Doctoral Dissertation

Academic Year 2017

Study on Denoising and Unmixing of Hyperspectral Images Exploiting Spectral Linearity

Graduate School of Environmental Engineering
The University of Kitakyushu

Mia Rizkinia

A Doctoral Dissertation
submitted to Graduate School of Environmental Engineering,
The University of Kitakyushu
in partial fulfillment of the requirements for the degree of
DOCTOR of Engineering

Mia Rizkinia

Supervisor:
Professor Masahiro Okuda

Study on Denoising and Unmixing of Hyperspectral Images Exploiting Spectral Linearity

Category: Science / Engineering

Summary

In a noise-free RGB image, a linearity is formed from pixels in its local patches. Color line is the linear cluster in the RGB color space that approximates the shape of color distribution in the local region. As an efficient feature to represent the inter-channel correlation of local regions, the color line is introduced in the field of color image processing. The feature is used to model the correlation among neighboring pixels as well as among the channels in many image processing frameworks. The color-line property is very useful to decorrelate the channels and has been applied to image denoising to reduce discoloration artifacts in RGB images.

In remote-sensing imagery, the need to extract more detailed information has expanded from multispectral images to hyperspectral images that enable pixel-constituent-level analysis. Hyperspectral images have better spectral resolution than multispectral images due to their large number of narrow and contiguous spectral bands. The hyperspectral data can be decomposed (unmixing) into a collection of spectral signatures (spectral library) and a set of corresponding fractions (abundances) that represent the proportion of each spectral signature contained in the pixels (endmember). Hyperspectral data have linearity in their spectral and spatial domains. The correlation among the spectral channels is high due to the narrow spectral resolution. The high correlation also holds among the pixel members of a local region due to the spatial similarity. In a physical sense, the pixels in such regions contain the same materials, either in the same or different fractions. The high spatial correlation also implies linearly dependent abundance vectors in the abundance matrix.

The aim of this study is to generalize the color line to the M-dimensional spectral line feature ($M > 3$) and introduce methods for denoising and unmixing

of hyperspectral images based on the spectral linearity. In the denoising task, we propose a local spectral component decomposition method based on the spectral line. We first calculate the spectral line of an M-channel image, then using the line, we decompose the image into three components: a single M-channel image and two gray-scale images. By virtue of the decomposition, the noise is concentrated on the two images, and thus the proposed algorithm needs to denoise only the two grayscale images, regardless of the number of the channels. As a result, image deterioration due to the imbalance of the spectral component correlation can be avoided. The experiment shows that the proposed method improves image quality with less deterioration while preserving vivid contrast. For unmixing, we propose an algorithm that exploits the low-rank local abundance by applying the nuclear norm to the abundance matrix for local regions of spatial and abundance domains. In our optimization problem, the local abundance regularizer is collaborated with the $L_{2,1}$ norm and the total variation for sparsity and spatial information, respectively. We conduct experiments for real and simulated hyperspectral data sets assuming with and without the presence of pure pixels. The experiments shows that the proposed algorithm yields competitive results and performs better than the conventional algorithms.

Keywords:

color line, spectral linearity, hyperspectral image, nuclear norm, denoising, unmixing, local spectral component decomposition

Graduate School of Environmental Engineering,
The University of Kitakyushu

Mia Rizkinia

Acknowledgements

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

In this chapter, we introduce two main terms of our overall study, i.e., color line and hyperspectral image. Color line plays a significant role in this study, as it becomes a key feature which is adopted in all research topics of this thesis. As for hyperspectral image, it is the object of our research. Brief explanation about remote sensing, hyperspectral unmixing and denoising are also provided to provide a comprehensive introduction to this study.

1.1 Color Line

The recent research in natural color image statistics discovers that the color distribution of a local region is almost linear, typically forms a line in RGB color space. The line is called color line, which represents the elongated cluster in the RGB histogram. The concept of the color line is illustrated in Figure 1.1, where it can be seen that a linearity is formed from each patch of a noise-free RGB image as the color line. The use of color-line property is reported in many applications in color image processing field, as follows [32, 53, 65]

1. **Segmentation**. It is straight forward to integrate the color lines model into a segmentation application. By the color line, it is probable to assign every pixel of an image to each of the color clusters since it can provide a set of the dominant colors in the image. As a simple approach, assigning pixels to their closest color lines by itself yields good results. However, in some cases, the results still need to be improved by using any spatial information or other important features like texture.

2. **Compression**. A compact representation of an image can be created by the virtue of the color line. For each pixel, we need only two values to be stored; an index to its color line and a parameter along the line (or intensity). This representation can be easily compressed later on.

3. **Color editing**. Manipulating color can be very efficient, with a very intuitive way by the use of color line. By applying simple transformation to the color line of the object, we can increase or decrease the color saturation of the object, or even completely replace the colors.

4. **Dehazing**. The task is to remove haze from a single input image with a dark channel prior. The dark channel prior is a kind of statistics of outdoor haze-free images, in which most of the local patches contain some pixels whose intensity is very low in at least one color channel. Using this prior with the haze imaging model, we can directly estimate the thickness of the haze and recover a high-quality haze-free image.

5. **Saturated color correction**.This application is correcting the color of saturated image pixels. The dynamic range of a typical natural or artificial scene is usually larger than the dynamic range that the cameras sensors can capture. Consequently, in many pictures, some of the pixels have at least one saturated color component, to which people are often not sensitive. In the histogram domain, this phenomenon appears in the form of a knee in the color clusters line. The saturated component can be corrected by substituting one of the nonsaturated color components and retrieving the saturated component. It is also possible to use one non saturated component to correct the other two. However, to readjust the dynamic range, gamma correction or other methods for high dynamic range compression is still required.

6. **Image smoothing/denoising**. Using the color line for denoising avoids discolorations which usually occurs in conventional channel-by-channel methods, due to the unbalance relation among the color components. To reduce the discoloration artifact, the outliers that are located away from the color lines are corrected.

Many studies have been conducted to broaden the possible applications of color-line property. Most of them take the natural images as the object. However, the study of color linearity for remote sensing imagery is still lacking. Since the imagery consists of more than three channels and covers a wider wavelength, the linearity can be observed among the adjacent spectral channels. Hence, we introduce a term, i.e.,*spectral line*, for the line feature in remote sensing imagery. We intend to adopt the color-line property to explore the possible applications in
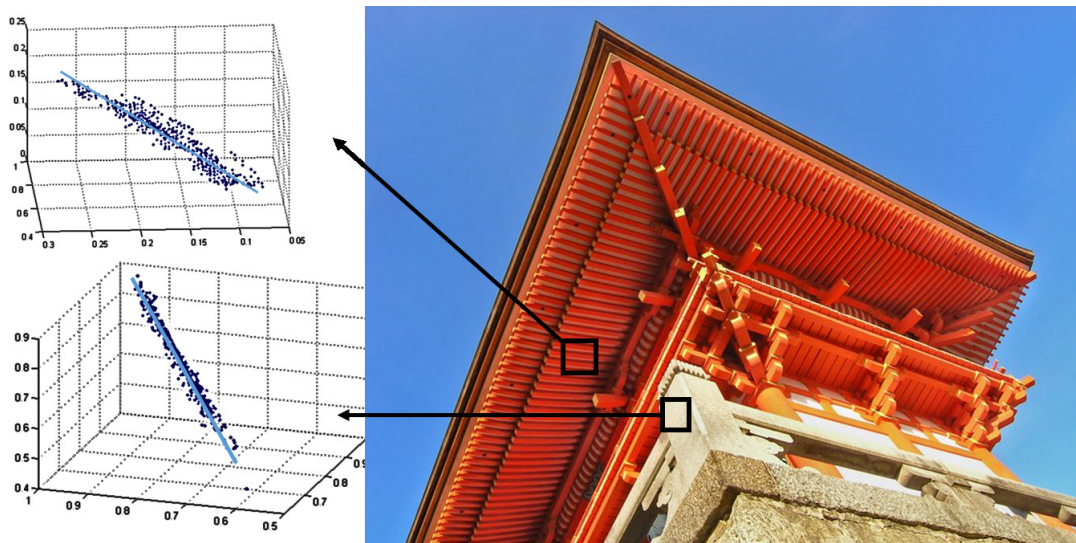
Figure 1.1: Color line for RGB image.

remote sensing field, which some of them are denoising and unmixing of hyperspectral images.

1.2   Remote Sensing

The term *remote sensing* was first introduced in the 1950s by Ms. Evelyn Pruitt of the U.S. Office of Naval Research with the general meaning refers to the science of observing, identifying and measuring an object or a scene without coming into direct contact with it [40]. In other words, it aims to obtain information about an observable scene that is not available to direct exploration. Earth observation through remote sensing has main general goals as follows [17]:

1. Identifying materials on the land cover and analyzing the acquired spectral signal by satellite/airborne sensors

2. Monitoring and modeling the processes on the Earth surface and their interaction with the atmosphere

3. Obtaining quantitative measurements and estimations of geo-bio-physical variables

Due to fact that these objectives require interaction among many disciplines, the remote sensing field has experienced a great evolution to a multi-disciplinary field of science that includes physics, biology, chemistry, signal theory, computer science, electronics, and communications. By remote sensing, it is possible to collect data on dangerous or inaccessible areas. Remote sensing gives benefit that replaces costly and slow data collection on the ground. In the process of data collection, it ensures the areas or objects are not disturbed. Analysis of the acquired multi-channel images enables to develop real-life applications with high social impact, such as urban growing monitoring, crop field identification, disaster prevention, target detection, or biophysical parameter estimation.

In remote sensing, specific instruments are deployed for particular applications. For instance, one of the most widely used technology is that of thematic mappers, such as Landsat. Landsat is mainly applied to update the land cover and land use maps, to identify particular materials, minerals, water or specific crops in the images. Landsat gains popularity due to the free availability of images and the continue and stable performance. They acquire images in several wavelengths of the electromagnetic spectrum, hence it is called multispectral mappers.

Generally speaking, remote sensing imaging instruments can be distinguished based on the type of energy sources they need for the data acquisition, two main types as follows [17]:

1. **Passive optical remote sensing**. It relies on illumination source from solar radiation. The satellite has an imaging spectrometer on board that retrieves signal from the emergent radiation from the Earth-atmosphere system in the observation direction. The radiation acquired by the satellite sensor is measured at different wavelengths. Then, the resulting spectral signature (or spectrum) is used to identify a given material. The field of spectroscopy is concerned with the measurement, analysis, and interpretation of such spectral signatures. Some examples of passive sensors are infrared, charge-coupled devices, radiometers, or multi and hyperspectral sensors.

2. **Active remote sensing**. It is called active because it relies on the energy that is emitted by its antenna. The emission towards the Earths surface and the energy is scattered back to the satellite. This back-scattering energy is the measured parameter. In addition, the time delay between emission and back-scattering is measured to establish the location, height, speed,

and direction of objects. Radar systems are examples of systems for active remote sensing, such as Real Aperture RAR (RAR) or Synthetic Aperture Radar (SAR).

Compared with the product of radar signals, the product generated from passive sensors can be better interpreted as an image in the sense of natural images. In the last decades, the field of passive sensors has greatly evolved in terms of the quality of the acquired images, both in spatial, spectral, and temporal resolutions. This ever-growing evolution increases the difficulty of the signal/image processing problems and the need for improved processing tools. This may be certainly very motivating for the machine learning and signal processing communities [17].

1.3   Hyperspectral Images

As one of the passive optical remote sensing instruments, the hyperspectral sensors are tools with many applications, such as geology, ecology, geomorphology, limnology, pedology, atmospheric science, and forensic science. Goetz et al. in 1985 gave the classical definition for hyperspectral remote sensing (also called imaging spectroscopy) as the acquisition of images in hundreds of contiguous registered spectral bands such that for each pixel, a radiant spectrum can be derived. The processing of the resultant images, called *hyperspectral images*, enables the identification of objects based on their spectral properties, which are mainly related to absorption features. The covered spectral bands are the spectral regions of VIS (visible), NIR (near infrared), SWIR (shortwave infrared), MWIR (midwave infrared), LWIR (longwave infrared), and recently also the UV (ultraviolet), as shown in Table 1.1. The sensor enables the data acquisition in high number of bands as well as high spectral resolution. This means that it provides a narrow bandwidth and a large sampling interval across the spectrum. Basically, this technology includes all spatial domains (microscopic to macroscopic), all platforms (ground, air, and space platforms) and all targets (solid, liquid, and gas). However, in this thesis, we focus on the space-platform hyperspectral images.

The data volume resulting from hyperspectral sensors can be seen as a data cube with two spatial dimensions (correspond to the coverage area) and one spectral dimension (corresponds to the spectral bands). In other words, each pixel is represented by its spectral fingerprint or spectral signature as the observed spectrum. The spectral signature is thus a vector comprising the values of reflected

Table 1.1: Wavelength ranges applied in hyperspectral imaging.

Name of Range	Abbreviation	Wavelength (μm)
Ultraviolet	UV	0.280.35
Visible	VIS	0.350.7
Near infrared	NIR	0.71
Shortwave infrared	SWIR	12.5
Midwave infrared	MWIR	35
Longwave infrared	LWIR	812
Thermal infrared	TIR	350
Infrared	IR	11000

radiation at each wavelength that is covered by the designated spectral bands. Now, modern hyperspectral instruments have capability to record the electromagnetic spectrum in hundreds of spectral bands [40]. As one of them, the Airborne Visible Infra-Red Imaging Spectrometer (AVIRIS) 1 with its 224 spectral bands is able to sample the VIS and NIR spectrum of the reflected energy/light from an area 2 to 12 kilometers wide and several kilometers long. Although AVIRIS is a widely used platform, it constitutes only one source of hyperspectral data. Table 1.2 summarizes other international Earth observation missions with hyperspectral sensors on board, which were already launched or to be launched in the near future.

With the ability providing spatial and spectral information simultaneously, hyperspectral remote sensing enables the identification of objects and other phenomena, as the spectral information is presented on a spatial rather than point (pixel) basis. In addition, allocating spectral information temporally in a spatial domain provides a new dimension that neither the traditional spectroscopy nor air photography can provide separately. Hyperspectral remote sensing thus provides superiority in geographic information system (GIS) with direct and indirect chemical and physical information. Thus, spatial recognition of the phenomenon in question is better performed by the hyperspectral remote sensing than by traditional GIS techniques [17].

However, the use of hyperspectral sensors is still relatively expensive. In addition, to operate the instrument and process the data, it still requires professional manpower. This has opened the opportunities to hyperspectral imaging discipline becoming very active. In addition to the growing number of scientific papers and

Table 1.2: Overview of some remote sensing missions including hyperspectral sensors.

	Hyperion	Prisma	EnMAP	HyspIR	Sentinel-5
Origin	USA	Italy	Germany	USA	Europe
Spatial Resolution (m)	30	5-30	30	60	7000
Spectral Range (nm)	400-2500	400-2500	420-2450	380-2500	270-2385
Spectral Resolution (nm)	10	10	6.5-10	10	0.25-1
Swath width (km)	7.7	30	30	120	2715
Launch	2000	2010	2012	2018	2021
Lifetime(years)	10	6	6	6	7.5

conferences focusing on this technology, commercial sensors are being developed, orbital sensors are in advanced planning stages, national and international funds are being directed toward using this technology, and interest from the private sector increases [17]. Hyperspectral imaging has been found to be very useful in many terrestrial, atmospheric, and marine applications. The high spectral resolution of hyperspectral image combined with temporal coverage enables better recognition of targets and an improved quantitative analysis of phenomena, especially for land use cover application. The technology is applied by decision makers, farmers, environmental watchers in both the private and government sectors, city planners, stock holders, and others. As the number and variety of processing tasks in hyperspectral imaging is enormous, the majority of algorithms can be organized according to the following specific tasks [40]:

1. **Dimensionality reduction**. In order to facilitate subsequent processing tasks, this task consists of reducing the dimensionality of the input hyperspectral data in the initial phase.

2. **Target and anomaly detection**. The tasks consist of searching the pixels of a hyperspectral data cube for specific and rare (either known or unknown) spectral signatures.

3. **Change detection**. It means finding the significant changes of the user's object of interest between two or more hyperspectral scenes of the same geographic region and different acquisition time.

4. **Classification**. This task aims to assign a label or class to each pixel of a hyperspectral data cube.
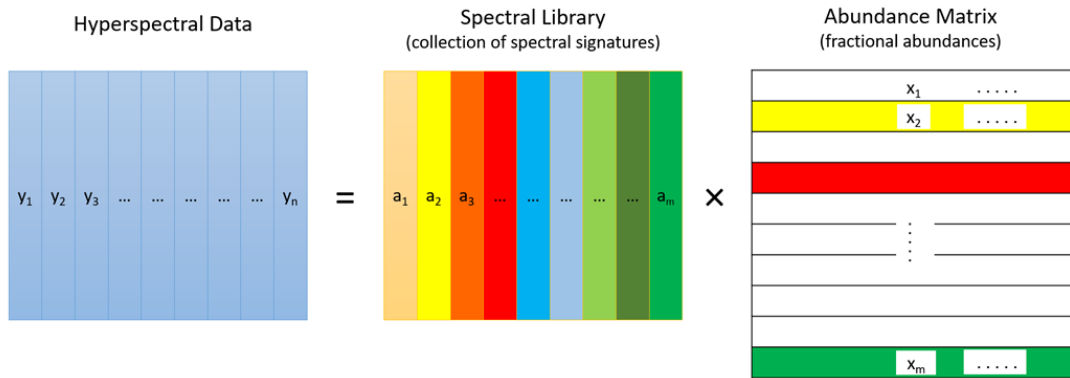
Figure 1.2: Illustration of unmixing of a hyperspectral data.

5. **Spectral unmixing**. This is a sub-pixel analysis that consists of estimating the fraction of the pixel area covered by each material present in the scene.

1.4   Hyperspectral Unmixing

Beyond the capability of multispectral remote sensing in classifying whole pixels, hyperspectral remote sensing enables further analysis in sub-pixel level, i.e., the constituent materials that comprise a pixel. The sub-pixel analysis is related to the fact that the sensors capture the pixels of interest as a combination of numerous disparate components on the Earth's surface. The mixtures are occurred in a scale of a pixel due to the low spatial resolution of the sensors, thus two or more materials occupy the same pixel. Another possible circumstance is when the sensors capture some distinct materials that have merged into homogeneous mixtures on the Earth's surface. Collecting data in hundreds of spectral bands, hyperspectral sensors have demonstrated the capability to quantitatively decompose or unmix these mixtures which is called *spectral unmixing*. Spectral unmixing is the procedure by which the measured spectrum of a mixed pixel is decomposed into a collection of constituent spectra (or *endmembers*), and a set of corresponding fractions, called *abundances*. The collection of endmembers' spectral signatures is called *spectral library*. Figure 1.2 illustrates this decomposition.

The goal of spectral unmixing is to estimate the materials present in one pixel, their pure signatures and the relative area that they occupy (i.e., their

8
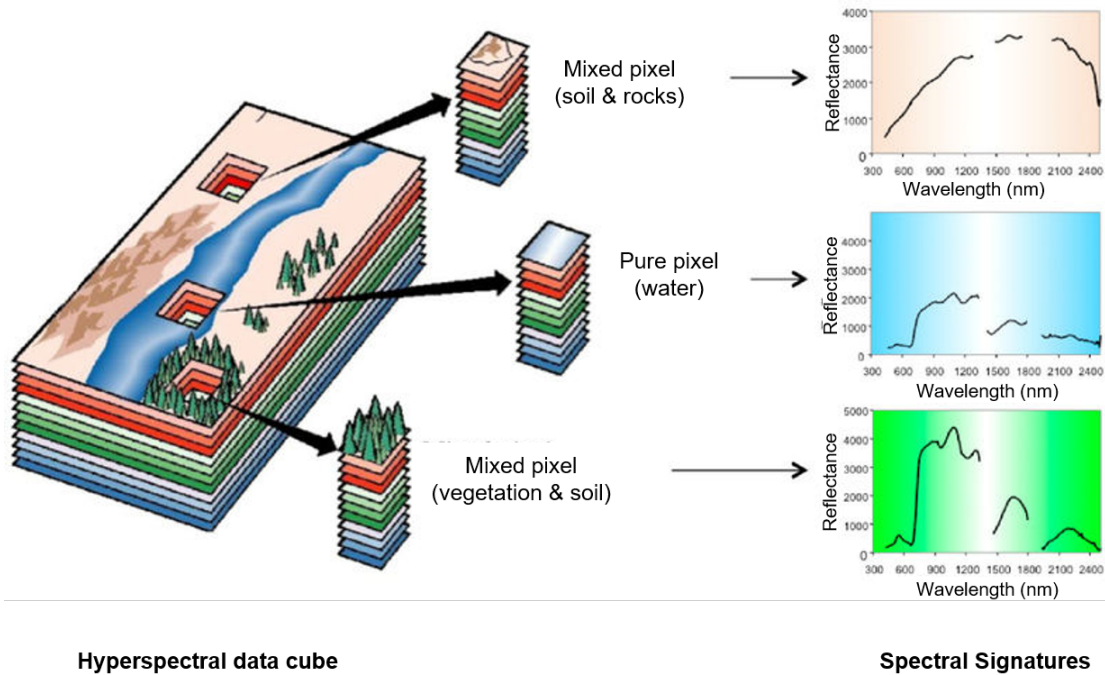
Figure 1.3: Illustration of pure and mixed pixels of a hyperspectral scene and the corresponding spectral signatures of each material [4].

fractional abundance) in a given pixel. The abundances indicate the proportion of each endmember present in the pixel. Generally, they correspond to familiar macroscopic objects in the scene, such as water, soil, metal, vegetation, etc. In practice, most of the pixels in a scene are mixed. A pixel containing more than one material is called *mixed pixel*, while a pixel containing only one constituent material is called *pure pixel*. Figure 1.3 illustrates pure and mixed pixels of a hyperspectral scene and the corresponding spectral signatures of each material.

Since the earliest days of hyperspectral image and signal processing, spectral unmixing has been an appealing research topic and exploitation purpose [40]. Regardless the spatial resolution, the spectral signatures collected in natural environments assuredly reflect a mixture of various material signatures found within the spatial coverage of the ground field view of the sensor. However, the very fine spectral resolution available from hyperspectral instruments enables to compensate the relatively low spatial resolution (which varies from tens to thousands of meters) and perform sub-pixel characterization by taking advantage of the fact that the observed spectrum of a pixel can be decomposed.

In this thesis, the spectral unmixing problem becomes one of our research topics, which currently represents a very active area in remote sensing in general, and in hyperspectral data analysis in particular. We exploit the spectral-line property for the hyperspectral unmixing problem.

1.5   Multi-channel Image Denoising

Multi-channel images, such as multispectral and hyperspectral images, are often noisy in many situations because sensors have narrower spectral sensitivity functions and thus capture less light than normal RGB imaging devices. Whereas various applications, such as classification, target detection, spectral unmixing, and change detection need detailed and accurate spectral information [10], the noise due to, for example, thermal electronics and dark current, unavoidably contaminates the image acquisition process [66, 74, 79], which disrupts detailed spectral information and furthermore degrades its performance in the listed applications. Thus, denoising the images is a crucial phase in the preprocessing steps of these applications.

It is effective for image denoising methods to exploit inter-channel correlation as well as spatial correlation. Unlike channel-by-channel methods that tend to produce an imbalance of colors, nowadays many smoothing and denoising methods take inter-channel correlation into account to avoid color deterioration as shown by state-of-the-art denoising methods [13, 19, 26, 55]. Meanwhile, in hyperspectral images, the narrow spectral resolution leads high correlation among the adjacent channels. In this thesis, we introduce a novel denoising method that consider the inter-channel correlation using spectral line property.

1.6   Thesis Organization

The structure of the thesis is summarized in Figure 1.4. As shown in the figure, the remaining structure comprises three chapters. After introducing two main terms of our overall study, i.e., color line and hyperspectral image, in Chapter 2, we describe the core of the proposed idea, which is the spectral-line concept as the background of our research. To relate with the designated applications, which are denoising and unmixing, we review the state-of-the-art of related works in this chapter. In Chapter 3, we describe our work of local spectral vector decomposition as an application of the spectral linearity in the denoising task. Finally, Chapter

4 describes our work that incorporates the spectral linearity to the hyperspectral unmixing problem by introducing the local abundance regularizer for sparse unmixing. Finally, Chapter 5 resumes the general conclusions and future research lines derived from the present study.
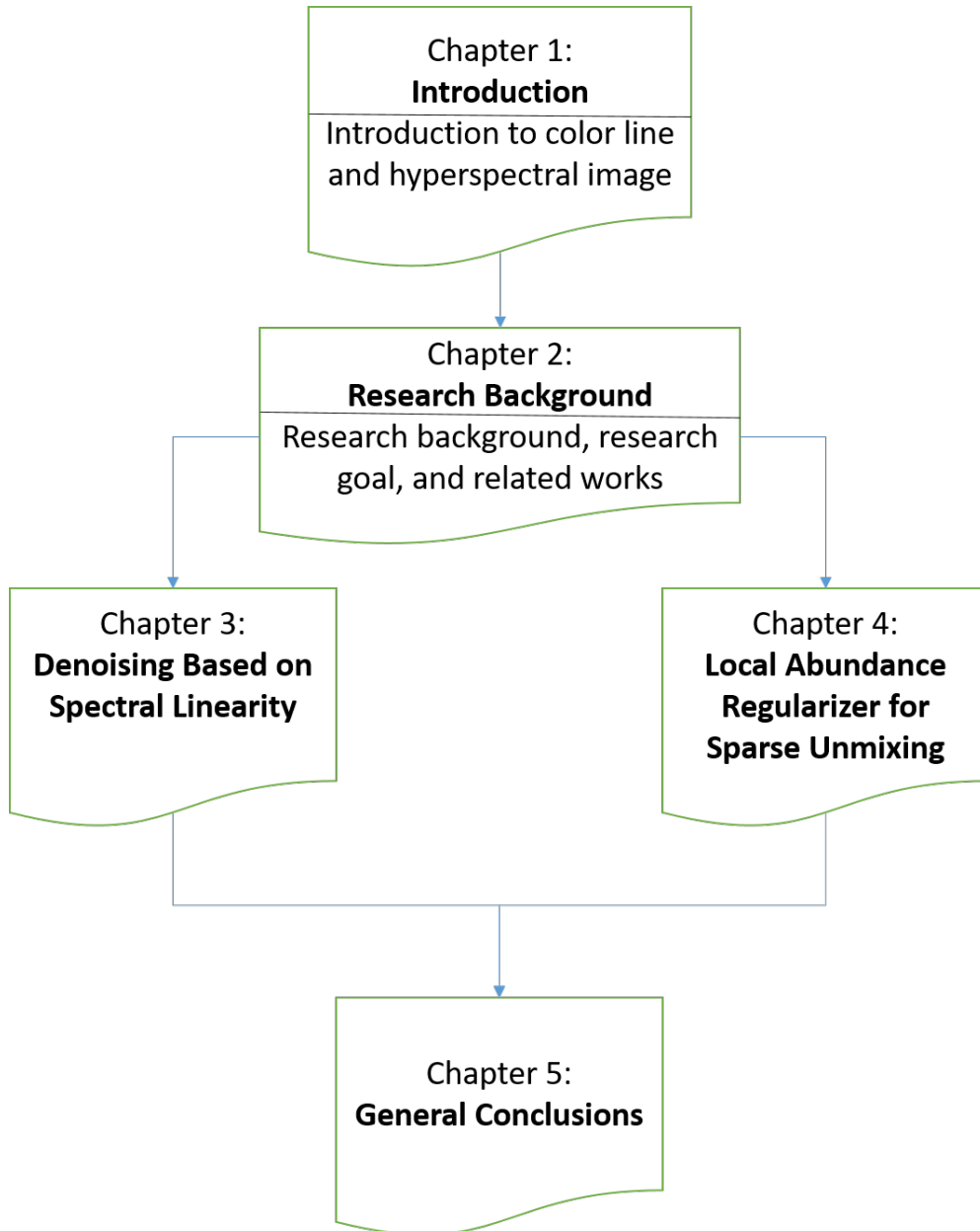
Figure 1.4: The structure of our thesis

Chapter 2
Research Background

2.1　Research Background

2.1.1　Observation of Spectral Linearity in Hyperspectral Images

The idea to exploit the spectral line for multi-channel image denoising comes from the finding that noisy RGB images tend to contain outliers located away from the color line. The color-line property is very useful to decorrelate the channels and has been applied to RGB image denoising to reduce discoloration artifacts [65]. As for multi-channel images, in this case hyperspectral images, the correlation among the channels is also expected high due to the narrow spectral resolution. Thus, in line with the principle of color line in RGB images, we can observe the inter-channel correlation of hyperspectral images by plotting the intensity distribution of a local region (refers to a sub-scene comprising adjacent pixels and channels).

We made observations using some real hyperspectral data. One of them is in the Indian Pines data, as shown in Figure 2.1. The figure visualizes the intensity distribution of a local spatial region depicted in the red rectangle, and the effect of noise as well as denoising to the distribution. The channels of 12-14, 132-134, and 155-157 represent locality in the spectral domain. The original data distribution of each local region shows a linearity of three adjacent channels.

To find some more convincing evidence, we observe the inter-channel correlation via the principal component analysis (PCA). PCA is known as a mathematical procedure that seeks a linear combination of components, with principal component as the indicator. The procedure transforms a number of possibly-correlated components into a smaller number of uncorrelated components, which are the principal components. The principal components can be indicated by eigenvalues or singular values (detailed explanation in Chapter 3). In our observations, for
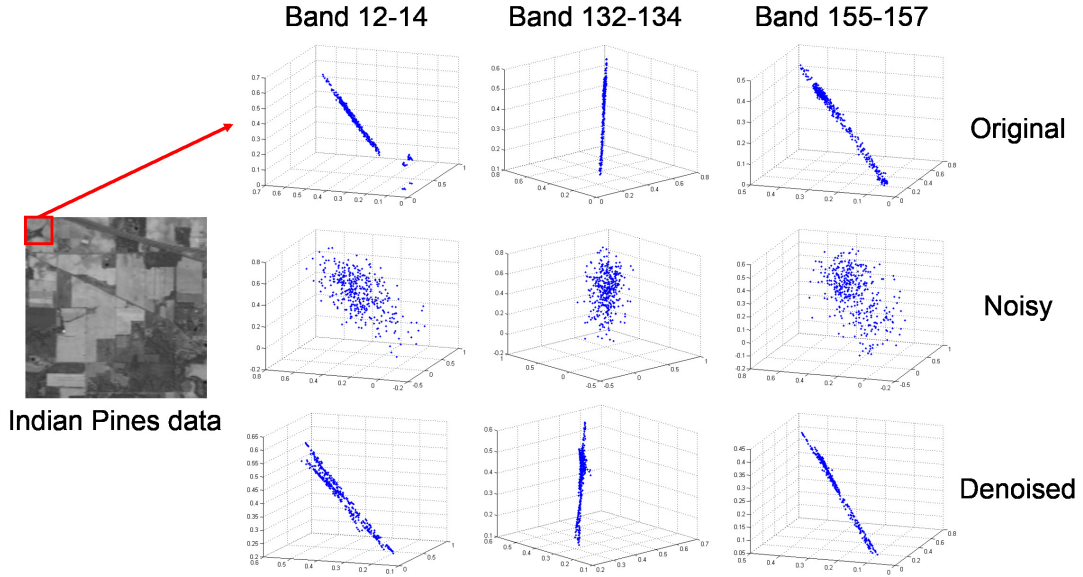
Figure 2.1: Spectral line in hyperspectral images and the effect of denoising.

various number of channels, we confirmed that the ratio between the maximum eigenvalues and the sum of all eigenvalues is high, which implies the channels are linearly correlated. For example, the Pavia center data has a ratio of 0.86 for all channels, and a ratio of 0.92, 0.93, and 0.86 for 50, 35, and 10 adjacent channels, respectively.

From these observations, we conclude that the line property also holds in the hyperspectral data, that is the spectral line.

2.1.2   Linearity in Abundance Domain

The local-spectral linearity of hyperspectral images brings a consequence to the matrix of which they are decomposed. In an unmixing task, it is known that a hyperspectral image is decomposed into a spectral library and an abundance matrix (see Figure 1.2). The spectral library is a collection of selected spectral signatures that are highly correlated. The abundance matrix consists of abundance vectors that corresponds to the spectral signatures. The high spatial correlation of the hyperspectral data implies linearly dependent abundance vectors in the abundance matrix citeLRR. Consequently, the local-spectral linearity of the hyperspectral image indicates high correlation among the abundance vectors that corresponds
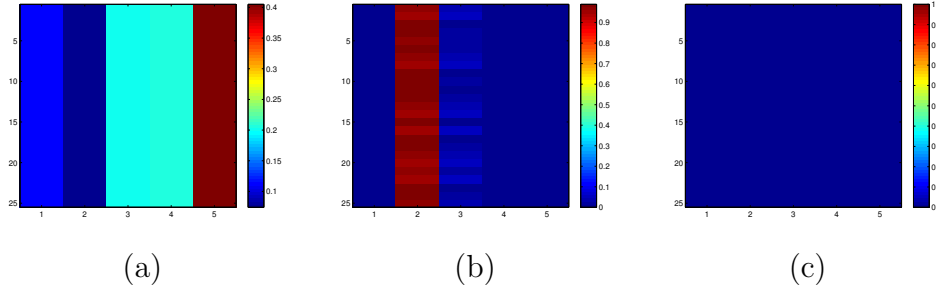
Figure 2.2: Examples of low rank matrix $\mathbf{H}_{\hat{\mathbf{x}}_b}$ from data (a) DS (rank = 1), (b) FR5 (rank = 2), and (c) when no endmember presences on the local region (rank = 0).

to the pixels in the local regions. In other words, the abundance matrix composed of these vectors is low rank (rank corresponds to the number of linearly independent columns of a matrix).

Beside two dimensions in the spatial domain, the abundance matrix has a dimension in abundance domain when transformed to a 3D form. Using simulated data, we observe the linearity of the data distribution in the abundance domain by taking the singular values of the true abundance matrix for each local block (block refers to the 3D, which the third dimension has a local coverage in the spectral signature/endmember direction). We found that there is one value that dominants to others (the ratio is close to one) in each local block. For example, the matrix of *DS* data and *FR5* data on Figure 2.2 have the ratio 1 and 0.9852, respectively. On the other hand, the ratio value will be less dominant as the region becomes the whole matrix (nonlocal), which become 0.5881 and 0.2262 for *DS* data and *FR5* data, respectively. It implies that the linearity in abundance domain is satisfied for the abundance matrix with local point of view.

We also observe the local blocks in vectorized form (in Chapter 4, denoted with $\mathbf{H}_{\hat{\mathbf{x}}_b}$) when the endmembers exist in the region. It is expected that the matrix $\mathbf{H}_{\hat{\mathbf{x}}_b}$ is low rank because the pixels in local regions tend to share the same active set of endmembers. Figure 2.2 shows the low rankness is hold in $\mathbf{H}_{\hat{\mathbf{x}}_b}$. In addition, the high correlation between spectral signatures in the spectral library promotes the columns in $\mathbf{H}_{\hat{\mathbf{x}}_b}$ to be low rank.

2.2   Research Goal

The aim of this study is to generalize the color-line property of RGB images to the M-dimensional spectral line feature ($M > 3$) for multi-channel images, and exploit it for some multispectral/hyperspectral image research topics, which are denoising and unmixing.

For denoising application, we first extend the color line to more general multi-channels and call it the spectral line. We design a denoising method for multi-channel images based on the local spectral linearity. In the case of the multi-channel images, we consider the intensity distribution of M channels in every local region, which corresponds to the color distribution in RGB color images. Then, we evaluate the effectiveness of our propose method for multi-channel images.

As for hyperspectral unmixing, our research aims to develop an unmixing algorithm based on the linearity in the abundance domain. We propose the local abundance regularizer and implant it to the sparse unmixing problem. In the implementation, we use the nuclear norm, a popular heuristic algorithm for the rank minimization, for 3D local regions. We introduce the 3D local block that slides through the three dimensions of the abundance cube and impose the nuclear norm to promote the low-rank structure of the local abundance cube. Then, we compare the performance with the state-of-the-art methods. The difference with the 3-tensors nuclear norm [33] is that our proposed scheme takes the local nuclear norm of the third dimension in addition to the first two dimensions. In other words, the sliding block also moves in a particular step size along the third dimension.

2.3   Related Works

2.3.1   Related Works in Hyperspectral Image Denoising

For image restoration, Blomgren et al. [13] proposed the total variation (TV) for color and other vector-valued images. Chan et al. [19] improved the TV method for nonlinear color models with regard to: the chromaticity-brightness (CB) and hue-saturation-value (HSV). As a nonlocal filtering approach, Color BM3D (CBM3D) [26] is one of the most powerful denoising methods for RGB images. To reduce color artifacts found in CBM3D, local color nuclear norm (LCNN) has been introduced [55], which exploits the correlation among the channels using the low rank property of a local region.

Similarly, it is expected that, for the hyperspectral image, exploiting the correlation in not only the spatial domain but also the spectral domain improves denoising performance because the hyperspectral images have high correlation between adjacent channels since they are retrieved from channels with a high spectral resolution. Channel-by-channel hyperspectral image denoising has a consequence in a low SNR because it ignores the spectral correlation [29]. By exploiting the spectral correlation, Atkinson et al. [7] restore hyperspectral images based on the discrete Fourier transform (DFT) and 2-dimensional discrete wavelet transform (2D-DWT) for decorrelation in the spectral and spatial domain, respectively. The same motivation is also found in another work [79], which utilizes local/global redundancy and correlation (RAC) in the spatial and spectral domain. However, this method is less competitive than other state-of-the-art methods when it comes to strong noise. Dabov et al. propose a video denoising method known as VBM3D [25], which is an extension of a single channel denoising method [28], where noise is reduced by using patches found in neighboring frames. This method performs well for multi-channel images, which will be demonstrated in this study.

As an efficient feature to represent the inter-channel correlation of local regions, a color line is introduced in the field of color image processing [53]. The feature is used to model the correlation among neighboring pixels as well as among the channels in many image processing frameworks. This work [53] precisely distinguishes one color from another by its color line. From this idea, they implement a color-line model for some applications, i.e. segmentation, compression, color editing and saturated color correction. For demosaicing, they also use natural image properties: least color variation and minimal corner value [54]. Fattal [32] exploits the color-line pixel regularity of a single image to introduce a new dehazing method. He derives a local formation model that explains color lines in hazy scenes and uses it for estimating scene transmission. Color-line-based noise reduction has also been introduced [71]. They elaborate the color-line model with conventional filters, such as the bilateral filter and the nonlocal means filter, to improve their performance. Ono et al. exploit the color correlation by minimizing a convex function with the LCNN [55]. This method does not have denoising capability in itself, and its main purpose is to remove color artifacts. This method outperforms other RGB denoising methods, but its superiority in hyperspectral denoising is still limited.

2.3.2   Related Works in Hyperspectral Unmixing

Hyperspectral unmixing can be reconstructed from the linear mixture model (LMM) and nonlinear mixture model [6, 43, 47, 73]. With the LMM, it is assumed that the spectra of each mixed pixel are linear combinations of the endmembers contained in the pixel. Despite the fact that it holds only for macroscopic mixture conditions [6, 81], it is widely used due to its computational tractability and flexibility in various applications. With the LMM, several unmixing techniques have been introduced based on either geometry [11, 52], statistics [8, 11], nonnegative matrix factorization (NMF) [45, 48, 58, 67], or sparse regression [11, 31, 34, 40, 46].

Although the geometry and statistical techniques are unsupervised and require only a little prior information about the data, they require an assumption that at least one pure pixel (a pixel containing only one endmember) exists for each endmember [68]. The NMF techniques do not require this assumption, however, they can obtain virtual endmembers with no physical meaning [22, 68]. On the other hand, in the sparse regression techniques, additional informations are introduced as prior knowledge that are added to the objective functions in the optimization problems and called regularizers, e.g., considering the abundance sparsity [9, 37, 69], information of endmembers known to exist in the data [68], or total local spatial differences [38].

An abundance sparsity regularizer algorithm, called sparse unmixing by variable splitting and augmented Lagrangian (SUnSAL), was introduced by Iordache et al. [37]. They applied the L_1 norm (the sum of the absolute values of the matrix columns) to the abundance matrix, substituting the L_0 norm (the number of nonzero elements of the matrix) to impose the sparsity. With the algorithm known as collaborative SUnSAL (CLSUnSAL), it is assumed that the pixels of a hyperspectral scene share the same active set of endmembers [39]. This assumption does not hold when an endmember is contained in several pixels instead of all pixels in the scene. For example, when the hyperspectral scene captures a location that contains locally homogeneous regions. Zhang et al. [77] proposed a local approach of the CLSUnSAL considering the fact that endmembers tend to be distributed uniformly in local spatial regions. Qu et al. [61] adopted joint sparsity combined with the low-rank model under the bilinear mixture model (BMM). The low-rank term corresponds to the low number of linearly independent columns of a matrix. They applied a local sliding window to the abundance matrix as the neighboring pixels tend to be homogeneous and constituted from the same materials.

Iordache et al. [38] proposed a spatial regularizer algorithm called sparse un-

mixing with the total variation regularizer (SUnSAL-TV), which uses an unmixing technique that is more powerful than the conventional unmixing ones. Neverthe-less, this semi-supervised algorithm may produce over-smoothed results and blur in the edges. The spatial information is also imposed in the sparse unmixing task in a nonlocal procedure [81]. Tang et al. [68] introduced an algorithm called sparse unmixing using a priori information (SUnSPI). The required prior knowledge is that some spectral signatures (endmembers) in the hyperspectral scene are known in advance. Despite the fact that the performance is superior compared to that of conventional unmixing algorithms, it is difficult to guarantee whether the as-sumption can always hold. Field investigation or prior hyperspectral-data analysis may be needed to provide such information. The structure tensor nonlocal TV (ST-NLTV) [23] models the high spatial and spectral correlation in multi-channel images for several applications, including the hyperspectral unmixing. The idea is extending the NLTV-based regularization to multi-channel images by taking advantage of the structure tensor resulting from the gradient of a multi-channel image.

In a region with high spatial similarity, e.g., local spatial region, the correla-tion among pixels spectral signatures can be reflected as linear dependence among their corresponding abundance vectors. The abundance matrix that is composed of these vectors should be low rank. This low-rankness has been recently applied for hyperspectral image denoising and recovery tasks [76, 78, 80], which results in superior performances. Furthermore, the low-rankness of the data also indicates high correlation among the abundance vectors corresponding to the pixels in such regions [61]. Giampouras et al. [35] proposed ADSpLRU algorithm by exploiting the low-rankness of abundance to the sparse unmixing problem to consider the spatial correlation of the abundance. However, they considered the low-rankness in the nonlocal fashion of the abundance dimension. In practice, to consider the local low-rankness of an image, Ono et al. [56] proposed the local color nuclear norm (LCNN). However, they locally applied the nuclear norm (the sum of the matrix singular values) only to the spatial dimension of RGB images. Yang et al. [72] also imposed the low-rank constraint for coupled sparse denoising and un-mixing problems. However, the use of the nuclear norm is not local, and superior performance is more dominant in the denosing task rather than the unmixing one. To the best of our knowledge, there is no sparse unmixing algorithm that takes into account the low-rankness of local spectral signatures (endmembers) in the abundance dimension, whereas the high correlation between the spectral signa-

tures can be guaranteed by the spectral angle (SA), which is a spectral similarity assessment defined as the angle between two spectral vectors. In turn, one can observe the linearity of the data distribution in local regions in terms of spatial as well as abundance dimension. This priori may lead to a novel approach for the sparse unmixing algorithm.

Chapter 3

Denoising Based on Spectral Linearity

Abstract – We propose a method for local spectral component decomposition based on the line feature of local distribution. Our aim is to reduce noise on multi-channel images by exploiting the linear correlation in the spectral domain of a local region. We first calculate a linear feature over the spectral components of an M-channel image, which we call the *spectral line*, and then using the line, we decompose the image into three components: a single M-channel image and two gray-scale images. By virtue of the decomposition, the noise is concentrated on the two images, and thus our algorithm needs to denoise only the two gray-scale images, regardless of the number of the channels. As a result, image deterioration due to the imbalance of the spectral component correlation can be avoided. The experiment shows that our method improves image quality with less deterioration while preserving vivid contrast. Our method is especially effective for hyperspectral images. The experimental results demonstrate that our proposed method can compete with other state-of-the-art denoising methods.

    Index Terms – Spectral line, local spectral component decomposition, denoising, hyperspectral image.

3.1 Introduction

To improve the performance of various applications of hyperspectral images, denoising step becomes an essential preprocess phase in hyperspectral image exploitation. Recently, different denoising methods for hyperspectral image have been proposed. The traditional 2D or 1D denoising methods, which reduce noise in the image band by band or pixel by pixel, are the simplest way. However, as only spatial or spectral noise is removed, the corresponding denoising result by

this way is not satisfying. If we only reduce noise in spatial or spectral domain, artifacts or distortions will be introduced in other domains. At the same time, this kind of methods will destroy the correlation in spatial or spectral domain. Spatial and spectral information should be considered jointly to remove the noise efficiently.

The aim of our research is to reduce noise on multi-channel images by exploiting the linear correlation in the spectral domain of a local region. As color-line property has shown the effectiveness to correlate the three channels of RGB images, we generalize the color line to the M-dimensional spectral line feature for multi-channel images, and introduce a method for local spectral component decomposition based on the spectral line.

3.1.1 Basic Theory

Denoising

All sensor devices, both analog and digital, have traits that make them susceptible to noise. The process of removing noise from a signal is called denoising. Noise in any data set must be low. There exists no absolute scale for noise but rather all noise is measured relative to the measurement. A common measure is the signal-to-noise ratio (SNR), the ratio of signal power to the noise power, often expressed in decibels (dB). A high SNR indicates high precision data, while a low SNR indicates noise contaminated data. In signal processing, a process that removes some unwanted components from a signal is called filtering, in which the algorithm is called filter.

BM3D

Block-matching and 3D filtering (BM3D) is a powerful denoising algorithm based on sparse 3D transform-domain collaborative filtering. The blocks are achieved by grouping similar 2D fragments of the image into 3D data arrays which are called groups. To deal with these 3D groups, a procedure of collaborative filtering is developed. In the first step, the input noisy image is processed by extracting reference blocks from it. For each such block, the algorithm finds blocks that are similar to the reference one (block matching) and stacks them together to form a 3D array (group). Then, it is followed by performing collaborative filtering to each group and return the obtained 2D estimates of all grouped blocks to their original locations. After processing all the reference blocks, the obtained block

estimates can overlap and thus there are multiple estimates for each pixel. Finally, these estimates are aggregated to form an estimate of the whole image [27].

Eigenvalue

Eigenvalues play an important role in situations where the matrix is a transformation from one vector space onto itself. An eigenvalue and eigenvector of a square matrix $\mathbf{A}$ are a scalar λ and a nonzero vector $\mathbf{x}$ so that

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x}. \tag{3.1}$$

The eigenvalue-eigenvector equation for a square matrix can be written

$$(\mathbf{A} - \lambda\mathbf{I})\mathbf{x} = 0, \quad \mathbf{x} \neq 0. \tag{3.2}$$

This implies that $\mathbf{A} - \lambda\mathbf{I}$ is singular and hence that $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$.

PCA

Principal component analysis (PCA) is a mathematical procedur that approximates a general matrix by a sum of a few simple matrices. Simple means rank one; all of the rows are multiples of each other, and so are all of the columns. Let $\mathbf{X}$ be any real m-by-n matrix. Singular value decomposition (SVD) decomposes $\mathbf{X}$ as

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top, \tag{3.3}$$

Here, $\mathbf{\Sigma}$ is a rectangular diagonal matrix of nonnegative singular values of $\mathbf{X}$; $\mathbf{U}$ is a matrix whose the columns are orthogonal unit vectors called the left singular vectors of $\mathbf{X}$; and $\mathbf{V}$ is a matrix whose columns are orthogonal unit vectors and called the right singular vectors of $\mathbf{X}$.

Equation (3.3) can be written as

$$\mathbf{X} = \mathbf{Q}_1 + \mathbf{Q}_2 + \dots \mathbf{Q}_p, \tag{3.4}$$

where $p = min(m, n)$. The component matrices $\mathbf{Q}_k$ are rank one outer products:

$$\mathbf{Q}_k = \sigma_k u_k v_k^\top, \tag{3.5}$$

where each column of $\mathbf{Q}_k$ is a multiple of u_k (the k-th column of $\mathbf{U}$), each row is a multiple of $v_k^\top$ (the transpose of the k-th column of $\mathbf{V}$). The component matrices

are orthogonal to each other $\mathbf{Q}_j\mathbf{Q}_k^\top = 0, \quad j \neq k$. The norm of each component matrix is the corresponding singular value σ_k ($\|\mathbf{Q}_k\| = \sigma_k$). Consequently, the contribution each $\mathbf{Q}_k$ makes to reproduce $\mathbf{X}$ is determined by the size of the singular value σ_k.

If the sum is truncated after $r < p$ terms,

$$\mathbf{X}_r = \mathbf{Q}_1 + \mathbf{Q}_2 + \ldots \mathbf{Q}_r, \tag{3.6}$$

the result is a rank r approximation to the original matrix $\mathbf{X}$. In fact, $\mathbf{X}_r$ is the closest rank r approximation to $\mathbf{X}$. Since the singular values are in decreasing order, the accuracy of the approximation increases as the rank increases.

The description and notation for PCA vary widely. A common description for the principal components is in terms of eigenvalues and eigenvectors of the cross-product matrix $\mathbf{X}^\top\mathbf{X}$. As

$$\mathbf{X}^\top\mathbf{X}\mathbf{V} = \mathbf{V}\boldsymbol{\Sigma}^2, \tag{3.7}$$

the columns of $\mathbf{V}$ are the eigenvectors $\mathbf{X}^\top\mathbf{X}$. The columns of $\mathbf{U}$, scaled by the singular values, can then be obtained from

$$\mathbf{U}\boldsymbol{\Sigma} = \mathbf{X}\mathbf{V}. \tag{3.8}$$

PCA is applied on the covariance matrix. The eigenvectors and eigenvalues of the covariance matrix correspond to the principal components and variances explained by the principal components, respectively. PCA of the covariance matrix provides an orthonormal eigen-basis for the space of the observed data, in which the maximum eigenvalues correspond to the principal components that are correlated with most of the covariability among a number of observed data.

3.1.2 Chapter Structure

The rest of this chapter is organized as follows. In Section 3.2, we first give an overview of our proposed algorithm. Then, we describe the whole algorithm based on the spectral line property. In Section 3.3, we present the experimental results and compare them with those of other methods for both RGB color images and multi-channel images including multispectral and hyperspectral images. Finally, we conclude our research in Section 3.4.

3.2   Algorithm

The spectral line is found by applying PCA to the local window centered at a pixel. In our case, a noisy input is given, which may result in inaccurate line estimation. One possible solution to address this problem is to apply pre-denoising before PCA, but the quality of the resultant image heavily depends on the pre-denoising method. For example, weak denoising does not improve the accuracy of the line estimation, but hard denoising may change the balance of the spectral information which also results in inaccurate estimation. In our method, we address this problem by an iterative fashion. First, we find an initial estimate for the spectral line from the noisy input, and then apply our decomposition algorithm. We apply a conventional denoising method to the two components yielded by the decomposition described in Section 3.2.2. Then, we estimate the spectral line again using the denoised image, and repeat this procedure. As for the denoising method, after testing some conventional methods, we adopt BM3D, which gives the best performance with our method. The effect of the denoising is adjusted to be relatively weak at each iteration since the method is applied iteratively. The whole procedure consists of four steps as illustrated in Figure 3.1.

1. We calculate the local spectral distribution in a window centered at each pixel and find the principal component for each pixel by PCA. We define the spectral line vector as the principal component and a spectral vector field as an image that has the spectral line vector at each pixel. Then, we align the direction of each vector by changing the sign so that the neighboring vector directions become smooth, which improves the resultant image (described in Sec. 3.2.1).

2. Using the spectral line vector, we decompose each pixel $\mathbf{Y}_i$ $(\in \mathbb{R}^M)$ of a M-channel image into the three components: a mean spectral component $\boldsymbol{\mu}_i$ $(\in \mathbb{R}^M)$, spectral line component D_i $(\in \mathbb{R})$, and residual component N_i $(\in \mathbb{R})$ (Sec. 3.2.2). The aim of the decomposition is to transfer noise only to the two components (spectral line and residual components). We assume that the noise is independent and zero-mean, and thus the mean spectral component tends to have little noise. Regardless of the number of channels, we only need to denoise the two components, which is especially effective for multi-channel images with $M > 3$.

3. We smoothing the spectral line and residual components (Sec. 3.2.3).

4. We reconstruct an image from the above components, and then go back to step 1 (Sec. 3.2.4).

3.2.1   Spectral Line Vector Field

Spectral Line Vector Estimation by PCA

The spectral line vector is formulated as the eigenvector that corresponds to the maximum eigenvalue by using PCA. The detail of the procedure is listed as follows:

1. Calculate the mean spectral component of each pixel i for each channel,

$$\boldsymbol{\mu}_i = \frac{1}{w} \sum_{j \in \mathcal{N}(i)} \mathbf{Y}_j \quad (i = 1, 2, \ldots, k), \tag{3.9}$$

where k is the number of pixels in an image, w is the number of pixels in a specified filter window $\mathcal{N}(i)$ and $\mathbf{Y}_j = [Y_j^1, Y_j^2, \ldots, Y_j^M]^\top$ is the intensity of neighboring pixels $j \in \mathcal{N}(i)$. M is the number of channels (e.g. $M = 3$ for an RGB image). This calculation produces the mean $\boldsymbol{\mu}_i = [\mu_i^1, \mu_i^2, \ldots, \mu_i^M]^\top$, which we call the mean spectral component.

2. Calculate the covariance matrix $\mathbf{C}_i \in \mathbb{R}^{M \times M}$ of neighboring pixels around each pixel i,

$$\mathbf{C}_i = \left(\frac{1}{w} \sum_{j \in \mathcal{N}(i)} \mathbf{Y}_j \mathbf{Y}_j^\top \right) - \boldsymbol{\mu}_i \boldsymbol{\mu}_i^\top. \tag{3.10}$$

3. To obtain the spectral line vector, find the maximum eigenvalue d_i of the covariance matrix $\mathbf{C}_i$, and subsequently derive its corresponding eigenvector as the spectral line vector $\mathbf{v}_i$.

Alignment of Spectral Line

The resulting eigenvector $\mathbf{v}_i$ may have sign s_i with an ambiguity ($s_i = +1$ or -1). The direction of the sign should vary smoothly in our framework, otherwise the resultant image will have jaggy artifacts, which we will demonstrate with an experiment in Sec.3.3.1.

To make the direction of the sign smoothly vary, we adopt the Jacobi relaxation method to determine the sign. For the vector direction alignment, the sign s_i should be set to fit the dominant direction of neighboring vectors by using the
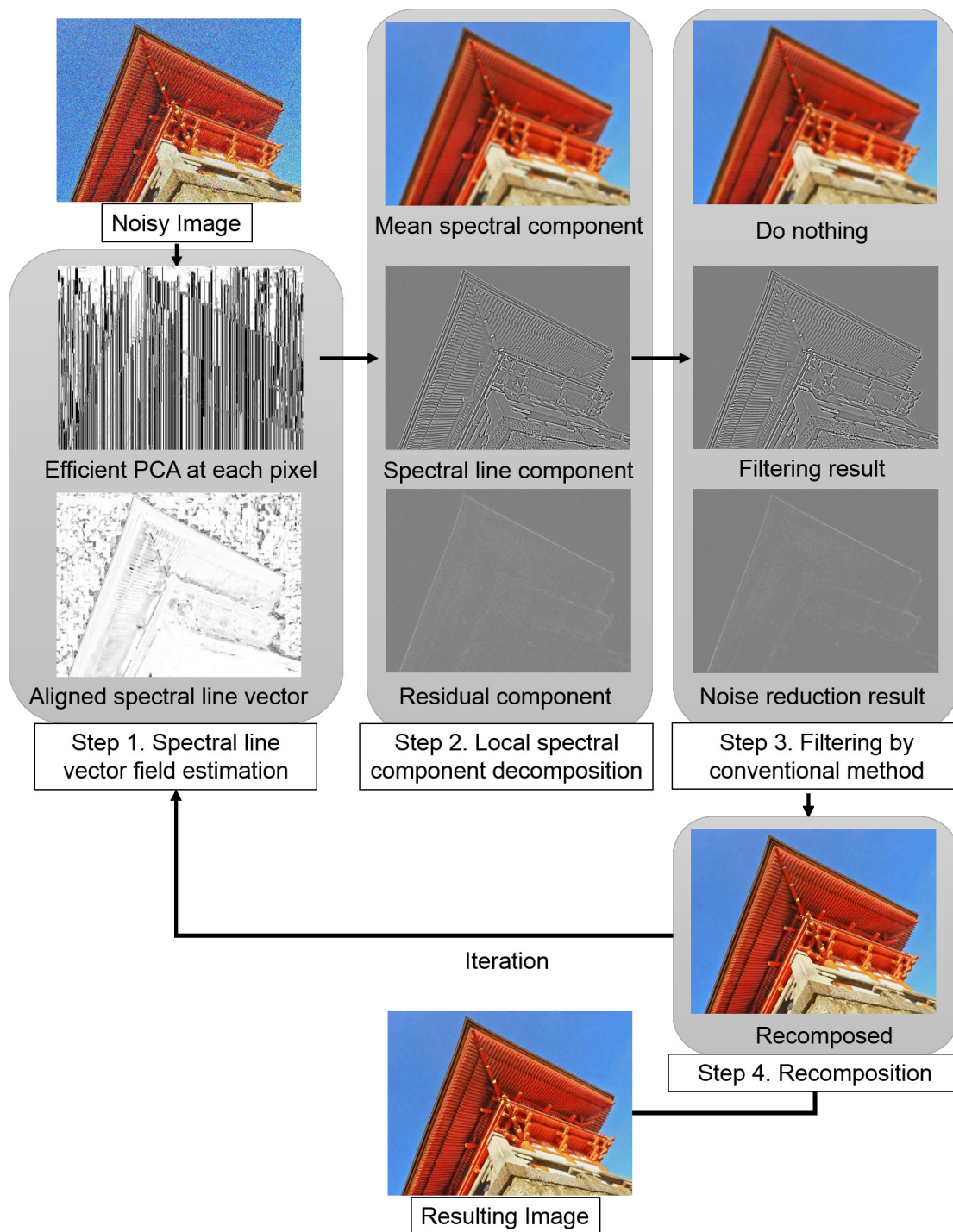
Figure 3.1: Flowchart of the proposed algorithm with four main steps conducted iteratively.

inner product as the criterion. To extend the pixel-wise flip to a larger region, a multi-resolution approach is used.

In the initialization, to determine the sign of each vector $\mathbf{v}_i$ that minimizes the following energy function among neighboring pixel pairs $\{i, j\} : \sum_{\{i,j\}} \|s_i \mathbf{v}_i - s_j \mathbf{v}_j\|^2$, we adopt the Jacobi relaxation method [14]:

$$s_p^{(t+1)} = \text{sign}\Big(\sum_{q \in \mathcal{N}(p), \ q \neq p} (s_p^{(t)} \mathbf{v}_p)^\top (s_q^{(t)} \mathbf{v}_q) \Big), \tag{3.11}$$

which means that the sign s_p of a pixel p is aligned with the dominant sign of 3×3 neighboring vectors $q \in \mathcal{N}(p)$, considering the inner product. In practice, we calculate it with the use of the box filter [59] for acceleration. After finding the sign, we update the spectral line vector as

$$\tilde{\mathbf{v}}_i = s_i \mathbf{v}_i$$

The result of the initial flip is then processed by multi-grid's V-cycle [14] as a multi-resolution approach. The multi-resolution pyramid for vector and sign images in Figure 3.2 is generated using Gaussian pyramid decomposition [16]. Additionally, in the decimation process, to give priority to pixels around edges that have large eigenvalues, we multiply the eigenvalue d_i as a weight for the pixel $d_i s_i \mathbf{v}_i$ and then apply the decimation filter and re-normalize the half-sized vector field. As for the multi-resolution eigenvalue images that consists of d_i, they are generated by using the same approach as the one used for the Gaussian pyramid.

Figure 3.3 visualizes the effect of the method, in which the pixel values represent the cosine of the angle between $\mathbf{v}_i$ and a fixed vector $\mathbf{a}$, that is $< \mathbf{v}_i, \ \mathbf{a} > / \|\mathbf{v}_i\| \|\mathbf{a}\|$ with the inner product $< \cdot, \cdot >$. The pixel values are normalized to show it appropriately for visualization. One can see from the figure that the sign flip works effectively for the vector field.

3.2.2   Spectral Component Decomposition

Using the mean spectral component $\boldsymbol{\mu}_i$ and the spectral line vector $\tilde{\mathbf{v}}_i$, we decompose the original pixel $\mathbf{Y}_i$ to the three terms. We begin with calculating the difference vector of the center pixel and the mean spectrum of the local window, $\Delta \mathbf{Y}_i = \mathbf{Y}_i - \boldsymbol{\mu}_i$. Using the spectral line vector $\tilde{\mathbf{v}}_i$, the spectral line component D_i is calculated as the inner product of the normalized spectral line vector $\tilde{\mathbf{v}}_i$ and $\Delta \mathbf{Y}_i$,
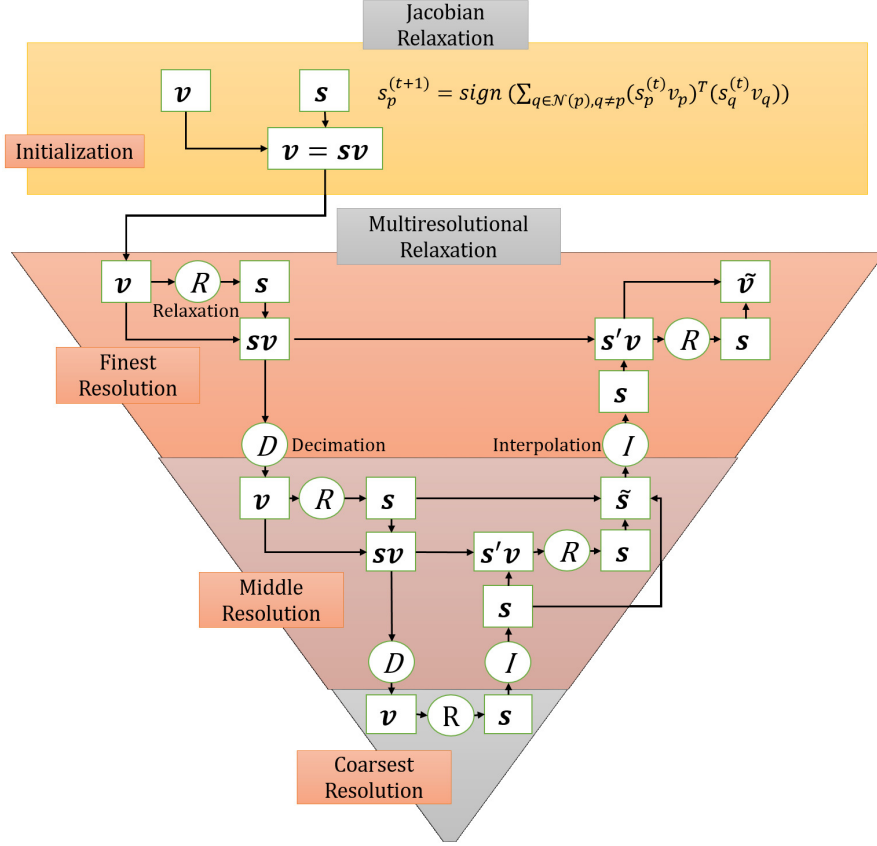
Figure 3.2: Flowchart of the spectral line vector alignment.

$$D_i = \tilde{\mathbf{v}}_i^\top \Delta \mathbf{Y}_i, \tag{3.12}$$

which is the component of $\Delta \mathbf{Y}_i$ w.r.t. the direction of $\tilde{\mathbf{v}}_i$, since $\tilde{\mathbf{v}}_i$ is normalized to $\|\tilde{\mathbf{v}}_i\| = 1$. Then, we calculate the difference between the two vectors:

$$\mathbf{r}_i = \Delta \mathbf{Y}_i - D_i \tilde{\mathbf{v}}_i. \tag{3.13}$$

Finally, a residual component is derived from L_2 norm of the vector $\mathbf{r}_i$,

$$N_i = \|\mathbf{r}_i\|. \tag{3.14}$$

Note that the dimensions of the mean spectral component $\boldsymbol{\mu}_i$, the spectral line vector $\tilde{\mathbf{v}}_i$ and residual vector $\mathbf{r}_i$ are the same as the number of channels, that is M, and the spectral line component D_i and the residual component N_i are scalars.
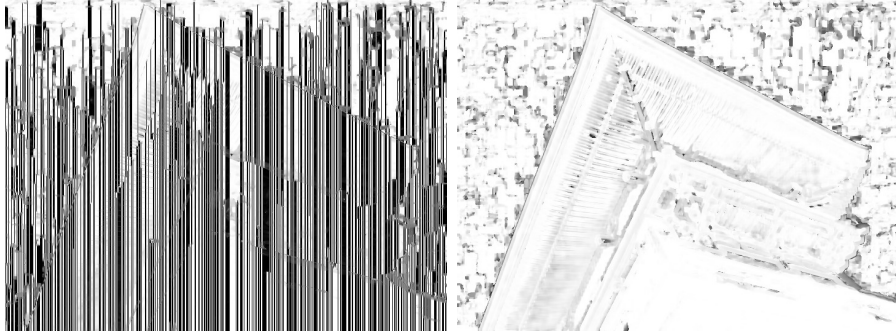
Figure 3.3: The effect of sign flip: sign maps (left) before and (right) after sign flip.

The main purpose of this decomposition is to concentrate noise, which is originally scattered in all the channels of an input image, into the two components, D_i and N_i. The noise in the components is reduced in the next step.

3.2.3   Filtering

The spectral line component D_i obtained in the previous step contains noise. Consequently, denoising the spectral line components is required in the spatial domain. We refine the spectral line component by denoising it with BM3D [28]. Since we iteratively apply BM3D, the denoising effect is adjusted not to be too strong. This procedure results in the filtered spectral line component $\overline{D}_i$.

The residual component N_i contains a lot of weak noise. To filter them out, we adopt a two-phase denoising procedure. First, we apply the Geman McClure robust function [12] to reduce the noise with small intensities by the formula

$$w(N_i) = \frac{N_i^2}{k + N_i^2}, \tag{3.15}$$

where k is a small constant. Afterward, it is added as the weight as

$$\hat{N}_i = w(N_i)N_i. \tag{3.16}$$

Then, the filtering step is followed by BM3D, which results in the residual component $\overline{N}_i$.

As for the mean spectral component $\boldsymbol{\mu}_i$, unlike the other components, there is no need to apply filtering because it has been already generated by averaging Equation (3.9).
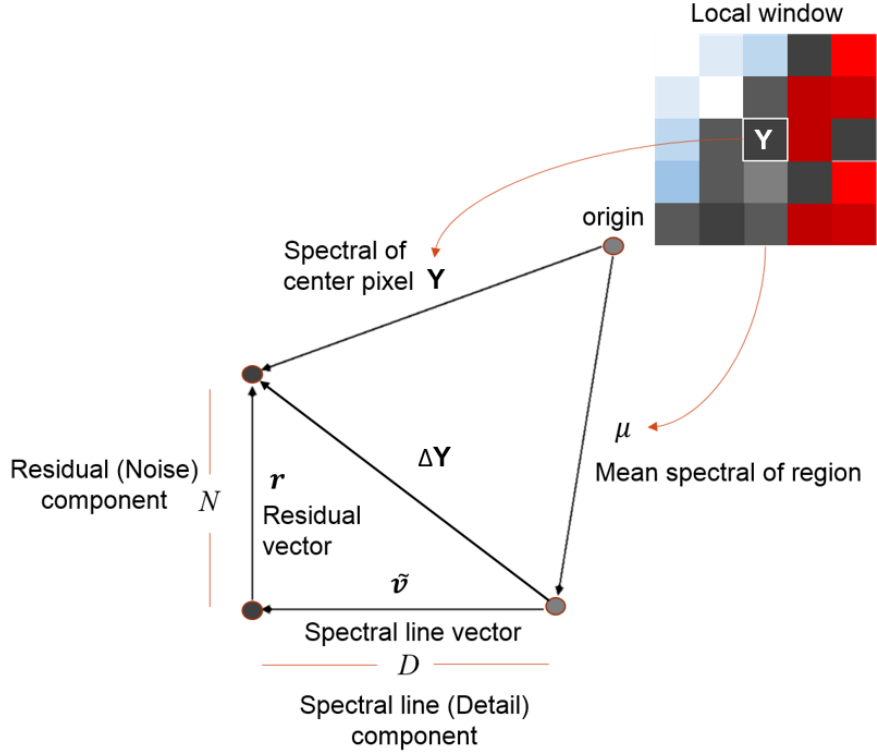
Figure 3.4: Spectral decomposition: In local spectral component decomposition, each pixel is decomposed into a spectral line vector, a spectral line component, and a residual component.

3.2.4   Recomposition

The final step is recomposition of the resulting image from its constituent components. The expected image can reconstructed as follows (see Figure 3.4):

$$\overline{\mathbf{Y}}_{\mathbf{i}} = \boldsymbol{\mu}_i + \overline{D}_i \tilde{\mathbf{v}}_i + \frac{\overline{N}_i}{N_i} \mathbf{r}_i. \tag{3.17}$$

3.3   Experiment

In the experiment, all the methods are implemented in MATLAB, except for calculating the mean spectral component, where the MEX file is used for running the box filter in C.

3.3.1   Effect of Sign Flip

First, to discuss the importance of sign flip before filtering, we perform an experiment that emphasizes how it works on an image. The Jacobi relaxation used for the vector sign alignment significantly improves the performance by flipping the sign to the same direction as the local mean color sign. Figure 3.5 shows the effect of the vector sign flip in our method. The top and bottom row show the results when the sign flip is not performed and performed, respectively. After PCA, the generated spectral line vectors are not smooth (top left). If we continue to the next steps without this procedure, the spectral line components and the filtered result are affected and fail to preserve the details. To avoid this problem, we use the Jacobi relaxation method for the spectral line vectors. As a result, the method can perform better as shown in the bottom images. The comparison of the final resultant images is depicted in Figure 3.6. The difference can easily be noticed in some regions of the image.

3.3.2   RGB color image

Parameter Setting

The parameters of the conventional methods are adjusted so as to give the best evaluation values. On the other hand, for the conventional methods used in combination with our method, we set their parameters to give a similar degree of denoising. Then, we will show that our method preserves more vivid contrast than the others with the same degree of denoising.

To achieve the expected results, the number of main iterations is set to 4. To calculate the color orientation, we set half of the window size to 3. We found after some trial-and-error that these values yield sufficient results for most of the tested images.

Experimental Results

Before evaluating the whole of our method, we examine the contribution of our decomposition step. Only for Figure 3.7, we use the anisotropic diffusion to denoise the spectral line component instead of BM3D to fairly evaluate the validity of the decomposition in the denoising process, and compare it with the stand-alone anisotropic diffusion. Thus, the two rows of Figure 3.7 show the comparison with the anisotropic diffusion. Figure 3.7 (a) and (b) depict original images and

Spectral line vector   Spectral line component   After smoothing
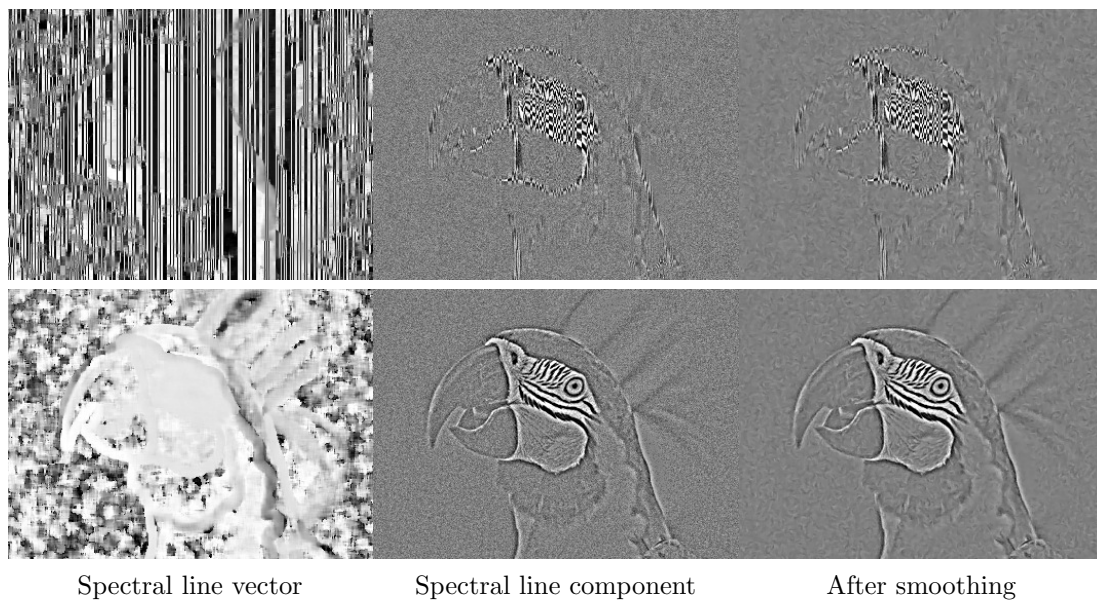
Figure 3.5: The results obtained without sign flip (top) and with sign flip (bottom) for the spectral line vector (left), the spectral line component (middle), and the smoothing result of spectral line component (right).

(a) Original   (b) With sign flip   (c) Without sign flip

Closeup of differences
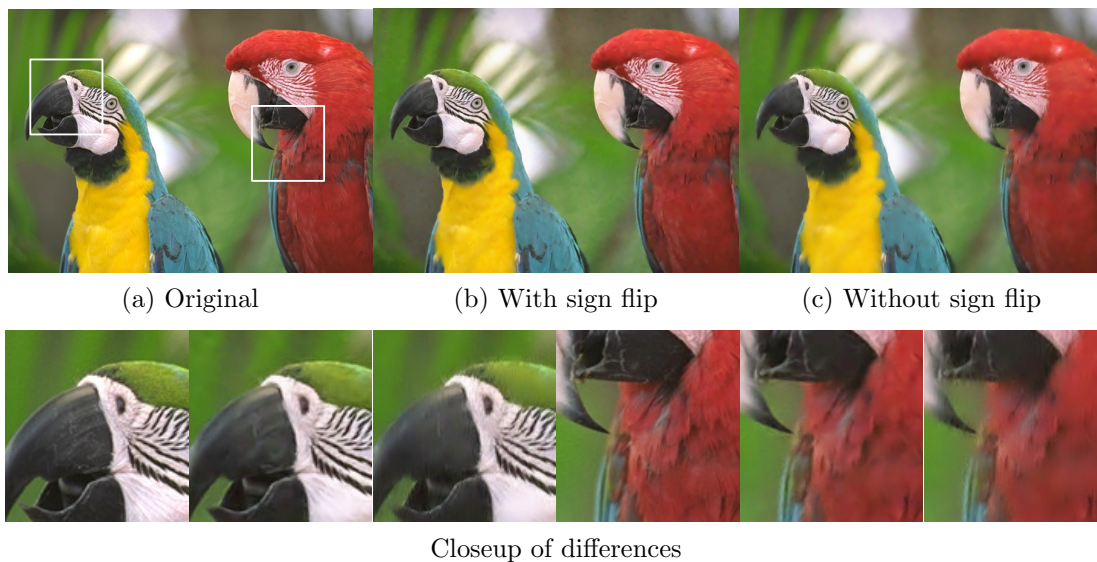
Figure 3.6: Effect of sign flip in the resultant image. From the left, (a) original image, (b) our method, and (c) our method without sign flip.

(a-1) Original (PSNR dB)   (b-1) Noisy   (c-1) Diffusion in RGB (30.01)   (d-1) Diffusion in YCbCr (33.33)   (e-1) Ours (34.64 dB)

(a-2) Original (PSNR dB)   (b-2) Noisy   (c-2) Diffusion in RGB (27.96)   (d-2) Diffusion in YCbCr (29.35)   (e-2) Ours (30.48)
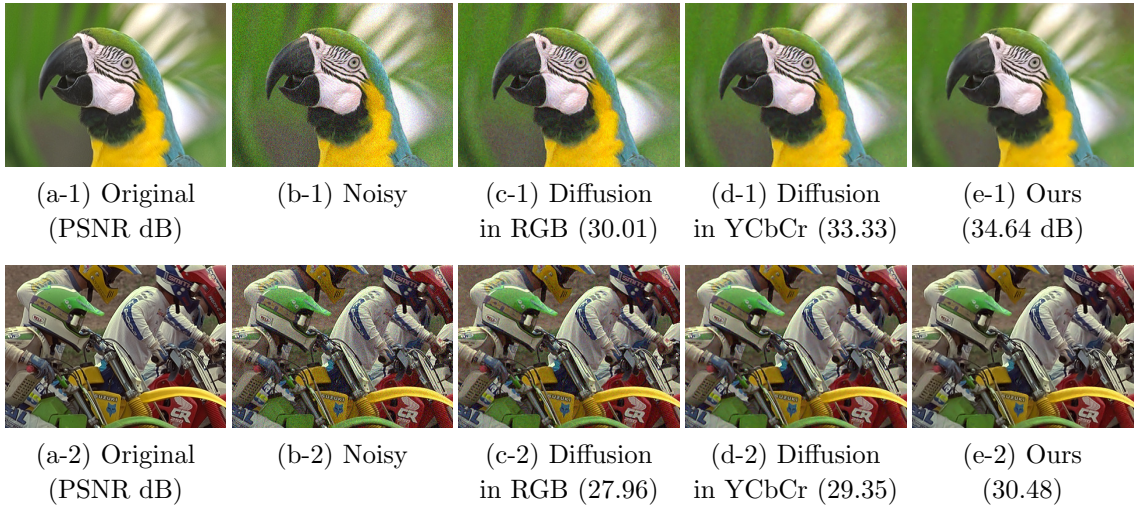
Figure 3.7: Comparison of smoothing results by the anisotropic diffusion [70] and by the anisotropic diffusion in our algorithm

(a-1) Original (PSNR)   (b-1) Noisy   (c-1) CBM3D (34.07)   (d-1) Ours (32.38)

(a-2) Original (PSNR)   (b-2) Noisy   (c-2) CBM3D (39.92)   (d-2) Ours (40.41)
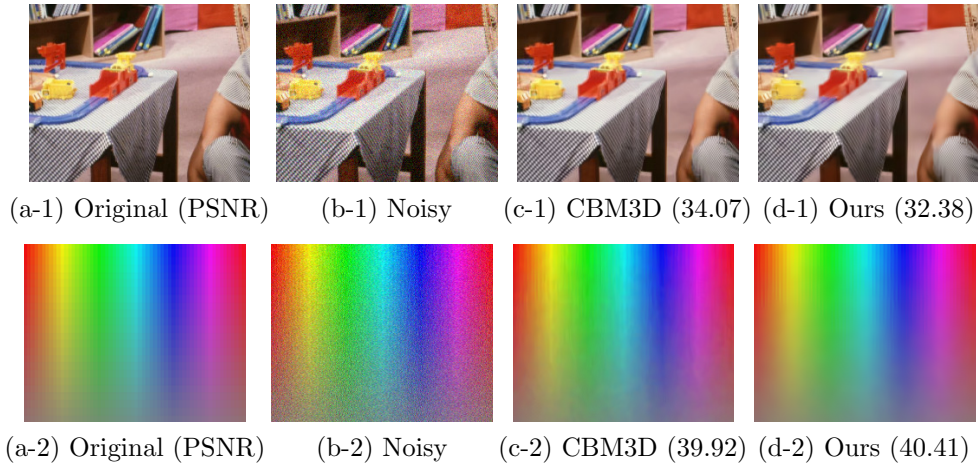
Figure 3.8: Comparison of denoising results by the Color BM3D [26] and by BM3D in our algorithm

noisy images, respectively, in which we add the Gaussian noise of a standard deviation $\sigma = 0.06$ for normalized intensity range [0, 1]. Figure 3.7 (c) and (d) are the results of the anisotropic diffusion alone, where the method is applied in the RGB and YCbCr color spaces, respectively, and (e) is obtained by the anisotropic diffusion after our decomposition. One can see that our method is able to significantly reduce the discoloration artifact, compared to the anisotropic diffusion

34

used alone. The artifacts can be noticed distinctly in the background part of the images. Furthermore, it confirms the validity of our method in both the appearance and numerical evaluations, which are better than those of the conventional method.

In the next example shown in Figure3.8, we compare our performance with CBM3D [26]. Whereas with the results of CBM3D, the differences at a glance are little, and our numerical evaluations are sometimes inferior. The difference in evaluation values comes from the remaining noise around the edges in the results of our method. Human perception, however, is insensitive to the noise.

In addition, the spectral line property partially does not hold for images with much more detailed texture as shown in the patterned tablecloth of the first row of Figure 3.8. While CBM3D works well for the images with the patterned texture, the original CBM3D tends to produce discolorations especially in smoothly varying gradation as shown in the second row of Figure 3.8, where our method significantly reduces the discoloration and improves the perceptual appearance. Thus, CBM3D does not work well for images with varying gradation, while our method is superior in this case.

3.3.3   Multi-channel Image

Data Description

In our method, only the two components (two channels) are denoised after the decomposition regardless of the number of channels in an input image, and thus the method is more effective for the multi-channel images (multispectral and hyperspectral images) than the RGB images. Four real multispectral data are used in this experiment, i.e. the cropped area of Kyushu Island in Japan, Yellow River area in China, Papua Island in Indonesia and Washington in USA. They are collected by Operational Land Imager (OLI), and instrument onboard the Landsat 8 satellite. The multispectral band consists of band $1-7$, dedicated for coastal aerosol, red, green, blue, NIR, SWIR 1 and SWIR 2 channel respectively, with wavelength range $0.43-2.29\mu m$. Regarding this, the parameter M as the number of processed bands is set to 7. We contaminate the data by Gaussian noise with standard deviation of 0.06.resulting SNR of 24.44.

For hyperspectral images, five real data sets are used in the experiment, i.e. Cuprite, Indian Pines, Salinas, Pavia University, and a subscene of Pavia Center data set. The first three and the last two are collected by airborne hyperspectral

Table 3.1: Data set description used in the experiment.

Data Set	Size	Sensor	Spatial Resolution
Cuprite	$512 \times 614 \times 224$	AVIRIS	$20\ m$
Indian Pines	$145 \times 145 \times 220$	AVIRIS	$20\ m$
Pavia-U	$610 \times 340 \times 103$	ROSIS	$1.3\ m$
Pavia-C	$200 \times 200 \times 102$	ROSIS	$1.3\ m$
Salinas	$512 \times 217 \times 224$	AVIRIS	$3.7\ m$

sensors AVIRIS and ROSIS, respectively. AVIRIS has a spectral resolution of 10 nm in the spectral range 400-2400 nm. ROSIS has a spectral resolution of 4 nm in the spectral range of 430-860 nm. The sizes and spatial resolutions for the data sets are given in Table 3.3. Because of the low SNR and water absorption factor [20], the channels of 1-3, 105-115, and 150-170 in the Cuprite data set are removed in the experiment. The Indian Pines data set is also corrected by removing the channels 104-108, 150-163, and 220 because they contain high noise level and no useful information [79]. These kinds of channels are also discarded in the Salinas data, consisting of channels 108-112, 154-167, and 224. Some areas of the images in Pavia University and Pavia Center that have no information are discarded before being used. After normalizing the data to [0, 1] range, we added Gaussian noise with $\sigma = 0.1$ to all the data.

Parameter Setting

In addition to involving parameters of the RGB and multispectral image experiment, the hyperspectral experiment needs additional parameters to be set regarding the difference in the number of channels. When the proposed algorithm is applied to hyperspectral images, we provide it with the flexibility to set M as the number of channels to be processed. Choosing the value of M is also a way to find the optimum result based on the correlation among the channels. In other words, it exploits the characteristic of hyperspectral images, that is the high correlation among neighboring channels which are processed at one time. When M is increased, however, it affects the complexity and accuracy of finding the estimated spectral line. In addition, the strength of the disrupting noise definitely affects the complexity as well.

Regarding this characteristic, M becomes an additional parameter setting in

Table 3.2: MPSNR comparison

Method	Cuprite	Indian	Pavia-C	Pavia-U	Salinas
NLM	31.09	33.99	26.25	27.28	37.54
PRI-NL-PCA	31.69	35.61	31.59	32.35	36.88
VBM3D	33.44	36.95	33.10	**33.69**	38.94
VBM4D	32.72	36.35	31.68	32.42	38.53
Ours	**36.03**	**37.84**	**33.17**	32.76	**41.35**
LCNN	34.77	36.62	33.84	33.90	39.24
Ours	**36.33**	**37.84**	**33.89**	**34.34**	**41.44**

(For the comparison with LCNN, the images are cropped so that the size is a multiple of the window size, and our method is simulated under the same condition. Note that according to our experiments, the PSNR of LCNN becomes a little lower without this treatment.)

the beginning of our algorithm. We have tested some values and examined the influence of the parameter M. After some trial-and-error, we adopt $M = 35$ that gives good performance. Outside these values, the algorithm results in lower PSNRs. As for the number of main iteration and half of the window size are adjusted to 3 and 6, respectively.

Evaluation Parameter

To evaluate the results, both qualitative and quantitative assessments are conducted in the experiment. The peak signal-to-noise ratio (PSNR) is calculated for every processed channel, and the mean with respect to the channels is calculated, which is denoted as MPSNR.

Experimental Results of Multispectral Images

To validate the effectiveness of our method, the results are compared to other powerful denoising methods, i.e. video block matching 3-D filtering (VBM3D) [25] and band-by-band nonlocal means (NLM) [15]. In the experiment, the parameters of our method and the competitor methods are set so as to give the best evaluation values. The experiment results are given in Table 3.3. Our method improves the image quality with PSNR reaching 10 dB increase than the noisy image. From the table, we can conclude that for all data, our method successfully achieves

higher PSNR than the nonlocal means method and slightly competes the result of VBM3D. However, the significant superiority of ours compared with VBM3D can be distinguished from visual appearance depicted in Figure 3.9 and 3.10. These images illustrate, (a) original image, (b) noisy image with additional Gaussian noise, (c) result of nonlocal means method, (d) result of VBM3D, and (e) our iterative method with 3 iterations. One can see from these images that VBM3D results in too smooth images compared to the original, whereas our resulting images are closer to the original.

Table 3.3: PNSR result (correspondents to first and second row of each data respectively).

Data	Noisy	NLM	VBM3D	Ours
Kyushu	24.4414	32.8880	34.2872	**34.3601**
Papua	24.4414	31.7794	33.8546	**34.1661**
China	24.4414	32.5792	34.3150	**34.4842**
Washington	24.4414	29.5791	31.5495	**32.0093**

Experimental Results of Hyperspectral Images

The effectiveness of the proposed approach is validated by comparing the results to those of other powerful denoising methods: the nonlocal mean filter (NLM) [15], PRI-NL-PCA [50], the video block matching 3-D filtering (VBM3D) [25] and its extension VBM4D [49]. PRI-NL-PCA is a state-of-the-art PCA-based denoising that exploits sparseness and self-similarity image property. Furthermore, we perform additional comparison with a hybrid approach of VBM3D and LCNN [55], which can improve the performance of the stand-alone VBM3D by exploiting channel correlations. Quantitatively, our proposed method gives higher MPSNR results than those of the other compared methods. As shown in Table 3.2, our method is greatly superior to the VBM4D method and is satisfactorily higher when compared to VBM3D. Except for the Pavia University data, VBM3D achieves a slightly higher MPSNR. As is shown in Figure 3.18 later, however, the contrast is partially more vivid in our result, and our result for the Pavia University is comparable to VBM3D even though the PSNR is lower. Compared to LCNN, our results are also superior for all data items.

(a) Original (PSNR) (b) Noisy (24.44) (c) NLM (32.57)

(d) VBM3D (34.31) (e) Ours (34.48)
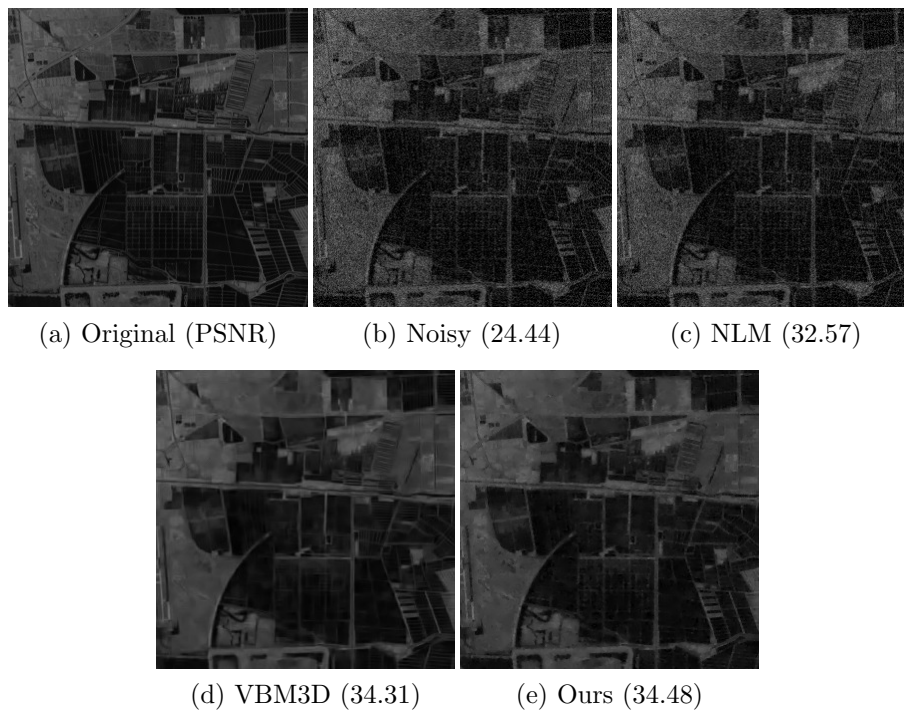
Figure 3.9: Comparison among images from the experiment of China data band 5, with resolution of 30 m

(a) Original (PSNR) (b) Noisy (24.44) (c) NLM (31.77)

(d) VBM3D (33.85) (e) Ours (34.16)
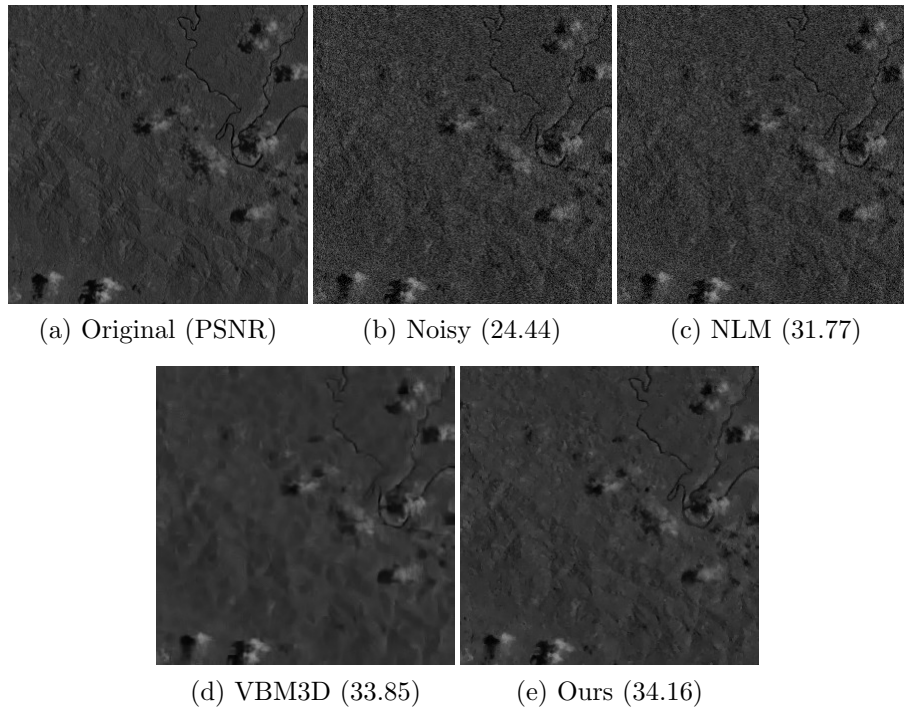
Figure 3.10: Comparison among images from the experiment of Papua data band 6, with resolution of 30 m.

(a) Original (PSNR dB)　　(b) Noisy　　(c) VBM3D (33.89)　　(d) VBM4D (32.80)　　(e) Ours (37.27)
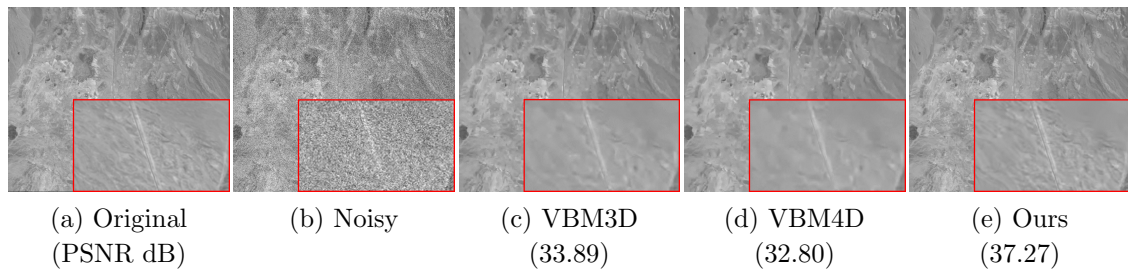
Figure 3.11: Cuprite data, 122nd channel

(a) Original (PSNR dB)　　(b) Noisy　　(c) VBM3D (31.02)　　(d) VBM4D (30.04)　　(e) Ours (34.12)
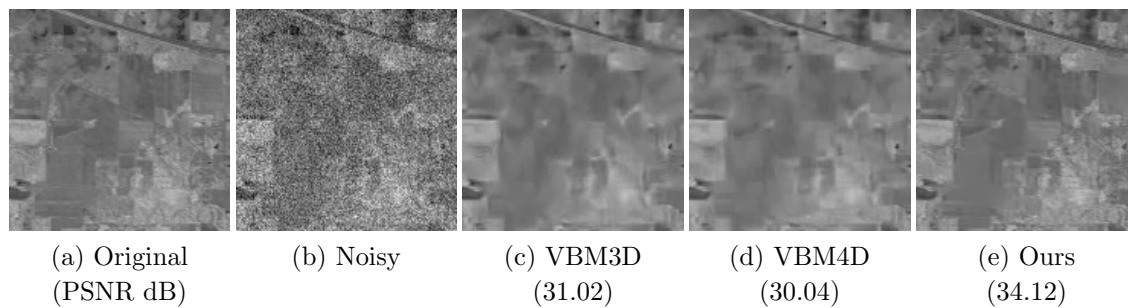
Figure 3.12: Indian data, 46th channel

(a) Original (PSNR dB)　　(b) Noisy　　(c) VBM3D (32.75)　　(d) VBM4D (31.51)　　(e) Ours (32.57)
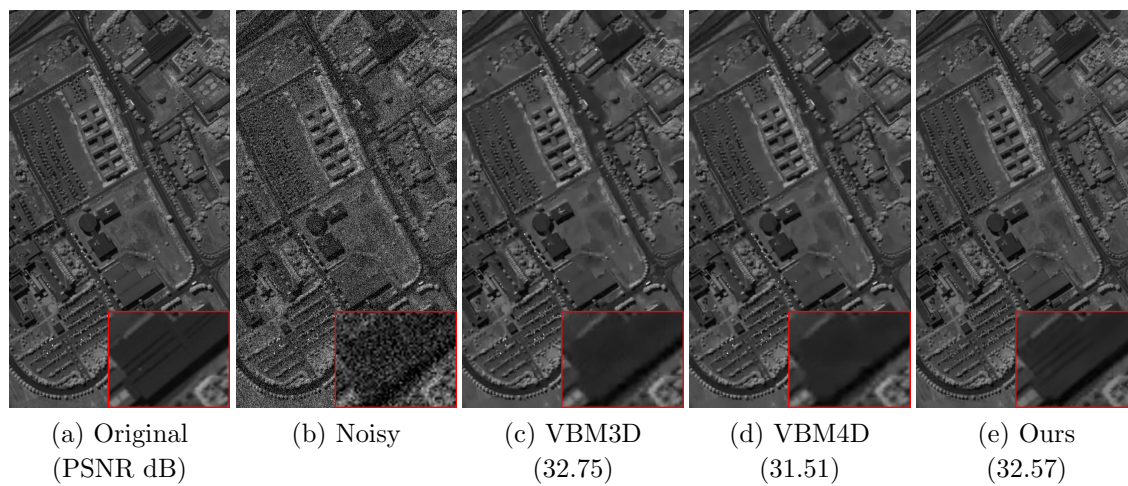
Figure 3.13: Pavia University data, 98th channel

(a) Original (b) Noisy (c) VBM3D (d) VBM4D (e) Ours
(PSNR dB) (32.53) (31.09) (33.19)
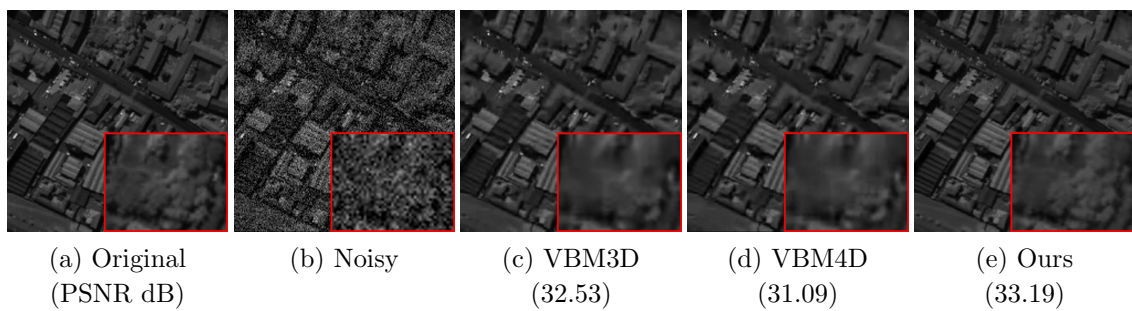
Figure 3.14: Pavia Center, 75th channel

(a) Original (b) Noisy (c) VBM3D (d) VBM4D (e) Ours
(PSNR dB) (34.53) (33.34) (37.97)
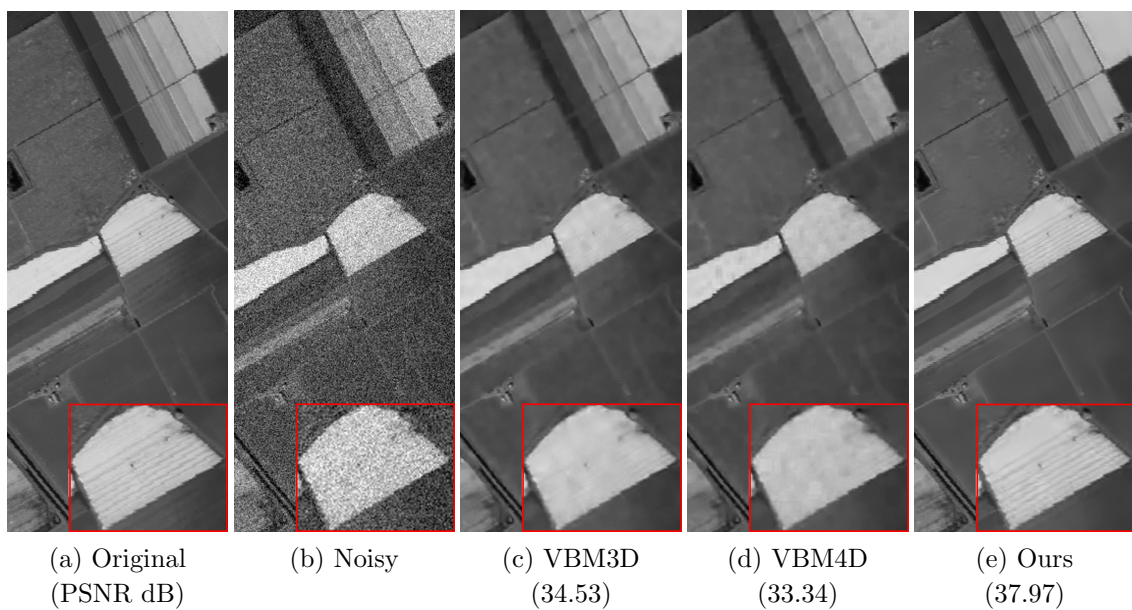
Figure 3.15: Salinas data, 47th channel

(a) Original (b) Noisy (c) LCNN (d) Ours
(PSNR dB) (35.63) (37.50)
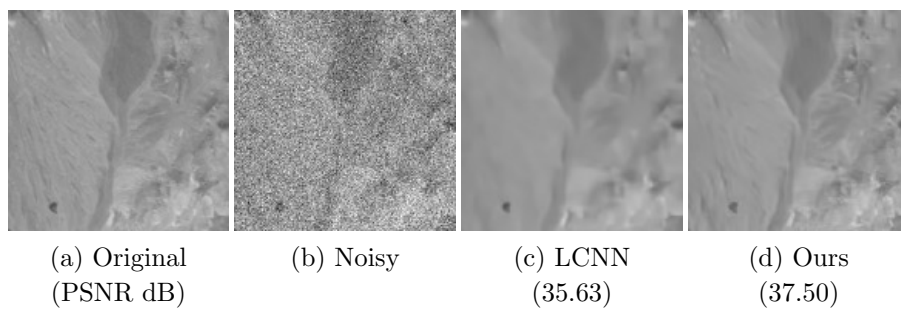
Figure 3.16: Cropped Cuprite data, 122nd channel for hybrid LCNN comparison

41

(a) Original   (b) Noisy   (c) LCNN   (d) Ours
(PSNR dB)   (31.86)   (34.14)
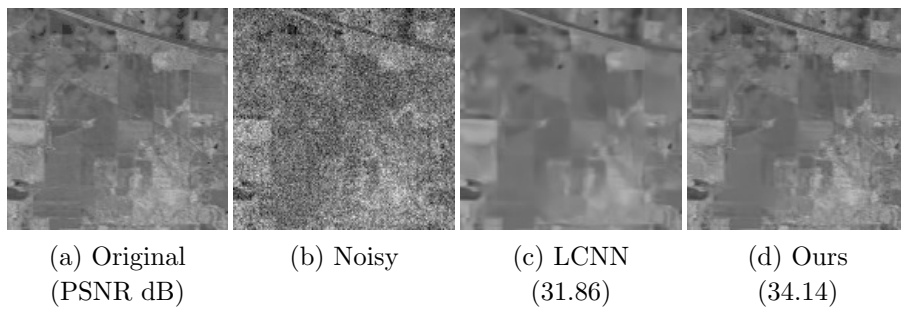
Figure 3.17: Cropped Indian data, 46th channel for hybrid LCNN comparison

(a) Original   (b) Noisy   (c) LCNN   (d) Ours
(PSNR dB)   (34.14)   (35.03)
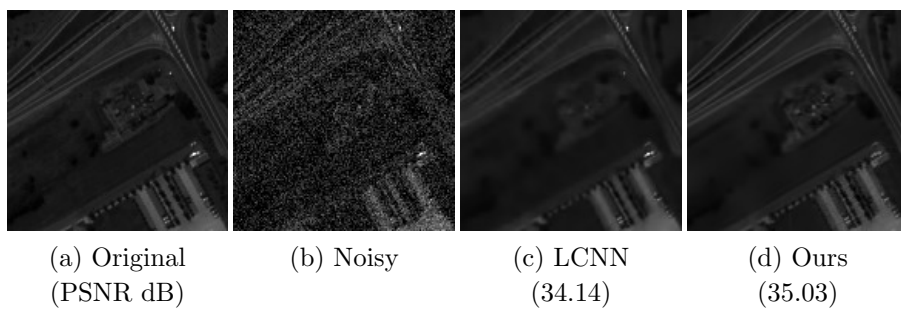
Figure 3.18: Cropped Pavia University data, 35th channel for hybrid LCNN comparison

(a) Original   (b) Noisy   (c) LCNN   (d) Ours
(PSNR dB)   (33.42)   (33.67)
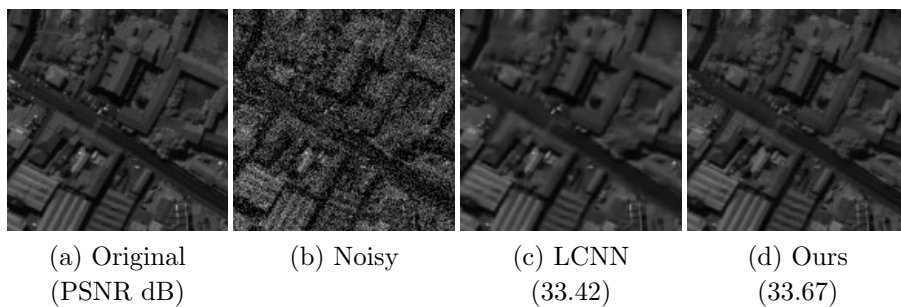
Figure 3.19: Cropped Pavia Center data, 75th channel for hybrid LCNN comparison

(a) Original   (b) Noisy   (c) LCNN   (d) Ours
(PSNR dB)   (35.20)   (37.59)
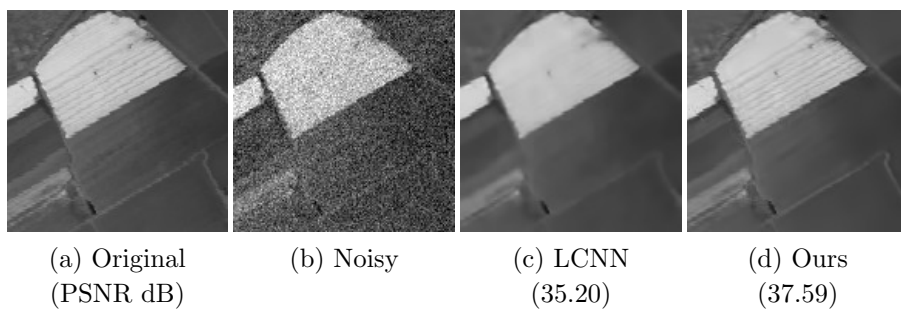
Figure 3.20: Cropped Salinas data, 47th channel for hybrid LCNN comparison

Figures 3.11 to 3.20 show the results for visual qualitative evaluation, where each of the images shows a single channel. For all data, except for Indian Pines, the additional magnified regions of interest are shown, where the differences among resultant images are easily found. On the Cuprite and Pavia Center, the red rectangle shows that our denoising results are not over-smoothed unlike the others' results, but there is a significantly closer similarity to the original image than the other methods. In particular, for VBM3D, which is a more competitive method, the resulting images are over-smoothed, which leaves a noticeable visual difference. Meanwhile, the Salinas and Pavia University images show more significant visual difference. As depicted by the red rectangle, the original image has stripes that can be recognized easier on our resultant images than on those of the other methods.

For the Indian Pines, the overall image has a distinct difference in appearance between ours and the compared images. Both of VBM3D and VBM4D produce images with too smooth areas that results in a salient difference compared to the original image. In addition, they fail to preserve the straight line features in the images. Meanwhile, our method still performs better than the others. It is clear that for perceptual appearance, our resulting image is the most similar to the original one.

Despite the LCNN improvement, Figure 3.16 to 3.20 show that this hybrid method is lower in visual quality than ours. After these experiments, we confirm that the local spectral component decomposition works superiorly in denoising multi-channel images rather than RGB images.

3.4   Conclusion

A new denoising method based on the spectral line has been proposed for the remote sensing field. Hyperspectral image denoising using a spectral line vector field uses the correlation among spectral information in the local region. The vectors are obtained by local spectral component decomposition followed by iterative filtering steps. Filtering the spectral line component and residual component gives significant effects in reducing the noise and smoothing the image. Moreover, the use of local spectral components contributes to achieving better results compared with the result of the stand-alone conventional method. The experiment demonstrated that the proposed method successfully achieved competitive performance compared to other powerful denoising methods. However, the increase in noise power and the number of channels processed affects the complexity of achieving

more accurate spectral line vector estimation. Future work may involve solving this computational complexity.

Chapter 4
Local Abundance Regularizer for Sparse Unmixing

Abstract – Sparse unmixing is widely used for hyperspectral imagery to estimate the optimal fraction (abundance) of materials contained in mixed pixels (endmembers) of a hyperspectral scene, by considering the abundance sparsity. This abundance has a unique property, i.e., high spatial correlation in local regions. This is due to the fact that the endmembers existing in the region are highly correlated. This implies the low-rankness of the abundance in terms of the endmember. From this prior knowledge, it is expected that considering the low-rank local abundance to the sparse unmixing problem improves estimation performance. In this study, we propose an algorithm that exploits the low-rank local abundance by applying the nuclear norm to the abundance matrix for local regions of spatial and abundance domains. In our optimization problem, the local abundance regularizer is collaborated with the $L_{2,1}$ norm and the total variation for sparsity and spatial information, respectively. We conducted experiments for real and simulated hyperspectral data sets assuming with and without the presence of pure pixels. The experiments showed that our algorithm yields competitive results and performs better than the conventional algorithms.

Index Terms – Sparse unmixing, hyperspectral, local abundance, nuclear norm.

4.1   Introduction

4.1.1   Hyperspectral Unmixing

Hyperspectral images come from the need to extract more detailed information than what multispectral images can provide, which enable pixel-constituent-level analysis of remote-sensing imagery. Hyperspectral images have better spectral

resolution than multispectral images due to their large number of narrow and contiguous spectral bands [63]. However, the detailed information provided by sensors faces a trade-off in which the sensors capture distinct materials on the Earth's surface mixed in one pixel. This is affected by one of the following factors [43,67,75]. The first factor is due to the low spatial resolution of the sensors; two or more separate materials occupy the same pixel. The other factor occurs when the sensors capture some distinct substances that have merged into a homogeneous mixture on the Earth's surface. This condition leads to a compelling solution, i.e., spectral unmixing.

The procedure of spectral unmixing works by decomposing the measured hyperspectral data into a collection of spectral signatures (spectral library) and a set of corresponding fractions (abundances) that represent the proportion of each spectral signature contained in the pixels [36, 43, 44, 64]. The spectral signatures that exist in the mixed pixels are called endmembers. In general, endmembers correspond to familiar macroscopic objects in a scene, such as water, metal, and vegetation, as well as constituents of intimate mixtures in microscopic scale.

Unmixing becomes a challenging, ill-posed inverse problem because of model inaccuracies, observation noise, environmental conditions, endmember variability, and data set size [10]. In the early research [43], the end-to-end unmixing problem is a sequence of three consecutive procedures.

1. **Dimension reduction:** Reduce the dimension of the data in the scene. This step is optional and is only invoked by some algorithms to reduce the computational load of subsequent steps.

2. **Endmember determination:** Estimate the set of distinct spectra (endmembers) that constitute the mixed pixels in the scene.

3. **Inversion:** Estimate the fractional abundances of each mixed pixel from its spectrum and the endmember spectra.

Many hyperspectral unmixing models have been investigated to find robust, stable, tractable, and accurate unmixing algorithms. As one of the results, many approaches enable the endmember determination and inversion steps implemented simultaneously. In general, there are three approaches that had been discussed in [11], i.e., geometrical, statistical, and sparse regression approaches. 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 endmember and abundance parameters. Sparse regression approaches, which formulate unmixing as a linear sparse regression problem, rely on the existence of spectral libraries that usually acquired in laboratory. The basic idea is exploiting a fact that the number of endmembers participating in a mixed pixel is usually very small compared with the (ever-growing) dimensionality and availability of spectral libraries. In the development, researchers added contextual informations on the top of the sparse unmixing problem to guide the endmember extraction and the abundance estimation steps.

4.1.2   Linear versus nonlinear spectral unmixing

Linear spectral unmixing is the most standard technique for spectral unmixing, besides the nonlinear one. It is based on linear mixture model (LMM) which is infers a set of endmembers (generally from the hyperspectral image scene), and their abundance fractions. The LMM assumes that the spectra collected by the sensor can be expressed in the form of a linear combination of endmembers, weighted by their corresponding abundances. Because each observed spectral signal is the result of an actual mixing process, it is expected that the driving abundances satisfy two constraints, i.e., they should be non-negative, and the sum of abundances for a given pixel should be unity. Although the linear unmixing has practical advantages, such as ease of implementation and flexibility in different applications, nonlinear unmixing describes mixed spectra in physical or statistical sense, by assuming that part of the source radiation is multiply scattered before being collected at the sensor. The distinction between linear and nonlinear unmixing has been widely studied in recent years.

Figure 4.1 illustrates the difference between linear and nonlinear model. The LMM holds when the mixing scale is macroscopic and the incident light interacts with just one material, as is the case in checkerboard type scenes. In this case, the mixing occurs within the instrument itself, due to the fact that the resolution of the instrument is not fine enough. Although the light from the materials is almost completely separated, it is mixed within the measuring instrument [11].

On the other side, nonlinear mixing usually occurs due to physical interactions between the light scattered by multiple materials in the scene. These interactions can be at a classical level or at a microscopic 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 captured by the sensor. Microscopic mixing occurs when two materials are homogeneously mixed. In this case, the interac-
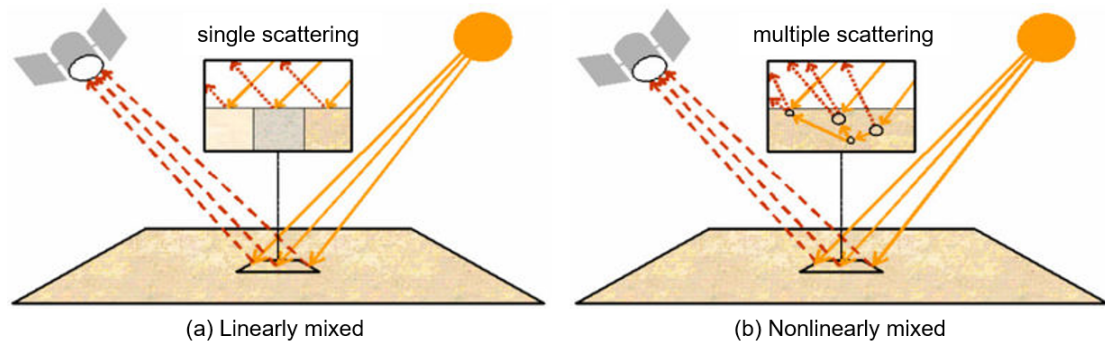
Figure 4.1: Illustration of linear and nonlinear mixture models in remotely sensed hyperspectral imaging [4].

tions consist of photons emitted from molecules of one material are absorbed by molecules of another material, which may in turn emit more photons [11].

In this study, we focus on linear spectral unmixing due to its generality and simplicity of implementation, not without acknowledging that nonlinear unmixing may provide better results in terms of spectral characterization in many applications. In addition, LMM is an acceptable approximation of the light scattering mechanisms in many real scenarios. Spanning back at least 30 years, LMM becomes the basis of most of unmixing models and algorithms.

4.1.3 Sparse regression-based unmixing

The spectral unmixing problem has recently been approached in a semi-supervised fashion, by assuming that the observed image signatures can be expressed in the form of linear combinations of a number of pure spectral signatures known in advance, i.e., spectra collected on the ground by a field spectro-radiometer (library). This approach can avoid the endmember determination step. Unmixing then amounts to finding the optimal subset of signatures in a (potentially very large) spectral library that can best model each mixed pixel in the scene.

Techniques which assume the presence of pure pixels (pixels contained only one material) in the data have to deal with the fact that the most common situation is that the hyperspectral image has no pure pixels at all. This situation is due to spatial resolution issues and the fact that the mixture problem happens at every scale, even at the intimate mixture level. While endmember determination techniques that do not assume the presence of pure pixels circumvent this problem,

the endmember signatures obtained are virtual and often lack physical meaning.

Conversely, unmixing with spectral libraries allows us to avoid these issues since the library signatures correspond to pure endmember spectra measured on the field, and thus retaining the physical meaning. Further, unmixing with spectral libraries also allows us to avoid the estimation of the number of endmembers in the scene, which is also a very challenging step. As a result, sparse unmixing amounts to finding the optimal subset of signatures in a spectral library that can best model each mixed pixel in the scene.

In practice, spectral unmixing based on linear model is a combinatorial problem which calls for efficient linear sparse regression techniques based on sparsity-inducing regularizers, since the number of endmembers participating in a mixed pixel is usually very small compared with the ever-growing dimensionality and availability of spectral libraries. By this approach, it is expected that it is not necessary to estimate a priori for the number of endmembers in the scene.

4.1.4 Proposed Idea

In this study, we develop a sparse regression-based unmixing algorithm, which is called joint local abundance sparse unmixing (J-LASU), in which we propose the local abundance regularizer and implanted it to the sparse unmixing problem using the nuclear norm for 3D local regions and evaluated the effect. We use the 3D local block sliding through the three dimensions of the abundance maps and imposed the nuclear norm to promote the low-rank structure of the local abundance cube. The difference with the 3-tensors nuclear norm [33] is that our proposed scheme takes the local nuclear norm of the third dimension in addition to the first two dimensions. In other words, the sliding block also moves in a particular step size along the third dimension. We preserve the use of the total variation (TV) regularizer for spatial consideration. The proposed algorithm was tested on simulated data as well as real hyperspectral data and compared with other sparse unmixing algorithms, i.e., CLSUnSAL, SUnSAL-TV, and ADSpLRU. The major contribution of this study is imposing our local abundance regularizer to a hybrid of state-of-the-art unmixing techniques that take into account collaborative sparsity and spatial difference. We also apply the proposed J-LASU to several scenes with and without pure pixels.

4.1.5   Chapter Structure

The structure of remaining chapters are as follows. Section 4.2 describes some mathematical preliminaries necessary for this topic. In Section 4.3, we discuss the problem formulation of hyperspectral unmixing as an introduction to the problem formulation of our proposed algorithm. In Section 4.4, we describe the proposed J-LASU algorithm starting with convincing evidence of the proposed concept. In Section 4.5, we describe the experiment and analysis. In Section 4.6, we discuss the results and findings. Finally, we conclude the research in Section 4.7.

Variables and notation: Column vectors are represented as boldface lowercase letters, e.g., $\mathbf{y}$, whereas matrices are represented as boldface uppercase letters, e.g., $\mathbf{Y}$. The following variables are frequently used in this chapter:

- $\mathbf{Y}$ is the hyperspectral data,

- $\mathbf{A}$ is the spectral library,

- $\mathbf{X}$ is the abundance matrix,

- $\hat{\mathbf{X}}$ is the 3D abundance data,

- m is the number of spectral signatures,

- l is the number of spectral bands,

- n is the number of pixels in $\hat{\mathbf{X}}$,

- n_c is the number of columns in $\hat{\mathbf{X}}$,

- n_r is the number of rows in $\hat{\mathbf{X}}$,

- B is the number of all local blocks in $\hat{\mathbf{X}}$,

- N is the number of pixels in each local abundance matrix,

- $\hat{\mathbf{X}}_b$ is the b-th local block,

- $\mathbf{H}_{\hat{\mathbf{x}}_b}$ is the b-th local abundance matrix.

4.2   Basic Theory

4.2.1   Convex Optimization

An optimization problem is defined in the form

$$
\begin{aligned}
&\min && f_0(\mathbf{x}) \\
&\text{s.t.} && f_i(\mathbf{x}) \le b_i, i = 1, \dots, m.
\end{aligned}
\tag{4.1}
$$

Here the vector $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_n)$ is the *optimization variable* of the problem, the function $f_0 : \mathbb{R}^n \mapsto \mathbb{R}$ is the *objective function*, the functions $f_i : \mathbb{R}^n \mapsto \mathbb{R}$, $i = 1, \dots, m$, are the *constraint functions* in an inequality form, and the constants $b_1, \dots, b_m$ are the limits, or bounds, for the constraints. A vector $\mathbf{x}^*$ is called *optimal solution* of the problem, if it has the smallest objective value among all vectors that satisfy the constraints: for any $\mathbf{z}$ with $f_1(\mathbf{z}) \le b_1, \dots, f_m(\mathbf{z}) \le b_m$, we have $f_0(\mathbf{z}) \ge f_0(\mathbf{x}^*)$.

Generally, optimization problems are considered into families or classes that characterized by particular forms of the objective and constraint functions. As an important example, the optimization problem in Equation (4.1) is called a linear program if the objective and constraint functions $f_0, \dots, f_m$ are linear. If the optimization problem is not linear, it is called a nonlinear program.

A convex optimization problem is one class of optimization problems in which the objective and constraint functions are convex, which means they satisfy the inequality

$$
f_i(\lambda \mathbf{x} + \beta \mathbf{y}) \le \lambda f_i(\mathbf{x}) + \beta f_i(\mathbf{y})
\tag{4.2}
$$

for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and all $\lambda, \beta \in \mathbb{R}$ with $\lambda + \beta = 1, \lambda \ge 0, \beta \ge 0$.

4.2.2   ADMM

The Alternating Direction Method of Multipliers (ADMM) is an algorithm for solving the following convex optimization problem

$$
\arg\min_{\mathbf{x}, \mathbf{y}} \quad f(\mathbf{x}) + g(\mathbf{y}) \qquad \text{s.t.} \qquad \mathbf{y} = \mathbf{H}\mathbf{x}.
\tag{4.3}
$$

where $\mathbf{x} \in \mathbb{R}^n, \mathbf{y} \in \mathbb{R}^{n^T}, \mathbf{H} \in \mathbb{R}^{n \times n^T}$, f and g are convex functions. The augmented Lagrangian for this problem is

$$
L(\mathbf{x}, \mathbf{y}, \mathbf{z}) = f(\mathbf{x}) + g(\mathbf{y}) + (\mu/2)\|\mathbf{H}\mathbf{x} - \mathbf{y} + \mathbf{z}\|_2^2.
\tag{4.4}
$$

where $\mathbf{z}$ is a scaled dual variable associated with the constraint, and $\mu > 0$ is a penalty parameter

In each iteration of ADMM, the procedure performs alternating minimization of the augmented Lagrangian over $\mathbf{x}$ and $\mathbf{y}$. At iteration k, the ADMM algoritm is carried out with the following steps

$$\mathbf{x}^{(k+1)} = \arg\min_{\mathbf{x}} \quad f(\mathbf{x}) + (\mu/2)\|\mathbf{H}\mathbf{x} - \mathbf{y}^{(k)} + \mathbf{z}^{(k)}\|_2^2, \tag{4.5}$$

$$\mathbf{y}^{(k+1)} = \text{prox}_{g/\mu}(\mathbf{H}\mathbf{x}^{(k+1)} + \mathbf{z}^{(k)}), \tag{4.6}$$

$$\mathbf{z}^{(k+1)} = \mathbf{z}^{(k)} + (\mathbf{H}\mathbf{x}^{(k+1)} - \mathbf{y}^{(k+1)}), \tag{4.7}$$

where *prox* denotes the proximal operator [24, 51],

4.2.3   Proximal Operator

The proximal operator is the analytic solution to the approximation problem, and defined as follows

$$\text{prox}_{\gamma h}(\bar{\mathbf{x}}) = \arg\min_{\mathbf{v} \in \mathbb{R}^n} \quad h(\mathbf{v}) + \frac{1}{2\gamma}\|\mathbf{v} - \bar{\mathbf{x}}\|_2^2, \tag{4.8}$$

where the uniqueness of the minimizer is guaranteed by the strict convexity of $h(\,\cdot\,) + \frac{1}{2\gamma}\|(\,\cdot\,) - \bar{\mathbf{x}}\|_2^2$.

4.2.4   Sparse Linear Regression

Suppose we have a pixel $\mathbf{y} \in \mathbb{R}^l$, i.e., a vector response for each of l spectral bands, and a collection of spectral signatures of materials $\mathbf{A} = (\mathbf{A}_1, \ldots, \mathbf{A}_m) \in \mathbb{R}^{l \times m}$ which is full rank. We want to use a linear model $\mathbf{y} \approx \mathbf{A}\mathbf{x}$, where $\mathbf{x} \in \mathbb{R}^m$. If $l > m$, this represents the classical linear regression problem defined as follows

$$\min_{\mathbf{x}} \quad \|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2. \tag{4.9}$$

The solution to this problem is well-defined and easy to find if $l > m$. However, in some applications $l \ll m$, and thus many of the extracted features $\mathbf{A}_1, \ldots, \mathbf{A}_m$ could be irrelevant. Therefore, it is expected to find a model $\mathbf{x}$ with many zero coefficients, as shown in Figure 4.2. For instance, it might reasonable to suppose
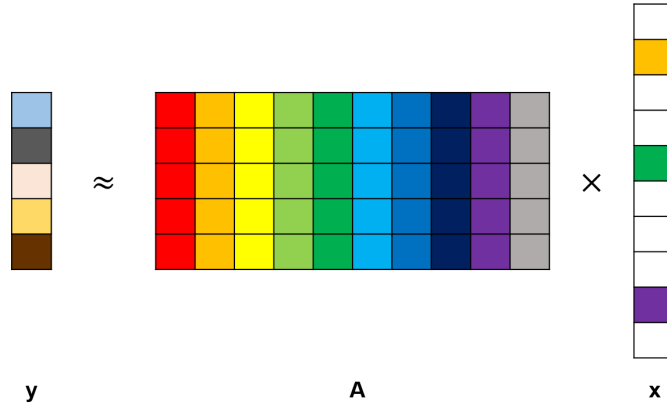
Figure 4.2: Illustration of linear regression mode $\mathbf{y} \approx \mathbf{Ax}$, with $l \ll m$ and a sparse $\mathbf{x}$.

that the pixel $\mathbf{y}$ can be expressed as a linear combination of the materials' spectral signatures in $\mathbf{A}$, but we expect that most materials will be unimportant to this combination, which means that they do not exist in the pixel. Thus, most components of $\mathbf{x}$ will be zero. In other words, $\mathbf{x}$ is *sparse*.

A very simple and intuitive way for incorporating sparsity into the linear problem simply involves the number of nonzero elements in $\mathbf{x}$ as a prior, since the vector is sparse if there are few nonzeros among the possibe elements in $\mathbf{x}$. Thus, the L_0 norm is introduced to promote sparsity and defined as

$$\|\mathbf{x}\|_0 = \#i : x_i \neq 0. \tag{4.10}$$

Thus if $\|\mathbf{x}\|_0 \ll m$, $\mathbf{x}$ is sparse, and the sparse linear problem is defined as

$$\min_{\mathbf{x}} \quad \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \mathbf{y} = \mathbf{Ax}. \tag{4.11}$$

4.2.5   Nuclear Norm

When a matrix is formed by linear-dependence vectors which are highly correlated, the matrix is either low rank, or it can be well-approximated by a low-rank matrix. The low-rank matrix estimation techniques have recently emerged as powerful estimation tools, which mainly use nuclear norm for promoting low-rankness of a matrix. The nuclear norm (also known as the trace norm) is defined as the sum of all singular values of a matrix,

$$\| \cdot \|_* : \mathbb{R}^{n_v \times n_h} \mapsto \mathbb{R}_+ : \mathbf{X} \mapsto \sum_{i=1}^{r} \sigma_i(\mathbf{X}), \tag{4.12}$$

where $0 \le r \le min(n_v, n_h)$ is the rank of a given matrix, and σ_i denotes the i-th singular value of the matrix .

As for the proximity operator of the nuclear norm is as follows

$$\text{prox}_{\gamma \| \cdot \|_*}(\mathbf{X}) = \mathbf{U_X} \Sigma_{\mathbf{X}}^{\gamma} \mathbf{V_X^T}, \tag{4.13}$$

where $\mathbf{U_X} \Sigma_{\mathbf{X}}^{\gamma} \mathbf{V_X^T}$ is the singular value decomposition of $\mathbf{X}$; $\mathbf{U_X} \in \mathbb{R}^{n_v \times n_v}$ and $\mathbf{V_X} \in \mathbb{R}^{n_h \times n_h}$ are unitary matrices, and $\Sigma_{\mathbf{X}}^{\gamma}$ is the diagonal matrix containing the singular values shrunk by $\gamma > 0$ as its main diagonal entries, i.e., diag($max\{\sigma_1 \mathbf{X} - \gamma, 0\}, \ldots, max\{\sigma_r \mathbf{X} - \gamma, 0\}), 0, \ldots, 0)$.

4.2.6   Total Variation

Other convex priors are smoothness priors, and the most popular one is the so-called Total Variation (TV) prior. TV is defined as total magnitude of differences of neighboring entries of a signal vector, which has frequently been used in image restoration. Its multi-channel versions, higher-order generalizations, and non-local extensions have also been studied. One of the multi-channel versions is in the use for hyperspectral unmixing tasks.

The TV works using a linear gradient operator of the given matrix $\mathbf{X} \in \mathbb{R}^{n \times n}$. if $u \in \mathbf{X}$, the gradient $\nabla \mathbf{u}$ is a vector given by

$$(\nabla \mathbf{u})_{i,j} = ((\nabla \mathbf{u})_{i,j}^v, (\nabla \mathbf{u})_{i,j}^h), \tag{4.14}$$

with

$$(\nabla \mathbf{u})_{i,j}^v = \begin{cases} \mathbf{u}_{i+1,j} - \mathbf{u}_{i,j} & \text{if } i < n, \\ 0 & \text{if } i = n, \end{cases} \tag{4.15}$$

$$(\nabla \mathbf{u})_{i,j}^h = \begin{cases} \mathbf{u}_{i,j+1} - \mathbf{u}_{i,j} & \text{if } j < n, \\ 0 & \text{if } j = n, \end{cases} \tag{4.16}$$

for $i, j = 1, \ldots, n$.

Then, the TV of u is defined by

$$TV(\mathbf{u}) = \sum_{\{1 \le i,j \le n\}} |(\nabla \mathbf{u})_{i,j}|, \tag{4.17}$$

This definition is isotropic and not differentiable, as

$$|(\nabla \mathbf{u})_{i,j}| = \sqrt{\left((\nabla \mathbf{u})_{i,j}^v\right)^2 + \left((\nabla \mathbf{u})_{i,j}^h\right)^2}. \tag{4.18}$$

Another variation that is commonly used due to the easiness in minimization is the anisotropic version, defined as follows

$$TV_{aniso}(\mathbf{u}) = \sum_{\{1 \leq i,j \leq n\}} \|(\nabla \mathbf{u})_{i,j}\|_1 = \sum_{\{1 \leq i,j \leq n\}} \left((\nabla \mathbf{u})_{i,j}^v + (\nabla \mathbf{u})_{i,j}^h\right). \tag{4.19}$$

The anisotropic TV is commonly used in unmixing problems due to the property of L_1 norm. As a regularization, L_1 norm produces many coefficients with zero values or very small values (sparse). This sparse property allows it to be effectively used along with sparse algorithms, that makes the calculation more computationally efficient.

4.3 Problem Formulation

4.3.1 Sparse Unmixing

Let $\mathbf{Y} \in \mathbb{R}^{l \times n}$ be the observed hyperspectral data, where l is the number of bands, and n is the number of pixels. The LMM for a hyperspectral image is based on the assumption that each pixel $\mathbf{y} \in \mathbb{R}^l$ in any given spectral band is a linear combination of m spectral signatures in the spectral library $\mathbf{A} \in \mathbb{R}^{l \times m}$, that is,

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e} \tag{4.20}$$

where $\mathbf{x} \in \mathbb{R}^m$ is the abundance vector, and $\mathbf{e} \in \mathbb{R}^l$ is the vector of noise and model error.

With sparse unmixing, it is assumed that the abundance vector $\mathbf{x}$ is sparse because the number of endmembers contained in a pixel is much lower than the number of spectral signatures in the library, which implies the vector $\mathbf{x}$ contains many intensities of zero. Figure 4.3 illustrates the LMM and sparse unmixing. Considering the ground truth, $\mathbf{x}$ has a constraint that needs to be imposed to the sparse unmixing model, i.e., the value of $\mathbf{x}$ can never be negative which is called the abundance nonnegativity constraint (ANC). The sparse unmixing problem based on the LMM for each mixed pixel can be formulated as

$$\min_{\mathbf{x}} \quad \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \leq \delta, \quad \mathbf{x} \geq 0 \tag{4.21}$$
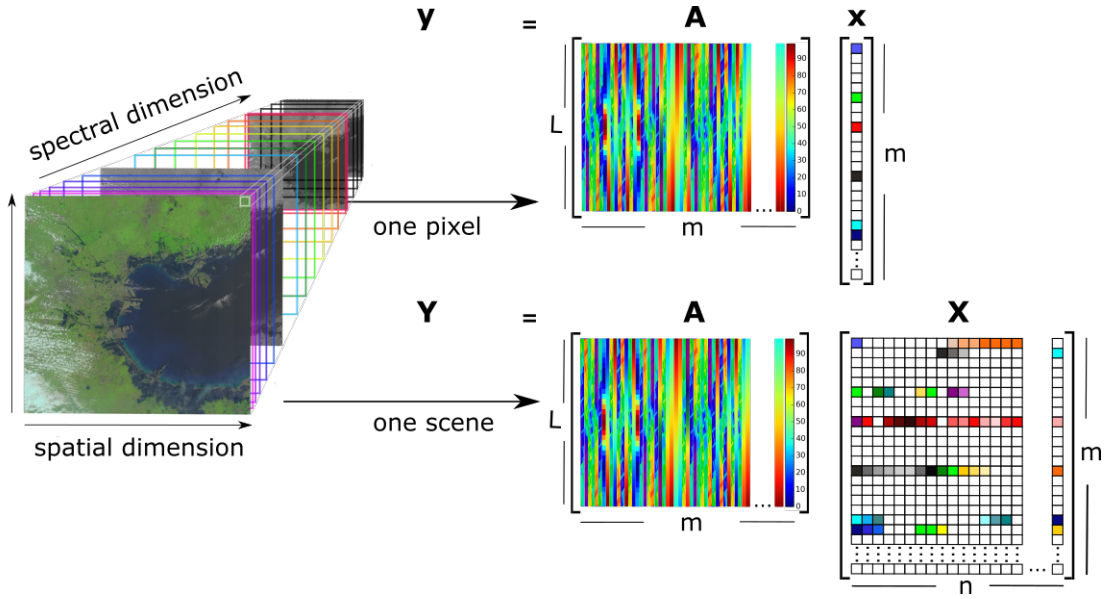
Figure 4.3: Illustration of hyperspectral image and sparse unmixing for pixel (**top**) and image (**bottom**).

where $\|\mathbf{x}\|_0$ denotes the number of nonzero elements in $\mathbf{x} \in \mathbb{R}^m$, and δ is the error tolerance value determined from the noise and model error. The nonconvexity of the L_0 term induces an NP-hard problem (meaning that the problem is combinatorial and very complex to solve in a straightforward way); however, it has been proven that a nonconvex optimization problem can be relaxed to a convex one by replacing L_0 with L_1 [18,81]. Greedy algorithms, such as the orthogonal basis pursuit (OMP) [57], and basis pursuit (BP) [21] are two alternative approaches to compute the sparsest solution. The basis pursuit replaces the L_0 norm in (P0) with the L_1 norm. Thus, the problem can be written as

$$\min_{\mathbf{x}} \quad \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \leq \delta, \quad \mathbf{x} \geq 0 \tag{4.22}$$

Applying this formula to the whole image, we estimate the abundance matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ for all the pixels in the hyperspectral data $\mathbf{Y}$ using the respective Lagrangian function as

$$\min_{\mathbf{X}} \quad \tfrac{1}{2}\|\mathbf{A}\mathbf{X} - \mathbf{Y}\|_F^2 + \lambda\|\mathbf{X}\|_1 \quad \text{s.t.} \quad \mathbf{X} \geq 0 \tag{4.23}$$

where $\| \cdot \|_F$ denotes the Frobenius norm of a matrix, and λ is the sparsity regularizer. This problem can be solved through optimization by using alternating direction method of multipliers (ADMM).

4.3.2   Spatial Regularization

Despite taking into account sparsity, SUnSAL ignores spatial correlation. In SUnSAL-TV, the relationship between each pixel vector and its adjacent pixel vectors is taken into account. The regularizer is defined in [38] as

$$\mathrm{TV}(\mathbf{X}) = \sum_{\{i,j \in \kappa\}} \|\mathbf{x}_i - \mathbf{x}_j\|_1 \tag{4.24}$$

which is the anisotropic TV with κ denoting the set of horizontal and vertical neighbors in $\mathbf{X}$.

Adding the TV regularizer to the problem in Equation (4.23) gives the optimization problem

$$\min_{\mathbf{X}} \quad \frac{1}{2}\|\mathbf{AX} - \mathbf{Y}\|_F^2 + \lambda\|\mathbf{X}\|_1 + \lambda_{TV}\mathrm{TV}(\mathbf{X}) \quad \text{s.t.} \quad \mathbf{X} \geq 0. \tag{4.25}$$

4.4   Proposed Algorithm

4.4.1   Local Abundance Correlation

Hyperspectral data $\mathbf{Y} \in \mathbb{R}^{l \times n}$ have linearity in their spectral [62] and spatial [61] domains. Qu et al. [61] provided prior knowledge that the high spatial correlation of the hyperspectral data, implies linearly dependent abundance vectors in the abundance matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$. The high correlation also holds among the pixel members of a local region due to the spatial similarity. In a physical sense, the pixels in such regions contain the same materials, either in the same or different fractions. Hence, the abundance matrix of the region can be estimated by the low-rank property [35, 61].

However, the success of sparse regression techniques is affected by the low sparsity as well as low correlation between spectral signatures in the library [38]. The former is represented by the number of endmembers existing in the scene, namely, the degree of sparsity [37]. The latter can be defined by an indicator representing the difficulty to accurately solve a linear system equation i.e., mutual coherence. The mutual coherence is defined as the largest cosine among endmembers in the library. In the hyperspectral case, the degree of sparsity is often low, but the mutual coherence is close to one. In fact, higher mutual coherence decreases the quality of the solution [39].

57
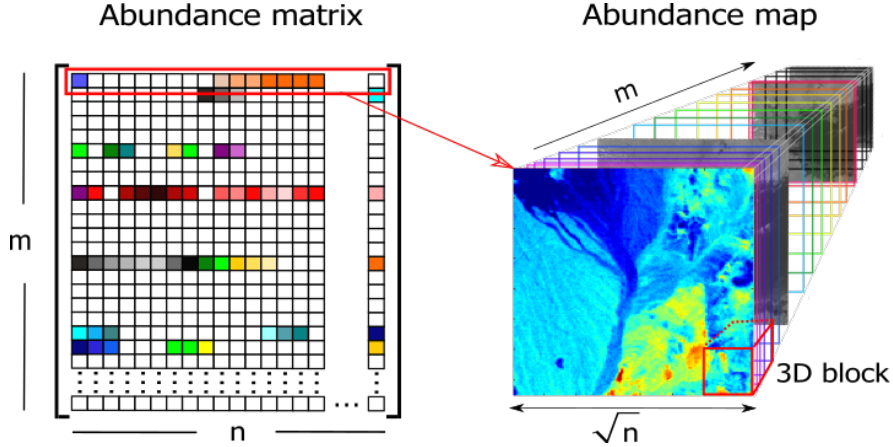
Figure 4.4: Illustration of endmember (m) direction in abundance dimension. 3D local block moves through pixels (n) as well as m direction of abundance maps.

To overcome the high mutual coherence as well as consider the low-rank property of the abundance, we exploit the high correlation of library's spectral signatures by using our LA regularizer. In our experiment with simulated data, we confirmed the idea by observing the linearity of the data distribution in abundance domain by taking the local maximum singular value of the true abundance matrix for each local block (a block refers to the three dimensions (3D), in which the third dimension has a local coverage in the endmember direction). We found that there is one value that dominates others (the ratio is close to one) in each local block. On the other hand, the value will be less dominant as the region becomes the whole matrix (nonlocal). This implies that the linearity in abundance domain is satisfied for the abundance matrix with the local point of view. Thus, we introduce our LA regularizer using the nuclear norm for the local blocks. Instead of the image, our algorithm uses the nuclear norm to the abundance matrix that constitutes the image. Another difference is that our local block slides through all dimensions, i.e., the two spatial dimensions and the endmember direction in the abundance dimension. Figure 4.4 illustrates the endmember direction. The block moves within the abundance maps of the 3D abundance cube.

In addition, we guarantee high correlation by selecting endmembers from the United States Geological Survey (USGS) library to form the spectral library $\mathbf{A}$ based on the SA. The USGS library is a collection of the measured spectral signatures of hundreds of materials and used as references for material identification in hyperspectral images. We can find the most similar signatures to each endmem-

ber of the simulated data by calculating the SA, besides the mutual coherence. This parameter represents the absolute value of spectral correlation [60]. The value ranges between 0–90 degrees. The lower the SA value, the more similar the compared signature vectors are. In the simulated-data experiment, we adjust the SA as one of our parameter settings.

4.4.2   Collaborative Sparsity Regularization

A hyperspectral image always contains a small number of endmembers. This means that, if the fractional abundances of the spectral library signatures are collected in a matrix with the number of columns equal to the number of pixels, there should be only a few lines with nonzero entries. In other words, the nonzero abundance lines should appear in a few distinct lines, which implies sparsity along the pixels of a hyperspectral image in terms of the sub-pixel information that they convey. Figure 4.5 illustrates this sparsity.

The theoretical background of this regularizer is supported by the results shown in [30], in which the authors exploit exactly the $L_{2,1}$ norm optimization problem for recovering jointly sparse multichannel signals from incomplete measurements. A valuable theoretical result of [30] is the proof of the superiority that multichannel sparse recovery has over the single channel methods, as the probability of recovery failure decays exponentially in the number of channels. In other words, sparse methods have more chances to succeed when the number of acquisition channels increases, which is extremely important for the sparse unmixing applications, as the number of spectral bands is often in the order of hundreds or even thousands.

In practice, the abundance matrix $\mathbf{X}$ has only a few endmembers (rows) with nonzero entries. Simultaneously, all the column entries of $\mathbf{X}$ share the same active set of endmembers. In other words, $\mathbf{X}$ is sparse among the rows while dense among the columns. To implement this prior, $L_{2,1}$ norm is used instead of L_1. It takes the sum of the L_2 norm of the abundance entries to promote the collaborative sparsity of the abundance matrix.

$$\|\mathbf{X}\|_{2,1} = \sum_{i=1}^{m} \|\mathbf{x}_i\|_2 \tag{4.26}$$

where $\mathbf{x}_i$ represents the i-th row of $\mathbf{X}$.
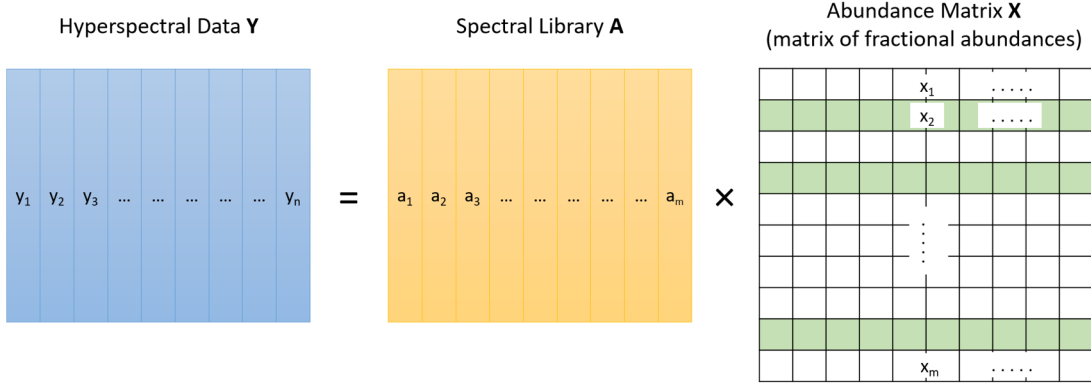
Figure 4.5: Graphical illustration of the performance of the collaborative regularizer. Active members of the considered spectral library **A** are represented in green color, and non-active members of the considered spectral library **A** are represented in white color.

4.4.3 Local Abundance Regularizer

The abundance matrix $\mathbf{X}$ is transformed to a cube to describe our local block formulation. First, let $\hat{\mathbf{X}} \in \mathbb{R}^{n_r \times n_c \times m}$ be the abundance data in 3D form, where m is the number of abundance matrices of the endmembers, n_c and n_r are the numbers of columns and rows, respectively, that satisfy $n = n_c \times n_r$, where n is the number of pixels in each abundance matrix. Then, for each abundance matrix $\hat{\mathbf{X}}_i \in \mathbb{R}^{n_r \times n_c}$ ($i = 1, \dots, m$), stacking the column on top of one another gives $\hat{\mathbf{x}}_i \in \mathbb{R}^n$, the vectorized form of the matrix.

In local regions, let $\hat{\mathbf{X}}_b \in \mathbb{R}^{n_b \times n_b \times m_b}$ denote the b-th local block, where $b = 1, \dots, B$. The B is the number of all local blocks in $\hat{\mathbf{X}}$. This local region formation is not as easily as described by using a sliding window on the abundance matrix $\mathbf{X}$. A sliding window on $\mathbf{X}$ will give different formation compared to our proposed 3D local block, as illustrated in Figure 4.6. In the example, the number of pixels is 12 (3 rows and 4 columns) and the number of endmembers is M. The size of the local block is [2 2 2]. From the figure, we can see that the entries of the sliding window in $\mathbf{X}$ and the proposed local block in cube version are different.

Then, for each abundance of each local block $\hat{\mathbf{X}}_{j,b} \in \mathbb{R}^{n_b \times n_b}$ ($j = 1, \dots, m_b$), we vectorize it into $\hat{\mathbf{x}}_{j,b} \in \mathbb{R}^N$, where N is the number of pixels in each local abundance matrix that satisfies $N = n_b \times n_b$, and j is the index of local abundance matrices. Figure 4.7 illustrates the procedure. With this in mind, we introduce the local
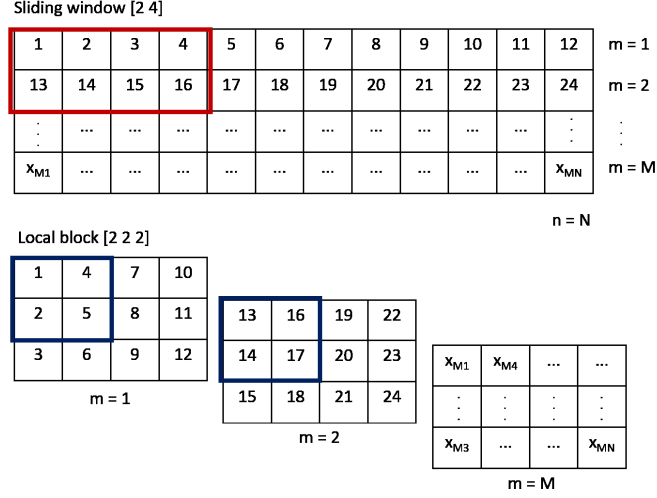
Figure 4.6: The difference between sliding window on an abundance matrix and the proposed local block

abundance matrix w. r. t the b-th block

$$\mathbf{H}_{\hat{\mathbf{x}}_b} = (\hat{\mathbf{x}}_{1,b}, \dots, \hat{\mathbf{x}}_{m_b,b}) \in \mathbb{R}^{N \times m_b}. \tag{4.27}$$

Finally, the function of our proposed LA regularization is defined as follows

$$\|\mathbf{X}\|_{LA*} = \sum_{b=1}^{B} \|\mathbf{H}_{\hat{\mathbf{x}}_b}\|_* \tag{4.28}$$

where $\| \cdot \|_*$ denotes the nuclear norm, $\mathbf{X} \mapsto \sum_{i=1}^{rank(\mathbf{X})} \sigma_i(\mathbf{X})$, with σ_i denotes the i-th singular value.

A fixed number of layers of the local block, m_b, is used even when the number of endmembers is larger than m_b, e.g., five. In this case, the first five endmembers are considered. When the number of endmembers is larger than five, our algorithm still considers all of the endmembers by shifting the local block in the third dimension (endmember direction) for each local spatial region. So, all endmembers of the spectral library will be considered. When using a large library, it is probable that none of them is present in the local block. In this case, the rank of local abundance matrix $\mathbf{H}_{\hat{\mathbf{x}}_b}$ is zero, and the low-rankness is hold.

If a local block has a linearity, then the corresponding local abundance vectors become almost linearly dependent, and vice versa. This means that the singular
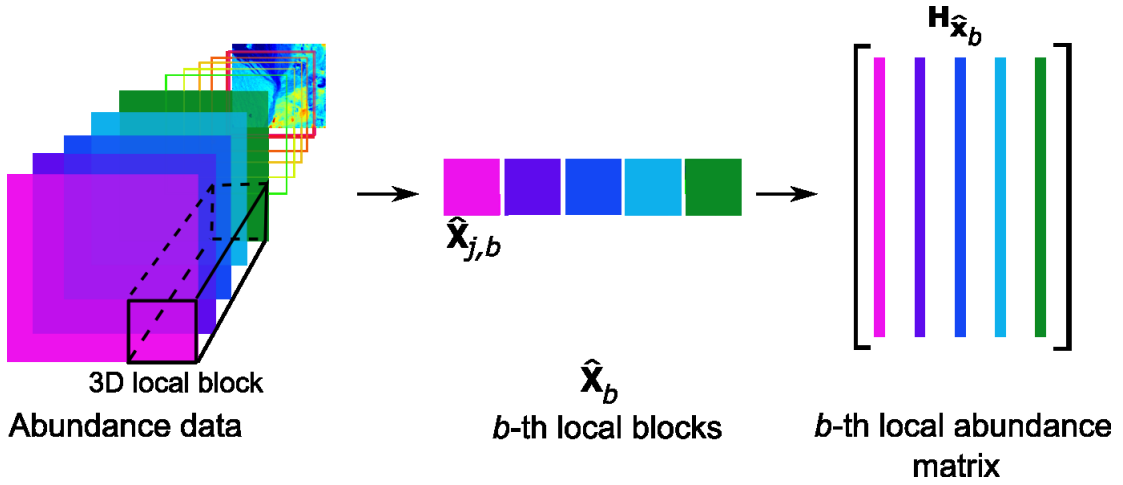
Figure 4.7: Process of vectorizing and arranging local abundance matrix of hyperspectral image.

values other than the first one of any local abundance matrix are expected to be small, that correspond to the non-active set of endmembers in the local region. Hence, by suppressing nuclear norm to the local abundance matrices, we can promote the linearity in the abundance domain.

It is possible that some of the local block abundance matrices are also of full rank. In this sense, it is not effective to penalize their nuclear norm. We avoid this condition by finding the optimum block size. We observed the portion of full-rank block for the simulated data sets. For the *DS* data set, the portion of full-rank block is 0. For the *FR1-5* data sets, the portions are 0.0067, 0.0072, 0.0079, 0.0013, 0.0012, respectively. We can say that the full-rank case is very rare in most data.

4.4.4   J-LASU

We formulate the new problem by adding the LA term with Equation (4.26) and the additional TV term. Thus, the problem of the proposed J-LASU algorithm in a convex form becomes

$$\min \ g(\mathbf{X}) = \tfrac{1}{2}\|\mathbf{AX} - \mathbf{Y}\|_F^2 + \lambda\|\mathbf{X}\|_{2,1} + \gamma\|\mathbf{X}\|_{TV} + \rho\|\mathbf{X}\|_{LA*} \quad \text{s.t.} \quad \mathbf{X} \geq 0 \tag{4.29}$$

where λ, γ, and ρ are the regularization parameter for the collaborative sparsity, TV, and LA term, respectively. The $L_{1,2}$ norm is applied to the abundance matrix $\mathbf{X}$ which is low rank. The nuclear norm is applied to the local abundance matrix

$\mathbf{H}_{\hat{\mathbf{x}}_b}$, which is different with $\mathbf{X}$. For the TV term, we use the anisotropic TV, which is used in SUnSAL-TV [38], defined as

$$\|\mathbf{X}\|_{TV} = \|\mathbf{DX}\|_1 \tag{4.30}$$

where $\mathbf{D} = [\mathbf{D}_h; \mathbf{D}_v]$, $\mathbf{D}_h : \mathbb{R}^{m \times n} \to \mathbb{R}^{m \times n}$ and $\mathbf{D}_v : \mathbb{R}^{m \times n} \to \mathbb{R}^{m \times n}$, are horizontal and vertical differential operators, respectively. The $\mathbf{D}_h \mathbf{X}$ computes the differences between the components of $\mathbf{X}$ and the corresponding right-side adjacent pixels with cyclic boundary assumption, and the same way for $\mathbf{D}_v \mathbf{X}$, which corresponds to the differences with the up-side adjacent pixels [38].

We estimate the abundance matrix $\mathbf{X}$ by solving problem in Equation (4.29) by using ADMM. The cost function in Equation (4.29) written in ADMM form becomes

$$f_1(\mathbf{X}) + f_2(\mathbf{V}) \quad \text{s.t.} \ \ \mathbf{V} = \mathbf{GX} \tag{4.31}$$

where

$$f_1(\mathbf{X}) = \frac{1}{2}\|\mathbf{AX} - \mathbf{Y}\|_F^2 \tag{4.32}$$

$$f_2(\mathbf{V}) = \lambda\|\mathbf{V}_1\|_{2,1} + \gamma\|\mathbf{V}_2\|_1 + \rho\|\mathbf{V}_3\|_{LA} + \iota_{R+}(\mathbf{V}_4) \tag{4.33}$$

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \\ \mathbf{V}_4 \end{bmatrix}, \text{and } \mathbf{G} = \begin{bmatrix} \mathbf{I} \\ \mathbf{D} \\ \mathbf{I} \\ \mathbf{I} \end{bmatrix}. \tag{4.34}$$

Here, the ι_{R+} term projects the solution onto the nonnegative orthant ($\iota_{R+}(x) = 0$ if $x \geq 0$ and $\iota_{R+}(x) = +\infty$ otherwise), and $\mathbf{I}$ is an identity matrix with a proportional size. The constraint in Equation (4.31) satisfies the relations

$$\mathbf{V}_1 = \mathbf{X}; \quad \mathbf{V}_2 = \mathbf{DX}; \quad \mathbf{V}_3 = \mathbf{X}; \quad \mathbf{V}_4 = \mathbf{X}. \tag{4.35}$$

Using a positive constant μ and the Lagrange multipliers $\mathbf{B}/\mu$ corresponding to the constraint $\mathbf{V} = \mathbf{GX}$, the cost function is minimized using ADMM. Then, the steps for the proposed algorithm are as follows

$$\mathbf{X}^{(k+1)} = \arg\min_{\mathbf{X}} \ f_1(\mathbf{X}) + \frac{\mu}{2}\|\mathbf{GX} - \mathbf{V}^{(k)} - \mathbf{B}^{(k)}\|_F^2 \tag{4.36}$$

$$\mathbf{V}^{(k+1)} = \arg\min_{\mathbf{V}} \ f_2(\mathbf{V}) + \frac{\mu}{2}\|\mathbf{GX}^{(k)} - \mathbf{V} - \mathbf{B}^{(k)}\|_2^2 \tag{4.37}$$

$$\mathbf{B}^{(k+1)} = \mathbf{B}^{(k)} - (\mathbf{GX}^{(k+1)} - \mathbf{V}^{(k+1)}). \tag{4.38}$$

To find the solution for $\mathbf{X}$ of the augmented Lagrangian formula, we calculate the solution of Equation (4.36) by taking the partial derivative as follows:

$$
\begin{aligned}
\mathbf{X}^{(k+1)} \quad &= \arg\min_{\mathbf{X}} \frac{1}{2}\|\mathbf{AX} - \mathbf{Y}\|_F^2 + \frac{\mu}{2}\|\mathbf{GX} - \mathbf{V}^{(k)} - \mathbf{B}^{(k)}\|_F^2 \\
&= \left(\mathbf{A}^T\mathbf{A} + \mu\mathbf{G}^T\mathbf{G}\right)^{-1}\left(\mathbf{A}^T\mathbf{Y} + \mu\mathbf{G}^T(\mathbf{V}^{(k)} + \mathbf{B}^{(k)})\right)
\end{aligned}
\tag{4.39}
$$

The detailed steps for computing the values of variables $\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3$, and $\mathbf{V}_4$ for each iteration are written in general form of the proximal operator.

$$
\begin{aligned}
\mathbf{V}_1^{(k+1)} \quad &= \quad \mathrm{prox}_{\frac{\lambda}{\mu}\|\cdot\|_{2,1}}(\mathbf{R}_1) \\
&= \quad \arg\min_{\mathbf{V}_1} \lambda\|\mathbf{V}_1\|_{2,1} + \frac{\mu}{2}\|\mathbf{V}_1 - \mathbf{R}_1\|_2^2
\end{aligned}
\tag{4.40}
$$

$$
\begin{aligned}
\mathbf{V}_2^{(k+1)} \quad &= \quad \mathrm{prox}_{\frac{\gamma}{\mu}\|\cdot\|_1}(\mathbf{R}_2) \\
&= \quad \arg\min_{\mathbf{V}_2} \gamma\|\mathbf{V}_2\|_1 + \frac{\mu}{2}\|\mathbf{V}_2 - \mathbf{R}_2\|_2^2
\end{aligned}
\tag{4.41}
$$

$$
\begin{aligned}
\mathbf{V}_3^{(k+1)} \quad &= \quad \mathrm{prox}_{\frac{\rho}{\mu}\|\cdot\|_{LA*}}(\mathbf{R}_3) \\
&= \quad \arg\min_{\mathbf{V}_3} \rho\|\mathbf{V}_3\|_{LA*} + \frac{\mu}{2}\|\mathbf{V}_3 - \mathbf{R}_3\|_2^2
\end{aligned}
\tag{4.42}
$$

$$
\begin{aligned}
\mathbf{V}_4^{(k+1)} \quad &= \quad \mathrm{prox}_{\frac{1}{\mu}(\iota_{R+})}(\mathbf{R}_4) \\
&= \quad \arg\min_{\mathbf{V}_4} \iota_{R+}(\mathbf{V}_4) + \frac{\mu}{2}\|\mathbf{V}_4 - \mathbf{R}_4\|_2^2
\end{aligned}
\tag{4.43}
$$

where $\mathbf{R}_1 = \mathbf{X}^{(k)} - \mathbf{B}_1^{(k)}$, $\mathbf{R}_2 = \mathbf{DX}^{(k)} - \mathbf{B}_2^{(k)}$, $\mathbf{R}_3 = \mathbf{X}^{(k)} - \mathbf{B}_3^{(k)}$, and $\mathbf{R}_4 = \mathbf{X}^{(k)} - \mathbf{B}_4^{(k)}$, and $\mathbf{B} = [\mathbf{B}_1; \mathbf{B}_2; \mathbf{B}_3; \mathbf{B}_4]$.

For $\mathbf{V}_1^{(k+1)}$, since the $L_{2,1}$ norm is not differentiable, the solution is obtained by the shrinkage for the group lasso as follows:

$$
v_{1(i)}^{(k+1)} = \begin{cases} r_{1(i)} - \frac{\lambda}{\mu}\frac{r_{1(i)}}{\|r_{1(i)}\|_2} & \text{if } \|r_{1(i)}\|_2 > \frac{\lambda}{\mu} \\ 0 & \text{otherwise} \end{cases}
\tag{4.44}
$$

where $v_{1(i)}^{(k+1)}$ and $r_{1(i)}$ denote the i-row of $\mathbf{V}_1^{(k+1)}$ and $\mathbf{R}_1$, respectively. This operation is denoted as $group-lasso(\cdot, \tau)$, where τ is the threshold.

The TV term in Equation (4.41) is solved by soft-thresholding on each element of $\mathbf{V}_2^{(k+1)}$.

$$v_{2(i,j)}^{(k+1)} = \begin{cases} r_{2(i,j)} - \frac{\gamma}{\mu} & \text{if } r_{2(i,j)} > \frac{\gamma}{\mu} \\ r_{2(i,j)} + \frac{\gamma}{\mu} & \text{if } r_{2(i,j)} < -\frac{\gamma}{\mu} \\ 0 & \text{if } -\frac{\gamma}{\mu} \leq r_{2(i,j)} \leq \frac{\gamma}{\mu} \end{cases} \tag{4.45}$$

where $v_{2(i,j)}^{(k+1)}$ and $r_{2(i,j)}$ denote the (i,j)-element of $\mathbf{V}_2^{(k+1)}$ and $\mathbf{R}_2$, respectively. This operation is denoted as $soft(\,\cdot\,,\tau)$, where τ is the threshold.

The solution of $\mathbf{V}_3^{(k+1)}$ in Equation (4.42) is acquired by constructing the LA matrices, applying singular value shrinkage to each matrix, and reconstructing the output abundance matrix, which is denoted as

$$\mathbf{V}_3^{(k+1)} = shr(\mathbf{X}^{(k)} - \mathbf{B}_3^{(k)}, \tfrac{\rho}{\mu}) \tag{4.46}$$

where $shr(\,\cdot\,,\tau)$ denotes the singular value shrinkage ($y \mapsto \text{diag}(max\{SVD(y) - \tau, 0\})$) of the LA matrices $\mathbf{H}_{\hat{\mathbf{x}}_b}$, where the singular value decomposition $SVD(\cdot)$ produces a vector containing the singular values in decreasing order and τ is the threshold.

Let $v_{4(i,j)}^{(k+1)}$ denotes the (i,j)-element of $\mathbf{V}_4^{(k+1)}$, finally, the solution of $\mathbf{V}_4^{(k+1)}$ is obtained by

$$v_{4(i,j)}^{(k+1)} = max(r_{4(i,j)}, 0) \tag{4.47}$$

where $r_{4(i,j)}$ denotes the (i,j)-element of $\mathbf{R}_4$.

The whole procedure of ADMM is summarized in Algorithm 1.

4.5   Experiment

We test the proposed algorithm on several simulated data sets for three signal-to-noise ratio (SNR) levels, i.e., 10, 20, and 30 dB, and two real data sets. We evaluate the results by conducting a fair comparison with the CLSUnSAL [39] and SunSAL-TV [38]. State-of-the-art low-rank algorithm is also compared, which is sparse and low-rank unmixing by using ADMM (ADSpLRU) [35].

4.5.1   Simulated Data Sets

To simulate the condition of hyperspectral data with and without the presence of pure pixels, we use two types of data distribution for data generation. Both use

---

Algorithm 1: ADMM in pseudocode for solving problem in Equation (4.29)

---

1   Initialization: set $k = 0, \mathbf{V}^0 = \mathbf{0}, \mathbf{B}^0 = \mathbf{0}$, choose $\mu > 0, \lambda, \gamma, \rho$;

2   **while** *the stopping criterion is not satisfied* **do**

3      $\mathbf{X}^{(k+1)} \leftarrow \left(\mathbf{A}^T\mathbf{A} + \mu\mathbf{G}^T\mathbf{G}\right)^{-1}\left(\mathbf{A}^T\mathbf{Y} + \mu\mathbf{G}^T(\mathbf{V}^{(k)} + \mathbf{B}^{(k)})\right)$;

4      $\mathbf{V}_1^{(k+1)} \leftarrow group-lasso(\mathbf{X}^{(k)} - \mathbf{B}_1^{(k)}, \lambda/\mu)$;

5      $\mathbf{V}_2^{(k+1)} \leftarrow soft(\mathbf{DX}^{(k)} - \mathbf{B}_2^{(k)}, \gamma/\mu)$;

6      $\mathbf{V}_3^{(k+1)} \leftarrow shr(\mathbf{X}^{(k)} - \mathbf{B}_3^{(k)}, \rho/\mu)$;

7      $\mathbf{V}_4^{(k+1)} \leftarrow max(\mathbf{X}^{(k)} - \mathbf{B}_4^{(k)}, 0)$;

8      $\mathbf{B}_1^{(k+1)} \leftarrow \mathbf{B}_1^{(k)} - \mathbf{X}^{(k+1)} + \mathbf{V}_1^{(k+1)}$;

9      $\mathbf{B}_2^{(k+1)} \leftarrow \mathbf{B}_2^{(k)} - \mathbf{DX}^{(k+1)} + \mathbf{V}_2^{(k+1)}$;

10     $\mathbf{B}_3^{(k+1)} \leftarrow \mathbf{B}_3^{(k)} - \mathbf{X}^{(k+1)} + \mathbf{V}_3^{(k+1)}$;

11     $\mathbf{B}_4^{(k+1)} \leftarrow \mathbf{B}_4^{(k)} - \mathbf{X}^{(k+1)} + \mathbf{V}_4^{(k+1)}$;

12   Update iteration: $k \leftarrow k + 1$;

13 **end**

---

the same library generated from 240 types of minerals selected randomly from the splib06 USGS library [5], which consists of 224 spectral bands ranging between 0.4–2.5 μm. The mutual coherence among the spectral signatures is very close to one, but we set the SA to be larger than 4.4 to make the sparse regression problem easier.

The first data set, *DS*, is a representation of the data with pure pixels and adopted from that of Iordache et al. [38] consisting of 224 bands for 75×75 pixels. The data generation follows the LMM with the abundance sum-to-one constraint imposed on each pixel. Five spectral signatures are randomly selected from the library as the endmembers and distributed spatially in the form of distinct square regions. In some pixels, the endmembers stay pure and in others they are mixed with two until there are five endmembers. In Figure 4.8, the red squares in each abundance map represent 100% intensity which means the pure pixel regions of each endmember. The background consists of mixed pixels with randomly fixed fractional abundance values of 0.1149, 0.0741, 0.2003, 0.2055, and 0.4051 for the five endmembers.

To demonstrate the proposed algorithm under the condition without the presence of pure pixels, the distribution with a distinct spatial pattern and mixture was selected. We use the fractal database (*FR*) [60] consisting of five data sets,
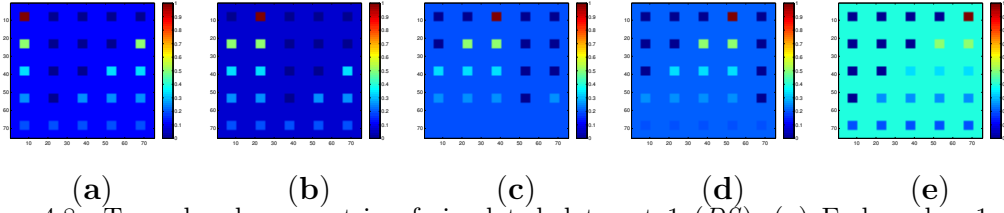
Figure 4.8: True abundance matrix of simulated data set 1 (*DS*). (**a**) Endmember 1. (**b**) Endmember 2. (**c**) Endmember 3. (**d**) Endmember 4. (**e**) Endmember 5.
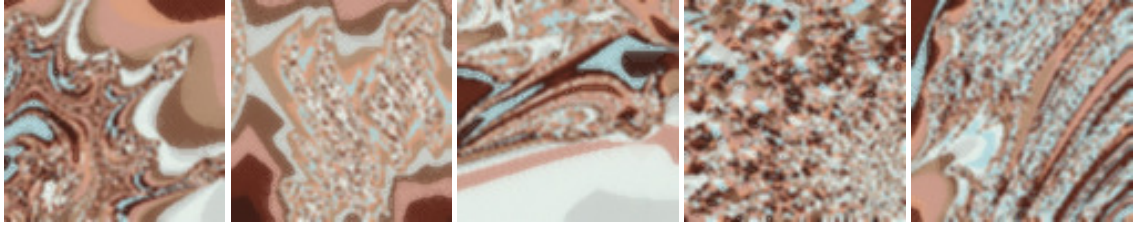
Figure 4.9: Fractal data sets represented in pseudocolor. (**a**) *FR1*. (**b**) *FR2*. (**c**) *FR3*. (**d**) *FR4*. (**e**) *FR5*.

namely *FR1, FR2, FR3, FR4*, and *FR5*. Each is composed of 100×100 pixels with 224 spectral bands for each pixel and contains no completely pure pixels that are close to the ground-truth characteristic in which completely pure pixels are rarely found. The distribution is generated such that pixels near the edges of regions are more highly mixed than those in the center of the regions. These center pixels have a purity index between 0.95–0.99, directly proportional to the broadness of the regions. In this experiment, we set the number of endmembers to 9. Figure 4.9 shows *FR1, FR2, FR3, FR4*, and *FR5* represented in pseudocolor.

4.5.2   Real Data Sets

For the real-data experiment, we use two real data from different sensors. The first hyperspectral scene is the widely used data set of Cuprite mining district, Nevada in 1997 [1]. We use a subscene with the size of 150×130 pixels whose area is shown in Figure 4.10a. The data are composed of 224 spectral bands with 3.7 m spatial resolution from the AVIRIS sensor. Prior to analysis, several bands were removed due to the low SNR; thus, remaining 188 bands. In this experiment, we use the USGS library of 498 spectral signatures as the standard spectral library

67
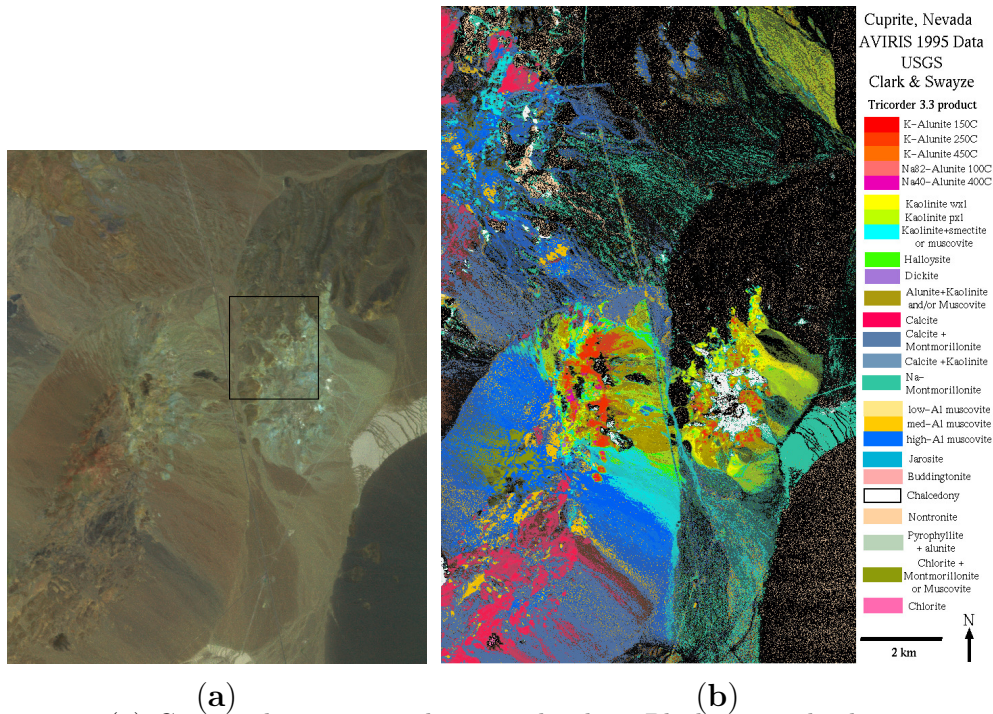
Figure 4.10: (**a**) Cuprite data generated in pseudocolor. Black rectangle shows area of our experiment. (**b**) USGS mineral distribution map of Cuprite mining district in Nevada [2].

for the data, with the corresponding bands removed. Figure 4.10b shows the USGS mineral distribution map of the Cuprite area [2]. From the figure, the area of interest contains at least three types of minerals: *alunite, chalcedony*, and *kaolinite*. The mineral map was produced using Tricorder 3.3 software in 1995, while the AVIRIS Cuprite data were collected in 1997. Hence, in our experiment, the mineral map was used only for visual qualitative evaluation, compared with the abundance maps of different sparse unmixing algorithms.

The second hyperspectral scene is Urban data captured by the HYDICE sensor over an area located at Copperas Cove near Fort Hood, TX, U.S., in October 1995. It consists of 307×307 pixels with 2 m of the pixel resolution. The wavelengths range from 0.4 to 2.5 μm divided into 210 spectral bands. After some bands with low SNRs due to dense water vapor and atmospheric effects are discarded, it remains 162 bands. We use a subscene with the size of 100×100 pixels. Figure 4.11a shows the subscene used in the experiment. The ground truth of the Urban data set is not available, however, we use the reference abundance maps obtained from [3]. The maps are achieved via the method provided in [41,42,84] and consist

68
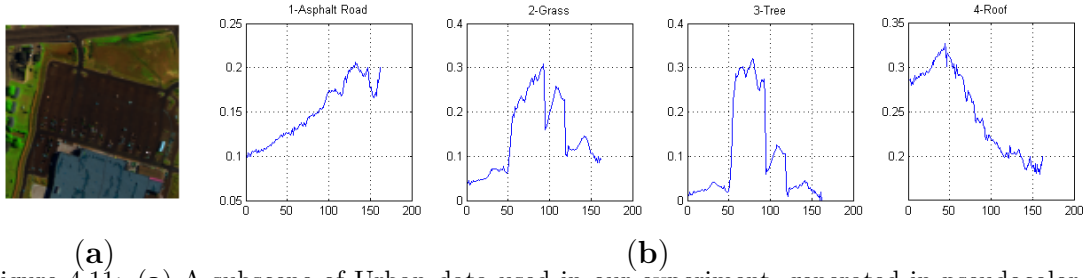
Figure 4.11: (**a**) A subscene of Urban data used in our experiment, generated in pseudocolor. (**b**) Spectral signatures of the endmembers [82–84], x-axis and y-axis represent the band number and reflectance unit (0–1), respectively.

of four endmembers, i.e., *asphalt, grass, tree,* and *roof.* Figure 4.11b shows the spectral signatures of the four endmembers.

4.5.3   Parameters Setting and Evaluation Metrics

In the simulated-data experiment, to build spectral library **A**, the spectral signatures in the USGS spectral library were selected and sorted such that the SAs between the spectral signatures were not less than 4.4 degrees in increasing order. The parameter settings of J-LASU are for the collaborative sparsity (λ), TV (γ), and LA nuclear norm (ρ) regularizer. For the compared algorithms, λ_{SP} is the sparsity term for CLSUnSAL, SUnSAL-TV, and ADSpLRU [35]. For SUnSAL-TV, the TV term is controlled by λ_{TV}. The low-rank regularizer parameter is denoted as λ_{LR} for ADSpLRU. These parameters are adjusted for every data set under different SNR levels. However, we use the same parameter settings for the five fractal data sets since the characteristics of the scenes tend to be similar. Table 4.1 summarizes these settings. The values of λ_{LR} were the optimal ones after the experiment for some recommended values. For the LA regularization, the block size is another parameter to be set to control the coverage of adjacent pixels in the spatial and the endmember directions. After several trials in this experiment, the optimum size was found to be [5 5 5] with no overlap.

We evaluate the performance of the algorithms using root mean square error (RMSE) [60, 85] and signal-to-reconstruction error (SRE) [37]. The RMSE measures the error between the original and reconstructed abundance matrices. The lower the RMSE, the more accurate the estimation is. The RMSE formula for the

Table 4.1: Parameter Settings

Data	SNR	CLSUnSAL	SUnSAL − TV		ADSpLRU		J − LASU		
		λ_{SP}	λ_{SP}	λ_{TV}	λ_{SP}	λ_{LR}	λ	γ	ρ
DS	10 dB	1e+1	1e-1	1e-1	1e+1	5e+0	5e-1	5e-1	5e-1
	20 dB	1e+0	5e-4	5e-2	1e+1	1e+0	25e-2	5e-2	3e-1
	30 dB	1e+0	5e-4	1e-2	1e+0	1e-1	5e-2	1e-2	8e-2
FR	10 dB	1e+1	5e-2	1e-1	1e+1	5e+0	5e-1	1e-1	25e-2
	20 dB	1e+0	5e-3	5e-2	1e+1	3e+0	25e-2	1e-2	1e-1
	30 dB	1e+0	5e-3	25e-3	1e+1	1e-2	5e-2	5e-3	5e-2
Cuprite	-	5e-3	5e-4	1e-2	1e+1	1e+0	1e-2	5e-4	1e-3
Urban	-	1e-4	1e-4	1e-4	1e-2	1e-2	1e-4	1e-4	1e-4

i-th endmember is defined as

$$\mathrm{RMSE}_i = \sqrt{\frac{1}{n}\sum_{h=1}^{n}(\mathbf{X}_{i,h} - \bar{\mathbf{X}}_{i,h})^2},\qquad(4.48)$$

where n, $\mathbf{X}$ and $\bar{\mathbf{X}}$ represent the number of pixels, true abundance matrices, and estimated abundance matrices, respectively. Then, we compute the mean value of all endmembers' RMSEs.

The SRE represents the ratio between the reconstructed abundance matrix and error, and is defined as

$$\mathrm{SRE} = 10\log_{10}\left(\|\mathbf{X}\|_F^2 / \|\mathbf{X} - \bar{\mathbf{X}}\|_2^F\right).\qquad(4.49)$$

For the simulated data, the original abundance matrix was generated for each data set. We compare the visual appearance among the maps of the estimated abundance matrix in addition to RMSE and SRE comparison. As for the first real data set, Cuprite, the comparison was among the estimated abundance maps of the sparse unmixing algorithms and the mineral map of each expected endmembers. For the second real data set, Urban, RMSE and SRE of each method are calculated with the ground truth abundance maps as the reference value.

4.5.4 Simulated-Data Experiment

Tables 4.2 and 4.3 show the RMSE and SRE values, respectively, of estimated abundances from the compared algorithms. The proposed J-LASU algorithm

Table 4.2: RMSE Comparison Result

Data	SNR	CLSUnSAL	SUnSAL $-$ TV	ADSpLRU	J $-$ LASU
DS	10	0.0084	0.0078	0.0097	**0.0035**
	20	0.0102	0.0046	0.0053	**0.0013**
	30	0.0039	0.0023	0.0038	**0.0008**
FR1	10	0.0130	0.0119	0.0140	**0.0103**
	20	0.0129	0.0087	0.0107	**0.0075**
	30	0.0062	0.0068	0.0073	**0.0050**
FR2	10	0.0140	0.0119	0.0149	**0.0104**
	20	0.0138	0.0083	0.0115	**0.0076**
	30	0.0062	0.0061	0.0066	**0.0052**
FR3	10	0.0136	0.0118	0.0130	**0.0099**
	20	0.0128	0.0077	0.0107	**0.0069**
	30	0.0056	0.0058	0.0057	**0.0044**
FR4	10	0.0123	0.0120	0.0135	**0.0103**
	20	0.0126	0.0089	0.0090	**0.0074**
	30	0.0057	0.0075	0.0058	**0.0049**
FR5	10	0.0118	0.0112	0.0139	**0.0092**
	20	0.0119	0.0080	0.0106	**0.0065**
	30	0.0049	0.0062	0.0061	**0.0043**

achieved better RMSE for all the simulated data. For the same level of SNR, J-LASU performed better than CLSUnSAL and SUnSAL-TV as well as ADSpLRU. The improvement also can be clearly seen in the *DS* data set from Figure 4.12. J-LASU preserved the square regions better than the others. Compared with the TV results, difference can be recognized in the small square regions in which J-LASU reconstructed the squares better. For the *FR* data sets, visually, the ADSpLRU abundance maps showed the most similar intensity with the corresponding true abundance maps. However, J-LASU is superior in preserving the gradation of intensity from edge to center of an abundance region, which is the drawback of the ADSpLRU. Compared with SUnSAL-TV, J-LASU was more accurate in determining whether an abundance is an outlier or just a low-intensity edge abundance. In addition, SUnSAL-TV produced stronger smoothing effects than J-LASU. In this case, J-LASU results are more similar with the true abundance map, which can easily be compared in the *FR2* data set.

(a)CLSUnSAL   (b) SUnSAL-TV   (c) ADSpLRU

(d) J-LASU   (e) the true abundance

Figure 4.12: Estimated abundance maps for simulated data sets *DS* for SNR 30 dB using CLSUnSAL, SUnSAL-TV, ADSpLRU, and J-LASU (a–d, respectively) compared to the true abundance (e).

(a)CLSUnSAL   (b) SUnSAL-TV   (c) ADSpLRU

(d) J-LASU   (e) the true abundance

Figure 4.13: Estimated abundance maps for simulated data sets *FR1* for SNR 30 dB using CLSUnSAL, SUnSAL-TV, ADSpLRU, and J-LASU (a–d, respectively) compared to the true abundance (e).

(a)CLSUnSAL   (b) SUnSAL-TV   (c) ADSpLRU

(d) J-LASU   (e) the true abundance
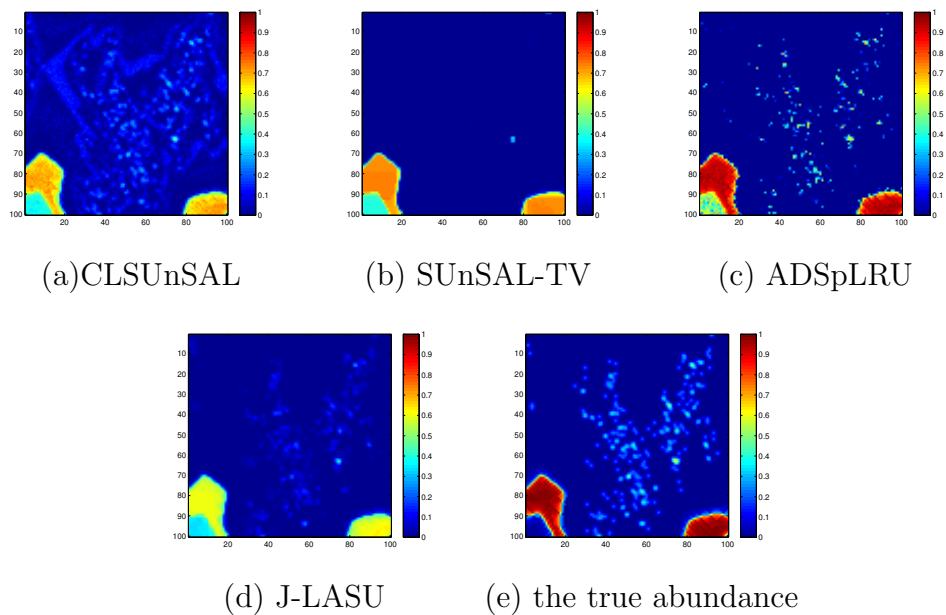
Figure 4.14: Estimated abundance maps for simulated data sets *FR2* for SNR 30 dB using CLSUnSAL, SUnSAL-TV, ADSpLRU, and J-LASU (a–d, respectively) compared to the true abundance (e).

(a)CLSUnSAL   (b) SUnSAL-TV   (c) ADSpLRU

(d) J-LASU   (e) the true abundance
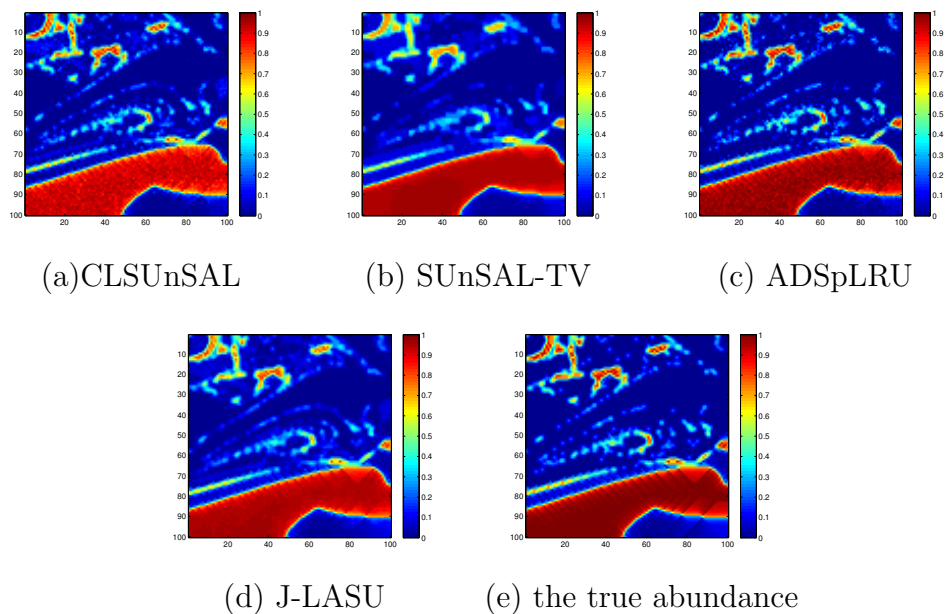
Figure 4.15: Estimated abundance maps for simulated data sets *FR3* for SNR 30 dB using CLSUnSAL, SUnSAL-TV, ADSpLRU, and J-LASU (a–d, respectively) compared to the true abundance (e).

73

(a)CLSUnSAL   (b) SUnSAL-TV   (c) ADSpLRU

(d) J-LASU   (e) the true abundance

Figure 4.16: Estimated abundance maps for simulated data sets *FR4* for SNR 30 dB using CLSUnSAL, SUnSAL-TV, ADSpLRU, and J-LASU (a–d, respectively) compared to the true abundance (e).

(a)CLSUnSAL   (b) SUnSAL-TV   (c) ADSpLRU

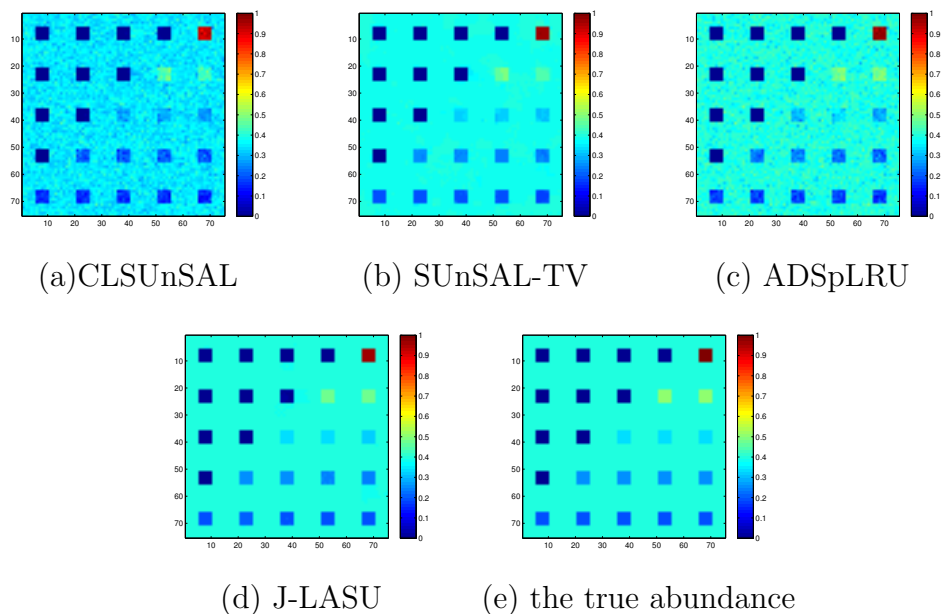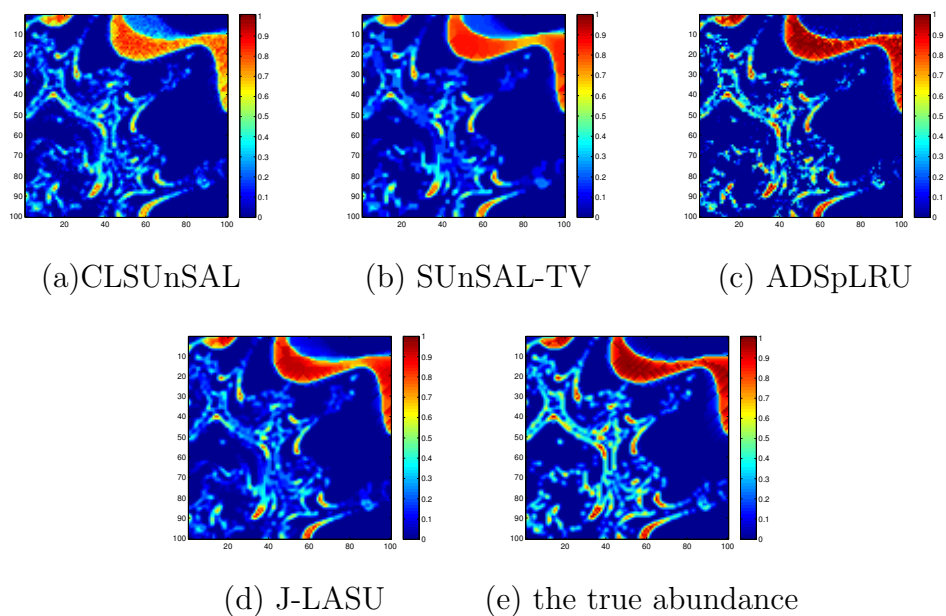(d) J-LASU   (e) the true abundance

Figure 4.17: Estimated abundance maps for simulated data sets *FR5* for SNR 30 dB using CLSUnSAL, SUnSAL-TV, ADSpLRU, and J-LASU (a–d, respectively) compared to the true abundance (e).
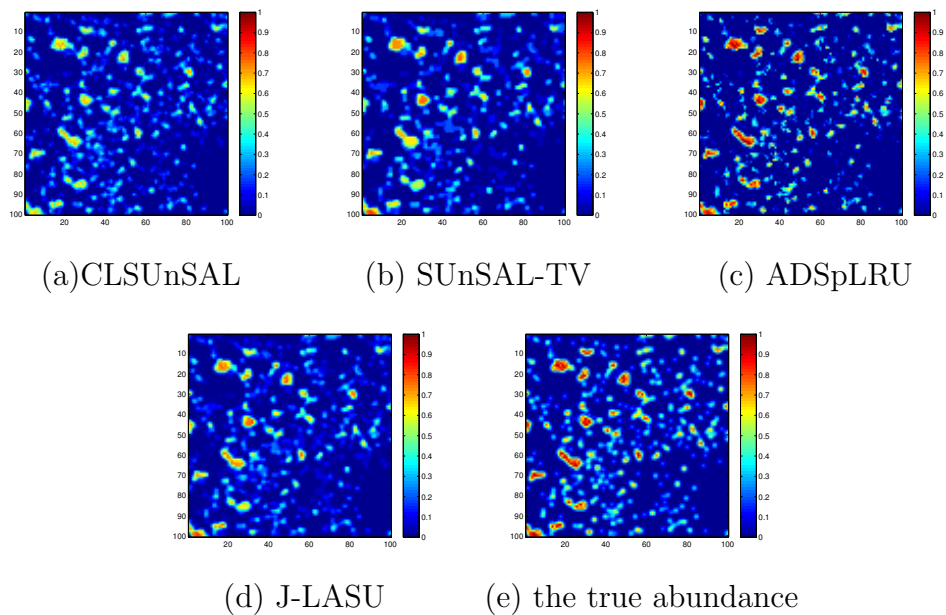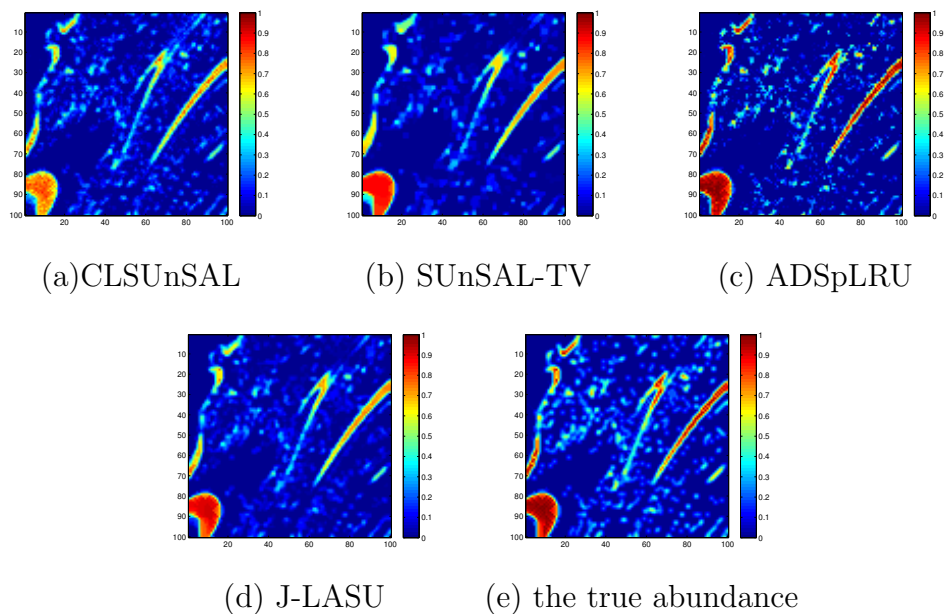
Table 4.3: SRE Comparison Result

Data	SNR	CLSUnSAL	SUnSAL − TV	ADSpLRU	J − LASU
DS	10	2.5467	5.1021	0.3110	**7.2571**
	20	2.1617	6.3470	4.5515	**15.2631**
	30	6.3299	10.5770	6.1799	**20.0581**
FR1	10	0.6435	2.018	0.851	**2.3625**
	20	1.3116	3.5071	2.1257	**4.2158**
	30	4.2204	4.8625	4.0937	**6.0185**
FR2	10	0.3457	2.2395	0.2493	**2.4491**
	20	1.1915	3.8690	1.0974	**4.3822**
	30	4.4628	5.604	4.5908	**6.3273**
FR3	10	1.6928	4.0113	2.1009	**4.3074**
	20	3.1706	5.8611	2.3815	**6.5586**
	30	6.8354	6.9782	7.0605	**8.7567**
FR4	10	0.3417	1.3213	0.2092	**1.6307**
	20	1.0942	2.5735	0.3275	**3.5269**
	30	4.1734	3.263	3.5545	**5.4870**
FR5	10	1.005	2.4054	0.2591	**3.0209**
	20	1.5711	4.1026	1.228	**5.4771**
	30	6.3324	5.6279	6.0702	**7.7098**

4.5.5   Real-Data Experiment

The visual comparisons among the five sparse unmixing algorithms and the mineral maps for the Cuprite data can be observed in Figure 4.18. The images in the first column show the comparison for *alunite* abundance maps. Among the results of the compared algorithms, The proposed J-LASU produced the map that was the most similar to the mineral map, with less outliers found in the lower-left side of the map. The same superiority was also found among the *chalcedony* and *kaolinite* abundance maps in the second and third columns, respectively. Compared to SUnSAL-TV, J-LASU had less outliers or lower intensity of outliers, most of which were found on the left-side region of the maps.

It should be noted that the estimated abundance maps of any sparse unmixing algorithm are not exactly the same as the mineral maps generated from the Tricorder software in terms of intensity. The software produced the pixel-level classification maps, while the sparse unmixing algorithms executed subpixel-level classification. However, the comparison of outliers in this thesis refers to the abundances that no longer exist in the mineral map. Overall, J-LASU estimated

Table 4.4: RMSE and SRE Comparison Result for Urban data

Algorithms	CLSUnSAL	SUnSAL − TV	ADSpLRU	J − LASU
RMSE	0.2135	0.2003	0.2077	**0.1948**
SRE	4.6831	5.4738	5.0805	**5.8719**

abundance maps had smooth gradation of intensity from the edge of a detected region to the center, and removed tiny regions that were found in the other algorithms' map, which seems to be the outliers in J-LASU algorithm.

For the Urban data, Figure 4.19 shows the ground truth and abundance maps of the four endmembers estimated by the compared algorithms. J-LASU algorithm resulted in the most similar maps to the ground truth, especially for the *asphalt* abundance map which is easier to be compared with those of the other algorithms. The quantitative comparisons also show that J-LASU yielded the best performance, with the highest SRE and lowest RMSE, as shown in Table 4.4. Compare to the simulated data, the Urban data experienced relatively high RMSEs for all compared algorithms. This is due to the fact that the ground truth abudance maps used for the Urban data are not achieved from a ground measurement, but from a method in which error possibly exists in term of method accuration.

4.6   Discussion

4.6.1   Sensitivity Test

We evaluate the performance of the proposed J-LASU algorithm when λ, γ, and ρ were not set to the optimal values. In the experiment, when a parameter was adjusted from 0 to 10, the other parameters were set to their optimal values. When the parameter increases from 0 to the optimal value, the RMSEs decrease and the SREs increase gradually. When it reaches a higher value, the results worsen. Hence, we can conclude that each parameter influences the performance of J-LASU.

To clearly evaluate the contribution of the LA regularizer, we conduct an experiment of our optimization problem with $\rho = 0$, which means no contribution of the LA regularizer. Figure 4.20 represent the RMSE of this condition at the three levels of SNR compared to those of J-LASU, where $\rho > 0$. For each simulated data set, other parameters were set to the optimum values. For all data, it was

observed that when $\rho = 0$, the RMSE was higher than the condition when the LA regularizer was used. In other words, adding our LA regularizer with an optimal regularization parameter will contribute improvement in RMSE.

We found that improvement in visual quality corresponds to the additional low-rank regularization. Figure 4.21 shows visual improvement due to the abundance regularizer. The abundance maps in the figure belong to endmember 5 of the *DS* data set and endmember 7 of the *FR 2* data set. For the *FR* abundance maps, one can see that after applying our LA regularizer with an optimal ρ, the active abundances have higher intensities. The active abundances in the left-edge of the map and around the speckles clearly appear, although in lower intensities than in the true abundance map. For the *DS* data set, when ρ is set to the optimal value, the small squares are preserved better than when $\rho = 0$.

(a-1)   **(a-2)**   **(a-3)**

(b-1)   **(b-2)**   **(b-3)**

(c-1)   **(c-2)**   **(c-3)**

(d-1)   **(d-2)**   **(d-3)**

(e-1)   **(e-2)**   **(e-3)**
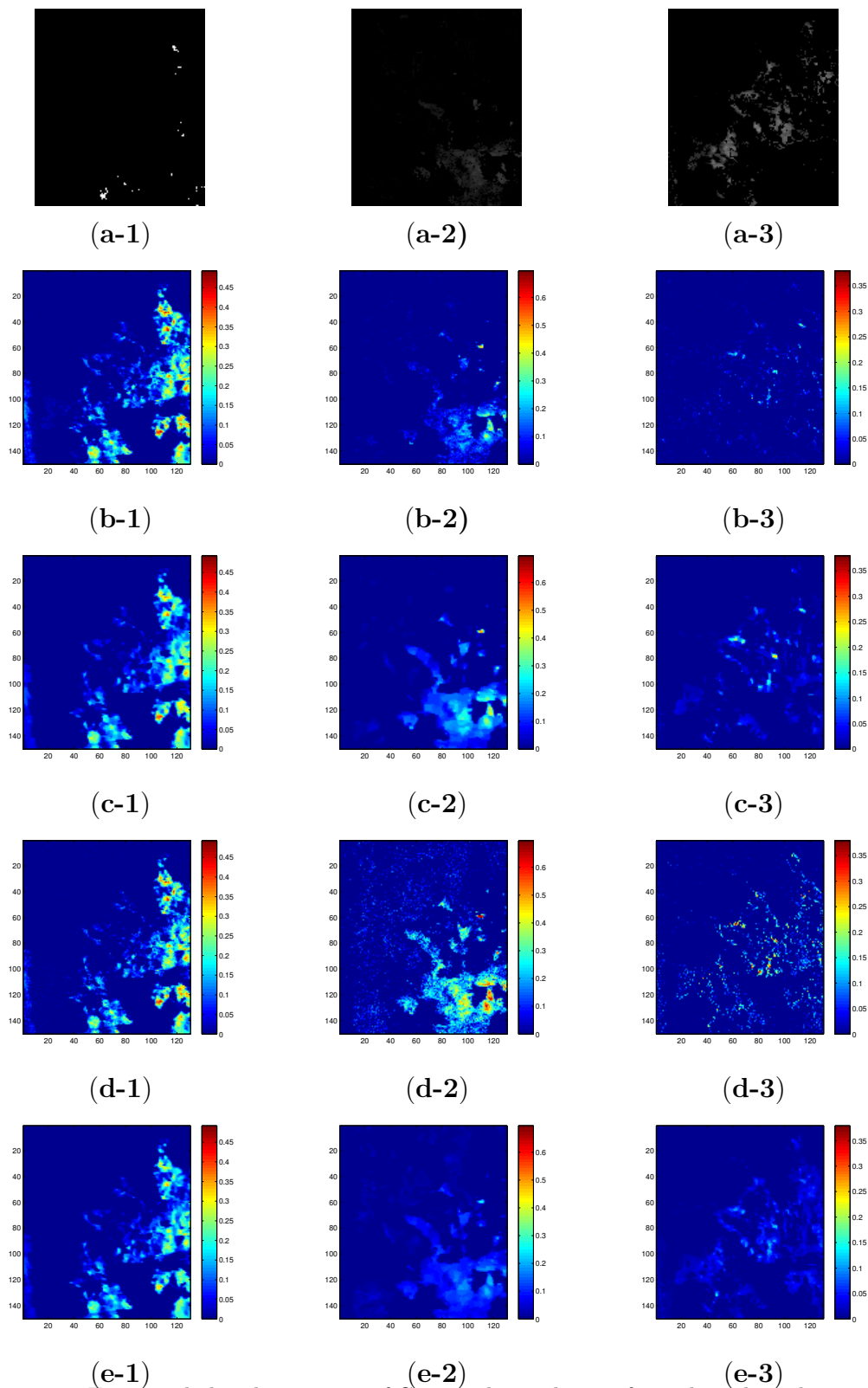
Figure 4.18: Estimated abundance maps of Cuprite data subscene for endmember *alunite, chalcedony,* and *kaolinite* (column **1**–**3**) using CLSUnSAL, SUnSAL-TV, ADSpLRU and J-LASU (row **b**–**e**). First row (**a**) shows classification maps of endmembers from USGS Tetracorder.

(a-1) (a-2) (a-3) (a-4)

(b-1) (b-2) (b-3) (b-4)

(c-1) (c-2) (c-3) (c-4)

(d-1) (d-2) (d-3) (d-4)

(e-1) (e-2) (e-3) (e-4)
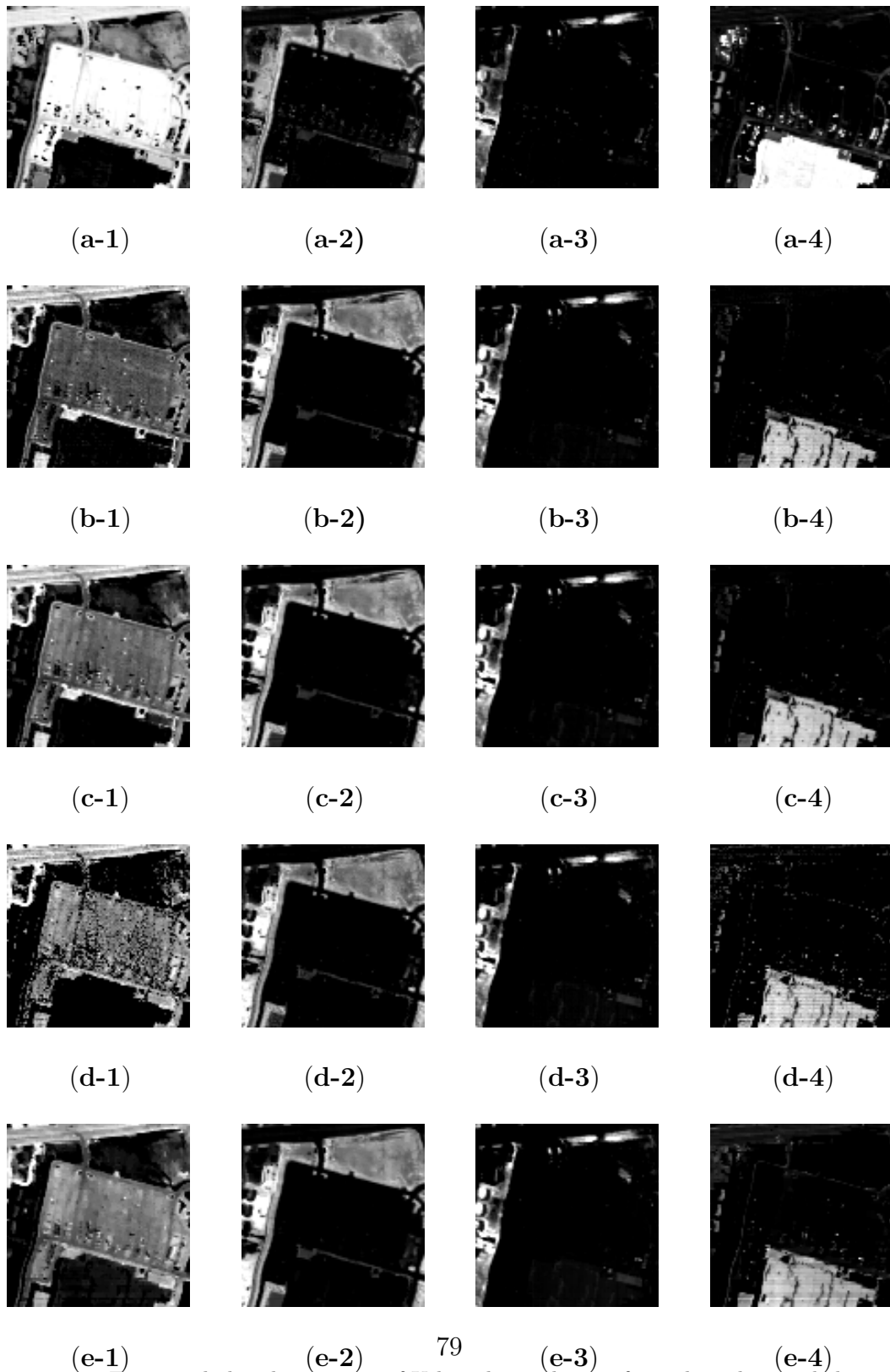
Figure 4.19: Estimated abundance maps of Urban data subscene for endmember *asphalt, grass, tree,* and *roof* (column **1**–**4**) using CLSUnSAL, SUnSAL-TV, ADSpLRU and J-LASU (row **b**–**e**). First row (**a**) shows the ground truth abundance maps.
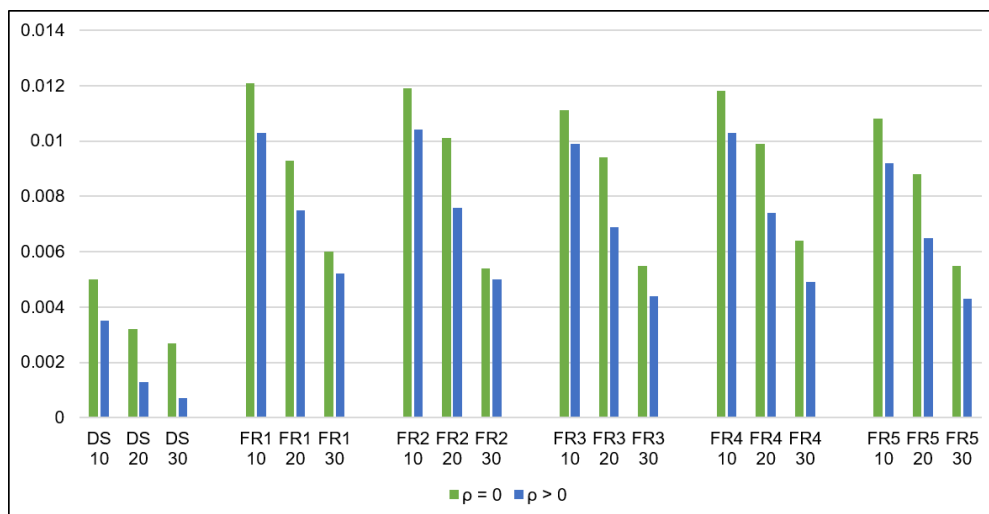
Figure 4.20: Effect of the LA regularizer represented by improvement in RMSE when $\rho > 0$.

(a-1)   (a-2)   (a-3)

(b-1)   (b-2)   (b-3)
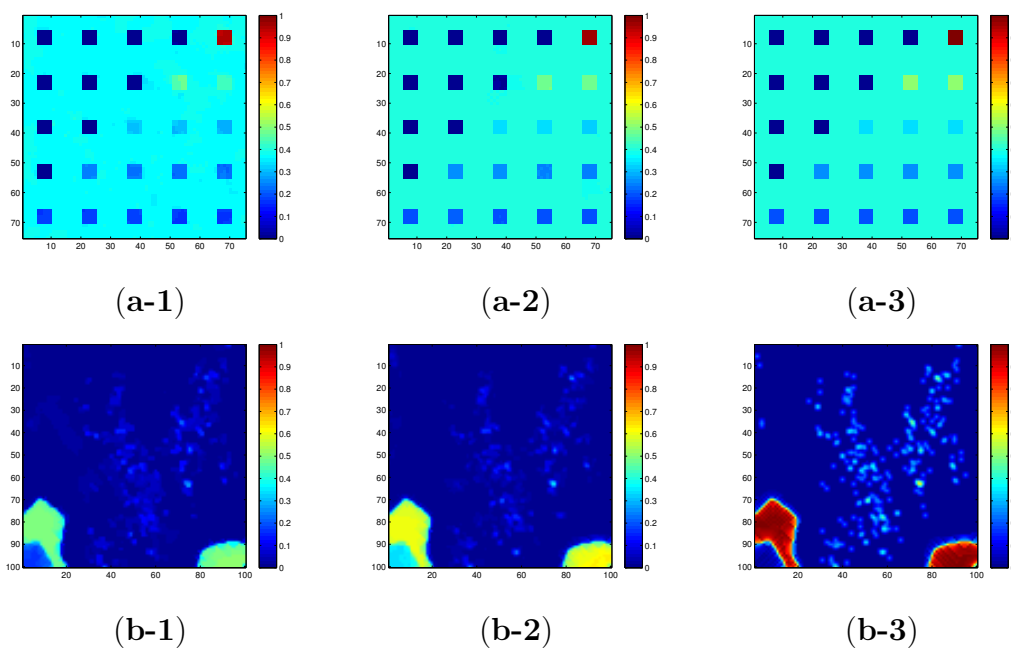
Figure 4.21: Effect of LA regularizer represented by improvement in RMSE when $\rho > 0$ for (a) *DS* data set and (b) *FR 2* data set. (a-1) and (b-1) Before, (a-2) and (b-2) after, (a-3) and (b-3) true abundance.
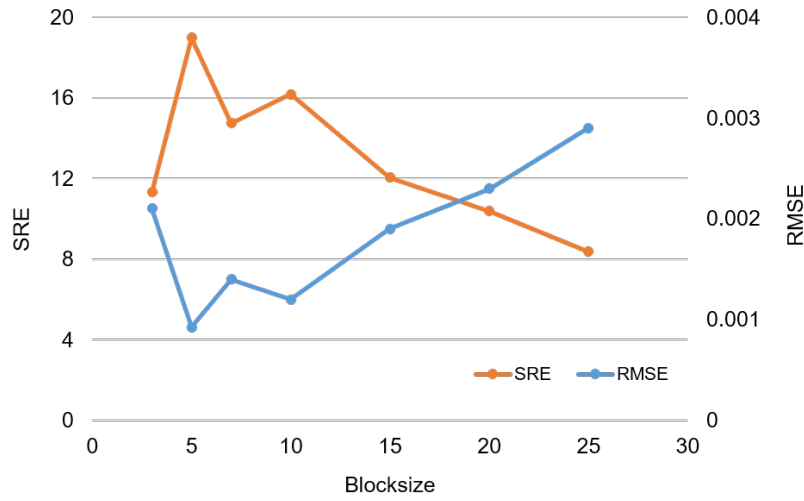
Figure 4.22: RMSE and SRE in relation to block size.

4.6.2   Effect of Block Size

The coverage of the local region affects the optimization results. In this region, the highly correlated abundance of the endmembers is taken into account by the local abundance nuclear norm. We conduct experiments to find the optimum size of the sliding block. We also observe the effect of the block size. Figure 4.22 shows the RMSE and SRE when the block size was adjusted in the DS data set. From the curves, we could determine that the radius of spatial similarity in the abundance map affects the optimum size of the sliding block. The distribution of spatial similarity in the DS data set, as shown in Figure 4.12, has a distinct pattern in which every 5×5 pixel has the same abundance value, giving the optimum block size in turn. However, the correlation does not hold for the data in which the spatial similarity is not represented in a square region, e.g., the FR and Cuprite data sets. In this circumstance, some trials were conducted prior to the experiment. After the trials, we found that the optimum size is 5 pixels. Hence, we select [5 5 5] as the optimum block size for all data.

81

Table 4.5: Comparison of running times for DS-data experiment

Algorithms	CLSUnSAL	SUnSAL − TV	ADSpLRU	J − LASU
Time/iteration (s)	0.92	0.54	0.24	2.77

4.6.3   Computational Complexity

The running-time comparison among the algorithms is summarized in Table 4.5. The experiment was conducted for the DS simulated data, which has 75×75 pixels, 224 bands, and 240 spectral signatures in the library. The algorithms ran on a desktop computer with 3.50-GHz Intel Core i5 processor and 8 GB of RAM. From the table, J-LASU was the slowest due to its high computational complexity.

For the complexity analysis, recall that n, m, N, and m_b are the number of pixels, spectral signatures in the library, pixels in each LA band, and local endmembers, respectively. For each iteration of J-LASU, the computation of $\mathbf{X}$ and the SVD step in the computation of $\mathbf{V}_3$ incur the most cost. The complexity of $\mathbf{X}$ computation is due to the use of conjugate gradient solver, which costs $\mathcal{O}(m)$ per iteration. The conjugate gradient is a popular iterative technique for solving the system of linear equation $\mathbf{Ax} = \mathbf{b}$, where the matrix $\mathbf{A}$ must be symmetric possitive definite (SPD), large and sparse. The SVD step costs $\mathcal{O}(m_b^2 N)$; however, this step is repeated as many times as the number of blocks (B) due to the sliding of the local block. Since B is calculated by $mn/m_b N$, the total cost of $\mathbf{V}_3$ is $\mathcal{O}(m_b mn)$, which is more complex than the computation of $\mathbf{X}$. Hence, the overall complexity costs $\mathcal{O}(m_b mn)$.

4.7   Conclusions

We propose the local abundance regularizer algorithm for the sparse unmixing problem to improve the accuracy of abundance estimation. By imposing the term to state-of-the-art unmixing algorithms, our algorithm incorporates both spatial and abundance correlation by using the low-rankness of the abundance. We implement the nuclear norm to the local abundance matrix, which defines the local region not only in the spatial, but also in the abundance dimension. The algorithm was run at certain SNR levels for several simulated data sets, which represent the conditions with and without pure pixels, and for two real data sets. The experimental results indicate that our proposed algorithm performs better than SUnSAL-TV and yields better results than the other state-of-the-art

algorithms. Relevant future research will be concerned with exploitation of the low-rankness of abundance for overlapping local regions.

Chapter 5
General Conclusions

In this work, we studied color-line property for possible multi-channel image processing tasks. As a useful feature to represent linearity of the image channels, color line has been exploited in several tasks in image processing field. For multi-channel images that consist of more than three channels, e.g., multispectral and hyperspectral images, we have shown that the color line can be adopted in which the linearity is formed from the intensities of adjacent spectral bands, thus we introduced the term spectral line.

In this thesis, we have presented the spectral line application for two hyperspectral image tasks, which are denoising and sparse unmixing. For denoising, we introduced a method based on our proposed local spectral component decomposition, in which the spectral line is extracted by this decomposition scheme, and followed by iterative filtering steps to recover the line. The work successfully demonstrated that the use of spectral line contributes to achieve better results compared with the result of the stand-alone conventional method. Another benefit of our method is that by the virtue of our decomposition scheme, it enables the denoising process only conducted for two extracted components, which are grayscale images, regardless of the number of the input channels.

As for our work in hyperspectral unmixing, the spectral-line property was exploited in term of the low-rankness of local abundance matrix, by applying the nuclear norm to the matrix. We introduced the local block that encompasses both spatial and abundance domains, that differs from the widely-used local window for 2D abundance matrix. We conducted experiments for real and simulated hyperspectral data sets assuming two conditions, i.e., with and without the presence of pure pixels. Our method is proved to be more accurate and exhibits the potential to improve the state-of-the-art unmixing methods.

The proven-applicability of color-line concept for multi-channel images provides a compelling direction for future work, e.g., applications in the field of biomedical imaging. For instance, experiments can be conducted for multi-channel

magnetic resonance imaging (MRI) with an approach of compressive sensing, to explore the applied concept. In addition to the unmixing algorithm, the implementation of unmixing algorithm in parallel computing environments can be considered for future developments in order to accelerate the algorithm even more. For example, recently, many research efforts have been dedicated to the implementation of hyperspectral unmixing algorithms for fast performance in specialized hardware devices, such as graphics processing units (GPUs) or field programmable gate arrays (FPGAs), which can be installed onboard hyperspectral imaging instruments. As for the sparse unmixing problem, it can be improved by exploring the loss function using another function, such as L_1 norm for a sparse error function. The observation can also be broadened to other types of noise, such as impulse noise, dead lines, stripes, and so on.

References

[1] AVIRIS data. `https://aviris.jpl.nasa.gov/data/free_data.html`. (accessed on 10 June 2016).

[2] Cuprite, nevada AVIRIS 1995 data usgs. `https://speclab.cr.usgs.gov/cuprite95.tgif.2.2um_map.gif`. (accessed on 11 January 2017).

[3] Datasets and ground truths. `http://www.escience.cn/people/feiyunZHU/Dataset_GT.html`. (accessed on 10 November 2017).

[4] Figure 1 presence of mixed pixels in remotely sensed hyperspectral images. `https://asp-eurasipjournals.springeropen.com/articles/10.1186/1687-6180-2013-68`. (accessed on 12 January 2018).

[5] USGS digital spectral library 06. `https://speclab.cr.usgs.gov/spectral.lib06/`. (accessed on 10 June 2016).

[6] Ahmed, A. M., Duran, O., Zweiri, Y., and Smith, M. Hybrid spectral unmixing: using artificial neural networks for linear/non-linear switching. *Remote Sensing 9*, 8 (2017).

[7] Atkinson, I., Kamalabadi, F., and Jones, D. L. Wavelet-based hyperspectral image estimation. In *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, vol. 2 (Jul. 2003), 743–745 vol. 2.

[8] Berman, M., Kiiveri, H., Lagerstrom, R., Ernst, A., Dunne, R., and Huntington, J. F. Ice: a statistical approach to identifying endmembers in hyperspectral images. *IEEE Trans. Geosci. Remote Sens. 42*, 10 (Oct 2004), 2085–2095.

[9] Bioucas-Dias, J. M., and Figueiredo, M. A. T. Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing. In *Proc. 2nd WHISPERS* (June 2010), 1–4.

[10] Bioucas-Dias, J. M., Plaza, A., Camps-Valls, G., Scheunders, P., Nasrabadi, N. M., and Chanussot, J. Hyperspectral remote sensing data analysis and future challenges. *IEEE Geosci. Remote Sens. Magazine 1*, 2 (Jun. 2013), 6–36.

[11] Bioucas-Dias, J. M., Plaza, A., Dobigeon, N., Parente, M., Du, Q., Gader, P., and Chanussot, J. Hyperspectral unmixing overview: geometrical, statistical, and sparse regression-based approaches. *IEEE J.Sel.Topics Appl.Earth Observ. Remote Sens. 5*, 2 (April 2012), 354–379.

[12] Black, M., and Rangarajan, A. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *Springer Int. J. Comput. Vis. 19*, 1 (1996), 57–91.

[13] Blomgren, P., and Chan, T. Color TV: total variation methods for restoration of vector-valued images. *IEEE Trans. Image Process. 7*, 3 (Mar 1998), 304–309.

[14] Briggs, W. L., Henson, V. E., and McCormick, S. F. *A multigrid tutorial : second edition*. Soci. Industrial Applied Math., Philadelphia, PA, 2000.

[15] Buades, A., Coll, B., and Morel, J. A review of image denosing algorithms, with a new one. *SIAM Multiscale Model. Simul. 4*, 2 (2005), 490–530.

[16] Burt, P., and Adelson, E. The Laplacian pyramid as a compact image code. *IEEE Trans. Communi. 31*, 4 (Apr. 1983), 532–540.

[17] Camps-Valls, G., Tuia, D., Gmez-Chova, L., Jimnez, S., and Malo, J., Eds. *Remote Sensing Image Processing*. Morgan Claypool Publishers, LaPorte, CO, USA, Sept 2011.

[18] Candes, E. J., and Tao, T. Decoding by linear programming. *IEEE Trans. Inf. Theor. 51*, 12 (Dec. 2005), 4203–4215.

[19] Chan, T., Kang, S., and Shen, J. Total variation denoising and enhancement of color images based on the CB and HSV color models. *Elsevier J. Vis. Comun. Image Represent. 12*, 4 (Dec 2001), 422–435.

[20] Chang, C.-I. *Hyperspectral data processing, algorithm design and analysis*. Wiley, Hoboken, NJ, USA, Mar. 2013.

[21] Chen, S., Donoho, D., and Saunders, M. Atomic decomposition by basis pursuit. *SIAM review 43*, 1 (2001), 129159.

[22] Chen, X., Chen, J., Jia, X., Somers, B., Wu, J., and Coppin, P. A quantitative analysis of virtual endmembers' increased impact on the collinearity effect in spectral unmixing. *Trans. Geosci. Remote Sens. 49*, 8 (Aug 2011), 2945–2956.

[23] Chierchia, G., Pustelnik, N., Pesquet-Popescu, B., and Pesquet, J. C. A nonlocal structure tensor-based approach for multicomponent image recovery problems. *IEEE Trans. Image Process 23*, 12 (Dec 2014), 5531–5544.

[24] Combettes, P. L., and Pesquet, J. C. Proximal splitting methods in signal processing. In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, Springer (New York, 2011), 185–212.

[25] Dabov, K., Foi, A., and Egiazarian, K. Video denoising by sparse 3D transform-domain collaborative filtering. In *Proc. European Conf. Signal Process.* (Sep. 2007), 145–149.

[26] Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K. Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space. In *Proc. IEEE Int. Conf. Image Process. (ICIP)*, vol. 1 (Sept 2007), I–313–I–316.

[27] Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Process. 16*, 8 (Aug. 2007), 2080–2095.

[28] Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K. Image restoration by sparse 3d transform-domain collaborative filtering. vol. 6812 (2008), 681207–681207–12.

[29] Demir, B., and Ertürk, S. Improved hyperspectral image classification with noise reduction pre-process. In *Proc. European Conf. Signal Process.* (Aug 2008), 1–4.

[30] Eldar, Y., and Rauhut, H. Average case analysis of multichannel sparse recovery using convex relaxation. *IEEE Trans. Inf. Theory 56*, 1 (Jan 2010), 505519.

[31] Esmaeili Salehani, Y., Gazor, S., Kim, I.-M., and Yousefi, S. ℓ_0-norm sparse hyperspectral unmixing using arctan smoothing. *Remote Sensing 8*, 3 (2016).

[32] Fattal, R. Dehazing using color-lines. *ACM Trans. Graph. 34*, 1 (Dec. 2014), 13:1–13:14.

[33] Friedland, S., and Lim, L.-H. Nuclear norm of higher-order tensors, Sep 2014.

[34] Ghasrodashti, E. K., Karami, A., Heylen, R., and Scheunders, P. Spatial resolution enhancement of hyperspectral images using spectral unmixing and bayesian sparse representation. *Remote Sensing 9*, 6 (2017).

[35] Giampouras, P. V., Themelis, K. E., Rontogiannis, A. A., and Koutroumbas, K. D. Simultaneously sparse and low-rank abundance matrix estimation for hyperspectral image unmixing. *IEEE. Trans. Geosci. Remote Sens. 54*, 8 (Aug 2016), 4775–4789.

[36] Heiden, U., Iwasaki, A., , A., Schlerf, M., Udelhoven, T., Uto, K., Yokoya, N., and Chanussot, J. Foreword to the special issue on hyperspectral remote sensing and imaging spectroscopy. 3904–3908.

[37] Iordache, M. D., Bioucas-Dias, J. M., and Plaza, A. Sparse unmixing of hyperspectral data. *IEEE Trans. Geosci. Remote Sens. 49*, 6 (June 2011), 2014–2039.

[38] Iordache, M. D., Bioucas-Dias, J. M., and Plaza, A. Total variation spatial regularization for sparse hyperspectral unmixing. *IEEE Trans. Geosci. Remote Sens. 50*, 11 (Nov 2012), 4484–4502.

[39] Iordache, M. D., Bioucas-Dias, J. M., and Plaza, A. Collaborative sparse regression for hyperspectral unmixing. *IEEE Trans. Geosci. Remote Sens. 52*, 1 (Jan 2014), 341–354.

[40] Iordache, M.-D., and Plaza, A. J. A sparse regression approach to hyperspectral unmixing (2011).

[41] Jia, S., and Qian, Y. Spectral and spatial complexity-based hyperspectral unmixing. *IEEE Trans. Geosci. Remote Sens. 45*, 12 (Dec 2007), 3867–3879.

[42] Jia, S., and Qian, Y. Constrained nonnegative matrix factorization for hyperspectral unmixing. *IEEE Trans. Geosci. Remote Sens. 47*, 1 (Jan 2009), 161–173.

[43] Keshava, N., and Mustard, J. F. Spectral unmixing. *IEEE Signal Process. Mag. 19*, 1 (Jan 2002), 44–57.

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

[45] Lee, D. D., and Seung, H. S. Learning the parts of objects by nonnegative matrix factorization. *Nature 401*, 6755 (1999), 788–791.

[46] Li, C., Ma, Y., Mei, X., Liu, C., and Ma, J. Hyperspectral unmixing with robust collaborative sparse regression. *Remote Sensing 8*, 7 (2016).

[47] Licciardi, G. A., Sellitto, P., Piscini, A., and Chanussot, J. Nonlinear spectral unmixing for the characterisation of volcanic surface deposit and airborne plumes from remote sensing imagery. *Geosciences 7*, 3 (2017).

[48] Liu, R., Du, B., and Zhang, L. Hyperspectral unmixing via double abundance characteristics constraints based nmf. *Remote Sensing 8*, 6 (2016).

[49] Maggioni, M., Boracchi, G., Foi, A., and Egiazarian, K. Video denoising, deblocking, and enhancement through separable 4-D nonlocal spatiotemporal transforms. *IEEE Trans. Image Process. 21*, 9 (Sept 2012), 3952–3966.

[50] Manjon, J. V., Coupe, P., and Buades, A. MRI noise estimation and denoising using non-local PCA. *Elsevier Medical Image Anal. 22*, 1 (May 2015), 35–47.

[51] Moreau, J. J. Fonctions convexes duales et points proximaux dans un espace hilbertien. *CR Acad. Sci. Paris S'er. A Math 255* (1962), 2897–2899.

[52] Nascimento, J. M. P., and Dias, J. M. B. Vertex component analysis: a fast algorithm to unmix hyperspectral data. *IEEE Trans. Geosci. Remote Sens. 43*, 4 (April 2005), 898–910.

[53] Omer, I., and Werman, M. Color lines: image specific color representation. In *Proc. IEEE Conf. Comput. Vis. Pattern. Recognit. (CVPR)*, vol. 2 (Jun. 2004), II–946–II–953 Vol. 2.

[54] Omer, I., and Werman, M. Using natural image properties as demosaicing hints. In *Proc. IEEE Int. Conf. Image Process. (ICIP)*, vol. 3 (Oct. 2004), 1665–1670 Vol. 3.

[55] Ono, S., and Yamada, I. A convex regularizer for reducing color artifact in color image recovery. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)* (Jun. 2013), 1775–1781.

[56] Ono, S., and Yamada, I. Color-line regularization for color artifact removal. *IEEE Trans. Comp. Imaging 2*, 3 (Sept 2016), 204–217.

[57] Pati, Y., Rezahfar, R., and Krishnaprasad, P. Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. In *Proc. the 27th Annual Asilomar Conference on Signals, Systems and Computers* (2003).

[58] Pauca, V. P., Piper, J., and Plemmons, R. J. Nonnegative matrix factorization for spectral data analysis. *Linear Algebra Appl. 416*, 1 (2006), 29 – 47. Special Issue devoted to the Haifa 2005 conference on matrix theory.

[59] Pires, B. R., Singh, K., and Moura, J. M. F. Approximating image filters with box filters. In *Proc. IEEE Int. Conf. Image Process. (ICIP)* (Sep. 2011), 85–88.

[60] Plaza, J., Hendrix, E. M. T., García, I., Martín, G., and Plaza, A. On endmember identification in hyperspectral images without pure pixels: a comparison of algorithms. *J Math Imaging Vis. 42*, 2 (Feb 2012), 163–175.

[61] Qu, Q., Nasrabadi, N. M., and Tran, T. D. Abundance estimation for bilinear mixture models via joint sparse and low-rank representation. *IEEE. Trans. Geosci. Remote Sens. 52*, 7 (July 2014), 4404–4423.

[62] Rizkinia, M., Baba, T., Shirai, K., and Okuda, M. Local spectral component decomposition for multi-channel image denoising. *IEEE Trans. Image Process 25*, 7 (July 2016), 3208–3218.

[63] Rizkinia, M., and Okuda, M. Local abundance regularization for hyperspectral sparse unmixing. In *2016 APSIPA-ASC* (Dec 2016), 1–6.

[64] Shippert, P. Why use hyperspectral imagery. *Photogramm. Eng. Remote Sens. 70*, 4 (Apr 2004), 377–396.

[65] Shirai, K., Okuda, M., and Ikehara, M. Color-line vector field and local color component decomposition for smoothing and denoising of color images. In *Proc. IAPR Int. Conf. Pattern Recognit. (ICPR)* (Nov 2012), 3050–3053.

[66] Skauli, T. Sensor noise informed representation of hyperspectral data, with benefits for image storage and processing. *Optics Express 19*, 14 (Jul. 2011), 13031–13046.

[67] Song, M., Ma, Q., An, J., and Chang, C.-I. An improved nmf algorithm based on spatial and abundance constraints. In *2016 PIERS* (Aug 2016), 4532–4537.

[68] Tang, W., Shi, Z., Wu, Y., and Zhang, C. Sparse unmixing of hyperspectral data using spectral a priori information. *IEEE Trans. Geosci. Remote Sens. 53*, 2 (Feb 2015), 770–783.

[69] Themelis, K. E., Rontogiannis, A. A., and Koutroumbas, K. Semi-supervised hyperspectral unmixing via the weighted lasso. In *in Proc. 2010 IEEE ICASSP* (2010).

[70] Tschumperlé, D., and Deriche, R. Vector-valued image regularization with PDEs: a common framework for different applications. *IEEE Trans. Pattern Anal. Machine Intelli. (PAMI) 27*, 4 (Apr. 2005), 506–517.

[71] Tu, W. C., Tsai, C. L., and Chien, S. Y. Collaborative noise reduction using color-line model. In *Proc. IEEE Int. Conf. Acoustics Speech Signal Process. (ICASSP)* (May 2014), 2465–2469.

[72] Yang, J., Zhao, Y. Q., Chan, J. C. W., and Kong, S. G. Coupled sparse denoising and unmixing with low-rank constraint for hyperspectral image. *IEEE. Trans. Geosci. Remote Sens. 54*, 3 (March 2016), 1818–1833.

[73] Yokoya, N., Chanussot, J., and Iwasaki, A. Nonlinear unmixing of hyperspectral data using semi-nonnegative matrix factorization. *IEEE Trans. Geosci. Remote Sens. 52*, 2 (Feb 2014), 1430–1437.

[74] Yuen, P. W., and Richardson, M. An introduction to hyperspectral imaging and its application for security, surveillance and target acquisition. *The Imaging Sci. J. 58*, 5 (2010), 241–253.

[75] Zhang, B., Zhuang, L., Gao, L., Luo, W., Ran, Q., and Du, Q. Pso-em: a hyperspectral unmixing algorithm based on normal compositional model. *IEEE Trans. Geosci. Remote Sens. 52*, 12 (Dec 2014), 7782–7792.

[76] Zhang, H., He, W., Zhang, L., Shen, H., and Yuan, Q. Hyperspectral image restoration using low-rank matrix recovery. *IEEE. Trans. Geosci. Remote Sens. 52*, 8 (Aug 2014), 4729–4743.

[77] Zhang, S., Li, J., Liu, K., Deng, C., Liu, L., and Plaza, A. Hyperspectral unmixing based on local collaborative sparse regression. *IEEE. Trans. Geosci. Remote Sens. Letters 13*, 5 (May 2016), 631–635.

[78] Zhao, Y., and Yang, J. Hyperspectral image denoising via sparsity and low rank. In *Proc. 2013 IEEE IGARSS* (July 2013), 1091–1094.

[79] Zhao, Y. Q., and Yang, J. Hyperspectral image denoising via sparse representation and low-rank constraint. *IEEE Trans. Geosci. Remote Sens. 53*, 1 (Jan. 2015), 296–308.

[80] Zhao, Y. Q., and Yang, J. Hyperspectral image denoising via sparse representation and low-rank constraint. *IEEE. Trans. Geosci. Remote Sens. 53*, 1 (Jan 2015), 296–308.

[81] Zhong, Y., Feng, R., and Zhang, L. Non-local sparse unmixing for hyperspectral remote sensing imagery. *IEEE J.Sel.Topics Appl.Earth Observ. Remote Sens. 7*, 6 (June 2014), 1889–1909.

[82] Zhu, F., Wang, Y., Fan, B., Meng, G., and Pan, C. Effective spectral unmixing via robust representation and learning-based sparsity. *CoRR abs/1409.0685* (2014).

[83] Zhu, F., Wang, Y., Fan, B., Meng, G., Xiang, S., and Pan, C. Spectral unmixing via data-guided sparsity. *CoRR abs/1403.3155* (2014).

[84] Zhu, F., Wang, Y., Xiang, S., Fan, B., and Pan, C. Structured sparse method for hyperspectral unmixing. *ISPRS Journal of Photogrammetry and Remote Sensing 88* (2014), 101–118.

[85] Zortea, M., and Plaza, A. Spatial preprocessing for endmember extraction. *IEEE Trans. Geosci. Remote Sens. 47*, 8 (Aug 2009), 2679–2693.