Endmember Induction Methods for Hyperspectral Imagery

Miguel Angel Veganzones

0.1. Introduction

0.1.1. Hyperspectral Imagery. Spectral imaging [**7**] refers to the collection of optical images taken in multiple wavelength bands that are spatially aligned such that at each pixel there is a vector representing the response to the same spatial location for all wavelengths. Hyperspectral imaging systems (HSI) differ from color and multispectral imaging systems (MSI) in three points:

- *Number of bands*: color and MSI images use to have three to ten spectral bands, while HSI images tend to have hundred of co-registered bands.
- *Spectral resolution* (center wavelength divided by the width of the spectral band, $\lambda/\Delta\lambda$): color and MSI system's spectral resolution is on the order of 10, HSI systems on the order of 100.
- *Contiguity*: MSI systems have their spectral bands widely and irregularly spaced, while HSI systems have contiguous and regularly spaced bands.

Although MSI systems and HSI systems have a common background in the field of Remote Sensing and the later appears as a natural evolution from the MSI systems when technology made possible the development of such kind of detector arrays, the nature of their data is very different and requires specific analysis tools. While color and MSI systems analysis techniques are very related to the spatial characteristics of the data and they usually deal with each spectral band individually, techniques used with hyperspectral imagery exploit the spectral information contained in the hundreds of contiguous and regularly spaced bands that can be seen as a continuous spectrum measured for each pixel.

HSI images use to be illustrated as an image cube with the spatial information on the face and the spectral along the sides. Figure 0.1.1 is a sample image from the JPL's airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor.

An schema of a typical hyperspectral imaging system (HSI) is showed in Figure 0.1.2. There are four basic parts into an Earth Observation (EO) acquisition system [**26**]:

- The *radiation source*, the sun and the thermal radiation of the surface in HSI systems.
- The *atmospheric path* that introduce lots of artifacts to the signal.
- The Earth surface, the *imaged scene* whose interaction with the illumination energy originates the radiance signal.
- The *sensor* where the electromagnetic field is sampled, spatially, spectrally, temporally and radiometrically to create the image cube.

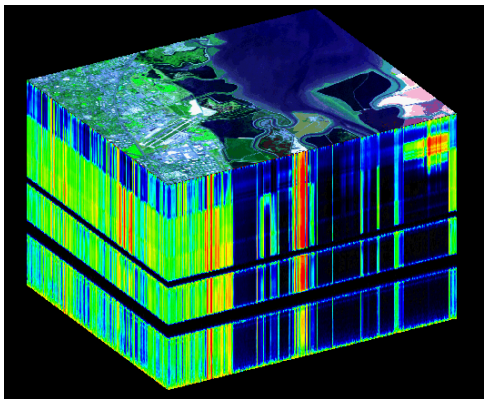

FIGURE 0.1.1. AVIRIS image cube of Moffet Field in California. The sensor has 224 channels along visible and near infrared bands.

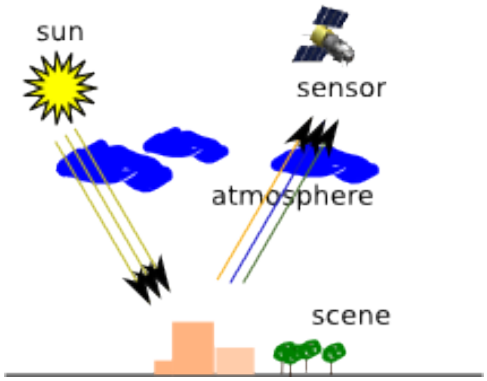

FIGURE 0.1.2. Hyperspectral imaging system (HSI).

0.1.2. Spectral Unmixing. The hyperspectral imagery could be seen as a mixing model where an hyperspectral image is the result of the linear combination of the pure spectral signature of ground components, named endmembers, with a fractional abundance matrix. Let $\mathbf{E} = [\mathbf{e_1}, \ldots, \mathbf{e_p}]$ be the pure endmember signatures (normally corresponding to macroscopic objects in scene, such as water, soil, vegetation, ...) where each $\mathbf{e_i} \in \Re^L$ is an L-dimensional vector. Then, the hyperspectral signature $\mathbf{r}$ at each pixel on the image is defined by the expression:

$$\mathbf{r} = \mathbf{s} + \mathbf{n} = \sum_{i=1}^{p} \mathbf{e_i}\phi_i + \mathbf{n} \tag{0.1.1}$$

where the hyperspectral signature $\mathbf{r}$ is formed by the sum of the pixel's signal $\mathbf{s}$ and an independent additive noise component $\mathbf{n}$. ϕ is the p-dimensional vector of fractional abundances at given pixel . This equation can be generalized to the full image by

$$\mathbf{H} = \mathbf{E\Phi} + \mathbf{n} \tag{0.1.2}$$

where $\mathbf{H}$ is the hyperspectral image and $\mathbf{\Phi}$ is a matrix of fractional abundances.

There are two constraints in the equation 0.1.1, the abundance non-negative constraint (ANC) 0.1.3 and the abundance sum-to-one constraint (ASC) 0.1.4, respectively defined as

$$\phi_i \geq 0, \text{ for all } 1 \leq i \leq p \tag{0.1.3}$$

$$\sum_{i=1}^{p} \phi_i = 1 \tag{0.1.4}$$

In the field of hyperspectral image processing, Spectral Unmixing [**24**] is the procedure by which the measured spectrum of a mixed pixel is decomposed into a collection of constituent spectra (endmembers) and a set of corresponding fractions (abundances), that indicate the proportion of each endmember present in the pixel. This is usually done by the computation of the fractional contribution of endmembers given the ANC and ASC restrictions. These restrictions require careful numerical methods for the computation of the fractional abundances [**48, 20**] when the endmembers are known, involving constrained non-negative least squares estimation, which can be a very computationally expensive process by itself. The image spectral signatures could be a good characterization of the image but, additionally, the abundance matrices could be used as spatial information about the image. The idea of using the spectral unmixing fractional abundances as a kind of feature extraction for classification purposes was introduced in [**14**]. Spectral Unmixing can be used for target detection, thematic map building and unsupervised segmentation.

The key problem for spectral unmixing is the definition of the set of endmember spectral signatures. A library of known pure ground signatures or laboratory samples could be used. However, this poses several problems, such as the illumination invariance, the difference in sensor intrinsic parameters and the *a priori* knowledge about the material composition of the scene. Besides the methodological questions, this approach is not feasible when trying to process large quantities of image data. Current approaches try to induce the endmembers from the image data. They either try to select some image pixel spectra as the best approximation to the endmembers in the image [**34, 15**], or to compute estimations of the endmembers on the basis of the transformations of the image data (i.e. [**10, 23**]). The latter is the predominant class of techniques in the literature. Previous reviews found in the literature [**35**] make some emphasis on the degree of automation to classify the algorithms. In this review the emphasis will be on the computational foundations, assuming that user interaction must be minimal or null. We distinguish three fundamental approaches:

- *Geometric approaches*, that try to find a simplex that covers the image data.
- *Lattice computing approaches*, that use some kind of lattice theoretic formalism or mathematical morphology approach.
- *Heuristic approaches*, that are not very rigorously formalized under a theoretical framework.

The hyperspectral images have a very high dimensionality and they suffer from the typical problems derived of the well known curse of dimensionality [**26**]. Information is usually spanned over a lower dimensionality subspace. That's why the analysis of hyperspectral data is usually preceded for a dimensionality reduction what also improves the computational efficiency and the data storage capability. However, the nature of hyperspectral data makes traditional dimensionality reduction techniques be not as successful as they are with multispectral data, and new methods focused in the hyperspectral data have been developed.

The number of spectral signatures that form an hyperspectral image is usually unknown. Recently, a new concept denoted as Virtual Dimensionality (VD) [**33, 9, 6**] has been used for automated search of the optimal number of endmembers in an hyperspectral image.

There are some problems that we will not touch in deep in this review. The problem of the endmember induction algorithms initialization is discussed in [**33**], where an Endmember Initialization Algorithm (EIA) is proposed. Most endmember induction algorithms are quite computationally expensive, so a mention is due to the efforts to obtain distributed implementations [**36**] that may help them to be a feasible approach for real-life applications. The use of off-the-shelf Graphical Processing Units [**47**] is a low cost way to obtain substantial speed-ups.

The outline of the paper is as follows: section 0.2 describes some techniques to reduce the dimensionality of hyperspectral data, section 0.3 describes the Virtual Dimensionality concept and some approaches to find the correct number of endmembers in image. Next sections present the different approaches for endmember induction: section 0.4 describes some geometrically oriented methods, section 0.5 describes methods based on lattice computing or mathematical morphology and section 0.6 describes some heuristic methods. Finally, some concluding remarks are given in section 0.7.

0.2. Dimensionality Reduction

Hyperspectral data sets consist of hundreds to thousands of spectral bands what is a huge amount of data to explore. [**25, 26**] showed that high-dimensional data spaces have some rather unusual and unintuitive characteristics, that is, high-dimensional space is mostly empty and data are usually concentrated in a lower dimensional structure. Because hyperspectral images contain a lot of spectral redundancy and data are localized in a lower-dimensionality subspace, the first step is usually to apply dimensionality reduction techniques to hyperspectral data what also deals to improvements in computational performance and data storage.

The most widely used dimensionality reduction algorithms are detailed below. Principal Component Analysis (PCA) [**21**] computes the Karhunen-Loéve transform that maximizes data variance while Maximum Noise Fraction (MNF) [**17**] or Noise-Adjusted Principal Components (NPCA) [**28**], which are mathematically equivalents, search for the projection that maximizes the signal-to-noise ratio (SNR). Singular Value Decomposition (SVD) [**19**] searches for the projection of maximum signal power. The number of dimensions to be retained is usually given by a criteria based on the number of eigenvalues needed to represent a certain percentage of energy to be preserved. PCA, SVD and MNF's major issue is that many subtle material substances that are uncovered by very high spectral resolution hyperspectral imaging sensors cannot be characterized by second-order statistics [**52**].

Independent Component Analysis (ICA) [**22**] uses high-order statistics to overpass this concern but however, its source statistical independence premise is generally violated in hyperspectral data and it can lead to poor results over some conditions [**31**]. Another dimensionality reduction method based on the selection of a subspace of the orthogonally transformed data is the Maximum Autocorrelation Factor (MAF) [**49**]. In [**38**], a novel multilineal algebra method is proposed to jointly archive denoising and dimensionality reduction. This method, referred to as LRTA_{dr}-(K_1,K_2,D_3), performs a Lower Rank Tensor Approximation and then an spectral Dimensionality Reduction.

0.2.1. Principal Component Analysis (PCA). Principal Component Analysis (PCA) is an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by any projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on. PCA identifies orthogonal axes for dimensionality reduction by performing an eigendecomposition of the sample covariance matrix of the data

$$\hat{\Sigma} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x_n} - \hat{\mu})(\mathbf{x_n} - \hat{\mu})^T \tag{0.2.1}$$

where $\hat{\mu}$ is the sample mean vector. The resulting eigendecomposition can be expressed as

$$\hat{\Sigma} = U\Lambda U^T \tag{0.2.2}$$

where U is the matrix of eigenvectors and Λ the diagonal matrix of eigenvalues. The magnitude of the eigenvalues indicates the power of the data along each associated eigenvector, so the most used criteria to reduce the data dimensionality is to reorder the eigenvectors according to their respective eigenvalues and keep as many eigenvectors as needed to represent a given percentage of the data variance.

0.2.2. Maximum Noise Fraction (MNF). PCA is theoretically the optimum transform for a given data in least square terms but it works independently of any estimate of the noise in the signal. Maximum Noise Fraction (MNF) incorporates information about the sources of additive noise in addition to the covariance of the data. The Noise-adjusted Principal Component Analysis (NPCA) is mathematically equivalent to the MNF where the first applies PCA to the previously noise-whitened data and the second transforms the data to maximize the signal-to-noise ratio. If the estimated noise covariance is given by $\hat{\Sigma_n}$ and the estimated signal covariance is $\hat{\Sigma_s}$, the MNF maximizes the signal-to-noise ratio by

$$\frac{U^T\hat{\Sigma_n}U}{U^T\hat{\Sigma_s}U} \tag{0.2.3}$$

The projection axes defined by U are given by the eigendecomposition of $\hat{\Sigma_n}/\hat{\Sigma_s}$, which are not necessarily orthogonal.

0.2.3. Singular Value Decomposition (SVD).

0.2.4. Independent Component Analysis (ICA). Independent Component Analysis (ICA) main idea is to assume that data are formed by a set of independent sources so it tries to find that sources by their statistical independence. To do it one more assumption has to be made, that at most one of the sources is Gaussian, due to the fact that a linear mixture of Gaussian sources is still a Gaussian source. Let $\mathbf{x}$ be the hyperspectral data expressed by

$$\mathbf{x} = A\mathbf{s} \tag{0.2.4}$$

where A is a mixing matrix and $\mathbf{s}$ is a d-dimensional signal source vector. ICA looks for a demixing matrix W that separates the signal source vector $\mathbf{s}$ into a set of statistically independent sources. Several different criteria have been proposed to measure source independence as minimizing mutual information [**51**], maximizing non-gaussianity [**22**] or by maximum likelihood estimation.

0.2.5. Maximum Autocorrelation Factor (MAF). PCA orthogonalization computes eigenvectors U and eigenvalues Λ by the spectral decomposition $U\Sigma(0)U^T = \Lambda$ for the matrix of multivariate covariances $\Sigma(0)$ at zero lag distance. This approach use is limited to multivariate covariances $\Sigma(h)$ with intrinsic coregionalization, that is, when all covariance and cross-covariance structures are proportional to each other

$$\Sigma(h) = Bc(h) \tag{0.2.5}$$

where $c(h)$ is the elementary covariance structure for all attributes.
The Linear Model of Coregionalization (LMC) given by

$$\Sigma(h) = \sum_{i=1}^{q} B_i c^i(h) \tag{0.2.6}$$

is an alternative to 0.2.5 where the coregionalization matrices B_i are diagonalized by the spectral decomposition $U_i\Sigma_i U_i^T = \Lambda^i$ and the scalar elementary covariance $c^i(h)$ with unit variance just multiply the eigenvalues in $c(h)$, unaffecting the eigenvectors. This approach does not provide a single matrix that can orthogonalize $\Sigma(h)$ for all lag distances. This approach is related to discriminant analysis of diagonalization of an asymmetric covariance matrix as

$$U[B_1 B^{-1}]U^T = \Lambda \tag{0.2.7}$$

[**49, 50**] introduces the Minimum/Maximum Autocorrelation Factors (MAF) method that allows the orthogonalization of data vector $Z(x)$ when the sample matrix covariance can be modelled by up to two nested structures in the LMC model in 0.2.6. MAF method developes a rotation that maximizes and minimizes the autocorrelation of the factors, this is

$$U_2\Gamma(\Delta)(U_1 B U_1^T)^{-1}U_2^T = \Lambda \tag{0.2.8}$$

where $\Gamma(\Delta)$ is the matrix variogram for non-standarized PCA factor scores $Y(x) = Z(x)U$ at a lag distance Δ smaller than the range, $U_1\Sigma U_1^T$ is a matrix

of PCA eigenvalues, B is the sum of the coregionalization matrices for the original attributes, same as $\Sigma(0)$, U_2 is the matrix of eigenvectors for the standarized variogram $\Gamma(\Delta)(U_1 B U_1^T)^{-1}$ at some lag Δ, and Λ is a matrix of eigenvalues.

The multivariate matrix of covariance is

$$\Sigma(h) = B_1 c^1(h) + B_2 c^2(h) \tag{0.2.9}$$

For $h = 0$ this is $\Sigma = B_1 + B_2$, and 0.2.9 becomes

$$\Sigma(0) = B_1 c^1(0) + B c^2(0) - B_1 c^2(0) \tag{0.2.10}$$

Computing the eigenvectors gives a symmetric rotation

$$U_1 B U_1^T = U_1 [B_1 + B_2] U_1^T \tag{0.2.11}$$

Scaling the eigenvectors by the standard deviation of the factors yields

$$A_1 = U \Lambda^{-\frac{1}{2}} \tag{0.2.12}$$

that standarized the PCA factors so the covariance matrix for the PCA scores for lag distance zero $\Sigma_Y(0)$ is the identity matrix that cannot be affected by any subsequent rotation. The PCA factors A_1 are applied to the coregionalization matrices as

$$A_1 B A_1^T = A_1 [B_1 + B_2] A_1^T \tag{0.2.13}$$

Since the eigenvalues are the variances of the factors, the factor scores $Y(x)$ of 0.2.7 are standarized by

$$\Sigma_Y(h) = A_1 B_1 A_1^T c^1(h) + A_1 B A_1^T c^2(h) - A_1 B_1 A_1^T c^2(h) \tag{0.2.14}$$

$\Sigma_Y(0)$ is a diagonal matrix so 0.2.14 becomes

$$\Sigma_Y(h) = V_1 c^1(h) + (I - V_1) c^2(h) \tag{0.2.15}$$

where $V_1 = A_1 B_1 A_1^T$ is used to denote the new coregionalization matrices. 0.2.15 shows that the variances of two non-orthogonal nested components are complementary and that the cross-variances have opposite signs to provide the identity matrix $\Sigma_Y(0)$.

The matrix $\Sigma_Y(h)$ is usually asymmetric. MAF builds on the assumption that $\Sigma_Y(h)$ can be substituted by the multivariate matrix variogram for factors that is symmetric

$$\Gamma_Y(h) = I - \frac{\Sigma_Y(h)^T + \Sigma_Y(h)}{2} \tag{0.2.16}$$

The matrix V_1 is made symmetric which yields

$$\Gamma_Y(h) = I - \frac{(V_1^T + V_1)(c^1(h) - c^2(h))}{2} - I c^2(h) \tag{0.2.17}$$

Assuming a lag distance $h = \Delta$ and $\Delta \neq 0$, MAF are obtained by a second computation of eigenvectors as

(0.2.18)
$$\Gamma_{MAF}(\Delta) = U_2 \Gamma_Y(\Delta) U_2^T = I - \frac{Q_2(V_1^T + V_1)}{2} \times Q_2^T(c^1(\Delta) - c^2(\Delta)) - Ic^2(\Delta)$$

0.2.6. LRTA$_{\mathbf{rd}}$-(K$_\mathbf{1}$,K$_\mathbf{2}$,D$_\mathbf{3}$). LRTA$_{\mathrm{rd}}$-(K$_1$,K$_2$,D$_3$) performs spatial lower rank approximation and spectral dimensionality reduction by the use of the Tucker3 decomposition that generalizes Singular Value Decomposition (SVD) for multilineal algebra. From this point of view an hyperspectral image is seen as a third-order tensor, two ways for rows and columns, and one way for spectral band, denoted by $X \in \Re^{I_1 \times I_2 \times I_3}$, with elements arranged as x_{i_1,i_2,i_3} where $i_1 = 1, \ldots, I_1$, $i_2 = 1, \ldots, I_2$, $i_3 = 1, \ldots, I_3$ and $\Re$ is the real manifold. Each index is called way or mode, and the number of levels on one mode is called dimension of that mode. The n-mode vectors are the I_n-dimensional vectors obtained from a tensor by varying index i_n while keeping the other indexes fixed. The so-called n-mode flattened matrix X_n of X its such that its columns are the n-mode vectors. The n-mode rank of multiway data $X \in \Re^{I_1 \times I_2 \times I_3}$, denoted by $Rank_n(X)$ is defined as the dimension of the vector space generated by the I_n-dimensional vectors obtained from X by varying index i_n while keeping the other indexes fixed. X is called a rank-(K$_1$,K$_2$,K$_3$) tensor if $Rank_n(X) = K_n$ for all $n = 1, 2, 3$.

It's assumed that each spectral band of an hyperspectral image is impaired by additive noise. This additive noise can also be represented as a three-way array N. Thus an hyperspectral image can be represented as $H = X + N$ where H, X and N are three-way array data of $\Re^{I_1 \times I_2 \times I_3}$. The purpose of LRTA-(K$_1$,K$_2$,K$_3$) is to find the lower rank-(K$_1$,K$_2$,K$_3$) multiway data $\hat{X}$, with $K_n < I_n$, for all $n = 1, 2, 3$, which minimizes the quadratic Frobenius norm $\left\| X - \hat{X} \right\|_F^2$. The best lower rank-(K$_1$,K$_2$,K$_3$) multiway approximation of X is expressed as

(0.2.19)
$$\hat{X} = X \times_1 P^{(1)} \times_2 P^{(2)} \times_3 P^{(3)}$$

where $P^{(n)} = U^{(n)} U^{(n)^T}$, with $n = 1, 2, 3$ and being $U^{(n)} = [u_1, \ldots, u_{K_n}]$ the projector upon the n-mode signal space. $P^{(n)}$ is archieved after an Alternating Least Squares (ALS) algorithm convergence [**27**].

LRTA-(K$_1$,K$_2$,K$_3$) is then turned into an spectral dimensionality reduction tool by incorporating to it the principles of PCA$_{\mathrm{dr}}$-(D$_3$). The aim of PCA dimensionality reduction is to extract an small number $D_3 < I_3$ of features, called principal components, generating a reduced matrix $Z \in \Re^{D_3 \times p}$

(0.2.20)
$$Z = H \times_3 \Lambda^{-1/2} U^{(3)^T}$$

LRTA$_{\mathrm{rd}}$-(K$_1$,K$_2$,D$_3$) jointly reduces the dimensionality of the spectral mode and protects the information along the spatial modes onto lower (K$_1$,K$_2$)-dimensional subspaces. The LRTA$_{\mathrm{rd}}$-(K$_1$,K$_2$,D$_3$) model reads

(0.2.21)
$$Z = H \times_1 P^{(1)} \times_2 P^{(2)} \times_3 \Lambda^{-1/2} U^{(3)^T}$$

0.3. Virtual Dimensionality

Dimensionality reduction is usually the first step to analyze hyperspectral images, but find the correct subspace dimensionality in which the hyperspectral data are localized is a very challenging problem. According to the definition given by [**13**], the Intrinsic Dimensionality (ID), also defined as effective dimensionality, is the minimum number of parameters required to account for the observed properties of the data. However, the traditional dimensionality reduction techniques seen in section 0.2 are difficult to apply to hyperspectral imagery and they may be not effective even if they are applicable since the ID in hyperspectral images is considerably smaller than the component dimensionality given by hundreds or thousands of bands and because the effect of interference in hyperspectral sensors is more serious than noise, and signal-to-interference ratio (SIR) has more impact on hyperspectral applications than signal-to-noise ratio (SNR). In [**8**] a new PCA-based algorithm, the Interference and Noise-Adjusted PCA (INAPCA) is developed to take into account this two problems, the high dimensionality of the data and the interferences that introduce hyperspectral sensors, and like this, find the correct ID for hyperspectral images.

[**6, 9**] introduces a new concept to deal with these difficulties, the Virtual Dimensionality (VD), which is the minimum number of spectrally distinct signal sources that characterize the hyperspectral data from a perspective view of target detection and classification. These signal sources may include unknown interfering sources, which cannot be identified by a priori knowledge. These works also include three eigen-thresholding based methods to determine VD of hyperspectral data, they all derived from the Neyman-Pearson detection theory.

A new minimum mean square error-based approach to infere the signal subspace in hyperspectral imagery is presented in [**4**]. The method, denoted as Hyperspectral signal identification by minimum error (HySime) infers the data subspace by minimizing the sum of the projection error power with the noise power, which are, respectively, decreasing and increasing functions of the subspace dimension.

0.3.1. Interference and Noise-Adjusted PCA (INAPCA). A signal model corrupted by noise is formulated by

$$\mathbf{x} = \mathbf{s} + \mathbf{n} \tag{0.3.1}$$

which is the base of signal-to-noise ratio based methods as NAPC. Separate interference from the model given by 0.3.1 helps to process hyperspectral images where interference is highly present due to the hyperspectral sensors high spectral resolution. A signal-to-interference plus noise ratio (SINR) model can be expressed by

$$\mathbf{x} = \mathbf{s} + \mathbf{i} + \mathbf{n} \tag{0.3.2}$$

[**8**] presents two approaches referred as SINR-PCA and IANW-PCA for the INACA to take care of the interference and noise prior to the use of PCA and NPCA.

Signal-to-Interference Plus Noise Ratio-Based PCA (SINR-PCA). The Signal-to-Interference Plus Noise Ratio-Based PCA (SINR-PCA) method is an improved version of NAPC transform where the SNR is replaced by SINR. Here, the noise

energy is assumed to be insignificant compared to the interference energy so SINR is almost equal to SIR. SINR-PCA adds interference to the linear mixing model formulation of hyperspectral images 0.1.2, so it is rewritten as

$$H = E\Phi + I\Psi + \eta \tag{0.3.3}$$

where $I = [\mathbf{i_1}, \ldots, \mathbf{i_m}]$ is a set of interferers with each interference signature $\mathbf{i_j}$ being a vector of same dimensionality than signal sources, and Ψ is a matrix of the fractional abundances of the interferers subject to he ANC and ASC constraints. Based on 0.3.3 the SINR-PCA method consists of three steps. First it applies an orthogonal subspace projector $P_E^{\perp}$ [**18**] to the data to obtain the desired interference-plus-noise subspace (the signal annihilated space) $\langle E \rangle^{\perp}$. In second stage the interference/noise covariance matrix, Σ_{i+n}, is obtained from $\langle E \rangle^{\perp}$. Finally, Σ_{i+n} is whitened before apply PCA transform to the data (NAPC method).

Interference-Annihilated Noise-Whitened PCA (AINW-PCA). The Interference-Annihilated Noise-Whitened PCA (AINW-PCA) treats the interference as an unwanted source and annihilates it before a PCA or a NPCA transform is applied. The AINW-PCA algorithm has also three steps. Firstly AINW-PCA finds the interference-annihilated subspace $\langle I \rangle^{\perp}$ by applying an orthogonal subspace projector $P_I^{\perp}$. Secondly, noise covariance matrix $\Sigma_{(EI)^{\perp}}$ is obtained from space $\langle EI \rangle^{\perp}$ by previously apply an $P_{EI}^{\perp}$ orthogonal subspace projector. Finally the NAPC transform is applied to the orthogonal complement space of I, $\langle I \rangle^{\perp}$, using the $\Sigma_{(EI)^{\perp}}$ covariance matrix.

0.3.2. Neyman-Pearson detection theory-based eigen-thresholding analysis (HFC method). The main idea of the HFC (Harsanyi-Farrand-Chang) method is that if an hyperspectral image's band doesn't contain a signal source, its sample correlation matrix eigenvalues and its sample covariance matrix eigenvalues will only reflect noise energy and so, they must be equal. Using a Neyman-Pearson test where the null hypothesis represents the case when the eigenvalues difference is zero, and the alternative hypothesis represents that the difference is greater than zero, the number of times that the test fails indicates how many signal sources are present in the image.

Being $R_{L\times L}$ the sample correlation matrix and $K_{L\times L}$ the sample covariance matrix, and $\left\{\hat{\lambda_1} \geq \hat{\lambda_2} \geq \ldots \geq \hat{\lambda_L}\right\}$ and $\{\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_L\}$ the ordered eigenvalues of $R_{L\times L}$ and $K_{L\times L}$ respectively. Assuming that signal sources have positive energies and noise variance in band l is given by σ_l^2, the problem of finding the VD can be formulated as a binary hypothesis as follows:

$$\begin{cases} H_0 : \ z_l = \hat{\lambda_l} - \lambda_l = 0 \\ H_1 : \ z_l = \hat{\lambda_l} - \lambda_l > 0 \end{cases} \qquad for\ l = 1, 2, \ldots, L \tag{0.3.4}$$

$\hat{\lambda_l}$ and λ_l can be modelled under hypothesis H_0 and H_1 as random variables with their asymptotic conditional probability densities given by

$$p_0(z_l) = p(z_l|H_0) \cong N(0, \sigma_l^2) \tag{0.3.5}$$

and

$$p_1(z_l) = p(z_l|H_1) \cong N(\mu, \sigma_l^2) \tag{0.3.6}$$

where μ_l is an unknown constant and σ_l^2 when samples are large enough, as demonstrated in [**6, 8**], is given by:

$$\sigma_l^2 = Var[z_l] \cong \sigma_{\hat{\lambda}_l}^2 + \sigma_{\lambda_l}^2 \qquad as\ N \to \infty \tag{0.3.7}$$

Defining the false alarm probability, P_F, and the detection probability, P_D, as

$$P_F = \int_{\tau}^{\infty} p_0(z_l)dz \tag{0.3.8}$$

$$P_D = \int_{\tau}^{\infty} p_1(z_l)dz \tag{0.3.9}$$

the use of the Neyman-Pearson detector in each band gives the VD.

This method can be modified by including a noise-whitening process to remove the second-order statistical correlation. The resulting method is referred to as Noise-Whitened HFC (NWHFC).

0.3.3. Noise Subspace Projection (NSP). The effectiveness of HFC and NWHFC methods relies in the condition that data is large enough to validate the use of 0.3.7. Noise Subspace Projection (NSP) was introduced in [**6, 8**] to resolve the cases where this condition is not fulfilled. NSP requires the computation of sample covariance matrix $K_{L\times L}$ or sample correlation matrix, $R_{L\times L}$, but not both.

Given $K_{L\times L}$, it can be represented by $K_{L\times L} = K_s + K_n$ where K_s and K_n are the covariance matrices of signals and noise respectively. Using the eigendecomposition of $K_{L\times L}$ it can be expressed as

$$K_{L\times L} = \sum_{l=1}^{VD}(\lambda_l + \sigma_l^2)\mathbf{u_l}\mathbf{u_l^T} + \sum_{l=VD+1}^{L}(\sigma_l^2)\mathbf{u_l}\mathbf{u_l^T} \tag{0.3.10}$$

where $\{\mathbf{u_l}\}_{l=1}^{VD}$ and $\{\mathbf{u_l}\}_{l=VD+1}^{L}$ represent two sets of orthonormal vectors to span signal space and noise space respectively. λ_l and σ_l^2 are the energies of signal sources and noise respectively in the l-th band. Assuming that the energies of signals are greater than noise energies the problem becomes the following binary composite hypothesis test

$$\begin{cases} H_0 :\ z_l = \mu_l = 1 \\ H_1 :\ z_l = \mu_l = \lambda_l/\sigma_l + 1 > 1 \end{cases} \qquad for\ l = 1, 2, \ldots, L \tag{0.3.11}$$

being

$$p_0(z_l) = p(z_l|H_0) \cong N(1, \sigma_l^2) \tag{0.3.12}$$

$$p_1(z_l) = p(z_l|H_1) \cong N(\mu_l, \sigma_l^2) \tag{0.3.13}$$

where σ_l^2 is given by $\sigma_{\lambda_l}^2$.

0.3.4. Hyperspectral signal identification by minimum error (HySime). Hyperspectral signal identification by minimum error (HySime) infers the signal subspace by minimizing the sum of the projection error power with the noise power, which are, respectively, decreasing and increasing functions of the subspace dimension. If the subspace dimension is underestimated, the projection error power is dominant; if it is overestimated, the dominant term is the noise power. The algorithm's details are presented at continuation.

The first step is to identify a set of orthogonal directions of which an unknown subset spans the signal subspace. This subset is determined by seeking the minimum square estimation between the data $\mathbf{y}$ and the original signal $\mathbf{x}$ taking into account that the data is a noisy projection of the original signal

$$\mathbf{y} = \mathbf{x} + \mathbf{n} \tag{0.3.14}$$

where noise is assumed to be zero-mean Gaussian distributed $n \sim N(0, \hat{R_n})$ with the covariance matrix $\hat{R}_n$ given by

$$\hat{R}_n = \frac{1}{N} \sum_i (\hat{\xi}_i \hat{\xi}_i^T) \tag{0.3.15}$$

and where $\hat{\xi}_i$ is the estimated noise. $\hat{\xi}_i$ is usually calculated by shift difference (nearest neighbor difference), however [**4**] follows a multiple regression methodology-based approach which outperforms the shift difference method. Let $E = [e_1, e_2, \ldots, e_L]$ be the eigenvectors of the signal sample correlation matrix $\hat{R}_x = [\hat{x_1}, \ldots, \hat{x_N}][\hat{x_1}, \ldots, \hat{x_N}]^T/N$ rewritten as $\hat{R}_x = E\Sigma E^T$ and estimated by

$$\hat{R}_x = \frac{1}{N} \sum_i ((y_i - \hat{\xi}_i)(y_i - \hat{\xi}_i)^T) \tag{0.3.16}$$

Given a permutation $\pi = \{i_1, \ldots, i_L\}$ of indices $i = 1, \ldots, L$ let space $\Re^L$ be decomposed into two orthogonal subspaces, the k-D subspace $\langle E_k \rangle$ spanned by $E_k \equiv [e_{i_1}, \ldots, e_{i_k}]$ and its orthogonal complement $\langle E_k \rangle^\perp$ spanned by $E_k \equiv \left[e_{i_{k+1}}, \ldots, e_{i_L}\right]$. Let $U_k = E_k E_k^T$ be the projection matrix onto $\langle E_k \rangle$ and $\hat{x_k} \equiv U_k y$ the projection of the observed spectral vector y onto the subspace $\langle E_k \rangle$. The first and the second order moments of $\hat{x_k}$ given x are

$$E[\hat{x_k}|x] = U_k E[y|x] = U_k E[x + n|x] = U_k x = x_k \tag{0.3.17}$$

$$E[(\hat{x_k} - x_k)(\hat{x_k} - x_k)^T|x] = E[(U_k y - U_k x)[(U_k y - U_k x)^T|x] = U_k \hat{R}_n U_k^T \tag{0.3.18}$$

The minimum squares estimation between x and $\hat{x_k}$ is given by

$$mse(k|x) = E[(x - \hat{x_k})^T (x - \hat{x_k})|x] = (x - x_k)^T (x - x_k) + \left(U_k \hat{R}_n U_k^T\right)^T \tag{0.3.19}$$

Computing the mean of 0.3.19 with respect to x:

$$mse(k) = E[(U_k^{\perp}x)^T(U_k^{\perp}x)] + \left(U_k\hat{R}_nU_k^T\right)^T = (U_k^{\perp}R_y)^T + 2(U_k\hat{R}_n)^T + c \tag{0.3.20}$$

where c is an irrelevant constant. The criterion to estimate the signal subspace, X, is the minimization of 0.3.20 with respect to all the permutations $\pi = \{i_1, \ldots, i_L\}$ of size L and with respect to k, with the correlation matrix R_y replaced with the sample correlation matrix $\hat{R}_y = YY^T/N$.

$$\hat{X} = \left\langle \left[e_{\hat{i_1}}, \ldots, e_{\hat{ik}} \right] \right\rangle \tag{0.3.21}$$

$$(\hat{k}, \hat{\pi}) = \arg\min_{k,\pi} \left\{ (U_k^{\perp}\hat{R}_y)^T + 2(U_k\hat{R}_n)^T \right\} \tag{0.3.22}$$

Given that $U_k = E_kE_k^T$ is a projection matrix and that $(AB)^T = (BA)^T$,0.3.22 can be rewritten as

$$(\hat{k}, \hat{\pi}) = \arg\min_{k,\pi} \left\{ c + \sum_{j=1}^{k} (-p_{i_j} + 2\sigma_{i_j}^2) \right\} \tag{0.3.23}$$

where c is an irrelevant constant, $p_{i_j} = e_{i_j}^T\hat{R_y}e_{i_j}$ and $\sigma_{i_j}^2 = e_{i_j}^T\hat{R}_ne_{i_j}$. Being $\delta_{i_j} = -p_{i_j} + 2\sigma_{i_j}^2$, the minimization is archived simply by including all the negative terms δ_i, with $i = 1, \ldots, L$, in the sum. The virtual dimensionality is then given by the number of negative terms $\hat{\delta}_i$.

0.4. Geometric-Based Induction Methods

Geometric methods follow the formal definition of the endmembers, they search for the vertices of a convex set that covers the image data. Because the distribution of the data in the hyperspace is usually tear-shaped they look for the minimum simplex that covers all the data. Unless said otherwise, the algorithms search for a prefixed number of endmembers, defined by the user.

The first such methods is the Minimum Volume Transform, proposed by [**12**] that introduces two non-orthonormal transforms, the dark-point-fixed (DPF) transform and the fixed-point-free (FPF) transform that map the data onto the minimal simplex that contain all the data points. One of the earliest approaches is the N-FINDR algorithm proposed in [**53**]. The N-FINDR algorithm is a selection algorithm. Its works are described as follows: it starts with a random collection of image pixel spectra, corresponding to the initial set of endmembers. Then, each of the remaining image pixels is considered as a candidate to replace each endmember, if doing so the volume of the simplex increases, then it is accepted as the new endmember. The process ends when no more replacements are possible. The Convex Cone Analysis (CCA) [14] is based on the fact that the vectors formed by discrete radiance spectra are linear combinations of nonnegative components, and they lie inside a nonnegative convex region. The object of CCA is to find the boundaries of this convex region, which can be used as endmember spectra. The approach followed in [**1**] searches for the optimal simplex using a simulated annealing algorithm (SA) whose state configuration is given by the partition of

the faces of the convex hull of the image pixel spectra. The partition in the configuration space defines a simplex covering the image data whose vertices are the candidate endmembers. The objective function minimized is the simplex volume. This approach is followed by the generation of endmember bundles that allow the computation of bounds on the abundance images. The Iterated Constrained Endmembers (ICE) [**2**] algorithm performs the minimization of a regularized residual sum of squares (RSS). The regularization term is the volume of the simplex. The name of the algorithm comes from the minimization schema applied. Given that the free parameters are the endmembers and the proportions (abundances) for each pixel the algorithm iterates the solution of the tow interleaved and interdependent minimization problems (much like in an Expectation Maximization process): first the proportions are computed by quadratic programming problem solving assuming that the endmembers are known, then the endmembers are computed as the direct minimization of the RSS functional. The addition of an sparsity promoting term in the RSS functional gives way to SPICE [**54**]. This sparsity promoting term is derived as the substitution of a Gaussian prior by a Laplacian prior in a Bayesian formulation of the RSS functional. The SPICE algorithm allows the selection of the appropriate number of endmembers based on the sparsity measure. The Vertex Component Analysis algorithm (VCA) is presented in [**32**]. The algorithm is unsupervised and exploits that the affine transformation of a simplex is also a simplex. It works with projected and unprojected data. The algorithm iteratively projects data onto a direction orthogonal to the subspace spanned by the endmembers already determined. The new endmember signature corresponds to the extreme of the projection. The algorithm iterates until all endmembers are exhausted. In [**11**] a simplex-based endmember extraction algorithm, called Simplex Growing Algorithm (SGA), is presented. It is a sequential algorithm to find a simplex with the maximum volume every time a new vertex is added. SGA improves N-FINDR by including a process of growing simplexes one vertex at a time until the desired number of vertices is reached, which results in a high computational complexity reduction; and by selecting an appropiate initial vector to avoid the use of random vectors as initial condition, which produces different sets of final endmembers if different sets of randomly generated initial endmembers are used. In [**30**] a method for endmember extraction for highly mixed data, when there are not pure pixels in the hyperspectral image, is presented. The proposed method, called Minimum Volume Constrained Nonnegative Matrix Factorization (MVC-NMF) takes advantage of the fast convergence of NMF schemes and at the same time eliminates the pure-pixel assumption. It consists in the reformulation of an NMF cost function introducing a volume regularization term, much like the ICE, substituting the RSS by the NMF criteria.

0.4.1. Minimum Volume Transform (MVT). In [**12**], two non-orthonormal Minimum Volume Transforms methods, the dark-point-fixed (DPF) transform and the fixed-point-free (FPF) transform are introduced. Both map the data onto the minimal volume simplex that contains all the data points. Both transforms require a dimensionality reduction before been applied, being MNF the method used in [**12**].

0.4.1.1. *Dark-Point-Fixed (DPF) transform.* The dark-point-fixed (DPF) transform takes advantage of the usually tear-drop-shaped distribution of the data, radiating away from the so-called dark point, the response to a target of nil reflectance

in all bands. DPF transform searches for the axes that enclose the data cloud with a minimum volume after translating the data origin to the darkpoint (generally not the origin, because of instrument settings and atmospheric backscatter). To do that, DPF transform projects the data from O to lie within the hyperplane

$$u^T x = 1. \tag{0.4.1}$$

Here, the objective is to determine the points on 0.4.1 such that the simplex $OP_1P_2 \dots P_n$ has the least possible volume. If P is the matrix whose ith column is the coordinate vector for P_i, then

$$u^T P = u^T = u^T Q \tag{0.4.2}$$

where Q is the inverse of P. The simplex volume is given by

$$\frac{abs(\det P)}{n!} \tag{0.4.3}$$

The condition for x to be within the face opposite O is that all the components of y should be nonnegative where $x = Py$. The task is then to maximize the quantity $abs(\det Q)$ subject to the nonnegative of Qx constraint.

The algorithm consists on first choose an initial Q and then refine it iteratively. The identity matrix I could be an initial choice for Q, but [**12**] suggests a way for a better initialization given in the 3-D case by

$$Q = \begin{bmatrix} 1-a & -a & -a \\ -b & 1-b & -b \\ -c & -c & 1-c \end{bmatrix} \tag{0.4.4}$$

being trivial its expansion to $n > 3$.

The refination of Q is made by varying one at a time the directions of the vectors forming its rows, while the remaining rows are rescaled but held fixed in direction. Geometrically, it implies to vary the orientation of one face, while preserving the others. In an algebraic form, if the kth row, q_{kj}, is going to be varied introducing an scale factor z_i for each remaining row i, by 0.4.2 the new value of q_{kj} will be $1 - \sum_{i \neq k} z_i q_{ij}$. To keep the nonnegative constraint the following conditions have to be considered

$$\sum_{i \neq k} q_{ij} x_j \geq 0 \qquad and \qquad z_i \sum_{i \neq k} q_{ij} x_j \geq 0 \tag{0.4.5}$$

In view of the value of q_{kj} and 0.4.1 is concluded that

$$\sum_{i \neq k} a_i z_i \leq 1 \tag{0.4.6}$$

where $a_i = \sum_j q_{ij} x_j$. The effect of variation over the determinant of Q is to multiply $\det Q$ by the product of the scale factors z_i. The problem is therefore reduced to maximizing the product of $n-1$ nonnegative variables z_i subject to 0.4.6.

0.4.1.2. *Fixed-Point-Free (FPF) transform.* If x is an n vector, the augmented vector associated to x is an $N = n + 1$ vector given by

$$X^T = (x^T, 1) \tag{0.4.7}$$

Let p_i, $0 \leq i \leq n$, the components of a full-rank simplex S in n space, and let P be the N-rowed square augmented matrix of S. The barycentric coordinates of x are $Y_0, \ldots, Y_n$ where

$$x = \sum_i Y_i p_i, \qquad \sum_i Y_i = 1 \tag{0.4.8}$$

Being Q the inverse of P, the barycentric coordinates can be resolved by $Y = QX$. Equation 0.4.2 is adapted to

$$E^T P = U^T \qquad hence \qquad E^T = U^T Q \tag{0.4.9}$$

denoting E the augmentation of the zero n vector and U the augmented vector associated to u.

The algorithm consists in maximizing the absolute determinant of a matrix Q subject to the nonnegative constraint of vectors QX. The method is analogous to DPF transform algorithm.

0.4.2. N-FINDR. [**53**] introduces the N-FINDR algorithm that works by inflating a simplex inside the data, beginning with a random set of pixels. Previously, data dimensionality has to be reduced to $n - 1$ dimensions, being n the number of endmembers searched for.

Let E be the matrix of endmembers augmented with a row of ones

$$E = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ e_1 & e_2 & \ldots & e_n \end{bmatrix} \tag{0.4.10}$$

where e_i is a column vector containing the spectra of the ith endmember. The volume of the simplex is proportional to the determinant of E

$$V(E) = \frac{abs(\det(E))}{(l-1)!} \tag{0.4.11}$$

The algorithm starts by selecting an initial random set of pixels as endmembers. Then for each pixel and each endmember, the endmember is replaced with the spectrum of the pixel and the volume recalculated by 0.4.11. If volume increases, the endmember is replaced by the spectrum of the pixel. The procedure ends when no more replacements are done. The algorithm needs of some random initializations to avoid local maxima.

0.4.3. Convex Cones Analysis (CCA). The idea of Convex Cone Analysis (CCA) introduced in [**23**] relies on the nonnegative values of radiance or mass spectra, so data formed by such spectra lies into a convex region. CCA tries to find the boundaries of this convex region as defined by its vertices, which are used as endmembers.

Given an $n \times m$ image with d bands, the hyperspectral cube is rearranged into an $nm \times d$ matrix S. The sample correlation matrix C is then obtained from the normalized spectral matrix S

$$C = S^T S \tag{0.4.12}$$

Then an eigendecomposition of C is made

$$C = U \Lambda U^T \tag{0.4.13}$$

where U is the matrix of eigenvectors and Λ the diagonal matrix of eigenvalues. Given the number of endmembers in image, p, CCA selects the eigenvectors corresponding to the p largest eigenvalues and looks for the boundaries of the convex cone, where the linear combinations of these eigenvectors are nonnegative

$$x = u_1 + \alpha_1 u_2 + \ldots + \alpha_{p-1} u_p \geq 0 \tag{0.4.14}$$

The coefficients can be multiplied by an scalar factor so as to make the coefficient of u_1 unity, giving $p-1$ free parameters. The eigenvector u_1 associated to the largest eigenvalue is aligned along the direction of the cone axis. The objective is to find a set of coefficients $\{\alpha_i\}_{i=1}^{p-1}$ that produces a linear combination containing $p-1$ elements of x that are exactly zero, with all the other elements being nonnegative. These points represent the corners of the convex cone.

To compute the convex cone the algorithm starts by rewriting 0.4.14 as

$$x = [u_1 \ldots u_p] \begin{bmatrix} 1 \\ \alpha_1 \\ \vdots \\ \alpha_{p-1} \end{bmatrix} = Ua \geq 0 \tag{0.4.15}$$

where u_i are d-dimensional column vectors. For $d > p$, $Ua = 0$ is an overdetermined system of linear equations. The boundary of the convex cone is the set of all solution vectors a that satisfy 0.4.15, or equivalently

$$\min(x) = 0 \tag{0.4.16}$$

where the minimum is taken over all $x_i \in x$, $i = 1, \ldots, d$. What is the same, a boundary occurs when at least one of the vector elements in the linear combination of eigenvectors is zero while the others are nonnegative. In resume, the algorithm searches through all possible combinations of $p-1$ bands to find all sets of coefficients $\alpha_1, \ldots, \alpha_{p-1}$ that satisfy 0.4.16. An small tolerance ε for negative numbers is introduced to allow for numerical errors.

0.4.4. Simulated Annealing (SA)-based method. [1] describes a method for constructing an uniquely simplex from a partition of the facets of the convex hull that encloses the data cloud. Let CV be the convex hull of a finite set of data points in a d-dimensional space. If the points in CV do not lie on a $(d-1)$-hyperplane, then they are the vertices of a set of $(d-1)$-simplexes completely covering the boundary of the convex closure of the data points and having the property that if

any two simplexes in the set intersect they do it along a common k-face ($k < d-1$). Each $(d-1)$-simplex in the set is named a facet of the CV.

Since a finite number of hull facets has only finitely many partitions, there are only finitely many simplexes the method can construct, and finding the one with the smallest volume is a combinatorial optimization problem resolved in the work by Simulated Annealing (SA).

A system configuration is a partition of the facets of the CV into $D+2$ sets, being D the data space dimensionality. Each of the first $D+1$ sets is nonempty, and its facets form a connected region on the CV, the last set contains all facets not in the union of the first $D+1$ sets. These $D+1$ sets determine the faces of an D-simplex containing the data. The algorithm begins by generating a random initial partition P_0. The algorithm iterates by generating random partitions and calculating the objective function given by the volume of the simplex computed for the current partition, V, then the partition is accepted if the difference between the volume of the current simplex and the smallest volume previously encountered (ΔV) is negative or zero, if it's positive the partition is accepted with a probability given by the Boltzmann probability distribution

$$P(\Delta V) = \exp(\frac{-\Delta V}{T}) \tag{0.4.17}$$

where T, referred to as the temperature, rules the annealing process. The value of T is initialized to the volume V_0 of the simplex generated by the initial partition P_0. After each $100NF$ iterations, where NF is the number of facets of the CV, T is decreased by $0.1V_0$, making smaller the probability of accepting a new partition. The algorithm stops when no partitions are accepted for a fixed value of T.

For a given partition the simplex is calculated by fitting an hyperplane orthogonally to the vertices of the facets of each first $D+1$ sets in the partition, and translating the $D+1$ hyperplanes until they enclose the data cloud. The simplex is determined by the intersections of the hyperplanes.

0.4.5. Iterated Constrained Endmembers (ICE). The Iterated Constrained Endmembers (ICE) method proposed in [**2**] searches for the simplex enclosing the dimensionality reduced data cloud by MNF method, that minimizes the following objective function

$$RSS_{reg} = (1-\mu)\frac{RSS}{N} + \mu V \tag{0.4.18}$$

where N is the number of samples in the data, RSS is the residual sum of squares, V is the sum (over the dimensionality reduced bands) of the variances of the simplex vertices and $0 \leq \mu \leq 1$ is a regularization term. The RSS formulation is given by

$$RSS = \sum_{j=1}^{d} (x_j - Pe_j)^T (x_j - Pe_j) \tag{0.4.19}$$

where P is an $M \times N$ matrix of proportions of the M endmembers, e_i, for all N pixels, x_k.

RSS_{reg} objective function incorporates the constraining of the size of the simplex, V, as additional information into the penalty term governed by the least squares error given the linear mixing model 0.1.2 and the restrictions 0.1.3 and 0.1.4. If $\mu \to 0$ the solution is the M-simplex in the hyperplane spanned by the first $M-1$ MNF bands, whose vertices have minimum total variance. On the other hand if $\mu \to 1$ all the endmembers converge to one point, the mean of the data, which has the minimum simplex volume.

ICE algorithm is an iterative minimization of 0.4.18. First, given endmember estimates, the proportion matrix P is calculated. Then, given P, 0.4.18 is minimized for each endmember e_j. Its conditional minimizer has the explicit solution

$$e_j = \left\{ P^T P + \lambda (I_M - \frac{11^T}{M}) \right\}^{-1} P^T x_j \tag{0.4.20}$$

where $\lambda = \frac{N\mu}{(M-1)(1-\mu)}$. The iterating process stops when the ratio of successive values of RSS_{reg} is less than a tolerance value, tol. Experimentally, [**2**] gives the following tentative values for the parameters of the ICE algorithm: $tol = 0.99999$ and $\mu = 0.05$.

0.4.6. SPICE. [**54**] introduces an extension of the ICE algorithm that incorporate sparsity-promoting priors to find the correct number of endmembers. The proposed algorithm is denoted as Sparsity Promoting Iterated Constrained Endmember detection algorithm (SPICE).

The SPICE algorithm incorporates a new term into ICE's objective function 0.4.18 which yields

$$RSS^*_{reg} = (1-\mu)\frac{RSS}{N} + \mu V + SPT \tag{0.4.21}$$

where SPT is the sparsity-promoting term introduced by SPICE by introducing a zero-mean Laplacian distribution for the parameters prior to promote sparsity

$$LSSP = -\frac{1}{2}\sum_{i=1}^{N}(x_i - \sum_{j=1}^{M} p_{ij}e_j)^2 - \sum_{j=1}^{M}\gamma_j \sum_{i=1}^{N}|p_{ij}| \tag{0.4.22}$$

so the SPT term should be of the form

$$SPT = \sum_{j=1}^{M}\gamma_j \sum_{i=1}^{N}|p_{ij}| = \sum_{j=1}^{M}\gamma_j \sum_{i=1}^{N} p_{ij} \tag{0.4.23}$$

with

$$\gamma_j = \frac{\Gamma}{\sum_{i=1}^{N} p_{ij}} \tag{0.4.24}$$

Γ is a constant associated with the degree that the proportion values are driven to zero. γ_j becomes larger as the sum of a particular endmember's proportion values becomes small, accelerating the minimization of those proportion values. The iterating process presented in ICE still works for minimizing 0.4.21. When solving for the proportion values given endmember estimates, each of the N terms

of the following sum need to be minimized given the constraints in 0.1.3 and 0.1.4 using quadratic programming

(0.4.25)
$$RSS_{reg,term1} = \frac{(1-\mu)}{N}\sum_{i=1}^{N}\left[(x_i - \sum_{j=1}^{M} p_{ij}e_j)^T(x_i - \sum_{j=1}^{M} p_{ij}e_j) + \sum_{j=1}^{M}\gamma_j^* p_{ij}\right]$$

where

(0.4.26)
$$\gamma_j^* = \frac{\Gamma^*}{\sum_{i=1}^{N} p_{ij}}, \qquad \Gamma^* = \frac{N\Gamma}{(1-\mu)}$$

After each iteration, endmembers with their maximum proportion values, $MAX(P_j) = \max_i \{p_{ij}\}$, under a given threshold can be pruned from the endmember set.

0.4.7. Vertex Component Analysis (VCA). Vertex Component Analysis introduced in [**32**] searches for the vertices of a simplex enclosing the data points by exploiting that the affine transformation of a simplex is also a simplex. The algorithm iteratively projects data onto a direction orthogonal to the subspace spanned by the endmembers already determined. The new endmember signature corresponds to the extreme of the projection. The algorithm iterates until all endmembers are exhausted.

VCA assumes the linear mixing model in 0.1.2 and introduces an scale factor, γ, modeling illumination variability due to surface topography

(0.4.27)
$$h = x + n = E\gamma\phi + n$$

Since the set $\left\{\phi \in \Re^p : 1^T\phi = 1, \phi \geq 0\right\}$ is a simplex, then the set $S_x = \left\{x \in \Re^D : x = E\phi, 1^T\phi = 1, \phi \geq 0\right\}$ is also a simplex. However even assuming zero noise, $n = 0$, the observed vector set belongs to $C_p = \left\{h \in \Re^D : h = E\gamma\phi, 1^T\phi = 1, \phi \geq 0, \gamma \geq 0\right\}$ that is a convex cone, owing to scale factor γ. The projective projection of the convex cone C_p onto a properly chosen hyperplane is a simplex with vertices corresponding to the vertices of the simplex S_x. The simplex $S_p = \left\{y \in \Re^D : y = h/(h^T u), r \in C_p\right\}$ is the projective projection of the convex cone C_p onto the plane $h^T u = 1$, where the chosen u assures that there is no observed vectors orthogonal to it.

In the absence of noise, observed vectors r lie in a convex cone C_p contained in a subspace of dimension p. VCA algorithm starts by applying a SVD dimensionality reduction technique to find the p-dimensionality subspace containing the data. Then points in C_p are projected onto a simplex S_p by computing $y = h/(h^T u)$. This simplex is contained into an affine set with $p-1$ dimensions. After identifying S_p, VCA iteratively projects data onto a direction orthogonal to the subspace spanned by the endmembers already determined. The new endmember signature corresponds to the extreme of the projection.

When SNR decreases the rescaling $y = h/(hu)$ process amplifies the noise, being preferable to identify directly the affine space of dimension $p-1$ by PCA. Based on experimental results, [**32**] proposes that if SNR is higher than a threshold $SNR_{th} = 15 + 10\log_{10}(p)$dB data should be projected onto the p-dimensionality subspace followed by the $y = h/(h^T u)$ rescaling, otherwise data should be projected directly onto $(p-1)$-dimensionality subspace.

The VCA algorithms first tests if the data's SNR is higher than SNR_{th}. If so, the projection matrix U_p is obtained by SVD and the projective projection obtained by

$$[Y]_{:,i} = \frac{[X]_{:,i}}{[X]_{:,i}^T u}, \qquad i = 1, \ldots, N \tag{0.4.28}$$

where $X = U_p^T H$ and u is an $1 \times p$ vector given by the mean of X. If SNR is lower than SNR_{th} the projection matrix U_{p-1} is obtained by PCA and the projective projection obtained by

$$Y = \begin{bmatrix} \mathbf{X} \\ \mathbf{c} \end{bmatrix} \tag{0.4.29}$$

where $[X]_{:,i} = U_{p-1}^T([H]_{:,i} - \bar{H})$ with $i = 1, \ldots, N$ and being $\bar{H}$ the sample mean of $[H]_{:,i}$. $\mathbf{c} = [c|c| \ldots |c]$ is an $1 \times N$ vector where $c = \arg\max_{i=1,\ldots,N} \|[X]_{:,i}\|$ that assures that the colatitude angle between u and any vector $[X]_{:,i}$ is between 0^o and 45^o avoiding numerical errors occurring for angles near 90^o.

Then VCA algorithm searches for the p endmembers iteratively. The founded endmembers are allocated in a $p \times p$ matrix A initialized as $A = [e_u|0|0| \ldots |0]$ where $e_u = [0, 0, \ldots, 0, 1]^T$. In each iteration a vector f orthonormal to the space spanned by the columns of the auxiliary matrix A is randomly generated by

$$f = \frac{(I - AA^{\#})w}{\|(I - AA^{\#})w\|} \tag{0.4.30}$$

where $A^{\#}$ is the pseudoinverse matrix of A, and $w = rand(0, I_p)$.

Then Y is projected onto f by $v = f^T Y$. Since pure endmembers is assumed to occupy the vertices of as simplex then $a \leq f^T [Y]_{:,i} \leq b$ for $i = 1, \ldots, N$, where values a and b correspond to pure pixels. The signature corresponding to $\max(|a|, |b|)$ given by $k = \arg\max_{i=1,\ldots,N} [v]_{:,i}$ is stored as a new endmember signature

$$[A]_{:,i} = [Y]_{:,i} \tag{0.4.31}$$

0.4.8. Simplex Growing Algorithm (SGA). Simplex Growing Algorithm (SGA) is presented in [**11**]. SGA find a set of endmembers by growing a sequence of simplexes increasing its vertices one at a time. The algorithm finishes when the number of vertices reaches the number of endmembers p.

The SGA algorithm begins by randomly generate a target pixel, t, and finding the first endmember e_1 by maximizing the absolute value of the determinant of the augmented matrix $\begin{bmatrix} 1 & 1 \\ t & r \end{bmatrix}$ over all sample vectors r where the sample data dimensionality has been reduced to the first principal component by PCA (or MNF)

$$e_1 = \arg\max_r \left\{ \left| \det \begin{bmatrix} 1 & 1 \\ t & r \end{bmatrix} \right| \right\} \tag{0.4.32}$$

The generated e_1 is always a pixel which has either a maximum or minimum value in the first component of dimensionality reduction transform.

Then SGA adds vertices iteratively one at a time. To do it, in each iteration i, for each sample data r the volume of the simplex specified by $e_1, \ldots, e_i, r$ is calculated by

$$V(e_1, \ldots, e_i, r) = \frac{\left|\det \begin{bmatrix} 1 & \ldots & 1 & 1 \\ e_1 & \ldots & e_i & r \end{bmatrix}\right|}{i!} \tag{0.4.33}$$

where the sample data dimensionality has been reduced to i dimensions. The new vertex is the one that yields the maximum of 0.4.33

$$e_{i+1} = \arg\max_r \{V(e_1, \ldots, e_i, r)\} \tag{0.4.34}$$

0.4.9. Minimum Volume Constrained Nonnegative Matrix Factorization (MVC-NMF). Nonnegative Matrix factorization (NMF) is a technique that finds a set of nonnegative basis vectors that approximates the original data through linear combinations. These basis vectors play a similar role as the endmembers, however any constraint is imposed to the model except nonnegativity which is not enough to deal with the endmember induction problem. [**30**] integrates the least squares analysis and the convex-geometry model by incorporating a volume constraint into the NMF formulation. The resulting method is denoted as Minimum Volume Constrained Nonnegative Matrix Factorization (MVC-NMF).

Given a nonnegative matrix $Y \in \Re^{m \times n}$ and a positive integer $r < \min(m, n)$, the task of NMF is to find two matrices $W \in \Re^{m \times r}$ and $H \in \Re^{r \times n}$ with nonnegative elements such that

$$Y \approx WH \tag{0.4.35}$$

or equivalently, the columns $\{y_i\}_{j=1}^n$ are expressed as

$$y_j \approx Wh_j \tag{0.4.36}$$

This can be solved by minimizing the Euclidean distance between Y and WH

$$\arg\min_{W,H} \left\{ \frac{1}{2} \sum_{i=1}^{m} \sum_{i=1}^{n} (Y_{ij} - (WH)_{ij})^2 \right\} \tag{0.4.37}$$

subject to $W \succeq 0$ and $H \succeq 0$. There are many solutions to this optimization problem which can be found in [**3**].

In a NMF problem all data points lie in a positive simplical cone C

$$C = \left\{ x | x = \sum_j \theta_j v_j \right\}, \quad \theta \succeq 0 \tag{0.4.38}$$

where $\{v_j\}_{j=1}^p$, $v_j \succeq 0$ are a set of basis vectors such that all the data points can be approximated as linear combinations of these bases and θ is a column vector with is components being the weight of the basis vectors.

The sum-to-one constraint 0.1.4 in the unmixing model confines the data points to reside within a simplex S

$$S = \left\{ x | x = \sum_j \theta_j v_j \right\}, \quad \theta \succeq 0, 1^T\theta = 1 \tag{0.4.39}$$

In the geometry-based endmembers induction techniques the best simplex is defined by the minimum volume simplex circumscribing the data cloud, or by the simplex that inscribes the data cloud with maximum volume. Under pure pixel assumption the vertices of the simplex correspond to the pure pixels. MVC-MNF method deals with highly-mixed data where the pure pixel assumption is not valid, extended the searching space outside the given data cloud.

Given the unmixing problem in 0.1.2 and combining the goal of minimum approximation error with the volume constraint, the optimization problem is defined by

$$\arg\min_{E,\Phi} \{f(E,\Phi)\} = \arg\min_{E,\Phi} \left\{ \frac{1}{2} \sum_{i=1}^{N} (X_i - E\Phi_i)^2 + \lambda J(E) \right\} \tag{0.4.40}$$

subject to $E \succeq 0$, $\Phi \succeq 0$ and $1_p^T\Phi = 1_N^T$. $J(E)$ is the penalty function, calculating the simplex volume determined by the endmember estimates. The regularization parameter $\lambda \in \Re$ is used to control the trade-off between the accurate reconstruction and the volume constraint. The simplex is the resulting of the balance between external force given by the first term of 0.4.40 that makes simplex increase, and the volume constraint that acts like an internal force that keeps the simplex small. The resulting simplex doesn't include every data point, particularly, the noisy pixels on the boundary, making the method robust to noisy data.

The penalty function $J(E)$ is determined by

$$J(E) = \frac{1}{2(p-1)!} \left\{ \det(\begin{bmatrix} 1_p^T \\ \tilde{E} \end{bmatrix}) \right\}^2 \tag{0.4.41}$$

where $\tilde{E} = (\tilde{e_1}, \ldots, \tilde{e_p})$ is a low dimensional transform of E given by

$$\tilde{E} = U^T(E - \mu 1_p^T) \tag{0.4.42}$$

where the matrix U is formed by the $p-1$ most significant principal components of data X through PCA, and column vector μ is the data mean. Then 0.4.41 can be reformulated as

$$J(E) = \frac{\tau}{2} \{\det(Z)\}^2 \tag{0.4.43}$$

where $\tau = \lambda/(p-1)!$ and the matrix $Z = C + BU^T(A - \mu 1_p^T)$ and the vector μ are constants. C and B are given by

$$C = \begin{bmatrix} 1_p^T \\ 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0_{p-1}^T \\ I \end{bmatrix} \tag{0.4.44}$$

with 0 being a $(p-1) \times p$ zero matrix, I is a $(p-1) \times (p-1)$ identity matrix and 0_{p-1} is a $(p-1)$-dimensional zero column vector.

The MVC-MNF algorithm to solve the optimization problem 0.4.40 given the penalty error 0.4.43 starts by selecting randomly p points from the given data and arrange them as columns of the initial E. Minimizing the objective function in 0.4.40 is treated as two optimization subproblems where the iteration rule is to update one matrix while the other one is kept fixed

$$E^{k+1} = \arg\min_{E} \left\{ f(E, \Phi^k) \leq f(E^k, \Phi^k) \right\} \tag{0.4.45}$$

$$\Phi^{k+1} = \arg\min_{\Phi} \left\{ f(E^{k+1}, \Phi) \leq f(E^{k+1}, \Phi^k) \right\} \tag{0.4.46}$$

For each subproblem the projected gradient learning is adopted to impose the nonnegative constraint and the update rule is expressed as

$$E^{k+1} = \max(0, E^k - \alpha^k \nabla_E f(E^k, \Phi^k)) \tag{0.4.47}$$

$$\Phi^{k+1} = \max(0, \Phi^k - \beta^k \nabla_\Phi f(E^{k+1}, \Phi^k)) \tag{0.4.48}$$

where α^k and β^k are an small learning rates. α^k is given by

$$\alpha^k = \rho^{m_k} \alpha_0 \tag{0.4.49}$$

where α_0 is the initial learning rate, $\rho \in (0, 1)$ is an scaling factor and m_k is the first integer such that

$$f(E^{k+1}, \Phi^k) - f(E^k, \Phi^k) \leq \sigma \rho^{m_k} \alpha_0 \nabla f(E^k, \Phi^k)^T (E^{k+1} - E^k) \tag{0.4.50}$$

The same procedure is used for β^k. The gradient learnings $\nabla_E f(E, \Phi)$ and $\nabla_\Phi f(E, \Phi)$ are given by

$$\nabla_E f(E, \Phi) = (E\Phi - X)\Phi^T + \tau \left\{ \det(Z) \right\}^2 U B^T (Z^{-1})^T \tag{0.4.51}$$

$$\nabla_\Phi f(E, \Phi) = E^T (E\Phi - X) \tag{0.4.52}$$

The algorithm stops when the euclidean norm of the gradient of the objective function is less than a threshold or after a fixed iterations number.

0.5. Lattice Computing-Based Induction Methods

Lattice computing can be defined as the collection of computational methods that either are defined on the algebra of lattice operators inf and sup, with the addition, or employ lattice theory to generalize previous approaches. Mathematical Morphology is a very successful case of this paradigm, but it also encompasses some fuzzy systems approaches and neural networks.

Lets review some basic concepts from lattice theory [**39, 44**]. The lattice theory is based on the algebraic lattice structure $(\Re_{\pm\infty}, \vee, \wedge, +, +')$ or $\Re_{\pm\infty}$-blog, where $\Re_{\pm\infty} = \Re \cup \{-\infty, \infty\}$ is the set of extended real numbers, $\vee$ and $\wedge$ denote, respectively, the discrete max and min operators (resp. sup and inf in a continuous setting) and $+$, $+'$ denote addition and its dual operation defined by

$$x +' y = y + x \qquad \forall x \in \Re,\ y \in \Re_{\pm\infty} \tag{0.5.1}$$

$$\infty +' (-\infty) = \infty = (-\infty) +' \infty \tag{0.5.2}$$

$$\infty + (-\infty) = -\infty = (-\infty) + \infty \tag{0.5.3}$$

If $x \in \Re_{\pm\infty}$, then its additive conjugate is given by $x* = -x$. Similarly, for a given vector $x \in \Re^n_{\pm\infty}$, its conjugate is defined by $x* = -x^T$.

Scalar addition is component wise, that is, if $a \in \Re_{\pm\infty}$ and $x \in \Re^n_{\pm\infty}$, then $a + x = (a + x_1, \ldots, a + x_n)^T$, the dual operation $a +' x$ is similar.

Given two $m \times n$ matrices $A = (a_{ij})$ and $B = (b_{ij})$ belonging to $\Re_{\pm\infty}$, the pointwise maximum of A and B, is $C = A \vee B$ where $c_{ij} = a_{ij} \vee b_{ij}$. Similarly, the pointwise minimum of A and B, is defined as $C = A \wedge B$ where $c_{ij} = a_{ij} \wedge b_{ij}$.

If A is an $m \times p$ matrix and B is a $p \times n$ matrix, then the max product of A and B, is the $m \times n$ matrix $C = A \boxtriangledown B$ whose i,j-th element, c_{ij}, is given by equation 0.5.4. Similarly, the min product of A and B, is the $m \times n$ matrix $C = A \boxtriangle B$ whose entries are computed following equation 0.5.5. Both products are collectively referred to as minimax products.

$$C = A \boxtriangledown B = [c_{ij}] \Leftrightarrow c_{ij} = \bigvee_{k=1,\ldots,n} \{a_{ik} + b_{kj}\} \tag{0.5.4}$$

$$C = A \boxtriangle B = [c_{ij}] \Leftrightarrow c_{ij} = \bigwedge_{k=1,\ldots,n} \{a_{ik} + b_{kj}\} \tag{0.5.5}$$

A vector $x \in \Re^n_{\pm\infty}$is called a max fixed point of A if $A \boxtriangledown x = x$ and a min fixed point of A if $A \boxtriangle x = x$.

A set of vectors $X = \left\{\mathbf{x}^1, \ldots, \mathbf{x}^k\right\} \subset \Re^n$ is said to be *max dominant* if and only if for every $\lambda \in \{1, \ldots, k\}$ there exists and index $j_\lambda \in \{1, \ldots, n\}$ such that

$$x^\lambda_{j_\lambda} - x^\lambda_i = \bigvee_{\xi=1}^{k} \left(x^\xi_{j_\lambda} - x^\xi_i\right) \qquad \forall i \in \{1, \ldots, n\} \tag{0.5.6}$$

Similarly, X is said to be *min dominant* if and only if for every $\lambda \in \{1, \ldots, k\}$ there exists and index $j_\lambda \in \{1, \ldots, n\}$ such that

$$x^\lambda_{j_\lambda} - x^\lambda_i = \bigwedge_{\xi=1}^{k} \left(x^\xi_{j_\lambda} - x^\xi_i\right) \qquad \forall i \in \{1, \ldots, n\} \tag{0.5.7}$$

An $n \times n$ square matrix A is said to be diagonally max dominant if and only if for each $i, j \in \{1, \ldots, n\}$ satisfies condition 0.5.8, and it is said to to be diagonally min dominant if and only if satisfies condition 0.5.9.

$$a_{jj} - a_{ij} = \bigvee_{k=1}^{n} (a_{jk} - a_{ik}) \tag{0.5.8}$$

$$a_{jj} - a_{ij} = \bigwedge_{k=1}^{n} (a_{jk} - a_{ik}) \tag{0.5.9}$$

The Automated Morphological Endmember Extraction (AMEE) method [**34**] is a mathematical morphology inspired algorithm for the induction of the endmembers from the data. It is based on the definition of multispectral erosion and dilation operators, which are then used to compute the Morphological Eccentricity Index (MEI) over kernels of increasing size that are computed over all the pixels in the image. The result is a MEI image whose maxima corresponds to the endmember pixels. The concept of morphological independence, later reformulated as lattice independence, was the basic tool in the approach proposed in [**15, 29**]. The set of endmembers was formulated as a set of morphologically independent vectors, either in a dilative or erosive sense, or both. There the Associative Morphological Memories, later renamed Lattice Associative Memories, are proposed as detectors of morphologically independent vectors. The algorithm works in a single pass over the sample data. This approach has been followed by the one proposed in [**46**]. The relationship between strong lattice independence and affine independence was proven. Then it was found that most vectors in the erosive and dilative lattice memories are strong lattice independent. Therefore, the mere construction of the lattice memories provides a way to obtain the convex hull of the data. Provided an endmember selection mechanism, the algorithm can obtain a set of endmembers in a single pass over the image.

0.5.1. Automated Morphological Endmember Extraction (AMEE). [**34**] presents a morphology-based algorithm for endmember induction, denoted as Automated Morphological Endmember Extraction (AMEE), using both spectral and spatial information. The algorithm is based on the dilation and erosion morphology operators extended from grayscale morphology to hyperspectral morphology:

$$d(x,y) = (f \otimes K)(x,y) = \arg \max_{(s,t)\in K} \{Dist(f(x+s, y+t), K)\} \tag{0.5.10}$$

$$e(x,y) = (f \otimes K)(x,y) = \arg \min_{(s,t)\in K} \{Dist(f(x+s, y+t), K)\} \tag{0.5.11}$$

where $f(x,y)$ denotes a data vector of D dimensions at spatial coordinates (x,y), K is a set of neighboring pixels defined by a plain kernel, and $Dist(f(x,y),K)$ is a metric that calculates the cumulative distance between one particular pixel $f(x,y)$ of the kernel and every other pixel $f(s,t)|(s,t) \in K$

$$Dist(f(x,y),K) = \sum_s \sum_t dist(f(x,y), f(s,t)), \qquad \forall (s,t) \in K \tag{0.5.12}$$

where $dist(f(x,y), f(s,t))$ is a pointwise linear distance measure between two D-dimensional vectors, i.e., the spectral angle distance.

This cumulative distance can order the vectors of a kernel in terms of their spectral purity, and so, the dilation and erosion operators can be seen as the most spectrally singular pixel and the most highly mixed element respectively.

As the computational cost of calculating the cumulative distance is cubic, [**34**] presents an alternative based on the distance between every spectral point and the centroid of the kernel data cloud, defined as

$$c_K = \frac{1}{M} \sum_s \sum_t f(s,t), \qquad \forall (s,t) \in K \tag{0.5.13}$$

where M is the number of elements in the kernel K. The resulting alternative distance function yields

$$Dist'(f(x,y),K) = dist(f(x,y),c_K) \tag{0.5.14}$$

where $dist(f(x,y),c_K)$ is again a pointwise linear distance measure. This distance function has a quadratic computational complexity.

The AMEE algorithm consists in elaborating an spectral purity index at kernel level, so the algorithm iterates over a set of L kernels with different sizes that are convolved with all data points. The spectrally purest pixel and the spectrally most highly mixed pixel are obtained at each kernel neighborhood by dilation 0.5.10 and erosion 0.5.11 operations. Then a Morphological Eccentricity Index (MEI) is associated with the purest pixel by comparing the result of the dilation to the result of erosion. The associated MEI value of selected pixels at subsequent iterations is updated by means of newly obtained values, and a final MEI image is generated after L iterations. The automated endmember selection is performed from MEI image by a threshold value T.

The MEI value is obtained by calculating the spatial coordinates of the maximum pixel, based on the dilation operation $d(x,y)$ at the kernel neighborhood of a selected pixel with coordinates (x,y), and the minimum pixel of the same kernel neighborhood, based on the erosion operation $e(x,y)$. Then the MEI index is given by a pointwise distance function, i.e., the spectral angle distance, between them

$$MEI(n,m) = dist(d(x,y),e(x,y)) \tag{0.5.15}$$

where (n,m) are the spatial coordinates of the maximum pixel.

0.5.2. Associative Morphological Memories (AMM). The work on Lattice Associative Memories (LAM) stems from the consideration of the algebraic lattice structure $(\Re, \vee, \wedge, +)$ as the alternative to the algebraic framework given by the mathematical field $(\Re, +, \cdot)$ for the definition of Neural Networks computation. The LAM were first introduced in [**42, 40**] as Morphological Associative Memories and [**41, 44**] set the works in the more general framework of Lattice Computing. Given a set of input/output pairs of pattern $(X,Y) = \left\{\left(\mathbf{x}^\xi, \mathbf{y}^\xi\right); \xi = 1,..,k\right\}$, a linear heteroassociative neural network based on the pattern's cross correlation is built up as $W = \sum_\xi \mathbf{y}^\xi \cdot \left(\mathbf{x}^\xi\right)'$. Mimicking this constructive procedure [**42, 40**] propose the following constructions of Lattice Memories (LM):

$$W_{XY} = \bigwedge_{\xi=1}^{k} \left[\mathbf{y}^\xi \times \left(-\mathbf{x}^\xi\right)'\right] \qquad \text{and} \qquad M_{XY} = \bigvee_{\xi=1}^{k} \left[\mathbf{y}^\xi \times \left(-\mathbf{x}^\xi\right)'\right] \tag{0.5.16}$$

where $\times$ is any of the $\boxtriangledown$ or $\boxtriangle$ operators respectively defined by 0.5.4 and 0.5.5.

If $X = Y$ then the LM memories are Lattice Autoassociative Memories (LAM). Conditions of perfect recall by the LM and LAM of the stored patterns proved in [**42, 40**] encouraged the research on them, because in the continuous case, the LAM is able to store and recall any set of patterns: $W_{XX} \boxtriangledown X = X = M_{XX} \boxtriangle X$, for any X. However, this result holds when we deal with noise-free patterns. Research on robust recall [**40, 43, 37**] based on the so-called kernel patterns lead to the notion of morphological independence, in the erosive and dilative sense, and finally to the definition of Lattice Independence (LI) and Strong Lattice Independence (SLI).

Given a set of vectors $\left\{\mathbf{x}^1, \ldots, \mathbf{x}^k\right\} \subset \Re^n$ a *linear minimax combination* of vectors from this set is any vector $\mathbf{x} \in \Re^n_{\pm\infty}$ which is a *linear minimax sum* of these vectors

$$x = \mathcal{L}\left(\mathbf{x}^1, \ldots, \mathbf{x}^k\right) = \bigvee_{j \in J} \bigwedge_{\xi=1}^{k} \left(a_{\xi j} + \mathbf{x}^\xi\right)$$

where J is a finite set of indices and $a_{\xi j} \in \Re^n_{\pm\infty}$, $\forall j \in J$, $\forall \xi = 1, \ldots, k$. The *linear minimax span* of vectors $\left\{\mathbf{x}^1, \ldots, \mathbf{x}^k\right\} = X \subset \Re^n$ is the set of all linear minimax sums of subsets of X, denoted $LMS\left(\mathbf{x}^1, \ldots, \mathbf{x}^k\right)$.

Given a set of vectors $X = \left\{\mathbf{x}^1, \ldots, \mathbf{x}^k\right\} \subset \Re^n$, a vector $\mathbf{x} \in \Re^n_{\pm\infty}$ is *lattice dependent* if and only if $x \in LMS\left(\mathbf{x}^1, \ldots, \mathbf{x}^k\right)$. The vector $\mathbf{x}$ is *lattice independent* if and only if it's not lattice dependent on X. The set X is said to be *lattice independent* if and only if $\forall \lambda \in \{1, \ldots, k\}$, $\mathbf{x}^\lambda$ is lattice independent of $X \backslash \left\{\mathbf{x}^\lambda\right\} = \left\{\mathbf{x}^\xi \in X : \xi \neq \lambda\right\}$.

The definition of lattice independence supersedes and improves the early definitions [**43**] of erosive and dilative morphological independence, which, however, have more intuitive appealing. Nevertheless, this definition has the additional advantage of establishing a formal parallelism with the definition of linear independence.

A set of lattice independent vectors $\left\{\mathbf{x}^1, \ldots, \mathbf{x}^k\right\} \subset \Re^n$ is said to be *strongly lattice independent* (SLI) if and only if X is max dominant 0.5.6 or min dominant 0.5.7 or both. Min and max dominance are the conditions for perfect recall. Per construction, the column vectors of Lattice Autoassociative Memories are min or max dominant, depending of their erosive or dilative nature, therefore they will be strongly lattice independent, *if* they are lattice independent.

[**44**] conjectures that if $X = \left\{\mathbf{x}^1, \ldots, \mathbf{x}^k\right\} \subset \Re^n$ is strongly lattice independent then X is affinely independent. This conjecture (stated as theorem in [**41**]) is the key result whose proof would relate the linear convex analysis and the non-linear lattice analysis. If true, it means that the construction of the LAM provides the starting point for obtaining sets of affine independent vectors that could be used as endmembers for the unmixing algorithms described below.

Let $X = \left\{\mathbf{x}^1, \ldots, \mathbf{x}^k\right\} \subset \Re^n$ and let W (M) be the set of vectors consisting of the columns of the matrix W_{XX} (M_{XX}). Let $F(X)$ denote the set of fixed points of the LAM constructed from set X. There exist $V \subset W$ and $N \subset M$ such that V and N are strongly lattice independent and $F(X) = F(V) = F(N)$ or, equivalently, $W_{XX} = W_{VV}$ and $M_{XX} = M_{NN}$. The key idea of this theorem is to test the lattice independence of the already known as min or max dominant sets of vectors. Removing lattice dependent vectors will not affect this min/max dominance property. The smart way to test lattice dependence lies in the fact that removing a lattice dependent vectors does not alter the set of fixed points of the remaining ones. This theorem is proved following a constructive reasoning, giving

Algorithm 1 Endmember Threshold Selection Algorithm (ETSA)

(1) Given a set of vectors $X = \left\{\mathbf{x}^1, \ldots, \mathbf{x}^k\right\} \subset \Re^n$compute the min and max autoassociative memories, W_{XX} and M_{XX}, from the data. Their columns vectors denoted as W and M will be the candidate endmembers.
(2) Register W and M relative to the data set by adding the maximum and minimum values of each data bands. Obtain $\overline{W}$ and $\overline{M}$ as follows:
  (a) Compute $v^i = \bigwedge_{\xi=1}^n x_i^\xi$ and $u^i = \bigvee_{\xi=1}^n x_i^\xi$.
  (b) Compute $\overline{m}^i = m^i + v^i$ and $\overline{w}^i = w^i + u^i$.
(3) Remove lattice dependent vectors from the joint set $\overline{W} \cup \overline{M}$.
(4) Compute the standard deviation of the candidate endmember vectors, denoted as $\sigma = \{\sigma_1, \ldots, \sigma_n\}$.
(5) Assume the first vector v_1 in the set $\overline{W} \cup \overline{M}$ as the initial endmember, $E = \{v_1\}$.
(6) Iterate for the remaining vectors $v \in \overline{W} \cup \overline{M}$:
  (a) If $\|v - e\| < \gamma\sigma$ for any $e \in E$ discard v, otherwise include v in E.

Algorithm 2 Endmember Induction Heuristic Algorithm (EIHA)

(1) Shift the data sample to zero mean
  $\{\mathbf{f}^c(i) = \mathbf{f}(i) - \overrightarrow{\mu}; i = 1, .., n\}$.
(2) Initialize the set of vertices $E = \{\mathbf{e}_1\}$ with a randomly picked sample. Initialize the set of lattice independent binary signatures $X = \{\mathbf{x}_1\} = \left\{\left(e_k^1 > 0; k = 1, .., d\right)\right\}$
(3) Construct the LAM's based on the lattice independent binary signatures: M_{XX} and W_{XX}.
(4) For each pixel $\mathbf{f}^c(i)$
  (a) compute the noise corrections sign vectors$\mathbf{f}^+(i) = (\mathbf{f}^c(i) + \alpha\overrightarrow{\sigma} > \mathbf{0})$ and $\mathbf{f}^-(i) = (\mathbf{f}^c(i) - \alpha\overrightarrow{\sigma} > \mathbf{0})$
  (b) compute $y^+ = M_{XX} \boxtimes \mathbf{f}^+(i)$
  (c) compute $y^- = W_{XX} \boxtimes \mathbf{f}^-(i)$
  (d) if $y^+ \notin X$ or $y^- \notin X$ then $\mathbf{f}^c(i)$ is a new vertex to be added to E, execute once 3 with the new E and resume the exploration of the data sample.
  (e) if $y^+ \in X$ and $\mathbf{f}^c(i) > \mathbf{e}_{y^+}$ the pixel spectral signature is more extreme than the stored vertex, then substitute $\mathbf{e}_{y^+}$ with $\mathbf{f}^c(i)$.
  (f) if $y^- \in X$ and $\mathbf{f}^c(i) < \mathbf{e}_{y^-}$ the new data point is more extreme than the stored vertex, then substitute $\mathbf{e}_{y^-}$ with $\mathbf{f}^c(i)$.
(5) The final set of endmembers is the set of original data vectors $\mathbf{f}(i)$ corresponding to the sign vectors selected as members of E.

way to an algorithm for the construction of the set of affine independent sets of vectors from LAM discussed in [**44, 16**].

Two endmember induction methods based on the lattice independence theorems are presented, the Endmember Threshold Selection Algorithm (ETSA) [**44**] given by 1 and the Endmember Induction Heuristic Algorithm (EIHA) [**16, 15**] described in 2.

0.6. Heuristic-Based Induction Methods

The heuristic methods collects a set of heterogeneous endmember extraction methods that use different approaches not grouped under an strict theoretical background for endmember induction. The most famous and widely used method, due to its inclusion in the ENVI software package, it is the Pixel Purity Index (PPI) [**5**]. The algorithm reduces the data dimensionality and makes a noise whitened process by MNF method, and then it determines the pixel purity by repeatedly projecting data onto random unit vectors. The extreme pixel in each projection is counted, identifying the purest pixels in scene. PPI requires human intervention to select those extreme pixels that best satisfy the target spectrum. Although PPI has been intensively used, its implementation aspects are kept unknown due to the limited published results. In [**10**] PPI is investigated and a fast iterative algorithm to implement PPI is proposed. The Fast Iterative PPI algorithm (FIPPI) improves PPI in several aspects. FIPPI produces an appropiate initial set of endmembers to speed up the process. Additionally, it estimates the number of endmembers to be generated by Virtual Dimensionality (VD). FIPPI is also an unsupervised and iterative algorithm, where an iterative rule is developed to improve each of the iterations until it reaches a final set of endmembers.

The Spatial-Spectral Endmember Extraction algorithm (SSEE) proposed in [**45**] is another projection based method that works by analyzing a scene in parts (subsets), such that it increases the spectral contrast of low contrast endmembers, thus improving the potential for these endmembers to be selected. The SSEE method uses a singular value decomposition (SVD) to determine a set of basis vectors that describe most of the spectral variance for subsets of the image. Then the full image dataset is projected onto the locally defined basis vectors to determine a set of candidate endmember pixels from where the final endmembers are selected. For that, it searches for spectrally similar but spatially independent endmembers. This is realized by imposing spatial constraints for averaging spectrally similar endmembers.

0.6.1. Fast Iterative Pixel Purity Index (FIPPI). The Fast Iterative Pixel Purity Index (FIPPI) [**10**] method is an improve of the Pixel Purity Index [**5**] algorithm for endmembers induction. PPI is an heuristic algorithm based on the projection of the dimensionality reduced dataset onto a set of k random unit vectors denoted as skewers, $\{skewer_j\}_{j=1}^{k}$, where k is a sufficiently large positive integer. All the data sample vectors are projected onto each $skewer_j$, selecting the extreme vectors that form an extrema set, denoted by $S(skewer_j)$. The PPI score for each point vector r is calculated by

$$N_{PPI}(r) = \sum_j I_{S(skewer_j)}(r) \tag{0.6.1}$$

where $I_{S(skewer_j)}$ is an indicator function of an extrema set defined as

$$I_{S(skewer_j)} = \begin{cases} 1, & if\ r \in S(skewer_j) \\ 0, & if\ r \notin S(skewer_j) \end{cases} \tag{0.6.2}$$

After finding the PPI scores, $N_{PPI}(r)$, for all the sample vectors, the vectors with a PPI scores such that $N_{PPI}(r) \geq t$ are selected as endmember candidates.

The PPI algorithm involves defining two parameters k and t, being k the number of random generated skewers and t the threshold value to find the endmember candidates.

PPI has several drawbacks, it's not an iterative process and does not guarantee that the generated endmembers are actually true endmembers due to the randomness of generated skewers, a different set of skewers generate a different set of endmembers; it's very sensible to noise, there is no criteria to select the correct values for the k and t parameters, which determine the number of final endmembers, and it requires human intervention to manually select a final set of endmembers.

FIPPI algorithm addresses these major drawbacks. FIPPI uses the HFC method to find the virtual dimensionality of the data being p the number of endmembers required to generate. Then apply a MNF transform to reduce the data to the first resulting p components. FIPPI also uses the Automated Target Generation Process (ATGP) [**33**] which is an Endmember Initialization Algorithm (EIA) to generate the initial set of p skewers, $\left\{skewers_j^{(0)}\right\}_{j=1}^{p}$. After algorithm initialization, FIPPI iterates projecting in each iteration k all the sample vectors onto each skewer, $skewer_j^{(k)}$ to find those which are at its extreme positions to form an extrema set, denoted by $S(skewer_j^{(k)})$, and then to find the sample vectors that produce the largest $N_{PPI}(r_j^{(k)})$ defined by 0.6.1 and let them be denoted as $\left\{r_j^{(k)}\right\}$. In each iteration the joint set $\left\{skewer_j^{(k+1)}\right\} = \left\{r_j^{(k)}\right\}_{N_{PPI}(r_j^{(k)})>0} \cup \left\{skewer_j^{(k)}\right\}$ is formed. If $\left\{skewer_j^{(k+1)}\right\} = \left\{skewer_j^{(k)}\right\}$ no new endmembers are added to the skewer set and the algorithm is terminated being the vectors with $N_{PPI}(r_j^{(k+1)}) > 0$ the desired endmembers.

0.6.2. Spatial-Spectral Endmember Extraction algorithm (SSEE). Spectral-based image endmember induction methods hinge on the ability to discriminate between pixels based on spectral characteristics. Those endmembers with distinct spectral features (high spectra contrast) are easy to select, whereas those with minimal unique spectral information (low spectral contrast) are more problematic. Spatial-Spectral Endmember Extraction (SSEE) algorithm works in the basis that it's possible for an endmember to have low spectral contrast with respect to the full image, but have high spectral contrast within a subset of the image. So, SSEE analyzes an hyperspectral image in subscenes such that the spectral contrast of low contrast endmembers is increased, improving the potential for these endmembers to be selected.

The algorithm has four steps, firstly the Singular Value Decomposition (SVD) method is applied to determine a set of eigenvectors that describe most of the spectral variance of image subsets, secondly the entire image data is projected onto the eigenvectors to determine a set of candidate endmember pixels, thirdly spatial constraints are used to combine and average spectrally similar candidate endmember pixels, and finally candidate endmembers are listed in spectral similarity order.

SSEE's first step is to apply SVD to subsets of the image that are square, equally sized and that do not overlap. Eigenvectors that account for 99% of total spectral variance are retained from each subset been compiled into a single set of vectors. For each subset the minimum number of vectors is set to 2, while the

maximum is defined by the variance threshold value. The minimum size of the subset is defined by the square root of the number of bands in image data.

In step two, the entire image data is projected onto the compiled vector set, retaining the pixels that lie at either extreme of the vectors. These pixels represent the candidate pixel endmember set, S_E.

Step three analyzes the spatial and spectral characteristics of the candidate endmember set to average spectrally similar endmember candidates that are spatially related. This is done by scanning the image with a sliding window of size equal to the subset size used in step one. For each candidate endmember $e_j \in S_E$, $j = 1, \ldots, K$ pixels within the window that are similar to the candidate endmembers based on a distance metric, i.e. the spectral angle or root mean square, form the candidate endmember pixels set, $\left\{P_{e_j}\right\}_{j=1}^{K}$. Then each of the candidate endmember pixels is averaged. The averaging process is repeated for n number of iterations reducing the effect of noise, finding image pixels that are spectrally similar but spatially related within the window and compressing candidate endmembers into clusters with reduced variance.

Finally, in the fourth step the endmember set derived from step three is reordered based on spectral angle, grouping similar spectra for user inspection.

0.7. Conclusions

Bibliography

[1] C. A. Bateson, G. P. Asner, and C. A. Wessman. Endmember bundles: a new approach to incorporating endmember variability into spectral mixture analysis. *Geoscience and Remote Sensing, IEEE Transactions on*, 38(2):1083–1094, 2000.

[2] M. Berman, H. Kiiveri, R. Lagerstrom, A. Ernst, R. Dunne, and J. F. Huntington. Ice: a statistical approach to identifying endmembers in hyperspectral images. *Geoscience and Remote Sensing, IEEE Transactions on*, 42(10):2085–2095, 2004.

[3] M. Berry, M. Browne, A. Langville, P. Pauca, and R. Plemmons. Algorithms and applications for approximate nonnegative matrix factorization. *Computational Statistics & Data Analysis*, 52(1):173, 155, 2007.

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

[5] J. Boardman, F. Kruse, and R. Green. Mapping target signatures via partial unmixing of aviris data. 1995.

[6] C.-I. Chang. *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*. Springer, 1 edition, July 2003.

[7] C.-I. Chang. *Hyperspectral Data Exploitation: Theory and Applications*. Wiley-Interscience, April 2007.

[8] C.-I. Chang and Q. Du. Interference and noise-adjusted principal components analysis. *Geoscience and Remote Sensing, IEEE Transactions on*, 37(5):2387–2396, 1999.

[9] 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.

[10] C.-I. Chang and A. Plaza. A fast iterative algorithm for implementation of pixel purity index. *Geoscience and Remote Sensing Letters, IEEE*, 3(1):63–67, 2006.

[11] C.-I. Chang, C.-C. Wu, W. Liu, and Y.-C. Ouyang. A new growing method for simplex-based endmember extraction algorithm. *Geoscience and Remote Sensing, IEEE Transactions on*, 44(10):2804–2819, 2006.

[12] M. D. Craig. Minimum-volume transforms for remotely sensed data. *Geoscience and Remote Sensing, IEEE Transactions on*, 32(3):542–552, 1994.

[13] K. Fukunaga. *Introduction to Statistical Pattern Recognition, Second Edition*. Academic Press, 2 edition, October 1990.

[14] D. Gillis, J. Bowles, and M. E. Winter. Using endmembers as a coordinate system in hyperspectral imagery. In *6th Annual International Symposium on Aerospace/Defense Sensing, Simulation and Controls*, Orlando FL, 2002.

[15] M. Grana and J. Gallego. Associative morphological memories for endmember induction. In *Geoscience and Remote Sensing Symposium, 2003. IGARSS '03. Proceedings. 2003 IEEE International*, volume 6, pages 3757–3759 vol.6, 2003.

[16] M. Grana, I. Villaverde, J. O. Maldonado, and C. Hernandez. Two lattice computing approaches for the unsupervised segmentation of hyperspectral images. In *Special issue: JCIS 2008*, Kylin Villa, Shenzhen, China, 2008.

[17] A. A. Green, M. Berman, P. Switzer, and M. D. 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.

[18] J. C. Harsanyi and C.-I. Chang. Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach. *Geoscience and Remote Sensing, IEEE Transactions on*, 32(4):779–785, 1994.

[19] G. Healey and D. Slater. Models and methods for automated material identification in hyperspectral imagery acquired under unknown illumination and atmospheric conditions. *Geoscience and Remote Sensing, IEEE Transactions on*, 37(6):2706–2717, 1999.

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

[21] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441, 1933.

[22] A. Hyvarinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430, June 2000.

[23] A. Ifarraguerri and C.-I. Chang. Multispectral and hyperspectral image analysis with convex cones. *Geoscience and Remote Sensing, IEEE Transactions on*, 37(2):756–770, 1999.

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

[25] D. Landgrebe. Hyperspectral image data analysis. *Signal Processing Magazine, IEEE*, 19(1):17–28, 2002.

[26] D. Landgrebe. *Signal Theory Methods in Multispectral Remote Sensing*. Wiley-Interscience, 2003.

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

[28] J. B. Lee, A. S. Woodyatt, and M. Berman. Enhancement of high spectral resolution remote-sensing data by a noise-adjusted principal components transform. *Geoscience and Remote Sensing, IEEE Transactions on*, 28(3):295–304, 1990.

[29] J.O. Maldonado, D. Vicente, M.A. Veganzones, and M. Grana. Spectral indexing for hyperspectral image cbir. Torrejon air base, Madrid (Spain), 2006.

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

[31] J. M. P. Nascimento and J. M. B. Dias. Does independent component analysis play a role in unmixing hyperspectral data? *Geoscience and Remote Sensing, IEEE Transactions on*, 43(1):175–187, 2005.

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

[33] A. Plaza and C.-I. Chang. Impact of initialization on design of endmember extraction algorithms. *Geoscience and Remote Sensing, IEEE Transactions on*, 44(11):3397–3407, 2006.

[34] A. Plaza, P. Martinez, R. Perez, and J. Plaza. Spatial/spectral endmember extraction by multidimensional morphological operations. *Geoscience and Remote Sensing, IEEE Transactions on*, 40(9):2025–2041, 2002.

[35] A. Plaza, P. Martinez, R. Perez, and J. Plaza. A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data. *Geoscience and Remote Sensing, IEEE Transactions on*, 42(3):650–663, 2004.

[36] A. Plaza, D. Valencia, J. Plaza, and Chein-I Chang. Parallel implementation of endmember extraction algorithms from hyperspectral data. *Geoscience and Remote Sensing Letters, IEEE*, 3(3):334–338, 2006.

[37] B. Raducanu, M. Grana, and F. X. Albizuri. Morphological scale spaces and associative morphological memories: Results on robustness and practical applications. *Journal of Mathematical Imaging and Vision*, 19(2):113–131, 2003.

[38] 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, 2008.

[39] G. X. Ritter. Image algebra. 1999.

[40] G. X. Ritter, J. L. Diaz-de Leon, and P. Sussner. Morphological bidirectional associative memories. *Neural Networks*, 12(6):851–867, July 1999.

[41] G. X. Ritter and P. Gader. *Fixed points of lattice transforms and lattice associative memories*, page 264. Advances in imaging and electron physics. Academic press, 2006.

[42] G. X. Ritter, P. Sussner, and J. L. Diaz-de Leon. Morphological associative memories. *Neural Networks, IEEE Transactions on*, 9(2):281–293, 1998.

[43] G. X. Ritter, G. Urcid, and L. Iancu. Reconstruction of patterns from noisy inputs using morphological associative memories. *Journal of Mathematical Imaging and Vision*, 19(2):95–111, 2003.

[44] G. X. Ritter, G. Urcid, and M. S. Schmalz. Autonomous single-pass endmember approximation using lattice-associative memories. In *Special issue: JCIS 2007*, Salt Lake City, Utah, USA, 2007.

[45] D. M. Rogge, B. Rivard, J. Zhang, A. Sanchez, J. Harris, and J. Feng. Integration of spatial-spectral information for the improved extraction of endmembers. *Remote Sensing of Environment*, 110(3):287–303, October 2007.

[46] M.S. Schmalz. Autonomous single-pass endmember approximation using lattice auto-associative memories. 2007.

[47] J. Setoain, M. Prieto, C. Tenllado, A. Plaza, and F. Tirado. Parallel morphological endmember extraction using commodity graphics hardware. *Geoscience and Remote Sensing Letters, IEEE*, 4(3):441–445, 2007.

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

[49] P. Switzer and A. A. Green. Min/max autocorrelation factors for multivariate spatial imagery. Technical Report 6, Stanford University, Department of Statistics, 1984.

[50] J. A. Vargas-Guzman and R. Dimitrakopoulos. Computational properties of min/max autocorrelation factors. *Computers & Geosciences*, 29(6):715–723, July 2003.

[51] J. Wang and C.-I. Chang. Applications of independent component analysis in endmember extraction and abundance quantification for hyperspectral imagery. *Geoscience and Remote Sensing, IEEE Transactions on*, 44(9):2601–2616, 2006.

[52] Jing Wang and Chein-I Chang. Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis. *Geoscience and Remote Sensing, IEEE Transactions on*, 44(6):1586–1600, 2006.

[53] M. E. Winter, M. R. Descour, and S. S. Shen. N-findr: an algorithm for fast autonomous spectral end-member determination in hyperspectral data. volume 3753, pages 266–275, Denver, CO, USA, October 1999. SPIE.

[54] A. Zare and P. Gader. Sparsity promoting iterated constrained endmember detection in hyperspectral imagery. *Geoscience and Remote Sensing Letters, IEEE*, 4(3):446–450, 2007.