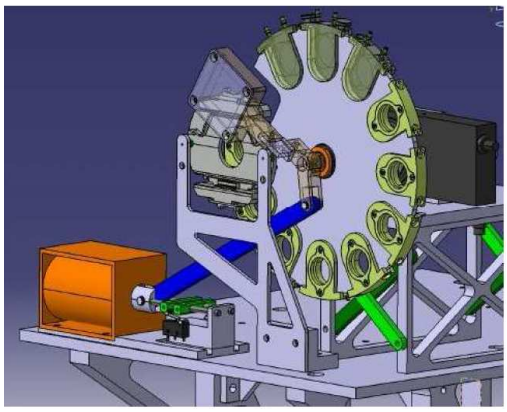


Figure 6.7: The filter wheel [30]

6.3 Horizontal lighting setup in Fresnel Institute

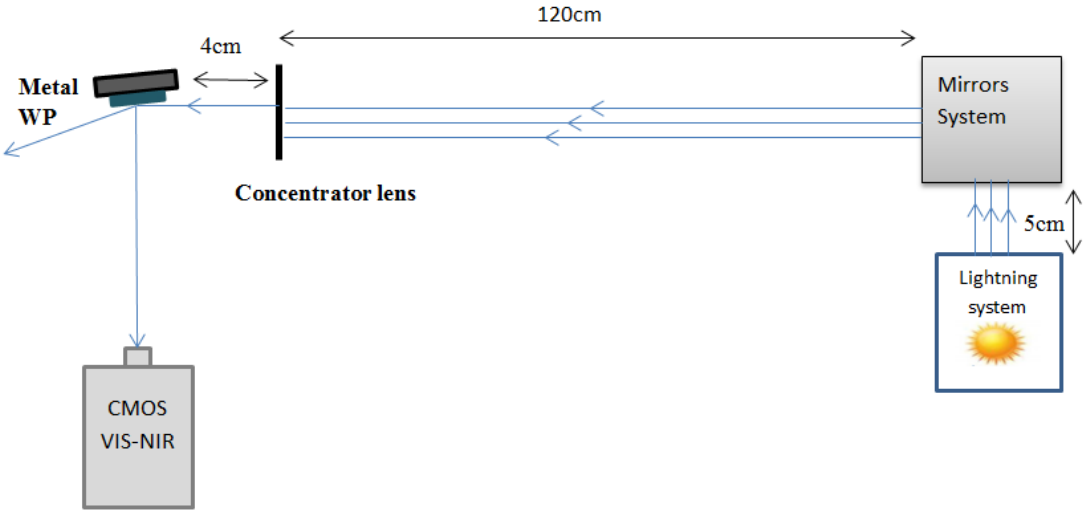


Figure 6.8: Schematic of the experimental setup

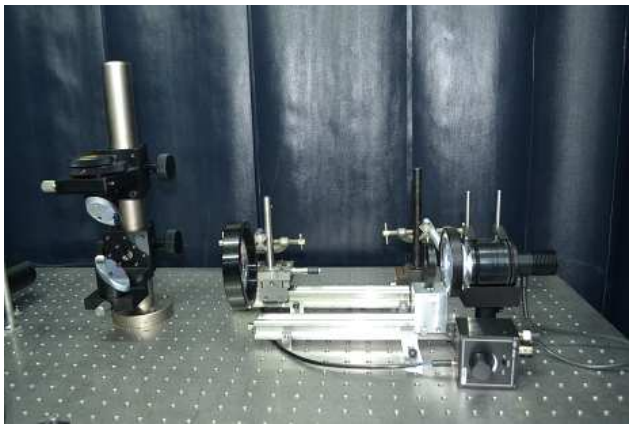


Figure 6.9: Light source

Figure 6.10: Target scene

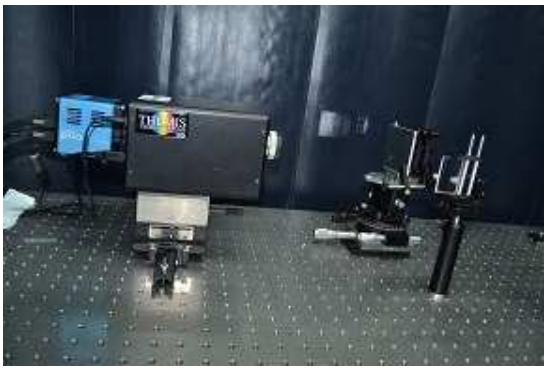


Figure 6.11: Hyperspectral camera

6.4 Summary of the main algorithms

Algorithm 6 is a summarized version of Algorithm 3. Algorithm 7 details the optimal filtering of a coefficient, and Algorithm 8 is a summarized version of Algorithm 4.

Algorithme 6 MWPT-MWF-RE

Input: MWF, fixed rank values: $\mathcal{R} \longrightarrow \mathcal{X}_1$

1. Compute Multidimensional Wavelet Packet Transform:

   MWPT: $\mathcal{R} \longrightarrow \mathcal{C}_{\mathbf{l}}^{\mathcal{R}} = \mathcal{R} \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3$

   MWPT: $\mathcal{X}_1 \longrightarrow \mathcal{C}_{\mathbf{l}}^{\mathcal{X}_1} = \mathcal{X}_1 \times_1 \mathbf{W}_1 \times_2 \mathbf{W}_2 \times_3 \mathbf{W}_3$.

2. Extract wavelet packet coefficients:

   $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}} = \mathcal{C}_{\mathbf{l}}^{\mathcal{R}} \times_1 \mathbf{E}_{m_1} \times_2 \mathbf{E}_{m_2} \times_3 \mathbf{E}_{m_3}$,

   $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_1} = \mathcal{C}_{\mathbf{l}}^{\mathcal{X}_1} \times_1 \mathbf{E}_{m_1} \times_2 \mathbf{E}_{m_2} \times_3 \mathbf{E}_{m_3}$.

3. For each couple of coefficients $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}}$ and $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}_1}$:

   Rank estimation and filtering for each coefficient:

   $\mathcal{C}_{\mathbf{l},\mathbf{m}}^{\mathcal{R}} \longrightarrow \hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}}$

4. Concatenate: $\hat{\mathcal{C}}_{\mathbf{l},\mathbf{m}}^{\mathcal{X}} \longrightarrow \hat{\mathcal{C}}_{\mathbf{l}}^{\mathcal{X}}$.

5. Retrieve estimated tensor

   $\hat{\mathcal{X}} = \hat{\mathcal{C}}_{\mathbf{l}}^{\mathcal{X}} \times_1 \mathbf{W}_1^T \times_2 \mathbf{W}_2^T \times_3 \mathbf{W}_3^T$.

Output: denoised tensor $\hat{\mathcal{X}}$.

Algorithme 7 Rank estimation and filtering for each coefficient

Input: $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$, $\mathcal{C}^{\mathcal{X}_1}_{\mathbf{l},\mathbf{m}}$

1. Get a 'small version' of the coefficients

   Subsample with factors S_1, S_2, S_3:

   $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}} \longrightarrow \mathcal{C}^{\mathcal{R}_S}_{\mathbf{l},\mathbf{m}}$ Noisy coefficient

   $\mathcal{C}^{\mathcal{X}_1}_{\mathbf{l},\mathbf{m}} \longrightarrow \mathcal{C}^{\mathcal{X}_{1S}}_{\mathbf{l},\mathbf{m}}$ Gross estimate coefficient

2. Adapt GA or PSO $\longrightarrow \hat{K_1}, \hat{K_2}, \hat{K_3}$:

   Criterion to minimize:

   $J_{\mathbf{m}}(K_1, K_2, K_3) = ||\text{Reference tensor} - \text{Estimated tensor}||^2$

   Reference tensor: Gross estimate coefficient $\mathcal{C}^{\mathcal{X}_{1S}}_{\mathbf{l},\mathbf{m}}$

   Estimated tensor: obtained from $\mathcal{C}^{\mathcal{R}_S}_{\mathbf{l},\mathbf{m}}$ by

   MWF: $\mathcal{C}^{\mathcal{R}_S}_{\mathbf{l},\mathbf{m}} \longrightarrow \hat{\mathcal{C}}^{\mathcal{X}_S}_{\mathbf{l},\mathbf{m}} = \mathcal{C}^{\mathcal{R}_S}_{\mathbf{l},\mathbf{m}} \times_1 \mathbf{H}_{1,\mathbf{m}} \times_2 \mathbf{H}_{2,\mathbf{m}} \times_3 \mathbf{H}_{3,\mathbf{m}}$,

   Filters $\mathbf{H}_{1,\mathrm{m}}$, $\mathbf{H}_{2,\mathrm{m}}$, $\mathbf{H}_{3,\mathrm{m}}$ depend on K_1,K_2, K_3.

   output of GA or PSO: estimated K_1,K_2, K_3.

3. Rescale rank values $\hat{K_1} = S_1 K_1$, $\hat{K_2} = S_2 K_2$, $\hat{K_3} = S_3 K_3$.

   Strong assumption:

   Features are still present in the 'small version' of the coefficients

4. Apply MWF to each $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}}$:

   MWF: $\mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}} \longrightarrow \hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}$:

   $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}} = \mathcal{C}^{\mathcal{R}}_{\mathbf{l},\mathbf{m}} \times_1 \mathbf{H}_{1,\mathbf{m}} \times_2 \mathbf{H}_{2,\mathbf{m}} \times_3 \mathbf{H}_{3,\mathbf{m}}$

   $\mathbf{H}_{1,\mathrm{m}}$, $\mathbf{H}_{2,\mathrm{m}}$ and $\mathbf{H}_{3,\mathrm{m}}$ depend on $\hat{K_1}$, $\hat{K_2}$, $\hat{K_3}$.

Output: denoised coefficient tensor $\hat{\mathcal{C}}^{\mathcal{X}}_{\mathbf{l},\mathbf{m}}$.

Algorithme 8 Estimation of the number of decomposition levels

Input: $\mathcal{R}$ of size $I_1 \times I_2 \times I_3$

set of candidate vectors $\mathbf{l}$

MWF: $\mathcal{R} \longrightarrow \mathcal{X}_1$

Fixed rank values half of tensor size $K_1 = \frac{I_1}{2}$, $K_2 = \frac{I_2}{2}$, $K_3 = \frac{I_3}{2}$

1. Subsample with factors S_1, S_2, S_3:
   Get a 'small version' of the tensors
   $\mathcal{R} \longrightarrow \mathcal{R}_{\mathcal{S}}$ Small noisy tensor
   $\mathcal{X}_1 \longrightarrow \mathcal{X}_{1S}$ Small gross estimate

2. For each candidate vector $\mathbf{l} = [l_1, l_2, l_3]^T$:

   (a) Apply algorithm MWPT-MWF-RE to get $\hat{\mathcal{X}}_{\mathcal{S}}$:
   with the number of decomposition levels in $\mathbf{l}$
   with fixed rank values half coefficient size
   $K_1 = \frac{I_1}{S_1\ 2^{1+l_1}}$,
   $K_2 = \frac{I_2}{S_2\ 2^{1+l_2}}$,
   $K_3 = \frac{I_3}{S_3\ 2^{1+l_3}}$,

   (b) Retrieve the error $e = ||\mathcal{X}_{1S} - \hat{\mathcal{X}}_{\mathcal{S}}||$.

3. Select the vector $\mathbf{l}$ for which the error e is minimum.

Output: optimal vector $\mathbf{l}$ in terms of least squares error.

Bibliography

[1] I. Atkinson, F. Kamalabadi, S. Mohan, and D. Jones. Wavelet-based 2-d multichannel signal estimation. Image Processing. 2000.

[2] Bellman and Richard. Dynamic programming and lagrange multipliers. *Proceedings of the National Academy of Sciences*.

[3] J. Benediktsson, J. Palmason, and J. Sveinsson. Classification of hyperspectral data from urban areas based on extended morphological profiles. *Geoscience and Remote Sensing, IEEE Transactions on*, 43(3):480–491, 2005.

[4] J. Bioucas-Dias and J. Nascimento. Hyperspectral subspace identification. *Geoscience and Remote Sensing, IEEE Transactions on*, 46(8):2435–2445, 2008.

[5] J. D. Carroll and J. J. Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of 'eckart-young' decomposition. *Psychometrika*, 35:283–319, 1970.

[6] J. D. Carroll and J.-J. Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-young" decomposition. *Psychometrika*, 35(3):283–319, 1970.

[7] R. Cattell. Parallel proportional profiles and other principles for determining the choice of factors by rotation. *Psychometrika*, 9:267–283, 1944.

[8] R. B. Cattell. The three basic factor-analytic research designs - their interrelations and derivatives. *Psychometrika*, 49:452–499, 1952.

[9] E. Ceulemans and H. A. L. Kiers. Selecting among three-mode principal component models of different types and complexities: A numerical convex hull based method. *British Journal of Mathematical and Statistical Psychology*, 59(1):133–150, 2006.

[10] C.-I. Chang. *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*. Kluwer Academic/Plenum Publishing Co., New York, 2003.

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

[12] J. Chanussot, C. Collet, and K. Chehdi. *Multivariate Image Processing*. Wiley-ISTE, British Library Cataloguing-in-Publication Data, 2009.

[13] K. Chen, T. Li, and T. Cao. Tribe-pso: A novel global optimization algorithm and its application in molecular docking. *Chemometrics and Intelligent Laboratory Systems*, 82(10):248–259, 2006.

[14] T. Chen. The past, present, and future of image and multidimensional signal processing. *IEEE Signal Processing Magazine*, 15(2):21–58, 1998.

[15] Y. Chenghai, J. H. Everitt, D. Qian, L. Bin, and J. Chanussot. Using high-resolution airborne and satellite imagery to assess crop growth and yield variability for precision agriculture. *Proc. IEEE*, 101(3):582–592, 2013.

[16] D. Cherifi, I. Hafnaoui, and A. Nait Ali. Multimodal score-level fusion using hybrid ga-pso for multibiometric system. *Informatica*, 39:209–216, 2005.

[17] R. N. Clark, G. A. Swayze, K. E. Livo, R. F. Kokaly, S. J. Sutley, J. B. Dalton, R. R. McDougal, and C. A. Gent. Imaging spectroscopy: Earth and planetary remote sensing with the usgs tetracorder and expert systems. *Journal of Geophysical Research: Planets*, 108(E12), 2003.

[18] M. Clerc. Tribes-un exemple d'oxptimisation par essaim particulaire sans parametres de contrôle. *Optimisation par Essaim Particulaire (OEP 2003), Paris, France*, 64, 2003.

[19] A. R. Conn, N. I. Gould, and P. Toint. A globally convergent augmented lagrangian algorithm for optimization with general constraints and simple bounds. *SIAM Journal on Numerical Analysis*, 28(2):545–572, 1991.

[20] Y. Cooren, M. Clerc, and P. Siarry. Performance evaluation of tribes, an adaptive particle swarm optimization algorithm. *Swarm Intelligence*, 3(2):149–178, 2009.

[21] D. Muti and S. Bourennane and J. Marot. Lower-rank tensor approximation and multiway filtering. *SIAM Journal on Matrix Analysis and Applications*, 30(3):1172–1204, 2008.

[22] I. Daubechies. Ten lectures on wavelets. *CBMS-NSF Lecture Notes, SIAM*, (61), 1992.

[23] L. David. Multispectral data analysis:a signal theory perspective. Technical report, Purdue University, Purdue University, USA, 1998.

[24] L. De Lathauwer, J. Castaing, and J.-F. Cardoso. Fourth-order cumulant based blind identification of underdetermined mixtures. *IEEE Transactions on Signal Processing*, 55:2965–2973, 2007.

[25] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21:1253–1278, April 2000.

[26] L. De Lathauwer, B. De Moor, and J. Vandewalle. On the best rank-(1) and rank-($r_1, \ldots, r_N$) approximation of higher-order tensors. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1324–1342, April 2000.

[27] A. Depeursinge, A. Foncubierta-Rodriguez, D. Van De Ville, and H. Müller. Three-dimensional solid texture analysis in biomedical imaging: Review and opportunities. *Medical Image Analysis*, 18(1):176–196, Jan. 2014.

[28] Y.-F. Dong, J.-H. Gu, N.-N. Li, X.-D. Hou, and W.-L. Yan. Combination of genetic algorithm and ant colony algorithm for distribution network planning. In *Machine Learning and Cybernetics, 2007 International Conference on*, volume 2, pages 999–1002. IEEE, 2007.

[29] D. Donoho. High-dimensional data analysis : the curse and blessing of dimensionality. *Math Challenges of the 21st Century. American Mathematical Society Edition*, 2000.

[30] M. Dupont. *Photon Counting Spectral Tomography: Development of the Prototype PIXSCAN and Proof of Concept*. Theses, Aix-Marseille Université, Apr. 2014.

[31] R. C. Eberhart, J. Kennedy, et al. A new optimizer using particle swarm theory. In *Proceedings of the sixth international symposium on micro machine and human science*, volume 1, pages 39–43. New York, NY, 1995.

[32] Y. Eckart and G. Young. The approximation of a matrix by another of lower rank. *Psychometrika*, 1:211–218, 1936.

[33] Y. Eckart and G. Young. The approximation of a matrix by another of lower rank. *Psychometrika*, 1:211–218, 1936.

[34] J. Ellis, H. Davis, and J. Zamudio. Exploring for onshore oil seeps with hyperspectral imaging. *Oil Gas J.*, 99(37):49–58, 2001.

[35] C. D. Everard, M. S. Kim, and H. Lee. A comparison of hyperspectral reflectance and fluorescence imaging techniques for detection of contaminants on spinach leaves. *Journal of Food Engineering*, 143:139 – 145, 2014.

[36] F. Gao and L. Han. Implementing the nelder-mead simplex algorithm with adaptive parameters. *Computational Optimization and Applications*, 51(1):259–277, 2010.

[37] P. Ghosh, M. Mitchell, J. A. Tanyi, and A. Y. Hung. Incorporating priors for medical image segmentation using a genetic algorithm. *Neurocomputing*, Feb. 2016.

[38] D. E. Goldberg et al. *Genetic algorithms in search optimization and machine learning*, volume 412. Addison-wesley Reading Menlo Park, January 1989.

[39] A. Green, M. Berman, P. Switzer, and M. Craig. A transformation for ordering multispectral data in terms of image quality with implications for noise removal. *Geoscience and Remote Sensing, IEEE Transactions on*, 26(1):65–74, 1988.

[40] R. O. Green, C. M. Sarture, C. J. Chovit, J. A. Faust, P. Hajek, and H. lan Novak. Aviris: A new approach to earth remote sensing. *Opt. Photon. News*, 6(1):30, Jan 1995.

[41] A. M. Grigoryan, E. R. Dougherty, and S. S. Agaian. Optimal Wiener and homomorphic filtration: Review. *Signal Processing*, 121:111–138, Apr. 2016.

[42] R. A. Harshman. Foundations of the parafac procedure: Models and conditions for an "explanatory" multi-modal factor analysis. *UCLA working papers in phonetics*, 16:1–84, 1970.

[43] R. A. Harshman. Book review : Three-mode principal component analysis. *Applied Psychological Measurement*, 9(3):327–332, 1985.

[44] R. Henrion. n-way principal component analysis theory, algorithms and applications. *Chemometrics and Intelligent Laboratory Systems*, 25:1–23, 1994.

[45] R. D. Hewson, T. J. Cudahy, M. Caccetta, A. Rodger, M. Jones, and C. Ong. Advances in hyperspectral processing for province- and continental- wide mineral mapping. In *IEEE IGARSS*, volume 4, pages 701–704, Jul. 2009.

[46] F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics*, 6:164–189, 1927.

[47] F. L. Hitchcock. Multiple invariants and generalized rank of a p-way matrix of tensor. *Journal of Mathematics and Physics*, 7:39–79, 1927.

[48] J. Holland. *Adaptation In Natural And Artificial Systems*. ISBN 0585038449, 9780585038445. Mit press edition, 1992.

[49] J. H. Holland. Genetic algorithms. *Scientific american*, 267(1):66–72, 1992.

[50] A. Hornberg. *Handbook of Machine Vision*. Weinheim, Germany: VILEY-VCH, 2006.

[51] H. Hudson and T. C. Lee. Maximum likelihood restoration and choice of smoothing parameter in deconvolution of image data subject to poisson noise. *Computational Statistics and Data Analysis*, 26(4):393–410, 1998.

[52] D. Jones, C. Pertunen, and B. Stuckman. Lipschitzian optimization without the lipschitz constant. *Journal of Optimization theory and Application*, 79(1):157–181, October. 1993.

[53] C.-F. Juang. A hybrid of genetic algorithm and particle swarm optimization for recurrent network design. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(2):997–1006, April 2004.

[54] A. Jukic, I. Kopriva, and A. Cichocki. Noninvasive diagnosis of melanoma with tensor decomposition-based feature extraction from clinical color image. *Biomedical Signal Processing and Control*, 8(6):755 – 763, 2013.

[55] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proc. IEEE International Conference on Neural Networks*, pages 1942 – 1948, 1995.

[56] B. N. Khoromskij and V. Khoromskaia. Low rank tucker-type tensor approximation to classical potentials. *Central European Journal of Mathematics*, 5:523–550, 2007.

[57] H. Kiers. Towards a standardized notation and terminology in multiway analysis. *Journal of Chemometrics*, 14:105–122, 2000.

[58] H. Kiers and I. Van Mechelen. Three-way component analysis : Principles and illustrative application. *Psychological Methods*, 6:84–110, 2001.

[59] T. Kolda. Orthogonal tensor decompositions. *SIAM J. Matrix Anal Appl*, 23:243–255, 2001.

[60] T. G. Kolda. Orthogonal tensor decompositions. *SIAM Journal on Matrix Analysis and Applications*, 23:243–255, 2001.

[61] T. G. Kolda. Orthogonal tensor decompositions. *SIAM Journal on Matrix Analysis and Applications*, 23(1):243–255, 2001.

[62] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, Sept. 2009.

[63] K. Kotwal and S. Chaudhuri. Visualization of hyperspectral images using bilateral filtering. *IEEE Trans. Geosci. Remote Sens.*, 48(5):2308–2316, 2010.

[64] C. Kronland-Martinet. *Développement de la tomographie intra-vitale au K-edge avec la caméra à pixels hybrides XPAD3*. PhD thesis, AMU, 2015.

[65] P. Kroonenberg. Three-mode principal component analysis. 1983.

[66] P. Kroonenberg. *Three-mode principal component analysis*. DSWO press, Leiden, 1983.

[67] P. Kroonenberg and J. De Leeuw. Principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika*, 45(1):69–97, March 1980.

[68] P. Kroonenberg and J. D. Leeuw. Principal component analysis of three-mode data by means of alternating least squares algorithms. *Psychometrika*, 45(1):69–97, 1980.

[69] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright. Convergence properties of the nelder-mead simplex method in low dimensions. *SIAM Journal of Optimization*, 9(1):112–147, 1998.

[70] D. Landgrebe. Hyperspectral image data analysis. *IEEE Signal Process. Mag.*, 19(1):17–28, 2002.

[71] A. N. Langville and W. J. Stewart. A kronecker product approximate preconditioner for sans. *Numerical Linear Algebra with Applications*, 11:723–752, 2004.

[72] H. Lanteri, M. Roche, O. Cuevas, and C. Aime. A general method to devise maximum-likelihood signal restoration multiplicative algorithms with non-negativity constraints. *Signal Processing*, 81(5):945 – 974, 2001.

[73] L. D. Lathauwer. *Signal processing based on Multilinear Algebra*. PhD thesis, K.U. Leuven, E.E. Dept. (ESAT), September 1997.

[74] L. D. Lathauwer, B. D. Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM Jour. on Matrix An. and Applic.*, 21:1253–78, 2000.

[75] L. D. Lathauwer, B. D. Moor, and J. Vandewalle. On the best rank-1 and rank-(r1 ,r2 ,. . .,rn) approximation of higher-order tensors. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1324–1342, 2000.

[76] L. D. Lathauwer, B. D. Moor, and J. Vandewalle. On the best rank-1 and rank-(r_1,r_2,. . .,r_n) approximation of higher-order tensors. *SIAM Journal on Matrix Analysis and Applications*, 21:1324–1342, 2000.

[77] D. Letexier and S. Bourennane. Adaptive flattening for multidimensional image restoration. *IEEE Signal Processing Letters*, 15:229–232, 2008.

[78] D. Letexier and S. Bourennane. Noise removal from hyperspectral images by multidimensional filtering. *IEEE Transactions on Geoscience and Remote Sensing*, 46(7):2061–2069, 2008.

[79] D. Letexier, S. Bourennane, and J. Blanc-Talon. Main flattening directions and quadtree decomposition for multi-way wiener filtering. *IEEE Signal, Image and Video Processing*, 1(3):253–256, 2007.

[80] D. Letexier, S. Bourennane, and J. Blanc-Talon. Non-orthogonal tensor matricization for hyperspectral image filtering. *IEEE Geoscience and Remote Sensing Letters*, 5(1):3–7, 2008.

[81] S. Lewis, A. Hudak, R. Ottmar, P. Robichaud, L. Lentile, S. Hood, J. Cronan, and P. Morgan. Using hyperspectral imagery to estimate forest floor consumption from wildfire in boreal forests of alaska, usa. *Int. J. Wildland Fire*, 20(2):255–271, 2011.

[82] L. Lieven De, M. Bart De, and V. Joos. A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[83] T. M. Lillesand, R. W. Kiefer, and J. W. Chipman. *Remote sensing and image interpretation*. John Wiley and Sons Ltd, 2004.

[84] T. Lin and S. Bourennane. Hyperspectral image processing by jointly filtering wavelet component tensor. *IEEE Trans. Geosci. Remote Sens.*, 51(6):3529–3541, 2013.

[85] T. Lin and S. Bourennane. Survey of hyperspectral image denoising methods based on tensor decompositions. *EURASIP Journal on Advances in Signal Processing*, 2013(1), 2013.

[86] K. Liu, J. P. C. da Costa, H. C. So, L. Huang, and J. Ye. Detection of number of components in candecomp/parafac models via minimum description length. *Digital Signal Processing*, 51:110 – 123, 2016.

[87] M. Liu, Y. Liu, H. Hu, and L. Nie. Genetic algorithm and mathematical morphology based binarization method for strip steel defect image with non-uniform illumination. *Journal of Visual Communication and Image Representation*, 37:70–77, May 2016.

[88] N. Liu, B. Zhang, J. Yan, Z. Chen, W. Liu, F. Bai, and L. Chien. Text representation: from vector to tensor. In *Fifth IEEE International Conference on Data Mining*, pages 725–728, 2005.

[89] X. Liu, S. Bourennane, and C. Fossati. Denoising of hyperspectral images using the parafac model and statistical performance analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 50(10):3717–3724, 2012.

[90] X. Liu, S. Bourennane, and C. Fossati. Denoising of hyperspectral images using the parafac model and statistical performance analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 50(10):3717–3724, Oct 2012.

[91] X. Liu, S. Bourennane, and C. Fossati. Nonwhite noise reduction in hyperspectral images. *IEEE Geoscience and Remote Sensing Letters*, 9(3):368–372, 2012.

[92] X. Liu, S. Bourennane, and C. Fossati. Reduction of signal-dependent noise from hyperspectral images for target detection. *IEEE trans. Geoscience and Remote Sensing*, 52(9):5396–5411, Sept. 2014.

[93] K.-F. Man, K. S. TANG, and S. Kwong. *Genetic algorithms: concepts and designs*. Springer Science & Business Media, 2012.

[94] J. Marot and S. Bourennane. Particle swarm optimization for blurred contour retrieval. In *Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22nd European*, pages 810–814. IEEE, 2014.

[95] J. Marot and S. Bourennane. *Advanced Concepts for Intelligent Vision Systems: 16th International Conference, ACIVS 2015, Catania, Italy, October 26-29, 2015. Proceedings*, chapter Improvement of a Wavelet-Tensor Denoising Algorithm by Automatic Rank Estimation, pages 779–790. Springer International Publishing, Cham, 2015.

[96] J. Marot, Y. Caulier, A. Kuleschov, K. Spinnler, and S. Bourennane. *Advanced Concepts for Intelligent Vision Systems: 10th International Conference, ACIVS 2008, Juan-les-Pins, France, October 20-24, 2008. Proceedings*, chapter Contour Detection for Industrial Image Processing by Means of Level Set Methods, pages 655–663. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[97] J. Marot, C. Fossati, and S. Bourennane. About Advances in Tensor Data Denoising Methods. *EURASIP Journal on Advances in Signal Processing*, 2008:1–13, 2008.

[98] E. Miller, S. Ladenheim, and C. Martin. HIGHER ORDER TENSOR OPERATIONS AND THEIR APPLICATIONS.

[99] S. Mirjalili, S. M. Mirjalili, and A. Lewis. Grey wolf optimizer. *Advances in Engineering Software*, 69:46 – 61, 2014.

[100] M. Mitchell. *An introduction to genetic algorithms*. MIT press, 1998.

[101] J. Mocks. Topographic components model for event-related potentials and some biophysical considerations. *IEEE Transactions on Biomedical Engineering*, 35(6):482–484, 1988.

[102] N. Mohananthini and G. Yamuna. Comparison of multiple watermarking techniques using genetic algorithms. *Journal of Electrical Systems and Information Technology*, Mar. 2016.

[103] P. Moradi and M. Gholampour. A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy. *Applied Soft Computing*, 43:117–130, June 2016.

[104] O. Morozov, M. Unser, and P. Hunziker. Reconstruction of large, irregularly sampled multidimensional images. a tensor-based approach. *IEEE Transactions on Medical Imaging*, 30(2):366–374, 2011.

[105] D. Muti and S. Bourennane. Multidimensional filtering based on a tensor approach. *Signal Processing*, 85:2338–2353, 2005.

[106] D. Muti and S. Bourennane. Multidimensional filtering based on a tensor approach. *Signal Proceesing Journal, Elsevier*, 85:2338–2353, May 2005.

[107] R. Neelamani, H. Choi, and R. Baraniuk. Forward : Fourier-wavelet regularized deconvolution for i11-conditioned systems. *IEEE Trans. on Signal Processing*, 52(2):418–433, Feb 2004.

[108] D. S. Nicholas, B. Rasmus, and B. G. Georgios. Parallel factor analysis in sensor array processing. *IEEE Transactions on Signal Processing*, 48(8), Aug. 2000.

[109] D. Nion and D. Sidiropoulos, Nicholas. A parafac-based technique for detection and localization of multiple targets in a mimo radar system. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2077–2080, Apr. 2009.

[110] D. Nion and N. Sidiropoulos. Tensor algebra and multidimensional harmonic retrieval in signal processing for mimo radar. *IEEE Transactions on Signal Processing*, 58(11):5693–5705, 2010.

[111] H. Othman and S.-E. Qian. Noise reduction of hyperspectral imagery using hybrid spatial-spectral derivative-domain wavelet shrinkage. *Geoscience and Remote Sensing, IEEE Transactions on*, 44(2):397–408, 2006.

[112] L. Parra, C. Spence, P. Sajda, A. Ziehe, and K. uller. Unmixing hyperspectral data. *Advances in Neural Information Processing Systems*, 12:942–948, 2000.

[113] L. Parra, C. Spence, P. Sajda, A. Ziehe, and K. uller. Unmixing hyperspectral data. *Advances in Neural Information Processing Systems*, 12:942–948, 2000.

[114] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, Jul 1990.

[115] S. Prasad, W. Li, J. E. Fowler, and L. M. Bruce. Information fusion in the redundant-wavelet-transform domain for noise-robust hyperspectral classification. *IEEE Trans. Geosci. Remote Sens.*, PP(99):1–13, 2012.

[116] W. T. Reeves. Particle systems, a technique for modeling a class of fuzzy objects. *ACM Transactions on Graphics (TOG)*, 2(2):91–108, 1983.

[117] N. Renard and S. Bourennane. Improvement of target detection methods by multiway filtering. *IEEE Transactions on Geoscience and Remote Sensing*, 46(8):2407–2417, 2008.

[118] N. Renard and S. Bourennane. Dimensionality reduction based on tensor modelling for classification methods. *IEEE Transactions on Geoscience and Remote Sensing*, 47(4):1123–1131, 2009.

[119] N. Renard, S. Bourennane, and J. Blanc-Talon. Denoising and dimensionality reduction using multilinear tools for hyperspectral images. *IEEE Geoscience and Remote Sensing Letters*, 5(2):138–142, 2008.

[120] N. Renard, S. Bourennane, and J. Blanc-Talon. Denoising and dimensionality reduction using multilinear tools for hyperspectral images. *IEEE Geoscience and Remote Sensing Letters*, 5(2):138 – 142, Apr. 2008.

[121] N. Renard, S. Bourennane, and J. Blanc-Talon. Denoising and dimensionality reduction using multilinear tools for hyperspectral images. *Geoscience and Remote Sensing Letters, IEEE,* 5(2):138–142, April 2008.

[122] D. Schonfeld and N. Bouaynaya. A new method for multidimensional optimization and its application in image and video processing. *IEEE Signal Processing Letters*, 13(8):485–488, 2006.

[123] H. Schwefel and G. Rudolph. *Contemporary Evolution Strategies*. Taylor and francis Group edition, 1995.

[124] S. Schweizer and J. Moura. Efficient detection in hyperspectral imagery. *Image Processing, IEEE Transactions on*, 10(4):584–597, 2001.

[125] G. Shaw and D. Manolakis. Signal processing for hyperspectral image exploitation. *IEEE Signal Process. Mag.*, 19(1):12–16, 2002.

[126] N. Sidiropoulos and R. Bro. On the uniqueness of multilinear decomposition of N-way arrays. *Journal of Chemometrics*, 14:229–239, 2000.

[127] N. Sidiropoulos, G. Giannakis, and R. Bro. Blind parafac receivers for DS-CDMA systems. *IEEE Trans. on Signal Processing*, 48(3):810–823, 2000.

[128] A. Smolic, P. Kauff, S. Knorr, A. Hornung, M. Kunter, M. Müller, Lang, and M. Three-dimensional video postproduction and processing. *Proceedings of the IEEE*, 99(4):607–625, 2011.

[129] C. M. Stellman, F. M. Olchowski, and J. V. Michalowicz. War horse (wide-area reconnaissance: hyperspectral overhead real-time surveillance experiment). In *Aerospace/Defense Sensing, Simulation, and Controls*.

[130] J. Sun and P. Yu. Window-based tensor analysis on high-dimensional and multi-aspect streams. In *Sixth International Conference on Data Mining*, pages 1076–1080, 2006.

[131] J.-T. Sun, H.-J. Zeng, H. Liu, Y. Lu, and Z. Chen. Cubesvd: A novel approach to personalized web search. In *Proceedings of the 14th international conference on World Wide Web*, pages 382–390, 2005.

[132] B. Tekin, U. Kamilov, E. Bostan, and M. Unser. Benefits of consistency in image denoising with steerable wavelets. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 1355–1358, May 2013.

[133] M. E. Timmerman and H. A. L. Kiers. Three-mode principal components analysis: Choosing the numbers of components and sensitivity to local optima. *British Journal of Mathematical and Statistical Psychology*, 53(1):1–16, 2000.

[134] K. Tiwari, M. Arora, and D. Singh. An assessment of independent component analysis for detection of military targets from hyperspectral images. *Int. J. Appl. Earth Obs. Geoinf.*, 13(5):730–740, 2011.

[135] S. Tongchim. Coarse-grained parallel genetic algorithm for solving the timetable problem. In *Proceedings of 3rd Annual National Symposium on Computational Science and Engineering (ANSCSE), Bangkok, Thailand*, pages 345–353, January 1999.

[136] L. R. Tucker. *Problems in Measuring Change*, chapter Implications of factor analysis of three-way matrices for measurement of change, pages 122–137. University of wisconsin Press, 1963.

[137] L. R. Tucker. *The extension of factor analysis to three-dimensional matrices*, chapter in Contributions to Mathematical Psychology. Holt, Rinehardt, Winston, New York, 1964.

[138] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.

[139] G. Vane, R. O. Green, T. G. Chrien, H. T. Enmark, E. G. Hansen, and W. M. Porter. The airborne visible/infrared imaging spectrometer (aviris). *Remote Sens. Environ.*, 44(2):127–143, 1993.

[140] M. Vasilescu and D. Terzopoulos. Multilinear independent components analysis. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'05)*, 1:547–553, 06 2005.

[141] M. A. O. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *Proceedings of the 7th European Conference on Computer Vision, ECCV 2002*, volume 1, pages 447–460, 2002.

[142] T. Veracini, S. Matteoli, M. Diani, and G. Corsini. Nonparametric framework for detecting spectral anomalies in hyperspectral images. *IEEE Geosci. Remote Sens. Lett.*, 8(4):666–670, 2011.

[143] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: from error visibility to structural similarity. *Image Processing, IEEE Transactions on*, 13(4):600 – 612, 2004.

[144] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: from error visibility to structural similarity. *Image Processing, IEEE Transactions on*, 13(4):600 – 612, 2004.

[145] M. Wax and T. Kailath. Detection of signals information theoretic criteria. *IEEE Trans. on Acoust., Speech, Signal Processing*, 33(2):387–392, April 1985.

[146] M. Wei and D. Shen. Minimum rank solutions to the matrix approximation problems in the spectral norm. *SIAM Journal on Matrix Analysis and Applications*, 33(3):940–957, 2012.

[147] D. Williams, Y. Zheng, P. G. Davey, F. Bao, M. Shen, and A. Elsheikh. Reconstruction of 3d surface maps from anterior segment optical coherence tomography images using graph theory and genetic algorithms. *Biomedical Signal Processing and Control*, 25:91–98, Mar. 2016.

[148] Z. Xue-wu, D. Yan-qiong, L. Yan-yun, S. Ai-ye, and L. Rui-yu. A vision inspection system for the surface defects of strongly reflected metal based on multi-class {SVM}. *Expert Systems with Applications*, 38(5):5930 – 5939, 2011.

[149] Y. Yuzeng, L. Weiqun, W. Shouqin, and Z. Yushan. Vegetational spectral characteristics in hongtoushan mining area, liaoning province: Potential of hyperspectral remote sensing in environment monitoring. In *Remote Sensing, Environment and Transportation Engineering (RSETE)*, pages 1–4, Jun. 2012.

[150] Y. Zhang and P. Zhang. Machine training and parameter settings with social emotional optimization algorithm for support vector machine. *Pattern Recognition Letters*, 54:36 – 42, 2015.

[151] Y.-x. Zheng and Y. Liao. Parameter identification of nonlinear dynamic systems using an improved particle swarm optimization. *Optik - International Journal for Light and Electron Optics*, 127(19):7865–7874, Oct. 2016.

[152] M. Zhu and S. Salcudean. Real-time image-based b-mode ultrasound image simulation of needles using tensor-product interpolation. *IEEE Transactions on Medical Imaging*, 30(7):1391–1400, 2011.

[153] A. Zidi, J. Juan, J. Marot, and S. Bourennane. Nonnegative matrix factorization with spatial prior and reference spectra application to remote hyperspectral image understanding. In *Visual Information Processing (EUVIP), 2014 5th European Workshop on*, pages 1–6. IEEE, 2014.

[154] A. Zidi, J. Marot, S. Bourennane, and K. Spinnler. Automatic rank estimation of parafac decomposition and application to multispectral image wavelet denoising. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3101–3105, Sept 2016.

[155] A. Zidi, J. Marot, S. Bourennane, and K. Spinnler. Bio-inspired optimization algorithms for automatic estimation of multiple subspace dimensions in a tensor-wavelet denoising algorithm. *Journal of Remote Sensing Technology*, 4(1):90–114, dec 2016.

[156] A. Zidi, J. Marot, K. Spinnler, and S. Bourennane. Unmixing of hyperspectral images with pure prior spectral pixels. In *VISAPP (1)*, pages 153–158, 2015.

[157] A. Zidi, K. Spinnler, J. Marot, and S. Bourennane. Multispectral image denoising in wavelet domain with unsupervised tensor subspace-based method. In *Visual Information Processing (EUVIP), 2016 6th European Workshop on*, pages 1–6. IEEE, 2016.