

# Chapter 2

## Models for Hyperspectral Image Analysis: From Unmixing to Object-Based Classification

Emmanuel Maggiori, Antonio Plaza and Yuliya Tarabalka

**Abstract** The recent advances in hyperspectral remote sensing technology allow the simultaneous acquisition of hundreds of spectral wavelengths for each image pixel. This rich spectral information of the hyperspectral data makes it possible to discriminate different physical substances, leading to a potentially more accurate classification and thus opening the door to numerous new applications. Throughout the history of remote sensing research, numerous methods for hyperspectral image analysis have been presented. Depending on the spatial resolution of the images, specific mathematical models must be designed to effectively analyze the imagery. Some of these models operate at a sub-pixel level, trying to decompose a mixed spectral signature into its pure constituents, while others operate at a pixel or even object level, seeking to assign unique labels to every pixel or object in the scene. The spectral mixing of the measurements and the high dimensionality of the data are some of the challenging features of hyperspectral imagery. This chapter presents an overview of unmixing and classification methods, intended to address these challenges for accurate hyperspectral data analysis.

### 2.1 Introduction

Hyperspectral remote sensors allow the simultaneous acquisition of hundreds of spectral bands with narrow bandwidths for each image pixel. For example, the AVIRIS sensor (airborne visible/infrared imaging spectrometer) provides images with 224 contiguous bands with a bandwidth of 10 nm each, and the ROSIS sensor (reflec-

---

E. Maggiori (✉) · Y. Tarabalka  
Université Côte d'Azur, Inria, BP 93, 2004, route des Lucioles, 06902  
Sophia Antipolis Cedex, France  
e-mail: emmanuel.maggiori@inria.fr

Y. Tarabalka  
e-mail: yuliya.tarabalka@inria.fr

A. Plaza  
University of Extremadura, Badajoz, Spain  
e-mail: aplaza@unex.es

© Springer International Publishing AG 2018  
G. Moser and J. Zerubia (eds.), *Mathematical Models for Remote Sensing  
Image Processing*, Signals and Communication Technology,  
[https://doi.org/10.1007/978-3-319-66330-2\\_2](https://doi.org/10.1007/978-3-319-66330-2_2)

tive optics system imaging spectrometer) provides 115 bands with a bandwidth of 4 nm each. In the spectral domain, pixels are represented as vectors for which each component is a measurement corresponding to specific wavelengths. The size of the vector is equal to the number of spectral bands that the sensor collects. For hyperspectral images, over a hundred of bands are typically available, while for conventional multispectral images up to ten bands are usually provided (see Chap. 3). This detailed spectral information increases the possibility of more accurately discriminating materials of interest [35]. The capabilities of hyperspectral sensors go beyond the identification of land cover, facilitating also the characterization of minerals [37], soils [17] and biodiversity [43]. Due to the increasing amount of data, the automatic analysis of hyperspectral images is then of paramount importance in remote sensing. One of the ultimate goals of remote sensing image analysis is to construct a thematic map associated to the image. Such a map indicates the elements present in the image, at every location, out of a set of possible classes of interest. These could go from physical substances to higher-level semantic objects, depending on the application.

While the spectral signatures collected at every pixel of a hyperspectral image are very detailed, they are usually a mixture of the signatures of the various materials found in their spatial vicinity [14]. Thus, if the spectrum is not *pure*, comparing a pixel's spectral signature with a set of reference signatures to identify the material is not an effective approach. In the earliest sensors, spatial resolution was low and the size of objects was comparable to the size of pixels. In such a low spatial resolution setting, spectral mixing compromises the key feature of the sensors: their ability to discriminate materials based on their spectral responses [13]. This drove much attention of the research community to the so-called *unmixing* of the spectral signatures, i.e., decomposing a mixed spectral signature into pure components. This can be seen as a sub-pixel analysis of the data. In hyperspectral images, the number of bands typically exceeds the amount of components in the mix, allowing to express the unmixing problem as an over-determined system of equations [47]. This chapter reviews one of the most common unmixing models, linear spectral unmixing, which assumes that the spectrum in a hyperspectral image is a linear combination of pure spectra.

As technology evolved, the spatial resolution of hyperspectral imagery increased, and objects of interest started to be composed of multiple pixels. Spectral mixing being less of a hassle, the assignment of a *unique* label to every pixel became an active research area, a process known as *classification* [13]. Since the assumption of classification is that pixels are pure, an unmixing technique is preferable if that property does not stand. The high-dimensional nature of hyperspectral imagery imposes certain challenges to perform classification, and conventional algorithms for multispectral images do not adapt well [68]. When there is a limited number of reference samples to train a classification system, as the number of dimensions increases (i.e., the number of spectral bands) the accuracy of the classification tends to drop. This is because the reliable estimation of statistical class parameters becomes more and more difficult as dimensionality increases. This phenomenon, the Hughes effect [59], is often referred to as the *curse of dimensionality*. A vast number of classification

techniques for hyperspectral imagery have been presented in the literature, which share the goal of attenuating the Hughes effect and accurately identifying the pixels' classes.

The first classification techniques were *pixelwise*, i.e., considering every pixel as an isolated entity and classifying it based solely on its spectrum. The next generation of techniques introduced the notion of a spatial arrangement of the pixels, with some interaction between spatially neighboring pixels at the time of classifying. This family of methods are known as *spectral-spatial* and tend to outperform purely pixelwise approaches [41]. The overall principle is to introduce a certain spatial regularity in the pixel label assignment, by incorporating information of the spatial neighbors. A third category is the so-called *object-based* analysis, which naturally emerged from the increase in the amount of pixels per object [15]. Object-based methods are spectral-spatial methods that seek to delineate readily usable objects to incorporate into other systems (such as geographic information systems). These techniques both segment the image into significant regions and label each of the segments. This chapter reviews pixelwise and spectral-spatial techniques, and described in detail a recent object-based model based on binary partition trees.

## 2.2 Unmixing

Spectral unmixing has been an alluring exploitation goal since the earliest days of hyperspectral image and signal processing [1, 14, 44, 106]. No matter the spatial resolution, the spectral signatures collected in natural environments are invariably a mixture of the signatures of the various materials found within the spatial extent of the ground instantaneous field view of the imaging instrument. For instance, the pixel vector labeled as “vegetation” in Fig. 2.1 may actually comprise a mixture of vegetation and soil, or different types of soil and vegetation canopies. In this case, several spectrally pure signatures (called *endmembers* in hyperspectral imaging terminology) are combined into the same (mixed) pixel. The availability of hyperspectral imagers with a number of spectral bands that exceeds the number of spectral mixture components [47] has allowed to cast the unmixing problem in terms of an over-determined system of equations in which, given a set of *endmembers*, the actual unmixing to determine apparent abundance fractions can be defined in terms of a numerical inversion process [16].

A standard technique for spectral mixture analysis is *linear* spectral unmixing [55, 90, 92], which assumes that the collected spectra at the spectrometer can be expressed in the form of a linear combination of endmembers weighted by their corresponding abundances. It should be noted that the linear mixture model assumes minimal secondary reflections and/or multiple scattering effects in the data collection procedure, and hence the measured spectra can be expressed as a linear combination of the spectral signatures of materials present in the mixed pixel (see Fig. 2.2a). Although the linear model has practical advantages such as ease of implementation and flexibility in different applications [28], *nonlinear* spectral unmixing may best

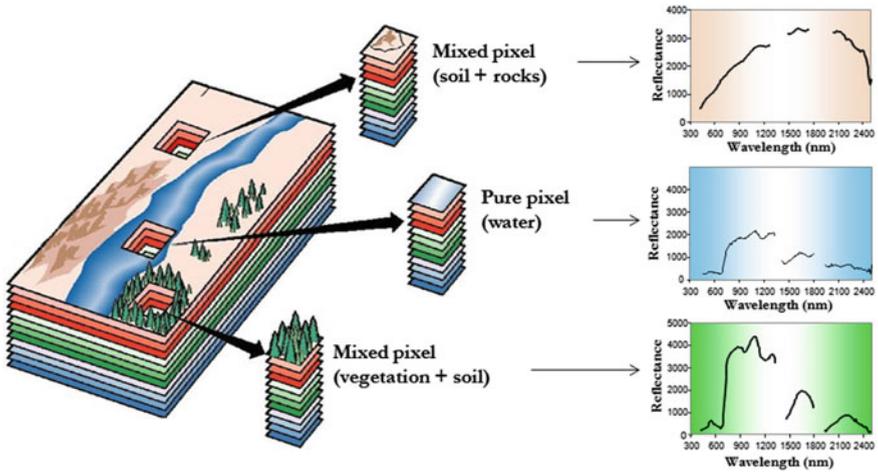


Fig. 2.1 Mixed pixels in hyperspectral imaging

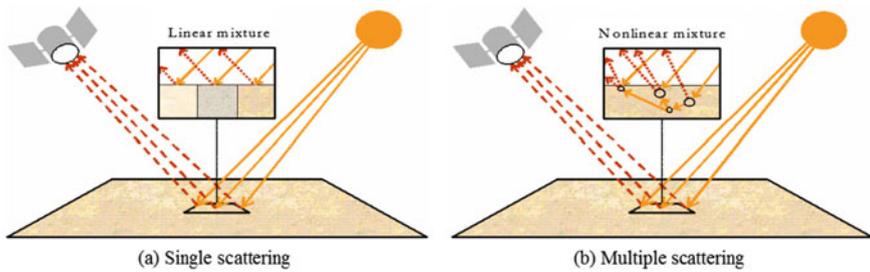


Fig. 2.2 Linear versus nonlinear mixture models: single versus multiple scattering

characterize the resultant mixed spectra for certain endmember distributions, such as those in which the endmember components are randomly distributed throughout the field of view of the instrument [49, 95]. In those cases, the mixed spectra collected at the imaging instrument is better described by assuming that part of the source radiation is multiply scattered before being collected at the sensor (see Fig. 2.2b). In this case, interactions can be at a *classical*, or *multilayered*, level or at a *microscopic*, or *intimate*, level. Mixing at the classical level occurs when light is scattered from one or more objects, is reflected off additional objects, and eventually is measured by hyperspectral imager. A nice illustrative derivation of a multilayer model is given by Borel and Gerstl [18] who show that the model results in an infinite sequence of powers of products of reflectances. Generally, however, the first order terms are sufficient and this leads to the bilinear model. Microscopic mixing occurs when two materials are homogeneously mixed [52]. In this case, the interactions consist of photons emitted from molecules of one material and absorbed by molecules of another material, which may in turn emit more photons. The mixing is modeled by Hapke [52] as occurring at the *albedo* level, i.e., the fraction of solar energy

reflected from the Earth, and not at the reflectance level (for more details on the physical quantities acquired by passive cameras see Chap. 3). The apparent albedo of the mixture is a linear average of the albedos of the individual substances but the reflectance is a nonlinear function of albedo, thus leading to a different type of nonlinear model.

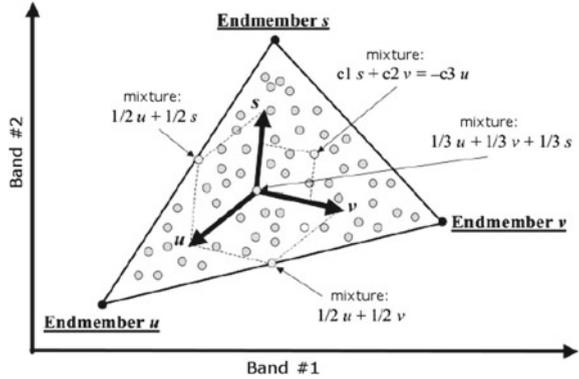
In the following, we focus on describing recent advances in the linear spectral unmixing domain. The reason is that, despite its simplicity, it is an acceptable approximation of the light scattering mechanisms in many real scenarios. Furthermore, in contrast to nonlinear mixing, the linear mixing model is the basis of a plethora of unmixing models and algorithms spanning back at least 25 years. A sampling can be found in [12, 27, 54, 56, 58, 63, 64, 78, 85, 87, 90, 92, 105, 107, 130], see also [14] and references therein. As shown in Fig. 2.2a, the linear mixture model assumes that mixed pixels are a linear combination of the endmembers. This scenario holds when the mixing scale is macroscopic [108] and the incident light interacts with just one material, as is the case in checkerboard type scenes [36, 51]. In this case, the mixing occurs within the instrument itself. It is due to the fact that the resolution of the instrument is not fine enough. The light from the materials, although almost completely separated, is mixed within the measuring instrument.

In order to define the linear mixture model in mathematical terms, let us assume that  $\mathbf{Y} \in \mathbb{R}^{l \times n}$  is a hyperspectral image with  $l$  bands and  $n$  pixels. In this case the matrix  $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_n]$  represents a hyperspectral image in a matrix form, in which the columns of the matrix  $\mathbf{Y}$  are the spectral signatures of the image pixels  $\mathbf{y}_i$ , and the rows of  $\mathbf{Y}$  are the bands of the hyperspectral image. Under the linear mixture assumption, we can model the hyperspectral data as follows:

$$\mathbf{Y} = \mathbf{M}\mathbf{A} + \mathbf{N}, \quad (2.1)$$

where  $\mathbf{M} \in \mathbb{R}^{l \times p}$ ,  $\mathbf{M} = [\mathbf{m}_1, \dots, \mathbf{m}_p]$  is a matrix containing endmembers  $\mathbf{m}_i$  in columns and  $\mathbf{A} \in \mathbb{R}^{p \times n}$ ,  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]$  contains the abundance fractions  $a_{j,k}$  associated to each endmember in each pixel. Finally,  $\mathbf{N} \in \mathbb{R}^{l \times n}$  is a matrix which represents the noise introduced in the model by the acquisition process. Usually two constraints are imposed to the abundance fractions in the linear mixture model. The first one is the abundance non-negativity (ANC), which enforces to all the abundances fractions to be non-negative [31], i.e.  $a_{j,k} \geq 0$ ,  $j = 1, \dots, p$ ,  $k = 1, \dots, n$ . The second constraint is the abundance sum-to-one (ASC), which enforces the abundances of a given pixel to sum to one, i.e.  $\sum_{j=1}^p a_{j,k} = 1$ ,  $k = 1, \dots, n$ . The unmixing process which considers both constraints is called fully constrained linear spectral unmixing (FCLSU). The linear mixture model can be interpreted graphically by using a scatter plot between two bands or, more generally, between two non-colinear projections of the spectral vectors. For illustrative purposes, Fig. 2.3 provides a simple graphical interpretation in which the endmembers are the most extreme pixels defining a simplex which encloses all the other pixels in the data, so that we can express every pixel inside the simplex as a linear combination of the endmembers. As a result, a key aspect when considering the linear mixture model is the correct identification of the endmembers, which are extreme points in the  $l$ -dimensional space.

**Fig. 2.3** Graphical interpretation of the linear mixture model



The solution of the linear spectral mixture problem described in (2.1) relies on two major requirements:

1. A successful estimation of how many endmembers,  $p$ , are present in the input hyperspectral scene  $\mathbf{Y}$ , and
2. the correct determination of a set  $\mathbf{M}$  of  $p$  endmembers and their correspondent abundance fractions at each pixel.

In order to address these issues, a standard spectral unmixing chain consisting of three steps is generally applied. In a first step, an (optional) dimensionality reduction step is conducted. This step is strongly related to the estimation of the number of endmembers present in the hyperspectral scene,  $p$ . Once the number of endmembers has been determined, an endmember extraction step identifies the pure spectral signatures present in a scene. Finally, the abundance estimation step requires as input the endmember signatures obtained in the endmember extraction process and produces as output the set of abundance maps associated to each endmember. Figure 2.4 shows the different steps involved in the processing chain, which are briefly summarized next and described in more detail in the following subsections (discussing specific implementation options for each step).

1. **Dimensionality reduction.** The dimensionality of the space spanned by spectra from an image is generally much lower than the available number of bands. Identifying appropriate subspaces facilitates dimensionality reduction, improving algorithm performance and data storage complexity. Furthermore, if the linear mixture model is accurate, the signal subspace dimension is one less than the number of endmembers, a crucial figure in hyperspectral unmixing.
2. **Endmember extraction.** This step consists in identifying the endmembers in the scene. *Geometrical* approaches exploit the fact that linearly mixed vectors are in a simplex set or in a positive cone. *Statistical* approaches focus on using parameter estimation techniques to determine endmembers. Different techniques may or may not include spatial information and assume or not the presence of pure pixels in the original data set.

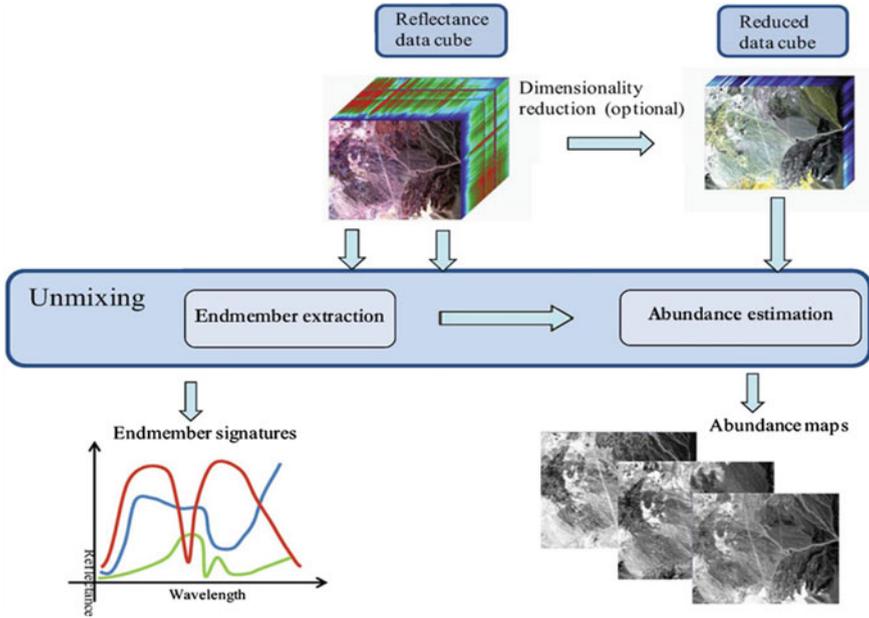


Fig. 2.4 Spectral unmixing chain

3. **Abundance estimation.** Given the identified endmembers, the abundance estimation step consists in solving a constrained optimization problem which minimizes the residual between the observed spectral vectors and the linear space spanned by the inferred endmembers in order to derive fractional abundances which are, very often, constrained to be nonnegative and to sum to one (i.e., they belong to the probability simplex). There are, however, some hyperspectral unmixing approaches in which the endmember determination and inversion steps are implemented simultaneously.

### 2.2.1 Dimensionality Reduction

The number of endmembers  $p$  present in a given scene is, very often, much smaller than the number of bands  $l$ . Therefore, assuming that the linear model is a good approximation, spectral vectors lie in or very close to a low-dimensional linear subspace. The identification of this subspace enables low-dimensional yet accurate representation of spectral vectors. It is usually advantageous and sometimes necessary to operate on data represented in the signal subspace. Therefore, a signal subspace identification algorithm is often required as a first processing step in the spectral unmixing chain. Unsupervised subspace identification has been addressed in many ways. Projection techniques seek for the best subspaces to represent data by optimizing

objective functions. For example, principal component analysis (PCA) maximizes the signal variance; singular value decomposition (SVD) maximizes power; minimum noise fraction (MNF) and noise-adjusted principal components (NAPC) minimize the ratio of noise power to signal power. NAPC is mathematically equivalent to MNF [70] and can be interpreted as a sequence of two principal component transforms: the first applied to the noise and the second applied to the transformed data set.

The identification of the signal subspace is a model order inference problem to which information theoretic criteria come to mind. These criteria have in fact been used in hyperspectral applications [30] adopting the approach introduced by Wax and Kailath [127]. In turn, Harsanyi, Farrand, and Chang [53] developed a Neyman-Pearson detection theory-based thresholding method to determine the number of spectral endmembers in hyperspectral data, referred to as virtual dimensionality (VD). This method is based on a detector built on the eigenvalues of the sample correlation and covariance matrices. A modified version includes a noise-whitening step [30]. The hyperspectral signal identification with minimum error (HYSIME) adopts a minimum mean squared error based approach to infer the signal subspace. The method is eigendecomposition based, unsupervised, and fully-automatic (i.e., it does not depend on any tuning parameters). It first estimates the signal and noise correlation matrices and then selects the subset of eigenvalues that best represents the signal subspace in the least square error sense.

### 2.2.2 *Endmember Extraction*

Over the last decade, several algorithms have been developed for automatic or semi-automatic extraction of spectral endmembers by assuming the presence of pure pixels in the hyperspectral data [92]. Classic techniques include the pixel purity index (PPI), N-FINDR, iterative error analysis (IEA), convex cone analysis (CCA), vertex component analysis (VCA), and orthogonal subspace projection (OSP), among many others [14]. Other advanced techniques for endmember extraction have been recently proposed [9, 26, 32, 33, 81, 89, 126, 132], but none of them considers spatial adjacency. However, one of the distinguishing properties of hyperspectral data is the multivariate information coupled with a two-dimensional (pictorial) representation amenable to image interpretation. Subsequently, most endmember extraction algorithms listed above could benefit from an integrated framework in which both the spectral information and the spatial arrangement of pixel vectors are taken into account. An example of this situation is given in Fig. 2.5, in which a hyperspectral data cube collected over an urban area (high spatial correlation) is modified by randomly permuting the spatial coordinates of the pixel vectors, thus removing the spatial correlation. In both scenes, the application of a spectral-based endmember extraction method would yield the same analysis results while it is clear that a spatial-spectral technique could incorporate the spatial information present in the original scene into the endmember searching process.

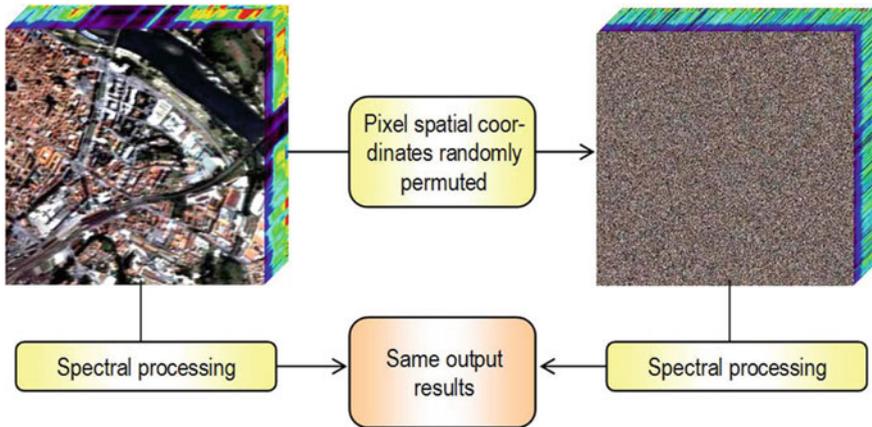
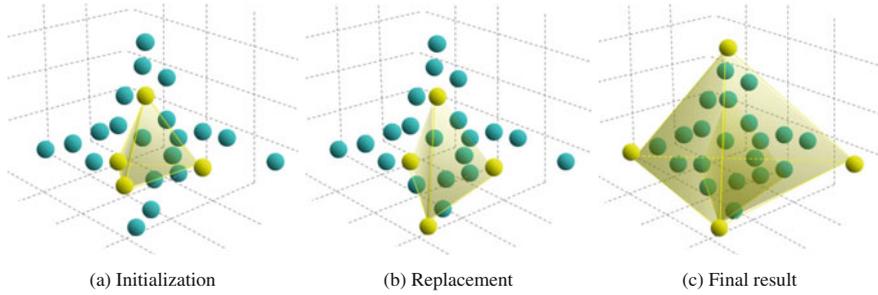


Fig. 2.5 Example illustrating the importance of spatial information in hyperspectral analysis

To the best of our knowledge, only a few attempts exist in the literature aimed at including the spatial information in the process of extracting spectral endmembers. Extended morphological operations [93] have been used as a baseline to develop the automatic morphological endmember extraction algorithm (AMEE) for spatial-spectral endmember extraction. Also, spatial averaging of spectrally similar endmember candidates found via SVD was used in the development of the spatial-spectral endmember extraction algorithm (SSEE). In the following, we describe in more detail three selected spectral-based algorithms (N-FINDR, OSP and VCA) and two spatial-spectral endmember extraction algorithms (AMEE and SSEE) that will be used in our comparisons in this chapter. The reasons for our selection are: (1) these algorithms are representative of the class of convex geometry-based and spatial processing-based techniques which have been successful in endmember extraction; (2) they are fully automated; (3) they always produce the same final results for the same input parameters; and (4) the number of endmembers to be extracted,  $p$ , is an input parameter for all algorithms, while AMEE and SSEE have additional input parameters related to the definition of spatial context around each pixel in the scene. This section concludes with a description of algorithms that, as opposed to the previously mentioned ones, do not assume the presence of pure pixels in the hyperspectral data. Techniques in this category comprise minimum volume simplex analysis (MVSA) and a variable splitting augmented Lagrangian approach (SISAL). Also, we deliberately do not cover sparse unmixing methods [60], which are detailed in other chapters of this book.

### 2.2.2.1 N-FINDR

This algorithm looks for the set of pixels with the largest possible volume by *inflating* a simplex inside the data. The procedure begins with a random initial selection of pixels (see Fig. 2.6a). Every pixel in the image must be evaluated in order to refine



**Fig. 2.6** Graphical representation of the N-FINDR algorithm

the estimate of endmembers, looking for the set of pixels that maximizes the volume of the simplex defined by selected endmembers. The volume of the simplex is calculated with every pixel in the place of each endmember. The corresponding volume is calculated for every pixel in each endmember position by replacing that endmember and finding the resulting volume (see Fig. 2.6b). If the replacement results in an increase of volume, the pixel replaces the endmember. This procedure is repeated until there are no more endmember replacements (see Fig. 2.6c). The mathematical definition of the volume of a simplex formed by a set of endmember candidates is proportional to the determinant of the set augmented by a row of ones. The determinant is only defined in the case where the number of features is  $p - 1$ ,  $p$  being the number of desired endmembers [29]. Since in hyperspectral data typically  $l \gg p$ , a transformation that reduces the dimensionality of the input data is required. Often, the PCA transform has been used for this purpose, although another widely used alternative that decorrelates the noise in the data is MNF. A possible shortcoming of this algorithm is that different random initializations of N-FINDR may produce different final solutions. In this chapter, we consider an N-FINDR algorithm implemented in an iterative fashion, so that each sequential run is initialized with the previous algorithm solution, until the algorithm converges to a simplex volume that cannot be further maximized.

### 2.2.2.2 Orthogonal Subspace Projection (OSP)

This algorithm starts by selecting the pixel vector with maximum length in the scene as the first endmember. Then, it looks for the pixel vector with the maximum absolute projection in the space orthogonal to the space linearly spanned by the initial pixel, and labels that pixel as the second endmember. A third endmember is found by applying an orthogonal subspace projection to the original image [54]. This is done by selecting the signature that has the maximum orthogonal projection in the space orthogonal to the space linearly spanned by the first two endmembers. This procedure is repeated until the desired number of endmembers,  $p$ , is found [98]. A shortcoming

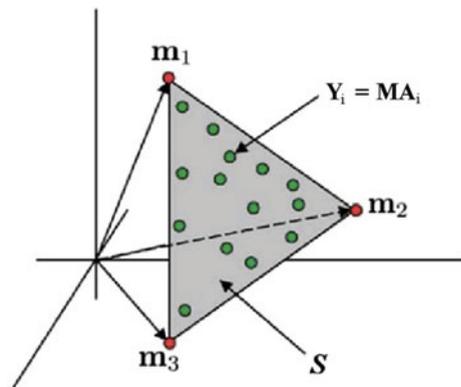
of this algorithm is its sensitivity to noise, since outliers are good candidates to be selected in the iterative process adopted by OSP. The VCA method discussed in the following subsection addresses this issue.

### 2.2.2.3 Vertex Component Analysis (VCA)

This algorithm also makes use of the concept of orthogonal subspace projections. However, as opposed to the OSP algorithm described above, VCA exploits the fact that the endmembers are the vertices of a simplex, and that the affine transformation of a simplex is also a simplex [84]. As a result, VCA models the data using a positive cone, whose projection onto a properly chosen hyperplane is another simplex whose vertices are the final endmembers. After projecting the data onto the selected hyperplane, the VCA projects all image pixels to a random direction and uses the pixel with the largest projection as the first endmember. The other endmembers are identified in sequence by iteratively projecting the data onto a direction orthogonal to the subspace spanned by the endmembers already determined, using a procedure that is quite similar to that used by OSP. The new endmember is then selected as the pixel corresponding to the extreme projection, and the procedure is repeated until a set of  $p$  endmembers is found [84]. For illustrative purposes, Fig. 2.7 shows a toy example depicting an image with three bands and three endmembers. Due to the mixing phenomenon, all the data is in the plane  $S$ . If we project the data onto that plane we can represent the same data in two dimensions instead of three. Then we can apply OSP to the projected dataset in order to obtain the endmembers.

A possible shortcoming of the VCA algorithm can be illustrated by the following example: if there are two endmembers with similar spectral signatures and the power of noise is high, then the subspace identification step could miss one of the two similar endmembers. This problem could be avoided by using spatial information as follows. The idea is that, although the endmembers are very similar in the spectral domain, they may be located in different areas in the spatial domain. As a result,

**Fig. 2.7** Toy example illustrating the impact of subspace projection on endmember identification



spatial information could help in the distinction of the endmembers. In the following subsections we describe different algorithms which make use of spatial information in order to solve some of these potential problems in the endmember identification process.

### 2.2.2.4 Automatic Morphological Endmember Extraction (AMEE)

The AMEE [91] algorithm runs on the full data cube with no dimensional reduction, and begins by searching spatial neighborhoods around each pixel vector in the image for the most spectrally pure and mostly highly mixed pixel. This task is performed by using extended mathematical morphology operators [93] of dilation and erosion, which are graphically illustrated in Fig. 2.8. Here, dilation selects the most spectrally pure pixel in a local neighborhood around each pixel vector, while erosion selects the most highly mixed pixel in the same neighborhood. Each spectrally pure pixel is then assigned an *eccentricity* value, which is calculated as the spectral angle (SA) [28, 64] between the most spectrally pure and mostly highly mixed pixel for each given spatial neighborhood. This process is repeated iteratively for larger spatial neighborhoods up to a maximum size that is predetermined. At each iteration the eccentricity values of the selected pixels are updated. The final endmember set is obtained by applying a threshold to the resulting greyscale eccentricity image, which results in a large set of endmember candidates. The final endmembers are extracted after applying the OSP method to the set of candidates in order to derive a final set of spectrally distinct endmembers  $\mathbf{M}$ , where  $p$  is an input parameter to the OSP algorithm.

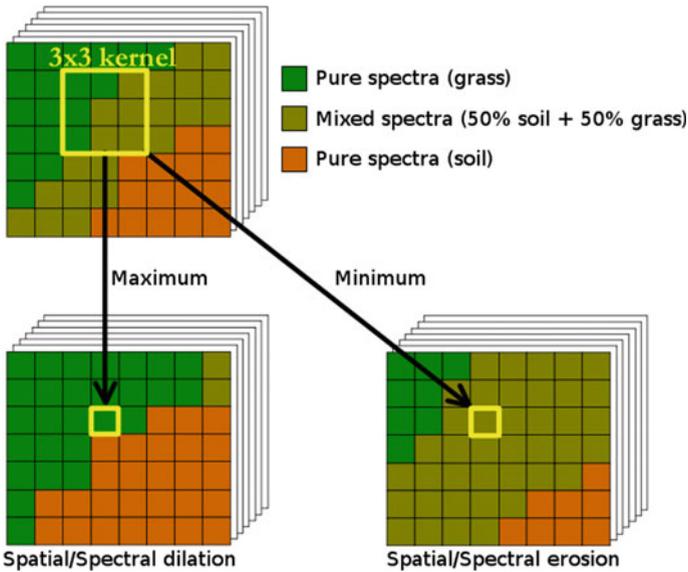
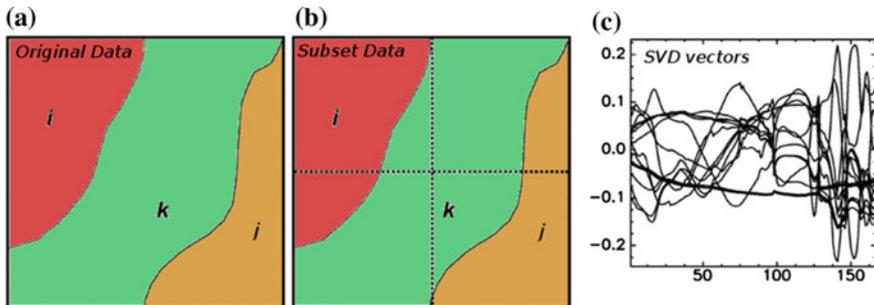


Fig. 2.8 Extended morphological operations of erosion and dilation

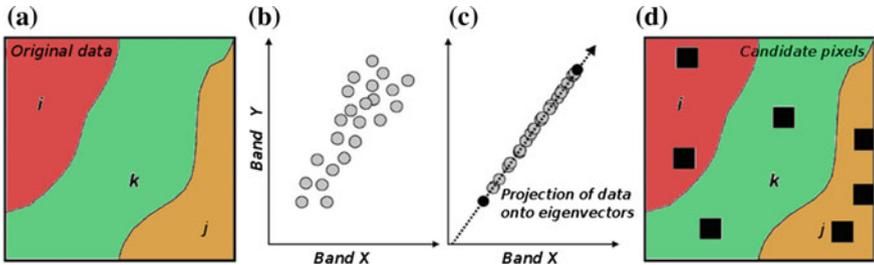
### 2.2.2.5 Spatial Spectral Endmember Extraction (SSEE)

The SSEE algorithm uses spatial constraints to improve the relative spectral contrast of endmember spectra that have minimal unique spectral information, thus improving the potential for these subtle yet potentially important endmembers to be selected. With SSEE, the spatial characteristics of image pixels are used to increase the relative spectral contrast between spectrally similar but spatially independent endmembers. The SSEE algorithm searches an image with a local search window centered around each pixel vector and comprises four steps [100]. First, the SVD transform is applied to determine a set of eigenvectors that describe most of the spectral variance in the window or partition (see Fig. 2.9). Second, the entire image data are projected onto the previously extracted eigenvectors to determine a set of candidate endmember pixels (see Fig. 2.10). Then, spatial constraints are used to combine and average spectrally similar candidate endmember pixels by testing, for each candidate pixel vector, which other pixel vectors are sufficiently similar in spectral sense (see Fig. 2.11). Instead of using a manual procedure as recommended by the authors in [100], we have used the OSP technique in order to derive a final set of spectrally distinct endmembers  $\mathbf{M}$ , where  $p$  is an input parameter to the OSP algorithm.

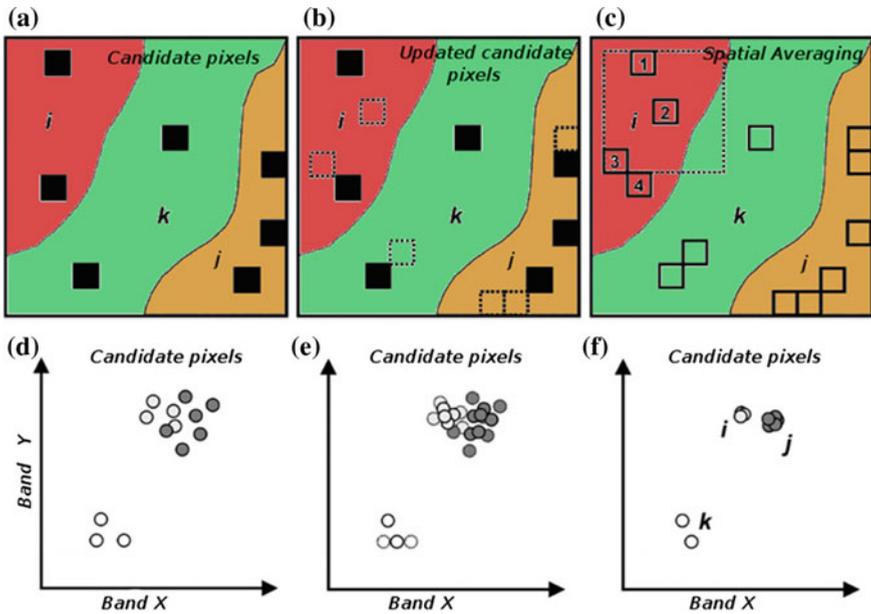
At this point, it is important to note that SSEE includes spatial information in a different way as AMEE does. The SSEE method uses first a spectral SVD method to extract some candidate endmembers and then includes the spatial information. On the other hand, AMEE combines the spatial and spectral information at the same time using extended morphological operations, and then uses a spectral endmember extraction technique in order to select the final endmember set. In both cases (as it is also the case of all endmember identification algorithms discussed thus far) the assumption is that pure spectral signatures are present in the original hyperspectral data. In the following subsection we describe methods which operate under the assumption that pure spectral signatures may not be present at all in the original hyperspectral scene.



**Fig. 2.9** First step of the SSEE algorithm. **A** Original data. **B** Subset data after spatial partitioning. **C** Set of representative SVD vectors used to describe spectral variance. (Figure reproduced from [100])



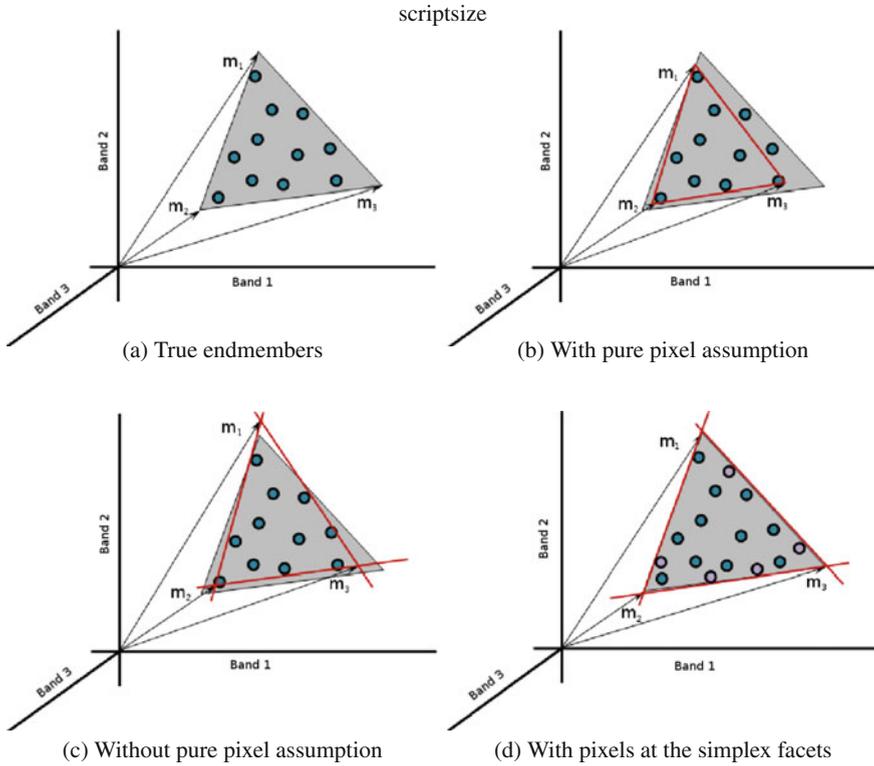
**Fig. 2.10** Second step of the SSEE algorithm. **A** Original data. **B** Spectral distribution in 2-dimensional space. **C** Projection of data onto eigenvectors. **D** Set of candidate pixels. (Figure reproduced from [100])



**Fig. 2.11** Third step of the SSEE algorithm. **A** Set of candidate pixels. **B** Updated candidate pixels after including pixels which are spectrally similar to those in the original set. **C** Spatial averaging process of candidate endmember pixels using a sliding window centered on each candidate. **D** First iteration of spatial-spectral averaging. Averaged pixels shown as thick lines, with original pixels shown as *thinner lines*. **E** Second iteration of spatial-spectral averaging. **F** Continued iterations compress endmembers into clusters with negligible variance. (Figure reproduced from [100])

### 2.2.2.6 Algorithms Without the Pure Pixel Assumption

This section describes endmember identification techniques which do not operate under the pure pixel assumption [57, 94]. In this case, the algorithms do not need the presence of pure pixels in the dataset in order to generate the endmembers. Figure 2.12



**Fig. 2.12** Illustration of different strategies for endmember extraction

shows a graphical interpretation of the difference between algorithms that assume and do not assume the presence of pure pixels in the dataset. Specifically, Fig. 2.12a represents the true endmembers. In this case, there is no pixel at the simplex vertices so the endmembers are not present in the original data. Figure 2.12b represents a possible solution of an algorithm which does not assume the presence of pure pixels in the dataset. As we can see in Fig. 2.12b, there are two pixels outside of the simplex, which are outliers in this particular case. Figure 2.12c represents a possible solution of a method which does not assume the presence of pure pixels in the data. In this case, the algorithm tries to estimate a set of endmembers by enclosing the whole dataset. This approach does not guarantee the correct identification of endmembers in the case that the data are highly mixed and there are no pixels in the facets of the simplex. However, if there are pixels in the simplex facets, the true endmembers can be correctly identified even if there are no pixels at the simplex vertex, as depicted in Fig. 2.12d.

Most of the techniques in this category adopt a minimum volume strategy aimed at finding the endmember matrix  $\mathbf{M}$  by minimizing the volume of the simplex defined by its columns and containing the endmembers. This is a non-convex optimization

problem much harder than those considered in the previous subsection in which the endmembers are assumed to belong to the input hyperspectral image.

Craig's seminal work [38] established the concepts regarding the algorithms of minimum volume type. Most of these algorithms formulate the endmember estimation as the nonnegative matrix factorization of the mixing and abundance matrices [9, 73, 81, 96, 131, 133], with a minimum volume constraint imposed on  $\mathbf{M}$ . Non-negative matrix factorization is a hard non-convex optimization problem prone to get stuck in local minima. Aiming at obtaining lighter algorithms with more desirable convergence properties, the works [2, 10, 25, 71] sidestep the matrix factorization by formulating the endmember estimation as an optimization problem with respect to  $\mathbf{Q} = \mathbf{M}^{-1}$ . The MVSA and SISAL algorithms implement a robust version of the minimum volume concept. Robustness is introduced by allowing the ANC to be violated. These violations are weighted using a soft constraint given by the hinge loss function, hinge( $\mathbf{x}$ ), an elementwise operator that returns 0 if  $x_i \geq 0$  and  $-x_i$  if  $x_i < 0$ , for every element  $x_i$  in  $\mathbf{x}$ . After reducing the dimensionality of the input data from  $l$  to  $p - 1$ , MVSA/SISAL aim at solving the following optimization problem:

$$\begin{aligned} \widehat{\mathbf{Q}} &= \arg \max_{\mathbf{Q}} \log(|\det(\mathbf{Q})|) - \lambda \mathbf{1}_p^T \text{hinge}(\mathbf{Q}\mathbf{Y}) \mathbf{1}_n \\ \text{s.t.} \quad & \mathbf{1}_p^T \mathbf{Q} = \mathbf{q}_m, \end{aligned} \quad (2.2)$$

where  $\mathbf{Q} \equiv \mathbf{M}^{-1}$ ,  $\mathbf{1}_p$  and  $\mathbf{1}_n$  are column vectors of ones of sizes  $p$  and  $n$ , respectively,  $\mathbf{q}_m \equiv \mathbf{1}_p^T \mathbf{Y}_p^{-1}$  with  $\mathbf{Y}_p$  being any set of linearly independent spectral vectors taken from the hyperspectral data set  $\mathbf{Y}$ , and  $\lambda$  is a regularization parameter. Here, maximizing  $\log(|\det(\mathbf{Q})|)$  is equivalent to minimizing the volume of  $\mathbf{M}$ .

### 2.2.3 Abundance Estimation

Once a set of endmembers  $\mathbf{M}$  have been extracted, their correspondent abundance fractions  $\mathbf{A}$  can be estimated (in least squares sense) by the following unconstrained expression [28]:

$$\mathbf{A} \approx (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \mathbf{Y}. \quad (2.3)$$

However, it should be noted that the fractional abundance estimations obtained by means of Eq. (2.3) do not satisfy the ASC and ANC constraints. As indicated in [30], a non-negative constrained least squares (NCLS) algorithm can be used to obtain a solution to the ANC-constrained problem in an iterative fashion [31]. In order to take care of the ASC constraint, we replace the hard constraint  $\mathbf{1}^T \mathbf{A} = 1$  by the soft constraint  $\sqrt{\delta} \|\mathbf{1}^T \mathbf{A} - 1\|_2^2$  added to the quadratic data term  $\|\mathbf{Y} - \mathbf{M}\mathbf{A}\|_2^2$ . This is equivalent to using a new endmember signature matrix, denoted by  $\mathbf{M}'$ , and a modified version of the abundance estimates  $\mathbf{A}$ , denoted by  $\mathbf{A}'$ , are introduced as follows:

$$\mathbf{M}' = \begin{bmatrix} \mathbf{M} \\ \delta \mathbf{1}^T \end{bmatrix}, \mathbf{A}' = \begin{bmatrix} \mathbf{A} \\ \delta \mathbf{1} \end{bmatrix}, \quad (2.4)$$

where  $\mathbf{1} = \underbrace{(1, 1, \dots, 1)}_p^T$  and  $\delta$  controls the impact of the ASC constraint. Using the two expressions in (2.4), a fully constrained estimate can be directly obtained from the NCLS algorithm by replacing  $\mathbf{M}$  and  $\mathbf{A}$  with  $\mathbf{M}'$  and  $\mathbf{A}'$ . The fully constrained (i.e., ASC-constrained and ANC-constrained) linear spectral unmixing model is referred to as FCLSU.

### 2.2.4 Experimental Validation

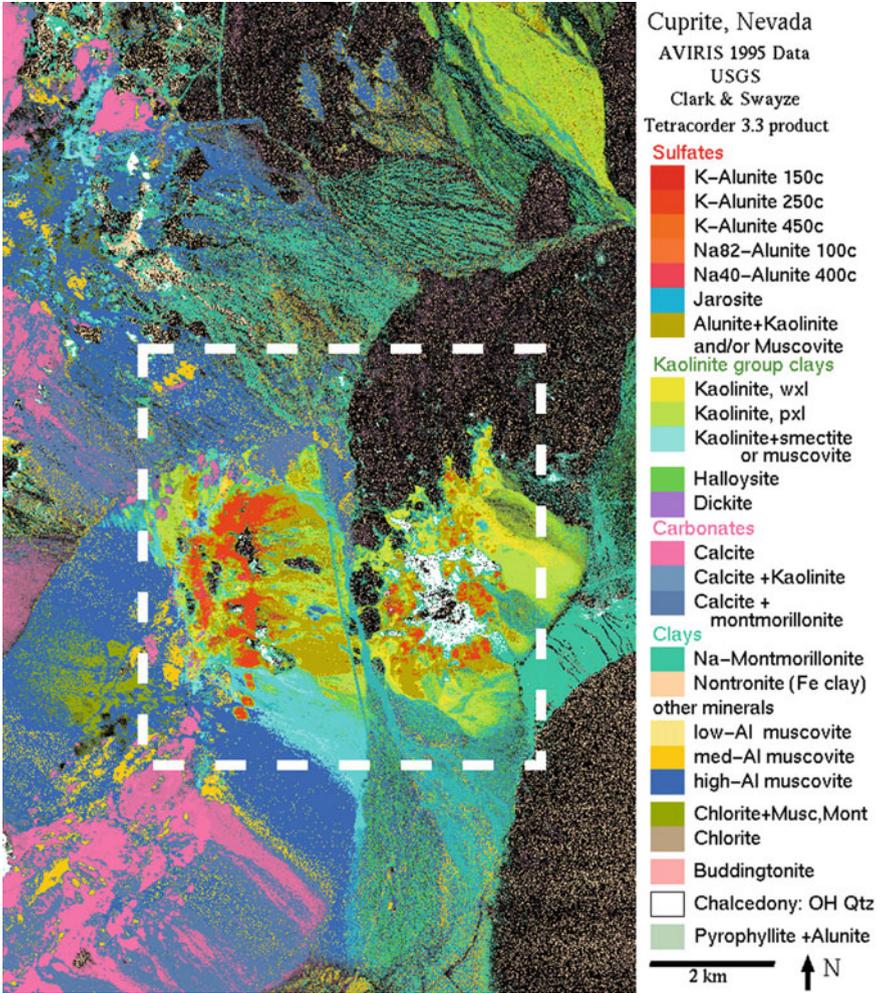
In this section we will describe the experiments performed with a real hyperspectral dataset collected by the airborne visible infrared imaging spectrometer (AVIRIS) over the Cuprite mining district. The scene, available online in reflectance units after atmospheric correction,<sup>1</sup> is characterized by the availability of some very reliable reference information available from the United States Geological Survey (USGS). Specifically, the portion used in experiments corresponds to a  $350 \times 350$ -pixel subset of the sector labeled as “f970619t01p02\_r02\_sc03.a.rfi” in the online data. The scene comprises 224 spectral bands between 0.4 and 2.5  $\mu\text{m}$ , with full width at half maximum of 10nm and spatial resolution of 20m per pixel. Prior to the analysis, several bands were removed due to water absorption and low signal-to-noise ratio (SNR) in those bands, leaving a total of 188 reflectance channels to be used in the experiments. The Cuprite site is well understood mineralogically, and has several exposed minerals of interest, all included in the USGS library considered in experiments, denoted “splib06” and released in September 2007.<sup>2</sup> In our experiments, we use spectra obtained from this library to validate endmember extraction algorithms. For illustrative purposes, Fig. 2.13 shows a mineral map produced in 1995 by USGS, in which the Tricorder 3.3 software product was used to map different minerals present in the Cuprite mining district.<sup>3</sup> It should be noted that the Tricorder map is only available for hyperspectral data collected in 1995, while the publicly available AVIRIS Cuprite data was collected in 1997. Therefore, a direct comparison between the 1995 USGS map and the 1997 AVIRIS data (as well as a comparison in terms of fractional abundances) is not possible.

We show a comparison of the results obtained for the endmember extraction algorithms in terms of accuracy and also in terms of computational complexity. Accuracy is measured in terms of the spectral angle (SA), i.e., the angle between two spectral signature vectors. Table 2.1 tabulates the SA scores, in degrees, obtained after comparing the USGS library spectra of *alunite*, *buddingtonite*, *calcite*, *kaolinite*

<sup>1</sup><http://aviris.jpl.nasa.gov/html/aviris.freedata.html>.

<sup>2</sup><http://speclab.cr.usgs.gov/spectral.lib06>.

<sup>3</sup>[http://speclab.cr.usgs.gov/cuprite95.tgif.2.2um\\_map.gif](http://speclab.cr.usgs.gov/cuprite95.tgif.2.2um_map.gif).

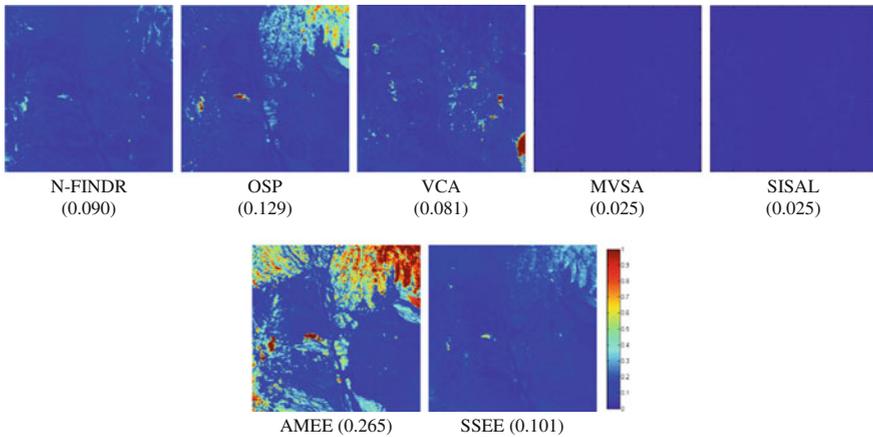


**Fig. 2.13** USGS map showing the location of different minerals in the Cuprite mining district in Nevada. The map is available online at: [http://speclab.cr.usgs.gov/cuprite95.tgif.2.2um\\_map.gif](http://speclab.cr.usgs.gov/cuprite95.tgif.2.2um_map.gif). The white rectangle depicts the area used in our experiments

and *muscovite*, with the corresponding endmembers extracted by different algorithms from the AVIRIS Cuprite scene. In all cases, the input parameters of the different endmember extraction methods tested have been carefully optimized so that the best performance for each method is reported. The smaller the SA values across the five minerals in Table 2.1, the better the results. It should be noted that Table 2.1 only displays the smallest SA scores of all endmembers with respect to each USGS signature for each algorithm. As a reference, the mean SA values across all five USGS signatures is also reported. The number of endmembers to be extracted was

**Table 2.1** Spectral similarity scores (in degrees) between USGS mineral spectra and their corresponding endmembers extracted by several algorithms from the AVIRIS Cuprite scene

Algorithm	Alunite	Buddingtonite	Calcite	Kaolinite	Muscovite	Mean
	GDS84	GDS85	WS272	KGa-1	GDS107	
N-FINDR	4.81	4.29	7.60	9.92	5.05	6.33
OSP	4.81	4.16	9.52	10.76	5.29	6.91
VCA	6.91	5.38	9.53	9.65	6.47	7.59
MVSA	12.72	8.41	5.69	15.04	5.36	9.44
SISAL	9.78	5.13	12.78	13.53	8.00	9.84
AMEE	4.81	4.17	5.87	8.74	4.61	5.64
SSEE	4.81	4.16	8.48	10.73	4.63	6.57



**Fig. 2.14** Errors measured for various endmember extraction algorithms after reconstructing the AVIRIS Cuprite scene

set to  $p = 19$  in all experiments after the consensus reached between HYSIME [11] and the VD concept [30], implemented using  $P_F = 10^{-3}$  as the input false alarm probability. In this experiment, the best performance (in terms of SA) was obtained by the endmember extraction algorithms AMEE which include both spatial and spectral information.

Additionally, Fig. 2.14 shows the root mean squared error (RMSE) maps obtained after reconstructing the AVIRIS Cuprite scene using  $p = 19$  endmembers extracted by different methods. As shown by this experiment, MVSA and SISAL provide the best results in terms of image reconstruction although they may provide unrealistic endmembers, as described in the previous experiment. To conclude this section, Table 2.2 reports the processing times of the compared algorithms.

**Table 2.2** Processing times (in seconds) measured in a desktop PC with intel core i7 920 CPU at 2.67 Ghz with 4 GB of RAM

Algorithm	Total processing time
N-FINDR	466.08
OSP	136.09
VCA	31.12
MVSA	$\approx 25000$
SISAL	170.40
AMEE	76.06
SSEE	1051.23

## 2.3 Classification

The general hyperspectral image *classification* problem can be described as follows: At the input a  $B$ -band hyperspectral data cube is given, which can be considered as a set of  $n$  pixel vectors  $\mathbf{X} = \{\mathbf{x}_j \in \mathbb{R}^B, j = 1, 2, \dots, n\}$ . Let  $\Omega = \{\omega_1, \omega_2, \dots, \omega_K\}$  be a set of information classes in the scene. Classification consists in assigning each pixel to one of the  $K$  classes of interest. An information class can represent either a physical substance (a ground cover material, for instance, *snow*, *water*, *wheat*), or a specific group of objects which may be made of several different physical materials (for instance, *roof*, *shadows*, *trees*).

In this chapter, we focus on *supervised* classification, which assumes that classes are defined by a set of training samples. Unsupervised classification, or clustering techniques have also been described in the literature. We refer the reader to [35] for a survey on unsupervised methods. An important assumption for classification techniques is that the spatial resolution of the image is high enough so that the data contains mostly pure pixels, i.e., pixels representing a single information class. In the opposite case, i.e., when the data is mostly composed of mixed pixels, spectral unmixing methods are more appropriate for image analysis.

The first attempts to classify hyperspectral images would assign each pixel to one of the classes based on its spectrum only [66]. These are often referred to as *pixelwise* (or non-contextual) classification techniques. However, with the increase of spatial resolution of hyperspectral sensors, objects in the image are typically large compared to the size of a pixel. In the ideal case, all the pixels of these objects should be assigned to the same class. It has then become then very important to simultaneously use spectral and spatial information for image classification [88]. *Spectral-spatial* classification (also referred to as spatial-contextual) assigns each pixel to one information class based on: (1) its own spectrum; (2) information extracted from its neighborhood, i.e., the spatial information. A multitude of methods have been proposed for this purpose, which differ in the ways of extracting spatial contextual information from the image scene and in the ways of combining spectral and spatial information.

The following sections review the keystone methods of pixelwise and spectral-spatial classification. We also include a detailed explanation of a recently-proposed mathematical model for spectral-spatial classification, based on a binary partition

tree representation [76]. This method first constructs a hierarchical region-based representation of the image stored in a tree structure, and then extracts objects of interest from the tree.

### 2.3.1 Supervised Pixelwise Classification

Landgrebe et al. were seminal in exploring procedures for hyperspectral data analysis and classification [66, 67]. They adapted pattern recognition procedures for this purpose. A simplified version of their proposed classification scheme, widely used until nowadays, is depicted in Fig. 2.15. There are two inputs to the system: the hyperspectral image and a set of observations of the ground which are labeled into classes of interest. From the hyperspectral image, there is first a process of feature extraction and selection. Features can be seen as an abstraction layer with meaningful descriptors derived from the raw input. This representation should be meaningful in the sense that it must be useful for the classification problem, describing and separating the classes of interest. Some input from the training labels themselves might be used to decide which features are relevant. The features associated to every pixel can be seen as a point in a high-dimensional space. The next step consists in *training* a classifier based on the set of labeled samples, i.e., partitioning the entire feature space into  $K$  exhaustive, nonoverlapping regions, so that every point in the feature space is uniquely associated with one of the  $K$  classes.

In the *pixelwise* approach, each image pixel is seen as a pattern to classify. One possibility is to use the pixel spectrum as the set of features that describe every pixel. Since this is often redundant, it is common to perform a more sophisticated feature extraction/selection step with the goal of reducing the dimensionality of the feature set and maximizing separability between classes. Different feature extraction techniques have been proposed and explored for this purpose, such as Discriminant Analysis Feature Extraction, Decision Boundary Feature Extraction and Non-parametric Weighted Feature Extraction [39, 67]. Once this step is accomplished, each pixel is classified according to its feature set.

The set of training samples is typically obtained by visually interpreting and manually labeling a small number of pixels in the data set, or by performing an *in situ* field campaign. The training data is used to define a model of the classes in the feature space. Assuming that each class can be described by a normal distribution,

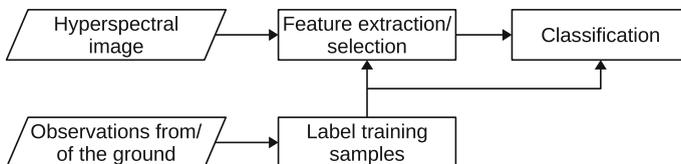


Fig. 2.15 Schematic diagram of the hyperspectral image classification process

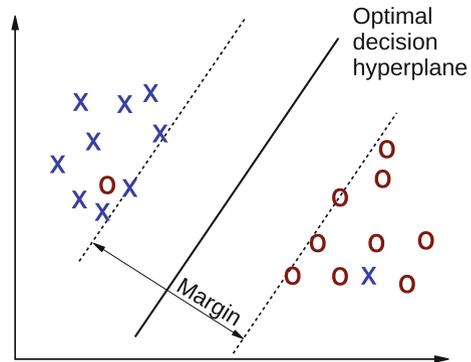
Gaussian Maximum Likelihood classification has been for many years the standard thematic mapping procedure in hyperspectral remote sensing [99]. Essentially, it assigns a given pixel to the class  $\omega_i$  that maximizes the posterior probability  $P(\omega_i|\mathbf{x})$  in the Gaussian model. A serious drawback of this method consists in the primary assumption about the shape of the class-conditional probability density functions. If this assumption is wrong, classification results are no longer accurate. Furthermore, the high number of features available, usually coupled with a limited number of training samples, makes estimation of statistical class parameters unreliable. As a result, with a limited training set, the classification accuracy tends to decrease as the dimensionality increases, an issue often referred to as the *Hughes phenomenon* [66, 88]. High-dimensional spaces are mostly empty [62], making density estimation even more difficult.

In the 1990s, *neural network* approaches for classifying hyperspectral images received a lot of attention [7, 80, 111, 129]. Neural network models have an advantage over statistical methods in that they are distribution-free and thus no prior knowledge about the statistical distribution of classes is needed. In a neural network, a set of weighted sums and nonlinearities describe the function that classifies the input features. The training procedure involves finding the appropriate weights, which is done iteratively. The interest in such approaches greatly increased in the 1990s with improvements in the training techniques [6]. Yet there has been a limited use of neural networks for hyperspectral image classification primarily due to their algorithmic and training complexity [99]. Genetic algorithms for classification of hyperspectral data have also been presented [121], capable to deal with nonlinearly separable patterns but computationally demanding.

Early in this century, kernel methods such as Support Vector Machines (SVMs) have become very popular for hyperspectral image analysis, proving to be extremely well suited to classify high-dimensional data when a limited number of training samples is available [22, 48]. The SVM method seeks to trace an optimal hyperplane that linearly separates features into two groups with a maximum margin (see Fig. 2.16). A *soft margin* is typically used, where misclassified samples (i.e., on the wrong side of the hyperplane) are tolerated but penalized. To account for nonlinear separation boundaries, the data points are mapped to a higher-dimensional space by using a *kernel* function, and the linear SVM classification is performed on the transformed space. More details on SVMs can be found in Chap. 10. For hyperspectral image classification, two kernel functions have been widely used: the polynomial kernel and the Gaussian radial basis function. While initially devised for binary classification, the  $K$ -class problem can be solved by training  $K$  classifiers to distinguish each class from all the rest (*one vs all*) or  $K(K - 1)/2$  classifiers to distinguish every pair of classes (*one vs one*) [104].

To conclude, SVMs directly exploit the geometrical properties of data, without involving a density estimation procedure. This method can efficiently handle high-dimensional data, exhibiting low sensitivity to the Hughes phenomenon [59, 79]. Therefore, it is an excellent approach to attenuate the usually time-consuming feature extraction/selection procedure, thus simplifying the traditional pattern recognition scheme of Fig. 2.15. In hyperspectral image classification with SVMs, the dimen-

**Fig. 2.16** Support vector machines (SVMs) search for an optimal hyperplane to linearly separate the data points with a maximum margin. This SVM uses a *soft* margin, adding robustness to difficult samples



sionality reduction step is often skipped and the spectrum directly used as the feature vector. Finally, SVMs exhibit a good generalization capability, fully exploiting the discrimination capability of the relatively few training samples available. All these advantages of the SVM method have made it the most widely used classifier for hyperspectral data in the last decade [13].

To further boost up classification accuracies, ensemble classification systems have been investigated for hyperspectral image classification. These approaches combine multiple learning algorithms to improve the predictive accuracy. Ham et al. [50] investigated the use of Random Forest framework and Ceamanos et al. [24] proposed an SVM-based ensemble approach, where separate SVM predictions are performed for subsets of spectral bands, and all outputs are used as the input for an additional SVM classifier.

All the described approaches assign each pixel to one of the classes based on its spectral properties alone, with no account being taken of how spatially adjacent pixels are classified. In the following, we summarize the key concepts for spectral-spatial classification of hyperspectral data.

### 2.3.2 Spectral-Spatial Classification

It is a proven fact that for images with high-spatial resolution, combining the spectral and the spatial information improves significantly the performance of classification methods. Surveys of spectral-spatial classification methods can be found in [13, 41].

A common spectral-spatial approach is to incorporate spatial information as part of the pixelwise classification process. Some feature extraction is applied to the surrounding area of a pixel and the result is integrated as part of the features associated to the individual pixel, in addition to the usual spectral features. In order to perform classification with a kernel method such as an SVM, the two sets of features must be combined. This can be done in different ways, ranging from a naive stacking of the feature vectors to more versatile methods. Different strategies of combining the two sources of information have been reviewed and compared in [23, 88].

To apply this scheme, one must define which is the neighborhood of a pixel from where spatial features are extracted. An idea as simple as the use of a fixed window already shows an improvement with respect to purely pixelwise approaches [88]. Benediktsson et al. [8] proposed to use morphological filters to obtain the spatial neighborhoods in an adaptive manner. In this method, a so-called structuring element is used to perform morphological opening and closing operations [109]. The effect of applying these operations is that image structures smaller than the structuring element are removed, otherwise preserved. These operations are applied with structuring elements of different sizes to create the *morphological profile*. This idea was applied in hyperspectral image classification [5] by computing the morphological profiles of the first principal components of the data, and combining them to obtain the features for classification.

Later on, Fauvel et al. [40] proposed to use the so-called self-complementary filters [110] for spatial feature extraction, which remove small structures from the image based on an area criterion, yielding a map of flat connected zones. This filter is applied on the first principal component of the hyperspectral image to extract adaptive spatial neighborhoods. The vector median [4] is then computed for each connected zone of the filtered result, and used as the spatial feature vector for all the pixels within the zone. Finally, SVM classification is performed with a weighted summation kernel to combine spectral and spatial information. More advanced morphological filters, called attribute filters, have been recently proposed to further enhance classification performance [3, 74].

Another important approach to characterize pixel entities using the spatial and the spectral information is the Markov random field (MRF) [42, 61]. MRFs (see also Chaps. 4 and 7) are probabilistic models widely used to include spatial context into image analysis schemes in terms of minimization of suitable energy functions [83]. The MRF energy function for image classification is commonly computed as a linear combination of a data term, which measures for each pixel the disagreement between a prior probabilistic model and the observed data, and a spatial context term, which expresses interaction between neighboring pixels. The first MRF-based models employed time-consuming energy minimization algorithms, such as iterated conditional modes and simulated annealing [82, 117]. More advanced methods, such as graph-cuts [19, 20] provided powerful alternatives from both theoretical and computational viewpoints, resulting in a growing use of the MRF-based models [72, 118]. For example, Tarabalka et al. [118] used probabilities derived from an SVM as the data term of an MRF energy, and used the  $\alpha$ -expansion graph cut algorithm [20] to solve the  $K$ -class classification problem in hyperspectral imagery.

Finally, an important family of methods involves the segmentation of images and the classification of each of the individual segments. Segmentation methods partition an image into non-overlapping homogeneous regions with respect to some criterion of interest or homogeneity criterion (e.g., based on the intensity or on the texture) [46]. Hence, each region in the segmentation map can be seen as a connected spatial neighborhood for all the pixels within this region. One of the pioneering spatial-spectral techniques belongs to this category: the well-known ECHO (Extraction and Classification of Homogeneous Objects) classifier [65], which has been extensively used by

the remote sensing community. It is based on region growing to find homogeneous groups of adjacent pixels, which are then classified as single objects by a Gaussian maximum likelihood method. Since then, different techniques have been proposed for hyperspectral image segmentation, such as watershed, partitional clustering and Hierarchical Segmentation (HSeg) [112, 113, 116]. From a segmentation map, an SVM classifier and majority voting can be applied to combine spectral and spatial information: for every region in the segmentation map, all the pixels are assigned to the most frequent class within this region, based on SVM classification results [113]. This method yields an improvement of classification accuracies when compared to spectral-spatial techniques using local spatial neighborhoods.

It is however a challenging task to perform hyperspectral image segmentation automatically. The performance is highly dependent both on the measure of region homogeneity and on the algorithm parameters. Several alternatives have been proposed to deal with this challenge. Tarabalka et al. [114, 115] proposed to perform a marker-controlled segmentation for this purpose. The classification probabilities are used to automatically select the most reliably classified pixels (i.e., pixels belonging with the high probability to the assigned class). The classification map is then obtained by building a minimum spanning forest from the image graph rooted on the selected markers. Another alternative for automatic segmentation consists in building first a hierarchy of segmentations at different levels of details, and then selecting from this hierarchy the regions at different scales that correspond to the objects of interest. Valero et al. proposed to use a binary partition tree (BPT) model for this purpose [122]. In this method, a BPT is first constructed by iteratively clustering similar regions based on a criterion specifically designed for hyperspectral images. Each BPT node is then modeled by its mean spectrum and classified by using an SVM. A so-called misclassification rate is computed for each node, which can be understood as the error incurred by assigning the entire node to the wrong class. A spectral-spatial classification map is finally built in a bottom-up traversal of the tree by extracting regions with a low misclassification rate. In the next section we describe an energy minimization BPT-based model recently proposed in [76].

### ***2.3.3 Object-Based Classification with Binary Partition Trees***

The goal of classification is to convert the image data into tangible information that can be interpreted and incorporated into other systems. The ultimate elementary units which we want to identify are the *objects* present in the image. In the earlier years of remote sensing research, the per-pixel or sub-pixel analysis were particularly relevant given that pixel sizes were coarser than the objects themselves. The boundary between pixel-based and object-based analysis was still vague. As sensors improved their spatial resolution, objects started to be comprised of many pixels and object-based analysis emerged as a natural consequence of this. While pixelwise and spectral-spatial classification may constitute the first of a series of steps in the image analysis pipeline, object-based methods aim at delineating readily usable objects

from the image [15]. Contrary to the well-established pixelwise and spectral-spatial approaches described before, this section presents a recent object-based classification model for hyperspectral imagery, based on binary partition trees.

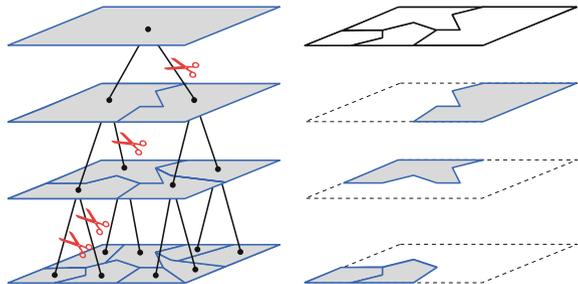
Binary partition trees (BPTs) were presented by Salembier and Garrido [103] as a way of representing a set of meaningful image regions in a compact and structured manner. The root node corresponds to the entire image, the following level represents the subdivision of the entire image into two disjoint regions, and so on. It constitutes then a hierarchical abstraction of an image, which can be navigated to extract meaningful regions at different scales. The typical workflow involves an initial tree construction stage, followed by a second stage of information extraction from the tree. For example, once a tree is constructed, an exhaustive segmentation of the image can be obtained by performing a horizontal “cut” on the structure (see Fig. 2.17). In this procedure, commonly referred to as *pruning*, branches can be selected at different scales, an inherent advantage of such hierarchical structure.

The construction of a BPT is done in a bottom-up fashion, by iteratively clustering pairs of similar regions together. The starting point is an initial subdivision of the image represented by a region adjacency graph (RAG), where every node conveys a region and the edges link spatial neighbors (i.e., candidates for merging). The typical initial RAG is the pixel grid, though nothing prevents the approach to be used with other inputs too (e.g., a RAG of small regions containing similar pixels, known as superpixel segmentation). Every edge in the RAG is labeled with a *dissimilarity* value that compares the two associated regions.

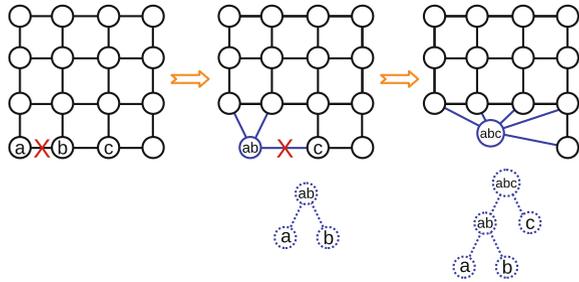
BPTs are constructed by following a global mutual best fitting region merging approach [69]: at each iteration, the two most similar regions in the current subdivision are merged together (i.e., the least weighted edge out of all edges in the RAG). When a merge occurs, a new region is added to the BPT, connected to its two corresponding children, as illustrated in Fig. 2.18). The process finishes when there are no more edges left in the RAG. A BPT constitutes then a record of the history of merges that occurred during the execution of a region merging algorithm.

The overall process can be implemented efficiently by using an updatable priority queue structure on top of the RAG edges to keep track of the highest priority element. Such a structure is first constructed in linear time and every subsequent update incurs in a logarithmic time cost. When two regions  $R_1$  and  $R_2$  are merged into a new

**Fig. 2.17** A binary partition tree (BPT) is a hierarchical subdivision of an image. An exhaustive partitioning can be extracted by “cutting” branches at different scales



**Fig. 2.18** A BPT is constructed by iteratively removing edges in a region adjacency graph (RAG). The resulting BPT encodes the history of the merges



region  $R_{12}$ , one must update the RAG (and the associated priority queue). The edge connecting  $R_1$  and  $R_2$  must be removed, but let us also remark that all the edges adjacent to  $R_1$  and  $R_2$  must also be eliminated from the RAG, since none of both regions exists anymore. We must then add the adjacency relations of the new region  $R_{12}$ . The computation is straightforward: the neighbors of the new region  $R_{12}$  are nothing but the union of the neighbors of the old  $R_1$  and  $R_2$  (with spatial care to remove any duplicates that may arise). The dissimilarity value associated to each of these edges must be computed and pushed to the priority queue. The complexity of the overall BPT construction process is  $O(n \log(n)M)$ ,  $n$  being the initial number of nodes and  $M$  the maximum number of neighbors of a merged region during the construction. Given that typically  $M \ll n$ , the algorithm is quasilinear in practice.

The final tree contains exactly  $2n - 1$  nodes, which is a very space-efficient representation. Let us remark though that only a subset of all possible planar subdivisions is represented by the tree, hence the research efforts to construct a good initial tree that conveys meaningful objects of the underlying image.

The key elements to define the behavior of a BPT are the *region model*, i.e. how regions are represented, and the *dissimilarity function*, i.e., the function to compare the region models, used to define the priority of the merges during tree construction. The next paragraphs review the contributions related to these two elements.

### Region Model

The object-based nature of BPTs allows to have rich representations of the region that go beyond pixel spectra. Every BPT node can convey *regional* information, describing the region as a whole and not as a set of individual pixels. The standard variation of the spectral signatures in the region or shape features such as compactness are some of the regional data that can be associated to every node.

To represent the spectrum of a region (and then compare it to the spectra of other regions) there are essentially two alternatives: parametric and non-parametric models. A parametric model makes assumptions about the homogeneity or Gaussian distribution inside the regions. A typical parametric model is to represent the spectrum of a region as the mean spectrum of its pixels. Non-parametric models, on the contrary, consist of per-band histograms of the pixel values, hence they represent the real observed distributions. In hyperspectral imagery, non-parametric models have a better performance since they can describe the internal variability of a region [122].

For example, a texture might correspond to several peaks in the histogram. When averaging spectra in regions with high variability, one might end up representing the region with a “false” spectrum that is not present in any of the individual pixels.

In addition to spectral data, the model usually stores the *area* of the region, since it is commonly used in the dissimilarity function. Other shape descriptors such as *solidity*, *rectangularity index*, *elongatedness* and *compactness* can also be efficiently stored and computed from the children nodes [77].

### Dissimilarity Function

To establish a priority for merging during BPT construction, it is required to provide a means to compare models of two regions. A dissimilarity function  $O(R_1, R_2)$  typically used for this purpose comprises two factors as follows:

$$O(R_1, R_2) = \min(|R_1|, |R_2|)^\beta D(R_1, R_2), \quad (2.5)$$

where  $|R_i|$  denotes the area of region  $R_i$ . The first part of (2.5),  $\min(|R_1|, |R_2|)^\beta$ , is the so-called *area-weighting* factor. This is an agglomerative force intended to cluster regions that are very small compared to the rest of the elements in the RAG. When no area-weighting is used (i.e.,  $\beta = 0$ ), the resulting BPT might isolate small noisy areas and connect them to the rest only near the root of the tree. With moderate values of  $\beta$ , small regions are merged at some point, forcing the trees to better look like a hierarchical subdivision. When  $\beta$  is too large, the trees might be too biased to be balanced, hampering their representation capabilities. Even though this parameter is barely discussed in the literature, being mostly set to  $\beta = 0.5$  or  $\beta = 1$ , we must point out that it is indeed a parameter that has to be selected. In our experience, no area-weighting leads to poor representations (e.g., the root containing two children: one noisy pixel and all the rest of the image), while low values of  $\beta$  solve this issue without biasing the trees too much. Alternatively, Calderero and Marques [21] proposed to keep track of the out-of-scale regions and force their merging at some point, while Valero et al. [122] used a weighted sum of pixel values in a window to initialize the histograms, as a way of smoothing out outliers.

The second factor,  $D(R_1, R_2)$ , compares both regions based on their spectra. Kullback-Leiber divergence and Bhattacharyya distance are popular choices both in hyperspectral imagery and other types of images [21, 122]. Spectra are seen as probability distributions and compared using standard information theory concepts. Every bin of one histogram is compared against the corresponding bin of the other histogram. However, using cross-bin measures, which go beyond individual bins, has proven to be more robust [122]. The average of Earth Mover’s Distances [101] among histograms of all bands can be used as a robust and efficient cross-bin dissimilarity function. Every distribution is seen as a pile of dirt, and the difference between two distributions is seen as the amount of work required to turn one pile into the other one.

### 2.3.3.1 Multi-class Segmentation with BPTs

The problem of object-based classification can be seen as the simultaneous segmentation of an image and the assignment of a label to every segment. This section first formulates this problem as the minimization of an energy, and describes an algorithm to extract the optimal segmentation with respect to that energy from a BPT. This algorithm outputs the lowest-energy solution from all the segmentations represented by the BPT, which are a subset of all possible image partitions. It is then a matter of high importance to construct good BPTs whose solution space contains relevant candidates for object-based analysis. For this we describe a supervised BPT construction technique that incorporates class probabilities to cluster objects together.

#### Multi-class Segmentation as Energy Minimization

Let  $\mathbf{X} = \{\mathbf{x}_i \in \mathbb{R}^B, i = 1, 2, \dots, n\}$  be a  $B$ -band image seen as a set of  $n$  pixel vectors. Multi-class segmentation consists in an exhaustive partitioning of the pixels into a non-overlapping set of regions  $\mathbf{R} = (R_j)$ , with associated labels  $\mathbf{L} = (L_j)$ , where every label  $L_j$  belongs to the set  $\Omega$  of available information classes. From each object class, we suppose we are given training examples from which we can derive posterior probabilities  $P(L_j|\mathbf{x}_i)$  of assigning a certain label  $L_j$  after the spectral observation  $\mathbf{x}_i$  is taken into account. Such posterior probability may be derived from a support vector machine [128]. The negative log-likelihood  $-\log P(L_j|\mathbf{x}_i)$  is typically used to express a *cost* that penalizes the assignment of label  $L_j$  to pixel  $\mathbf{x}_i$ .

Our task is to find the labeled partitioning  $(\mathbf{R}, \mathbf{L})$  from a BPT that minimizes the following energy:

$$E(\mathbf{R}, \mathbf{L}) = \lambda|\mathbf{R}| - \sum_{R_j \in \mathbf{R}} \sum_{\mathbf{x}_i \in R_j} \log P(L_j|\mathbf{x}_i). \quad (2.6)$$

Let us first observe that the same label  $L_j$  is assigned to all pixels  $\mathbf{x}_i$  in region  $R_j$ , since the entire segments take a single label. The first term is a regularizer on the number of regions in the partition  $|\mathbf{R}|$ , and controls the coarseness of the output through parameter  $\lambda$ . In the absence of this term (i.e.,  $\lambda = 0$ ), the optimal solution is to create one segment per pixel and assign to it the lowest-cost label. To introduce the notion of object we must then set  $\lambda > 0$ . We here set this parameter manually, but let us mention that in recent work the regularization term was directly learned from training samples [77].

From a BPT, the best possible labeled segmentation with respect to Eq. 2.6 can be extracted efficiently [102]. This task can be interpreted as the extraction of a minimal horizontal s-t *cut* on the tree (see Fig. 2.17), i.e., with a source at every leaf and a sink at the root. Let us denote  $C(R)$  the energy of the cut on  $R$  with minimal (2.6) among all possible cuts.

Considering that the branches in the tree are independent, the globally optimal cut can be found by a dynamic programming algorithm. Let us denote  $\mathcal{E}(R) = \min_{L \in \Omega} E(\{R\}, \{L\})$  the lowest possible energy of a region  $R$  (by assigning the label

that incurs the lowest cost). The tree is traversed in a bottom-up manner. Whenever a region  $R$  is visited, the following property is evaluated:

$$\mathcal{E}(R) \leq C(R_{left}) + C(R_{right}), \quad (2.7)$$

where  $R_{left}$  and  $R_{right}$  are the children of  $R$ . If the property does not stand, we set  $C(R) = C(R_{left}) + C(R_{right})$  and keep the best cuts of both children. Otherwise, we set  $C(R) = \mathcal{E}(R)$  and replace the cuts by  $R$  with label  $L$ . This process is executed recursively until reaching the root of the tree. The overall algorithm is linear in the image size, since only one BPT traversal is required, and guarantees the optimal cut in the space of solutions represented by the BPT.

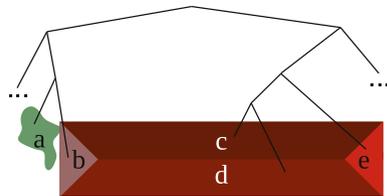
### Supervised BPT Construction

Even though the globally optimal cut on a BPT can be found efficiently, not all the possible ways of segmenting an image are represented in the structure. In some images, objects have considerable internal variability. For example, it is known that the different parts of a roof often contrast more with each other than with other surrounding objects [45]. This is more prevalent in high-resolution imagery and in cluttered urban scenes.

In such images, it is common to observe objects that are split into different branches of the tree instead of being contained in a single node. This behavior is illustrated in Fig. 2.19, which shows a BPT built on an image of a non-uniform roof. During BPT construction, a part of the roof (b) is merged first to something else (a) than to the rest of the object (c-d-e) because it is more similar in terms of Eq. (2.5). As a consequence, the entire building (b-c-d-e) cannot be extracted by selecting a single node in the tree.

Figure 2.20 illustrates this phenomenon on real image data. Two fragments of the *Pavia Center* image, which will be introduced in the experimental section, are shown in Fig. 2.20a. The scene contains multiple buildings, streets and cars adjacent to each other. A BPT with a non-parametric region model and Earth Mover’s Distance was constructed for this image. The energy minimization scheme (2.6) was then applied, and the objects labeled as *tile* isolated from the rest to aid the interpretation. Figure 2.20b depicts the surface covered by *tiles*, as predicted by the BPT cut. This way of illustrating the classification is purely pixelwise, since no distinction about the *objects* extracted from the tree is made. This is a common way of illustrating results in the literature, even when the goal is to perform object detection (e.g., [123]).

**Fig. 2.19** Faulty BPT: the object ( $bcde$ ) is not represented in a single node, since a part of it ( $b$ ) merged first to something else



However, observing the actual objects extracted from the BPT (Fig. 2.20c) we can see that the regions hardly correspond to actual objects in the image. Even though the surface covered by these objects might be satisfactory from a pixelwise perspective, an object-based analysis would certainly be less impressive.

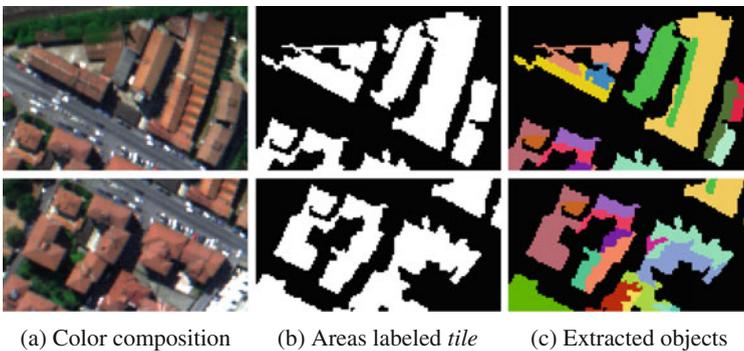
Let us recall that the use of non-parametric region models is to represent internal variability. However, commonly used dissimilarity functions such as (2.5) penalize the merging of dissimilar regions. In an unsupervised context, where there is no notion of object class, there is little to do to deal with this, since there is no reason to cluster dissimilar regions together. However, when class probabilities are available we propose to include an additional force that clusters regions belonging to the same class, despite being spectrally dissimilar. The new function is as follows:

$$O(R_1, R_2) = \min(|R_1|, |R_2|)^\beta \left[ (1 - \alpha)D(R_1, R_2) - \alpha \log P(L_{R_1} = L_{R_2}) \right]. \quad (2.8)$$

As in the original dissimilarity function (2.5), there is an area-weighting factor and an unsupervised term  $D(R_1, R_2)$ , which is computed by comparing spectral histograms of regions without any preliminary training. Equation 2.8 adds a *supervised* term  $P(L_{R_1} = L_{R_2} | R_1, R_2)$ , the probability of assigning the same label to both regions. This way, while the unsupervised term penalizes spectral dissimilarity, the supervised term will encourage merging regions that are likely to belong to the same class. The trade-off between both terms is controlled by parameter  $\alpha$ .

The term  $P(L_{R_1} = L_{R_2} | R_1, R_2)$  is computed by marginalizing over the classes as follows:

$$P(L_{R_1} = L_{R_2} | R_1, R_2) = \sum_{j=1}^K P(L_j | R_1) P(L_j | R_2), \quad (2.9)$$



**Fig. 2.20** Analyzing classification from a pixelwise **b** vs object-based **c** perspective. Even though the area covered by the *tile* objects might be satisfactory (**b**), the objects that constitute this area often do not correspond to the real objects (**c**)

where  $K$  is the number of classes and  $P(L_j|R_k)$ , with  $k \in \{1, 2\}$ , represents the probability of assigning a certain label  $L_j$  to segment  $R_k$ . We must now define a way to compute  $P(L_j|R_k)$  based on the posteriors of the individual pixels contained in the region. One way to do this is to compute the probability of assigning the label to all pixels, conditioned by the fact that all labels are known to be equal inside the region:

$$P(L_j|R_k) = \prod_{\mathbf{x}_i \in R_k} P(L_j|\mathbf{x}_i) / \left[ \sum_{\omega_m \in \Omega} \prod_{\mathbf{x}_i \in R_k} P(\omega_m|\mathbf{x}_i) \right]. \quad (2.10)$$

Alternatively, one can estimate  $P(L_j|R_k)$  by averaging the individual pixel probabilities:

$$P(L_j|R_k) = \frac{1}{|R_k|} \sum_{\mathbf{x}_i \in R_k} P(L_j|\mathbf{x}_i). \quad (2.11)$$

While the first expression is closer to a strict Bayesian interpretation, we found the second one to be a simple yet useful approximation.

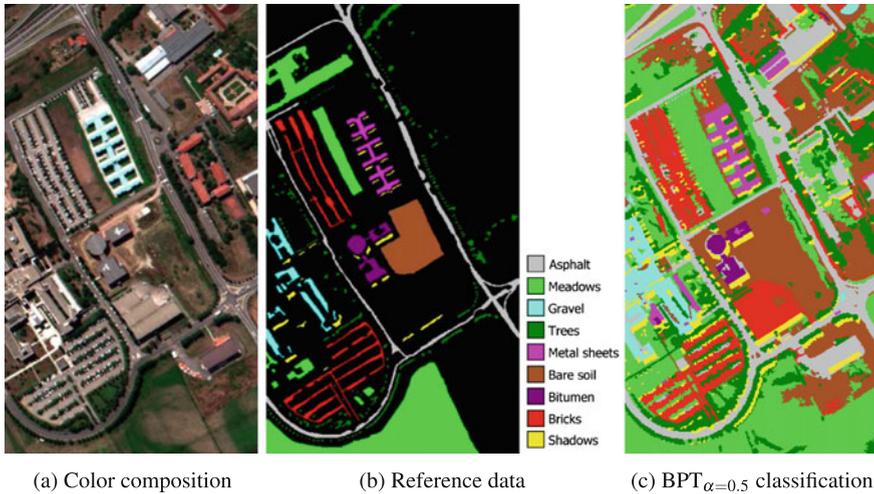
By introducing (2.8) we expect to better cluster semantically significant objects together. The advantage of such an outcome is two-fold: first of all, the classification accuracy is improved. Secondly, there is a notion of object, which constitutes a higher-level interpretation of the input image rather than mere pixelwise labeling.

### 2.3.4 Experimental Results

This section describes two series of experiments to analyze and compare different methods of hyperspectral image classification. We report results for the most representative pixelwise and spectral-spatial methods discussed in the previous sections, as well as the BPT model. The first set of experiments is performed on a dataset over the *University of Pavia*, Italy. The goal of this evaluation is to compare the different approaches in terms of per-pixel classification accuracy, with the particular goal of verifying that the introduction of spatial information improves the results.

A second set of experiments is carried out on the *Pavia center* hyperspectral dataset. The goal of these experiments is to evaluate the behavior of the techniques from an object-based perspective, providing an object overlap measure between reference and detected objects. We compare the typical unsupervised BPT construction approach and the supervised alternative introduced in Sect. 2.3.3.1.

Both images were acquired with the Reflective Optics System Imaging Spectrometer (ROSIS-03). This optical sensor provides 115 bands with a spectral coverage ranging from 0.43 to 0.86  $\mu\text{m}$  and 1.3 m spatial resolution.



**Fig. 2.21** University of Pavia hyperspectral dataset

## University of Pavia

This image is of size  $610 \times 340$  and contains 103 spectral channels (after excluding 12 noisy bands). Figure 2.21a illustrates a false color composition of the hyperspectral image. The reference data contains nine classes of interest, as depicted in Fig. 2.21b.

In the next paragraphs we summarize the classification methods that were executed and compared on this dataset. As described in Sects. 2.3.1–2.3.3, these techniques have become the standard in the remote sensing literature.

**SVM** A support vector machine (SVM) was trained on 50 randomly selected samples for every class. A multi-class one vs one SVM with Gaussian kernel was used (with parameters  $C = 128$  and  $\gamma = 0.125$ , set by fivefold cross-validation).

**Graph cut** A graph cut with  $\alpha$ -expansion [20] (which proved to be effective in hyperspectral image classification [118]) was executed on probabilities derived the SVM. Its regularity parameter was set empirically to optimize the accuracy.

**HSeg** The technique presented in [113] was also implemented, which consists in first performing a segmentation and then labeling every segment. A recursive hierarchical image segmentation (HSeg) is used, followed by a majority voting procedure in which every segment is labeled as the majority class of the SVM predictions inside the segment. Parameter ‘spclust\_wght’ was set to 0.1, following the original publication.

**BPT** For the binary partition tree (BPT) model described in this chapter, the tree was constructed by using a non-parametric model with 30 histogram bins per band, the Earth Mover’s Distance to compare histograms and mild area-weighting ( $\beta = 0.1$ ). The coarseness parameter  $\lambda$  in (2.6) was empirically set to 40. Two variants were tested: (a) totally unsupervised construction, i.e., setting  $\alpha = 0$  in (2.8), which is equivalent to the old function (2.5); (b) supervised construction with equal contribution from both terms in (2.8), i.e.,  $\alpha = 0.5$ .

**Table 2.3** Numerical evaluation on *University of Pavia* dataset (in %)

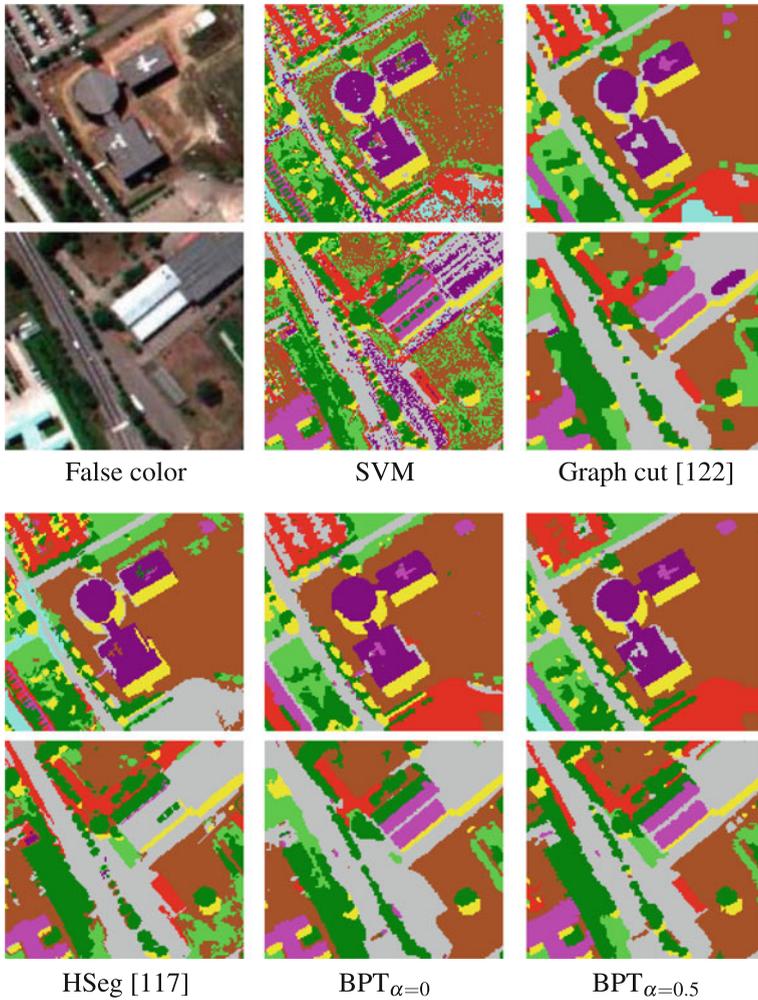
	SVM	Graph cut	HSeg [113]	BPT $_{\alpha=0}$	BPT $_{\alpha=0.5}$
AA	88.03	95.13	95.35	95.49	<b>97.32</b>
OA	80.38	91.69	90.75	<b>94.45</b>	93.13
Asphalt	77.66	94.58	95.40	97.83	<b>99.15</b>
Meadows	72.74	86.10	83.63	<b>92.65</b>	86.07
Gravel	79.55	86.58	98.98	84.87	<b>99.17</b>
Trees	95.95	<b>97.38</b>	96.28	91.44	96.72
Metal	99.61	<b>100.0</b>	99.15	99.07	99.92
Bare soil	89.38	<b>98.39</b>	94.86	97.99	98.31
Bitumen	94.37	95.55	95.23	<b>99.92</b>	97.19
Bricks	82.89	97.74	97.88	95.59	<b>99.31</b>
Shadows	<b>100.0</b>	99.89	96.77	<b>100.0</b>	<b>100.0</b>

The test dataset was created by excluding the pixel used from SVM from the ground truth. To measure the performance we use the average accuracy (AA) and overall accuracy (OA). The first one computes for every class the percentage of correctly classified pixels from the test data, and averages these values over all the classes. The latter is the proportion of correctly classified pixels. The pixels used for SVM training are excluded in the evaluation. The numerical results are deployed on Table 2.3. The accuracies for individual classes are also included in the table. We can verify that purely pixelwise methods such as SVM have a lower performance than spectral-spatial approaches. The BPT models (with  $\alpha = 0$  and  $\alpha = 0.5$ ) outperform the other techniques. The inclusion of class probabilities in tree construction ( $\alpha = 0.5$ ) boosts the AA with a mild decrease of OA with respect to the unsupervised construction.

The overall classification map for the  $BPT_{\alpha=0.5}$  method is shown in Fig. 2.21c and two fragments are amplified and compared with other methods in Fig. 2.22. These results show that in general BPTs constitute an improvement with respect to the other techniques. The benefit of supervised ( $\alpha = 0.5$ ) over unsupervised ( $\alpha = 0$ ) construction is not entirely clear in this dataset. First of all, there are few objects of every class in the reference data and the labeled pixels do not cover the entire surface of the objects. Moreover, there seems to be a significant contrast between objects and their surroundings, a situation in which the supervised term in (2.8) may not be very relevant. While BPTs have proved to be competitive from a pixelwise perspective, we require a different dataset to evaluate the performance of the methods from an object-based perspective and compare the unsupervised vs supervised construction models.

### Pavia Center

This image has spatial dimensions  $400 \times 300$  and contains 102 bands. A color composition of the image is shown in Fig. 2.23a. Compared to the *University* dataset, this



**Fig. 2.22** *University of Pavia*: closeups of classification maps with different methods

image presents a more cluttered scene with objects composed of dissimilar parts. As illustrated previously in Fig. 2.20, BPT nodes may not correspond to entire objects because parts of them grow into other adjacent objects during the construction.

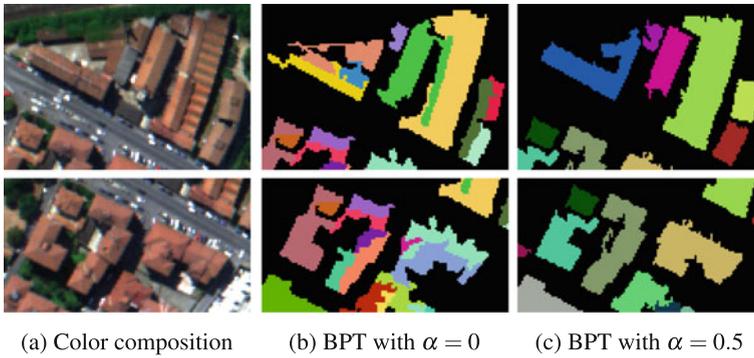
A reference image that labels entire objects and not just isolated pixels was built, including four classes (see Fig. 2.23b). This reference was constructed by combining the labeling of isolated pixels provided with the original image, visual inspection and official Italian records of building boundaries, which are available for this area through the OpenStreetMap.org database. Since the boundaries of buildings are well defined, there is a particular interest in analyzing the performance of BPTs to extract buildings.



**Fig. 2.23** Experiments on *Pavia Center* hyperspectral image

An SVM is first trained on randomly selected samples (parameters  $C = 128$ ,  $\gamma = 2^{-5}$ ). The SVM classification is shown in Fig. 2.23c. A BPT is then constructed on top of the SVM probabilities, in a similar experimental setting as with the *University of Pavia* dataset, and the classification map is extracted by setting  $\lambda = 20$  in Eq. (2.6). The resulting classification map with supervised tree construction ( $\alpha = 0.5$ ) is shown in Fig. 2.23d.

Figure 2.23e, f and the close-ups of Fig. 2.24 compare the results obtained by applying the unsupervised and supervised approaches for BPT construction. These figures isolate the *tile* objects from the rest and assign a random color to every individual object. From these illustrations we can appreciate that including class probabilities during BPT construction has the effect of better clustering the objects together. To validate this numerically we compute the overlap between every building (belonging either to *tiles* or *bitumen* classes) in the reference data and the most



**Fig. 2.24** Unsupervised **b** versus supervised **c** BPT construction. In the supervised case, regions are better clustered together to represent significant objects

**Table 2.4** Numerical evaluation on the *Pavia Center* dataset

	SVM	Graph cut	$BPT_{\alpha=0}$	$BPT_{\alpha=0.5}$
Building overlap	0.51	0.51	0.54	<b>0.56</b>
Overall accuracy	0.88	<b>0.94</b>	0.91	<b>0.94</b>

overlapping building region in the BPT output. The overlap is measured with Dice's coefficient defined as:  $2|R_1 \cap R_2|/(|R_1| + |R_2|)$ . The resulting overlap coefficients are averaged over all reference buildings to produce an estimation of how well the BPT output matches the reference data from an object-based perspective. The numerical results, together with the overall accuracy, are summarized in Table 2.4, which also includes the values for SVM and graph cut. A first observation we can make is that  $BPT_{\alpha=0.5}$  performs better than  $BPT_{\alpha=0}$ , corroborating the visual impression from Fig. 2.23e, f. Secondly, while graph cut is known to improve the SVM classification, we can see that this is true from a pixelwise perspective (in terms of OA) but not from an object-based perspective (in terms of building overlap). Finally, the use of  $BPT_{\alpha=0.5}$  outperforms the other methods in terms of object overlap. This validates the idea of including class probabilities during tree construction for a better object-based analysis of hyperspectral imagery.

## 2.4 Challenges

The classification of hyperspectral imagery presents a number of challenges proper to the nature of this image modality. The integration of spatial and spectral information is one of the most widely addressed issues, as we have reviewed throughout this chapter. This concern will certainly continue to intrigue the scientific community and will remain an active research area. However, the imbalance between the high

dimensionality of hyperspectral data and the low amount of training samples is still probably one of the largest sources of difficulty.

There is a growing trend to study new training schemes to deal with the limited availability of labeled data. Notably, *semi-supervised* algorithms are arising in the hyperspectral literature (e.g., [97, 119]). These algorithms combine a low amount of labeled training data with *unlabeled* samples, under the assumption that the latter can be obtained with little effort. A smart combination of labeled and unlabeled data may significantly improve the accuracy of classification. Among semi-supervised algorithms, *active learning* methods interact with the user to actively query for helpful labels [86, 120, 125].

With the recent advent of *deep learning* in multiple application domains, it will certainly gain increasing attention in the hyperspectral image analysis community. Some first research efforts in this direction can be already identified in the literature [34, 75].

To conclude, we can say that hyperspectral remote sensing image analysis uses and adapts frontier concepts, frameworks and algorithms from the fields of signal and image processing, statistical inference and machine learning. The compendium of techniques presented in this chapter reflects the increasing sophistication of a field that is rapidly maturing at the intersection of many different disciplines.

**Acknowledgements** This work has been supported by:

- Junta de Extremadura (decreto 297/2014, ayudas para la realización de actividades de investigación y desarrollo tecnológico, de divulgación y de transferencia de conocimiento por los Grupos de Investigación de Extremadura, Ref. GR15005);
- Centre national d'études spatiales (CNES, France).

## References

1. Adams, J.B., Smith, M.O., Johnson, P.E.: Spectral mixture modeling: a new analysis of rock and soil types at the Viking lander 1 site. *J. Geophys. Res.* **91**, 8098–8112 (1986)
2. Ambikapathi, A., Chan, T.-H., Ma, W.-K., Chi, C.-Y.: Chance-constrained robust minimum-volume enclosing simplex algorithm for hyperspectral unmixing. *IEEE Trans. Geosci. Remote Sens.* **49**(11), 4194–4209 (2011)
3. Aptoula, E., Dalla Mura, M., Lefèvre, S.: Vector attribute profiles for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **54**(6), 3208–3220 (2016)
4. Astola, J., Haavisto, P., Neuvo, Y.: Vector median filters. *Proc. IEEE* **78**(4), 678–689 (1990)
5. Benediktsson, J.A., Palmason, J.A., Sveinsson, J.R.: Classification of hyperspectral data from urban areas based on extended morphological profiles. *IEEE Trans. Geosci. Remote Sens.* **43**(3), 480–491 (2005)
6. Benediktsson, J.A., Swain, P.H.: *Statistical Methods and Neural Network Approaches for Classification of Data from Multiple Sources*. Ph.D. thesis, Purdue Univ., School of Elect. Eng., West Lafayette, IN (1990)
7. Benediktsson, J.A., Swain, P.H., Ersoy, O.K.: Conjugate gradient neural networks in classification of very high dimensional remote sensing data. *Int. J. Remote Sens.* **14**(15), 2883–2903 (1993)

8. Benediktsson, J.A., Pesaresi, M., Arnason, K.: Classification and feature extraction for remote sensing images from urban areas based on morphological transformations. *IEEE Trans. Geosci. Remote Sens.* **41**(9), 1940–1949 (2003)
9. Berman, M., Kiiveri, H., Lagerstrom, R., Ernst, A., Dunne, R., Huntington, J.F.: ICE: a statistical approach to identifying endmembers in hyperspectral images. *IEEE Trans. Geosci. Remote Sens.* **42**(10), 2085–2095 (2004)
10. Bioucas-Dias, J.: A variable splitting augmented lagrangian approach to linear spectral unmixing. In: *IEEE Whispers*, pp. 1–4 (2009)
11. Bioucas-Dias, J.M., Nascimento, J.M.P.: Hyperspectral subspace identification. *IEEE Trans. Geosci. Remote Sens.* **46**(8), 2435–2445 (2008)
12. Bioucas-Dias, J.M., Plaza, A.: Hyperspectral unmixing: geometrical, statistical, and sparse regression-based approaches. In: *Proceedings of the SPIE Image and Signal Process, Remote Sens*, XVI, vol. 7830, pp. 1–15 (2010)
13. Bioucas-Dias, J.M., Plaza, A., Camps-Valls, G., Scheunders, P., Nasrabadi, N.M., Chanussot, J.: Hyperspectral remote sensing data analysis and future challenges. *IEEE Geosci. Remote Sens. Mag.* **1**(2), 6–36 (2013)
14. Bioucas-Dias, J.M., Plaza, A., Dobigeon, N., Parente, M., Du, Q., Gader, P., Chanussot, J.: Hyperspectral unmixing overview: geometrical, statistical and sparse regression-based approaches. *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.* **5**(2), 354–379 (2012)
15. Blaschke, T.: Object based image analysis for remote sensing. *ISPRS J. Photogr. Remote Sens.* **65**(1), 2–16 (2010)
16. Boardman, J.W., Kruse, F.A., Green, R.O.: Mapping target signatures via partial unmixing of Aviris data. *Proc. JPL Airborne Earth Sci. Workshop* **95–7**, 23–26 (1995)
17. Boggs, J.L., Tsegaye, T.D., Coleman, T.L., Reddy, K.C., Fahsi, A.: Relationship between hyperspectral reflectance, soil nitrate-nitrogen, cotton leaf chlorophyll, and cotton yield: a step toward precision agriculture. *J. Sustain. Agric.* **22**(3), 5–16 (2003)
18. Borel, C.C., Gerslt, S.A.W.: Nonlinear spectral mixing models for vegetative and soil surfaces. *Remote Sens. Environ.* **47**, 403–416 (1994)
19. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(9), 1124–1137 (2004)
20. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(11), 1222–1239 (2001)
21. Calderero, F., Marques, F.: Region merging techniques using information theory statistical measures. *IEEE Trans. Image Process.* **19**(6), 1567–1586 (2010)
22. Camps-Valls, G., Bruzzone, L.: Kernel-based methods for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **43**(6), 1351–1362 (2005)
23. Camps-Valls, G., Gomez-Chova, L., Muñoz-Marí, J., Vila-Francés, J., Calpe-Maravilla, J.: Composite kernels for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **3**(1), 93–97 (2006)
24. Ceamanos, X., Waske, B., Benediktsson, J.A., Chanussot, J., Fauvel, M., Sveinsson, J.R.: A classifier ensemble based on fusion of support vector machines for classifying hyperspectral data. *Intern. J. Image Data Fusion* **1**(4), 293–307 (2010)
25. Chan, T.-H., Chi, C.-Y., Huang, Y.-M., Ma, W.-K.: A convex analysis based minimum-volume enclosing simplex algorithm for hyperspectral unmixing. *IEEE Trans. Signal Process.* **57**(11), 4418–4432 (2009)
26. Chan, T.-H., Ma, W.-K., Ambikapathi, A., Chi, C.-Y.: A simplex volume maximization framework for hyperspectral endmember extraction. *IEEE Trans. Geosci. Remote Sens.* **49**(11), 4177–4193 (2011)
27. Chang, C., Zhao, X., Althouse, M.L.G., Pan, J.J.: Least squares subspace projection approach to mixed pixel classification for hyperspectral images. *IEEE Trans. Geosci. Remote Sens.* **36**(3), 898–912 (1998)
28. Chang, C.-I.: *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*. Kluwer Academic/Plenum Publishers, New York (2003)

29. Chang, C.-I.: *Hyperspectral Data Exploitation: Theory and Applications*. Wiley, New York (2007)
30. Chang, C.-I., Du, Q.: Estimation of number of spectrally distinct signal sources in hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.* **42**(3), 608–619 (2004)
31. Chang, C.-I., Heinz, D.: Constrained subpixel target detection for remotely sensed imagery. *IEEE Trans. Geosci. Remote Sens.* **38**, 1144–1159 (2000)
32. Chang, C.-I., Plaza, A.: A fast iterative algorithm for implementation of pixel purity index. *IEEE Geosci. Remote Sens. Lett.* **3**(1), 63–67 (2006)
33. Chang, C.-I., Wu, C.-C., Liu, W., Ouyang, Y.-C.: A new growing method for simplex-based endmember extraction algorithm. *IEEE Trans. Geosci. Remote Sens.* **44**(10), 2804–2819 (2006)
34. Chen, Y., Zhao, X., Jia, X.: Spectral-spatial classification of hyperspectral data based on deep belief network. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **8**(6), 2381–2392 (2015)
35. Chutia, D., Bhattacharyya, D.K., Sarma, K.K., Kalita, R., Sudhakar, S.: Hyperspectral remote sensing classifications: a perspective survey. *Trans. GIS* **20**(4), 463–490 (2015)
36. Clark, R.N., Roush, T.L.: Reflectance spectroscopy: quantitative analysis techniques for remote sensing applications. *J. Geophys. Res.* **89**(7), 6329–6340 (1984)
37. Cloutis, E.A.: Hyperspectral geological remote sensing: evaluation of analytical techniques. *Int. J. Remote Sens.* **17**(12), 2215–2242 (1996)
38. Craig, M.D.: Minimum-volume transforms for remotely sensed data. *IEEE Trans. Geosci. Remote Sens.* **32**, 542–552 (1994)
39. Fauvel, M.: *Spectral and Spatial Methods for the Classification of Urban Remote Sensing Data*. Ph.D. thesis, Grenoble Institute of Technology (2007)
40. Fauvel, M., Chanussot, J., Benediktsson, J.A.: A spatial-spectral kernel-based approach for the classification of remote-sensing images. *Pattern Recogn.* **45**(1), 381–392 (2012)
41. Fauvel, M., Tarabalka, Y., Benediktsson, J.A., Chanussot, J., Tilton, J.C.: Advances in spectral-spatial classification of hyperspectral images. *Proc. IEEE* **101**(3), 652–675 (2013)
42. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.* **6**(6), 721–741 (1984)
43. Ghiyamat, A., Shafri, H.Z.M.: A review on hyperspectral remote sensing for homogeneous and heterogeneous forest biodiversity assessment. *Int. J. Remote Sens.* **31**(7), 1837–1856 (2010)
44. Gillespie, A.R., Smith, M.O., Adams, J.B., Willis, S.C., Fisher, A.F., Sabol, D.E.: Interpretation of residual images: spectral mixture analysis of AVIRIS images, Owens Valley, California. In: Green, R.O. (ed.) *Proceedings of the 2nd AVIRIS Workshop*, vol. 90–54, pp. 243–270 (1990)
45. Gökhan Akçay, H., Aksoy, S.: Building detection using directional spatial constraints. In: *IEEE IGARSS*, pp. 1932–1935 (2010)
46. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*, 2nd edn. Prentice Hall, Englewood Cliffs (2002)
47. Green, R.O., Eastwood, M.L., Sarture, C.M., Chrien, T.G., Aronsson, M., Chippendale, B.J., Faust, J.A., Pavri, B.E., Chovit, C.J., Solis, M., et al.: Imaging spectroscopy and the airborne visible/infrared imaging spectrometer (AVIRIS). *Remote Sens. Environ.* **65**(3), 227–248 (1998)
48. Gualtieri, J.A., Crompton, R.F.: Support vector machines for hyperspectral remote sensing classification. *Proc. SPIE* **3584**, 221–232 (1998)
49. Guilfoyle, K.J., Althouse, M.L., Chang, C.-I.: A quantitative and comparative analysis of linear and nonlinear spectral mixture models using radial basis function neural networks. *IEEE Trans. Geosci. Remote Sens.* **39**, 2314–2318 (2001)
50. Ham, J., Chen, Y., Crawford, M.M., Ghosh, J.: Investigation of the random forest framework for classification of hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* **43**(3), 492–501 (2005)
51. Hapke, B.: Bidirectional reflectance spectroscopy. I. theory. *J. Geophys. Res.* **86**, 3039–3054 (1981)

52. Hapke, B.: *Theory of Reflectance and Emittance Spectroscopy*. Cambridge University Press, Cambridge (1993)
53. Harsanyi, J., Farrand, W., Chang, C.-I.: Determining the number and identity of spectral endmembers: an integrated approach using Neyman–Pearson eigenthresholding and iterative constrained RMS error minimization. *Proc. Them. Conf. Geol. Remote Sens.* **1**, 1–10 (1993)
54. Harsanyi, J.C., Chang, C.-I.: Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach. *IEEE Trans. Geosci. Remote Sens.* **32**(4), 779–785 (1994)
55. Heinz, D., Chang, C.-I.: Fully constrained least squares linear mixture analysis for material quantification in hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.* **39**, 529–545 (2001)
56. Heinz, D.C., Chang, C.-I., Althouse, M.L.G.: Fully constrained least squares-based linear unmixing. *IEEE IGARSS* **1**, 1401–1403 (1999)
57. Hendrix, E.M.T., Garcia, I., Plaza, J., Martin, G., Plaza, A.: A new minimum volume enclosing algorithm for endmember identification and abundance estimation in hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* **50**(2), 2744–2757 (2012)
58. Hu, Y.H., Lee, H.B., Scarpace, F.L.: Optimal linear spectral unmixing. *IEEE Trans. Geosci. Remote Sens.* **37**, 639–644 (1999)
59. Hughes, G.: On the mean accuracy of statistical pattern recognizers. *IEEE Trans. Inf. Theory* **14**(1), 55–63 (1968)
60. Iordache, M.D., Bioucas-Dias, J., Plaza, A.: Sparse unmixing of hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* **49**(6), 2014–2039 (2011)
61. Jackson, Q., Landgrebe, D.: Adaptive bayesian contextual classification based on Markov random fields. *IEEE Trans. Geosci. Remote Sens.* **40**(11), 2454–2463 (2002)
62. Jimenez, L.O., Landgrebe, D.A.: Supervised classification in high-dimensional space: geometrical, statistical, and asymptotical properties of multivariate data. *IEEE Trans. Syst. Man Cybern.* **28**(1), 39–54 (1998)
63. Keshava, N., Kerekes, J., Manolakis, D., Shaw, G.: An algorithm taxonomy for hyperspectral unmixing. In: *Proceedings of the SPIE AeroSense Conference on Algorithms for Multispectral and Hyperspectral Imagery VI*, vol. 4049, pp. 42–63 (2000)
64. Keshava, N., Mustard, J.F.: Spectral unmixing. *IEEE Signal Process. Mag.* **19**(1), 44–57 (2002)
65. Kettig, R.L., Landgrebe, D.A.: Classification of multispectral image data by extraction and classification of homogeneous objects. *IEEE Trans. Geosci. Electron.* **14**(1), 19–26 (1976)
66. Landgrebe, D.: Hyperspectral image data analysis. *IEEE Signal Proces. Mag.* **1053–5888**, 17–28 (2002)
67. Landgrebe, D.A.: *Signal Theory Methods in Multispectral Remote Sensing*. Wiley, New York (2003)
68. Landgrebe, D.A.: Multispectral land sensing: where from, where to? *IEEE Trans. Geosci. Remote Sens.* **43**(3), 414–421 (2005)
69. Lassalle, P., Inglada, J., Michel, J., Grizonnet, M., Malik, J.: A scalable tile-based framework for region-merging segmentation. *IEEE Trans. Geosci. Remote Sens.* **53**(10), 5473–5485 (2015)
70. Lee, J.B., Woodyatt, S., Berman, M.: Enhancement of high spectral resolution remote-sensing data by noise-adjusted principal components transform. *IEEE Trans. Geosci. Remote Sens.* **28**(3), 295–304 (1990)
71. Li, J., Bioucas-Dias, J.: Minimum volume simplex analysis: a fast algorithm to unmix hyperspectral data. *IEEE IGARSS* **3**, 250–253 (2008)
72. Li, J., Bioucas-Dias, J.M., Plaza, A.: Spectral-spatial hyperspectral image segmentation using subspace multinomial logistic regression and markov random fields. *IEEE Trans. Geosci. Remote Sens.* **50**(3), 809–832 (2013)
73. Li, J., Bioucas-Dias, M., Plaza, A.: Collaborative nonnegative matrix factorization for remotely sensed hyperspectral unmixing. In: *IEEE IGARSS* (2012)

74. Liao, W., Dalla Mura, M., Chanussot, J., Bellens, R., Philips, W.: Morphological attribute profiles with partial reconstruction. *IEEE Trans. Geosci. Remote Sens.* (2015)
75. Ma, X., Jie, G., Hongyu, W.: Hyperspectral image classification via contextual deep learning. *EURASIP J. Image Video Process.* **2015**(1), 1–12 (2015)
76. Maggiori, E., Tarabalka, Y., Charpiat, G.: Improved partition trees for multi-class segmentation of remote sensing images. In: *IEEE IGARSS*, pp. 1016–1019. *IEEE* (2015)
77. Maggiori, E., Tarabalka, Y., Charpiat, G.: Optimizing partition trees for multi-object segmentation with shape prior. In: *26th British Machine Vision Conference* (2015)
78. Mazer, A.S., Martin, M.: Image processing software for imaging spectrometry data analysis. *Remote Sens. Environ.* **24**(1), 201–210 (1988)
79. Melgani, F., Bruzzone, L.: Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote Sens.* **42**(8), 1778–1790 (2004)
80. Merényi, E.: Intelligent understanding of hyperspectral images through self-organizing neural maps. In: *Proceedings of the 2nd International Conference on Cybernetics and Information Technologies, Systems and Applications (CITSA 2005)*, Orlando, FL, USA, pp. 30–35 (2005)
81. Miao, L., Qi, H.: Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization. *IEEE Trans. Geosci. Remote Sens.* **45**(3), 765–777 (2007)
82. Moser, G., Serpico, S.B.: Combining support vector machines and Markov random fields in an integrated framework for contextual image classification. *IEEE Trans. Geosci. Remote Sens.* **51**(5), 2734–2752 (2013)
83. Moser, G., Serpico, S.B., Benediktsson, J.A.: Land-cover mapping by Markov modeling of spatial-contextual information in very-high-resolution remote sensing images. *Proc. IEEE* **101**(3), 631–651 (2013)
84. Nascimento, J.M.P., Bioucas-Dias, J.M.: Vertex component analysis: a fast algorithm to unmix hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* **43**(4), 898–910 (2005)
85. Parente, M., Plaza, A.: Survey of geometric and statistical unmixing algorithms for hyperspectral images. In: *IEEE WHISPERS*, pp. 1–4 (2010)
86. Patra, S., Bruzzone, L.: A batch-mode active learning technique based on multiple uncertainty for svm classifier. *IEEE Geosci. Remote Sens. Lett.* **9**(3), 497–501 (2012)
87. Petrou, M., Foschi, P.G.: Confidence in linear spectral unmixing of single pixels. *IEEE Trans. Geosci. Remote Sens.* **37**, 624–626 (1999)
88. Plaza, A., Benediktsson, J.A., Boardman, J., Brazile, J., Bruzzone, L., Camps-Valls, G., Chanussot, J., Fauvel, M., Gamba, P., Gualtieri, J.A., Marconcini, M., Tilton, J.C., Trianni, G.: Recent advances in techniques for hyperspectral image processing. *Remote Sens. Environ.* **113**(Supplement 1), S110–S122 (2009)
89. Plaza, A., Chang, C.-I.: Impact of initialization on design of endmember extraction algorithms. *IEEE Trans. Geosci. Remote Sens.* **44**(11), 3397–3407 (2006)
90. Plaza, A., Martin, G., Plaza, J., Zortea, M., Sanchez, S.: Recent developments in spectral unmixing and endmember extraction. In: Prasad, S., Bruce, L.M., Chanussot, J. (eds.) *Optical Remote Sensing*, chap. 12, pp. 235–267. Springer, Berlin (2011)
91. Plaza, A., Martinez, P., Perez, R., Plaza, J.: Spatial/spectral endmember extraction by multi-dimensional morphological operations. *IEEE Trans. Geosci. Remote Sens.* **40**(9), 2025–2041 (2002)
92. Plaza, A., Martinez, P., Perez, R., Plaza, J.: A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* **42**(3), 650–663 (2004)
93. Plaza, A., Martinez, P., Plaza, J., Perez, R.: Dimensionality reduction and classification of hyperspectral image data using sequences of extended morphological transformations. *IEEE Trans. Geosci. Remote Sens.* **43**(3), 466–479 (2005)
94. Plaza, J., Hendrix, E.M.T., Garcia, I., Martin, G., Plaza, A.: On endmember identification in hyperspectral images without pure pixels: a comparison of algorithms. *J. Math. Imaging Vision* **42**(2–3), 163–175 (2012)

95. Plaza, J., Plaza, A., Perez, R., Martinez, P.: On the use of small training sets for neural network-based characterization of mixed pixels in remotely sensed hyperspectral images. *Pattern Recogn.* **42**, 3032–3045 (2009)
96. Qian, Y., Jia, S., Zhou, J., Robles-Kelly, A.: Hyperspectral unmixing via  $l_{1/2}$  sparsity-constrained nonnegative matrix factorization. *IEEE Trans. Geosci. Remote Sens.* **49**(11), 4282–4297 (2011)
97. Ratle, F., Camps-Valls, G., Weston, J.: Semisupervised neural networks for efficient hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **48**(5), 2271–2282 (2010)
98. Ren, H., Chang, C.-I.: Automatic spectral target recognition in hyperspectral imagery. *IEEE Trans. Aerosp. Electron. Syst.* **39**(4), 1232–1249 (2003)
99. Richards, J.A.: Analysis of remotely sensed data: the formative decades and the future. *IEEE Trans. Geos. Remote Sens.* **43**(3), 422–432 (2005)
100. Rogge, D.M., Rivard, B., Zhang, J., Sanchez, A., Harris, J., Feng, J.: Integration of spatial-spectral information for the improved extraction of endmembers. *Remote Sens. Environ.* **110**(3), 287–303 (2007)
101. Rubner, Y., Tomasi, C., Guibas, L.J.: A metric for distributions with applications to image databases. In: *ICCV*, pp. 59–66 (1998)
102. Salembier, P., Foucher, S., López-Martínez, C.: Low-level processing of PolSAR images with binary partition trees. In: *IEEE IGARSS* (2014)
103. Salembier, P., Garrido, L.: Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. *IEEE Trans. Image Process.* **9**(4), 561–576 (2000)
104. Scholkopf, B., Smola, A.J.: *Learning with Kernels*. MIT Press, Cambridge (2002)
105. Settle, J.J.: On the relationship between spectral unmixing and subspace projection. *IEEE Trans. Geosci. Remote Sens.* **34**, 1045–1046 (1996)
106. Settle, J.J., Drake, N.A.: Linear mixing and the estimation of ground cover proportions. *Int. J. Remote Sens.* **14**, 1159–1177 (1993)
107. Shaw, G., Burke, H.: Spectral imaging for remote sensing. *Linc. Lab. J.* **14**(1), 3–28 (2003)
108. Singer, R.B., McCord, T.B.: Mars: large scale mixing of bright and dark surface materials and implications for analysis of spectral reflectance. In: *Proceedings of the Lunar and Planetary Science and Conference*, pp. 1835–1848 (1979)
109. Soille, P.: *Morphological Image Analysis*, 2nd edn. Springer, Heidelberg (2003)
110. Soille, P.: Beyond self-duality in morphological image analysis. *Image Vis. Comput.* **23**(2), 249–257 (2005)
111. Subramanian, S., Gat, N., Sheffield, M., Barhen, J., Toomarian, N.: Methodology for hyperspectral image classification using novel neural network. *Proc. SPIE* **3071**, 128–137 (1997)
112. Tarabalka, Y., Benediktsson, J.A., Chanussot, J.: Spectral-spatial classification of hyperspectral imagery based on partitional clustering techniques. *IEEE Trans. Geos. Remote Sens.* **47**(9), 2973–2987 (2009)
113. Tarabalka, Y., Benediktsson, J.A., Chanussot, J., Angulo, J., Fauvel, M.: Classification of hyperspectral data using support vector machines and adaptive neighborhoods. In: *Proceedings of the 6th EARSeL SIG is Workshop* (2009)
114. Tarabalka, Y., Benediktsson, J.A., Chanussot, J., Tilton, J.C.: Multiple spectral-spatial classification approach for hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* **48**(11), 4122–4132 (2010)
115. Tarabalka, Y., Chanussot, J., Benediktsson, J.A.: Segmentation and classification of hyperspectral images using minimum spanning forest grown from automatically selected markers. *IEEE Trans. Syst. Man Cybern. Part B* **40**(5), 1267–1279 (2010)
116. Tarabalka, Y., Chanussot, J., Benediktsson, J.A.: Segmentation and classification of hyperspectral images using watershed transformation. *Pattern Recogn.* **43**(7), 2367–2379 (2010)
117. Tarabalka, Y., Fauvel, M., Chanussot, J., Benediktsson, J.A.: SVM- and MRF-based method for accurate classification of hyperspectral images. *IEEE GRSL* **7**(4), 736–740 (2010)
118. Tarabalka, Y., Rana, A.: Graph-cut-based model for spectral-spatial classification of hyperspectral images. In: *IEEE IGARSS*, pp. 3418–3421. *IEEE* (2014)

119. Tuia, D., Camps-Valls, G.: Urban image classification with semisupervised multiscale cluster kernels. *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.* **4**(1), 65–74 (2011)
120. Tuia, D., Volpi, M., Copa, L., Kanevski, M., Muñoz-Marí, J.: A survey of active learning algorithms for supervised remote sensing image classification. *IEEE J. Sel. Topics Signal Process.* **5**(3), 606–617 (2011)
121. Vaiphasa, C.: Innovative genetic algorithm for hyperspectral image classification. In: *Proceedings of the International Conference on Map Asia*, vol. 20 (2003)
122. Valero, S., Salembier, P., Chanussot, J.: Hyperspectral image representation and processing with binary partition trees. *IEEE Trans. Image Process.* **22**(4), 1430–1443 (2013)
123. Valero, S., Salembier, P., Chanussot, J.: Object recognition in urban hyperspectral images using binary partition tree representation. In: *IEEE IGARSS* (2013)
124. Vapnik, V.: *Statistical Learning Theory*. Wiley, New York (1998)
125. Wan, L., Tang, K., Li, M., Zhong, Y., Qin, A.K.: Collaborative active and semisupervised learning for hyperspectral remote sensing image classification. *IEEE Trans. Geosci. Remote Sens.* **53**(5), 2384–2396 (2015)
126. Wang, J., Chang, C.-I.: Applications of independent component analysis in endmember extraction and abundance quantification for hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.* **44**(9), 2601–2616 (2006)
127. Wax, M., Kailath, T.: Detection of signals by information theoretic criteria. *IEEE Trans. Acoust. Speech Signal Process.* **33**(2), 387–392 (1985)
128. Wu, T.-F., Lin, C.-J., Weng, R.C.: Probability estimates for multi-class classification by pairwise coupling. *J. Mach. Learn. Res.* **5**, 975–1005 (2004)
129. Yang, H., Meer, F.V.D., Bakker, W., Tan, Z.J.: A back-propagation neural network for mineralogical mapping from AVIRIS data. *Int. J. Remote Sens.* **20**(1), 97–110 (1999)
130. Yuhas, R.H., Goetz, A.F.H., Boardman, J.W.: Discrimination among semi-arid landscape endmembers using the spectral angle mapper (SAM) algorithm. *Proc. Ann. JPL Airborne Geosci. Workshop* **1**, 147–149 (1992)
131. Zare, A., Gader, P.: Sparsity promoting iterated constrained endmember detection in hyperspectral imagery. *IEEE Geosci. Remote Sens. Lett.* **4**(3), 446–450 (2007)
132. Zare, A., Gader, P.: Hyperspectral band selection and endmember detection using sparsity promoting priors. *IEEE Geosci. Remote Sens. Lett.* **5**(2), 256–260 (2008)
133. Zymnis, A., Kim, S.-J., Skaf, J., Parente, M., Boyd, S.: Hyperspectral image unmixing via alternating projected subgradients. In: *41st Asilomar Conference on Signals, Systems, and Computers* (2007)