The Pennsylvania State University

The Graduate School

College of Engineering

AN EVALUATION OF THREE ENDMEMBER EXTRACTION ALGORITHMS:

ATGP, ICA-EEA, AND VCA

A Thesis in

Electrical Engineering

by

Isaac D. Gerg

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

May 2008

The thesis of Isaac D. Gerg. was reviewed and approved* by the following:

Richard L. Tutwiler
Associate Professor of Acoustics
Thesis Advisor

Timothy J. Kane
Professor of Electrical Engineering

John F. Doherty
Professor of Electrical Engineering

W. Kenneth Jenkins
Professor of Electrical Engineering
Head of the Electrical Engineering Department

* Signatures are on file in the Graduate School

Abstract

Hyperspectral unmixing is necessary for material abundance map (MAM) creation. Most unmixing algorithms search for a set of pure pixels, called endmembers, of which all other pixels in the image are linear combinations of this set. Hence, endmember extraction is an important process in the creation of useful material abundance maps. Material abundance maps are created using a three step process: estimation of the number of materials in an image, unmixing of the image to determine the spectral signatures (endmembers) of these fundamental materials, and finally, some type of constrained least squares algorithm using the recovered endmember spectra to generate the abundance maps. This thesis formally evaluates the first two steps of this process using simulated and real data. First, a common material count estimation algorithm, known as virtual dimensionality (VD), is examined. Second, three endmember extraction algorithms are evaluated: Automatic Target Generation Process (ATGP), ICA-Based Endmember Extraction Algorithm (ICA-EEA) and Vertex Component Analysis (VCA). Finally, the derivation of a constrained least squares technique is given from which the results of the first two steps are used as input to create abundance maps on real data from the AVIRIS sensor. This process serves as a means of qualitatively evaluating the efficacy of these algorithms on live data from which ground truth information could not be realized. The three unmixing algorithms arise from different schools of thought. The ICA-EEA uses independent component analysis (ICA) to isolate endmembers present in a scene. Conversely, ATGP and VCA work on the principle of orthogonality in that endmembers are extracted by iteratively projecting the data orthogonally to the current span of detected endmembers. The two methods differ in how they select the next orthogonal direction. Additionally, the authors of VCA state ICA based methods cannot perform well due to the *sum-to-one* constraint in linear endmember mixing. The results of this thesis prove otherwise. All methods require that *pure* pixels are present in the scene.

Table of Contents

List of Tables viii

List of Figures xi

Acknowledgments xii

A Note on Notation xiii

Orthorectification xiv

1 Introduction 1

1.1 Utility of Hyperspectral Data 1

1.2 Linear Mixing Model 4

1.3 Nonlinear Mixing Model 5

1.4 Thesis Overview 7

2 Virtual Dimensionality 8

2.1 Harsanyi-Farrand-Chang Algorithm 8

3 Automatic Target Generation Procedure 11

3.1 The ATGP Algorithm 11

4 Independent Component Analysis - Endmember Extraction Algorithm (ICA-EEA) 13

4.1 Independent Component Analysis (ICA) 13

4.1.1 Introduction 13

4.1.2 The Cocktail Party Problem 14

4.1.3 The FastICA Algorithm 14

4.2 Independent Component Analysis Endmember Extraction Algorithm (ICA-EEA) 19

5 Vertex Component Analysis 21

5.1 The VCA Algorithm 21

5.2 Similarities to the Pixel Purity Index Algorithm 23

5.3 Similarities to the ATGP Algorithm 23

5.4 Computational Complexity 24

6 Fully Constrained Least Squares 25

6.1 Least Squares With No Constraints 25

6.2 Least Squares With Sum-to-One Constraint (ASC) 26

6.3 Least Squares with Sum-to-One & Nonnegativity Constraints (ASC+ANC) 27

7 Experiment **28**

7.1 Definition of Signal-to-Noise Ratio 28

7.2 AVIRIS Noise Model 29

7.3 Test Data Generation 30

7.4 Estimating the Number of Materials Present In a Scene 32

7.5 Monte-Carlo Analysis of Algorithms 35

7.6 Endmember Matching 35

7.7 Datasets Tested 35

7.7.1 Synthetic Mixture Database Endmembers (SMDE) Dataset . . 36

7.7.2 Random Mixture Database Endmembers (RMDE) Dataset . . 36

7.7.3 Random Mixture Random Endmembers (RMRE) Dataset . . 36

7.7.4 Random Mixture Database Endmembers with Illumination Variation (RMDEI) Dataset 37

7.7.5 Moffett Field, CA AVIRIS Dataset 38

8 Results **39**

8.1 Virtual Dimensionality Algorithm 41

8.1.1 Synthetic Mixture, Database Endmembers Dataset 41

8.1.2 Random Mixture, Database Endmembers Dataset 42

8.1.3 Random Mixture, Database Endmembers with Varying Illumination Dataset 43

8.1.4 Random Mixture, Random Endmembers Dataset 44

8.1.5 Moffett Field, CA Dataset 45

8.2 Automatic Target Generation Procedure 46

8.2.1 Synthetic Mixture, Database Endmembers Dataset 46

8.2.2 Random Mixture, Database Endmembers Dataset 47

8.2.3 Random Mixture, Random Endmembers Dataset 48

8.2.4 Random Mixture, Database Endmembers with Varying Illumination Dataset 49

8.2.5 Synthetic Mixture, Database Endmembers Dataset Using AVIRIS SNR Model 50

8.2.6 Random Mixture, Database Endmembers Dataset Using AVIRIS SNR Model 50

8.2.7 Random Mixture, Random Endmembers Dataset Using AVIRIS SNR Model 50

8.2.8 Random Mixture, Database Endmembers with Varying Illumination Dataset Using AVIRIS SNR Model 50

8.2.9 Moffett Field, CA Dataset 50

8.3 ICA-EEA Algorithm 51

8.3.1 Synthetic Mixture, Database Endmembers Dataset 51

8.3.2 Random Mixture, Database Endmembers Dataset 52

8.3.3 Random Mixture, Random Endmembers Dataset 53

8.3.4 Random Mixture, Database Endmembers with Varying Illumination Dataset 54

8.3.5 Synthetic Mixture, Database Endmembers Dataset Using AVIRIS SNR Model 55
8.3.6 Random Mixture, Database Endmembers Dataset Using AVIRIS SNR Model 55
8.3.7 Random Mixture, Random Endmembers Dataset Using AVIRIS SNR Model 56
8.3.8 Random Mixture, Database Endmembers with Varying Illumination Dataset Using AVIRIS SNR Model 56
8.3.9 Moffett Field, CA Dataset 56
8.4 VCA-Authors Algorithm 57
8.4.1 Synthetic Mixture, Database Endmembers Dataset 57
8.4.2 Random Mixture, Database Endmembers Dataset 58
8.4.3 Random Mixture, Random Endmembers Dataset 59
8.4.4 Random Mixture, Database Endmembers with Varying Illumination Dataset 60
8.4.5 Synthetic Mixture, Database Endmembers Dataset Using AVIRIS SNR Model 61
8.4.6 Random Mixture, Database Endmembers Dataset Using AVIRIS SNR Model 61
8.4.7 Random Mixture, Random Endmembers Dataset Using AVIRIS SNR Model 62
8.4.8 Random Mixture, Database Endmembers with Varying Illumination Dataset Using AVIRIS SNR Model 62
8.5 VCA-Raw Pixels Algorithm 63
8.5.1 Synthetic Mixture, Database Endmembers Dataset 63
8.5.2 Random Mixture, Database Endmembers Dataset 64
8.5.3 Random Mixture, Random Endmembers Dataset 65
8.5.4 Random Mixture, Database Endmembers with Varying Illumination Dataset 66
8.5.5 Synthetic Mixture, Database Endmembers Dataset Using AVIRIS SNR Model 67
8.5.6 Random Mixture, Database Endmembers Dataset Using AVIRIS SNR Model 67
8.5.7 Random Mixture, Random Endmembers Dataset Using AVIRIS SNR Model 68
8.5.8 Random Mixture, Database Endmembers with Varying Illumination Dataset Using AVIRIS SNR Model 68
8.5.9 Moffett Field, CA Dataset 68

9 Discussion 99
9.1 Algorithm Consistency Problem 99
9.2 Algorithm Initialization Sensitivity 99
9.3 Algorithm Complexity 99
9.4 ATGP 100
9.5 ICA-EEA 100

9.6 VCA 101
9.7 Moffett Field, CA Dataset 101

10 Conclusion 103
10.1 Utility of This Work 103
10.2 Future Work 103

A AVIRIS SNR Versus Wavelength 109

B Affinity Matrices of Endmembers for SMDE Dataset 115

C Histograms of Endmember Affinity for SMDE Dataset 116

D Affinity Matrices of Endmembers for RMDE Dataset 117

E Histograms of Endmember Affinity for RMDE Dataset 118

F Affinity Matrices of Endmembers for RMRE Dataset 119

G Histograms of Endmember Affinity for RMRE Dataset 120

H Affinity Matrices of Endmembers for RMDEI Dataset 121

I Histograms of Endmember Affinity for RMDEI Dataset 122

List of Tables

1 Parameters of Moffett Field, CA AVIRIS Collected Dataset 38

2 List of dataset abbreviations. 39

3 List of all experiments conducted and their associated results table. . 40

4 Results of the VD algorithm for five false alarm rates(P_f). The white faceted plane represents the true number of materials in the data cube. The horizontal axis represents the number materials in the simulated data cube. The into-the-page axis represents the SNR of the simulated data cube. The vertical axis represent the number of materials estimated in the data cube. 41

5 Results of the VD algorithm for five false alarm rates(P_f). The white faceted plane represents the true number of materials in the data cube. The horizontal axis represents the number materials in the simulated data cube. The into-the-page axis represents the SNR of the simulated data cube. The vertical axis represent the number of materials estimated in the data cube. 42

6 Results of the VD algorithm for five false alarm rates(P_f). The white faceted plane represents the true number of materials in the data cube. The horizontal axis represents the number materials in the simulated data cube. The into-the-page axis represents the SNR of the simulated data cube. The vertical axis represent the number of materials estimated in the data cube. 43

7 Results of the VD algorithm for five false alarm rates(P_f). The white faceted plane represents the true number of materials in the data cube. The horizontal axis represents the number materials in the simulated data cube. The into-the-page axis represents the SNR of the simulated data cube. The vertical axis represent the number of materials estimated in the data cube. 44

8 Results of the VD algorithm run upon the Moffett Field, CA dataset for five false alarm rates(P_f). The number of materials estimated by the algorithm is indicated by $\hat{q}$. 45

9 Results of the ATGP algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 46

10 Results of the ATGP algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 47

11 Results of the ATGP algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 48

12 Results of the ATGP algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 49

13 Results of the ATGP algorithm on the SMDE dataset using the AVIRIS SNR model. 50
14 Results of the ATGP algorithm on the RMDE dataset using the AVIRIS SNR model. 50
15 Results of the ATGP algorithm on the RMRE dataset using the AVIRIS SNR model. 50
16 Results of the ATGP algorithm on the RMDEI dataset using the AVIRIS SNR model. 51
17 Results of the ICA-EEA algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 51
18 Results of the ICA-EEA algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 52
19 Results of the ICA-EEA algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 53
20 Results of the ICA-EEA algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 54
21 Results of the ICA-EEA algorithm on the SMDE dataset using the AVIRIS SNR model. 55
22 Results of the ICA-EEA algorithm on the RMDE dataset using the AVIRIS SNR model. 55
23 Results of the ICA-EEA algorithm on the RMRE dataset using the AVIRIS SNR model. 56
24 Results of the ICA-EEA algorithm on the RMDEI dataset using the AVIRIS SNR model. 56
25 Results of the VCA-Authors algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 57
26 Results of the VCA-Authors algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 58
27 Results of the VCA-Authors algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 59
28 Results of the VCA-Authors algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 60
29 Results of the VCA-Authors algorithm on the SMDE dataset using the AVIRIS SNR model. 61
30 Results of the VCA-Authors algorithm on the RMDE dataset using the AVIRIS SNR model. 61

31 Results of the VCA-Authors algorithm on the RMRE dataset using the AVIRIS SNR model. 62
32 Results of the VCA-Authors algorithm on the RMDEI dataset using the AVIRIS SNR model. 62
33 Results of the VCA-Raw Pixels algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 63
34 Results of the VCA-Raw Pixels algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 64
35 Results of the VCA-Raw Pixels algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 65
36 Results of the VCA-Raw Pixels algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube. 66
37 Results of the VCA-Raw Pixels algorithm on the SMDE dataset using the AVIRIS SNR model. 67
38 Results of the VCA-Raw Pixels algorithm on the RMDE dataset using the AVIRIS SNR model. 67
39 Results of the VCA-Raw Pixels algorithm on the RMRE dataset using the AVIRIS SNR model. 68
40 Results of the VCA-Raw Pixels algorithm on the RMDEI dataset using the AVIRIS SNR model. 68

List of Figures

1 Rooftop likelihood map fused with map data provided by Google Earth. Scene is of Moffett Field, CA 2

2 Synthetic color image formed by three spectral bands at visible wavelengths. 3

3 Synthetic color image formed by three spectral bands at infrared (IR) wavelengths. 3

4 50% reflectance endmembers with various levels of noise added. . . . 29

5 Angle error verses SNR for a fifty percent reflectance signature. . . . 30

6 AVIRIS SNR at each sensor band sampled from 400-2500nm. Measured 2005 [17]. 31

7 Original color (RGB) image. 32

8 Original color image converted to the HSV colorspace. 33

9 Hue portion of HSV image. 33

10 Clustering of hue portion of HSV image. 33

11 Simulated Abundance Maps 34

12 Mixed and downsampled image hyperspectral image. 34

13 Decorrelation Stretched Image of Moffett Field, CA Dataset 38

14 Abundance map of Sagebrush over Moffett Field, CA overlayed onto Google Earth. 104

15 Abundance map of Lichen over Moffett Field, CA overlayed onto Google Earth. 105

16 Abundance map of Oak Tree over Moffett Field, CA overlayed onto Google Earth. 106

17 Abundance map of Antigorite over Moffett Field, CA overlayed onto Google Earth. 107

Acknowledgments

Thank you again Rick for supervising another thesis of mine! You introduced me to image processing during the fall semester of 2004 and to the problem of hyperspectral unmixing during the spring semester of 2006. I am very thankful for this! Thank you Dr. Kane for serving on my committee and enlightening me to the many other aspects of remote sensing (i.e. the atmosphere!). I've had a great experience in your Remote Sensing of Earth Systems class and hope it continues to grow and turns into a yearly event. Thank you Dr. Doherty for serving on my committee and providing inputs to improve the quality of my writing and linear algebra skills (whoops!). You all have been selfless, motivating, but most importantly, inspirational. With the full work schedule and perils of classes, you have all supported me so much and I humbly thank you for it.

I would like to thank Michael Eastwood and Robert Green of the NASA Jet Propulsion Laboratory for providing me with the AVIRIS SNR data captured in Appendix A.

I would like to thank my family; you have loved me, guided me, believed in me, and picked me up not a moment too late. Thank you! I am eternally grateful for all of your love: Mom, Dad, and Ian.

Finally, I thank my Lord and Savior, Jesus Christ. Without You, I would have never been able to hold it together working full time while going to school. You worked in me to complete the task You set out for me to do. Thank you for revealing Yourself to me during the spring of 2006. I am Yours.

I know that You can do all things, and that no purpose of Yours can be hindered. Job 42:2

A Note on Notation

Hyperspectral data is often expressed many ways to better describe the mathematical handling of the data; mainly as a vector of pixels when referring to the data in a space or a matrix of pixels when referring to data as an image.

For consistency, a common notation is defined to differentiate these concepts clearly. Hyperspectral data examined like an image will be defined as a matrix $\mathbf{M}_{m \times n \times p}$ of dimension $m \times n \times p$ where m is defined as the number of rows in the image, n is defined as the number of columns in the image, and p is defined as the number of bands in the image. Therefore, a single element of such an image will be accessed using $\mathbf{M}_{i,j,k}$ and a single pixel of an image will be accessed using $\mathbf{M}_{i,j,:}$ Hyperspectral data formed as a vector of vectors (i.e. 2D matrix) is defined as $\mathbf{M}_{(m \cdot n) \times p}$ of dimension $(m \cdot n) \times p$. A single element is accessed using $\mathbf{M}_{i,j}$ and a single pixel is accessed using $\mathbf{M}_{:,j}$. Notice the multi-element notation is consistent with Matlab^{TM}; this is intentional.

The list below provides a summary of the notation convention used throughout this thesis.

$\mathbf{M}$ Data matrix. Defined as an image of spectral signatures or vectors: $\mathbf{M}_{m \times n \times p}$. Or, defined as a long vector of spectral signatures: $\mathbf{M}_{(m \cdot n) \times p}$.

N The total number of pixels. For example $N = m \cdot n$.

m Number of rows in the image.

n Number of columns in the image.

p Number of bands.

q Number of classes / endmembers.

$\mathbf{U}$ Matrix of endmembers. Each column of the matrix represents an endmember vector.

$\mathbf{b}$ Observation vector; a single pixel.

$\mathbf{x}$ Weight vector. A matrix of weight vectors forms an abundance map.

Orthorectification

Often, the results of data analysis on hyperspectral imagery are more meaningful when fused with other data such as street maps from Google Earth. Images obtained from an airborne platform are often skewed due to the imaging geometry. This skewing process results in imagery which cannot resolve pixel distances linearly (i.e. a fixed pixel distance between two points does not correspond to a fixed ground distance.) This results in the inability to simply lay processed imagery over map data with accurate alignment. The process of *orthorectification* corrects for alignment unevenness due to the imaging geometry. Once corrected, a fixed distance d measured anywhere in the processed imagery will correspond to a fixed distance d' on the map.

An orthorectification algorithm is provided below to help obtain georegistered abundance map overlays such as that of figure 1. It is a simple model assuming the curvature of the Earth and relative terrain altitude are negligible. It also assumes the imagery is obtained at nadir with no sensor movement distortion. The overall ground sampling distance (GSD) is approximated using the GSD at nadir. The algorithm creates a mapping from x_g, the coordinates on the ground in meters, to x_p, the coordinates in the image in pixels.

Given:

- $\mathbf{I}$ is the input image with cross-track coordinates defined as $[-\frac{x_p}{2} \frac{x_p}{2}]$ in pixels.
- a is the distance from imager to target in meters.
- $\beta_{\frac{1}{2}}$ is the image field of view in radians.
- w is the width of the image in the cross-track direction in pixels.
- x_g is the distance from nadir in the cross track direction for which x_p is sought.

Compute:

1. $\alpha = \frac{\beta_{\frac{1}{2}}}{w}$; Radians per pixel.

2. $GSD = a \cdot \alpha$; Ground sampling distance (GSD) at nadir.

3. $\theta = atan(\frac{GSD \cdot x_g}{a})$; Angle of incidence.

4. $x_p = \frac{\theta}{\alpha}$

1 Introduction

1.1 Utility of Hyperspectral Data

Hyperspectral remote sensing (HSRS) typically implies bulk wavelength examination of the electromagnetic spectrum from wavelengths of approximately 400-2500nm with very good spectral resolution, usually on the order of ten nanometers. It is a passive observation system by which our sun provides incident power energy on a target which is reflected to a hyperspectral imaging sensor [1]. Due to its passive nature and the optical frequency band, HSRS works poorly or not at all in adverse weather conditions and only very partially at night (due to stored energy emitted at infrared (IR) wavelengths).

Hyperspectral imagery (HSI) provides additional knowledge about a scene when compared to traditional color imagery (CI). CI is restricted to wavelengths only the human eye can detect which is typically in the range of 400-700nm. The additional spectral range obtained through HSI gives each material a *spectral fingerprint* by which the material can be directly identified against a database or identified through spectral features such as absorbtion lines resulting from chemical processes such as photosynthesis. For example, in CI, one may crudely identify crops on farmland (corn, wheat, etc), but using HSI, one may enumerate exactly what type of crop is present and its agricultural characteristics such as health and harvest time. Another example is automatic target recognition (ATR); looking for a specific type of building holding a terrorist based on the roof type. Using CI, one can create a list of candidate buildings based on roof color. However, it is quickly realized that several types of roofing exhibit the same color, thus resulting in many false alarms. The field of search can be quickly trimmed by using HSI and eliminating candidate rooftops based on spectral correlation. The final results can be fused with map data to further determine the true target. Figure 1 depicts an example target map fused with road data from

Google Earth. The map depicts the likelihood of the desired roof target being present at a particular pixel.

HSRS differs from traditional multispectral remote sensing by the spectral resolution and the number of spectral bands sampled used during the observation process. Typically, HSI is composed of over fifty spectral bands with spectral resolutions generally under 100nm. Similarly to CI, the extra amount of spectral bands acquired in HSI allow for greater pixel discrimination than traditional multi spectral imagery (MSI).

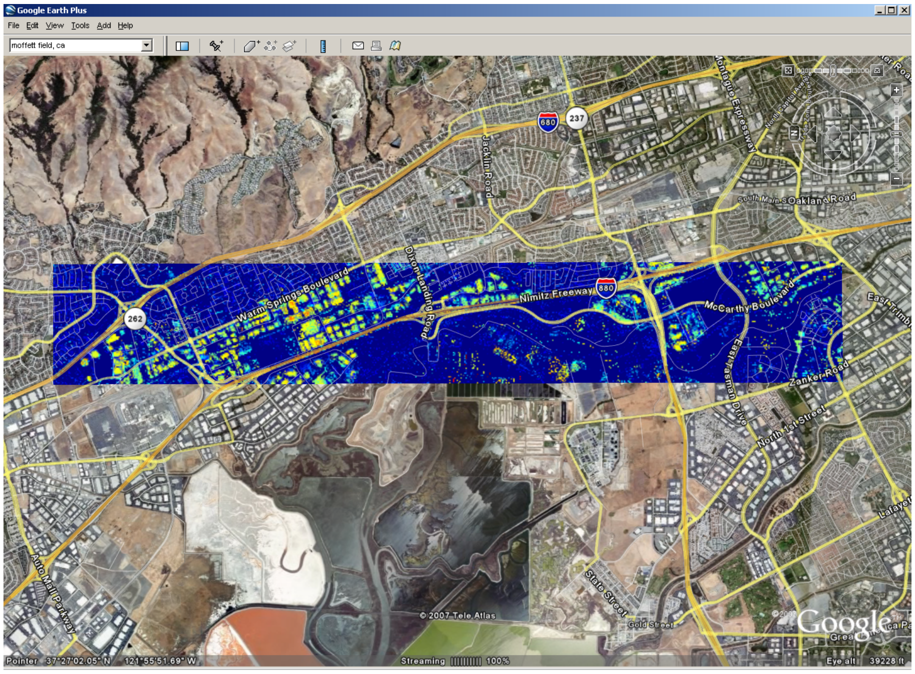


Figure 1: Rooftop likelihood map fused with map data provided by Google Earth. Scene is of Moffett Field, CA

Figure 2 depicts a synthetic color image of Moffett Field, CA obtained from the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) [2] using visible wavelengths. Figure 3 depicts the same scene but obtained at IR wavelengths. Notice similar colored areas of the synthetic color image are different in color in the IR image. This demonstrates the additional discrimination power of HSI.

The radiance of the VR and SWIR regions at the sensor is composed of three

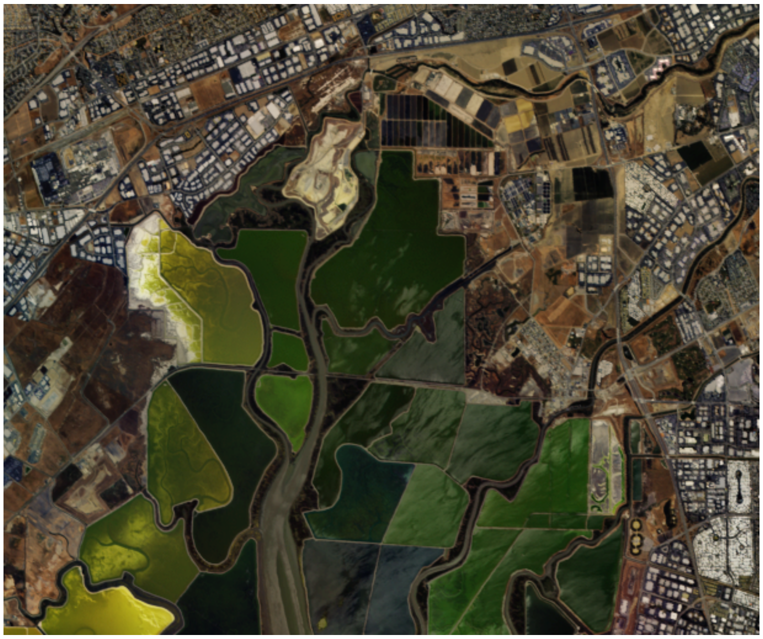


Figure 2: Synthetic color image formed by three spectral bands at visible wavelengths.

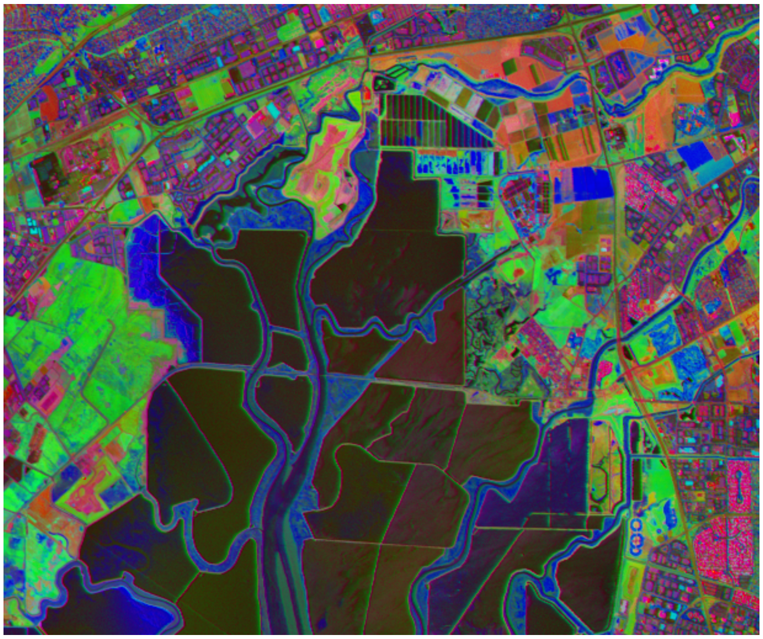


Figure 3: Synthetic color image formed by three spectral bands at infrared (IR) wavelengths.

radiation components [1]: L_λ^{su} the unscattered surface-reflected radiation, L_λ^{sd} the down-scattered surface-reflected skylight, and L_λ^{sp} the path-scattered radiation due to Rayleigh and Mie scattering. Together, these three components form the total upwelling radiance incident upon the sensor.

$$L_\lambda^{s} = L_\lambda^{su} + L_\lambda^{sd} + L_\lambda^{sp} \tag{1}$$

Often L_λ^{sd} and L_λ^{sp} make negligible contributes to L_λ^{s} when conditions are clear.

1.2 Linear Mixing Model

Spatial resolution in HSI is usually low ($GSD > 20m$) resulting in many materials observed through a single pixel. This phenomena results in *spectral mixing* whereby the resultant spectral signature is a mixture of individual material spectra. The linear mixture model (LMM) ignores the contributions of multi-path scattering and non-material incident radiation (L_λ^{sd} and L_λ^{sp}) due to the difficulty in estimating these components. Instead, it models the resultant spectral signature as a linear combination of individual material signatures weighted by their percentage area in the pixel. The percentage area of a pixel for a particular material is called the material's *abundance*. The LMM implies the abundances, also called *weights*, of each material must sum-to-one and be nonnegative. These constraints are known as the abundance sum-to-one constraint (ASC) and the abundance nonnegativity constraint (ANC) respectively.

One must determine the signatures present and their respective abundances to create a solution to the LMM. This thesis deals directly with this problem by providing methods to:

1. Estimate the number of materials present in an HSI scene.

2. Estimate the spectral signatures comprising the LMM.

3. Estimate the abundance of each material comprising the LMM.

The process of determining these quantities is known as *hyperspectral unmixing.* The spectral signatures determined using such methods are known as *endmembers.*

Mathematically, the LMM is written as

$$\mathbf{b} = \mathbf{U}\mathbf{x} + \epsilon \tag{2}$$

where $\mathbf{b}$ is an $p \times 1$ observation vector, $\mathbf{U}$ is a $p \times n$ column-wise endmember matrix, and $\mathbf{x}$ is a $p \times 1$ endmember weight vector. The vector $\mathbf{x}$ observes the following constraints

$$\sum_i \mathbf{x}_i = 1 \tag{3}$$

$$\mathbf{x}_i \geq 0, \forall i \tag{4}$$

Equations (3) and (4) represent the ASC and ANC constraints respectively.

As an example, say a pixel is composed of three materials (asphalt, grass, and tree) with the following respective abundances: 50%, 20%, and 30%. The LMM describes this observed pixel as

$$\mathbf{m}_{observed} = 0.5 \cdot \mathbf{u}_{asphalt} + 0.2 \cdot \mathbf{u}_{grass} + 0.3 \cdot \mathbf{u}_{tree} + \epsilon \tag{5}$$

1.3 Nonlinear Mixing Model

Spectral mixing in reality is not as simple as the LMM. Multiple scattering mechanisms are always present in addition to linear mixing due to atmospheric affects. This results in nonlinear mechanisms affecting the resultant spectra. The mixing model accounting for these nonlinear scattering mechanisms is called the nonlinear mixing model (NLMM). It is difficult to create solutions for this model due to all the physical phenomena occurring and the dependence of these scattering mechanisms on wavelength.

Simple methods are needed to model the NLMM due to the difficultly of accurately accounting for all phenomena. One such method is to model the NLMM as the LMM but to relax the ASC constraint by making the abundances sum to a number *near* one. One such way to achieve this is to sum the model to a number chosen from a strictly positive probability density function (pdf) with a mean of one. Ground truth knowledge can be used to determine suitable parameters for the distribution.

Mathematically, we model the NLMM similar to the LMM but adjust the constraints of $\mathbf{x}$ to reflect these processes. As mentioned, one way to model this is to simply make the sum of elements of $\mathbf{x}$ add up to a number which is chosen from a random distribution centered around one. Additionally, the nonnegativity constraint is still imposed. Mathematically, the NLMM is described similar to the LMM with the following modification to the ASC constraint of equation (3):

$$\sum_i \mathbf{x}_i = y \tag{6}$$

where y is chosen from a positive distribution centered around one. A bounded choice for y, which is what is used in this thesis, is the uniform distribution centered around one

$$y \sim U(0.9, 1.1) \tag{7}$$

As an example, say a pixel is composed of three materials (asphalt, grass, and tree) with the following respective abundances: 50%, 20%, and 30%. The NLMM describes this observed pixel as

$$\mathbf{m}_{observed} = \gamma(0.5 \cdot \mathbf{u}_{asphalt} + 0.2 \cdot \mathbf{u}_{grass} + 0.3 \cdot \mathbf{u}_{tree}) + \epsilon \tag{8}$$

where γ represents the nonlinearities of the observation process such as viewing angle (e.g. $\gamma = \cos(\theta)$) or multi-path scattering.

1.4 Thesis Overview

This thesis describes algorithms to assist in HSI unmixing. Specifically, this thesis examines

- the virtual dimensionality algorithm used to estimate the number of materials in a HSI datacube
- three HSI unmixing algorithms used to detect pure endmembers: ATGP, ICA-EEA, and VCA
- the use of the unmixing algorithms with a fully constrained least squares algorithm to generated material abundance maps on real data

The next few chapters will describe the above algorithms in detail. Following this, the experimental Monte-Carlo results of each algorithm are shown. Discussion of each algorithm's pros and cons as well as motivations for improvement are additionally provided. Finally, the conclusions of the research are presented. Appendices detailing the datasets and the SNR model used in the experiments are provided for clarity and verbosity.

2 Virtual Dimensionality

The virtual dimensionality (VD) of a hyperspectral dataset provides an estimate of the number of distinct endmembers in the cube. There are several algorithms available to determine the VD of an image. The authors of the algorithms in this thesis all compute VD using the Harsanyi-Farrand-Chang (HFC) method [3]. This method is attractive to use because it is simple to implement [4] and has shown to be effective when used on AVIRIS data [5].

2.1 Harsanyi-Farrand-Chang Algorithm

The algorithm begins by computing the sample correlation matrix, $\mathbf{R}_{p\times p}$, and the sample covariance matrix, $\mathbf{K}_{p\times p}$, of the hyperspectral data. Next, the eigenvalues of each of the matrices are computed. Let the correlation matrix eigenvalues be $\{\lambda_1^{\mathbf{R}}, \lambda_2^{\mathbf{R}}, ..., \lambda_p^{\mathbf{R}}\}$ and the covariance matrix eigenvalues be $\{\lambda_1^{\mathbf{K}}, \lambda_2^{\mathbf{K}}, ..., \lambda_p^{\mathbf{K}}\}$ where the eigenvalues are sorted in descending order. Recall the definitions of the correlation and covariance matrices [6]:

$$\begin{aligned} Corr(x,y) &= \mathbf{R} = E\{xy^*\} \quad (9) \\ Cov(x,y) &= \mathbf{K} = E\{xy^*\} - \overline{x}\overline{y^*} \quad (10) \end{aligned}$$

The covariance and correlation matrices are similar with the exception of the additional term in the covariance calculation. Thus, in the case of an image having q number of endmembers, we expect $\lambda_i^{\mathbf{R}} > \lambda_i^{\mathbf{K}}$ for $i = 1, ..., q$ and $\lambda_i^{\mathbf{R}} = \lambda_i^{\mathbf{K}}$ for $i = q+1, ..., p$ because the signals (endmembers in this case) are non-random and the noise is white with zero mean.

The HFC algorithm poses the determination of q as a binary hypothesis problem.

$$H_0 : z_i = \lambda_i^{\mathbf{R}} - \lambda_i^{\mathbf{K}} = 0 \tag{11}$$

$$H_1 : z_i = \lambda_i^{\mathbf{R}} - \lambda_i^{\mathbf{K}} > 0 \tag{12}$$

for $i = 1, ..., p$. There is an endmember contributing to the correlation eigenvalue in addition to the noise when H_1 is true. When no endmember is contributing the correlation eigenvalue H_0 is expected to be true.

In white Gaussian noise, $\lambda_i^{\mathbf{R}}$ and $\lambda_i^{\mathbf{K}}$ are modeled as Gaussian random variables with known distributions and thus the Neyman-Pearson hypothesis test cane be used for the model. The variance is defined as

$$\sigma_{z_l}^2 = Var[\lambda_i^{\mathbf{R}} - \lambda_i^{\mathbf{K}}] \tag{13}$$

$$= Var[\lambda_i^{\mathbf{R}}] + Var[\lambda_i^{\mathbf{K}}] - 2Cov(\lambda_i^{\mathbf{R}}, \lambda_i^{\mathbf{K}}) \tag{14}$$

for $l = 1, 2, 3, ..., p$. Thus, for each data dimension, a new $\sigma_{z_l}^2$ must be computed. Approximations can be used for the variances and covariances above when N is sufficiently large [3]. These approximations are the following

$$Var[\lambda_i^{\mathbf{K}}] \cong \frac{2}{N}(\lambda_i^{\mathbf{K}})^2 \tag{15}$$

$$Var[\lambda_i^{\mathbf{R}}] \cong \frac{2}{N}(\lambda_i^{\mathbf{R}})^2 \tag{16}$$

$$Cov(\lambda_i^{\mathbf{R}}, \lambda_i^{\mathbf{K}}) \cong \frac{2}{N}(\lambda_i^{\mathbf{R}}\lambda_i^{\mathbf{K}}) \tag{17}$$

The false alarm probability is now constructed as

$$P_f = \int_{\tau}^{\infty} p_0(z)dz \tag{18}$$

where $p_0(z)$ is a Gaussian ditribution with mean zero and variance $\sigma_{z_l}^2$ from equation

(14). τ is determined by selecting suitable values for P_f. Signal energy, and thus a unique endmember, is present when $\lambda_i^{\mathbf{R}} - \lambda_i^{\mathbf{K}} > \tau$ for a given band i.

The HFC algorithm is often examined for $P_f = 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}$, and 10^{-1}; however, an estimate of $P_f = 10^{-3}$ is often used in practice.

3 Automatic Target Generation Procedure

The Automatic Target Generation Procedure (ATGP) [7] algorithm works by iterative orthogonal projections of the data then finding the largest magnitude vector of this projection. There are two versions of the ATGP algorithm with the version used here being the automatic detection and classification algorithm (ATDCA). We will refer to this version of the algorithm as ATGP for simplicity.

3.1 The ATGP Algorithm

The ATGP algorithm begins with inputs $\mathbf{M}$ and q where $\mathbf{M}$ is a $p \times N$ data matrix and q is the desired number of endmember to detect. The output is a matrix $\mathbf{U}$ of detected endmembers.

1. Find the index of the pixel exhibiting largest magnitude.

$$i = \underset{x}{\operatorname{argmax}}\{|\mathbf{M}_{:,x}|\} \tag{19}$$

2. Initialize the column-wise endmember matrix to the pixel found in 1.

$$\mathbf{U} = \mathbf{M}_{:,i} \tag{20}$$

3. Find a vector $\mathbf{v}$ orthogonal to the current span of detected endmembers. This vector can be found by utilizing properties of the Moore-Penrose matrix inverse. The Moore-Penrose matrix inverse is defined as

$$\mathbf{U}^{+} = (\mathbf{U}^{T}\mathbf{U})^{-1}\mathbf{U}^{T} \tag{21}$$

and has the property of generating an orthogonal projection onto the nullspace of $\mathbf{U}$ by

$$\mathbf{P} = \mathbf{I} - \mathbf{U}\mathbf{U}^{+} \tag{22}$$

4. Project all the data onto the nullspace of $\mathbf{W}$, $\mathbf{P}$.

$$\mathbf{Y}_{:,i} = \mathbf{P}\mathbf{M}_{:,i} \tag{23}$$

   where $i = 1, 2, 3, ..., N$

5. Find the index of the pixel in $\mathbf{Y}$ exhibiting largest magnitude. $i = \text{argmax}_x\{|\mathbf{Y}_{:,x}|\}$

6. Augment the endmember matrix $\mathbf{U}$ with the new pixel.

$$\mathbf{U} = \begin{bmatrix} \mathbf{U} & \mathbf{Y}_{:,i} \end{bmatrix} \tag{24}$$

7. Go to step 3. Repeat until q endmembers are found.

It is worth noting the selection of the next endmember after projecting the data onto the orthogonal subspace $\mathbf{P}$. The ATGP algorithm chooses the next endmember to be the maximum length vector of this projection despite all the vectors being orthogonal to the $\mathbf{U}$. This fact denotes the ATGP algorithm as having preference for *high power* endmembers. We will show how this preference differs with the VCA algorithm.

4 Independent Component Analysis - Endmember Extraction Algorithm (ICA-EEA)

This section describes the independent component analysis endmember extraction algorithm (ICA-EEA). This is accomplished by first describing the independent components analysis algorithm (ICA), specifically the FastICA [8] algorithm, and how it is used to form ICA-EEA.

4.1 Independent Component Analysis (ICA)

4.1.1 Introduction

Often in the field of signal processing we seek to determine how two sources are mixed with the goal of unmixing them from randomly mixed (weighted) observations. Specifically, we model the mixing of two sources as

$$\mathbf{b} = \mathbf{U}\mathbf{x} + \epsilon \tag{25}$$

where $\mathbf{U}$ is a square mixing matrix, $\mathbf{x}$ is a column vector of sources, and $\mathbf{b}$ is a column vector of observations. We assume that $\mathbf{U}$ is full rank and therefore $\mathbf{U}^{-1}$ exists. Given $\mathbf{U}$ and $\mathbf{b}$ with $\epsilon = \mathbf{0}$, we can construct the original sources from the following equation

$$\mathbf{s} = \mathbf{U}^{-1}\mathbf{b} \tag{26}$$

Generally, $\mathbf{U}$ and $\mathbf{s}$ are unknown in many problems. However, an estimate of $\mathbf{x}$ can be determined if an estimate $\mathbf{U}$ can be realized. The estimates of $\mathbf{U}$ and $\mathbf{x}$ shall be denoted as $\hat{\mathbf{U}}$ and $\hat{\mathbf{x}}$ respectively.

The sources are independent in many source separation problems. The goal of independent component analysis (ICA) is to determine an unmixing matrix in such a way as to make the estimates of the sources as independent as possible

$$\hat{\mathbf{x}} = \hat{\mathbf{U}^{-1}}\mathbf{b} \tag{27}$$

ICA is often used in the problem of blind source separation (BSS). BSS is the concept of separating independent sources from many observed mixes of the sources. This type of phenomena occurs in many physical situations. One such case is the cocktail party problem.

4.1.2 The Cocktail Party Problem

Imagine being in a room at a cocktail party with many guests. Everyone at the party is talking at the same time. A problem arises: how does one separate the individual conversations in the room? In this problem, we have q guests at the party and model each guest's conversation as a source. We place p microphones randomly around the room where each obtains an observation. This process is modeled using equation (25); note that $\mathbf{U}$ is unknown to us.

With $\mathbf{U}$ unknown it may seem impossible to estimate $\mathbf{x}$ from $\mathbf{b}$ as we must also find $\mathbf{U}$. However, we can model the sources as being mutually independent which is often a reasonable assumption. Thus, we can use the ICA algorithm to determine an estimate of $\mathbf{U}$ in order to find an estimate of $\mathbf{x}$, $\hat{\mathbf{x}}$, as modeled in equation (27).

4.1.3 The FastICA Algorithm

The development of the FastICA algorithm [8] begins with the definition of differential entropy

$$H(\mathbf{y}) = -\int f(\mathbf{y}) \log f(\mathbf{y}) d\mathbf{y} \tag{28}$$

From this definition, the concept of negentropy is realized

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y}) \tag{29}$$

where $\mathbf{y}_{gauss}$ is a Gaussian random vector with the same covariance as $\mathbf{y}$. Negentropy has a useful property in that it is invariant for linear transforms. Negentropy is also interpreted as a measure of nongaussianity.

Using differential entropy, the notion of mutual information between several random variables arises. FastICA uses the idea of mutual information to express the measure of dependence between random variables. Expressing mutual information, I, using negentropy and constraining the variables to be uncorrelated, we get

$$I(y_1, y_2, ..., y_n) = J(\mathbf{y}) - \sum_i J(y_i) \tag{30}$$

From this it is possible to construct the FastICA algorithm. The algorithm attempts to determine an unmixing matrix such that the mutual information of the unmixed components is minimized. It is also possible to show that this model is similar to finding the directions in which the negentropy is maximized.

We must measure the negentropy of a random variable in order to find the directions where the negentropy is maximized. Often, an approximation is used of the form

$$J(y_i) \approx c[E\{G(y_i)\} - E\{G(\mathbf{v})\}]^2 \tag{31}$$

where $G(\cdot)$ is practically any non-quadratic function, c is an ignorable constant, and v is a random variable distributed normally with unit variance and zero mean. We assume y_i is also zero mean with unit variance.

Maximization of the function J_G is performed to find one independent component.

$$J_{\mathbf{G}}(\mathbf{u}) = [E\{G(\mathbf{u^T x})\} - E\{G(v)\}]^2 \tag{32}$$

A common choice for the function G [8] is

$$G_1(r) = \log(\cosh(r)) \tag{33}$$

$$g_1(r) = G_1(r)' = \tanh(r) \tag{34}$$

The approach in (32) is extended to determine all the independent components. If one recalls the definition of mutual information, we see that the mutual information among random variables is minimized when the sum of the negentropies is maximized. Therefore, if we maximize over the sum of all the negentropy estimates, and using the decorrelation constraint, we can obtain the following optimization problem

$$\max_{\mathbf{u}_i, i=1,\ldots,n} \sum_{i=1}^{n} J_G(\mathbf{u}_i) \text{ subject to } E\{(\mathbf{u}_k^T\mathbf{x})(\mathbf{u}_j^T\mathbf{x})\} = \delta_{jk} \tag{35}$$

We could use the familiar gradient descent algorithm to maximize the resulting optimization problem of (32). This results in an algorithm that can operate on time domain data in real-time. This method has the advantage of adaptability in the presence of non-stationary environments. However, use of gradient descent often leads to slow convergence and its speed generally depends on the learning rate parameter. Thus, a bad learning rate parameter can destroy convergence and even lead to large oscillation around the optimal solution.

The authors of FastICA introduce a fixed-point iteration algorithm to defeat the perils of gradient descent. These algorithms are able to adapt faster and more reliable than their gradient descent counterparts; however, they must operate in batch mode preventing fast processing in real-time environments with ease.

We construct an optimization problem to derive the fixed-point algorithm. The optima of the system, described in (35), are points at which

$$E\{\mathbf{u}g(\mathbf{u^T b})\} - \beta\mathbf{u} = 0 \tag{36}$$

$$\beta = E\{\mathbf{u}_0^T\mathbf{b} \cdot g(\mathbf{u}_0^T\mathbf{b})\} \tag{37}$$

This result is derived by observing the Kuhn-Tucker conditions as denoted in [9]. FastICA chooses to solve (36) using Newton's method. This method is of the form (for a single variable)

$$b_{n+1} = b_n - \frac{f(b_n)}{f'(b_n)} \tag{38}$$

For the case of many variables [10], we get

$$\mathbf{b}_{b+1} = \mathbf{b}_n - f(\mathbf{b}_n)J_f^{-1}(\mathbf{b}_n) \tag{39}$$

Thus, instead of dividing by $f'(\mathbf{b}_n)$, as done in (38), we multiply by the inverse of the Jacobian of f which is given by

$$J_f(\mathbf{u}) = E\{\mathbf{b}\mathbf{b}^{\mathbf{T}}g'(\mathbf{u}^{\mathbf{T}}\mathbf{b})\} - \beta\mathbf{I} \tag{40}$$

and not to be confused with J of (31) and (32). To simplify the inversion of the Jacobian, the first term is approximated using the fact that the data is whitened as a preconditioning step

$$E\{\mathbf{b}\mathbf{b}^{\mathbf{T}}g'(\mathbf{u}^{\mathbf{T}}\mathbf{b})\} \approx E\{g'(\mathbf{u}^{\mathbf{T}}\mathbf{b})\}\mathbf{I} \tag{41}$$

This result gives an approximate Newton iteration

$$\mathbf{u}^+ = \mathbf{u} - \frac{E\{\mathbf{b}g(\mathbf{u}^{\mathbf{T}}\mathbf{b})\} - \beta\mathbf{u}}{E\{g'(\mathbf{u}^{\mathbf{T}}\mathbf{b})\} - \beta} \tag{42}$$

$$\mathbf{u}^* = \frac{\mathbf{u}^+}{||\mathbf{u}^+||} \tag{43}$$

We can factor (42) and (43) by $\beta - E\{g'(\mathbf{u}^T\mathbf{b})\}$ to get the final fixed-point algorithm

$$\mathbf{u}^{+} = E\{\mathbf{b}g(\mathbf{u^T b})\} - E\{g'(\mathbf{u^T b})\}\mathbf{u} \tag{44}$$

$$\mathbf{u}^{*} = \frac{\mathbf{u}^{+}}{||\mathbf{u}^{+}||} \tag{45}$$

Optimizations using Newton's method may suffer from convergence problems. To mitigate this problem, one may multiply the update term by a learning rate parameter to improve convergence. This gives us a stabilized fixed-point algorithm

$$\mathbf{u}^{+} = \mathbf{u} - \mu(t)\left[\frac{E\{\mathbf{x}g(\mathbf{u^T b})\} - \beta\mathbf{u}}{E\{g'(\mathbf{u^T b})\} - \beta}\right] \tag{46}$$

$$\mathbf{u}^{*} = \frac{\mathbf{u}^{+}}{||\mathbf{u}^{+}||} \tag{47}$$

where $\mu(t)$ is the learning rate parameter which can be defined by a small constant (e.g. $\mu(t) = 0.1$ or 0.01) or a function of time (e.g. $\mu(t) = e^{-0.09t}$).

Equations (46) and (47) give us an optimization scheme to determine one independent component. To obtain all of the components, n neurons may be used, but their outputs must be decorrelated after every iteration to prevent more than one of them converging to the same maxima. The authors of FastICA present three methods for achieve this. We will only describe the method used in this implementation.

A method to achieve decorrelation after each iteration is based on the Gram-Schmidt orthogonalization . Using this method, we estimate an independent component using a single unit $\mathbf{u}_p$ and then initialize the next unit, $\mathbf{u}_{p+1}$, using a projection of the previous units

$$\mathbf{u}_{p+1} = \mathbf{u}_{p+1} - \sum_{j=1}^{p} \mathbf{u}_{p+1}^{T}\mathbf{u}_j\mathbf{u}_j \tag{48}$$

$$\mathbf{u}_{p+1} = \frac{\mathbf{u}_{p+1}}{||\mathbf{u}_{p+1}||} \tag{49}$$

where the vectors $\mathbf{u}$ make up the columns of the estimated mixing matrix $\hat{\mathbf{U}}$.

Finally, the mixing matrix $\hat{\mathbf{U}}$ is inverted to determine the unmixing matrix and obtain an estimate of the sources as in (26).

4.2 Independent Component Analysis Endmember Extraction Algorithm (ICA-EEA)

ICA-EEA finds pure endmembers by decomposing the data using the ICA algorithm into independent components. The components are scored and ordered; only the n highest components are kept. Pure endmembers are derived from the maximum absolute value of each component. Abundance maps are derived from each component by normalizing the data to the interval $[0, 1]$. ICA-EEA begins by performing the ICA algorithm and generating p independent components where p is the number of bands in the image. Hypothetically, any ICA algorithm could be used to generate the p independent components. However, the fast ICA algorithm is the ICA algorithm chosen by the authors of [4]. We use the FastICA algorithm in our experiments with the contrast function given by equation (34). Random vectors are used to initialize FastICA causing the independent components, a matrix of p components each $m \times n$ ($\mathbf{C}_{m\times n\times p}$), to be generated to be in random order. ICA-EEA scores the independent components using

$$S(C_{:,:,i}) = \frac{1}{12}\left[\kappa_i^3\right]^2 + \frac{1}{48}\left[\kappa_i^4 - 3\right]^2 \tag{50}$$

$$\kappa_i^3 = E\left[\zeta_i^3\right] = \frac{1}{N}\sum_{i=1}^{N}(z_n^i)^3 \tag{51}$$

κ_i^4 is defined similarly. The independent components are then arranged in decreasingly order by score. Only the q top scoring independent components are kept; the rest are discarded. Endmembers are selected by finding the pixel in the original HSI datacube corresponding to the pixel in the independent component with the largest absolute value. Abundance maps, noted by $\alpha_{m\times n}$, are then created by normalizing each independent component to an interval $[0, 1]$ using

$$\alpha_{i,j} = \frac{|Ci, j| - \eta}{\nu - \eta} \tag{52}$$

where

$$\eta = \min_i |C_{i,:}| \tag{53}$$

$$\nu = \max_i |C_{i,:}| \tag{54}$$

Table 42 displays the results of the algorithm on real data.

5 Vertex Component Analysis

The vertex component analysis (VCA) algorithm [11] assumes the data forms a high dimensional simplex due to the ANC and ASC constraints. It then utilizes a property of simplexes, the affine transformation of a simplex yields another simplex, to find the vertices of the simplex. It exploits this property to find maximal projections of the data, which it records as an endmember, then projects the data onto a subspace orthogonal to this endmember and repeats. Each repetition yields one endmember. Thus, the process repeats until n endmembers are found.

5.1 The VCA Algorithm

Like the ATGP algorithm, the VCA algorithm begins with inputs $\mathbf{M}$ and q where $\mathbf{M}$ is a $p \times N$ data matrix and q is the desired number of endmember to detect.

VCA begins by defining a matrix of column vectors, $\mathbf{U}$. The vectors composing the matrix are the current set of endmembers found in the image. The set begins with an initial vector containing all zeros with a one as the last element.

$$\mathbf{U}_{initial} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \tag{55}$$

The data is then projected onto a vector orthogonal to $\mathbf{U}$. Such a vector is constructed by using a matrix pseudo-inverse, specifically, the Moore-Penrose inverse [12]. The Moore-Penrose matrix inverse gives a solution such that the pseudo-inverse of a matrix $\mathbf{U}^{+}$ has the property of producing an orthogonal projection onto $R(\mathbf{U})$ by $\mathbf{U}\mathbf{U}^{+}$. From this, we can obtain an vector orthogonal to $\mathbf{U}$ using

$$\mathbf{v} = (\mathbf{I} - \mathbf{U}\mathbf{U}^{+})\mathbf{w} \tag{56}$$

where $\mathbf{w}$ is a vector of elements selected from a Gaussian distribution of zero mean. This vector $\mathbf{w}$ is used to ensure the projection is not null since the probability of $\mathbf{w}$ causing (56) to result in a null projection is essentially zero. It is important to note that this equation is not the same as equation (23). The additional vector multiplication of $\mathbf{w}$ is an important difference to ATGP and further explained below.

VCA computes an orthonormal vector $\mathbf{v}$ from $\mathbf{U}$ using equation (56). The pixel with the largest magnitude in this projection is selected as new endmember pixel $\mathbf{e}_i$, augmented to $\mathbf{U}$, and the process is repeated until all p endmembers are found. The first endmember $\mathbf{e}_0$ found replaces the initial vector $\mathbf{U}_{initial}$ instead of augmenting the matrix on the first iteration.

VCA has a few preconditioning steps to reduce computational complexity and mitigate effects of noise. First, VCA centers the data and performs a singular value decomposition (SVD). The columns corresponding to the largest eigenvalues of the decomposition, the dominant columns, are saved and the data reprojected onto this space. In effect, the operation performs the principal component transform (PCT) as PCT and SVD are equivalent when the data is centered. VCA then rescales the data to mitigate illumination inconsistencies. However, this rescaling may amplify noise in the data and therefore corrupt the extrema of the projected simplex. VCA mitigates this by reducing the dimension of the data to $p - 1$ instead of p when computing the SVD when the measured SNR for the image is below a threshold. The SNR estimate of the image is calculated by

$$SNR \simeq 10 \log_{10} \frac{P_{R_q} - \frac{q}{p} P_R}{P_R - P_{R_q}} \tag{57}$$

where

$$P_R = E[\mathbf{b^T b}] \tag{58}$$

$$P_{R_q} = E[\mathbf{b^T Q}_q \mathbf{Q}_q^T \mathbf{r}] \tag{59}$$

$$\tag{60}$$

in the case of using SVD for the data preconditioning. Note that $\mathbf{Q}_q$ is a matrix composed of q the dominant columns of the SVD. The threshold was empirically determined by the authors of VCA to be $SNR_{th} = 15 + \log_{10}(q)$ though it thesis finds the threshold of $SNR_{th} = 28 + \log_{10}(q)$ to be more effective.

5.2 Similarities to the Pixel Purity Index Algorithm

VCA is similar to the pixel purity index (PPI) algorithm [13] as both algorithms work by finding endmembers as extrema of vector projections. Whereas VCA selects projections orthogonal to the current endmember subspace, PPI projects data onto random vectors (which the authors call *skewers*) and selects endmembers as those which often result in the extrema of the projection. Additionally, PPI performs dimensionality reduction as a method to reduce computational complexity and mitigate noise effects. PPI uses the maximum noise fraction (MNF) to reduce dimensionality which is equivalent to the PCT when the noise is white. The algorithm shares traits similar to PPI, however, the algorithm has been shown to perform better both algorithmically and computationally.

5.3 Similarities to the ATGP Algorithm

The VCA and ATGP algorithms are very similar. However, it is important to note their differences: the method of orthogonal projection to determine the next endmember. Notice, when finding vectors orthogonal to a span, there are often *several* vectors which are solutions. This brings us to the fundamental difference between the ATGP and VCA algorithms. The ATGP algorithm selects the highest power (maximum length) vector of all possible orthogonal solutions as the next endmember whereas the VCA algorithm selects the highest power endmember of a particular orthogonal direction where this direction is chosen at random via $\mathbf{w}$. This property of both algorithms accounts for some endmembers being missed though it is shown later that VCA suffers greater than ATGP.

The ATGP algorithm also differs from the VCA algorithm in that it requires no data conditioning (SVD / PCT) and does not perform illumination normalization.

5.4 Computational Complexity

VCA has a computation complexity, in flops, of $2p^2N$. PPI has a computation complexity, in flops, of $2psN$ where s is the number of skewers used.

6 Fully Constrained Least Squares

Recall the linear mixing equation $\mathbf{Ux} = \mathbf{b}$ where $\mathbf{U}$ is a $p \times q$ matrix of column vectors representing q endmember signatures each with p dimensions, $\mathbf{x}$ is a $q \times 1$ vector representing the abundance fractions of each endmember, and $\mathbf{b}$ is a $p \times 1$ vector representing the target endmember signature for decomposition. This section discusses methods to solve for $\mathbf{x}$ using unconstrained least squares, least squares with sum-to-one constraint (ASC), and least squares with sum-to-one (ASC) and nonnegativity constraints (ANC).

A full derivation of the ASC constraint is provided to show a correct solution to this problem. The derivation of this constraint is rarely shown in HSI unmixing literature. Additionally, the solution to this problem shown in [14] is incorrect. This has been verified with the authors via [15].

6.1 Least Squares With No Constraints

In the case where there are no constraints the *Moore-Penrose inverse*, or often simply stated as the pseudo-inverse, can be utilized to solve for $\mathbf{x}$ given that $p > n$.

$$\mathbf{x} = (\mathbf{U}^T\mathbf{U})^{\mathbf{-1}}\mathbf{U}^{\mathbf{T}}\mathbf{b} \tag{61}$$

For convenience, we define the quantity

$$\mathbf{U}^{+} = (\mathbf{U}^T\mathbf{U})^{\mathbf{-1}}\mathbf{U}^{\mathbf{T}} \tag{62}$$

This gives the final solution for the a least squares solution with no constraints:

$$\mathbf{x}_{LS} = \mathbf{U}^{+}\mathbf{b} \tag{63}$$

6.2 Least Squares With Sum-to-One Constraint (ASC)

In this case, the following constraint is posed upon the least squares solution

$$\mathbf{1}^T\mathbf{x} = 1 \tag{64}$$

where $\mathbf{1}$ is defined as a $q \times 1$ column vector of 1's ($\mathbf{1} = [1_1 1_2 1_3 ... 1_q]^T$). This is the ASC constraint. This least squares problem can be solved using the Lagrange multiplier method. We start by defining the Lagrangian function and its partial derivatives.

$$\begin{aligned} J(\mathbf{x}, \lambda) &= \frac{1}{2}(\mathbf{b} - \mathbf{U}\mathbf{x})^T(\mathbf{b} - \mathbf{U}\mathbf{x}) + \lambda(\mathbf{1}^T\mathbf{x} - 1) && (65) \\ \frac{dJ}{d\lambda} &= \mathbf{1}^T\mathbf{x} - 1 && (66) \\ \frac{dJ}{d\mathbf{x}} &= (-\mathbf{U}^T)(\mathbf{b} - \mathbf{U}\mathbf{x}) + \lambda\mathbf{1} && (67) \\ &= -\mathbf{U}^T\mathbf{b} + \mathbf{U}^T\mathbf{U}\mathbf{x} + \lambda\mathbf{1} && (68) \end{aligned}$$

Solving for $\mathbf{x}$ in $\frac{dJ}{dx}$ we get our constraint

$$\begin{aligned} \mathbf{U}^T\mathbf{U}\mathbf{x} &= \mathbf{U}^T\mathbf{b} - \lambda\mathbf{1} && (69) \\ \mathbf{x} &= (\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{b} - (\mathbf{U}^T\mathbf{U})^{-1}\lambda\mathbf{1} && (70) \\ \mathbf{x} &= \mathbf{U}^+\mathbf{b} - (\mathbf{U}^T\mathbf{U})^{-1}\lambda\mathbf{1} && (71) \\ \mathbf{x} &= x_{LS} - (\mathbf{U}^T\mathbf{U})^{-1}\lambda\mathbf{1} && (72) \end{aligned}$$

Setting $\frac{dJ}{d\lambda} = 0$ we get:

$$\mathbf{1}^T\mathbf{x} = 1 \tag{73}$$

Solving for λ in equation (72) by multiplying both sides by $\mathbf{1}^T$ we get:

$$\begin{aligned} \mathbf{1}^T\mathbf{x} &= \mathbf{1}^T\mathbf{U}^+\mathbf{b} - \mathbf{1}^T(\mathbf{U}^T\mathbf{U})^{-1}\lambda\mathbf{1} && (74) \\ 1 &= \mathbf{1}^T\mathbf{U}^+\mathbf{b} - \mathbf{1}^T(\mathbf{U}^T\mathbf{U})^{-1}\lambda\mathbf{1} && (75) \\ 1 - \mathbf{1}^T\mathbf{U}^+\mathbf{b} &= -\mathbf{1}^T(\mathbf{U}^T\mathbf{U})^{-1}\lambda\mathbf{1} && (76) \\ \lambda &= \frac{\mathbf{1}^T\mathbf{U}^+\mathbf{b} - 1}{\mathbf{1}^T(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{1}} && (77) \end{aligned}$$

Using the solution for λ from equation (77) and inserting it into the solution for $\mathbf{x}$ from equation (72), we arrive at the sum-to-one constrained least squares solution:

$$\mathbf{x}_{ASC\_LS} = \mathbf{x}_{LS} - \frac{(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{1}^T(\mathbf{1}^T\mathbf{x}_{LS} - 1)}{\mathbf{1}^T(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{1}} \qquad (78)$$

6.3 Least Squares with Sum-to-One & Nonnegativity Constraints (ASC+ANC)

There are several methods to solve the least squares problem while adhering to both the ASC and ANC constraints since the solution is generally non-unique. The method chosen for analysis of the AVIRIS data can be found in [14]. The derivation directly uses the solution to equation (78) shown here.

7 Experiment

The algorithms were tested against several different datasets to understand how they behaved under varying conditions; mainly, SNR, number of endmembers present in the scene, endmember variability, and illumination inconstancy.

7.1 Definition of Signal-to-Noise Ratio

The Signal-to-Noise Ratio (SNR) provides a measure of how much noise is present in each spectral band of hyperspectral data. The definition of SNR used in this thesis is defined in [16] as:

$$SNR = \frac{\lambda_{50\%,i}}{\sigma_i} \tag{79}$$

where i is a specific band (wavelength), $\lambda_{50\%,i}$ is a mean reflectance value of 50% occurring for the band i, and σ_i is the standard deviation of reflectance values for band i. SNR generally has a large dynamic range. Therefore, SNR is often expressed in terms of the decibel (dB) using power ratios

$$SNR_{dB} = 20\log_{10}\left(\frac{\lambda_{50\%,i}}{\sigma_i}\right) \tag{80}$$

Figure 4 depicts an example of how noise at various SNRs affect an endmember of exactly fifty percent reflectance at all wavelengths.

SNR is often not clearly defined in much of the literature and an example is provided to clarify its definition here. Suppose one wishes to compute the SNR for the first band using the 2005 AVIRIS SNR model as listed in Appendix A. The linear SNR to be simulated is 171:1. Setting $SNR = 171$ and solving for σ in equation (79) one gets $\sigma = 0.002924$. Now, using equation (80) and setting $SNR_{dB} = 44.6$ and solving for σ, one gets $\sigma = 0.002924$.

Unmixing algorithms require an error metric to determine how well a detected

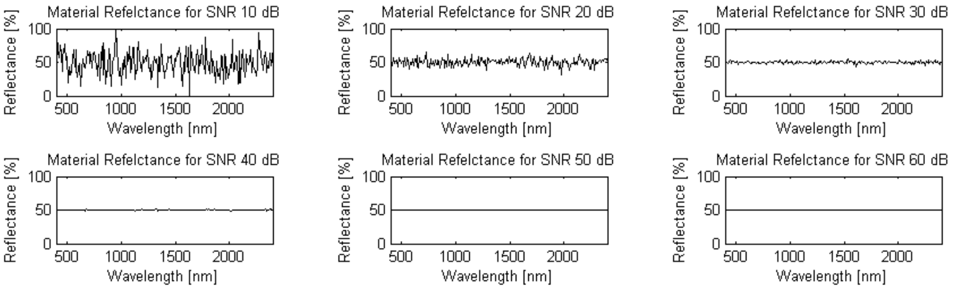


Figure 4: 50% reflectance endmembers with various levels of noise added.

endmember matches against truth. The error metric must be scale invariant when dealing with reflectance or radiance data as unique signatures will differ by a constant value from scene to scene due to varying illumination effects. A commonly used error metric, which is scale invariant, is the spectral angler mapper (SAM) metric. Sometimes this metric is also referred to as the spectral angle error (SAE) metric. SAM measures the angle between two spectra and is defined as:

$$SAM(\mathbf{a}, \mathbf{b}) = \cos^{-1}\left(\frac{\mathbf{a}^T\mathbf{b}}{\parallel \mathbf{a} \parallel \parallel \mathbf{b} \parallel}\right) \tag{81}$$

where $\parallel \cdot \parallel$ represents the Euclidian norm.

Figure 5 depicts the results of one hundred Monte Carlo runs of angle error versus SNR for a fifty percent reflectance signature corrupted by various levels of AWGN.

All of the experiments in this thesis conducted using simulated data by adding white Gaussian noise to each band unless otherwise stated.

7.2 AVIRIS Noise Model

The additive white gaussian noise model does not always accurately reflect the true noise model of a real sensor. To deal with this problem, each dataset is tested with a noise model based on the AVIRIS sensor [17] as of the year 2005. Figure 6 depicts the SNR of the AVIRIS sensor as a function of the wavelength. Only bands with an SNR > 50dB were utilized when using AVIRIS data or the AVIRIS SNR model.

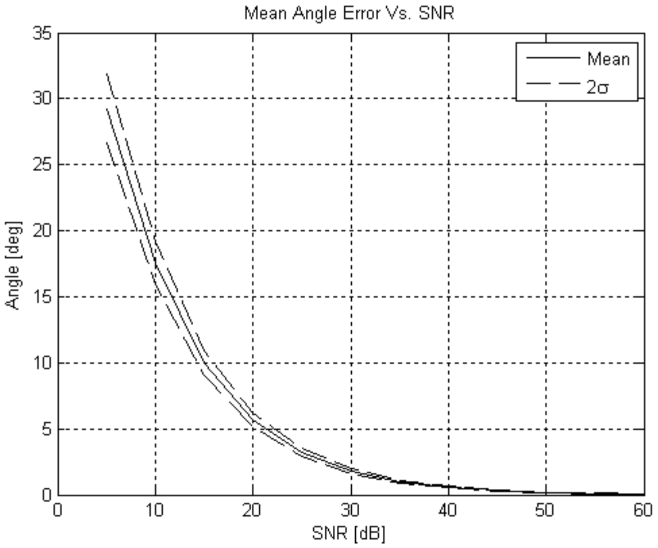


Figure 5: Angle error verses SNR for a fifty percent reflectance signature.

7.3 Test Data Generation

All of the data is resampled to 224 equally spaced bands inclusive of 400nm to 2500nm. Additionally, the endmember database was also resampled to these wavelengths.

Ground truth data of each pixel's mix is required in order to test each algorithm's ability to correctly unmix a hyperspectral image. Obtaining the mix information for a hyperspectral image is not an easy task. At a minimum, it requires extensive knowledge of the materials making up the terrain and their exact locations. This information is difficult to obtain even through field work. Thus, a method to simulate hyperspectral data is required.

Hyperspectral data cubes were simulated using two methods in order to obtain ground truth data. Both methods operate by creating abundance maps of each material and then creating each pixel by the weighted sum of each material's abundance. In the first method, called *synthetic mixing*, the data was simulated by creating high resolution abundance maps for each material and downsampling them to obtain the mixture. The downsampling operation preserved the ASC and ANC constraints. In the second method, the data was simulated by creating random mixtures from

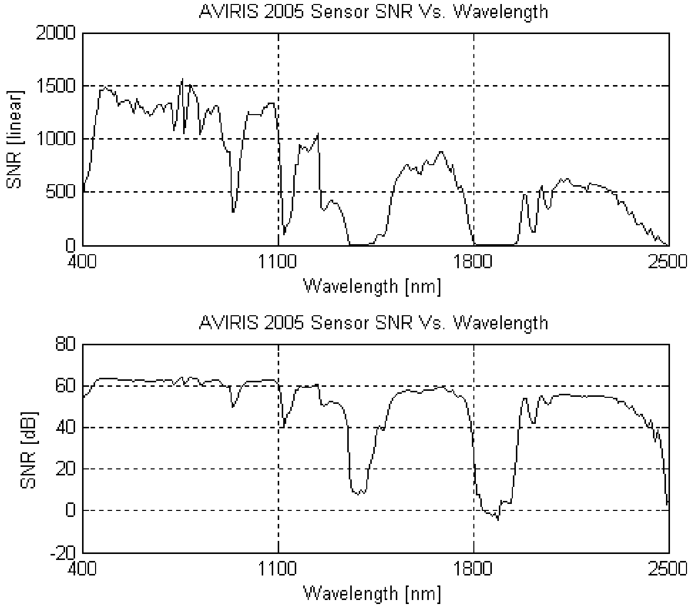


Figure 6: AVIRIS SNR at each sensor band sampled from 400-2500nm. Measured 2005 [17].

a Dirchlet distribution. The Dirchlet distribution has a unique property in that a sequences of q numbers can be generated from it that have a predetermined sum. Setting up the distribution in such a way as to have a sequence of q number which sum to one, the Dirchlet distribution simulates a uniformly mixed environment which is well suited for analysis.

The mixture distributions for both methods are designed such that the sum of abundances for each pixel equals one. In the case where illumination variance is modeled, we design the distribution such that the sum is a number chosen from a probability density function.

Using the first method, high resolution abundance maps were obtained using the following steps:

1. Starting with a color image, convert the image to the HSV colorspace.

2. Extract the hue plane.

3. Normalize the hue plane to obtain the full dynamic range of values. (i.e. [0, 1])

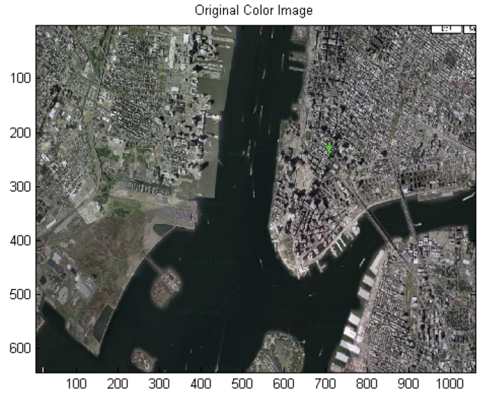


Figure 7: Original color (RGB) image.

4. Cluster the normalized hue plane into q clusters.

5. At this point, each pixel represents only one material. Create abundance maps from the clustered image.

6. Downsample abundance maps to desired resolution in order to obtain mixture.

This method simulates mixing mechanics often seen in reality:

- *Spatial locality.* Similar materials tend to be grouped locally.

- *Geometry.* The organization of materials observes a sensible geometry. For example, material groups are not uniformly distributed in reality as they are often bounded by manmade objects such as roads and buildings.

Figures 7, 8,9,10,11, and 12 depict an example of simulating hyperspectral data using method one.

The database endmembers used in the SMDE, RMDE, and RMDEI datasets were taken from [18].

7.4 Estimating the Number of Materials Present In a Scene

All of the unmixing algorithms require *a priori* knowledge of the number of materials present in the scene to operate. Additionally, it is assumed that the ratio of the

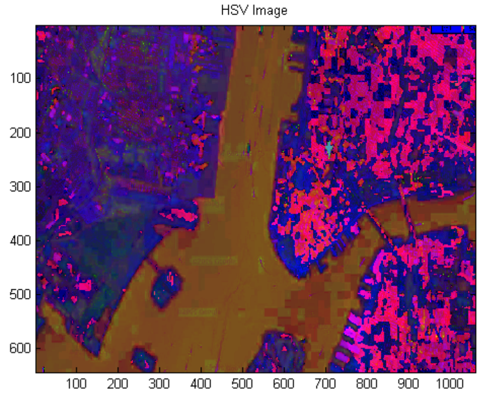


Figure 8: Original color image converted to the HSV colorspace.

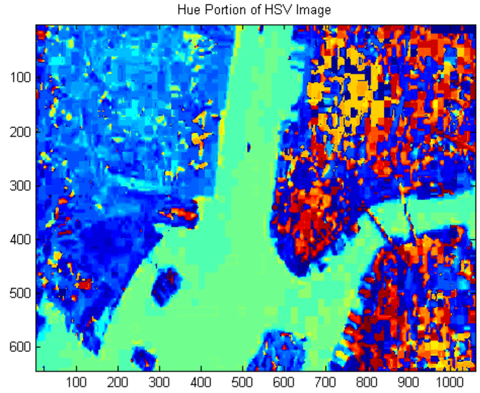


Figure 9: Hue portion of HSV image.

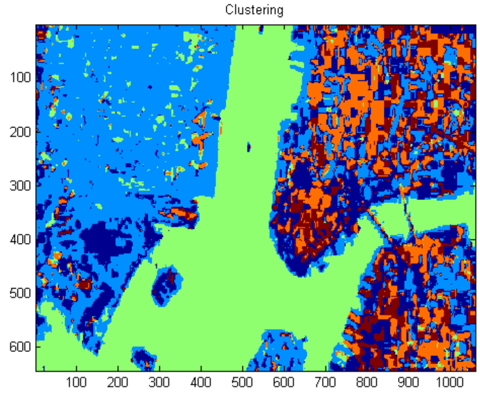
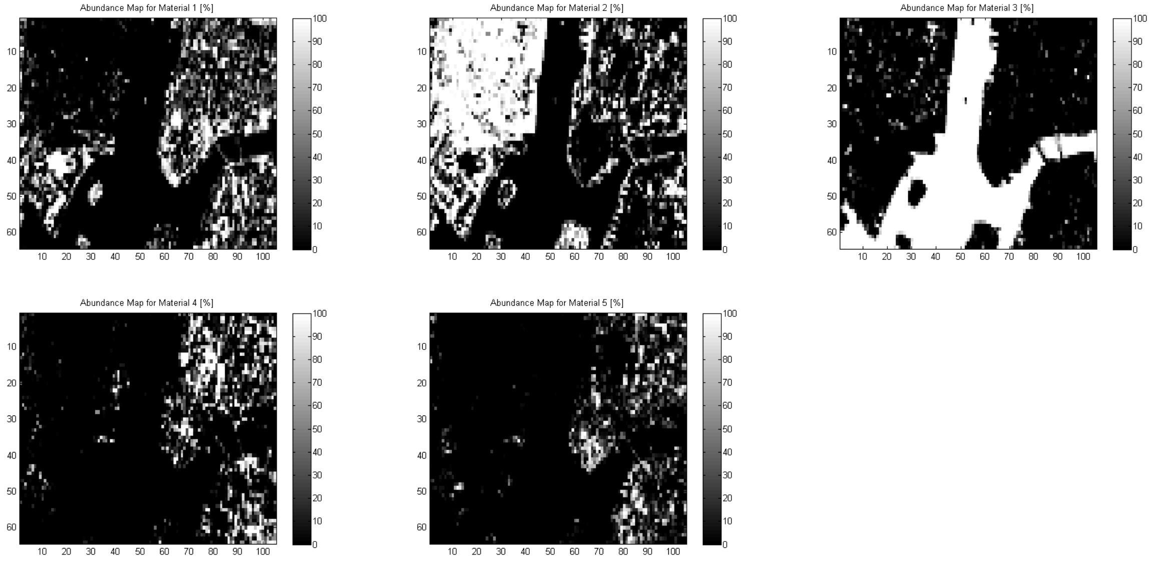


Figure 10: Clustering of hue portion of HSV image.

Figure 11: Simulated Abundance Maps



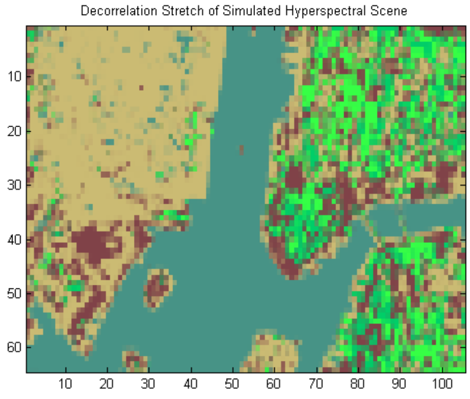


Figure 12: Mixed and downsampled image hyperspectral image.

number of materials present in a scene versus the number of spectral bands comprising the sensor is small.

A common technique used to estimate the number of materials present in a scene is called virtual dimensionality (VD). The accuracy of this method is tested in the experiment.

7.5 Monte-Carlo Analysis of Algorithms

The VCA algorithms were run ten times for each trial and the results were averaged. This averaging was not necessary for the ATGP algorithm as it yields deterministic results for each run. The ICA-EEA algorithm, although yielding non-deterministic results, was run only once for each configuration due to the long runtime it exhibited. This long runtime is due to the high computational complexity of each iteration and the estimation of each component often requires more than ten iterations.

7.6 Endmember Matching

The recovered endmembers were matched to the endmember set used to create the scene to determine the SAM error, number of unique detections, and the number of pure detection for each run. Matching was done using the SAM metric. Pure detections were defined as those pixels with $> 95\%$ ground truth purity.

7.7 Datasets Tested

Five datasets were used in evaluation of the four algorithms (VD, ATGP, ICA-EEA, and VCA). Four of the five datasets are simulated with the remaining dataset being an AIVIRS collected scene from Moffett Field, California in the United States.

7.7.1 Synthetic Mixture Database Endmembers (SMDE) Dataset

The SMDE dataset was constructed using the first dataset generation method (syntectic mixing) described above with an initial image of downtown New York City, New York in the United States. This image is depicted in figure 7. The endmembers are selected from the database of [18]. The pixel mixtures do not exceed one and pure pixels were purposefully injected to ensure their existence. The affinity matrices for the dataset are available for reference in Appendix B. The histograms of the affinity matrices for the data are available for reference in Appendix C.

7.7.2 Random Mixture Database Endmembers (RMDE) Dataset

The random mixture database endmembers (RMDE) dataset is composed of a random selection of endmembers from the database of [18]. Each pixel is a random mixture of these endmembers. The mixture is determined using a Dirchlet distribution and thereby forcing the sum of the mixture to always equal one. Pure pixels were injected to ensure their existence. The affinity matrices for the dataset are available for reference in Appendix D. The histograms of the affinity matrices for the data are available for reference in Appendix E.

7.7.3 Random Mixture Random Endmembers (RMRE) Dataset

The random mixture random endmembers (RMRE) dataset is composed of a random selection of endmembers where the endmembers are generated randomly using a uniform distribution $U[0, 1]$. The distribution was chosen as to accurately model reflectance values which are typically constrained to be between zero and one. Each pixel is a random mixture of these endmembers. The mixture is determined using a Dirchlet distribution and thereby forcing the sum of the mixture to always equal one. Pure pixels were injected to ensure their existence. The affinity matrices for the dataset are available for reference in Appendix F. The histograms of the affinity matrices for the data are available for reference in Appendix G.

It is important to note why randomly generated endmembers are used in the examination of these algorithms. Randomly generated endmembers span a much larger space than the database endmembers. Their randomness forms a set of highly discriminable spectral signatures. This effect is reflected in the affinity matrix histograms of Appendix G where the majority of the endmembers are roughly forty degrees apart. Thus, an HSI unmixing algorithm should be able to *at least* recover these endmembers because of their high discriminability. In this way, this dataset provides bounds for how well and HSI unmixing algorithm can perform. One would expect the results of an optimal unmixing algorithm operating on this dataset to be close to the curve depicted in figure 5.

7.7.4 Random Mixture Database Endmembers with Illumination Variation (RMDEI) Dataset

The random mixture database endmembers with illumination variation (RMDEI) dataset is composed of a random selection of endmembers from the database of [18]. Each pixel is a random mixture of these endmembers. The mixture is determined using a Dirchlet distribution however the parameters are varied at every pixel so the sum is distributed as $U(0.9, 1.0)$. The variation of the mixture sum along with the variation of this sum over the field of the image is designed to model interference whether from topographic modulation of multipath effects. Pure pixels were injected to ensure their existence. The affinity matrices for the dataset are available for reference in Appendix H. The histograms of the affinity matrices for the data are available for reference in Appendix I.

Table 1: Parameters of Moffett Field, CA AVIRIS Collected Dataset

Date of collection	June 20, 1997
Time of collection	approximately 18:50 PST
Length of collection	approximately three minutes
Location (latitude, longitude)	+037.449470, -121.806630
Spatial structure	urban and vegetative areas
Atmospheric conditions	pilot noted clear conditions

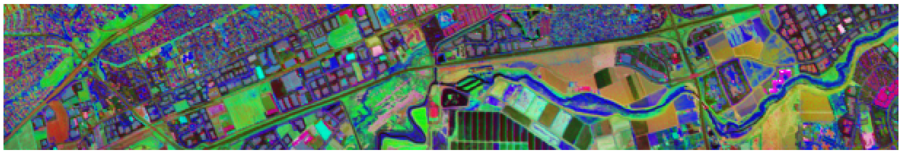


Figure 13: Decorrelation Stretched Image of Moffett Field, CA Dataset

7.7.5 Moffett Field, CA AVIRIS Dataset

The Moffett Field, CA AVIRIS dataset [2] was taken from the NASA JPL Airborne Visible / Infrared Imaging Spectrometer (AVIRIS) sensor. Table 1 lists the collection metadata. The data was received post-processed and given in units of reflectance. A decorrelation stretched image computed from bands 199, 126 and 51 (RGB) is shown in figure 13.

8 Results

Four simulated datasets and one live dataset was used in evaluation of the algorithms. For the simulated datasets, the number of endmembers varied from five to sixty in increments of five materials and the SNR varied from twenty to sixty dB in increments of five dB. Additionally, the simulated datasets were evaluated using the 2005 AVIRIS SNR model.

Table 2 provides a list of the dataset abbreviations. Table 3 lists the experiments conducted.

Abbreviation	**Dataset Name**
SMDE	Synthetic Mixture, Database Endmembers
RMDE	Random Mixture, Database Endmembers
RMDEI	Random Mixture, Database Endmembers with Varying Illumination
RMRE	Random Mixture, Random Endmembers
SMDEA	Synthetic Mixture, Database Endmembers Using Aviris SNR Model
RMDEA	Random Mixture, Database Endmembers Using Aviris SNR Model
RMDEIA	Random Mixture, Database Endmembers with Varying Illumination Using Aviris SNR Model
RMREA	Random Mixture, Random Endmembers Using Aviris SNR Model

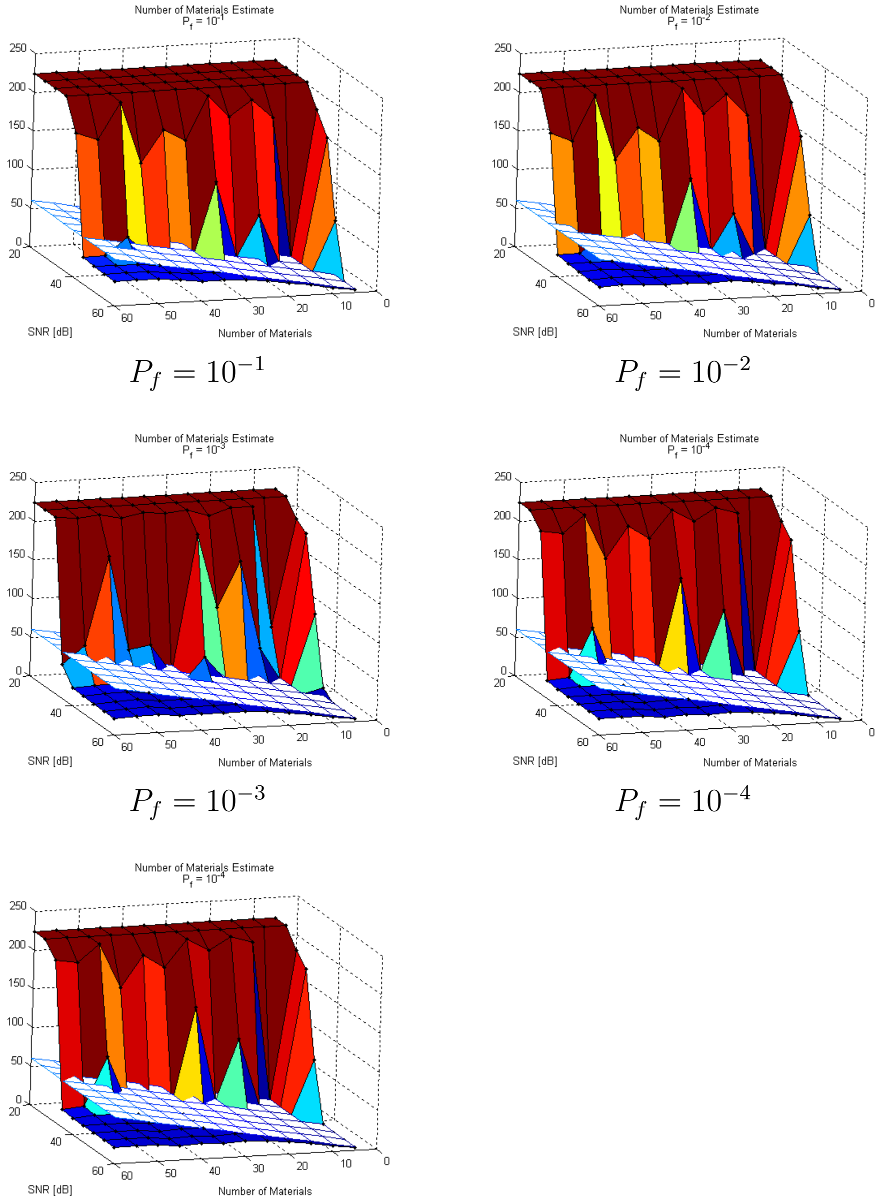
Table 2: List of dataset abbreviations.

Table 3: List of all experiments conducted and their associated results table.

Dataset	**Algorithm**	**Results**
SMDE	Virtual dimensionality	Table 4
	ATGP	Table 9
	ICA-EEA	Table 17
	VCA-Authors	Table 25
	VCA-Raw Pixels	Table 33
RMDE	Virtual dimensionality	Table 5
	ATGP	Table 10
	ICA-EEA	Table 18
	VCA-Authors	Table 26
	VCA-Raw Pixels	Table 34
RMDEI	Virtual dimensionality	Table 6
	ATGP	Table 12
	ICA-EEA	Table 20
	VCA-Authors	Table 28
	VCA-Raw Pixels	Table 36
RMRE	Virtual dimensionality	Table 7
	ATGP	Table 11
	ICA-EEA	Table 19
	VCA-Authors	Table 27
	VCA-Raw Pixels	Table 35
SMDEA	ATGP	Table 13
	ICA-EEA	Table 21
	VCA-Authors	Table 29
	VCA-Raw Pixels	Table 37
RMDEA	ATGP	Table 14
	ICA-EEA	Table 22
	VCA-Authors	Table 30
	VCA-Raw Pixels	Table 38
RMDEIA	ATGP	Table 16
	ICA-EEA	Table 24
	VCA-Authors	Table 32
	VCA-Raw Pixels	Table 40
RMREA	ATGP	Table 15
	ICA-EEA	Table 23
	VCA-Authors	Table 31
	VCA-Raw Pixels	Table 38
Moffett Field, CA	Virtual dimensionality	Table 8
	ATGP derived abundance maps	Table 41
	ICA-EEA derived abundance maps	Table 42
	VCA-Raw Pixels derived abundance maps	Table 43

8.1 Virtual Dimensionality Algorithm

8.1.1 Synthetic Mixture, Database Endmembers Dataset



$P_f = 10^{-1}$

$P_f = 10^{-2}$

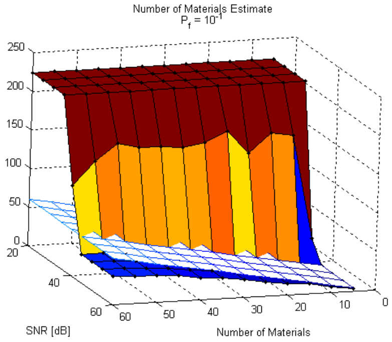
$P_f = 10^{-3}$

$P_f = 10^{-4}$

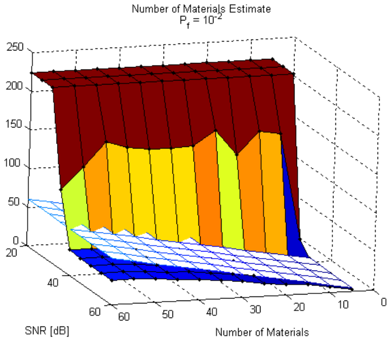
Table 4: Results of the VD algorithm for five false alarm rates(P_f). The white faceted plane represents the true number of materials in the data cube. The horizontal axis represents the number materials in the simulated data cube. The into-the-page axis represents the SNR of the simulated data cube. The vertical axis represent the number of materials estimated in the data cube.

8.1.2 Random Mixture, Database Endmembers Dataset

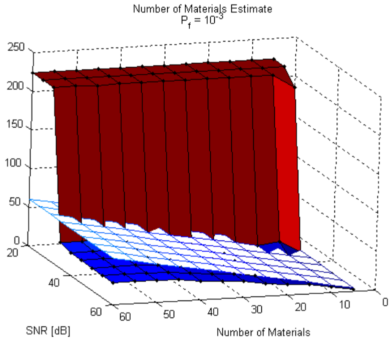
Table 5: Results of the VD algorithm for five false alarm rates(P_f). The white faceted plane represents the true number of materials in the data cube. The horizontal axis represents the number materials in the simulated data cube. The into-the-page axis represents the SNR of the simulated data cube. The vertical axis represent the number of materials estimated in the data cube.



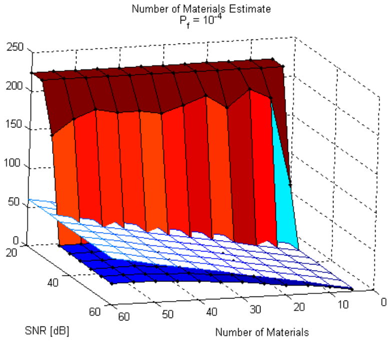
$P_f = 10^{-1}$



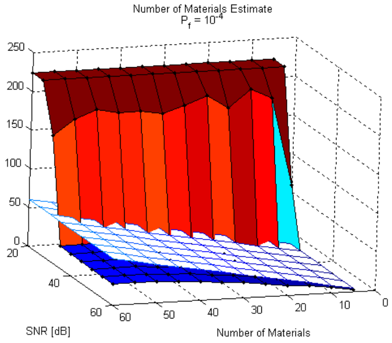
$P_f = 10^{-2}$



$P_f = 10^{-3}$



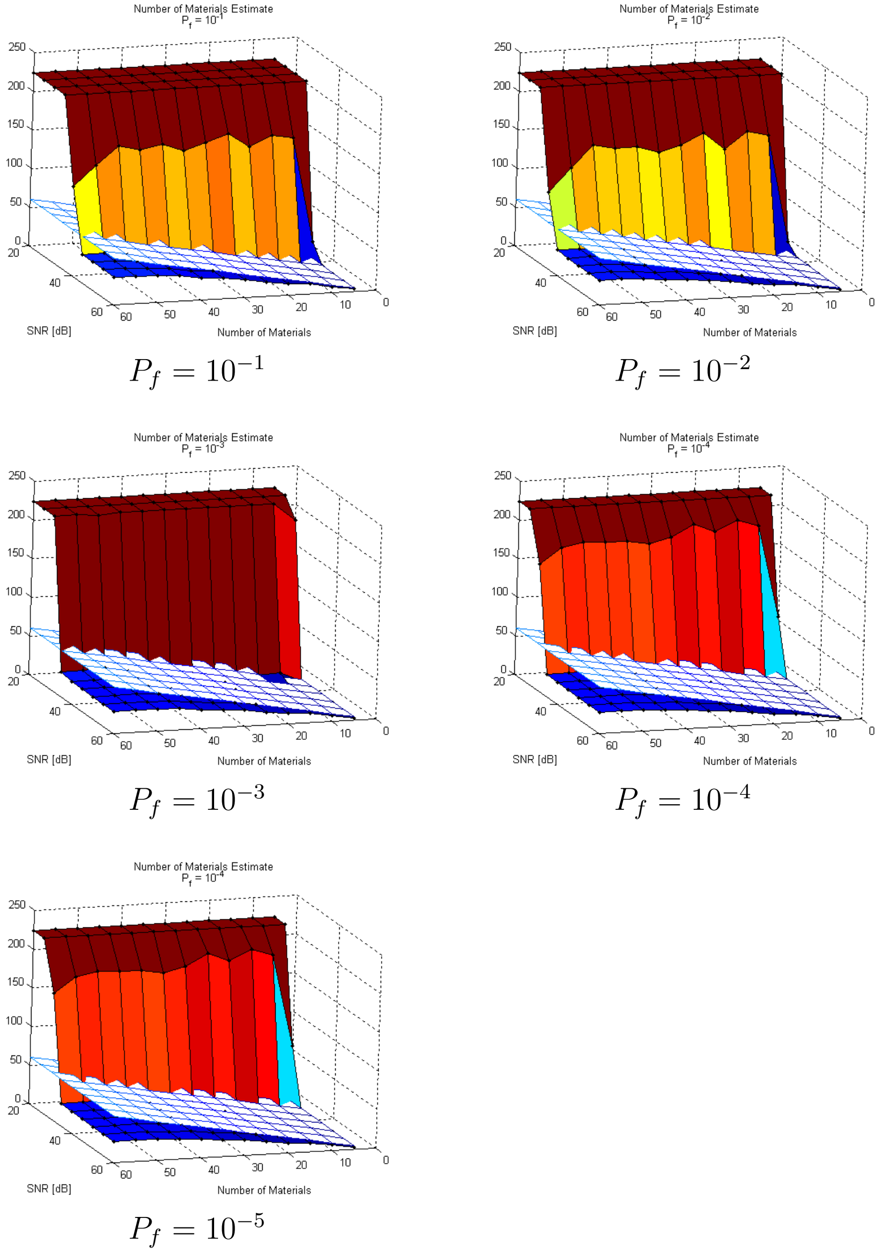
$P_f = 10^{-4}$



$P_f = 10^{-5}$

8.1.3 Random Mixture, Database Endmembers with Varying Illumination Dataset

Table 6: Results of the VD algorithm for five false alarm rates(P_f). The white faceted plane represents the true number of materials in the data cube. The horizontal axis represents the number materials in the simulated data cube. The into-the-page axis represents the SNR of the simulated data cube. The vertical axis represent the number of materials estimated in the data cube.



$P_f = 10^{-1}$

$P_f = 10^{-2}$

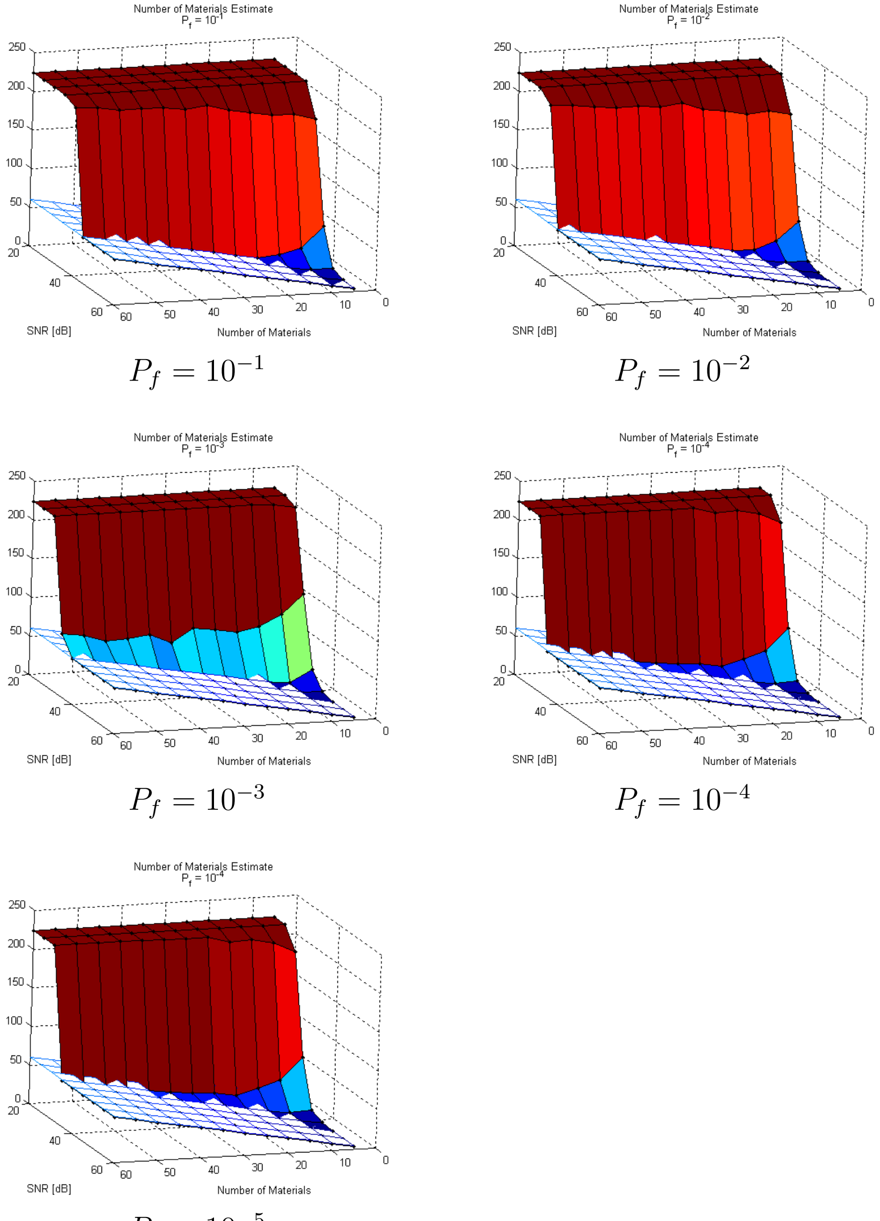
$P_f = 10^{-3}$

$P_f = 10^{-4}$

$P_f = 10^{-5}$

8.1.4 Random Mixture, Random Endmembers Dataset

Table 7: Results of the VD algorithm for five false alarm rates(P_f). The white faceted plane represents the true number of materials in the data cube. The horizontal axis represents the number materials in the simulated data cube. The into-the-page axis represents the SNR of the simulated data cube. The vertical axis represent the number of materials estimated in the data cube.



$P_f = 10^{-1}$

$P_f = 10^{-2}$

$P_f = 10^{-3}$

$P_f = 10^{-4}$

$P_f = 10^{-5}$

8.1.5 Moffett Field, CA Dataset

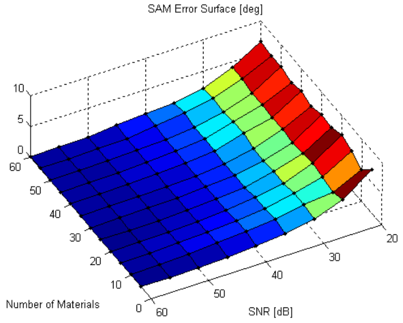
Table 8: Results of the VD algorithm run upon the Moffett Field, CA dataset for five false alarm rates(P_f). The number of materials estimated by the algorithm is indicated by $\hat{q}$.

P_f	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}
$\hat{q}$	120	64	50	41	35

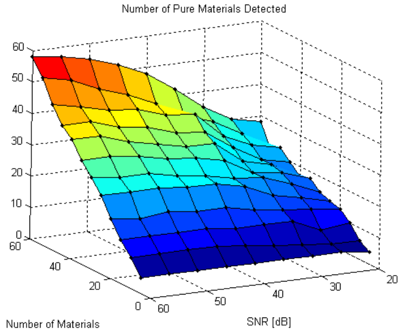
8.2 Automatic Target Generation Procedure

8.2.1 Synthetic Mixture, Database Endmembers Dataset

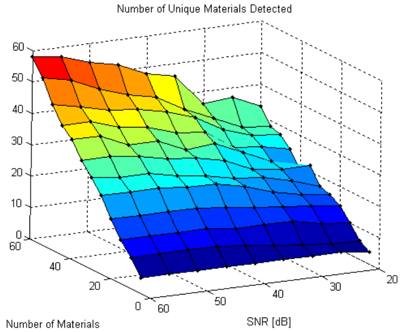
Table 9: Results of the ATGP algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]



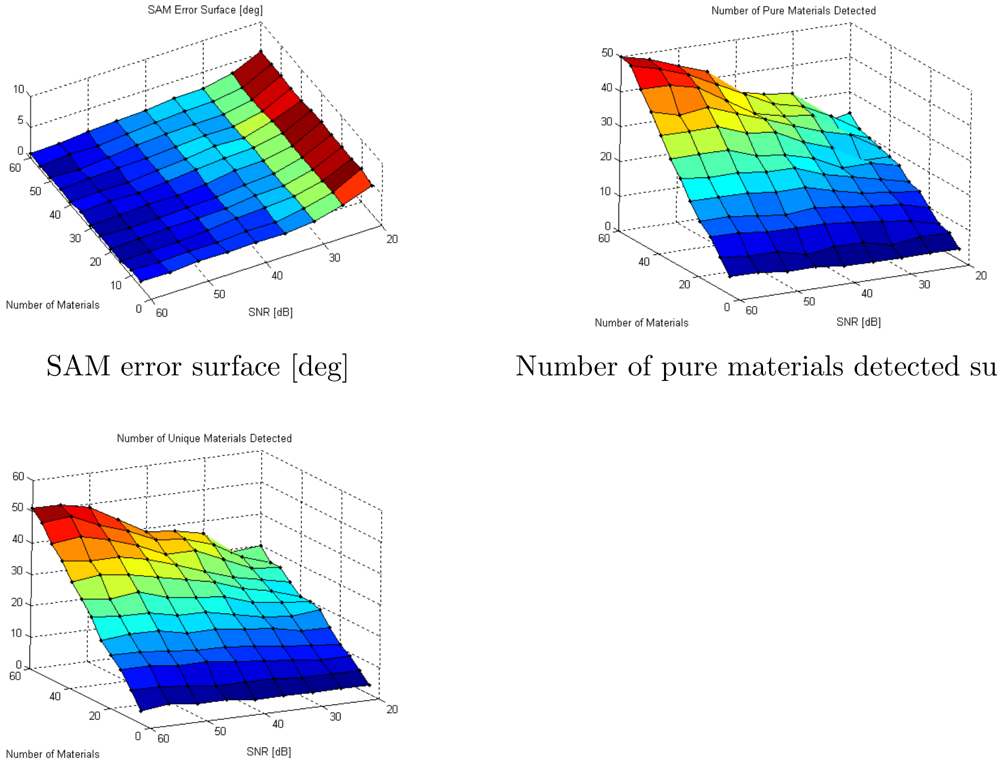
Number of pure materials detected surface



Number of unique materials detected surface

8.2.2 Random Mixture, Database Endmembers Dataset

Table 10: Results of the ATGP algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



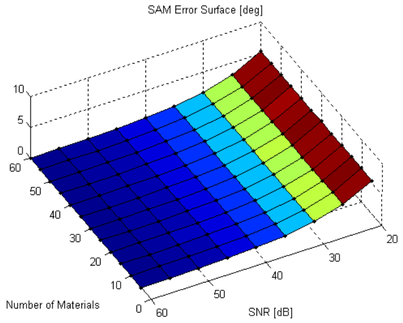
SAM error surface [deg]

Number of pure materials detected surface

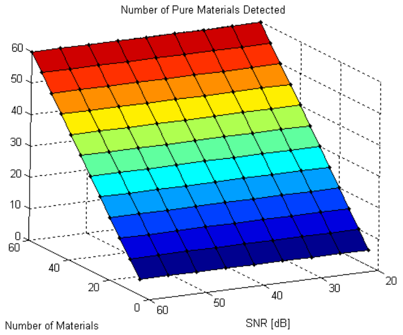
Number of unique materials detected surface

8.2.3 Random Mixture, Random Endmembers Dataset

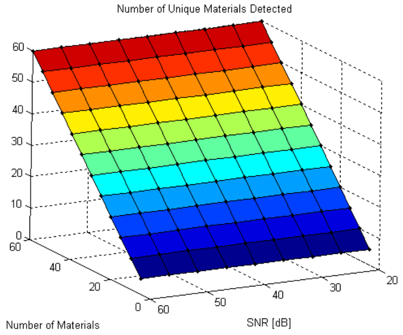
Table 11: Results of the ATGP algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]



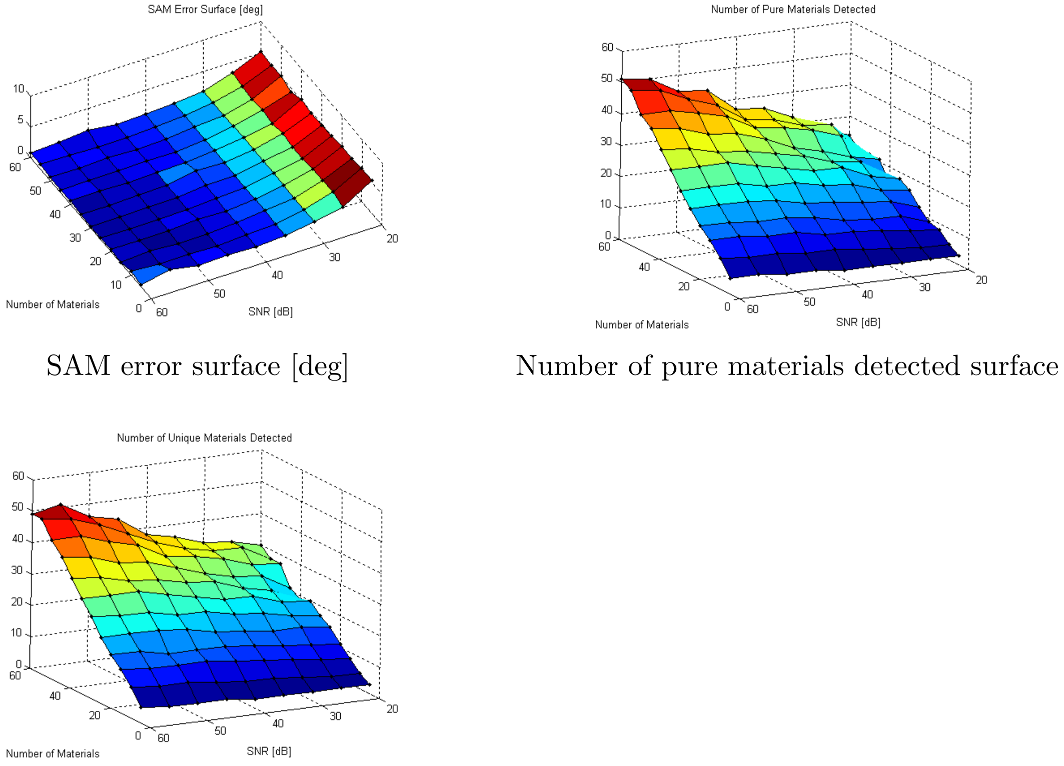
Number of pure materials detected surface



Number of unique materials detected surface

8.2.4 Random Mixture, Database Endmembers with Varying Illumination Dataset

Table 12: Results of the ATGP algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]

Number of pure materials detected surface

Number of unique materials detected surface

8.2.5 Synthetic Mixture, Database Endmembers Dataset Using AVIRIS SNR Model

Table 13: Results of the ATGP algorithm on the SMDE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	0.42	0.98	0.47	0.31	0.33	0.43	0.41	0.78	0.40	0.56	0.68	0.85
Number of Unique Detections	5	9	13	18	23	26	30	33	38	41	46	54
Number of Pure Detections	5	9	13	18	23	26	30	32	37	40	46	53

8.2.6 Random Mixture, Database Endmembers Dataset Using AVIRIS SNR Model

Table 14: Results of the ATGP algorithm on the RMDE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	1.1	1.2	1.1	0.88	0.68	0.84	0.72	0.97	1.1	0.97	1.3	0.81
Number of Unique Detections	4	8	12	15	20	24	29	33	36	39	43	48
Number of Pure Detections	4	7	11	14	19	23	27	31	32	36	37	46

8.2.7 Random Mixture, Random Endmembers Dataset Using AVIRIS SNR Model

Table 15: Results of the ATGP algorithm on the RMRE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	0.23	0.20	0.24	0.18	0.20	0.20	0.20	0.20	0.21	0.21	0.20	0.20
Number of Unique Detections	5	10	15	20	25	30	35	40	45	50	55	60
Number of Pure Detections	5	10	15	20	25	30	35	40	45	50	55	60

8.2.8 Random Mixture, Database Endmembers with Varying Illumination Dataset Using AVIRIS SNR Model

8.2.9 Moffett Field, CA Dataset

Abundance maps are displayed in table 41.

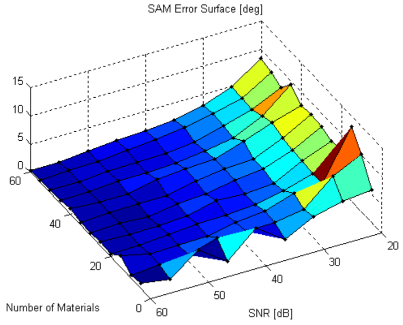
8.3 ICA-EEA Algorithm

8.3.1 Synthetic Mixture, Database Endmembers Dataset

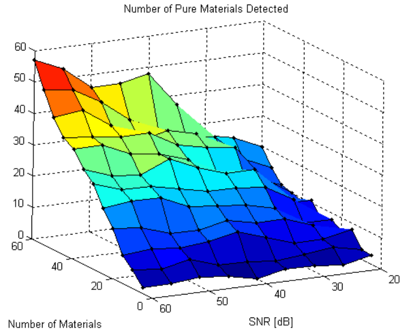
Table 16: Results of the ATGP algorithm on the RMDEI dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	0.96	0.58	1.2	1.1	0.58	1.2	1.2	1.0	1.5	1.3	1.5	1.8
Number of Unique Detections	4	8	13	15	21	24	28	33	36	40	43	46
Number of Pure Detections	4	7	12	15	20	23	26	32	35	36	41	44

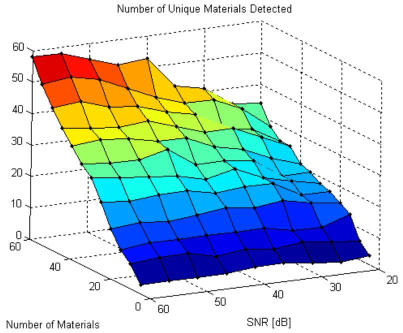
Table 17: Results of the ICA-EEA algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]



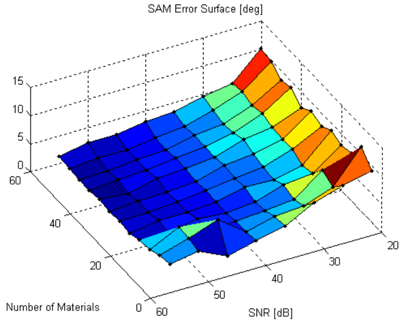
Number of pure materials detected surface



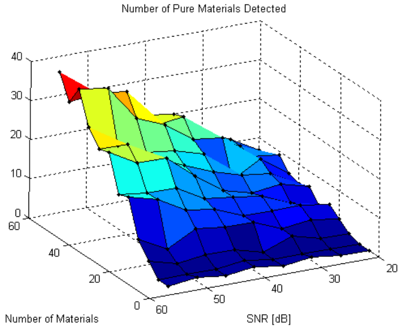
Number of unique materials detected surface

8.3.2 Random Mixture, Database Endmembers Dataset

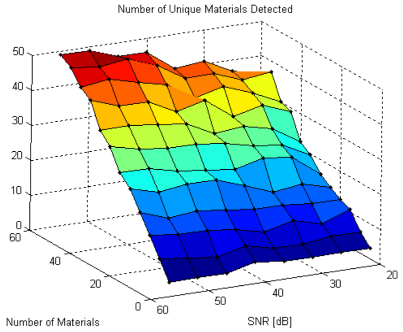
Table 18: Results of the ICA-EEA algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]



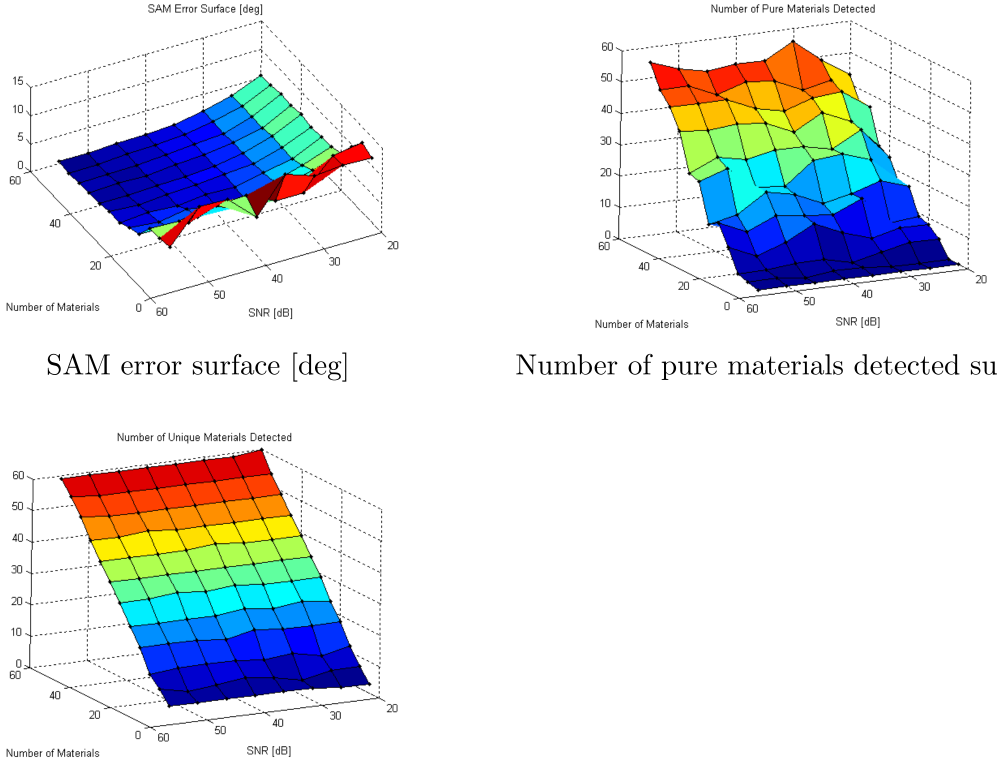
Number of pure materials detected surface



Number of unique materials detected surface

8.3.3 Random Mixture, Random Endmembers Dataset

Table 19: Results of the ICA-EEA algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



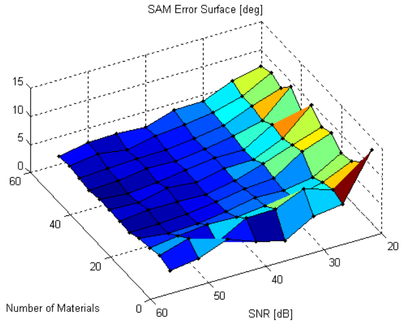
SAM error surface [deg]

Number of pure materials detected surface

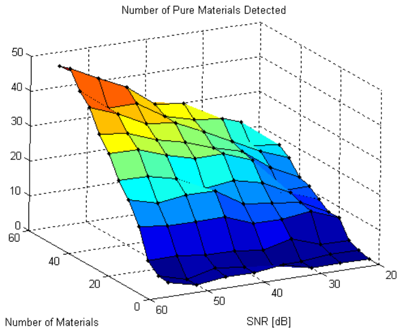
Number of unique materials detected surface

8.3.4 Random Mixture, Database Endmembers with Varying Illumination Dataset

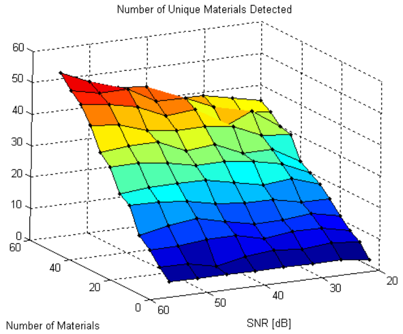
Table 20: Results of the ICA-EEA algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]



Number of pure materials detected surface



Number of unique materials detected surface

8.3.5 Synthetic Mixture, Database Endmembers Dataset Using AVIRIS SNR Model

This results for sixty materials present are not shown because FastICA did not converge.

Table 21: Results of the ICA-EEA algorithm on the SMDE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55
SAM Error [deg]	2.2	1.2	0.91	0.92	0.50	0.87	0.80	1.4	0.61	0.63	0.75
Number of Unique Detections	4	8	14	17	23	27	32	36	41	46	52
Number of Pure Detections	2	6	11	14	22	24	28	35	40	42	50

8.3.6 Random Mixture, Database Endmembers Dataset Using AVIRIS SNR Model

This results for sixty materials present are not shown because FastICA did not converge.

Table 22: Results of the ICA-EEA algorithm on the RMDE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55
SAM Error [deg]	3.6	2.2	1.3	1.2	1.1	1.5	1.1	1.4	1.4	1.2	1.4
Number of Unique Detections	3	8	12	16	21	24	29	35	40	43	47
Number of Pure Detections	0	5	11	15	21	23	27	35	37	42	47

8.3.7 Random Mixture, Random Endmembers Dataset Using AVIRIS SNR Model

This results for sixty materials present are not shown because FastICA did not converge.

Table 23: Results of the ICA-EEA algorithm on the RMRE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55
SAM Error [deg]	7.5	9.1	4.1	2.1	0.75	1.1	0.81	0.54	0.53	0.69	0.6
Number of Unique Detections	3	6	12	18	24	30	34	39	45	49	54
Number of Pure Detections	1	0	10	17	23	29	34	39	43	49	54

8.3.8 Random Mixture, Database Endmembers with Varying Illumination Dataset Using AVIRIS SNR Model

This results for sixty materials present are not shown because FastICA did not converge.

Table 24: Results of the ICA-EEA algorithm on the RMDEI dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55
SAM Error [deg]	3.5	2.4	1.8	1.4	1.0	1.4	1.4	1.1	1.2	1.6	1.5
Number of Unique Detections	3	9	13	16	21	26	29	34	40	42	48
Number of Pure Detections	1	6	12	17	21	25	29	34	39	43	49

8.3.9 Moffett Field, CA Dataset

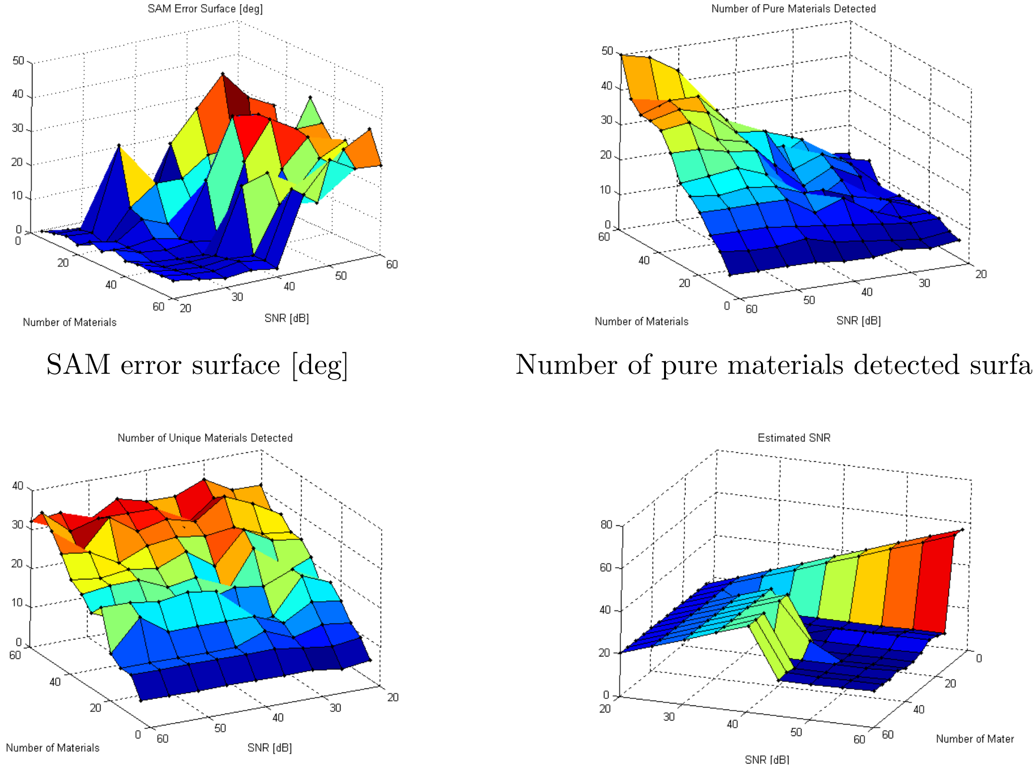
Abundance maps are displayed in table 42.

8.4 VCA-Authors Algorithm

In addition to the usual criteria for evaluation, we examine the SNR estimate produced by the algorithm. Note, the VCA-Authors algorithm is VCA as described in [11] which includes the PCT as a data pre-conditioning step to help remove noise. Additionally, the final endmembers are derived from the PCT transformed data as opposed to being chosen from the raw, unconditioned data.

8.4.1 Synthetic Mixture, Database Endmembers Dataset

Table 25: Results of the VCA-Authors algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]

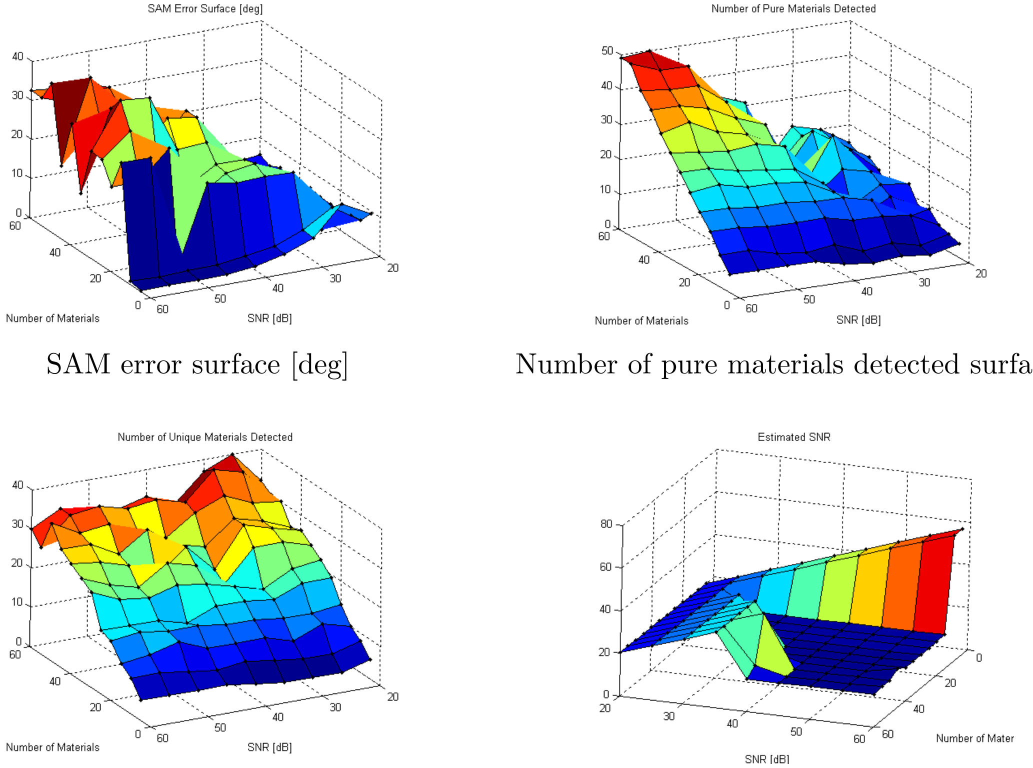
Number of pure materials detected surface

Number of unique materials detected surface

SNR estimate surface

8.4.2 Random Mixture, Database Endmembers Dataset

Table 26: Results of the VCA-Authors algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]

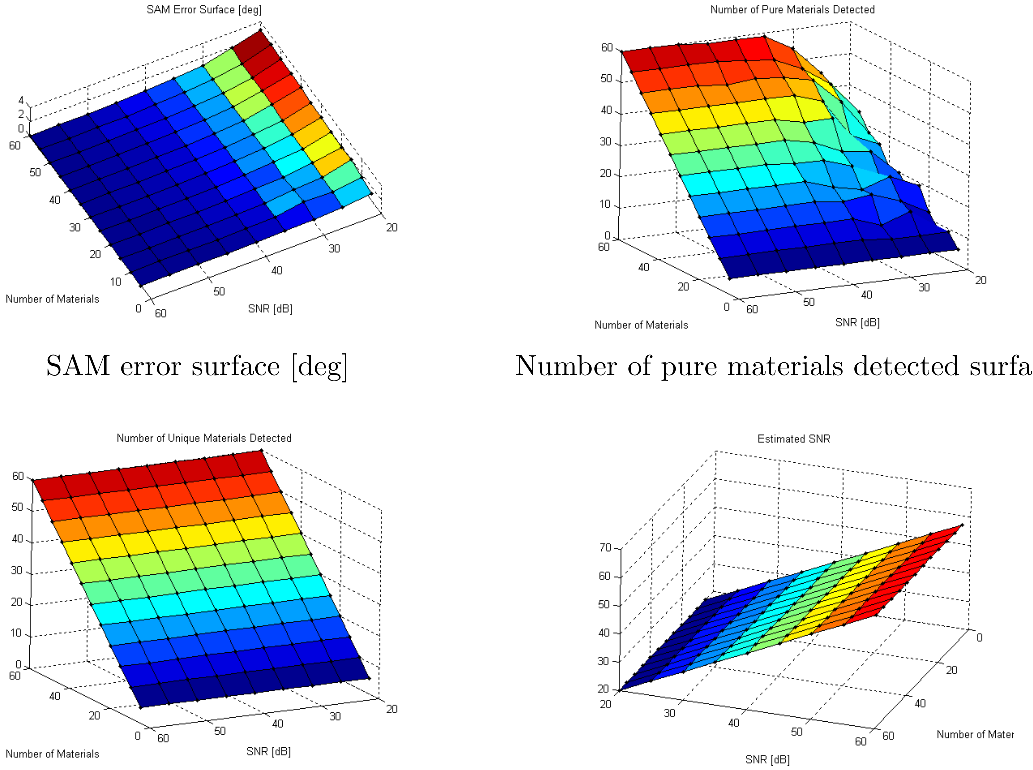
Number of pure materials detected surface

Number of unique materials detected surface

SNR estimate surface

8.4.3 Random Mixture, Random Endmembers Dataset

Table 27: Results of the VCA-Authors algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]

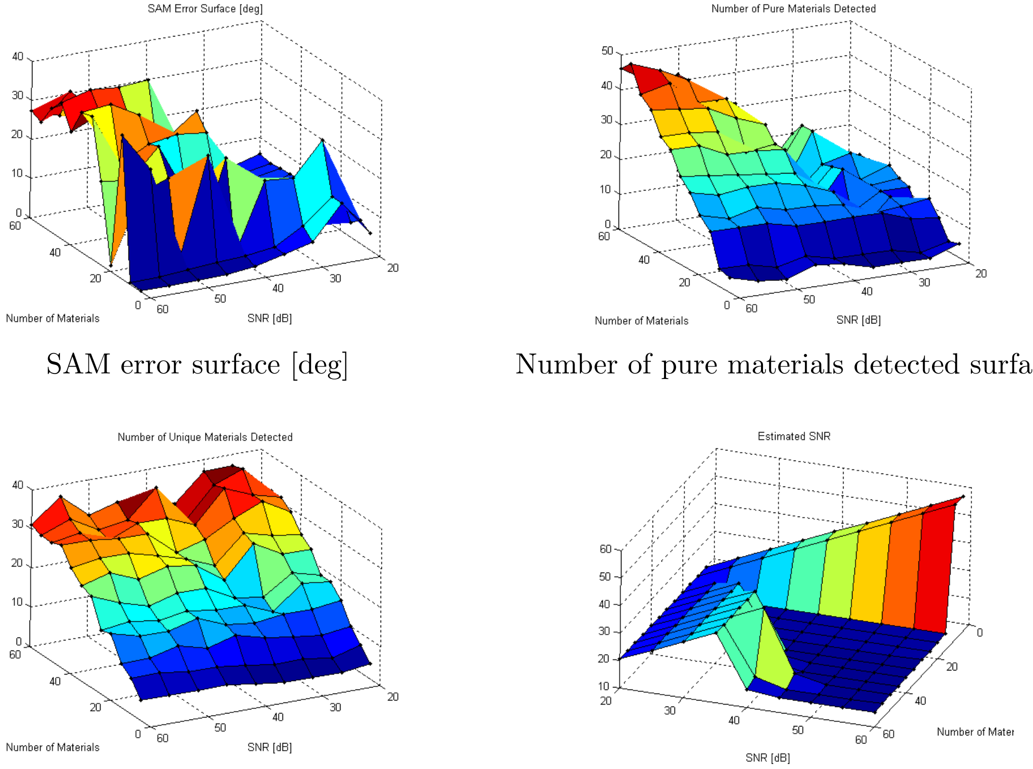
Number of pure materials detected surface

Number of unique materials detected surface

SNR estimate surface

8.4.4 Random Mixture, Database Endmembers with Varying Illumination Dataset

Table 28: Results of the VCA-Authors algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]

Number of pure materials detected surface

Number of unique materials detected surface

SNR estimate surface

8.4.5 Synthetic Mixture, Database Endmembers Dataset Using AVIRIS SNR Model

Table 29: Results of the VCA-Authors algorithm on the SMDE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	0.042	0.66	0.92	0.78	0.51	1.2	0.96	3.2	1.7	0.88	1.7	2.3
Number of Unique Detections	5	9	13	17	23	24	26	30	30	38	42	41
Number of Pure Detections	5	9	13	17	23	19	21	24	25	36	37	36
SNR Estimate [dB]	49	51	50	52	51	53	53	55	55	55	56	55

8.4.6 Random Mixture, Database Endmembers Dataset Using AVIRIS SNR Model

Table 30: Results of the VCA-Authors algorithm on the RMDE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	2.2	2.3	1.4	1	1.9	2.5	2.2	2.4	2.6	2.2	2	3.6
Number of Unique Detections	4	8	13	16	21	24	29	31	35	39	41	48
Number of Pure Detections	4	5	11	15	13	17	24	21	25	31	31	39
SNR Estimate [dB]	50	51	8.6	8.1	52	53	53	54	54	54	55	N/A

8.4.7 Random Mixture, Random Endmembers Dataset Using AVIRIS SNR Model

Table 31: Results of the VCA-Authors algorithm on the RMRE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	0.038	0.064	0.063	0.076	0.073	0.089	0.088	0.092	0.096	0.11	0.12	0.14
Number of Unique Detections	5	10	15	20	25	30	35	40	45	50	55	60
Number of Pure Detections	5	10	15	20	25	30	35	40	45	50	55	59
SNR Estimate [dB]	49	50	50	50	50	50	50	50	50	50	50	50

8.4.8 Random Mixture, Database Endmembers with Varying Illumination Dataset Using AVIRIS SNR Model

Table 32: Results of the VCA-Authors algorithm on the RMDEI dataset using the AVIRIS SNR model.

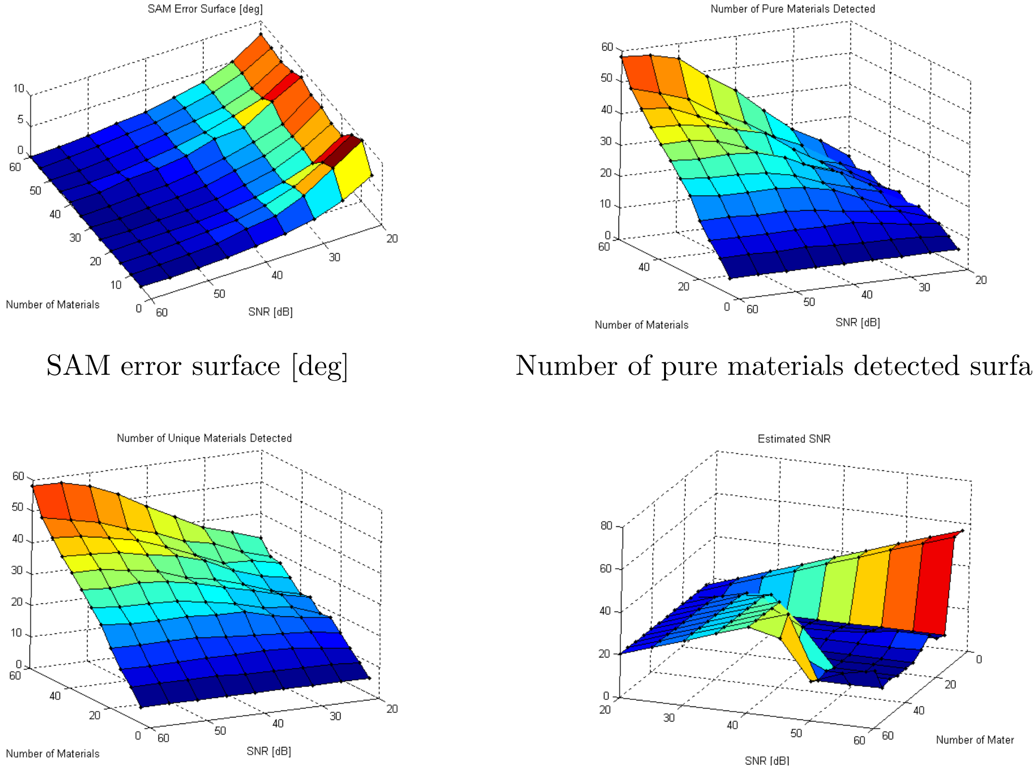
Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	2.6	2.7	1.4	1.1	2.2	2.4	2.2	2.3	2	2.4	2.6	7.8
Number of Unique Detections	4	8	13	17	22	23	29	33	35	38	44	41
Number of Pure Detections	2	4	12	15	13	15	14	22	25	26	26	37
SNR Estimate [dB]	50	51	8.6	8.3	52	53	53	54	54	54	55	N/A

8.5 VCA-Raw Pixels Algorithm

In addition to the usual criteria for evaluation, we examine the SNR estimate produced by the algorithm. Note, the VCA-Authors algorithm is VCA as described in [11] which includes the PCT as a data pre-conditioning step to help remove noise. Additionally, the final endmembers are derived from the raw, unconditioned data (original image) and not the PCT transformed data.

8.5.1 Synthetic Mixture, Database Endmembers Dataset

Table 33: Results of the VCA-Raw Pixels algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]

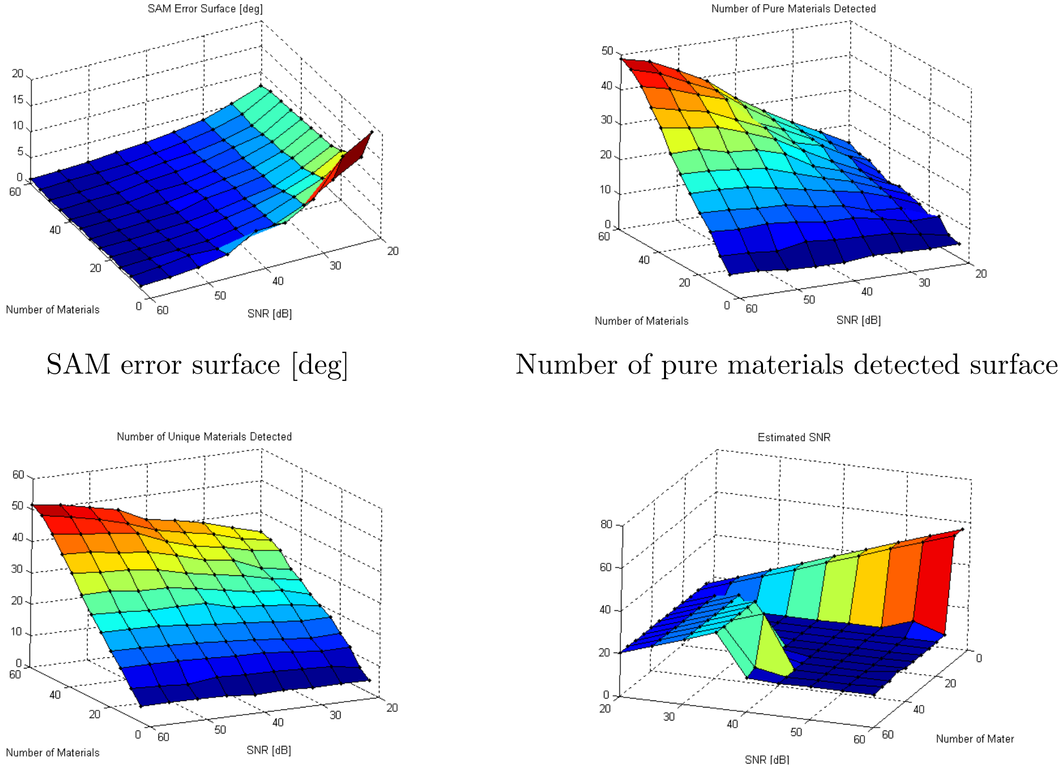
Number of pure materials detected surface

Number of unique materials detected surface

SNR estimate surface

8.5.2 Random Mixture, Database Endmembers Dataset

Table 34: Results of the VCA-Raw Pixels algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]

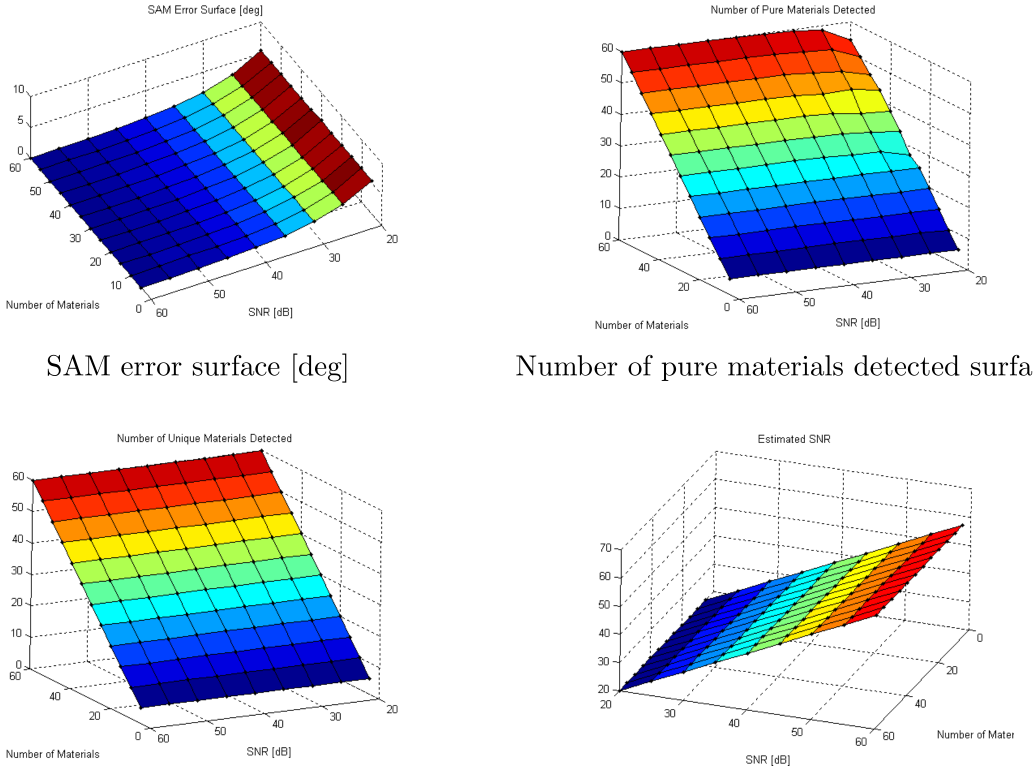
Number of pure materials detected surface

Number of unique materials detected surface

SNR estimate surface

8.5.3 Random Mixture, Random Endmembers Dataset

Table 35: Results of the VCA-Raw Pixels algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]

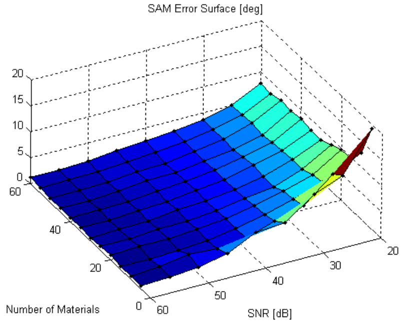
Number of pure materials detected surface

Number of unique materials detected surface

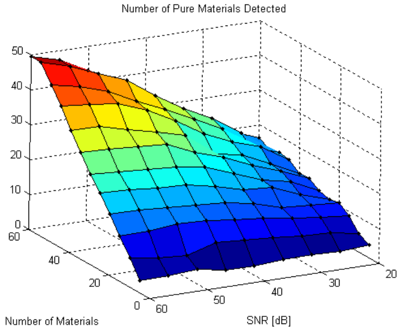
SNR estimate surface

8.5.4 Random Mixture, Database Endmembers with Varying Illumination Dataset

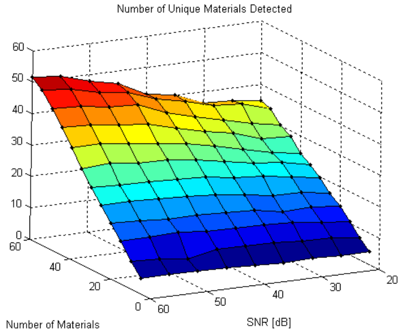
Table 36: Results of the VCA-Raw Pixels algorithm. The horizontal axis represents the SNR of the simulated data cube. The into-the-page axis represents the number materials in the simulated data cube.



SAM error surface [deg]



Number of pure materials detected surface



Number of unique materials detected surface

8.5.5 Synthetic Mixture, Database Endmembers Dataset Using AVIRIS SNR Model

Table 37: Results of the VCA-Raw Pixels algorithm on the SMDE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	0.36	0.84	1.2	0.70	0.64	0.52	0.84	1.4	0.83	0.94	1.1	1.2
Number of Unique Detections	5	9	13	18	23	25	29	33	36	39	44	51
Number of Pure Detections	5	9	12	16	21	22	28	30	34	34	40	44
SNR Estimate [dB]	49	51	50	52	51	53	53	55	55	55	56	55

8.5.6 Random Mixture, Database Endmembers Dataset Using AVIRIS SNR Model

Table 38: Results of the VCA-Raw Pixels algorithm on the RMDE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	0.5	2.7	2.2	1.2	1.3	2	1.6	1.3	2	1.8	2.0	1.7
Number of Unique Detections	4	9	12	17	21	24	29	33	36	44	43	48
Number of Pure Detections	4	7	10	16	15	20	25	25	29	31	30	37
SNR Estimate [dB]	50	51	51	52	52	53	53	54	54	54	55	N/A

8.5.7 Random Mixture, Random Endmembers Dataset Using AVIRIS SNR Model

Table 39: Results of the VCA-Raw Pixels algorithm on the RMRE dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	0.27	0.22	0.22	0.2	0.2	0.25	0.22	0.2	0.21	0.2	0.21	0.2
Number of Unique Detections	5	10	15	20	25	30	35	40	45	50	55	60
Number of Pure Detections	5	10	15	20	25	30	35	40	45	50	55	60
SNR Estimate [dB]	49	50	50	50	49	50	50	50	50	50	50	50

8.5.8 Random Mixture, Database Endmembers with Varying Illumination Dataset Using AVIRIS SNR Model

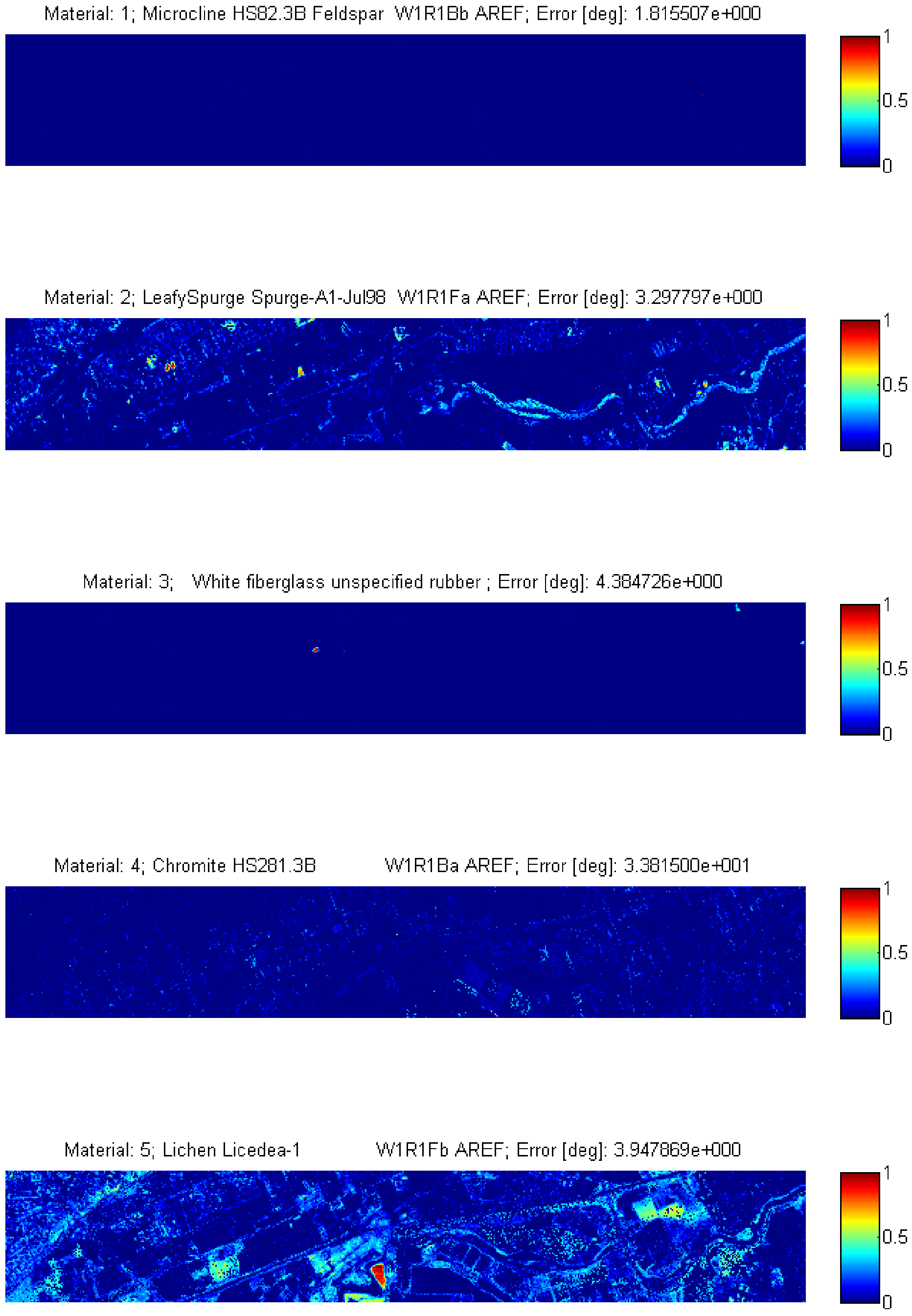
Table 40: Results of the VCA-Raw Pixels algorithm on the RMDEI dataset using the AVIRIS SNR model.

Number of Materials	5	10	15	20	25	30	35	40	45	50	55	60
SAM Error [deg]	3.2	3.3	0.74	1.4	1.3	1.2	1.6	2	1.7	1.6	1.7	2.1
Number of Unique Detections	4	7	12	16	21	25	29	32	36	39	43	45
Number of Pure Detections	4	6	11	16	18	19	23	25	29	36	38	33
SNR Estimate [dB]	50	51	16	16	52	53	53	54	54	54	55	N/A

8.5.9 Moffett Field, CA Dataset

Abundance maps are displayed in table 43.

Table 41: Abundance maps for Moffett Field, CA Dataset. Endmembers derived from ATGP algorithm.



Material: 1; Microcline HS82.3B Feldspar W1R1Bb AREF; Error [deg]: 1.815507e+000

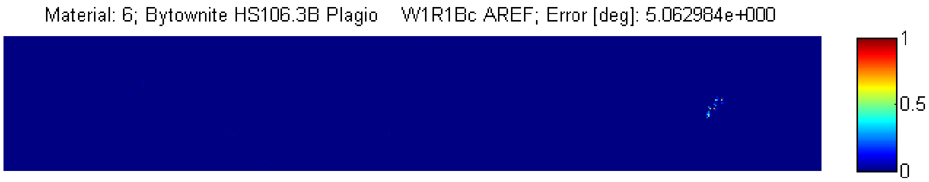
Material: 2; LeafySpurge Spurge-A1-Jul98 W1R1Fa AREF; Error [deg]: 3.297797e+000

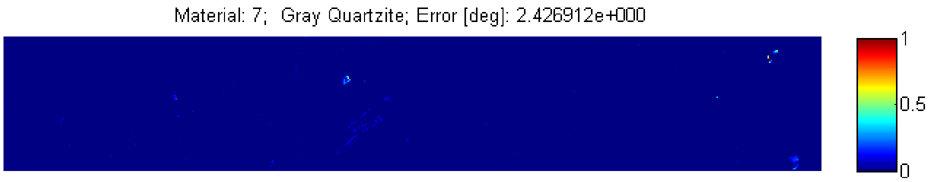
Material: 3; White fiberglass unspecified rubber ; Error [deg]: 4.384726e+000

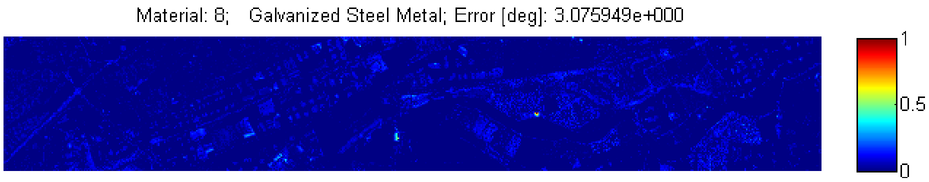
Material: 4; Chromite HS281.3B W1R1Ba AREF; Error [deg]: 3.381500e+001

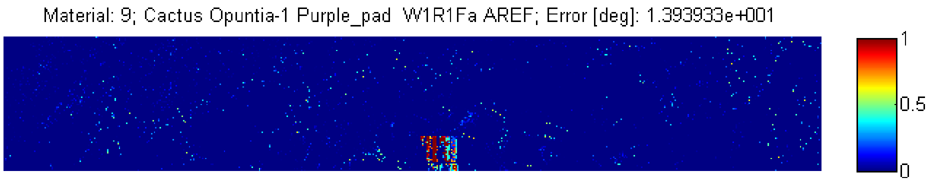
Material: 5; Lichen Licedea-1 W1R1Fb AREF; Error [deg]: 3.947869e+000

Table 41 – Continued


Material: 6; Bytownite HS106.3B Plagio W1R1Bc AREF; Error [deg]: 5.062984e+000


Material: 7; Gray Quartzite; Error [deg]: 2.426912e+000


Material: 8; Galvanized Steel Metal; Error [deg]: 3.075949e+000


Material: 9; Cactus Opuntia-1 Purple_pad W1R1Fa AREF; Error [deg]: 1.393933e+001

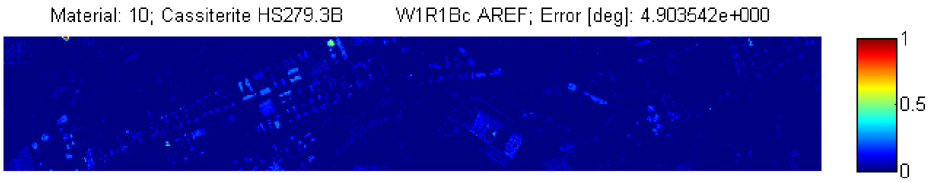
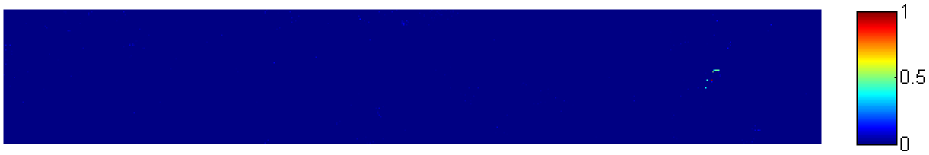
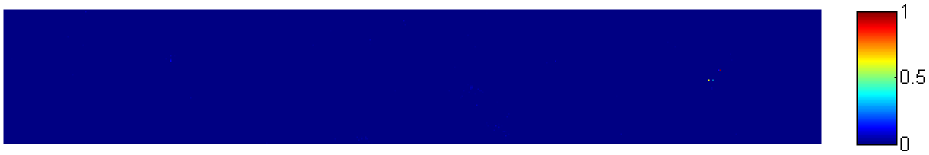

Material: 10; Cassiterite HS279.3B W1R1Bc AREF; Error [deg]: 4.903542e+000

Table 41 – Continued

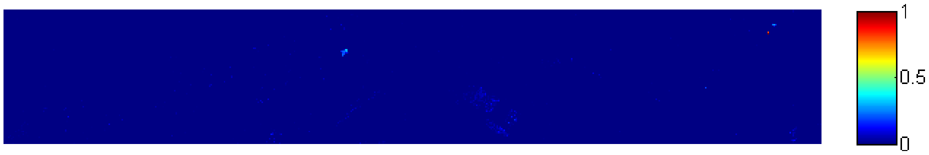
Material: 11; Kerogen BK-Cornell W1R1Ba AREF; Error [deg]: 3.706060e+000



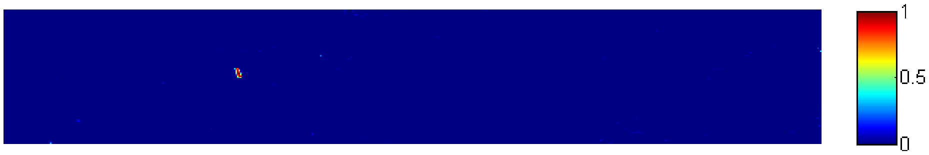
Material: 12; Argillaceous Limestone ; Error [deg]: 2.045718e+000



Material: 13; Dipyre BM1959,505.HLsp W1R1Bb AREF; Error [deg]: 1.675909e+000



Material: 14; White fiberglass unspecified rubber ; Error [deg]: 5.326605e+000



Material: 15; Very pale brown to brownish yellow interior dry gravelly silt loam ; Error [deg]: 4.208866e+000

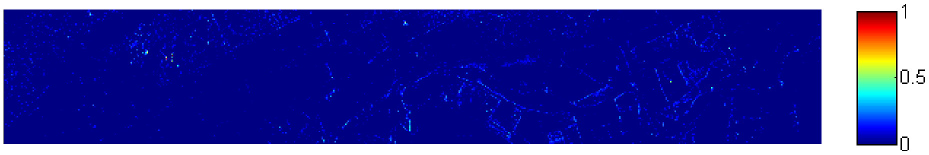
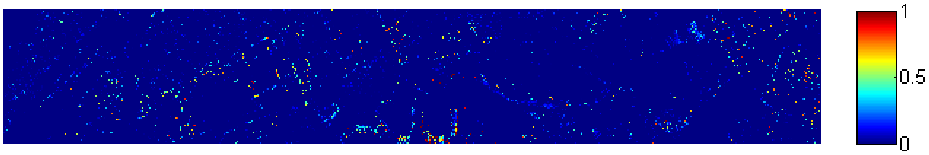
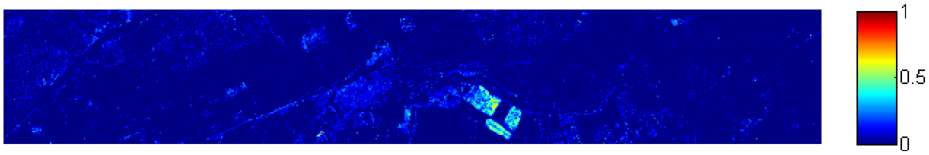
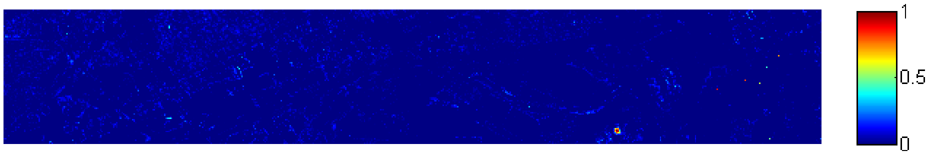
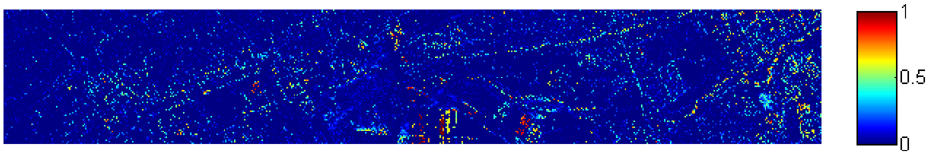


Table 41 – Continued


Material: 16; Grass_dry.7+.3green AMX30 W1R1Ba AREF; Error [deg]: 1.323112e+001


Material: 17; Lichen Xanthoparmelia-2 W1R1Fb AREF; Error [deg]: 3.920372e+000


Material: 18; Plastic GGA-54 Grnhouse Roof W1R1Ba AREF; Error [deg]: 6.177183e+000


Material: 19; Sagebrush Sage-Leaves-1 dry W1R1Fa AREF; Error [deg]: 1.822974e+001

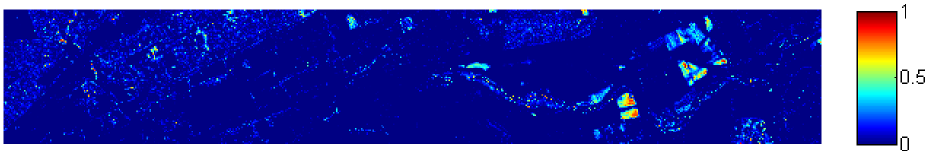
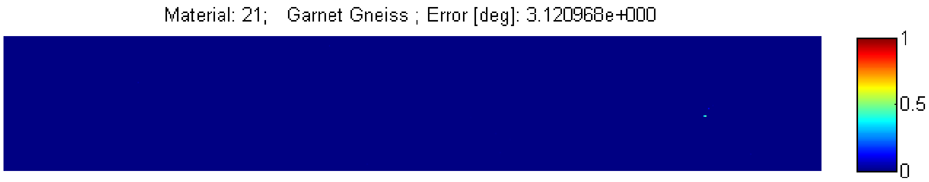
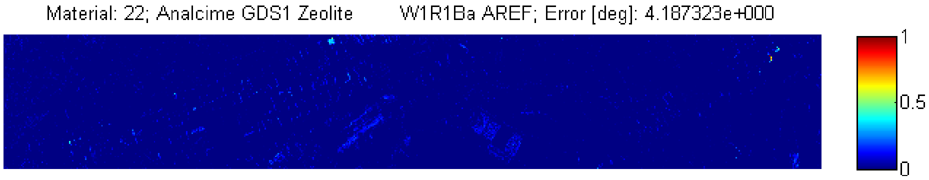
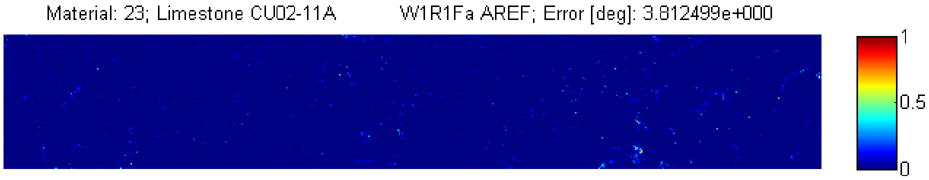
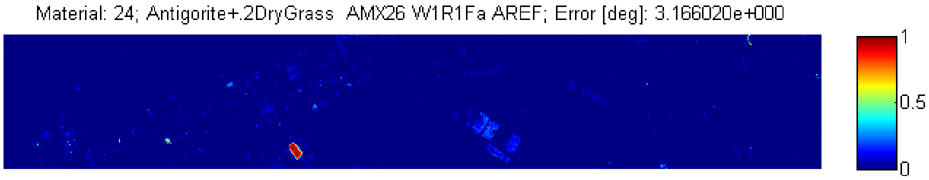

Material: 20; Oak Oak-Leaf-1 fresh W1R1Fa AREF; Error [deg]: 2.972115e+000

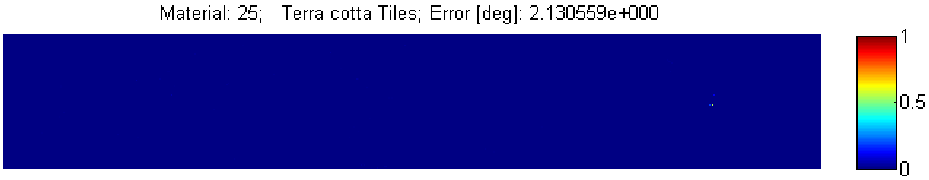
Table 41 – Continued



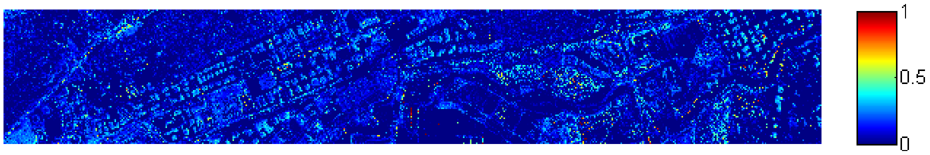




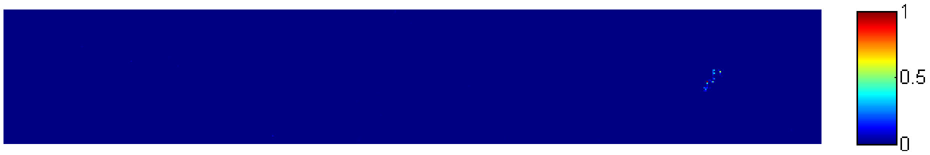




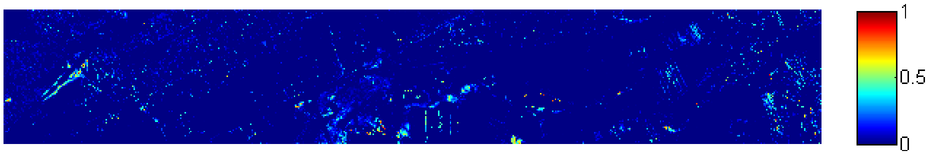
Material: 26;   Reddish asphalt Shingle; Error [deg]: 4.939936e+000



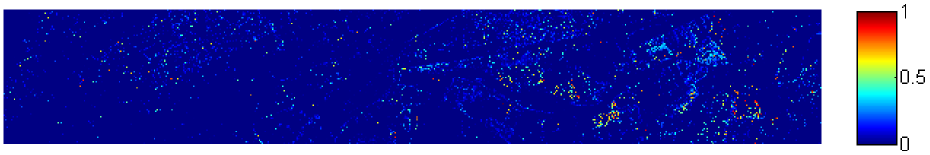
Material: 27; Spessartine WS481 Garnet   W1R1B? AREF; Error [deg]: 3.939407e+000



Material: 28; Juniper_Bush IH91-4B   W1R1Ba AREF; Error [deg]: 8.861385e+000



Material: 29; Stonwll_Ply+.2grnGrass AMX34 W1R1Ba AREF; Error [deg]: 4.510930e+000



Material: 30; Dry grass; Error [deg]: 3.494427e+000

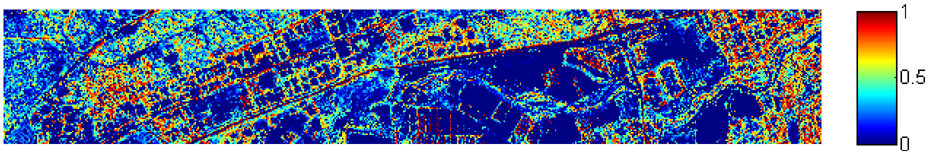
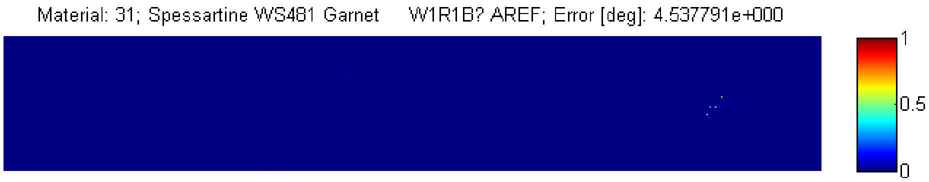
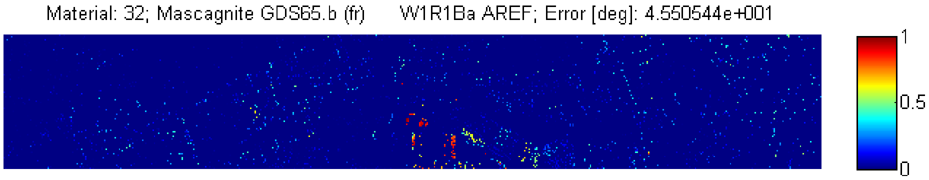
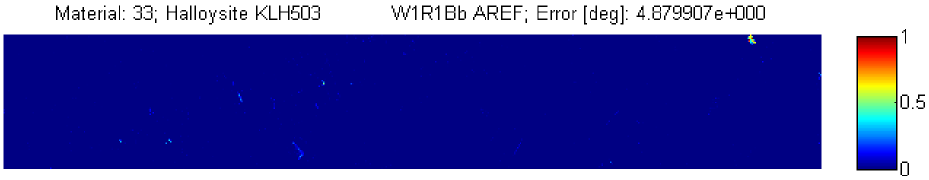
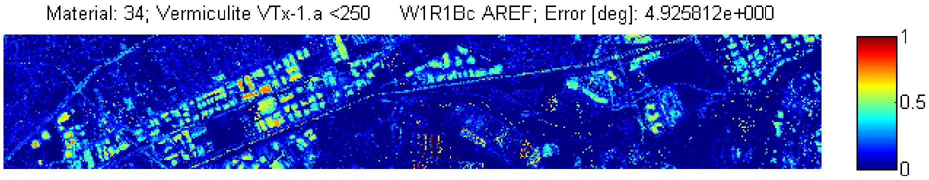


Table 41 – Continued


Material: 31; Spessartine WS481 Garnet   W1R1B? AREF; Error [deg]: 4.537791e+000
1
0.5
0


Material: 32; Mascagnite GDS65.b (fr)   W1R1Ba AREF; Error [deg]: 4.550544e+001
1
0.5
0


Material: 33; Halloysite KLH503   W1R1Bb AREF; Error [deg]: 4.879907e+000
1
0.5
0


Material: 34; Vermiculite VTx-1.a <250   W1R1Bc AREF; Error [deg]: 4.925812e+000
1
0.5
0

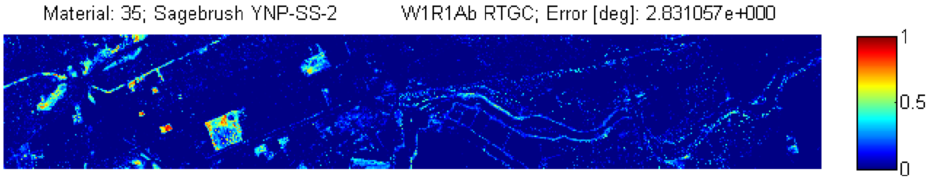
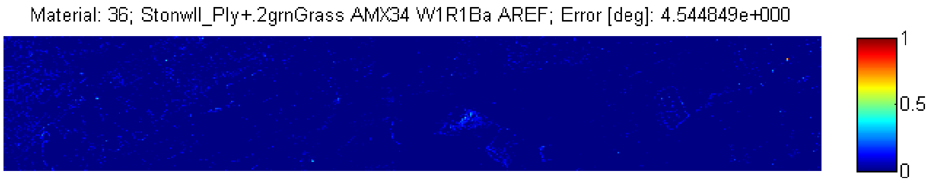
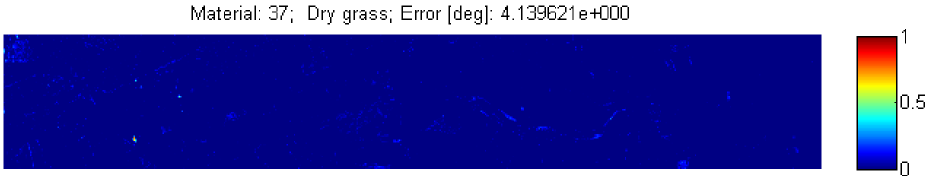
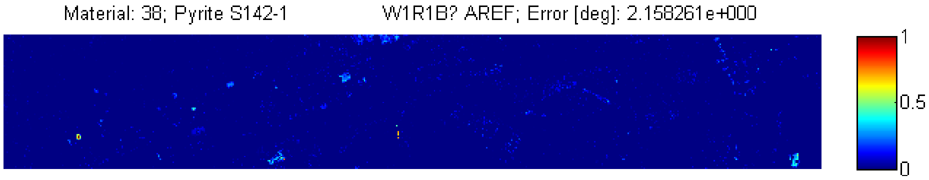
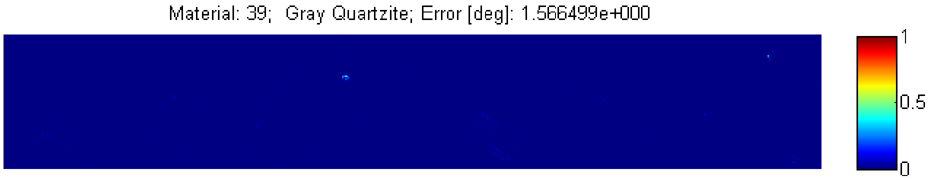

Material: 35; Sagebrush YNP-SS-2   W1R1Ab RTGC; Error [deg]: 2.831057e+000
1
0.5
0

Table 41 – Continued









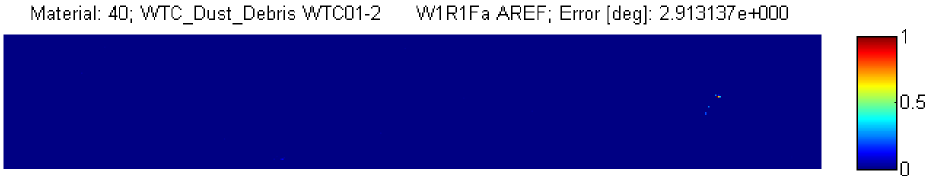
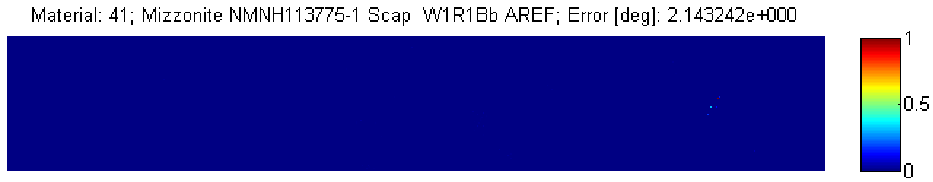
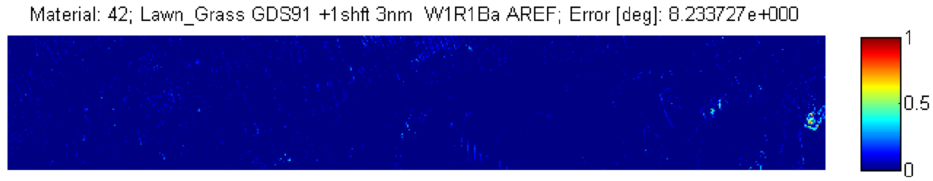
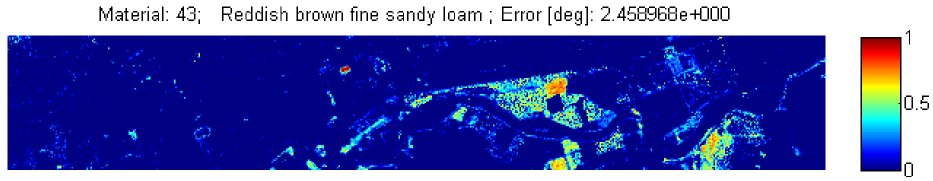
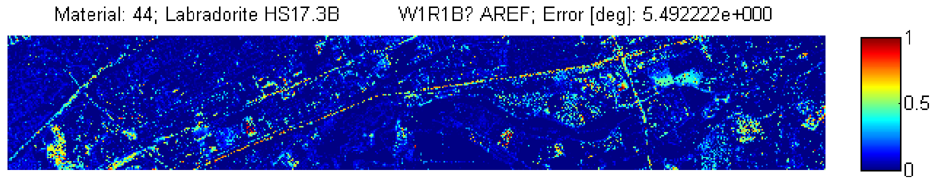


Table 41 – Continued


Material: 41; Mizzonite NMNH113775-1 Scap W1R1Bb AREF; Error [deg]: 2.143242e+000


Material: 42; Lawn_Grass GDS91 +1shft 3nm W1R1Ba AREF; Error [deg]: 8.233727e+000


Material: 43; Reddish brown fine sandy loam ; Error [deg]: 2.458968e+000


Material: 44; Labradorite HS17.3B W1R1B? AREF; Error [deg]: 5.492222e+000

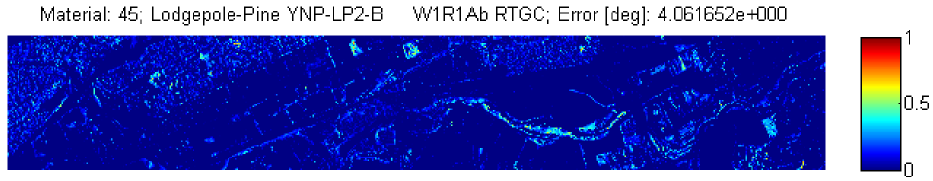
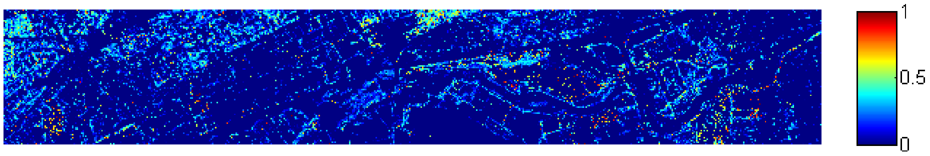
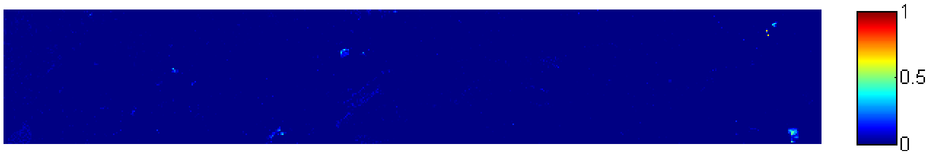
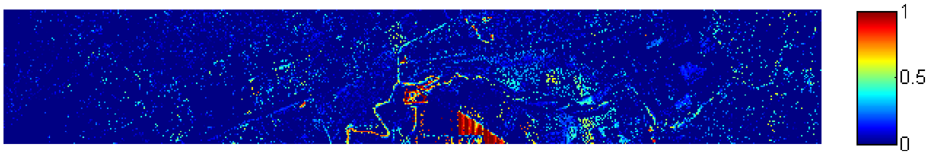

Material: 45; Lodgepole-Pine YNP-LP2-B W1R1Ab RTGC; Error [deg]: 4.061652e+000

Table 41 – Continued


Material: 46; Stonwll_Ply+.1grnGrass AMX33 W1R1Ba AREF; Error [deg]: 2.820768e+000

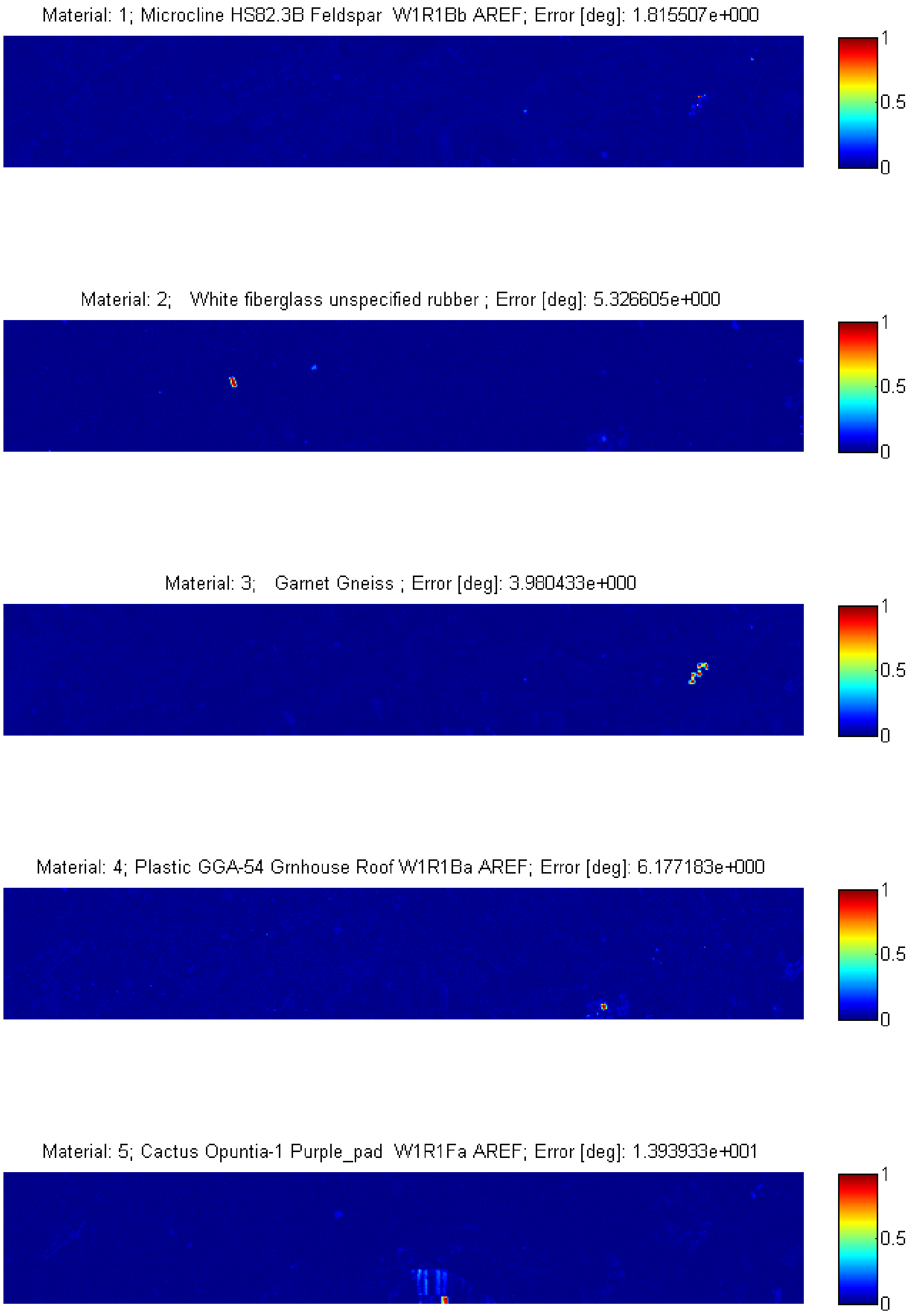

Material: 47; Aluminum Metal; Error [deg]: 1.678708e+000


Material: 48; Mascagnite GDS65.b (fr) W1R1Ba AREF; Error [deg]: 2.793295e+001

Material: 49; Garnet Gneiss ; Error [deg]: 3.980433e+000

Material: 50; Labradorite HS17.3B W1R1B? AREF; Error [deg]: 3.612413e+000

Table 42: Abundance maps for Moffett Field, CA Dataset. Endmembers derived from ICA-EEA algorithm.



Material: 1; Microcline HS82.3B Feldspar W1R1Bb AREF; Error [deg]: 1.815507e+000

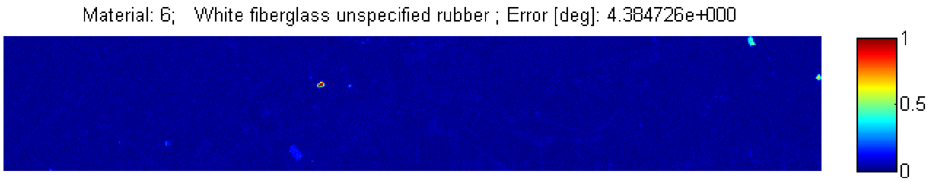
Material: 2; White fiberglass unspecified rubber ; Error [deg]: 5.326605e+000

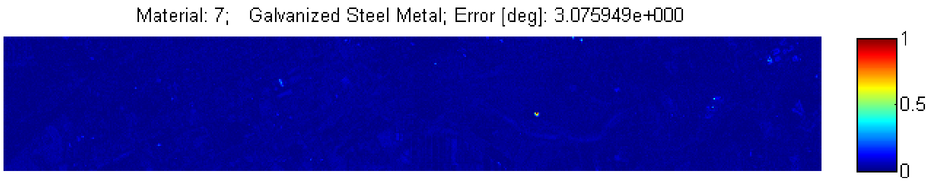
Material: 3; Garnet Gneiss ; Error [deg]: 3.980433e+000

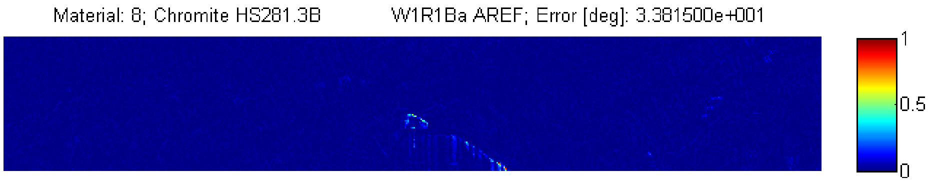
Material: 4; Plastic GGA-54 Grnhouse Roof W1R1Ba AREF; Error [deg]: 6.177183e+000

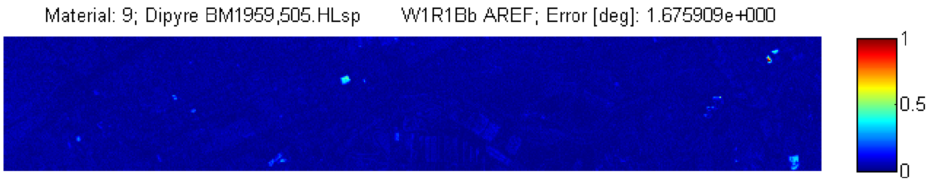
Material: 5; Cactus Opuntia-1 Purple_pad W1R1Fa AREF; Error [deg]: 1.393933e+001

Table 42 – Continued









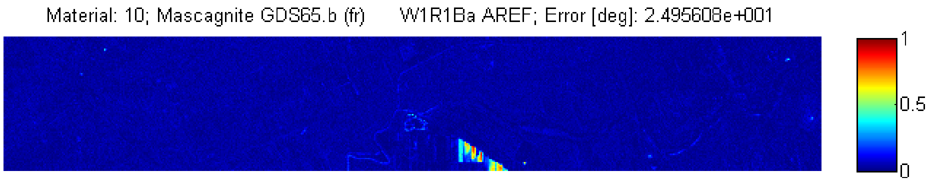
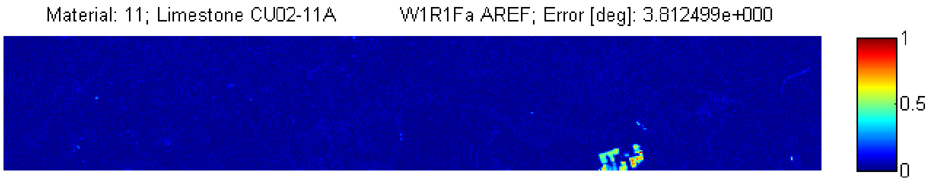
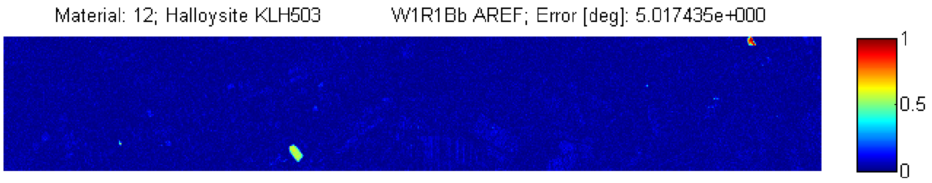
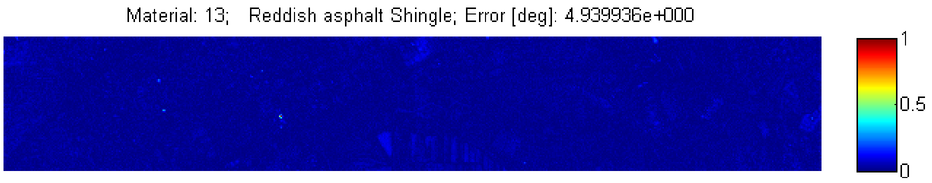
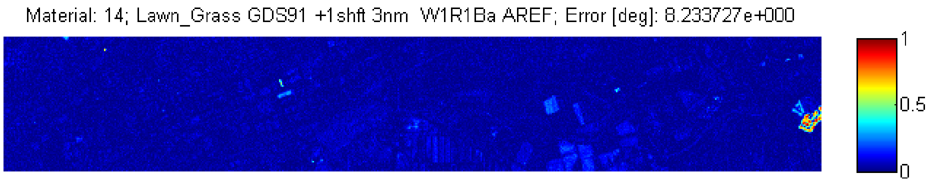


Table 42 – Continued


Material: 11; Limestone CU02-11A W1R1Fa AREF; Error [deg]: 3.812499e+000


Material: 12; Halloysite KLH503 W1R1Bb AREF; Error [deg]: 5.017435e+000


Material: 13; Reddish asphalt Shingle; Error [deg]: 4.939936e+000


Material: 14; Lawn_Grass GDS91 +1shft 3nm W1R1Ba AREF; Error [deg]: 8.233727e+000

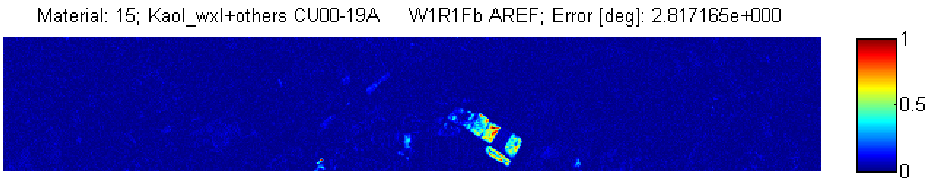
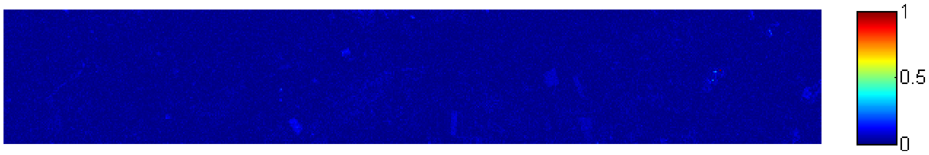
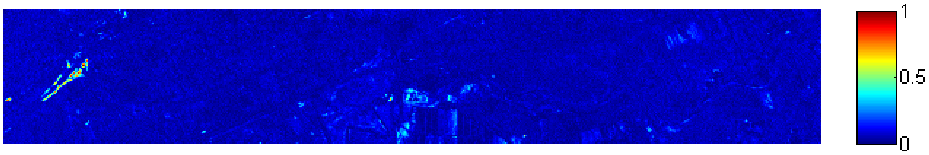

Material: 15; Kaol_wxl+others CU00-19A W1R1Fb AREF; Error [deg]: 2.817165e+000

Table 42 – Continued

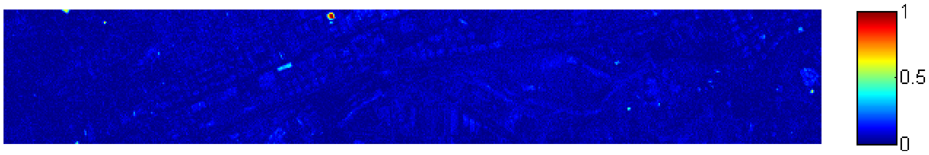
Material: 16; Argillaceous Limestone ; Error [deg]: 2.045718e+000



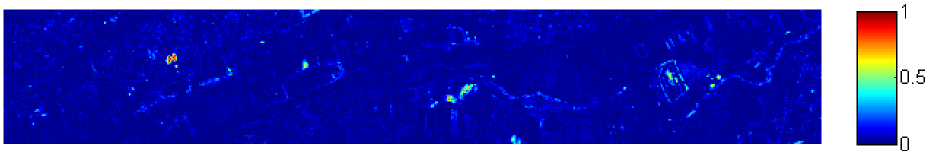
Material: 17; Juniper_Bush IH91-4B   W1R1Ba AREF; Error [deg]: 8.861385e+000



Material: 18; Painted_Aluminum GDS333 LgGr W1R1Fa AREF; Error [deg]: 2.620058e+000



Material: 19; LeafySpurge Spurge-A1-Jul98 W1R1Fa AREF; Error [deg]: 3.927492e+000



Material: 20; Dry grass; Error [deg]: 3.494427e+000

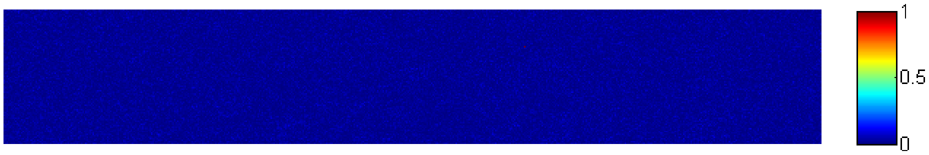
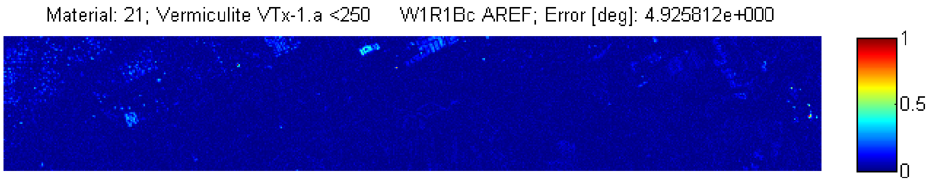
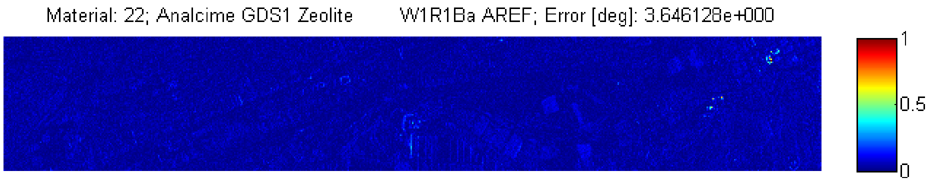
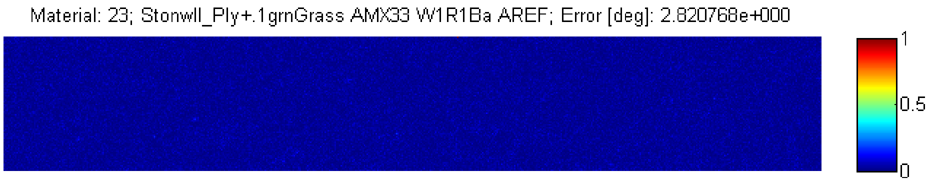
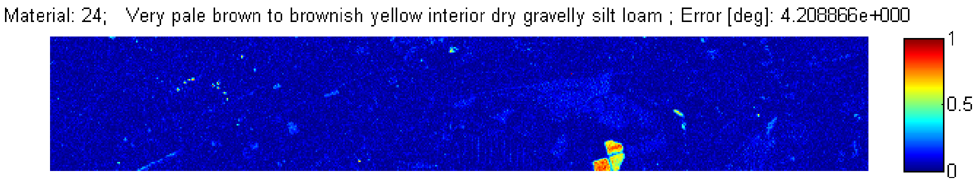


Table 42 – Continued


Material: 21; Vermiculite VTx-1.a <250 W1R1Bc AREF; Error [deg]: 4.925812e+000
1
0.5
0


Material: 22; Analcime GDS1 Zeolite W1R1Ba AREF; Error [deg]: 3.646128e+000
1
0.5
0


Material: 23; Stonwll_Ply+.1grnGrass AMX33 W1R1Ba AREF; Error [deg]: 2.820768e+000
1
0.5
0


Material: 24; Very pale brown to brownish yellow interior dry gravelly silt loam ; Error [deg]: 4.208866e+000
1
0.5
0

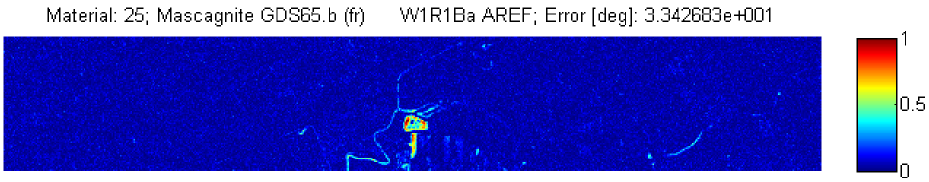
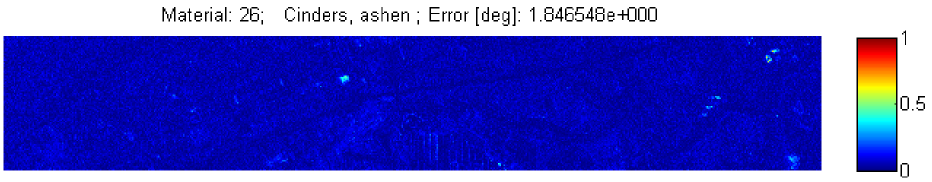
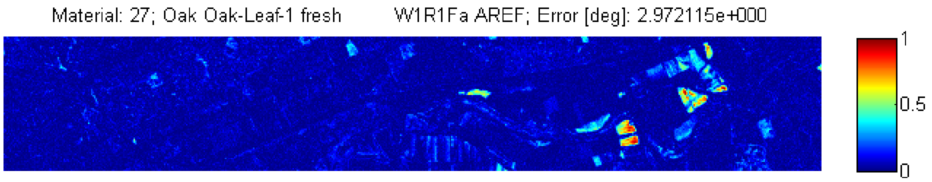
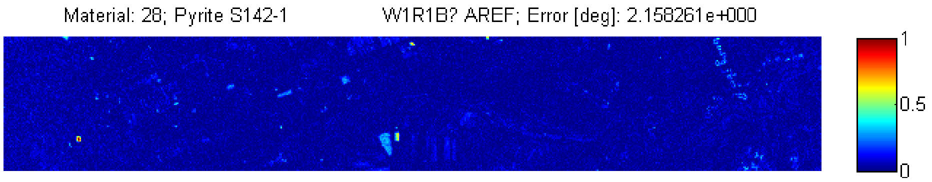
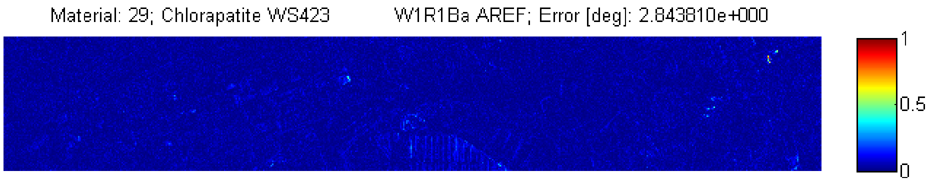

Material: 25; Mascagnite GDS65.b (fr) W1R1Ba AREF; Error [deg]: 3.342683e+001
1
0.5
0

Table 42 – Continued


Material: 26;   Cinders, ashen ; Error [deg]: 1.846548e+000


Material: 27; Oak Oak-Leaf-1 fresh   W1R1Fa AREF; Error [deg]: 2.972115e+000


Material: 28; Pyrite S142-1   W1R1B? AREF; Error [deg]: 2.158261e+000


Material: 29; Chlorapatite WS423   W1R1Ba AREF; Error [deg]: 2.843810e+000

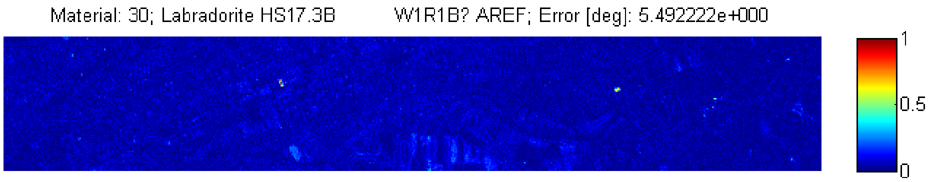
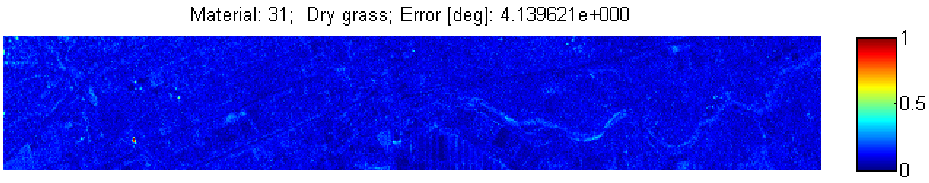
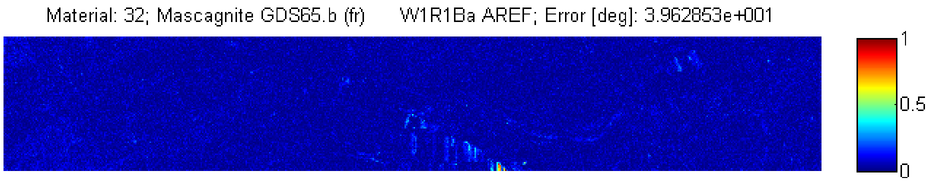
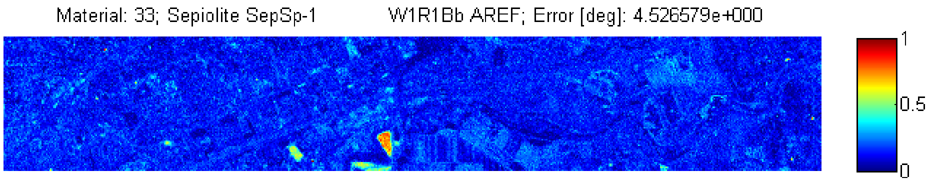
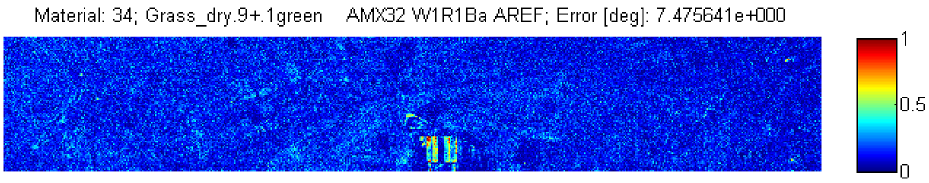

Material: 30; Labradorite HS17.3B   W1R1B? AREF; Error [deg]: 5.492222e+000

Table 42 – Continued









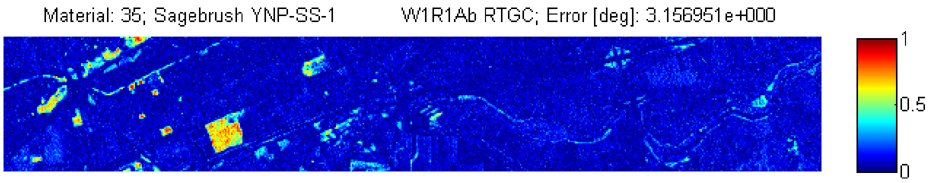
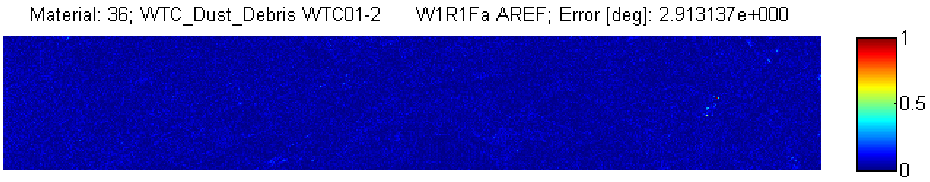
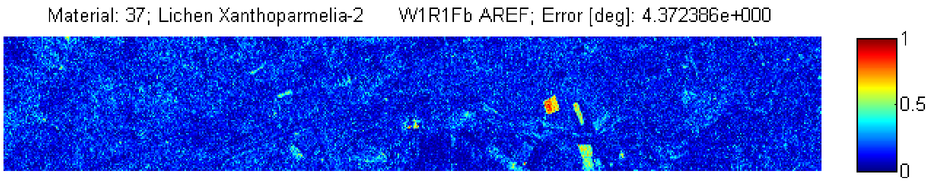
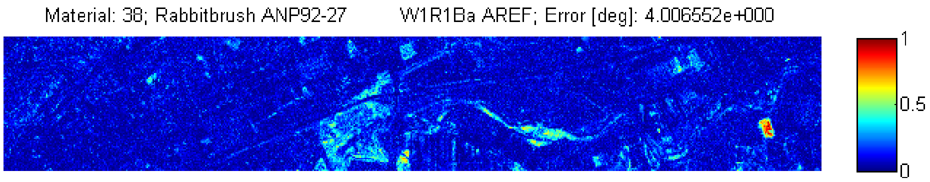
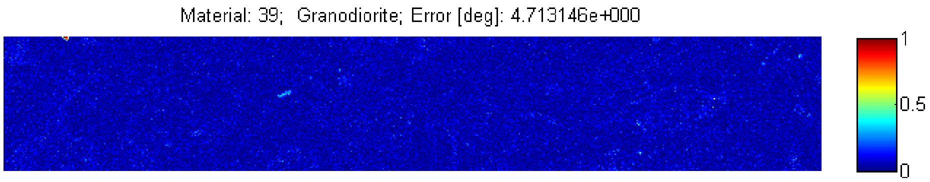


Table 42 – Continued


Material: 36; WTC_Dust_Debris WTC01-2   W1R1Fa AREF; Error [deg]: 2.913137e+000
1
0.5
0


Material: 37; Lichen Xanthoparmelia-2   W1R1Fb AREF; Error [deg]: 4.372386e+000
1
0.5
0


Material: 38; Rabbitbrush ANP92-27   W1R1Ba AREF; Error [deg]: 4.006552e+000
1
0.5
0



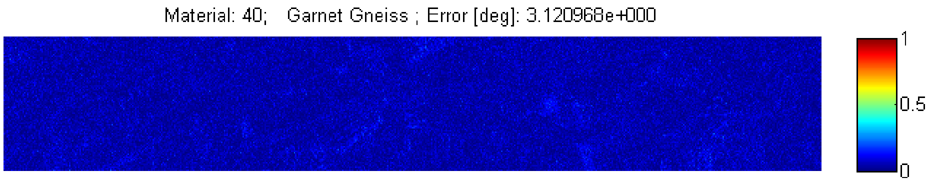
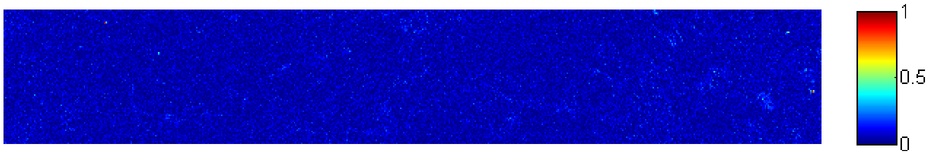
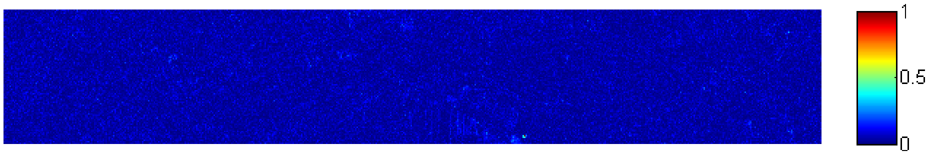


Table 42 – Continued

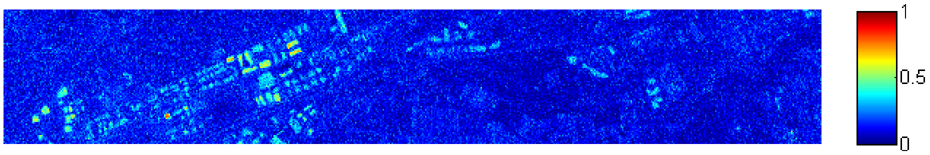
Material: 41; Mizzonite HS350.3B HLSp Scap W1R1Ba AREF; Error [deg]: 4.409249e+000



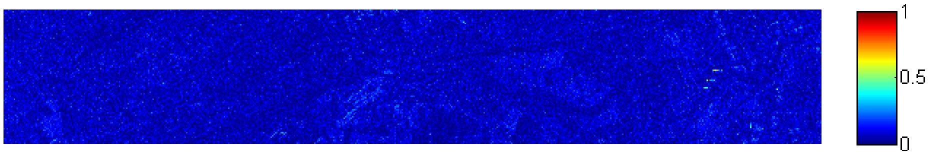
Material: 42; Grass_dry.7+.3green AMX30 W1R1Ba AREF; Error [deg]: 1.323112e+001



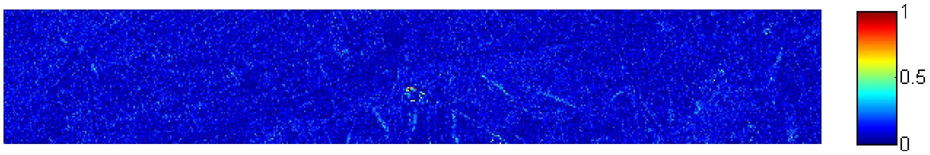
Material: 43; Antigorite+.2DryGrass AMX26 W1R1Fa AREF; Error [deg]: 3.633828e+000

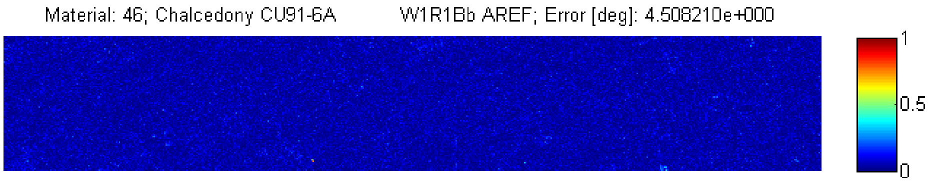


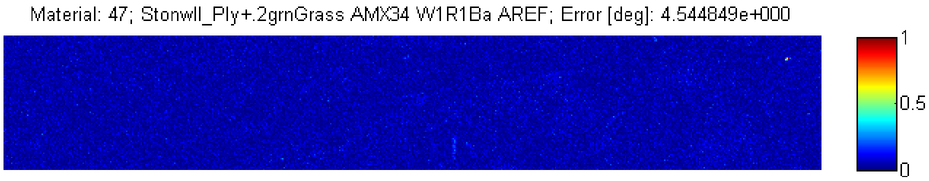
Material: 44; Andradite GDS12 Garnet W1R1Bc AREF; Error [deg]: 2.580414e+000

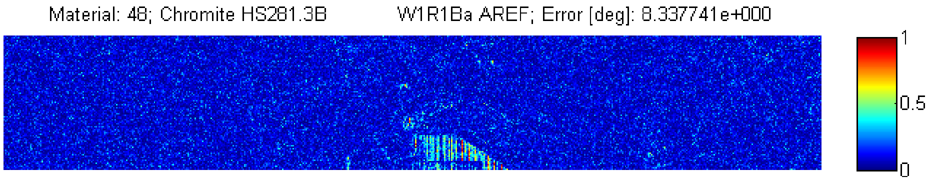


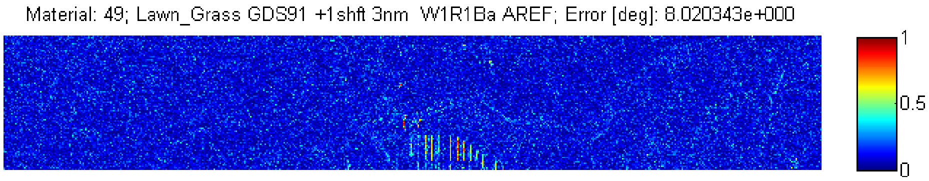
Material: 45; Melting_snow mSnw9+0.5_veg W1R1Fa AREF; Error [deg]: 1.701608e+001




Material: 46; Chalcedony CU91-6A W1R1Bb AREF; Error [deg]: 4.508210e+000


Material: 47; Stonwll_Ply+.2grnGrass AMX34 W1R1Ba AREF; Error [deg]: 4.544849e+000


Material: 48; Chromite HS281.3B W1R1Ba AREF; Error [deg]: 8.337741e+000


Material: 49; Lawn_Grass GDS91 +1shft 3nm W1R1Ba AREF; Error [deg]: 8.020343e+000

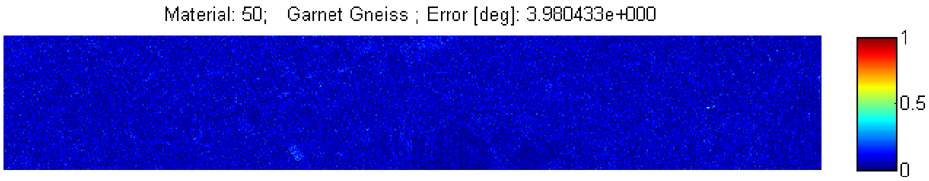
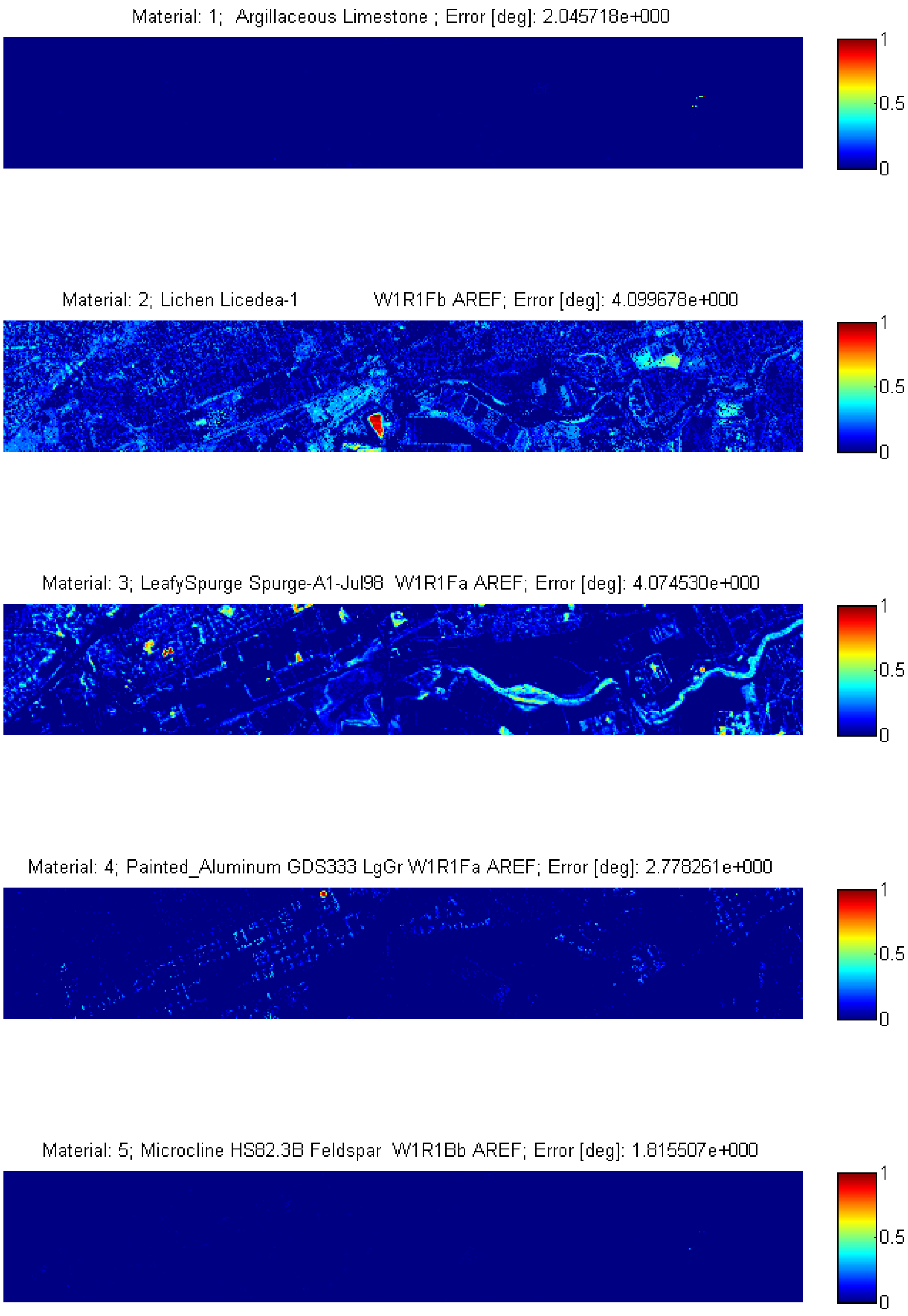

Material: 50; Garnet Gneiss ; Error [deg]: 3.980433e+000

Table 43: Abundance maps for Moffett Field, CA Dataset. Endmembers derived from VCA Raw Pixels algorithm.



Material: 1; Argillaceous Limestone ; Error [deg]: 2.045718e+000

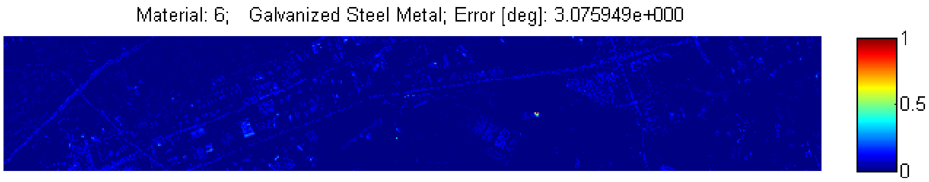
Material: 2; Lichen Licedea-1 W1R1Fb AREF; Error [deg]: 4.099678e+000

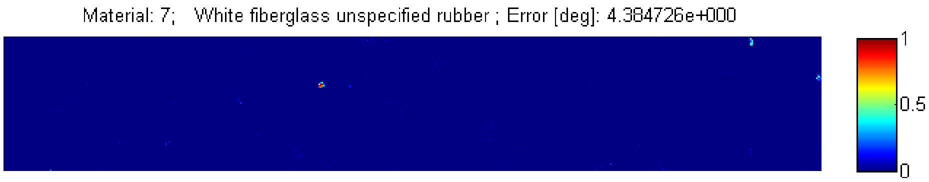
Material: 3; LeafySpurge Spurge-A1-Jul98 W1R1Fa AREF; Error [deg]: 4.074530e+000

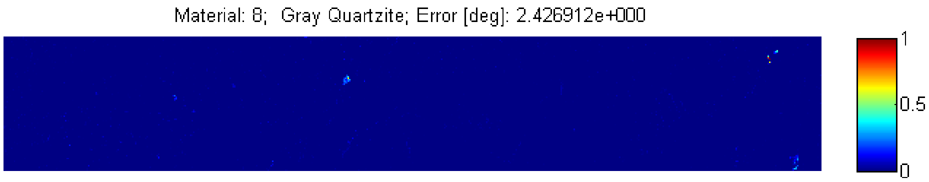
Material: 4; Painted_Aluminum GDS333 LgGr W1R1Fa AREF; Error [deg]: 2.778261e+000

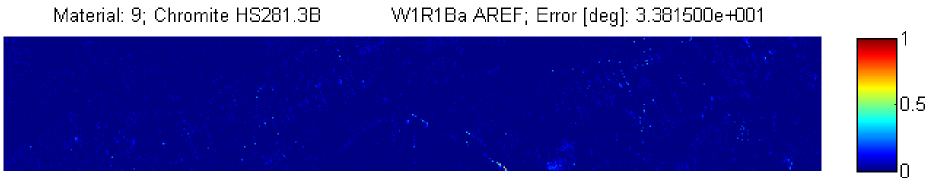
Material: 5; Microcline HS82.3B Feldspar W1R1Bb AREF; Error [deg]: 1.815507e+000

Table 43 – Continued









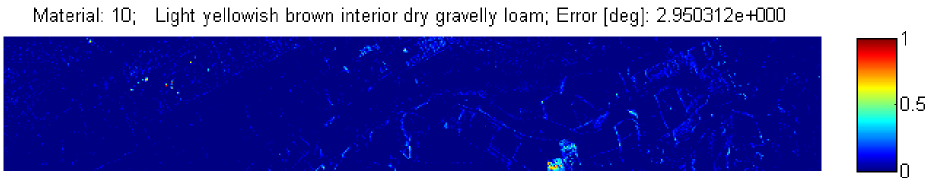
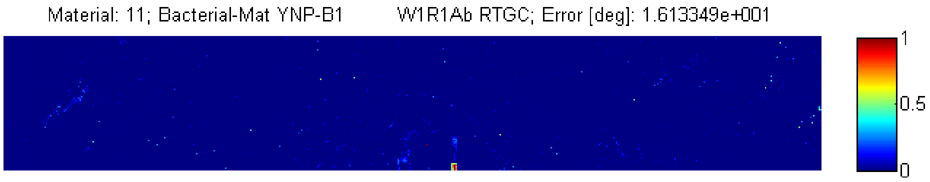
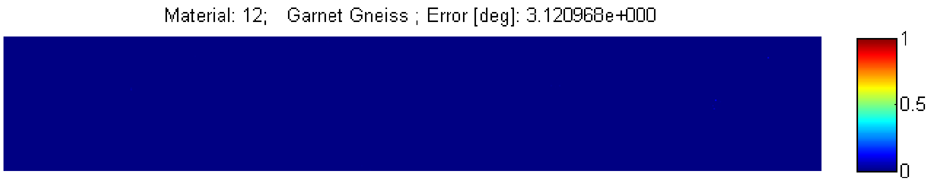
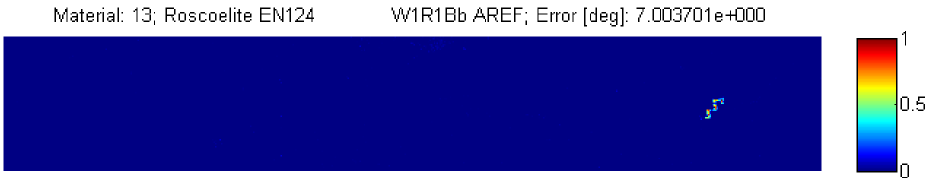
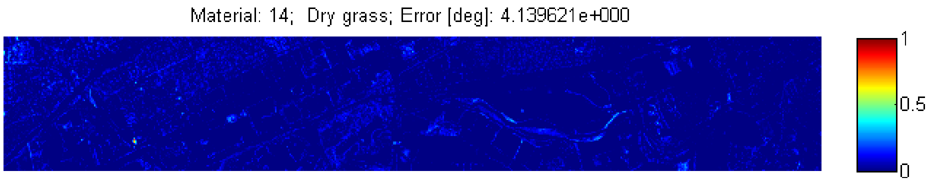


Table 43 – Continued









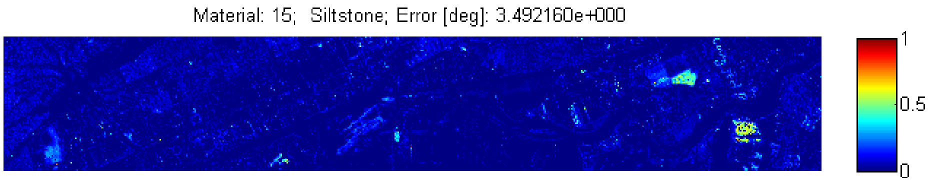
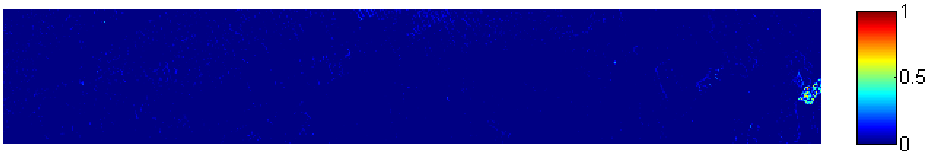
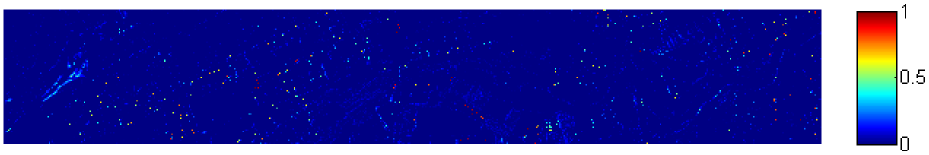
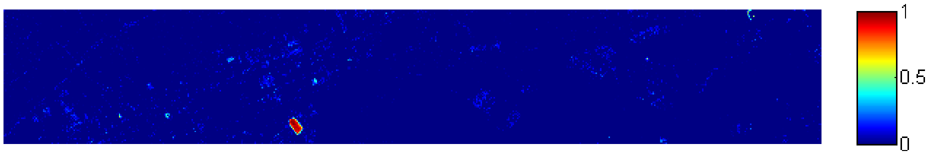
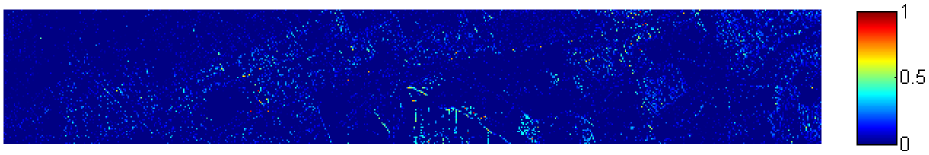


Table 43 – Continued


Material: 16; Lawn_Grass GDS91 +1shft 3nm W1R1Ba AREF; Error [deg]: 8.233727e+000


Material: 17; Grass_dry.7+.3green   AMX30 W1R1Ba AREF; Error [deg]: 1.323112e+001


Material: 18; Antigorite+.2DryGrass AMX26 W1R1Fa AREF; Error [deg]: 3.274267e+000


Material: 19; Pectolite NMNH94865.a   W1R1Ba AREF; Error [deg]: 1.724171e+001

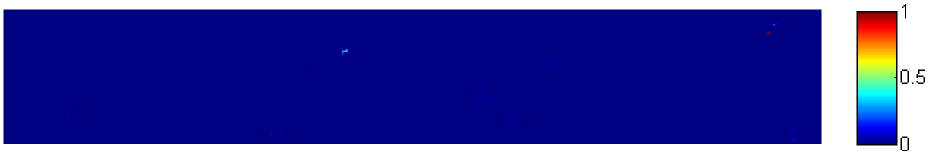
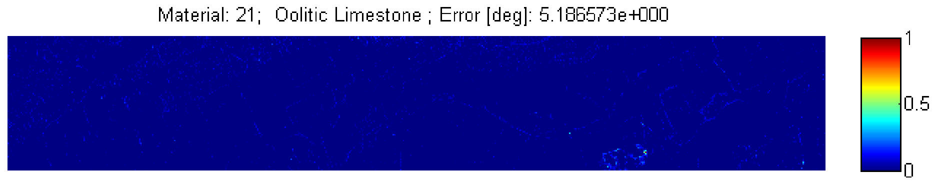
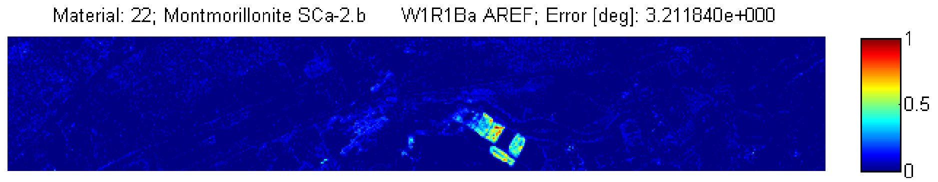
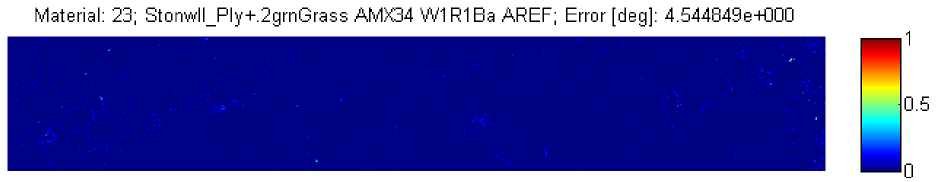
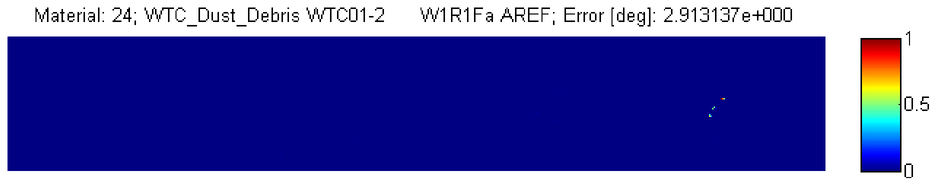

Material: 20; Dipyre BM1959,505.HLsp   W1R1Bb AREF; Error [deg]: 1.675909e+000

Table 43 – Continued









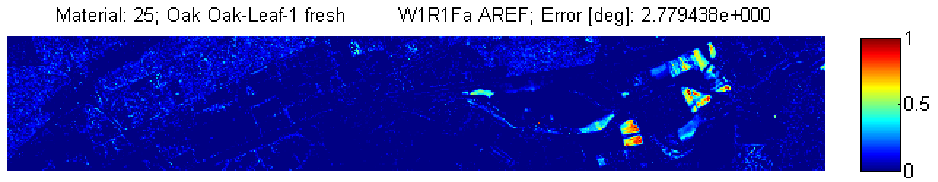
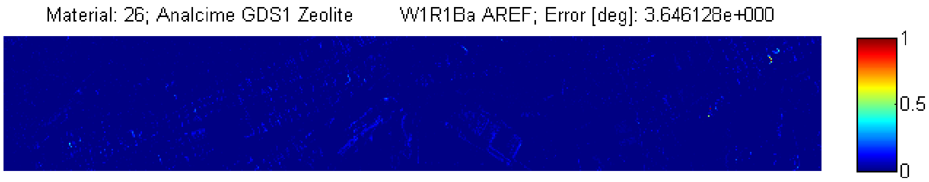
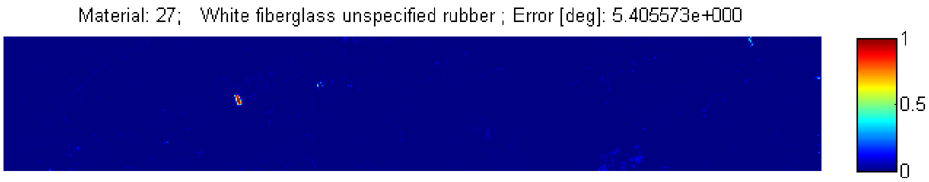
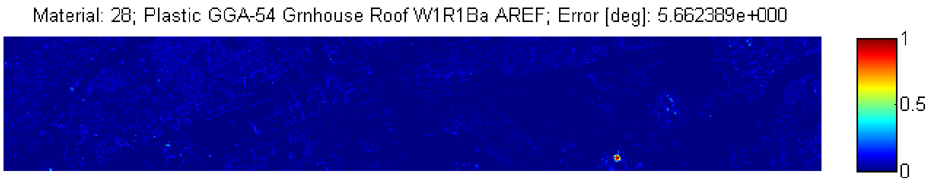
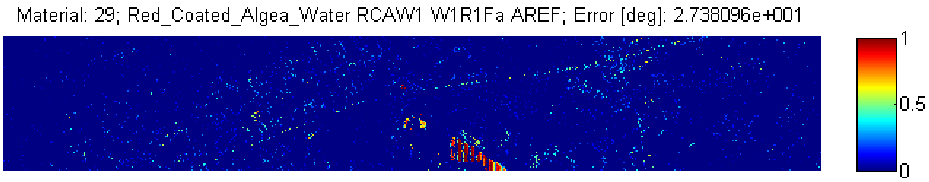


Table 43 – Continued









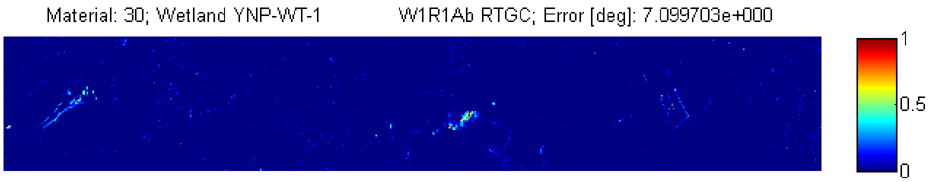
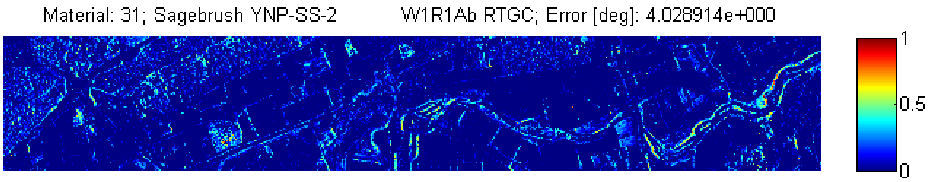
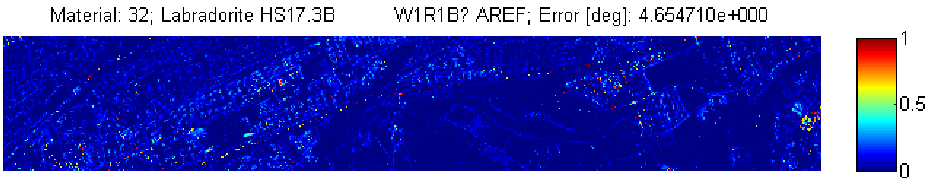
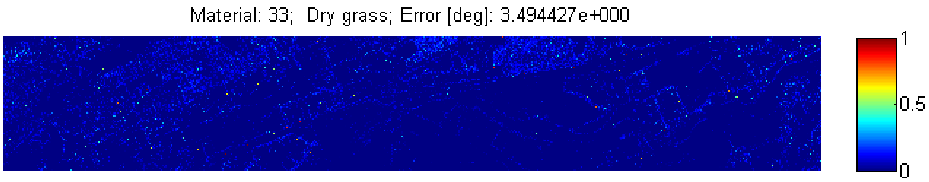
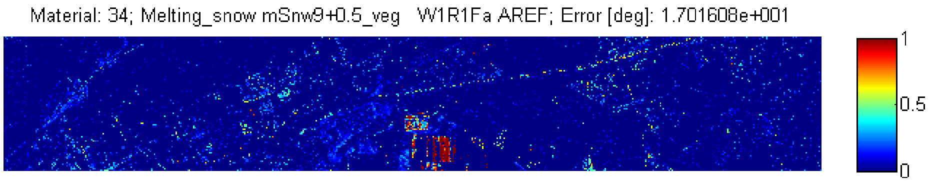


Table 43 – Continued









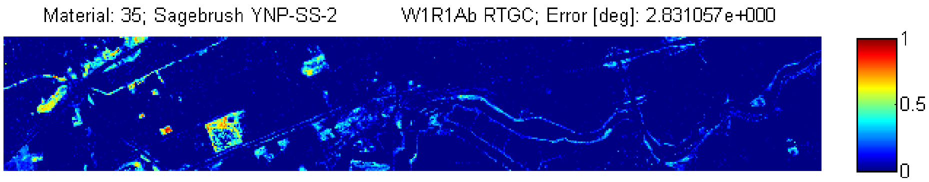
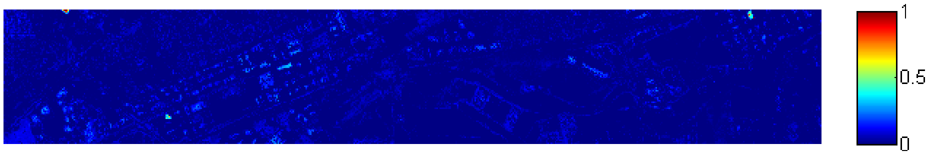
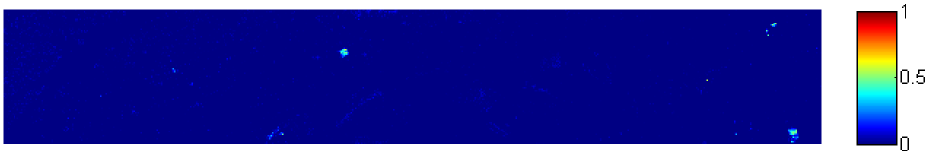


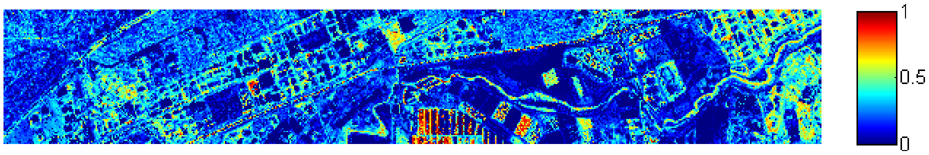
Table 43 – Continued



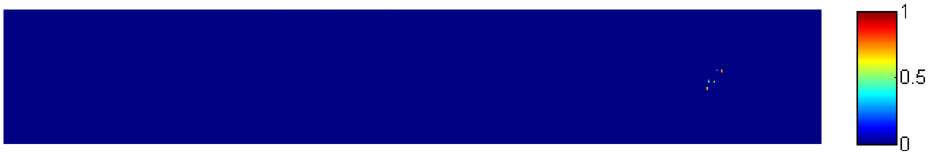
Material: 36; Granodiorite; Error [deg]: 4.887420e+000



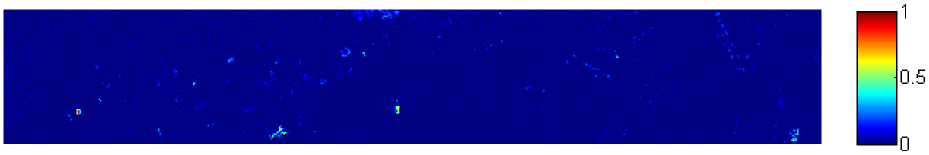
Material: 37;   Cinders, ashen ; Error [deg]: 1.846548e+000



Material: 38; Flower Platycodon-2 White   W1R1Fa AREF; Error [deg]: 8.026147e+000

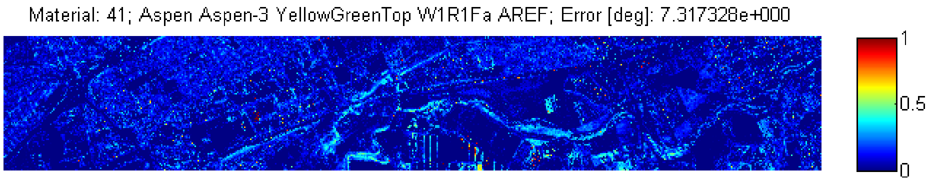


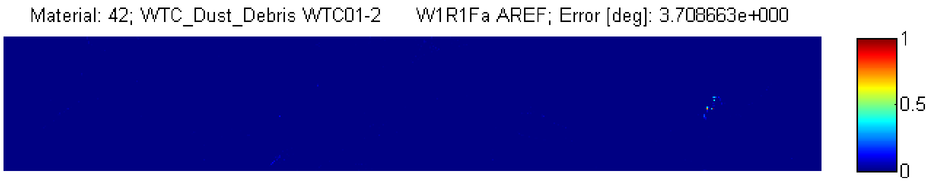
Material: 39; Spessartine WS481 Garnet   W1R1B? AREF; Error [deg]: 4.537791e+000

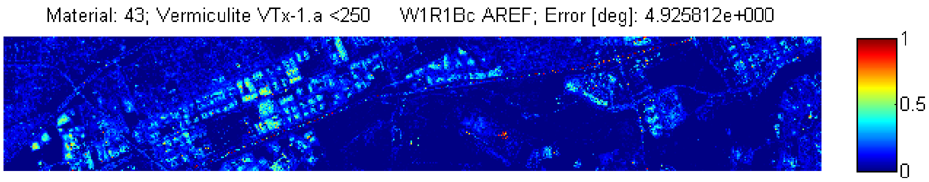


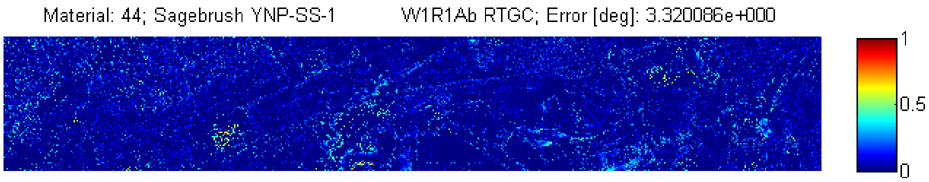
Material: 40; Pyrite S142-1   W1R1B? AREF; Error [deg]: 2.042628e+000

Table 43 – Continued


Material: 41; Aspen Aspen-3 YellowGreenTop W1R1Fa AREF; Error [deg]: 7.317328e+000
1
0.5
0


Material: 42; WTC_Dust_Debris WTC01-2 W1R1Fa AREF; Error [deg]: 3.708663e+000
1
0.5
0


Material: 43; Vermiculite VTx-1.a <250 W1R1Bc AREF; Error [deg]: 4.925812e+000
1
0.5
0


Material: 44; Sagebrush YNP-SS-1 W1R1Ab RTGC; Error [deg]: 3.320086e+000
1
0.5
0

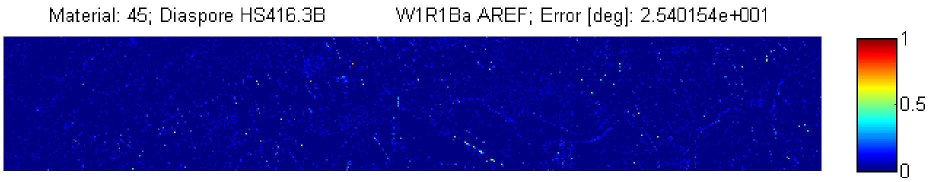
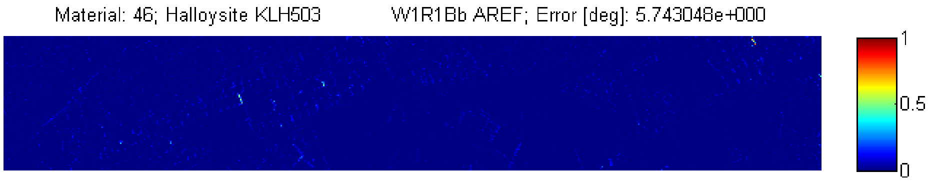
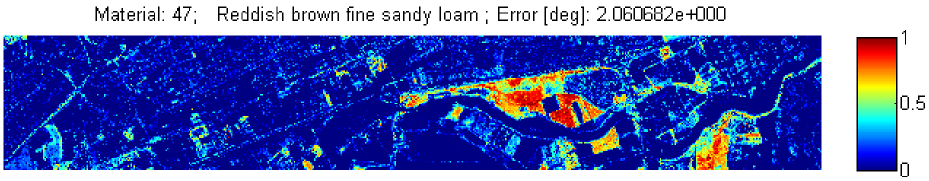
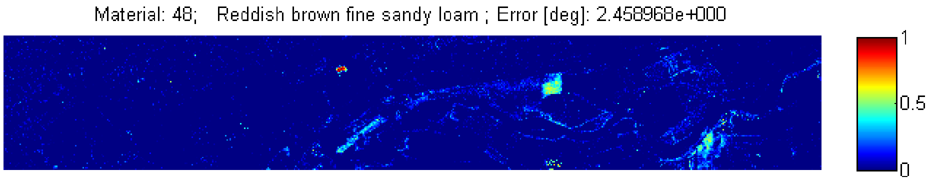
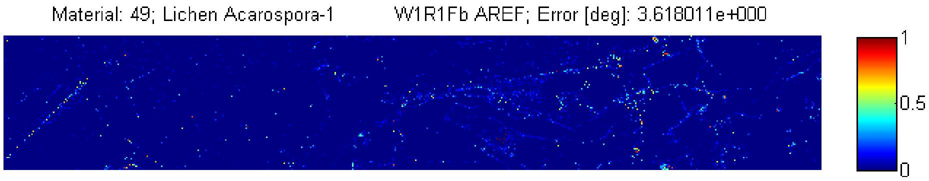

Material: 45; Diaspore HS416.3B W1R1Ba AREF; Error [deg]: 2.540154e+001
1
0.5
0

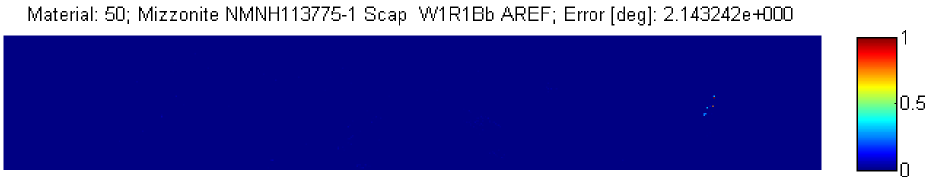
Table 43 – Continued











9 Discussion

The efficacy of the algorithms and associated strengths and weakness are discussed in this section.

9.1 Algorithm Consistency Problem

Both ICA-EEA and VCA suffer from an inability to produce consistent results each run. It is recommended each algorithm be executed several times to reach a good consensus of a true result. The randomness in VCA's results stems from the use of a random vector projection to prevent data projection onto the nullspace. The randomness in ICA-EEA's results stems from the use of selecting random initialization vectors as the default starting points for the Newton iterations. ATGP does not suffer from this problem and provides deterministic results each run.

9.2 Algorithm Initialization Sensitivity

Both the ICA-EEA and VCA algorithms require initialization parameters which are chosen independent of the data. For VCA, the initial endmember space is initialized by a vector of $n-1$ zeros and a single one. For ICA-EEA, FastICA chooses random vectors as the initial starting points in the Newton iterations. It is recommended to seed the ICA-EEA algorithm with the results of the ATGP algorithm [4] [19]. It is worth noting, algorithm initialization effects for the legacy algorithms N-FINDR and PPI were examined in [20].

9.3 Algorithm Complexity

ATGP and VCA are similar in computational complexity especially if you neglect the VCA preconditioning steps. The differences in complexity of ATGP and VCA arise from the method used to project the data onto an orthogonal subspace. ATGP uses a projection matrix of size $p \times p$ and multiples each pixel by this projection matrix giving a complexity of $O(2p^2N)$. VCA projects all the pixels onto a random vector giving a complexity of $O(pN)$. Additionally, the ATGP algorithm must compute the magnitude of each resulting projection which is of complexity $O(2pN)$.

The complexity for ICA-EEA depends largely on the parameters used in the FastICA algorithm. Complexity is difficult to measure with this algorithm due to the algorithms non-deterministic nature. However, a lower bound can be created by remembering β in (46) must be computed each iteration and has complexity $O(2pN + 2N)$. Therefore, the overall complexity of the FastICA algorithm is no lower than approximately $O\{(2pN + 2N)kq\}$ where k is the number of iterations per component and q is the number of endmembers to be extracted.

9.4 ATGP

ATGP produced the best results of all the algorithms. Its error surface is most similar to the results of figure 5 of all the unmixing algorithm. The ATGP algorithms degrades as endmember variability decreases.

9.5 ICA-EEA

ICA-EEA took the longest to run out of all the algorithms. The solution for ICA-EEA is iteratively determined via the FastICA algorithm and stops when a convergence threshold is achieved or a maximum number of iterations has been executed. This attributes make the ICA-EEA algorithm have a non-deterministic run-time.

Despite what was found in [21], the ICA-EEA algorithm worked reasonable well when the number of endmembers was large. This may be attributed to endmember dependence, resulting from the abundance sum-to-one constraint (ASC), becoming weaker as the number of materials in the mixture increases. Figure 19 shows this phenomenon clearly as the SAM error (in the SAM error surface plot) decreases as the number of materials in the mixture increases.

9.6 VCA

The importance of selecting the q or $q-1$ projection should not be underestimated with this algorithm. The authors recommend the threshold to be $15 + 10\log_{10}(q)$; however, it was determined that this threshold does not always work. Our results showed a threshold of $SNR_{th} = 28 + \log_{10}(q)$ worked much better. One of the pitfalls of this threshold calculation is that it is compared with an SNR estimate whereby the estimate assumes the noise is white; this is not always the case. If the SNR calculation is not accurate, the threshold of selecting either projection is meaningless. Thus, both the threshold equation *and* the SNR estimate calculations must be done with care to ensure accuracy. Figure 33 shows how SNR can be estimated incorrectly for the SMDE dataset.

Note the differences in detection surfaces of figures 35 and 34. The only difference in these datasets are the reference set of endmembers. From this, we can infer that the distinction of reference endmembers among themselves is very important. The VD estimate surfaces also display this fact in figures 5 and 7.

The principal components transform (PCT) done as a preconditioning step greatly affected the algorithm's results. As the VCA algorithm is originally stated, the endmembers are selected from the pixels from which the PCT was performed. This method of endmember selection was found to be suboptimal as shown in the comparison of figures 25 and 33. This problem is corrected by simply selected endmembers from the raw, unconditioned data.

9.7 Moffett Field, CA Dataset

ATGP and VCA yield very similar results when processing this dataset. This is due to the similarities in the algorithms themselves. ICA-EEA had similar results to ATGP and VCA when the material abundance map was sparse. The abundance maps provided by ICA-EEA exhibit more material sparseness than those of ATGP and VCA in general. This sparseness is expected as ICA-EEA finds the most non-Gaussian components which appear as independent, localized patches of material.

10 Conclusion

10.1 Utility of This Work

This thesis provides the algorithms for three techniques to unmix hyperspectral data. It evaluates each of the algorithms against common datasets to determine their efficacy to extract a unique set of pure endmembers against varying endmember diversity and illumination inconsistency. Additionally, the thesis evacuates the virtual dimensionality algorithm at several false alarm rates against the same datasets to measure its efficacy in correctly determining the number of endmembers present in a scene.

The thesis provides a bound on SAM error verses SNR in figure 5. This figure can be used as a reference in determining system performance and discrimination capabilities.

The thesis provides a method for creating simulated, spatially dependent ground truth datasets using color imagery and hue clustering. The SMDE dataset demonstrates the validity and usefulness of such a method to great ground truth data.

Finally, the thesis demonstrates the ability of the ATGP algorithm, along with a FCLS algorithm, to create useful abundance maps in a automated fashion. These abundance maps can overlayed onto Google Earth after orthorectification. This is shown in figures 17, 15, 16, and 14.

10.2 Future Work

Many areas of further study can be continued to support this work.

1. The preference of orthogonal direction when selecting the next endmember in the ATGP and VCA algorithms. In some situations, it may be beneficial to modify the ATGP algorithm to preference for the pixel having the lowest maximum power of all the possible projections (directions). It is believed this would better flush out darker endmembers.

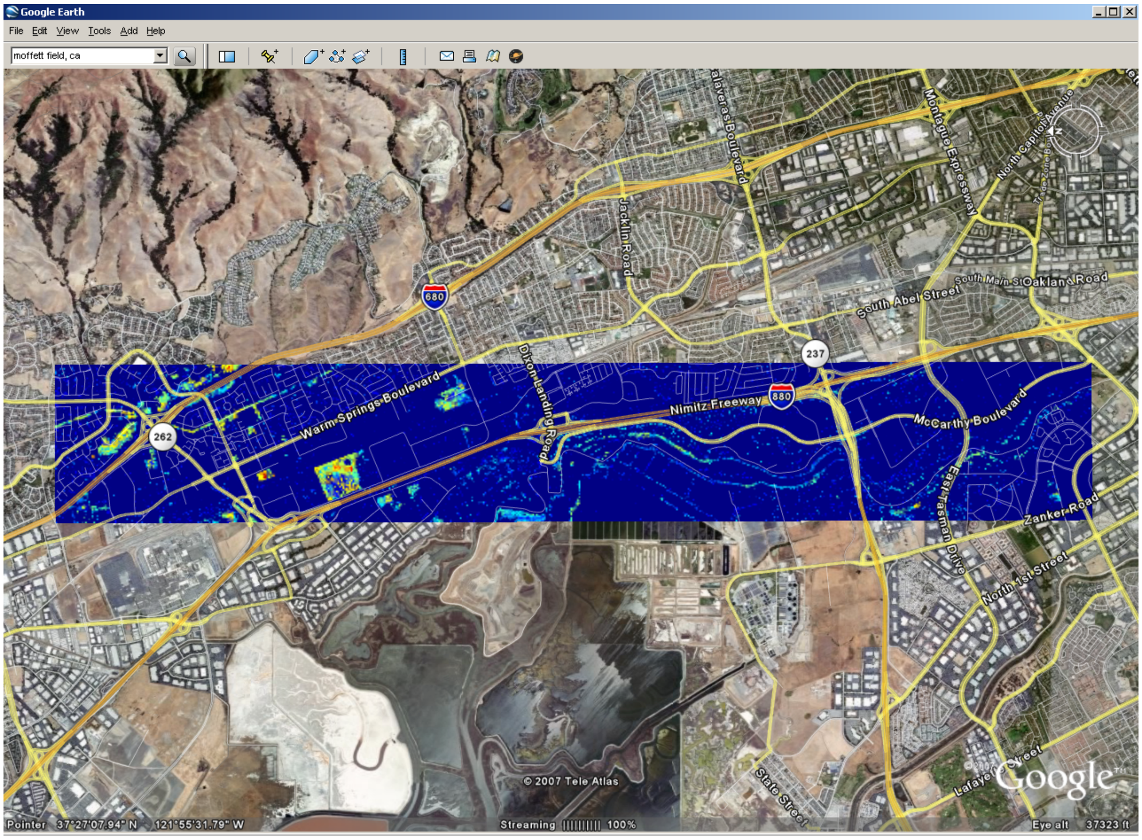
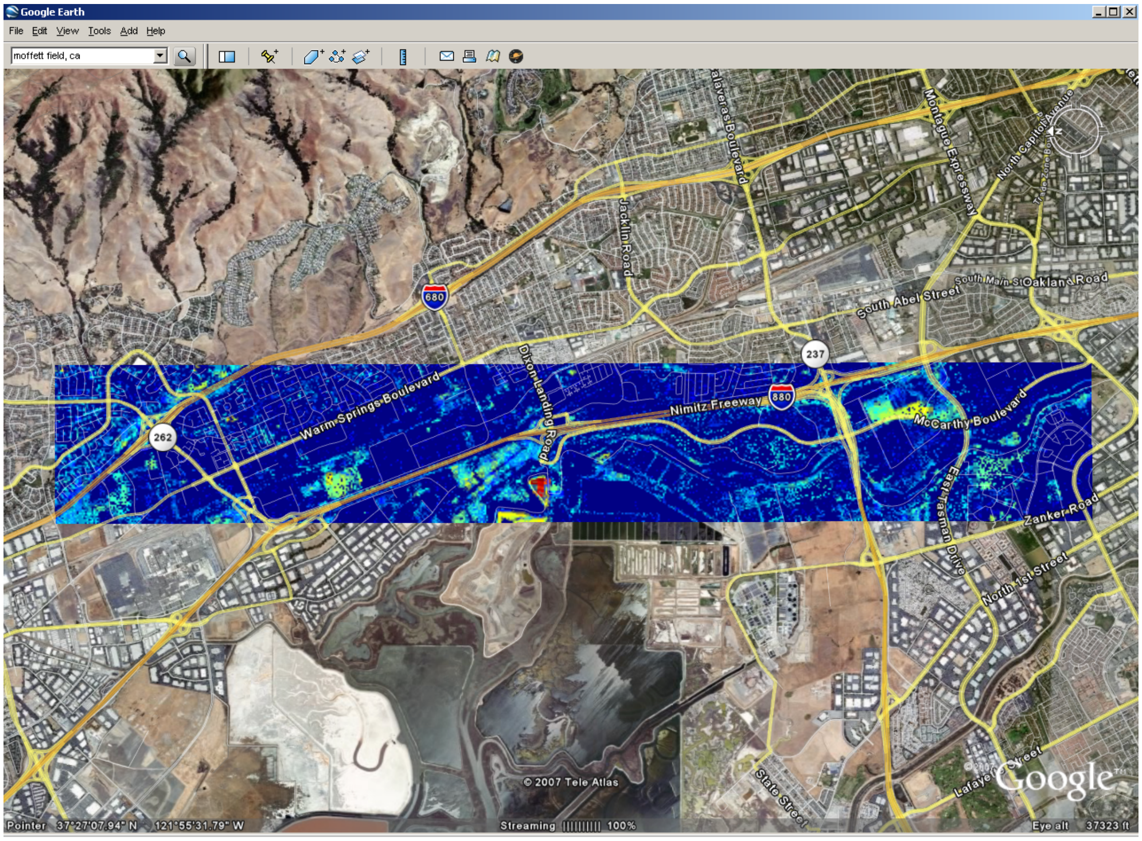


Figure 14: Abundance map of Sagebrush over Moffett Field, CA overlayed onto Google Earth.



moffett field, ca

Figure 15: Abundance map of Lichen over Moffett Field, CA overlayed onto Google Earth.

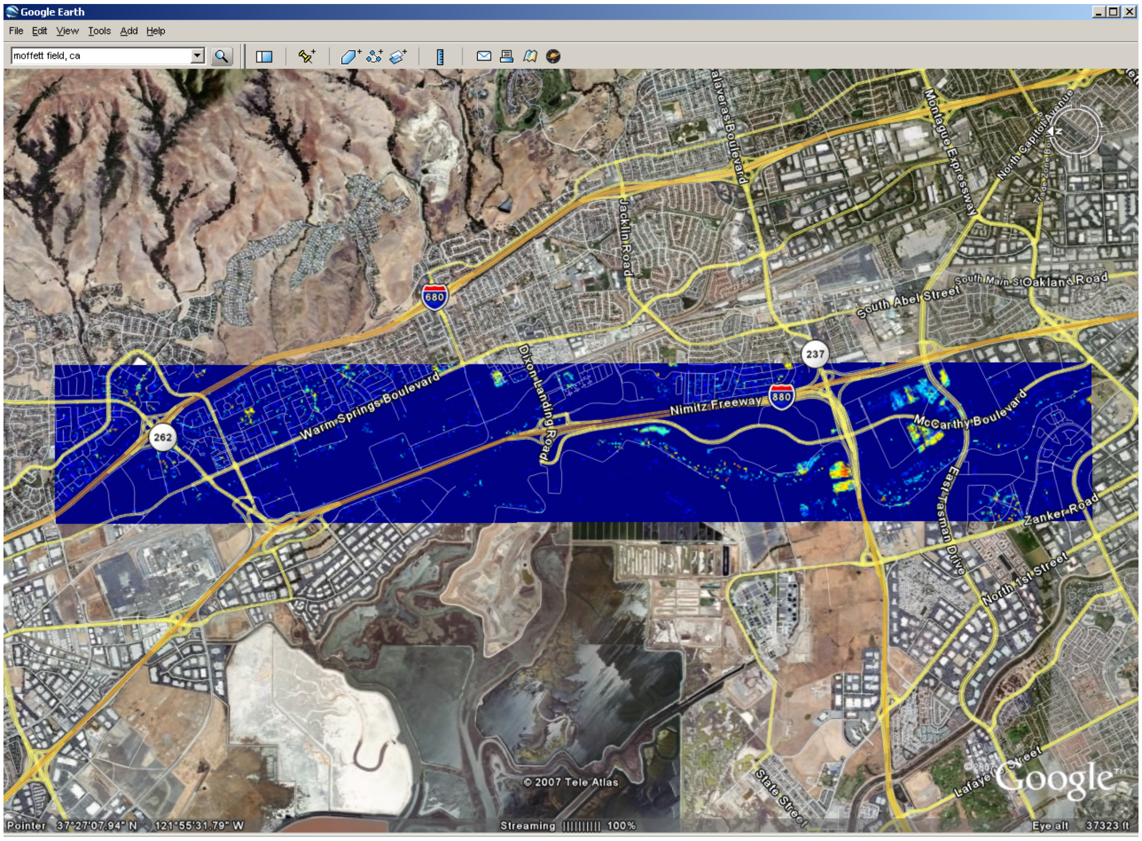


Figure 16: Abundance map of Oak Tree over Moffett Field, CA overlayed onto Google Earth.

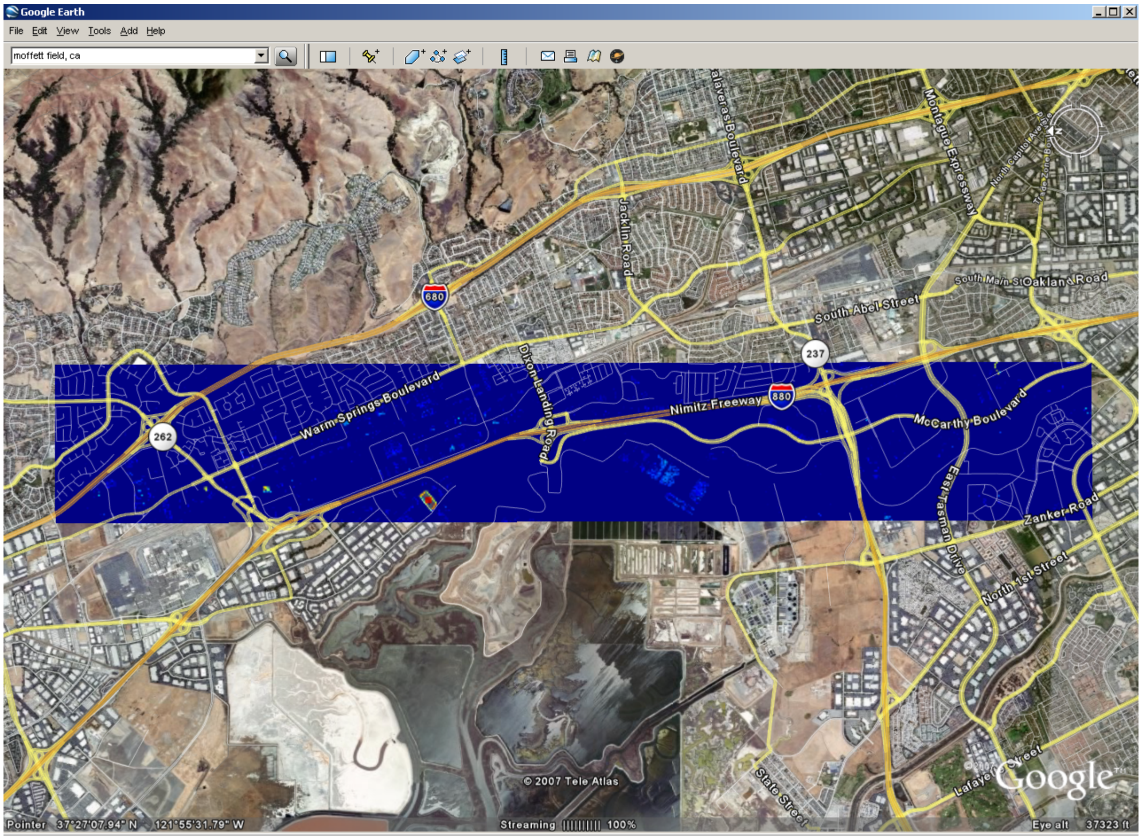


Figure 17: Abundance map of Antigorite over Moffett Field, CA overlayed onto Google Earth.

2. The varying of FastICA parameters and their effects on efficacy and computation performance in ICA-EEA.
3. The use of ICA algorithms other than FastICA and their effects on ICA-EEA.
4. New algorithms or algorithm modifications to provide consistent results in ICA-EEA and VCA. This includes initialization algorithms for ICA-EEA.
5. Formal evaluations of algorithms providing solutions to the constrained least squares problem.
6. Field work to investigate spectral mixing.

Finally, there is a need to study the ability of unmixing algorithms to uniquely distinguish similar endmembers. At what point do these algorithms fail to distinguish between similar endmembers in the presence of noise both white and colored. This study is important if we wish to determine how detection power can be improved and what its limitations are with current hyperspectral sensors.

A AVIRIS SNR Versus Wavelength

The original SNR data for AVIRIS from [17] is listed here for reference. The table represents the SNR of the AVIRIS sensor as a function of wavelength. Measured 2005.

Table 44: AVIRIS SNR Vs. Wavelength . Measured 2005.

Wavelength [nm]	**SNR [linear]**	**SNR [dB]**
3.66094183E+02	1.71409960E+02	4.46807211E+01
3.75715176E+02	1.93949230E+02	4.57537612E+01
3.85337344E+02	2.12057117E+02	4.65290570E+01
3.94960687E+02	4.16684604E+02	5.23961491E+01
4.04585206E+02	5.66017792E+02	5.50566017E+01
4.14210899E+02	6.19763116E+02	5.58445145E+01
4.23837768E+02	7.19810972E+02	5.71443692E+01
4.33465812E+02	9.58843053E+02	5.96349505E+01
4.43095032E+02	1.19086765E+03	6.15172700E+01
4.52725426E+02	1.30349645E+03	6.23021971E+01
4.62356996E+02	1.45963133E+03	6.32848635E+01
4.71989741E+02	1.45199211E+03	6.32392852E+01
4.81623661E+02	1.48146713E+03	6.34138404E+01
4.91258756E+02	1.46070945E+03	6.32912768E+01
5.00895027E+02	1.46118996E+03	6.32941336E+01
5.10532472E+02	1.40101378E+03	6.29288482E+01
5.20171093E+02	1.43730962E+03	6.31510066E+01
5.29810889E+02	1.31324048E+03	6.23668852E+01
5.39451861E+02	1.34902195E+03	6.26003803E+01
5.49094007E+02	1.35383331E+03	6.26313039E+01
5.58737329E+02	1.36418522E+03	6.26974668E+01
5.68381826E+02	1.34417252E+03	6.25691002E+01
5.78027498E+02	1.26252967E+03	6.20248318E+01
5.87674345E+02	1.24279472E+03	6.18879880E+01
5.97322368E+02	1.37842601E+03	6.27876692E+01
6.06971565E+02	1.29503684E+03	6.22456425E+01
6.16621938E+02	1.24216655E+03	6.18835966E+01
6.26273486E+02	1.24935040E+03	6.19336852E+01
6.35926210E+02	1.22038660E+03	6.17299486E+01
6.45580108E+02	1.27207050E+03	6.20902236E+01
6.55235182E+02	1.23553001E+03	6.18370660E+01

Table 44 – Continued

Wavelength [nm]	**SNR [linear]**	**SNR [dB]**
6.64891431E+02	1.43582439E+03	6.31420265E+01
6.53248750E+02	1.22215763E+03	6.17425445E+01
6.62809990E+02	1.27969126E+03	6.21421041E+01
6.72372750E+02	1.32155529E+03	6.24217067E+01
6.81937030E+02	1.32796932E+03	6.24637608E+01
6.91502830E+02	1.25978875E+03	6.20059545E+01
7.01070150E+02	1.30818096E+03	6.23333565E+01
7.10638990E+02	1.33844558E+03	6.25320143E+01
7.20209350E+02	1.08637476E+03	6.07195933E+01
7.29781230E+02	1.19926204E+03	6.15782817E+01
7.39354630E+02	1.48068741E+03	6.34092677E+01
7.48929550E+02	1.56084279E+03	6.38671832E+01
7.58505990E+02	1.05138568E+03	6.04352412E+01
7.68083950E+02	1.22483590E+03	6.17615582E+01
7.77663430E+02	1.51091522E+03	6.35848019E+01
7.87244430E+02	1.47408767E+03	6.33704663E+01
7.96826950E+02	1.41536054E+03	6.30173417E+01
8.06410990E+02	1.32831876E+03	6.24660461E+01
8.15996550E+02	1.04476901E+03	6.03804057E+01
8.25583630E+02	1.09621080E+03	6.07978815E+01
8.35172230E+02	1.22292397E+03	6.17479891E+01
8.44762350E+02	1.28138830E+03	6.21536151E+01
8.54353990E+02	1.22292082E+03	6.17479668E+01
8.63947150E+02	1.28772393E+03	6.21964554E+01
8.73541830E+02	1.31472007E+03	6.23766659E+01
8.83138030E+02	1.30081782E+03	6.22843295E+01
8.92735750E+02	1.11053809E+03	6.09106691E+01
9.02334990E+02	9.42283817E+02	5.94836347E+01
9.11935750E+02	8.82309221E+02	5.89124164E+01
9.21538030E+02	8.80827652E+02	5.88978188E+01
9.29038030E+02	3.10011534E+02	4.98275570E+01
9.45065635E+02	3.15350873E+02	4.99758807E+01
9.54537046E+02	4.56180276E+02	5.31827301E+01
9.64007861E+02	7.66714085E+02	5.76926688E+01
9.73478080E+02	9.48553148E+02	5.95412334E+01
9.82947703E+02	1.13390814E+03	6.10915575E+01
9.92416730E+02	1.25089814E+03	6.19444390E+01
1.00188516E+03	1.23472682E+03	6.18314176E+01
1.01135300E+03	1.22496850E+03	6.17624984E+01
1.02082024E+03	1.23123859E+03	6.18068444E+01

Table 44 – Continued

Wavelength [nm]	SNR [linear]	SNR [dB]
1.03028688E+03	1.22164297E+03	6.17388860E+01
1.03975293E+03	1.22020818E+03	6.17286787E+01
1.04921838E+03	1.29469933E+03	6.22433784E+01
1.05868323E+03	1.29084409E+03	6.22174758E+01
1.06814749E+03	1.33355511E+03	6.25002194E+01
1.07761115E+03	1.33000000E+03	6.24770328E+01
1.08707422E+03	1.33081224E+03	6.24823357E+01
1.09653669E+03	1.14767620E+03	6.11963875E+01
1.10599857E+03	9.05993118E+02	5.91424980E+01
1.11545985E+03	2.83397226E+02	4.90479119E+01
1.12492053E+03	1.02975589E+02	4.02546857E+01
1.13438062E+03	1.94348373E+02	4.57716182E+01
1.14384011E+03	2.21749402E+02	4.69172492E+01
1.15329900E+03	3.59757422E+02	5.11201952E+01
1.16275730E+03	7.55323123E+02	5.75626556E+01
1.17221500E+03	9.45753994E+02	5.95155637E+01
1.18167211E+03	9.11768637E+02	5.91976930E+01
1.19112862E+03	9.21179973E+02	5.92868897E+01
1.20058454E+03	8.81441217E+02	5.89038671E+01
1.21003986E+03	9.11280787E+02	5.91930443E+01
1.21949458E+03	9.73333027E+02	5.97652292E+01
1.22894871E+03	1.01686855E+03	6.01452963E+01
1.23840224E+03	1.04719751E+03	6.04005720E+01
1.24785517E+03	1.02824028E+03	6.02418923E+01
1.25730751E+03	1.03596543E+03	6.03069052E+01
1.25400065E+03	3.52438633E+02	5.09416702E+01
1.26398069E+03	3.33741712E+02	5.04682098E+01
1.27396047E+03	3.56814050E+02	5.10488389E+01
1.28394000E+03	4.06046979E+02	5.21715257E+01
1.29391927E+03	4.21582957E+02	5.24976609E+01
1.30389829E+03	3.91839015E+02	5.18621535E+01
1.31387705E+03	3.53165872E+02	5.09595746E+01
1.32385555E+03	3.03650846E+02	4.96474899E+01
1.33383380E+03	1.86772960E+02	4.54262800E+01
1.34381179E+03	9.01318980E+01	3.90975703E+01
1.35378953E+03	6.92154988E+00	1.68040671E+01
1.36376701E+03	2.74474953E+00	8.77005440E+00
1.37374423E+03	2.55073937E+00	8.13332171E+00
1.38372120E+03	2.47787777E+00	7.88159758E+00
1.39369791E+03	3.07318486E+00	9.75177370E+00

Table 44 – Continued

Wavelength [nm]	**SNR [linear]**	**SNR [dB]**
1.40367437E+03	2.57881306E+00	8.22839723E+00
1.41365057E+03	3.25905839E+00	1.02618428E+01
1.42362651E+03	8.45606491E+00	1.85433662E+01
1.43360220E+03	1.46499761E+01	2.33167383E+01
1.44357763E+03	2.66836443E+01	2.85249029E+01
1.45355281E+03	8.53536569E+01	3.86244427E+01
1.46352773E+03	1.07354592E+02	4.06164125E+01
1.47350239E+03	8.18323486E+01	3.82585003E+01
1.48347680E+03	1.58216119E+02	4.39850146E+01
1.49345095E+03	3.32147466E+02	5.04266189E+01
1.50342485E+03	4.91103167E+02	5.38234547E+01
1.51339849E+03	5.78755480E+02	5.52499023E+01
1.52337187E+03	6.39958448E+02	5.61230355E+01
1.53334500E+03	6.81427543E+02	5.66683937E+01
1.54331787E+03	7.09647291E+02	5.70208510E+01
1.55329049E+03	7.39560450E+02	5.73794736E+01
1.56326285E+03	7.57627052E+02	5.75891095E+01
1.57323495E+03	6.99617369E+02	5.68972117E+01
1.58320680E+03	7.15196690E+02	5.70885099E+01
1.59317839E+03	7.31231594E+02	5.72810990E+01
1.60314973E+03	6.76314977E+02	5.66029801E+01
1.61312081E+03	6.58043478E+02	5.63650918E+01
1.62309163E+03	7.65111030E+02	5.76744893E+01
1.63306220E+03	8.01579583E+02	5.80789329E+01
1.64303251E+03	7.58368564E+02	5.75976064E+01
1.65300257E+03	7.56483671E+02	5.75759912E+01
1.66297237E+03	8.05498162E+02	5.81212911E+01
1.67294191E+03	8.73059480E+02	5.88208766E+01
1.68291120E+03	8.72858649E+02	5.88188784E+01
1.69288023E+03	8.19318308E+02	5.82690532E+01
1.70284901E+03	7.25666603E+02	5.72147427E+01
1.71281753E+03	6.90370300E+02	5.67816420E+01
1.72278579E+03	7.52504513E+02	5.75301822E+01
1.73275380E+03	6.29641087E+02	5.59818612E+01
1.74272155E+03	5.29767245E+02	5.44817021E+01
1.75268905E+03	5.55629415E+02	5.48957046E+01
1.76265629E+03	4.81421838E+02	5.36505157E+01
1.77262327E+03	3.70282207E+02	5.13706569E+01
1.78259000E+03	2.29641644E+02	4.72210129E+01
1.79255647E+03	1.12367859E+02	4.10128421E+01

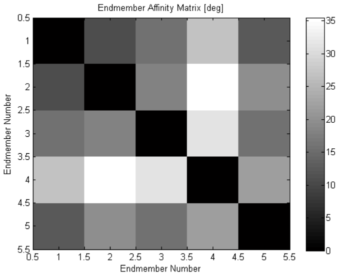
Table 44 – Continued

Wavelength [nm]	**SNR [linear]**	**SNR [dB]**
1.80252269E+03	2.36009960E+01	2.74586066E+01
1.81248865E+03	2.51989647E+00	8.02765396E+00
1.82245435E+03	1.00447693E+00	3.87993394E-02
1.83241980E+03	1.03064798E+00	2.62207106E-01
1.84238499E+03	9.01369974E-01	-9.01938265E-01
1.85234993E+03	9.02022314E-01	-8.95654374E-01
1.86231461E+03	7.64712093E-01	-2.33004084E+00
1.87227903E+03	9.00263226E-01	-9.12609788E-01
1.88224320E+03	6.11002968E-01	-4.27913361E+00
1.87923220E+03	1.67421124E+00	4.47620506E+00
1.88929280E+03	1.73369509E+00	4.77945439E+00
1.89935180E+03	1.47839901E+00	3.39583329E+00
1.90940920E+03	1.58019236E+00	3.97419915E+00
1.91946500E+03	1.54452530E+00	3.77590051E+00
1.92951920E+03	1.57944251E+00	3.97007647E+00
1.93957180E+03	4.89274539E+00	1.37910523E+01
1.94962280E+03	2.02349680E+01	2.61220504E+01
1.95967220E+03	6.21483228E+01	3.58685883E+01
1.96972000E+03	2.67708766E+02	4.85532518E+01
1.97976620E+03	4.78191465E+02	5.35920364E+01
1.98981080E+03	4.62792920E+02	5.33077341E+01
1.99985380E+03	1.89655899E+02	4.55593271E+01
2.00989520E+03	1.22509697E+02	4.17634093E+01
2.01993500E+03	2.28847211E+02	4.71909125E+01
2.02997320E+03	5.07340590E+02	5.41059922E+01
2.04000980E+03	5.61452610E+02	5.49862621E+01
2.05004480E+03	3.85349876E+02	5.17171045E+01
2.06007820E+03	3.36999712E+02	5.05525906E+01
2.07011000E+03	3.74738557E+02	5.14745676E+01
2.08014020E+03	5.20552405E+02	5.43292891E+01
2.09016880E+03	5.50233199E+02	5.48109358E+01
2.10019580E+03	5.78134306E+02	5.52405748E+01
2.11022120E+03	6.21615563E+02	5.58704376E+01
2.12024500E+03	6.01407743E+02	5.55833803E+01
2.13026720E+03	6.22449219E+02	5.58820785E+01
2.14028780E+03	6.27187956E+02	5.59479542E+01
2.15030680E+03	5.81111426E+02	5.52851883E+01
2.16032420E+03	5.54856647E+02	5.48836159E+01
2.17034000E+03	5.54611972E+02	5.48797848E+01
2.18035420E+03	5.40123334E+02	5.46498588E+01

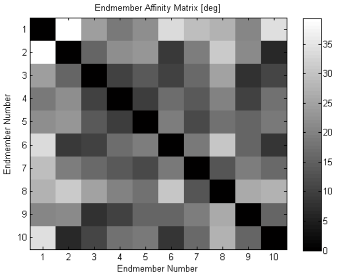
Table 44 – Continued

Wavelength [nm]	**SNR [linear]**	**SNR [dB]**
2.19036680E+03	5.42665956E+02	5.46906515E+01
2.20037780E+03	5.08875795E+02	5.41322359E+01
2.21038720E+03	5.81866238E+02	5.52964632E+01
2.22039500E+03	5.76006532E+02	5.52085482E+01
2.23040120E+03	5.73608193E+02	5.51723069E+01
2.24040580E+03	5.54500136E+02	5.48780331E+01
2.25040880E+03	5.56042790E+02	5.49021643E+01
2.26041020E+03	5.41539289E+02	5.46725994E+01
2.27041000E+03	5.39240021E+02	5.46356423E+01
2.28040820E+03	5.22409031E+02	5.43602135E+01
2.29040480E+03	5.17100070E+02	5.42714919E+01
2.30039980E+03	4.75581185E+02	5.35444933E+01
2.31039320E+03	4.78941938E+02	5.36056574E+01
2.32038500E+03	3.80809649E+02	5.16141589E+01
2.33037520E+03	4.12905071E+02	5.23170043E+01
2.34036380E+03	3.36707706E+02	5.05450611E+01
2.35035080E+03	2.83041459E+02	4.90370011E+01
2.36033620E+03	3.14285152E+02	4.99464773E+01
2.37032000E+03	2.35979966E+02	4.74575027E+01
2.38030220E+03	1.99795868E+02	4.60117300E+01
2.39028280E+03	1.93373994E+02	4.57279613E+01
2.40026180E+03	2.09168287E+02	4.64099168E+01
2.41023920E+03	1.36226831E+02	4.26852531E+01
2.42021500E+03	1.01466157E+02	4.01264242E+01
2.43018920E+03	1.50774481E+02	4.35665568E+01
2.44016180E+03	1.17187629E+02	4.13776353E+01
2.45013280E+03	4.64186155E+01	3.33338437E+01
2.46010220E+03	8.47493741E+01	3.85627300E+01
2.47007000E+03	4.07071833E+01	3.21934211E+01
2.48003620E+03	1.35927714E+01	2.26661602E+01
2.49000080E+03	1.44823893E+00	3.21680434E+00
2.49996380E+03	1.75295314E+00	4.87540613E+00
2.50992520E+03	2.71175883E+00	8.66502128E+00

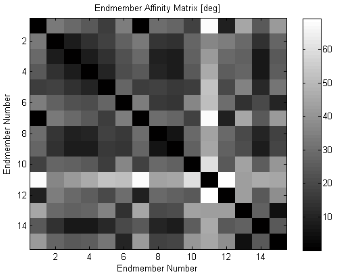
B Affinity Matrices of Endmembers for SMDE Dataset



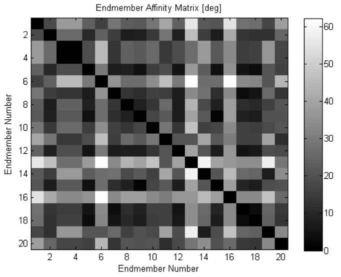
5 endmembers



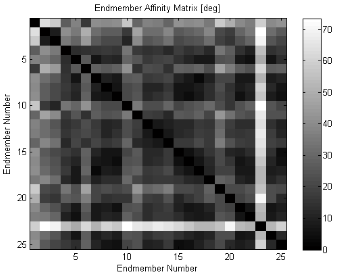
10 endmembers



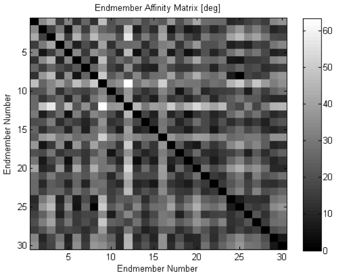
15 endmembers



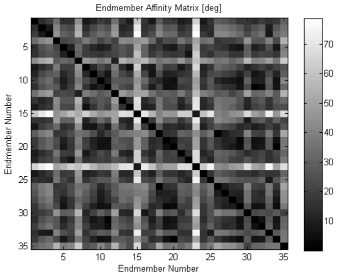
20 endmembers



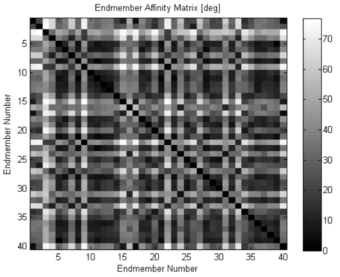
25 endmembers



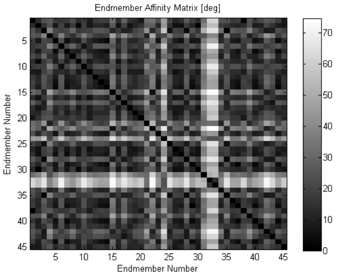
30 endmembers



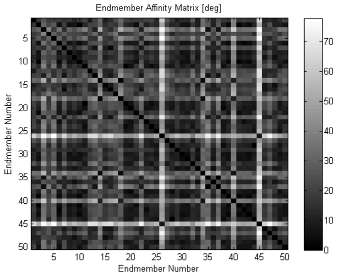
35 endmembers



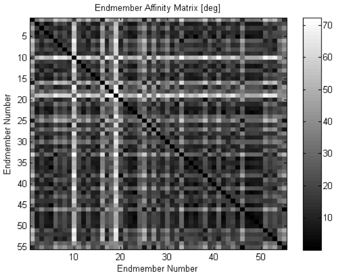
40 endmembers



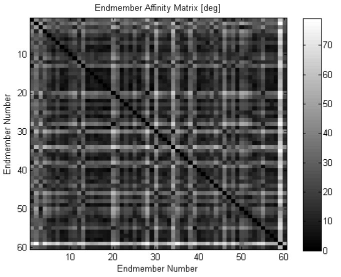
45 endmembers



50 endmembers

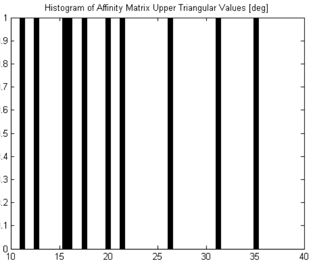


55 endmembers

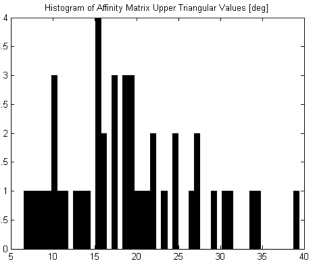


60 endmembers

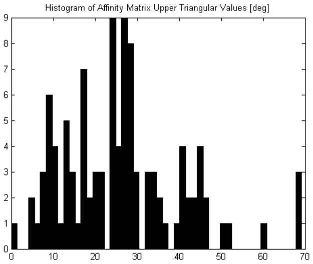
C Histograms of Endmember Affinity for SMDE Dataset



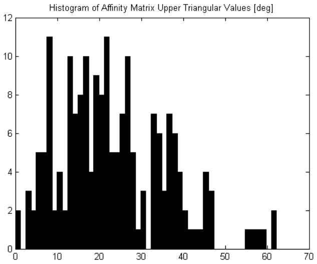
5 endmembers



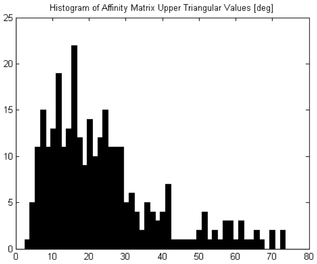
10 endmembers



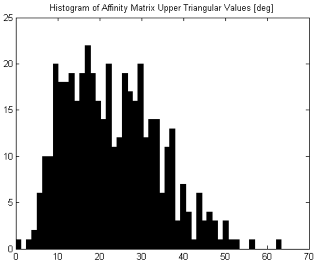
15 endmembers



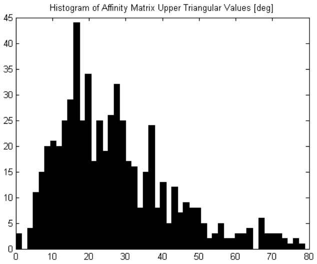
20 endmembers



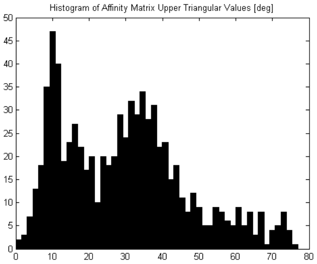
25 endmembers



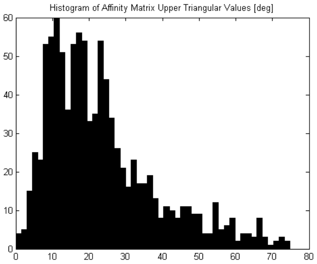
30 endmembers



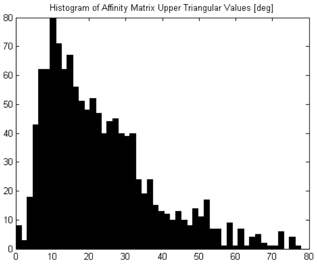
35 endmembers



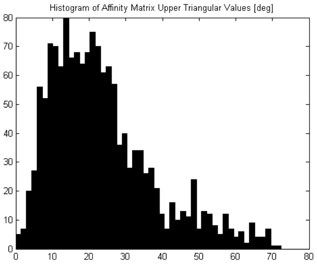
40 endmembers



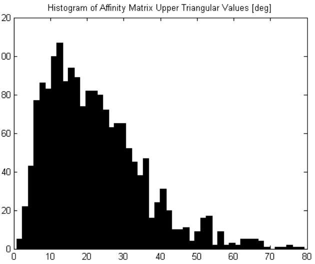
45 endmembers



50 endmembers

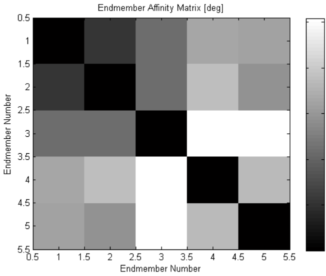


55 endmembers

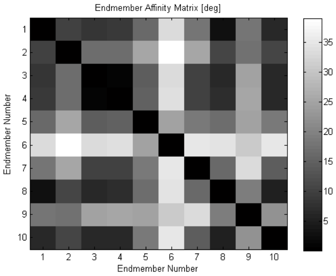


60 endmembers

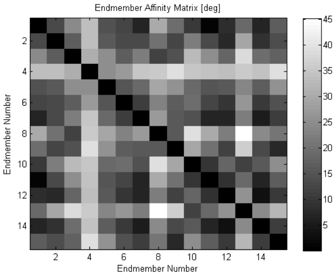
D Affinity Matrices of Endmembers for RMDE Dataset



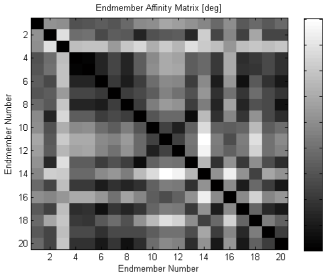
5 endmembers



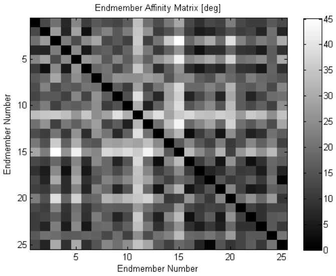
10 endmembers



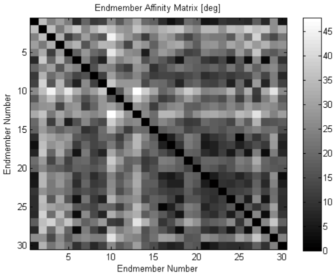
15 endmembers



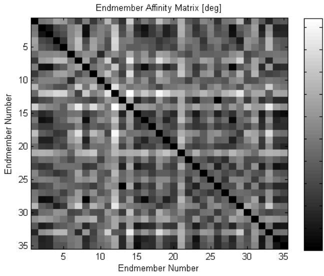
20 endmembers



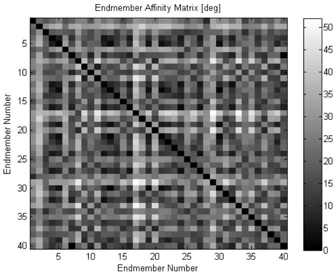
25 endmembers



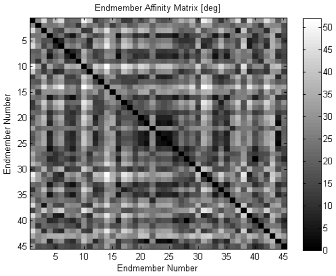
30 endmembers



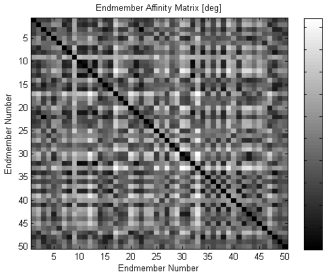
35 endmembers



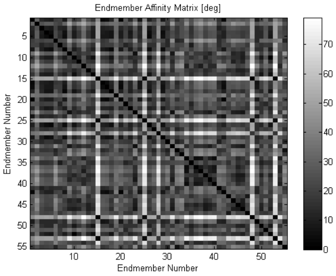
40 endmembers



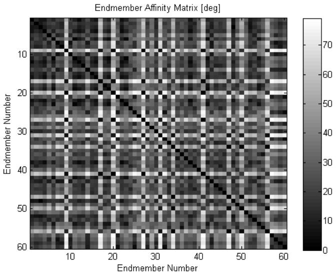
45 endmembers



50 endmembers

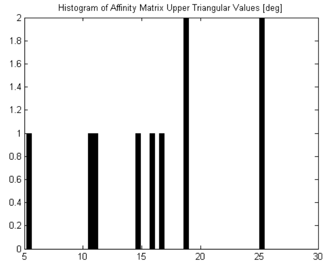


55 endmembers

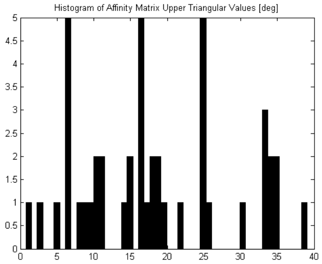


60 endmembers

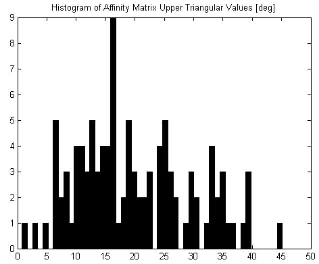
E Histograms of Endmember Affinity for RMDE Dataset



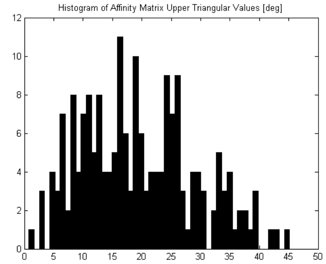
5 endmembers



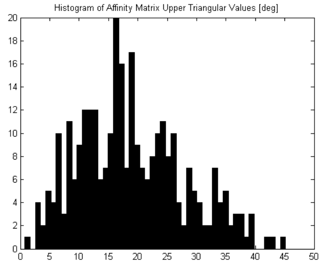
10 endmembers



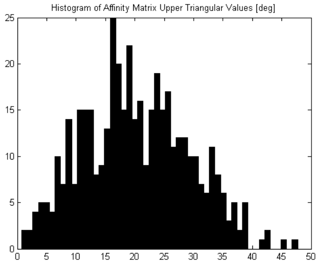
15 endmembers



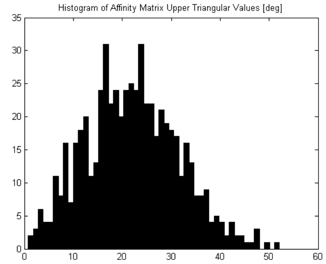
20 endmembers



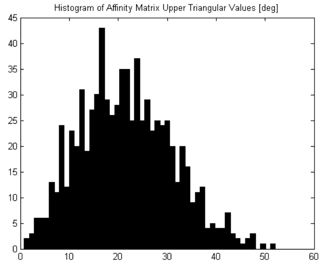
25 endmembers



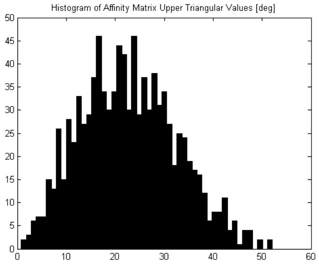
30 endmembers



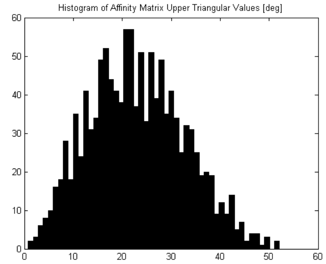
35 endmembers



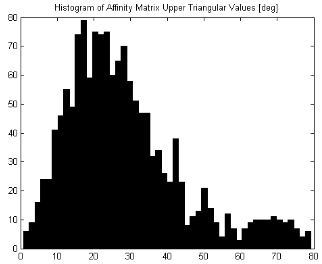
40 endmembers



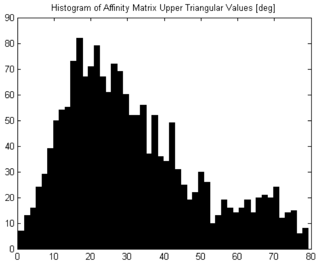
45 endmembers



50 endmembers

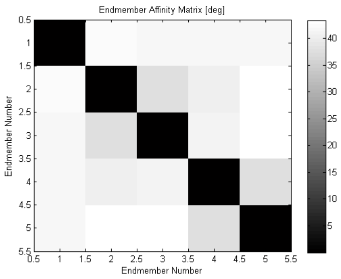


55 endmembers

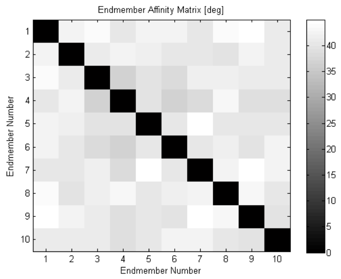


60 endmembers

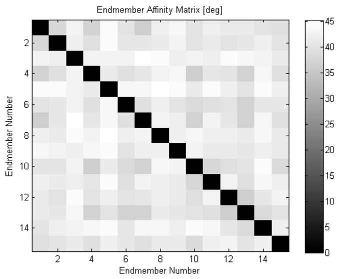
F Affinity Matrices of Endmembers for RMRE Dataset



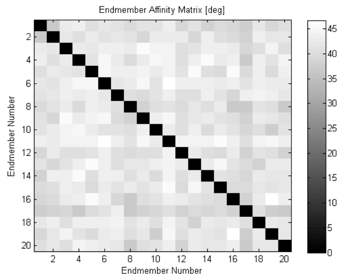
5 endmembers



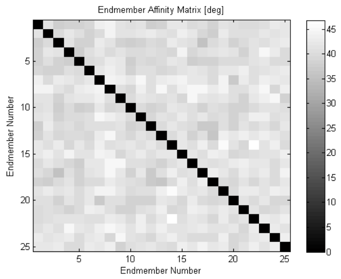
10 endmembers



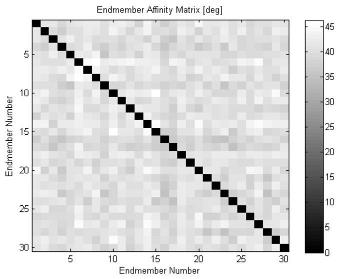
15 endmembers



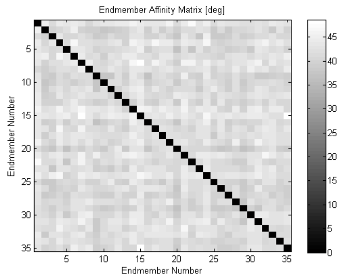
20 endmembers



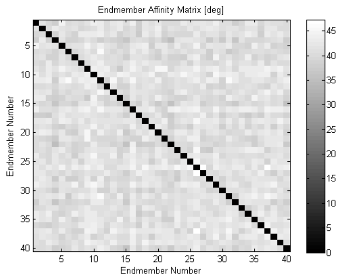
25 endmembers



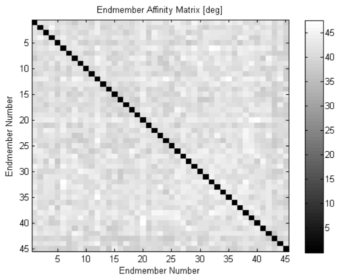
30 endmembers



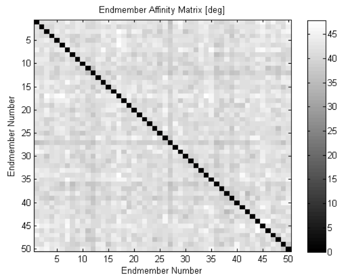
35 endmembers



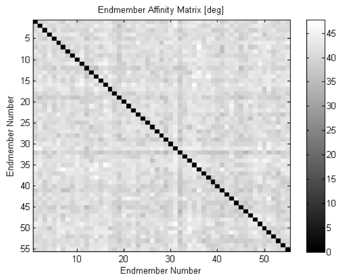
40 endmembers



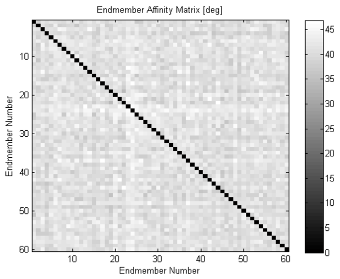
45 endmembers



50 endmembers

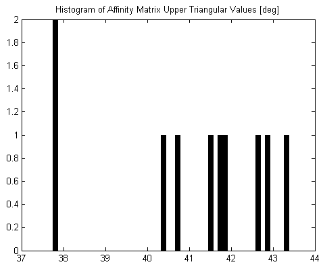


55 endmembers

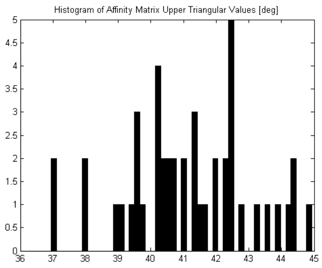


60 endmembers

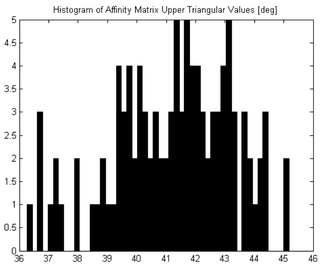
G Histograms of Endmember Affinity for RMRE Dataset



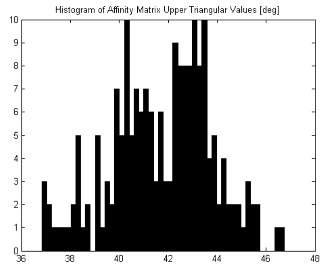
5 endmembers



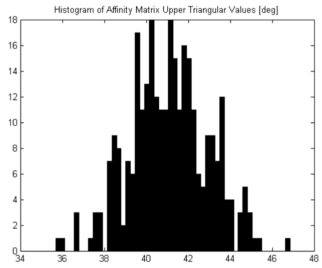
10 endmembers



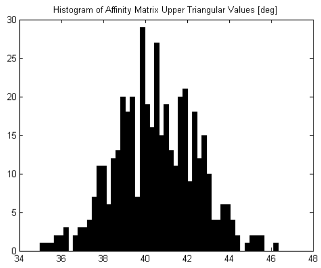
15 endmembers



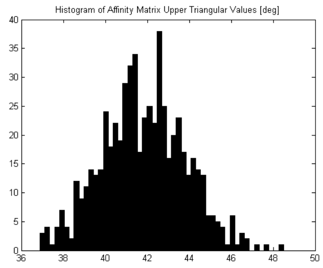
20 endmembers



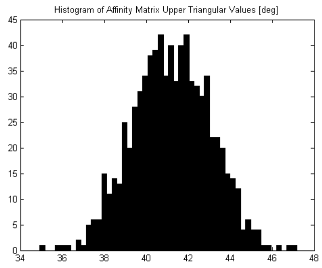
25 endmembers



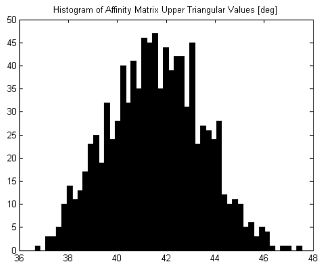
30 endmembers



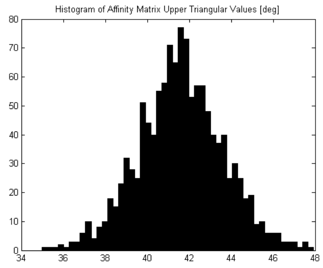
35 endmembers



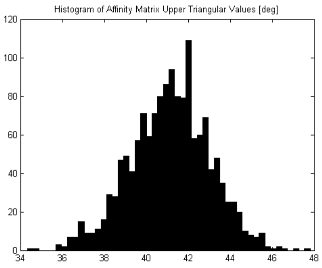
40 endmembers



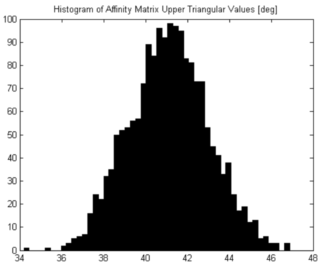
45 endmembers



50 endmembers

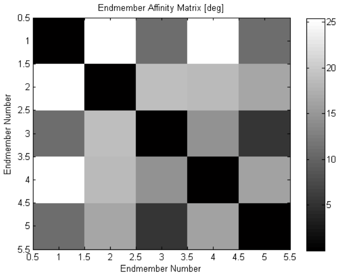


55 endmembers

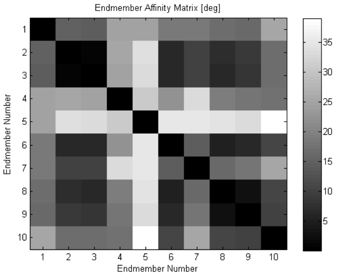


60 endmembers

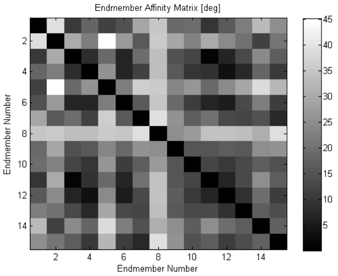
H Affinity Matrices of Endmembers for RMDEI Dataset



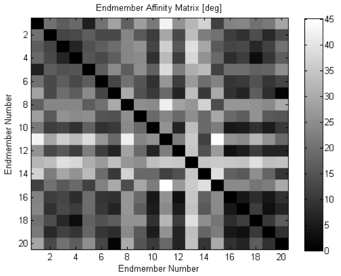
5 endmembers



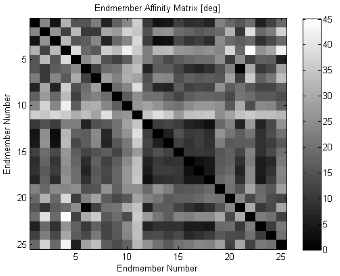
10 endmembers



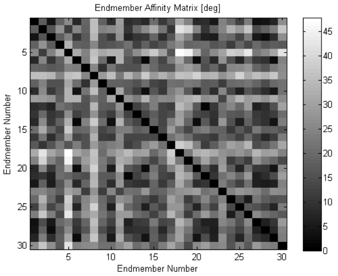
15 endmembers



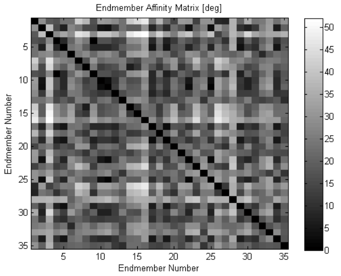
20 endmembers



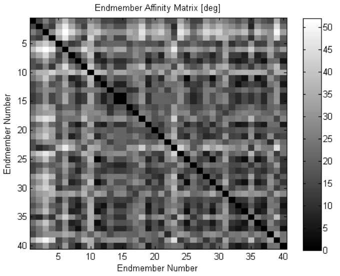
25 endmembers



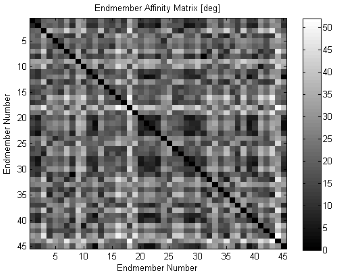
30 endmembers



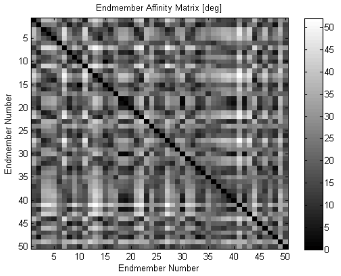
35 endmembers



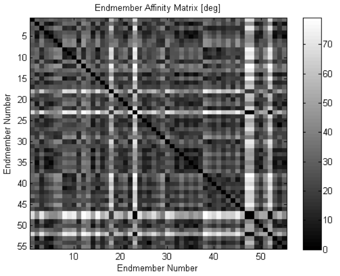
40 endmembers



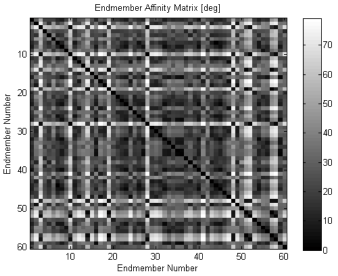
45 endmembers



50 endmembers

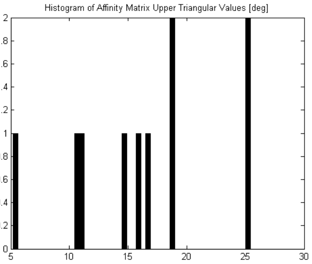


55 endmembers

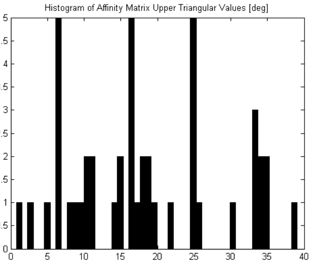


60 endmembers

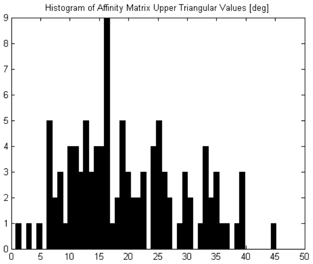
I Histograms of Endmember Affinity for RMDEI Dataset



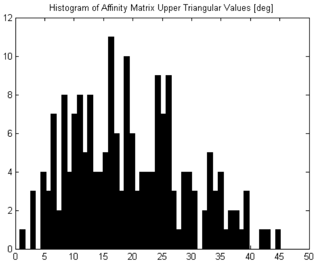
5 endmembers



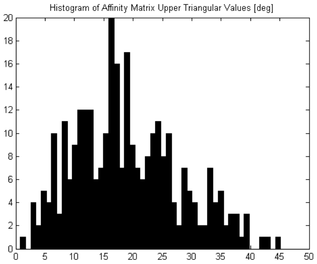
10 endmembers



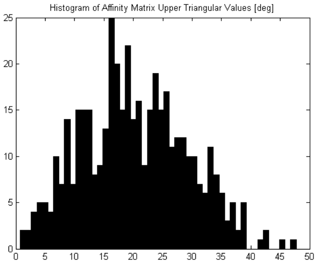
15 endmembers



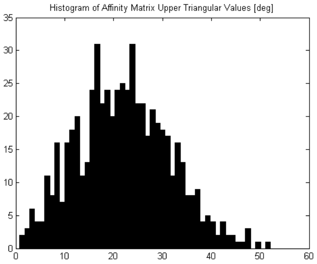
20 endmembers



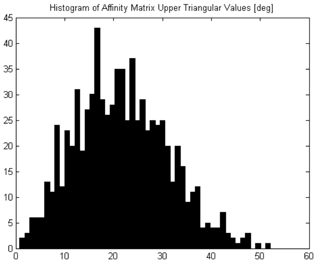
25 endmembers



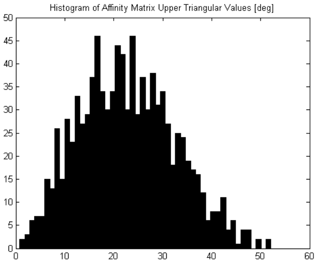
30 endmembers



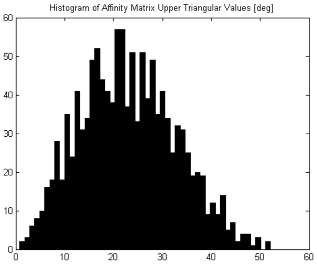
35 endmembers



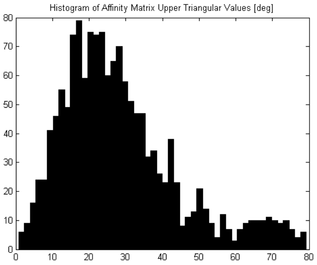
40 endmembers



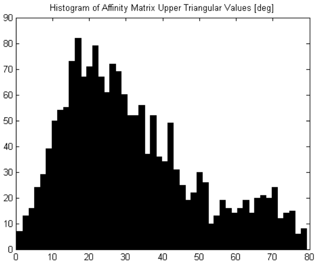
45 endmembers



50 endmembers



55 endmembers



60 endmembers

References

[1] R. A. Schowengerdt, *Remote Sensing: Models and Methods for Image Processing*, 2nd ed. Academic Press, 1997.

[2] NASA. (2007, jan) Aviris standard data product of moffett field, ca usa. NASA Jet Propulsion Laboratory. [Online]. Available: ftp://137.78.180.72/pub/outgoing/stddata/f970620t01p02_r03.rfl.tar.gz

[3] C.-I. Chang and Q. Du, "Estimation of number of spectrally distinct signal sources in hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, mar 2004.

[4] J. Wang and C.-I. Chang, "Applications of independent component analysis in endmember extraction and abundance quantification for hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 9, pp. 2601–1616, sep 2006.

[5] J. Harsanyi, W. Farrand, and C.-I. Chang, "Detection of subpixel spectral signatures in hyperspectral image sequences," in *Annual Meeting, Proceedings of the American Society of Photogrammetry and Remote Sensing*, 1994, pp. 236–247.

[6] M. H. Hayes, *Stastical Digital Signal Processing and Modeling.* John Wiley & Sons, Inc., 2005, p. 66.

[7] H. Ren and C.-I. Chang, "Automatic spectral target recognition in hyperspectral imagery," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1232–1249, oct 2003.

[8] E. Bingham and A. Hyvärinen, "A fast fixed point algorithm for independent component analysis of complex valued signals," *International Journal of Neural Systems*, vol. 10, no. 1, pp. 1–8, feb 2000.

[9] D. Luenberger, *Optimization by Vector Space Methods.* Wiley, 1969.

[10] P. Tuner, *Guide to Scientific Computing*, 2nd ed. Boca Raton, FL 33431: CRC Press, 2001.

[11] J. M. P. Nascimento and J. M. B. Dias, "Vertex component analysis: A fast algorithm to unmix hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 4, apr 2005.

[12] G. H. Golub and C. F. VanLoan, *Matrix Computations*, 4th ed. John Hopkins University Press, 1985.

[13] C.-I. Chang and A. Plaza, "A fast iterative algorithm for implementation of pixel purity index," *IEEE Transactions on Geoscience and Remote Sensing Letters*, vol. 3, no. 1, pp. 63–67, jan 2006.

[14] D. Heinz, C.-I. Chang, and M. L. G. Althouse, "Fully constrained least-squares based linear unmixing," in *IEEE International Geoscience and Remote Sensing Symposium*, vol. 2, 1999, pp. 1401–1403.

[15] C.-I. Chang. (2007, may) Re: Possible error in "fully constrained least-squares bases linear unmixing". Email response from Chein-I Chang. Remote Sensing Signal and Image Processing Laboratory, Univeristy of Maryland, Baltimore County.

[16] G. A. Swayze, R. N. Clark, A. F. Goetz, T. G. Chrien, and N. S. Gorelick, "Effects of spectrometer band pass, sampling, and signal-to-noise ratio on spectral identification using the tetracorder algorithm," *Journal of Geophysical Research (Planets)*, vol. 108, no. E9, 5105, 2003.

[17] R. O. Green. (2007, apr) 2005 aviris snr table. Email from Jet Propulsion Laboratory (Michael Eastwood). NASA Jet Propulsion Laboratory. Microsoft Excel spreadsheet.

[18] Clark *et al.* (2007, jan) Usgs digital spectral library splib05a. United States Geological Survey. [Online]. Available: ftp://ftpext.cr.usgs.gov/pub/cr/co/denver/speclab/pub/spectral.library/splib05.library/ASCII/splib05a.asciifiles.tar

[19] J. Wang and C.-I. Chang, "Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis," *IEEE Transactions On Geoscience and Remote Sensing*, vol. 44, no. 6, jun 2006.

[20] A. Plaza and C.-I. Chang, "Impact of initialization on design of endmember extraction algorithms," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 11, pp. 3397–3407, nov 2006.

[21] J. M. P. Nascimento and J. M. B. Dias, "Does independent component analysis play a role in unmixing hyperspectral data?" *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 1, pp. 175–187, jan 2005.