10-29-2010

Using n-dimensional volumes for mathematical applications in spectral image analysis

Amanda Ziemann

SCHOOL OF MATHEMATICAL SCIENCES
ROCHESTER INSTITUTE OF TECHNOLOGY
ROCHESTER, NEW YORK

CERTIFICATE OF APPROVAL

M.S. DEGREE THESIS

The M.S. Degree Thesis of Amanda K. Ziemann
has been examined and approved by the
thesis committee as satisfactory for the
thesis required for the M.S. degree
in Applied and Computational Mathematics

Dr. William Basener, Thesis Advisor

Dr. David Messinger

Dr. David Ross

Dr. Hossein Shahmohamad, Grad. Program Director

Date: October 29, 2010

Using n-Dimensional Volumes for Mathematical Applications in Spectral Image Analysis

by

Amanda K. Ziemann

A thesis submitted in partial fulfillment of the
requirements for the degree of Master of Science
in Applied and Computational Mathematics
from the School of Mathematical Sciences
Rochester Institute of Technology

October 29, 2010

Abstract

The ability to detect an object or activity – such as a military vehicle, construction area, campsite, or vehicle tracks – is highly important to both military and civilian applications. Sensors that process multi and hyperspectral images provide a medium for performing such tasks. Hyperspectral imaging is a technique for collecting and processing imagery at a large number of visible and non-visible wavelengths. Different materials exhibit different trends in their spectra, which can be used to analyze the image. For an image collected at n different wavelengths, the spectrum of each pixel can be mathematically represented as an n-element vector. The algorithm established in this work, the Simplex Volume Estimation algorithm (SVE), focuses specifically on change detection and large area search.

In hyperspectral image analysis, a set of pixels constitutes a data cloud, with each pixel corresponding to a vector endpoint in Euclidean space. The SVE algorithm takes a geometrical approach to image analysis based on the linear mixture model, which describes each pixel in an image collected at n spectral bands as a linear combination of n+1 pure-material component spectra (known as endmembers). Iterative endmember identification is used to construct a volume function, where the Gram matrix is used to calculate the hypervolume of the data at each iteration as the endmembers are considered in Euclidean spaces of increasing dimensionality. Linear algebraic theory substantiates that the volume function accurately characterizes the inherent dimensionality of a set of data, and supports that the volume function provides a tool for identifying the subspace in which the magnitude of the spread of the data is the greatest. A metric is extracted from the volume function, and is used to quantify the relative complexity within a single image or the change in complexity across multiple images. The SVE algorithm was applied to hyperspectral images for the tasks of change detection and large area search, and the results from these applications will demonstrate the feasibility of this method as a cueing tool for analysts.

Acknowledgements

First and foremost, I offer my sincerest appreciation to my advisors Dr. William Basener and Dr. David Messinger. It is difficult to overstate my gratitude to them, as I attribute the work of my Master's thesis to their patience, encouragement, and effort. Without them, this research would not have been possible. Their enthusiasm for the research served as an inspiration, and allowed me to comfortably and seemlessly bridge the gap from Applied Mathematics to Imaging Science. An entirely new set of opportunities were opened up to me due to their guidance, and for that I am forever grateful.

Thank you to Dr. David Ross for serving on my committee, and for his endless support. His questions always kept me on my toes, and pushed me to strive for mathematical robustness and clarity. A very special thanks goes to Dr. Ariel Schlamm, a graduate-student-turned-PhD in Imaging Science during the time in which this thesis was completed. I would have been lost without her kindness and support.

This thesis was done while I was in the School of Mathematical Sciences (SMS), and the research was performed through a collaboration with the Center for Imaging Science (CIS). Thank you to SMS–my home department–for challenging me academically and fostering a true love and appreciation for the field of mathematics. An additional thank you to CIS for welcoming me with open arms, and affording me the chance to apply mathematics in new and exciting ways. Thank you to Dr. Messinger and the NGA University Research Initiative for providing funding for this project.

Last but not least, I would like to thank my family and friends. Their love and encouragement makes anything seem possible.

Dedicated to my mom and dad.

Contents

List of Figures

Chapter 1

Introduction

This thesis will present a geometrically-based methodology for performing spectral image analysis. In particular, mathematical theory based in linear algebra was used to estimate the n-dimensional volume approximated by a set of pixels when the inherent dimensionality of the data was not known. The concept of the Linear Mixture Model in spectral imagery was utilized along with this novel mathematical approach to develop a new method for performing two types of spectral analysis: (1) characterizing changes between spectral images that have processed the same scene, and (2) identifying areas of higher relative complexity within a spectral image.

1.1 Hyperspectral Imaging

The ability to detect an object or activity – such as a military vehicle, construction area, campsite, or vehicle tracks – is of great importance to both military and civilian applications.[1] Sensors that process hyperspectral (and multispectral) images provide a medium for performing such detection tasks. These sensors supply image data that characterizes the image spatially as well as spectrally, which allows for an image or scene to be analyzed on a materialistic level that is more than strictly visual. In particular, materials that may visually appear similar, such as a camouflaged car in the middle of a forest, will appear very distinct when examined spectrally. The spectrum of light reflecting off of each object, also known as its respective *spectral signature*, can differ greatly between two materials. The driving force behind hyperspectral imaging is that, for a given wavelength, the radiance (*i.e.*, the reflected, emitted or absorbed amount of radioactivity) varies with material.[1]

Instead of the standard RGB processing of digital images, hyperspectral imaging en-

tails measuring the radiance or reflectance of each pixel at a very large number of spectral wavelengths from the electromagnetic spectrum (shown in Figure 1.1). These images are

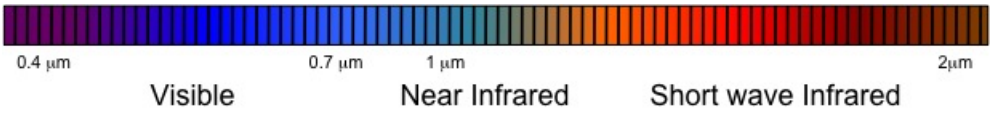


Figure 1.1: Portion of the electromagnetic spectrum, indicating the various regions of the spectrum from which wavelengths may be identified for use in measuring radiance or reflectance. The typical wavelength range is 0.4 μm – 2.0 μm.

captured by satellites or airplanes equipped with remote sensors, and depending on the time of day, the resulting wavelength values per pixel are some combination of the spectral signatures of the materials, sunlight, and atmospheric constituents.[1] In Figure 1.2, we

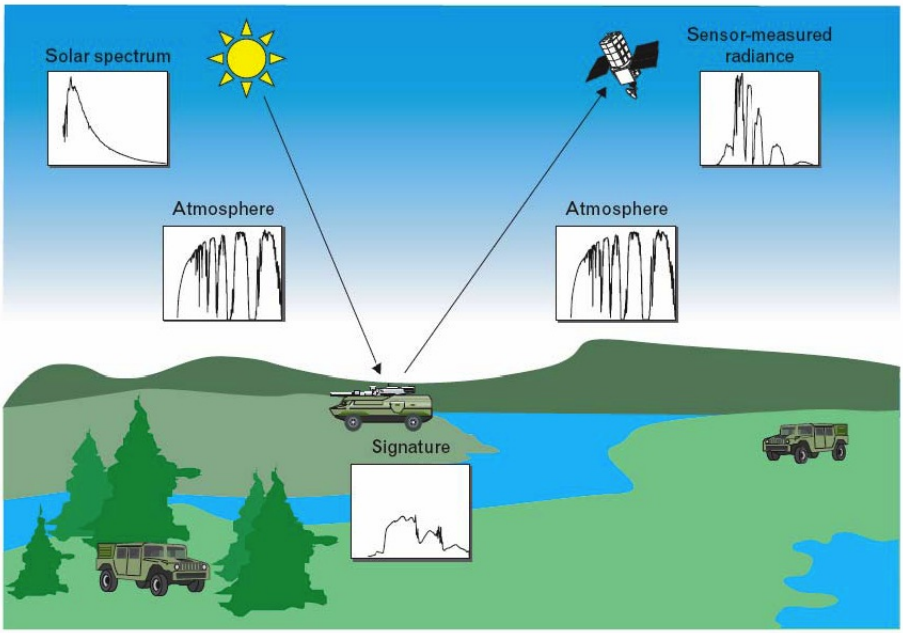


Figure 1.2: Pictoral representation of the pathway that solar radiation takes to the remote sensor. The sensor detects the energy reflected by surface materials and measures how this intensity varies in different parts of the electromagnetic spectrum.[1]

have a diagram illustrating the pathway of solar radiation in the collection of a hyperspectral image. The signal of interest here is the reflectance of the surface material, which is

gathered at each of the chosen wavelengths, and is given by

$$\text{reflectance spectrum } (\lambda) = \frac{\text{reflected radiation at band } (\lambda)}{\text{incident radiation at band } (\lambda)}. \tag{1.1}$$

This reflectance spectrum, or spectral signature, is the fraction of incident energy or radiation, usually sunlight, that is reflected by a surface material. It is given as a function of the wavelength (λ) of the energy.[1]

Before analyzing hyperspectral images, it is important to understand how the atmospheric presence affects the observed radiation of a surface material and its relationship with the associated reflectance spectra. If the spectrum of the solar energy was flat and the atmospheric presence was the same for all wavelengths, then the observed spectrum would be of the same shape as the reflectance spectrum. However, due to the presence of oxygen, water vapor, and other constituents in the atmosphere, radiation is selectively absorbed by the atmosphere at different wavelengths. This causes the observed radiance spectrum to be the solar spectrum altered by the transmitted radiance of the atmosphere as well as the reflectance spectrum of the surface material.[1] Subsequently, any attempt to characterize the spectral properties of a surface material through the atmosphere must first account for the scattering and absorption of the atmospheric transmittance, the effects of illumination, and the response of the sensor. Through advanced processing techniques, the spectral reflectance of a material is able to be recovered from its observed radiance as collected by the sensor. All images processed in this paper were done so with measurements in reflectance.

When an image is processed at many different wavelengths, it can be thought of as a *hyperspectral cube*, where the third dimension is incremented by the sampled bands. The materials in each pixel exhibit a certain reflectance at each band, resulting in a cube where every cross-section is the image when processed at a particular wavelength. In essence, it is a stack of two dimensional images of the scene captured at varying discrete, narrow spectral channels. Figure 1.3 demonstrates a hyperspectral cube, and provides the spectrum corresponding to a pixel of grass as an example.

As mentioned earlier, the reflectance spectrum of a material is its spectral signature, and is so named because it can be used to characterize different materials. As shown in Figure 1.3, the spectral signature can be plotted as the measure of reflectance (or radiance) across the sampled wavelengths. In Figure 1.4, we have several example spectra that illustrate how the signatures vary between different materials.

The observed spectral radiance data, or extracted spectral reflectance data, can be considered as a collection of points in the n-dimensional real Euclidean space $\mathbb{R}^n$, where n is the number of sampled spectral bands.[1] Every pixel can be thought of as an n-dimensional vector, where the i^{th} entry is the reflectance or radiance measure for that pixel

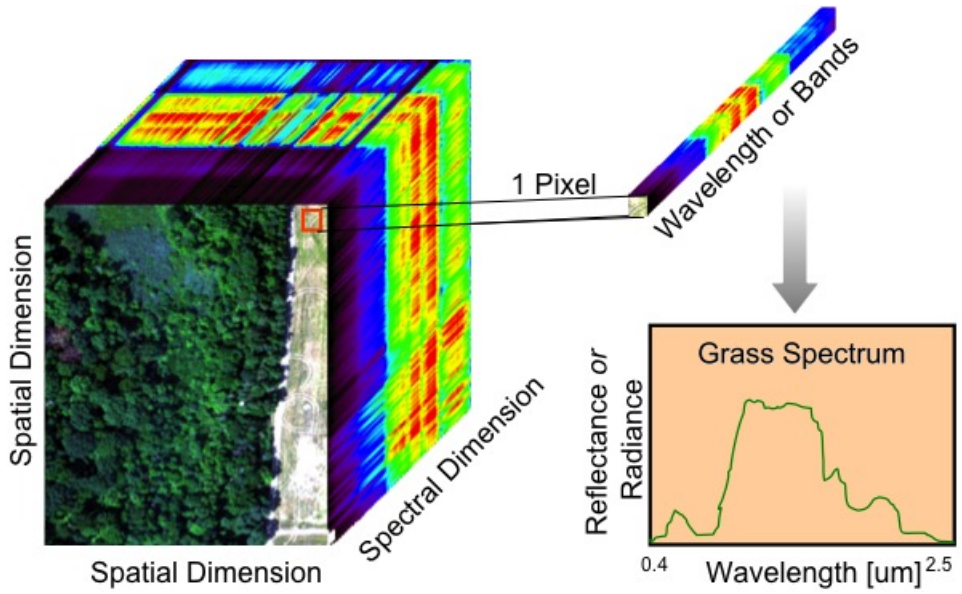


Figure 1.3: Hyperspectral images are often thought of as cubes. Each cross-section of the cube is the image when processed at a particular wavelength. When the reflectance values of a specific pixel at each wavelength are considered, it results in a spectrum vector for that pixel. A sample grass pixel and its corresponding spectrum plot is shown.

at the i^{th} sampled wavelength. In RGB digital images, the vectors in question are three-dimensional, with red wavelength, green wavelength, and blue wavelength components. When processed hyperspectrally, the dimensionality of the pixels (as vectors) is the number of sampled bands. In plotting these vectors, and in particular their endpoints, the image can be thought of as constituting a cloud of data.

1.2 Linear Mixture Model

If every material had a unique and fixed spectrum, then in theory we could differentiate between different materials by comparing their spectra.[1] In practice, however, this ideal situation does not hold. Due to variations in material surfaces, as well as the presence or absence of shadows, the observed spectra from sample image chips of the same material are never identical. Other variables, such as noise from the sensor, atmospheric conditions, and time of day, prevent homogenous materials from being characterized by a unique spectral signature.

The relationship between the spatial resolution of the sensor and the spatial variability

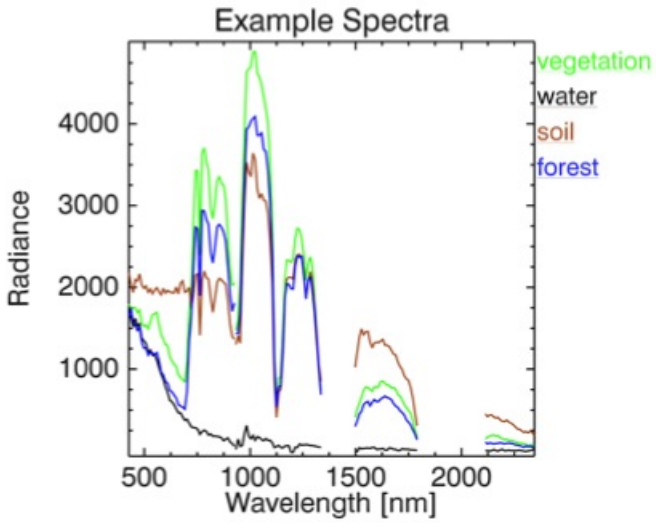


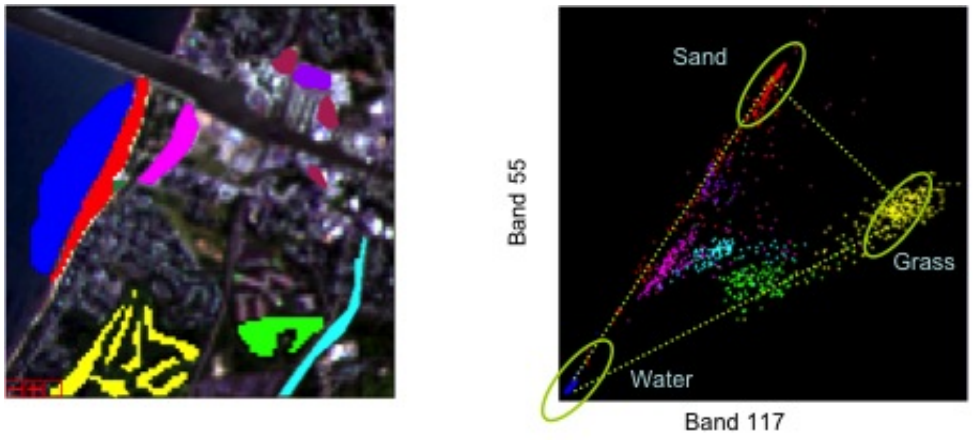
Figure 1.4: Example of material–specific spectral responses. Because the responses are measured in radiance, they include atmospheric effects due to water vapor, gas constituents, aerosols, etc. The data was collected by the NASA Hyperion hyperspectral sensor on board the EO-1 satellite.

in the ground scene provides another complication when attempting to characterize pixels by unique signatures. If a single pixel corresponds to an area on the ground that is occupied by multiple materials, then the sensor integrates the reflectance value of each of those materials into one reflectance value for the pixel. The result is a hyperspectral cube comprised of *pure pixels* and *mixed pixels*, where a pure pixel contains a single material and a mixed pixel contains multiple materials.[1] The presence of mixed-pixel interference, coupled with spectral variability within materials, provide the main obstacles for algorithms designed to exploit hyperspectral imagery.

There are several mathematical models that are designed to characterize the variability of pixels within a hyperspectral dataset. The three main categories are probability density models, subspace models, and linear spectral mixing models. The most widely used spectral mixing model is the Linear Mixture Model (LMM), which is the model employed here in the Simplex Volume Estimation algorithm.

The LMM operates under the assumption that there are materialistically pure pixels in the scene, and that every pixel can be described as some non-negative linear combination of these pure pixels. The spectra of the pure pixels are called *endmembers*, and their

vertices form the corners of a simplex that approximates the data cloud.[2] Figure 1.5 shows an example of the spectral separation of materials in an image for two spectral bands in the scene. The pure materials, *i.e.*, sand, water, and grass, comprise the corners of the simplex, and the mixed pixels exist inside of the simplex.



(a) (b)

Figure 1.5: (a) An AVIRIS RGB extracted from a hyperspectral image of the scene, where both "pure" and "mixed" pixel regions were selected by hand. (b) A scatter plot of the pixels projected into the plane of two spectral bands in the scene. As assumed by the LMM, the pure pixels in the scene occupy the corners of the simplex, and the mixed pixels lie inside of the simplex as a combination of the endmembers.

The mathematical equations and constraints defining the LMM for a set of pixels are

$$\begin{aligned} \boldsymbol{x} &= \sum_{k=1}^{M} a_k \boldsymbol{s}_k + \boldsymbol{w} = \boldsymbol{S}\boldsymbol{a} + \boldsymbol{w} \\ \sum_{k=1}^{M} a_k &= 1 \text{ (additivity constraint)} \\ a_k &\leq 0 \text{ (positivity constraint)} \end{aligned} \tag{1.2}$$

where $\boldsymbol{x}$ is a pixel in the scene, $\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_M$ are the M endmember spectra (assumed to be linearly independent), $a_1, a_2, \ldots, a_M$ are the corresponding abundances that weight the fraction of the material present in the pixel, and $\boldsymbol{w}$ is an additive-noise vector.[1] There are several ways to identify the endmembers in a scene, but here we employ the Max-D algorithm,[3] which determines the endmembers through a series of geometric projections.

When based in three or four dimensions, the LMM takes on a familiar geometric shape as a triangle or tetrahedron where the corners are the endpoints of the endmember vectors. When generalized to higher dimensions, the simplex in k dimensions is characterized by the $k+1$ endmembers (as endpoints) that define its corners.[1] It is important to note that while each pixel may be gathered across n different wavelengths, the inherent dimensionality of the datacloud is oftentimes far less than the number of bands that are processed. The linear mixture model has been successfully used to estimate the subpixel abundances of pure materials within mixed pixels for several applications in hyperspectral imaging.

1.3 Goals of HSI Analysis

Multi and hyperspectral image analysis has several main goals:

- large area search
- change detection
- target detection
- anomaly detection
- spectral classification

Here, we focus in particular on large area search and change detection.

1.3.1 Large Area Search

The goal of large area search is straightforward: to cue an analyst to regions of interest within an image for further analysis. However, actually conducting large area search of an image is a challenging task. It requires a considerably high probability of detection (*i.e.*, any missed detections render the search a failure), while accepting a moderate number of false alarms.[4] (False alarms are given by detected areas that are not actually regions of interest.) In general, the large area search problem does not seek to identify specified targets with a known spectral signature (as is the case in target detection).[5] Rather, it aims to identify "interesting" areas in an image when the signature for which one is searching is not necessarily well-defined.

The main area of interest here is in the application of large area search to the identification of localized regions of a multispectral or hyperspectral image that contain evidence of manmade materials. This is different from other search problems, such as finding and

counting all occurrences of a specific target. The assumption is that a large-area image has been collected over an area of interest, but with little to no knowledge about the actual content of that image. Colloquially, this is the problem of finding a needle in a haystack without actually knowing what the needle looks like.[4]

1.3.2 Change Detection

Change detection is the process of analyzing two registered images that were collected over the same scene at different periods in time, and then finding the significant materialistic changes between those images. The challenge comes in developing an algorithm that detects actual materialistic changes, and not just cosmetic changes within an unchanged material. That is to say, a slightly overgrown lawn in one image that is freshly cut in the next should not be flagged as a change, even though the spectral signature might vary slightly between the two. However, the absence of a building structure in one image and the presence of it in the other should certainly be flagged as an area of interest. An effective change detection algorithm is one that is able to distinguish between these "non-interesting" changes (*i.e.*, a mowed lawn) and "interesting" changes (*i.e.*, a new building), so as to cue the interesting changes to analysts.

1.4 Basic Statistical Methods

There are several very well-known statistical methods for HSI analysis, including the RX algorithm for anomaly detection and the matched filter algorithm for target detection. Before discussing the development behind the geometrically-based SVE algorithm for large area search and change detection, it is important to review these statistical methods as well as their potential problem areas. Anomaly detection algorithms aim to find pixels that are deemed "anomalous" when compared to the other pixels within the image, and target detection algorithms aim to identify within an image specific targets, or materials of interest, about which there is not necessarily any prior knowledge.

1.4.1 RX Algorithm

Developed by Reed and Yu, the RX algorithm first considers the pixels as a cloud of vectors. It then finds the mean of the data, call it $\boldsymbol{x}$, and characterizes any pixels that are greater than some distance δ from the mean $\boldsymbol{x}$ as anomalous[6]. A box is centered around each pixel, and the covariance of the data within each box is calculated. Every pixel is "ranked," where the value is given by the number of standard deviations that the pixel differs from the background. Higher-ranked pixels are the anomalies. This algorithm

has one major drawback: it assumes that the data is normally distributed. This is rarely the case with hyperspectral images, particularly with technological advancements yielding images with higher spatial resolution. It also requires some knowledge *a priori* in that the size of the box cannot easily be determined without knowing where the anomalies are located.

1.4.2 Matched Filter

The matched filter algorithm for target detection first considers a test pixel. It then compares the test pixel against a target spectrum to see how target-like it is, and against the image mean (or covariance) to see how background-like it is. Based on these comparisons, the pixel is then classified as either a target or background pixel. Similar to the RX algorithm, matched filter also operates under the assumption that the data maintain a Gaussian distribution. Again, this is rarely the case with hyperspectral images, which is a large drawback with matched filter.

1.5 Problem to be Addressed

The initial goal in the construction of this algorithm was very straightforward: develop a geometrically-based approach to estimating local complexity in hyperspectral images as well as to detecting changes between images. The driving force behind formulating the algorithm around the geometry of the data was that in statistically-based analysis methods, the necessary assumptions about the distribution of the data oftentimes do not hold when applied to images with increased spatial resolution. This is not to say that these statistical algorithms do not function well in analyzing spectral images, but rather that their window of application is limited to images of certain resolutions. With the ever-increasing amount of technological advances in the capturing of spectral images, the spatial resolution of these images is constantly improving. As such, it is helpful to have an algorithm that does not critically depend upon the resolution of an image. In formulating the algorithm presented here–the Simplex Volume Estimation algorithm– around the geometry of the data, this potential hindrance is avoided.

The complexity estimation in SVE is used to cue areas within an image that have higher materialistic complexity. This is extremely useful in large area search because SVE requires no prior knowledge about the content of an image. In fact, the Simplex Volume Estimation algorithm is designed specifically to analyze images about which, beforehand, we know nothing. All of the results presented in Section 8.1 are examples of large area search. The change detection procedure in SVE is used to cue areas of materialistic changes between two images that have processed the same scene at different points in time. The

idea is that in obtaining regional complexity estimates in an image, those complexity metric values can be compared between images to identify changed areas. Experimental results from testing sets of registered images are presented in Section 8.2.

Chapter 2

n-Dimensional Volume

The generalization of Euclidean geometry to higher dimensions is given by the Euclidean space, denoted $\mathbb{E}^n$ for n dimensions. This differs from $\mathbb{R}^n$ in that $\mathbb{E}^n$ does not have a natural coordinate system nor any distinct points, while $\mathbb{R}^n$ is a particular Euclidean space that has an intrinsic system of coordinates based in the real numbers as well as a distinguished point—the origin.[7] The mathematical properties of Euclidean geometry that are presented here are applied to the specific case of the real-number based system. In particular, because of the n-tuple representation of the elements in $\mathbb{R}^n$, the real Euclidean n-space can be considered as an n-dimensional vector space over the real field.

The idea of volume, or measure occupied by a solid body, is naturally understood in one, two, and three dimensions. In the same way that the real number coordinate system can be generalized to higher dimensions, the idea of volume can also be extended to multiple dimensions. While this is beyond the scope of a visual representation, it can be done with mathematical definitions. We start by establishing the axioms of n-dimensional volume, as these will lay the foundation for the theorems and proofs presented in Chapter 3. In that chapter, we will apply the volume axioms to the specific case of the n-dimensional analogue of a parallelogram, known as an n-parallelotope.

2.1 Axioms of n-Dimensional Volume

We denote the *n-dimensional volume of a set* as the function vol^n that is defined on subsets of $\mathbb{R}^n$. In particular, the subset must be non-negative and bounded, which is consistent with the characteristics of the spectral data to which this concept of volume is applied. Now, consider U, V to be figures with measurable volume. For the volume function vol^n, we establish the following axioms:[8]

a) For one dimensional figures ($n = 1$), we define: $vol^1(point) = 0$ and $vol^1(segment) =$ length of segment, *i.e.*, the difference of the two vectors that define the segment's endpoints.

b) The volume function vol^n is countably additive.

$$vol^n\left(\bigcup_{i=1}^{\infty} U_i\right) = \sum_{i=1}^{\infty} vol^n U_i \quad \text{if} \quad U_i \cap U_j \neq \emptyset \quad \text{for} \quad i \neq j.$$

In other words, given a collection of mutually disjoint figures, the volume of their union is equal to the sum of their respective volumes.

c) If $U \subseteq V$, then $vol^n U \subseteq vol^n V$.

d) Consider if $\mathbb{R}^n = \mathbb{R}^a \oplus \mathbb{R}^b$, an orthogonal direct sum with dim $\mathbb{R}^a = a$, dim $\mathbb{R}^b = b$, $U \subset \mathbb{R}^a$, and $V \subset \mathbb{R}^b$. Then, for the cartesian product $U \times V = \{(\boldsymbol{u}, \boldsymbol{v}) \mid \boldsymbol{u} \in U \text{ and } \boldsymbol{v} \in V\} \in \mathbb{R}^a \oplus \mathbb{R}^b$, we have that

$$vol^{a+b}(U \times V) = vol^a U \cdot vol^b V.$$

We know that the area of a rectangle is given by the product of the magnitudes of its sides; this is a strong generalization of that property to higher dimensions.

e) For a linear operator $f : \mathbb{R}^n \to \mathbb{R}^n$,

$$vol^n(f(U)) = |\det f| vol^n U, \; n = \dim \mathbb{R}^n.$$

One of the results of these axioms, in particular part (d), has to do with the volume of a set when computed in a dimension that is higher than the inherent dimensionality of the set. To illustrate this idea in a simple form, we can start by considering a rectangle of data, which is a two-dimensional figure whose volume formula is

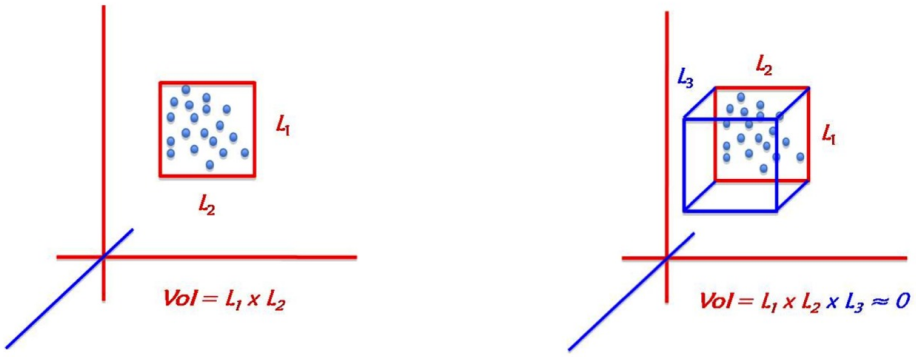
$$vol^2\{\text{2-D rectangle}\} = \text{length} \times \text{width}.$$

Next, we consider the three-dimensional version of a rectangle, which is known as a cuboid or a rectangular parallelepiped,[9] and has a volume formula of

$$vol^3\{\text{3-D cuboid}\} = \text{length} \times \text{width} \times \text{height}.$$

When considered in three dimensions, a rectangle has no measure for "height," and so it effectively has a "height" of magnitude zero. This means that the three-dimensional volume of a two-dimensional rectangle is

$$vol^3\{\text{2-D rectangle}\} = \text{length} \times \text{width} \times 0 = 0. \tag{2.1}$$



(a) (b)

Figure 2.1: A simple example of volume calculations in multiple dimensions.[10] (a) Two-dimensional data with volume calculated in two dimensions. (b) Two-dimensional data with volume calculated in three dimensions.

An illustration of the idea in Equation 2.1 is offered in Figure 2.1.

Using the axioms of the volume function vol^n as described above, the idea of the volume going to zero can easily be generalized to higher dimensional sets of data (that need not have rectangular surfaces). This result will play an important role in the theory presented in Chapter 3.

Theorem 1. *Consider a bounded set U in $\mathbb{R}^n$ that lies specifically in the subspace $\mathbb{R}^a$. Then the $(a+b)$-dimensional volume of U, where $a < a+b$, is zero.*

Proof. Observe that in $\mathbb{R}^n$, we have $U = U \times \{\mathbf{0}\}$ (the zero vector). Also, note that $vol^b(\{\mathbf{0}\}) = 0$ for $b > 0$. Lastly, we use the formula in axiom (d) of the volume function to obtain

$$vol^{a+b}(U \times \{\mathbf{0}\}) = vol^a U \cdot vol^b(\{\mathbf{0}\}) = vol^a U \cdot 0 = 0. \tag{2.2}$$

■

Chapter 3

The Gram Matrix

For a given set of vectors, linear algebraic techniques may be used to compute the volume of an n-dimensional figure that is defined by those vectors. When calculating the volume of an n-dimensional parallelotope – the generalization of a parallelogram to multiple dimensions – the Gram matrix is particularly useful. Developed by Danish mathematician Jörgen Pedersen Gram, the Gram matrix $\boldsymbol{G}$ is a symmetric matrix of inner products.[8] Given a set of k vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k$, $\boldsymbol{G}$ is defined as the $k \times k$ matrix where each entry is determined by

$$\boldsymbol{G}_{i,j} =< \boldsymbol{v}_i, \boldsymbol{v}_j >, \quad i, j \in 1, 2, \ldots, k. \tag{3.1}$$

Because the application presented here is to data in hyperspectral images, all of the vectors that are used are real-valued (and nonnegative). Consequently, the inner products that are calculated to generate each element of the Gram matrix simply become dot products. The relevant Gram matrix now has the $\{i, j\}^{th}$ element given by the dot product of the i^{th} and j^{th} vectors, and if we consider these vectors as columns of the matrix $\boldsymbol{V}$, we have

$$\begin{aligned} \boldsymbol{G}(\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k) &= \boldsymbol{V}^t \cdot \boldsymbol{V} \\ &= \begin{bmatrix} \boldsymbol{v}_1^t \\ \boldsymbol{v}_2^t \\ \vdots \\ \boldsymbol{v}_k^t \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \ldots & \boldsymbol{v}_k \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{v}_1 \cdot \boldsymbol{v}_1 & \boldsymbol{v}_1 \cdot \boldsymbol{v}_2 & \ldots & \boldsymbol{v}_1 \cdot \boldsymbol{v}_k \\ \boldsymbol{v}_2 \cdot \boldsymbol{v}_1 & \boldsymbol{v}_2 \cdot \boldsymbol{v}_2 & \ldots & \boldsymbol{v}_2 \cdot \boldsymbol{v}_k \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{v}_k \cdot \boldsymbol{v}_1 & \boldsymbol{v}_k \cdot \boldsymbol{v}_2 & \ldots & \boldsymbol{v}_k \cdot \boldsymbol{v}_k \end{bmatrix}. \end{aligned} \tag{3.2}$$

The Gram matrix has the property that the square root of its determinant, known as the Gramian, is equal to the volume of the parallelotope with the given vectors as its edges.[8] In other words, it can be used to compute the volume of the parallelotope formed by a set of vectors. This idea is the foundation behind the algorithm presented here. In this section we review some relevant definitions and theorems about the rank of a matrix, offer several theorems and proofs related to the Gram matrix, and then conclude these with a proof of the relation between the square root of the Gramian and the volume of the corresponding parallelotope.

3.1 Definitions and Theorems

Definition 1. *Rank of a matrix.*[11]
The *rank* of an $m \times k$ matrix $\boldsymbol{A}$ is the dimension of its row and column spaces (note that the row and column spaces have the same dimension). Denoted $rank(\boldsymbol{A})$, it has the property that $rank(\boldsymbol{A}) \leq \min\{m, k\}$.

Definition 2. *Null space of a matrix.*[11]
The *null space* of an $m \times k$ matrix $\boldsymbol{A}$ is the subspace of $\mathbb{R}^n$ that consists of all solutions to the homogeneous linear system $\boldsymbol{Ax} = \boldsymbol{0}$. It is denoted $null(\boldsymbol{A})$ and its dimensionality is denoted $nullity(\boldsymbol{A})$.

Theorem 2. The Rank Theorem.[12, 11] *For an $m \times k$ matrix $\boldsymbol{A}$ and the homogeneous system*

$$\boldsymbol{Ax} = \boldsymbol{0},$$

the set of solutions constitutes a vector space (in particular, the null space $null(\boldsymbol{A})$) of dimensionality $nullity(\boldsymbol{A}) = k - rank(\boldsymbol{A})$. Equivalently, for the same matrix $\boldsymbol{A}$ and homogenous system, it holds that

$$nullity(\boldsymbol{A}) + rank(\boldsymbol{A}) = k. \tag{3.3}$$

Theorem 3. The Rank of the Gram Matrix.[12] *Consider an $m \times k$ matrix $\boldsymbol{A}$ (i.e., a matrix with k-many m-dimensional column vectors) as well as the corresponding $k \times k$ Gram matrix $\boldsymbol{G}$. Then,*

$$rank(\boldsymbol{G}) = rank(\boldsymbol{A}). \tag{3.4}$$

Proof. We will construct this proof by first showing that $\boldsymbol{G}$ and $\boldsymbol{A}$ have equivalent null spaces, and then we will invoke the Rank Theorem to show that they are of equal rank. To illustrate that they have equivalent null spaces, we will show that (i) $null(\boldsymbol{G}) \subseteq null(\boldsymbol{A})$ and (ii) $null(\boldsymbol{A}) \subseteq null(\boldsymbol{G})$.

(i) Suppose that $\boldsymbol{x} \in null(\boldsymbol{G})$. Then $\boldsymbol{Gx} = \boldsymbol{0}$, or equivalently (because $\boldsymbol{G} = \boldsymbol{A^t A}$ by definition), $\boldsymbol{A^t Ax} = \boldsymbol{0}$. Left multiplying both sides by $\boldsymbol{x^t}$, we have

$$\begin{aligned} \boldsymbol{x^t A^t Ax} &= \boldsymbol{x^t 0} \\ \boldsymbol{x^t A^t Ax} &= 0 \\ \boldsymbol{(Ax)^t Ax} &= 0 \\ \boldsymbol{(Ax)} \cdot \boldsymbol{(Ax)} &= 0 \end{aligned}$$

It follows from this last line[11] that $\boldsymbol{Ax} = \boldsymbol{0}$. This means that $\boldsymbol{x} \in null(\boldsymbol{A})$, and hence $null(\boldsymbol{G}) \subseteq null(\boldsymbol{A})$ as desired.

(ii) Similarly, suppose that $\boldsymbol{x} \in null(\boldsymbol{A})$. Then we have that $\boldsymbol{Ax} = \boldsymbol{0}$. Left multiplying both sides by $\boldsymbol{A^t}$, we have

$$\begin{aligned} \boldsymbol{A^t Ax} &= \boldsymbol{A^t 0} \\ \boldsymbol{A^t Ax} &= \boldsymbol{0} \end{aligned}$$

By substituting $\boldsymbol{G}$ in for $\boldsymbol{A^t A}$ in the last line, we obtain $\boldsymbol{Gx} = \boldsymbol{0}$. This means that $\boldsymbol{x} \in null(\boldsymbol{G})$, and hence $null(\boldsymbol{A}) \subseteq null(\boldsymbol{G})$ as desired.

By (i) and (ii), we can conclude that

$$null(\boldsymbol{G}) = null(\boldsymbol{A}). \tag{3.5}$$

Because $\boldsymbol{G}$ and $\boldsymbol{A}$ have the same null space, it follows that

$$nullity(\boldsymbol{G}) = nullity(\boldsymbol{A}). \tag{3.6}$$

Lastly, we invoke Theorem 2 (the Rank Theorem), which tells us that $nullity(\boldsymbol{G}) = k - rank(\boldsymbol{G})$ and that $nullity(\boldsymbol{A}) = k - rank(\boldsymbol{A})$. By using Equation 3.6, we can combine these two equalities together to give us

$$k - rank(\boldsymbol{G}) = k - rank(\boldsymbol{A}), \tag{3.7}$$

from which it follows that

$$rank(\boldsymbol{G}) = rank(\boldsymbol{A}). \tag{3.8}$$

■

Theorem 4. Volume of a Parallelotope.[8] *The parallelotope with edges* $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\}$ *has a volume equal to* $\sqrt{det(\boldsymbol{G})}$*, where* $\boldsymbol{G}$ *is the corresponding Gram matrix of the edges.*

Proof. [8]
Case 1: $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\}$ are linearly dependent.
If the vectors $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\}$ are linearly dependent, the parallelotope corresponding to these edges lies in a space with dimension less than k. Then, according to the result in Theorem 1, the k-dimensional volume of the set is zero. In other words, $vol^k(\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\}) = 0$. Next, we want to show that the $\sqrt{det(\boldsymbol{G})}$ is zero. Consider the matrix $\boldsymbol{A}$ with $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\}$ as column vectors. Because the vectors are linearly dependent, it follows that $rank(\boldsymbol{A}) < k$. Then $\boldsymbol{G}$, a $k \times k$ matrix, has $rank(\boldsymbol{G}) < k$ since $rank(\boldsymbol{G}) = rank(\boldsymbol{A})$ by Theorem 3. That means that the columns of $\boldsymbol{G}$ are linearly dependent, and so $\boldsymbol{G}$ is singular and $det(\boldsymbol{G})$ is zero[11]. Hence $\sqrt{det(\boldsymbol{G})} = vol^k(\boldsymbol{A}) = 0$.

Case 2: $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\}$ are linearly independent.
Let the vectors $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_k\}$ be an orthonormal basis in $\mathbb{R}^k$ and let f be a linear mapping such that $f : \mathbb{R}^k \to \mathbb{R}^k$ transforms $\boldsymbol{e}_i$ into $\boldsymbol{a}_i$, $i = 1, ..., k$. If $\boldsymbol{\mathcal{A}}$ is the matrix of this mapping in the basis $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_k\}$:

$$(\boldsymbol{a}_1 \ldots \boldsymbol{a}_k) = (\boldsymbol{e}_1 \ldots \boldsymbol{e}_k)\boldsymbol{\mathcal{A}}$$

then the Gram matrix corresponding to $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\}$ equals $\boldsymbol{\mathcal{A}^t\mathcal{A}}$, because the Gram matrix of $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_k\}$ is the identity matrix:

$$\begin{aligned}
\boldsymbol{G}(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k) &= \big((\boldsymbol{e}_1 \ldots \boldsymbol{e}_k)\boldsymbol{\mathcal{A}}\big)^t\big((\boldsymbol{e}_1 \ldots \boldsymbol{e}_k)\boldsymbol{\mathcal{A}}\big) \\
&= \boldsymbol{\mathcal{A}}^t(\boldsymbol{e}_1 \ldots \boldsymbol{e}_k)^t(\boldsymbol{e}_1 \ldots \boldsymbol{e}_k)\boldsymbol{\mathcal{A}} \\
&= \boldsymbol{\mathcal{A}}^t\boldsymbol{I}_k\boldsymbol{\mathcal{A}} \\
&= \boldsymbol{\mathcal{A}}^t\boldsymbol{\mathcal{A}}
\end{aligned}$$

(Note that this matrix $\boldsymbol{\mathcal{A}}$ is different from the matrix in Case 1 in that this matrix does not have the vectors $\{\boldsymbol{a}_i\}$ as its column vectors; to avoid confusion, script $\boldsymbol{\mathcal{A}}$ has been used.) It follows from here that

$$\sqrt{det(\boldsymbol{G})} = \sqrt{|det(\boldsymbol{\mathcal{A}^t\mathcal{A}})|} = |det(\boldsymbol{\mathcal{A}})|.$$

Furthermore, $|det(\boldsymbol{\mathcal{A}})| = |det(f)|$ and f transforms the unit cube into the parallelotope that we are considering. And lastly, according to axiom (e) of the volume function, $|det(f)|$ also equals the volume of the parallelotope in question. Hence, the volume of the parallelotope with edges given by $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\}$ has a volume equal to $\sqrt{det(\boldsymbol{G})}$. ■

From this section, we see that the Gram matrix serves as a very useful tool when calculating the volume of a parallelotope defined by a set of vectors. As long as we are able to identify the outer edges of the set, we can easily implement this method of determining the square root of the Gramian. Because of the work shown in Case 1 of the proof for Theorem 4, we can also see that the Gram matrix is helpful when attempting to establish whether the set of vectors is linearly dependent or linearly independent. In the following section we will show that this is particularly useful in estimating the dimensionality of a set of data, as the data will have a volume of zero when calculated in any dimension that is greater than that of its inherent dimensionality.

Chapter 4

Methodology

The goal of SVE is to cue analysts to areas of interest within an image, whether it be due to complexity or to detected changes from another image of the same scene. In order to identify these interesting regions, the areas within the image(s) have to be characterized by numbers and then compared relatively. True to its name, SVE extracts these metrics after using a simplex-based geometric model to identify the endmembers, or vectors corresponding to the vertices of the simplex approximating the data. The Gram matrix is then used to compute information about the volume of the parallelotope whose edges are defined by the given set of endmembers.

4.1 Characterizing the Dataset

When considering an n-dimensional data set, the simplex approximated by the data is characterized by the $n + 1$ endmembers (as datapoints) that define the corners of the simplex.[1] The Linear Mixture Model (LMM) presented in Section 1.2 assumes the corners of the simplex to be the materialistically "pure" pixels within the image, whereas the "mixed" pixels (*i.e.*, everything else) lie inside of the simplex edges in the n-dimensional space.[13] The idea is that because a mixed pixel comprises multiple materials, its vector representation will be a linear combination of the vectors corresponding to the pure pixels of those materials. As such, the Linear Mixture Model allows us to use the properties of the enclosing simplex to estimate the complexity of a collection of pixels when considered relative to other collections of pixels within the image.[4, 10] We delineate these collections of pixels through a tiling procedure that segments the image into equal-sized square regions; this allows for analysis of the image (or images) on a local level. When applying this technique specifically to change detection, we tile two registered images in the same

way and then compare the complexity of corresponding tiles.

When using the LMM, we operate under the assumption that more materialistically diverse regions of an image will occupy a larger volume in the n-dimensional real vector space $\mathbb{R}^n$. Further, those pure materials will be well-separated and will characterize the corners of the enclosing simplex.[10] This is made apparant by the behavior of such pixels when considered in vector form. In particular, large and uniform regions of an image will occupy the spectral space in a relatively tight, single-material cluster where variability is driven only by any spectral variations within the material itself. Alternatively, more highly cluttered regions of an image will be composed of multiple distinct materials that are wider spread throughout the space and will correspond to a simplex with a much higher volume than that of a less cluttered region.[10] There is a directly proportional relationship between the number of distinct materials within a region and the degree of the magnitude of its inherent volume; this relationship is of important use when analyzing the complexity of a region relative to another region in SVE.

4.2 Identifying Endmembers

If we have the ability to approximate the spectral volume for a collection of pixels, then we have an important tool for characterizing images. We want to use mathematical techniques to analyze a dataset, but are unable to do so without first answering a critical question: *what is the inherent dimensionality of the data?* Specifically, how many endmembers do we need in order to represent the collection of pixels? Unfortunately, it is not as simple as using the number of components within a given vector. That value is directly determined by the number of spectral bands at which the hyperspectral image was initially collected, which can be upwards of 126+ different wavelengths (and although there is no theoretical limitation on that number, there are practical limitations). The data tend to live in a manifold with dimension less than n, *i.e.*, less than the number of spectral bands. As a result, the inherent dimensionality of the data – and subsequently the number of endmembers needed to define that set of data – is usually far lower than n. The question of how many endmembers to use has plagued many image-analysis algorithms, and this one is no exception.

The challenge becomes finding a way to calculate the k-dimensional volume of an inherently k-dimensional dataset, but without knowing the exact value of k. Fortunately, because we are working with actual spectral data, we can narrow down the possible values that k could have. Because the number of endmembers is related to the number of pure materials in the test set, the value for k tends to change depending on the size of the set of pixels and can range anywhere from one to over fifty. We can use this information to place an upper bound on k (where the lower bound on our possible dimensionality is

theoretically one). For each set of pixels, we take an iterative approach in that we choose an exceptionally high number of endmembers relative to the number of pixels in the dataset to use as an upper bound, and each time we increase the index, we compute the volume of the parallelotope in that dimension. This results in the approximated parallelotope volume calculated as a function of the dimensionality (or number of endmembers) for the set of pixels.[10]

In order to identify the endmembers of a given set of pixels, we employ the Max-D algorithm.[3] For a given set of data and a specified number of desired endmembers, Max-D will compute and return those endmembers through a series of vector and planar projections. Although we use the Max-D algorithm, other methods may also be utilized. However, it is important to note that the Max-D algorithm returns the endmembers sorted in order of decreasing vector magnitude, which is a key component in the method behind the SVE algorithm.[10]

4.3 Parallelotope Volume

To calculate the volume of the parallelotope approximated by a set of vectors, a common and simple mathematical approach is to calculate the determinant of the matrix that has those vectors as columns. Unfortunately, taking the determinant in this way requires one of two things: (i) the vectors form a square matrix or (ii) the vectors are reduced in dimensionality so as to form a square matrix. The first possibility is incredibly unlikely, as the number of spectral bands for each pixel would have to exactly match the dimensionality of the set. In actuality, the inherent dimensionality of the data is usually quite low (sometimes as small as 3), while the endmembers are collected in hyper-dimensions. The second possibility is highly undesirable because statistical methods would need to be used in order to reduce the dimensionality of the vectors in the matrix, and doing so would result in a loss of accuracy in the data. For these reasons, attempting to use the determinant in this way for SVE is limiting and impractical.

The ideal technique for calculating the volume is one that simply allows us to use the set of endmembers without having to manipulate them in any way. Fortunately, the Gram matrix does just that.[8] As described in Section 3, the Gram matrix is a square matrix that corresponds to a given set of vectors such that if $\boldsymbol{V}$ is the matrix with those vectors as its columns, then the relevant Gram matrix is $\boldsymbol{G} = \boldsymbol{V}^{\boldsymbol{t}}\boldsymbol{V}$. The Gram matrix has the property that the square root of its determinant (the Gramian), $\sqrt{|det(\boldsymbol{G})|}$, is equal to the volume of the parallelotope with that initial set of vectors as edges. A proof for this is offered with Theorem 4. The Gram matrix also has the characteristic that if the vectors given are not linearly independent, then it will return a value of zero for the square root of its determinant. When applied to a set of endmembers, a Gramian of zero indicates

that the number of endmembers has exceeded the inherent dimensionality of the data. These properties demonstrate how the Gram matrix addresses the two main challenges we face in developing this algorithm around the idea of analyzing the volume of a particular set of data. First, it does not require that the vectors be of any particular form (*i.e.*, a square matrix), and second, it has a way of revealing when we have gone beyond a dataset's inherent dimensionality. This makes the Gram matrix a vital component of the SVE algorithm.

4.4 Local Gram Matrix

In the Simplex Volume Estimation algorithm, we invoke properties of the Gram matrix in order to calculate the volume of the parallelotope approximated by a set of vectors. For the set $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$, consider the matrix $\boldsymbol{V}$ that has those vectors as columns. The corresponding $k \times k$ Gram matrix is $\boldsymbol{G} = \boldsymbol{V^t V}$, where each $\{i, j\}^{th}$ element is given by the dot product between the i^{th} and j^{th} vectors so that

$$\boldsymbol{G}_{i,j} = \boldsymbol{v}_i \cdot \boldsymbol{v}_j. \tag{4.1}$$

For our purposes, the most important of its characteristics is that the Gram matrix allows us to compute the volume of the parallelotope with the given vectors as its edges. As proven in Theorem 4, $\sqrt{|det(\boldsymbol{G})|}$ is equal to the volume of the relevant parallelotope.

When applied to spectral data, this is a very useful technique for computing a volume estimate that is specifically related to a set of pixels. However, there is a bit of a concern: because we are working in $\mathbb{R}^n$, each endmember is based at the origin, even though the cloud of data may not live close to the origin. This means that a parallelotope with a corner at the origin is not necessarily an accurate representation of the volume of the data. To remedy this, we simply *localize the Gram matrix* about the one of the endmembers used to approximate the data.[10] This is done by identifying the endmember that is closest to the mean of the dataset, call this vector $\boldsymbol{x}$. Then, in order to localize the vectors about this point, we subtract each of the remaining vectors from this mean vector. This leaves us with a formula for the local Gram matrix, denoted by $\boldsymbol{G'}$, where

$$\boldsymbol{G'}(\boldsymbol{x})_{i,j} = (\boldsymbol{x} - \boldsymbol{v}_i) \cdot (\boldsymbol{x} - \boldsymbol{v}_j). \tag{4.2}$$

For a set of k vectors, the local Gram matrix will be $(k-1) \times (k-1)$ in dimension. The recentering of the parallelotope about the data yields a more precise volume estimation and will have greater sensitivity to changes within the volume of the set.

4.5 Gram Volume Function

The Gram matrix allows us to approximate the volume of the parallelotope spanned by a given set of data.[10] However, the ability to perform this calculation is dependent upon knowing the inherent dimensionality of the data. That is to say, if the set of data inherently constitutes a k-dimensional data cloud, then we need to know the value of k in order to calculate the volume. We need to know k so that we can indicate to Max-D the desired number of endmembers. This poses a challenge because, for a given set of data, we do not know *a priori* the dimensionality of the set. To address this problem, we simply choose an upper bound for the the possible values of k that is extremely high relative to the number of pixels in the set and then iterate through those values. During each i^{th} iteration, we calculate the i-dimensional volume where $i = 3, \ldots, k$. When the volume is calculated in a dimension that is higher than the inherent dimensionality, the volume has an approximate magnitude of zero (due to noise in the data, it does not truly reach zero).

When the magnitude of the volume determined by $\boldsymbol{G'}$ goes to zero, this indicates that we have exceeded the inherent dimensionality of the data. In other words, the volume of the corresponding parallelotope is being calculated in more dimensions than it actually occupies.[10] For a given set of data, these iterative calculations yield a Gram volume function where the independent variable is the number of endmembers used.

To examine how the volume changes between datasets composed of varying materials, we generated the Gram volume functions for several regions of interest. The regions of interest (ROIs) sampled were qualitatively assessed to have increasing complexity, and that is reflected in the graphs of the relevant Gram volume functions. The image chips shown in Figure 4.1 are characterized as (a) trees, (b) grass, (c) foothills, (d) grass, road, and trees, (e) a construction site, and (f) a small city. These ROIs were analyzed in full hyperspectral dimensionality and each is approximately 4,000 pixels.

(a) (b) (c) (d) (e) (f)

Figure 4.1: Several ROIs in a hyperspectral scene, in increasing order of complexity. (a) trees (b) grass (c) foothills (d) grass, road, and trees (e) a construction site (f) a small city.

For each image chip, the corresponding Gram volume function is shown in Figure 4.2. The plots demonstrate the volume of the parallelotope approximated by the dataset, plot-

ted as a function of the number of endmembers (which is related to the dimensionality). Before analyzing the content of these specific plots, it is important to note two general

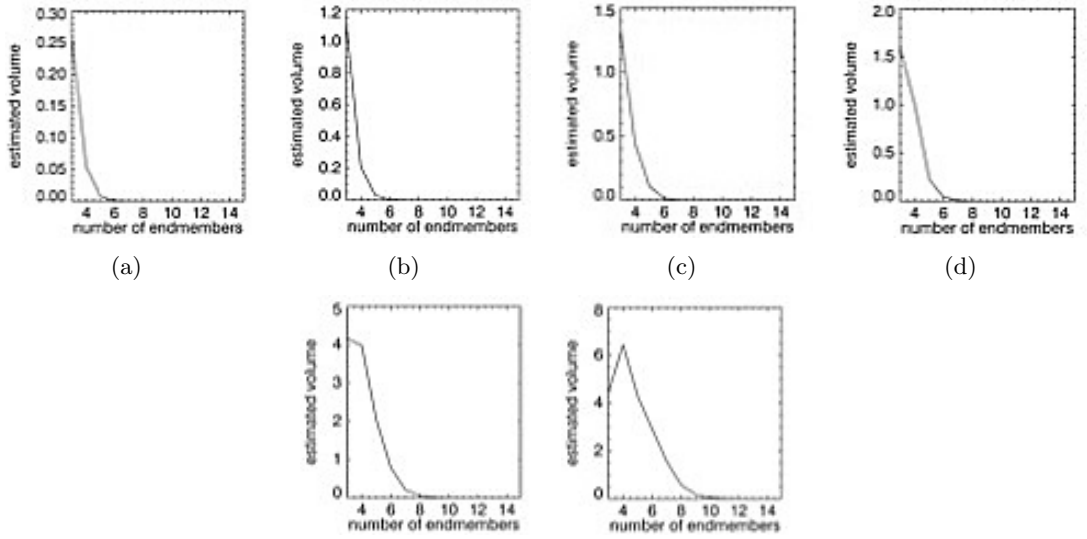


Figure 4.2: Complexity plots given by the Gram volume function of several ROIs in a hyperspectral scene, in increasing order of complexity. The dependent axis is the number of endmembers, and the independent axis is the estimated volume of the data set. (a) trees (b) grass (c) foothills (d) grass, road, and trees (e) a construction site (f) a small city.

things about the resulting graphs. The first is that the lowest number of endmembers in a graph is three. The reason for this is that after those n endmembers are obtained, they are localized about the endmember closest to the mean, which decreases the dimension of the Gram matrix to $(n-1) \times (n-1)$. With only one endmember to begin with, localization would cause the local Gram matrix to be the zero matrix. With only two endmembers to begin with, localization would lead to a local Gram matrix with a single entry. For these reasons, it is only practical to begin the iteration at three endmembers. This is supported by the data used, as two-dimensional datasets rarely occur naturally. Second, the estimated volume that is the dependent variable of the graphs is only the magnitude of the volume. Specifically, it has no units. When volume is calculated in k dimensions, it has k-dimensional units, *i.e.*, $units^k$. However, we are only interested in the magnitude of the volume, and so we are able to compare all of those volume measures on the same graph.

There are several points of interest in the plots in Figure 4.2. Most notable of these is that as the complexity of the tile content increase from (a) to (f) in Figures 4.1 & 4.2,

the overall magnitude of the plot also increases.[4] Regardless of scene content, however, the general shape of the plot is relatively unchanged; the maximum is achieved at low dimensionality and the plots monotonically decrease from there. The achievement of the maximum of the plot at a low dimensionality is due to the way in which the endmembers are determined, as MaxD returns them in order of decreasing magnitude. This means that the initial endmembers that are found correspond to the largest spread in the dataset, and so the lower-dimensional volume of those higher-magnitude vectors is relatively larger than the volume in higher dimensions. It is only in the case of the city tile that we observe an initial increase in the plot of the volume function, which is attributable to the presence of multiple, bright (i.e., higher-magnitude) endmembers.[4] Additionally, notice that the number of endmembers at which the plots approach zero increases with complexity; this is indicative of the increase in inherent dimensionality. However, due to the method, this inherent dimensionality value is necessarily quantized to an integer value. All of these properties provide us with several distinguishing features in the varying plots, and offer several candidates for the metric by which the graphs will be characterized.

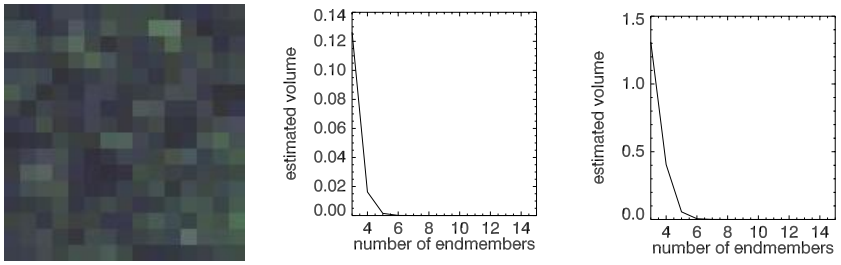
4.6 Metrics

In order to analyze the tiles within an image and compare them between images, we need to assign a numerical value to each tile that is related to the volume of the dataset. These values, or metrics, are extracted from the Gram volume function. As illustrated by the complexity plots in Figure 4.2, the function, when graphed, has three distinct metrics. They are (1) the inherent dimensionality of the data, (2) the area under the curve of the graph, and (3) the peak of the curve in the graph. For consistancy, we need to assign the same type of metric to each tile. As described above, the inherent dimensionality of the data is given by the number of endmembers at which the graph effectively levels off to zero. This occurs when the corners defining our parallelotope are no longer linearly independent. Because the number of endmembers at which the function levels off is not as sensitive to changes in the image, we do not use this for our metric. The second metric, which is the area under the curve of the graph, is given by a right-rectangular area approximation where each rectangle has a width of one unit. Lastly, the peak of the curve is indicative of the dimensionality of the real vector subspace $\mathbb{R}^m$ in $\mathbb{R}^n$ in which the set of data occupies the largest volume.[10] (Note, however, that this is not necessarily–and in fact, unlikely to be–the same as the inherent dimensionality.)

There appears to be a direct relation between the complexity (or number of distinct materials) within an image and the magnitude of its volume. In particular, datasets that occupy a larger spread in the spectral space will have greater peak and area metric values than those of a dataset that comprises fewer materials. Further, the introduction of a

new material to a tile will result in an increase in magnitude of the corresponding Gram volume function graph.

To illustrate the sensitivity of the volume to small changes within a dataset, we selected a 210 pixel region of interest (ROI) from a vegetative forest area in a hyperspectral scene and generated the graph of its Gram volume function. Then we replaced two of the forest pixels with two pixels from a different material–a rooftop–into the ROI and ran the algorithm on the modified chip. As a result, each one of the three metrics changed significantly. The peak of the graph increased by a factor of 10, as did the volume under



(a) (b) (c)

Figure 4.3: An example of the sensitivity of the Gram volume function to very small changes. (a) RGB of a region of interest from an area of dense vegetation in a hyperspectral image. (b) Gram volume function of tile. (c) Gram volume function of changed tile after two pixels from a city rooftop are swapped in.

the curve. The number of endmembers at which the graph went to zero went up by one endmember, which is indicative of the inherent dimensionality increasing because of the addition of a new material into the scene. The ROI and graphs corresponding to this are offered in Figure 4.3. Notice how the introduction of these two very different spectra into the set of pixels remarkably changed the volume function. This shows the dramatic increase in the volume of the convex hull.

4.7 Metric Testing

The peak of the graph and the volume under the curve appear to be the most sensitive to variability within an image, so they are the strongest candidates for our tile metric. In order to test this, we implement a *pixel injection scheme*, where we take an ROI of natural material and iteratively inject (or replace original pixels in the scene with) uniformly incremented amounts of manmade material pixels. This allows us to observe how much of a change occurs in these metrics when new and distinct materials are added to the tile.

We used three ROIs of grass pixels, foothill pixels, and forest pixels respectively. In order to make sure that the results were consistent, we performed multiple statistical trials and took the average. For each ROI, we performed 100 different statistical trials, and for each of these trials we performed 100 iterative pixel injections. This means that the measure of each metric (peak or area) that corresponds to a specific number of injected pixels for a particular ROI is the average of 100 different statistical trials. In Figure 4.4 below we show all of the graphs from the pixel injection metric testing.

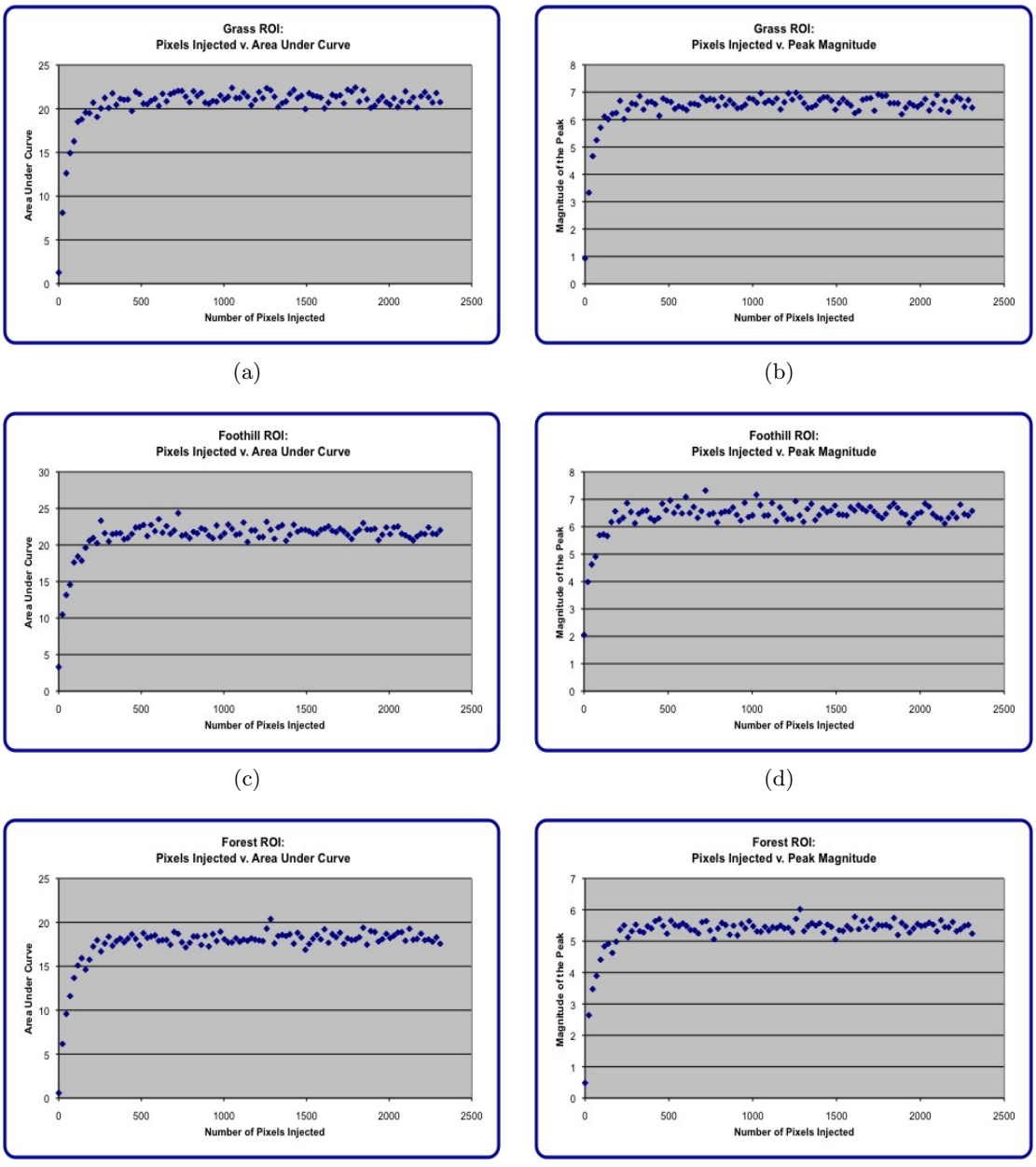


Figure 4.4: The charts resulting from the pixel injection trials. (a) Changes in the area metric for a grass ROI. (b) Changes in the peak metric for a grass ROI. (c) Area metric for a foothill ROI. (d) Peak metric for a foothill ROI. (e) Area metric for a forest ROI. (f) Peak metric for a forest ROI.

From the charts presented in Figure 4.4, we can instantly see that the metrics are very sensitive to materialistic changes within a set of pixels. The rapid increase in magnitude for both the area and the peak occurs immediately after the first injection, and continues to rise until about 250 pixels have been injected. From there, the values for the metric begin to level off. This is to be expected as the convex hull characterized by this set of pixels is not growing in volume; while more pixels are being added to part of the datacloud occupied by the manmade pixels, the actual shape of the cloud is not changing. Rather, the pixels are just becoming concentrated in different areas. This trend in the data of rapidly increasing with the initial injections is exactly the result that we desired. The goal of SVE is to detect materialistic changes within an image, specifically when new materials are introduced. These charts illustrate that the peak and and area metrics do just that.

4.8 Tiling Scheme

When analyzing an entire image, it can be difficult to extract information about the image on a local level when it is processed globally. By implementing a tiling scheme,[14] we are able to analyze the whole image by processing individual regions. Specifically, we spatially segment an image into equal-sized squares and then analyze these tiles one-by-one, assigning each of them a metric based on the relevant algorithm (here, we use the parallelotope volume technique in SVE). A comparison of the metrics between tiles presents an overall analysis of the image that allows tiles of "interest" in the image (where interest is relative to the goal of the algorithm and its metric) to be highlighted.

This tiling scheme also provides several added benefits to change detection algorithms. Requiring precise registration between two images presents a very difficult standard and may limit the number of images to which an algorithm may be applied. Rather than compare the two images on a pixel-by-pixel basis, SVE is able to compare them on a tile-by-tile basis, which has the less strict requirement of only needing closely-registered images. Tiles are far more forgiving with the registration of images and allow change analysis to be done on regions of the image rather than on specific pixels. When performing change detection, SVE tiles the two images in the exact same way, which allows for comparison between the tiles that pertain to the same part of the processed scene.

With SVE in particular, this tiling scheme is necessary in order for the idea of approximated parallelotope volume to be a viable analysis technique. While obtaining the volume spanned spectrally by an entire image would tell us some information about the overall complexity of the image, it would not cue us to the more complex areas of the image (or, in change detection, the areas where a change has occurred). The parallelotope volume is also more sensitive to changes when the dataset is relatively small (several hundred to a few thousand pixels), as opposed to the volume spanned by several million pixels. Another

benefit is decreased processing time.

Chapter 5

Outline of Algorithm

The SVE algorithm is able to process images in two different ways:

(i) for a pair of registered images corresponding to the same scene, SVE can highlight the areas of materialistic changes between those images, and

(ii) for a single image, SVE can indicate the areas of higher complexity relative to the rest of the image, *i.e.*, the areas of "interest".

In order to analyze images in these ways, we ultimately need to be able to identify each tile in each image by a certain number that will characterize its relative "interest", whether it be in change detection or as a complexity measure. That is to say, we must be able to extract metrics that either will serve as a *cue to changes* for a pair of images or a *cue to complexity* for a single image. As described in Chapters 6 & 7, this requires a very similar method of processing the images by SVE, with the exception of a slight variation in the calculation of the metrics. In Figure 5.1 we have a graphic outline of SVE that shows the comparison between the change detection method and the complexity method of image processing. Note that identical steps are indicated by the same color, whereas differing steps are indicated by different colors. In particular, the *cue to changes* is done by computing the *change in metric* for each pair of tiles, given by subtracting the volume metrics between the two images. The *cue to complexity* is done by computing the volume metric for each tile in the single image. Chapter 7 offers a detailed explanation of the change detection algorithm development, and Chapter 6 delves further into the complexity estimation algorithm development.

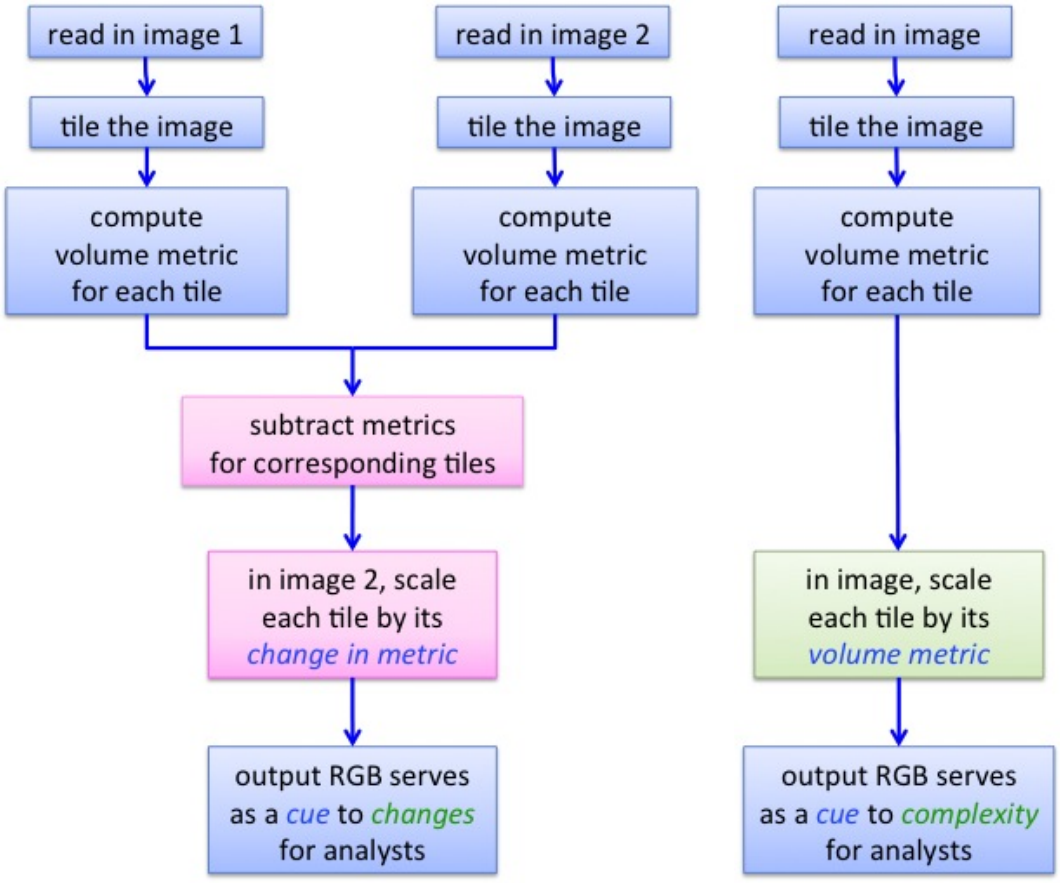


Figure 5.1: Flow chart offering a comparison of the two ways in which SVE can implement image analysis (for either a pair of images or a single image). (a) Change detection flow chart. (c) Complexity measure flow chart.

Chapter 6

Application to Complexity Estimation

While the Simplex Volume Estimation algorithm provides a way to analyze changes between images, it is not always possible to obtain two hyperspectral images of the same scene. That is to say, analysts sometimes only have one image of a scene, and so any analysis must be done relative to the image itself. Intuitively, the more "complex" an image is, the more difficult it will be to extract information from it. In general, though, a measure of this complexity is still unknown. Here, we offer an extension of the SVE algorithm that characterizes the complexity of an image on a local level. We use the metric provided in SVE to quantify, within a single image, areas of relatively more or less complexity.

6.1 Metric

The approach to this analysis is very similar to that of its application to change detection. We continue to operate under the assumption that more complex regions of an image (i) contain a larger number of materials relative to other areas in the image and (ii) those materials will be relatively more spectrally diverse.[4] In contrast, regions of less complexity are composed of a small number of distinct materials that are more spectrally similar. When considered in the context of measuring the volume of the parallelotope approximated by a set of vectors, the more complex regions will exhibit a larger volume relative to that of less complex regions. We saw in Figures 4.1 & 4.2 that more spectrally diverse ROIs exhibit larger volumes and hence larger metrics. This means that, when processed by SVE, the relatively more complex tiles of an image (which occupy larger volumes) will have higher metrics. We use this as the basis for our complexity analysis.

In other words, the metric extracted from the Gram volume function as described in Section 4.6 (in this case, the peak) serves as our complexity metric.

6.2 Algorithm

When SVE executes its procedure for change detection, it does so requiring two approximately registered hyperspectral images of the same scene. In order to adapt it to complexity analysis, we alter the algorithm so that it is only processing a single image. We offer the following pseudo-code.

- read in one image
- tile the image
- iterate through tiles
  - iterate through endmembers, 3 to 15 (change depending on image)
    - calculate local Gram volume
    - store in volume function
  - obtain volume as a function of endmembers (see Figure 4.2)
  - extract peak metric from volume function
- identify the maximum peak metric for all tiles
- scale all values by the maximum
- output scaled RGB - brighter tiles indicate more change

This is graphically represented by the flow chart in Figure 6.1, which is also offered in Figure 5.1, where it is compared to the complexity algorithm in SVE.

Here, we scale each tile in the single processed image by the maximum peak metric within the image. The difference between this complexity pseudo-code and the change detection pseudo-code offered in (include section!!!!!) is that in change detection, we scale each tile by the maximum change in peak metric values between two images. When SVE processes a single image and outputs the corresponding scaled RGB image, the brightest tiles indicate areas of higher relative complexity. This cues an analyst to areas within the image that are the most "interesting," or are the most spectrally (and thus materialistically) diverse.

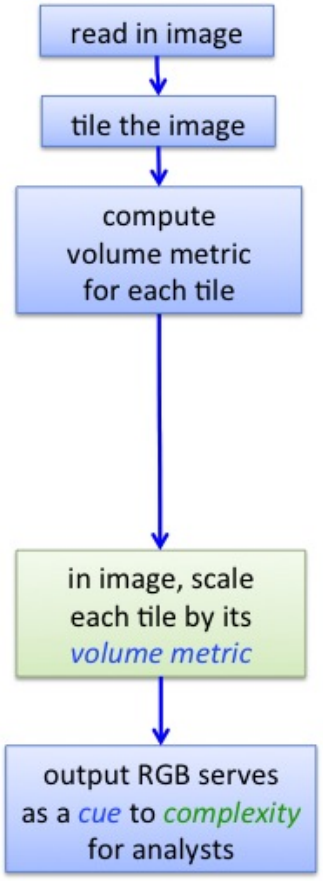


Figure 6.1: Flow chart of the complexity algorithm in SVE. A comparison of this against the change detection algorithm is offered in Figure 5.1.

6.3 Experiment

To test this method of complexity analysis with SVE, we processed hyperspectral images and then magnified the cued "areas of interest" to see if they were, in fact, more spectrally diverse. This analysis was applied to several sets of multi and hyperspectral images. The first of these image sets was taken of Cooke City, MT and collected using the HyMAP sensor as part of the 2006 CHARM collection.[15] These images have 126 spectral bands, and were collected with an approximate ground sample distance (GSD) of 2-3 m.[10] This scene offers both vegetative and man-made materials, although it is mostly vegetation. It was expected that SVE would identify the metropolitan area as being more complex, as well as any other man-made materials throughout the scene.

The second set of images that were tested were taken over Esperanza, CA, shortly after a forest fire had occured. These images were multispectral and collected over 4 bands at a GSD of approximately 2.5 m. The images contain a few obvious areas of higher complexity, but most of the image is not able to be roughly classified by the naked eye. The large area search algorithm will identify any areas within the image that are as complex as the city areas, and will scale the remaining "complex" regions by how complex they are relative to the most complicated tile in the scene.

The third set of images were collected using the Worldview-2 sensor across 8 bands with 2 m GSD. The scene is of a relatively complicated-looking water scene, and analysis is used to show if there is in fact anything complex happening in the scene. The last set of images was taken over Afghanistan using the RapidEye sensor, which has 5 bands and a 5 m GSD. These images help to identify a potential area of interest that would otherwise not have been obvious.

For this analysis, all images were delineated into the same pattern of 30×30 pixel tiles. In each collection, the data for the images was measured in approximate surface reflectance. For each tile, the upper bound on the number of endmembers used when implementing the SVE method was 15 endmembers, as testing showed that none of the tiles (even the more urban-centric ones) exceeded this dimensionality, nor came relatively close to it. This provided a Gram volume function that was certain to drop to zero before 15 endmembers was reached, and as such allowed for the peak in the volume function to be used as the metric of change. Results of the complexity experiments are shown in Section 8.1.

Chapter 7

Application to Change Detection

7.1 Metric

The application of these metrics to the SVE algorithm is done by computing the Gram volume function for each tile. From there, the three metrics for that tile are extracted from the functon. Although we collect the values for all three metrics, the one that has been chosen for use in SVE is the peak of the function. So for each tile in the image, we assign it the value of the peak in its volume graph. Because a change in volume likely indicates a change in materials, we subtract the peak values between corresponding tiles in the two different images. For each pair of tiles (one from each image), this difference yields a *change value*. Since we are analyzing the tiles in the image on a local level and relative to the other tiles, we normalize all of these values by scaling them against the largest of the change values. In other words, we identify the maximum change metric in the set of tiles and divide all of the other change metrics by that value. This scales down the assigned value for each tile in the images to a number on the interval $[0, 1]$.

Since we are looking to identify where the changes occurred, we choose to only apply this scaling to the second of the two images. The second image is assumed to be the more current image and consequently the one where we want to cue areas of interest. We then multiply each tile of pixels by its respective scaled value, which in turn scales the components of the vectors. This modifies the brightness of the pixels so as to provide visual cues to an analyst. When output to an RGB image, brighter tiles are indicative of a materialistic change within the tile, while darker tiles are those which, by this method, contain few to no changes.

This visual representation through modifiying the brightness is important because it allows for a better understanding of what is going on materialistically in the image.

Ultimately this is what is needed for the algorithm to be useful to an analyst. Without the scaling, we would not be able to offer a visual of the changed tiles. To better understand the distinction between the complexity of a tile and the amount of change between two tiles, consider a hyperspectral image that is processed of a city. Because of the high variability in materials for that type of location, each tile would likely approximate a larger parallelotope volume in the spectral space. However, for change detection purposes, we only want to know what has changed in each tile, and how interesting of a change it is relative to other tiles. And so even though a tile from a city image may exist in several dimensions and have a large parallelotope volume, we are not interested in it if it does not change significantly between images. As a result, we need a way for SVE to scale down the brightness of such tiles so as to indicate that they do not possess any "interesting" changes. By subtracting the volumes and dividing by the maximum change value, SVE does exactly that.

7.2 Algorithm

The Simplex Volume Estimation algorithm was implemented in the IDL/ENVI programming language. The algorithm begins by using a tiling scheme to delineate the pixels in the image into square sections for comparison, which alleviates the need for precisely-registered images. Next, it performs a nested iteration by first cycling through the tiles and then, for each tile, iterating through the number of endmembers that are returned by Max-D. During each endmember iteration, it calculates the volume of the parallelotope as given by the square root of the Gramian. For a particular tile, this results in a set of numbers where each number corresponds to the i-dimensional volume (in this application, the tested dimensions run from $i = 3, ..., 15$) of the parallelotope, which together constitute a Gramian volume function of endmembers vs. magnitude of volume. A metric based on the maximum value in the Gramian volume function for each tile, or the "peak" of the function, is then extracted (as described in Section 4.6). For corresponding tiles between the two images, these peak metric values are compared, and substantial differences in value are considered to be indicative of a change. (The term "substantial" is relative to the largest change in metrics for all of the pairs of corresponding tiles.) Lastly, the change values are all scaled by the maximum value and the algorithm outputs a scaled RGB image. We offer the following pseudo-code:[10]

- read in two registered images
- tile the images
- for each image, iterate through tiles

  - iterate through endmembers, 3 to 15 (change depending on image)
    * calculate local Gram volume
    * store in volume function
  - obtain volume as a function of endmembers (see Figure
  - extract peak metric from volume function
  - compute difference in metric for this tile between images
- identify the maximum change value for all tiles
- scale all values by the maximum
- output scaled RGB - brighter tiles indicate more change

This is graphically represented by the flow chart Figure 7.1, which is also offered in Figure 5.1, where it is compared to the complexity algorithm in SVE.

7.3 Experiment

To test the SVE method for change detection and image analysis in hyperspectral imagery, the algorithm was applied to several sets of hyperspectral images. The first set of these images was collected using the HYDICE sensor,[1, 10] and it has a spectral range covering 0.4 to 2.5 μm and approximately 2-3 m spatial resolution. The two images within this set were collected over the same scene, separated by a few days taken at different times of day, but at approximately the same altitude. The content in the initial image is that of several targets placed out in an open and grassy field. In the second image, the targets were moved to the trees; that portion of the image (involving the trees) was not analyzed here, but the ability of SVE to detect the presence/removal of the known targets was tested.

The second of the sets of images was collected using the HyMAP sensor as a part of the CHARM collection in 2006[15]. The images in this collection have 126 spectral bands corresponding to approximately the same spectral range, and also have approximately 2-3 m ground sample distance (GSD).[10] This scene is mostly composed of natural vegetation, but also contains a small, semi-metropolitan developed area. The two images within this set were collected on the same day, separated in time by approximately one hour; between these collections, targets that were placed in the large and circular field (which appears just right of center in the scenes) were removed. Moreover, it is highly likely that there was activity in the town or city that consequently led to observable changes between the two images in the collection.

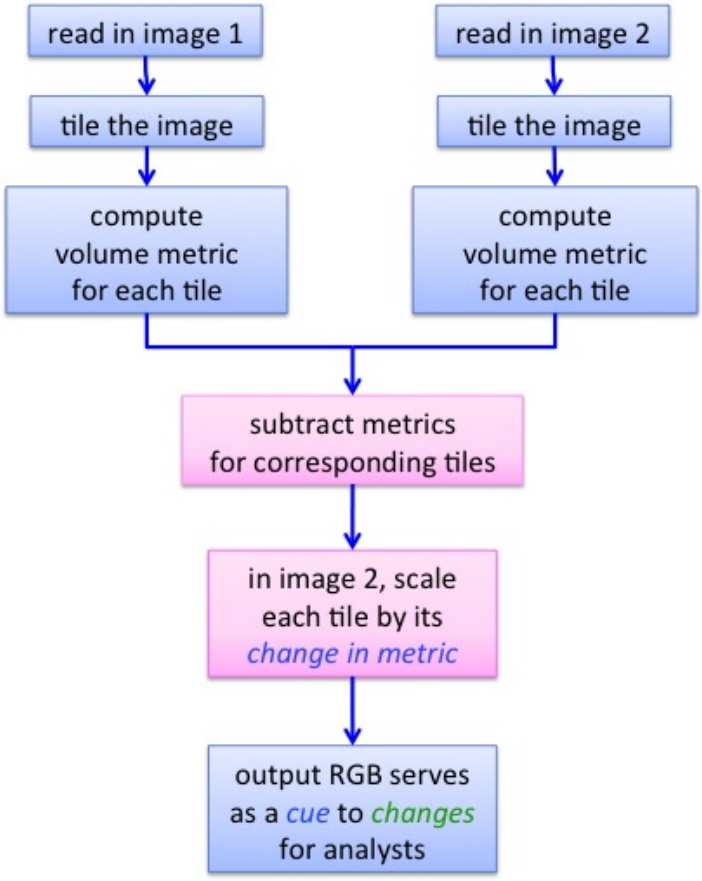


Figure 7.1: Flow chart of the change detection algorithm in SVE. A comparison of this against the complexity algorithm is offered in Figure 5.1.

For this analysis, both images were delineated into the same pattern of 30×30 pixel tiles. In each collection, the data for the images was measured in approximate surface reflectance. For each tile, the upper bound on the number of endmembers used when implementing the SVE method was 15 endmembers, as testing showed that none of the tiles (even the more urban-centric ones) exceeded this dimensionality, nor came relatively close to it. This provided a Gram volume function that was certain to drop to zero before 15 endmembers was reached, and as such allowed for the peak in the volume function to be used as the metric of change. Results of the change detection experiments are shown in Section 8.2.

Chapter 8

SVE Results

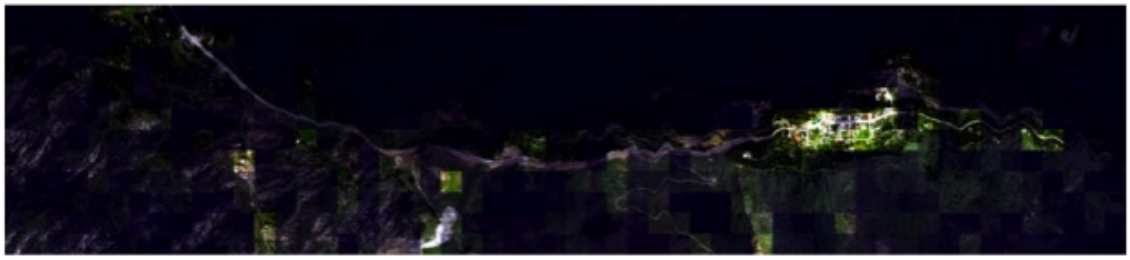
8.1 Complexity Results

In Figures 8.1– 8.13, we have several examples of the complexity measure of images. These applications of SVE were done to images where there was no prior knowledge of the content. Cued areas are investigated further through high-resolution magnifications of those areas. Figure 8.1 shows the large area search results of the Cooke City, MT image collected using the HyMAP sensor. Notice how the city is brightest and is thus cued as being the most complex. This follows intuition, as cities are expected to have many more materials than areas of vegetation.

Next, Figures 8.2 - 8.8 show the results of the analysis of the Esperanza, CA scene. Those images were collected multispectrally over 4 spectral bands, with a 2.5 m GSD. The original image shows what the city looks like before any analysis was done. The ensuing figures zoom in on particularly more complex areas of the scene, showing how SVE cues to areas that are not initially obvious as being of higher complexity. Figure 8.4 shows how one of the brightest tiles in the scene is not in the obvious complex or visible town, but rather is in the middle of the scene. By zooming in on those tiles and then observing the panchromatic image of that location (as shown in Figures 8.5 - 8.8), we are able to see the exact materialistic content of those tiles, and why they were cued as being more complex. Doing so reveals the presence of roads, building structures, water towards, and vehicles, all of which were not apparent when the entire image was viewed.

The third set of images is of a water scene, gathered by the Worldview-2 sensor across 8 bands with 2.5 m GSD. The multispectral image is complexity analyzed and scaled using SVE, and then the corresponding panchromatic image is used to observe the cued area of interest under higher resolution. This is shown in Figure 8.9. Upon first glance, the image

(a)



(b)

Figure 8.1: Complexity analysis of the Cooke City, MT image.

Figure 8.2: $4k \times 2k$ image of an area in Esperanza, CA. The image was processed at 4 bands with $\sim 2.5m$ GSD. What are the areas of interest? Will SVE work on 4-band imagery?

appears to be rather complex, showing a variety of colors and what appears to be quite a bit of texture. Upon analysis, however, nearly all of the image is blacked out. Only three pixels are brightened, indicating that those are the only areas containing anything that is more complex than the rest of the image (*i.e.*, water). When the panchromatic image is viewed, we can see that there are in fact two boats in the water. Although the larger

boat might be visible in the original RGB, the second boat is not obvious. This shows how SVE operates on relative complexity, as the water was not cued as being anything interesting, even though it does have a spectral signature. This is important as it adds to the potential utility of SVE.

The last set of complexity-analyzed images is shown in Figures 8.10 - 8.13. These images were taken in Afghanistan using the RapidEye sensor, which is multispectral at 5 bands, and has approximately 5 m GSD. This 5k $\times$ 5k pixel main image contains a variety of settings, including a city, an airport, desert, and sparse towns. Analysis of the city and airport scenes in Figure 8.11 and Figure 8.12, respectively, show how the structural elements are cued as being more complex. Upon close inspection of the complexity-scaled original image, two lone bright pixels on the upper-right-hand side is somewhat discernable, while the entire area surrounding those two pixels is black. When viewed in RGB and zoomed in, as shown in Figure 8.13, it is clear that those pixels correspond to a remote desert site. Again, this shows how SVE has the ability to identify pixels corresponding to areas of higher materialistic complexity.

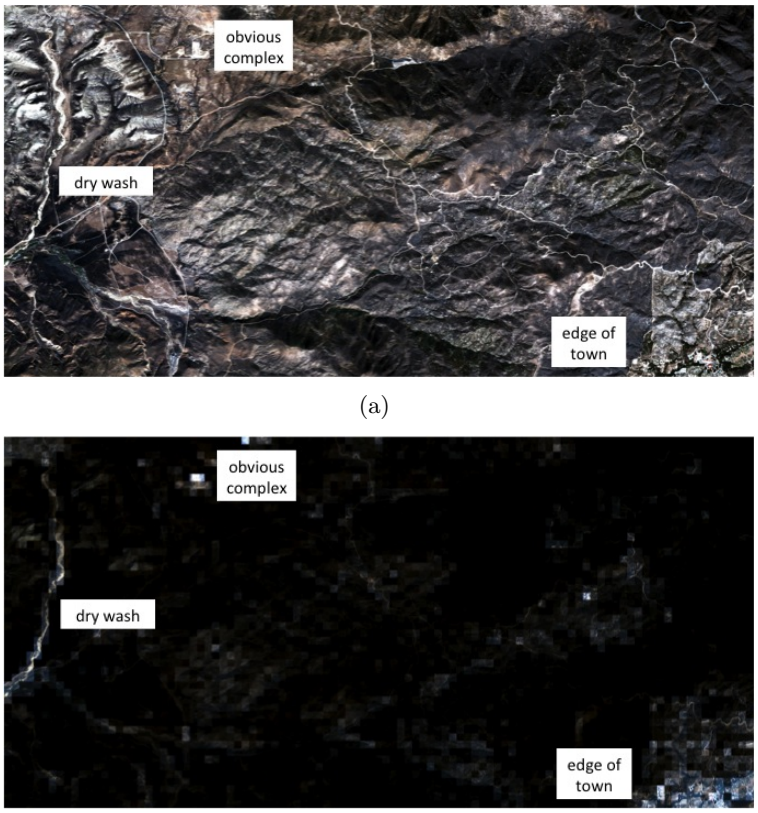
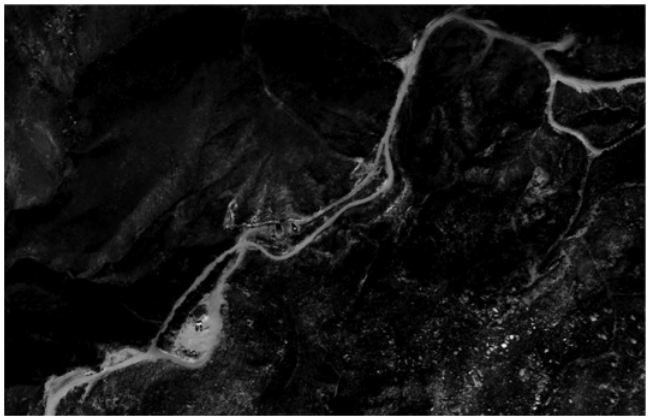


Figure 8.3: Comparison of obvious areas of complexity in the Esperanza image with cued areas of complexity when processed by SVE. (a) RGB with labeled areas observed to have higher complexity. (b) Tiled and complexity-scaled image.

Figure 8.4: Tiled and complexity-scaled Esperanza image. The observable areas of higher complexity were cued (as indicated by the brighter tiles near the labeled areas), but there is also a cluster of several bright tiles on the right side of the image. This indicates potential areas of interest that were not obvious.

(a)

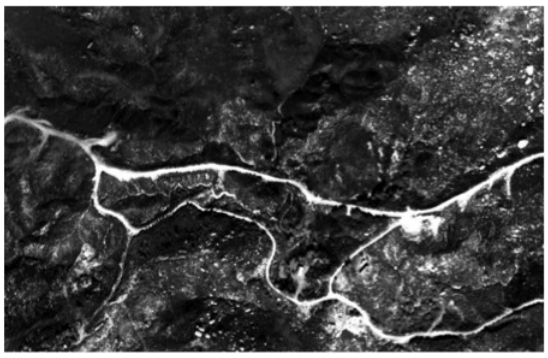


(b)

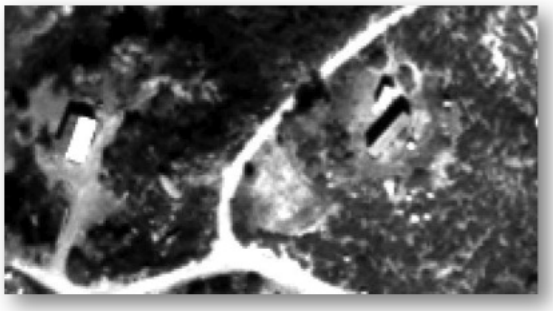
(c)

Figure 8.5: Esperanza image with higher-resolution analysis of the far-left cued tile of interest (i.e., tile of higher complexity). (a) Complexity-scaled RGB. (b) Cued tile in higher resolution. (c) Zoomed in cued tile which shows the presence of roads and building structures.

(a)



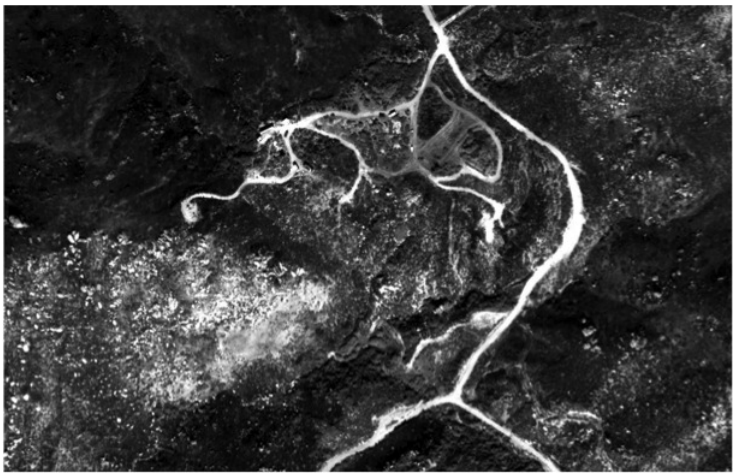
(b)



(c)

Figure 8.6: Esperanza image with higher-resolution analysis of the 2nd-from-left cued tile of interest (i.e., tile of higher complexity). (a) Complexity-scaled RGB. (b) Cued tile in higher resolution. (c) Zoomed in cued tile which shows the presence of roads and building structures.

(a)

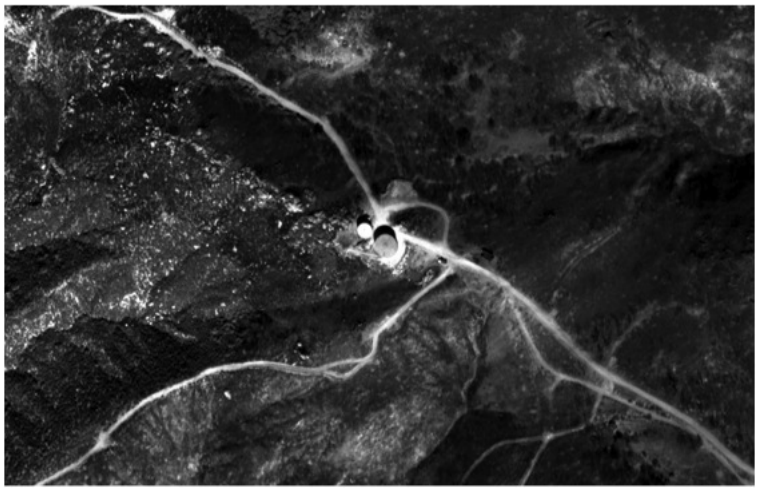


(b)

(c)

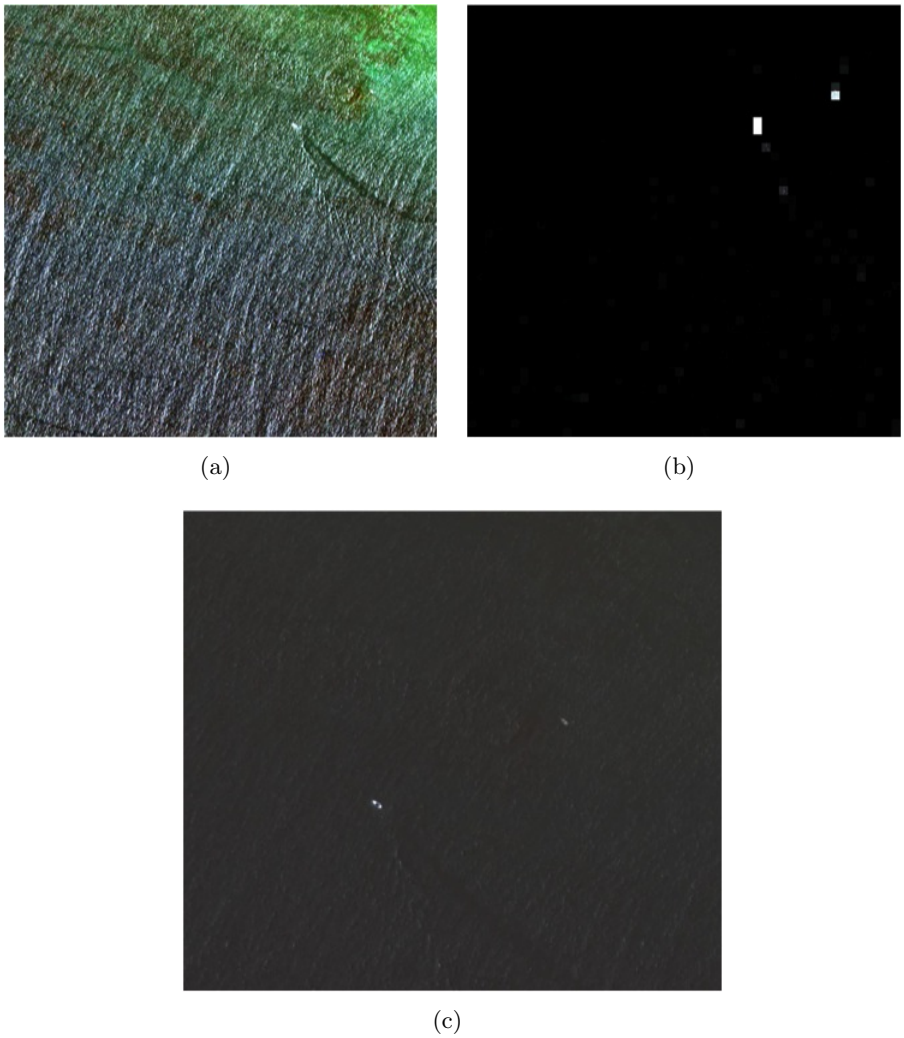
Figure 8.7: Esperanza image with higher-resolution analysis of the 2nd-from-right cued tile of interest (i.e., tile of higher complexity). (a) Complexity-scaled RGB. (b) Cued tile in higher resolution. (c) Zoomed in cued tile which shows the presence of roads and building structures.

(a)



(b)

Figure 8.8: Esperanza image with higher-resolution analysis of the far-right cued tile of interest (i.e., tile of higher complexity). (a) Complexity-scaled RGB. (b) Cued tile in higher resolution showing the presence of roads and water towers.

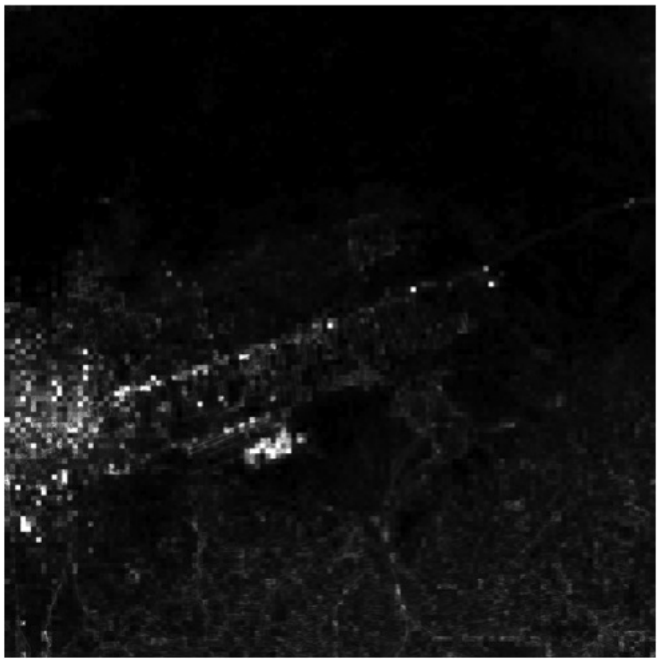


(a) (b)

(c)

Figure 8.9: Complexity analysis of an 8-band image of water processed by Worldview-2. (a) RGB of water scene with one boat visible towards the upper-right corner. (b) Complexity-scaled RGB. (c) Zoomed in cued area in grayscale, showing the presence of a second boat that is not obvious in the initial image.

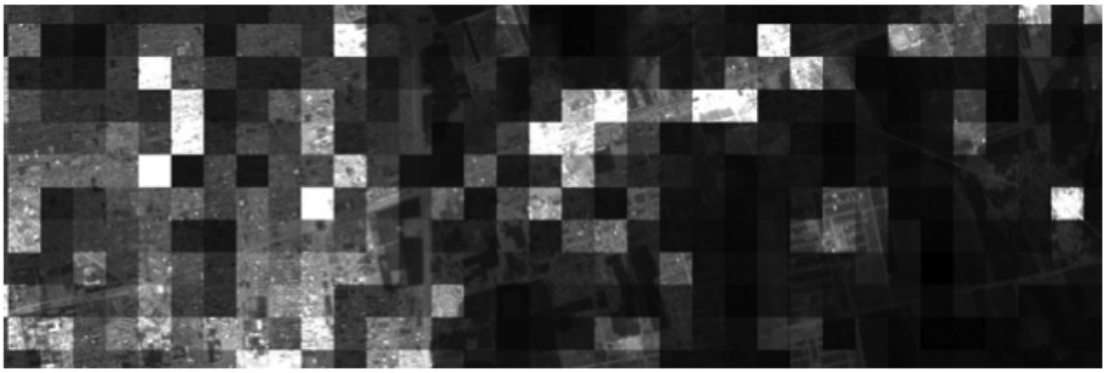
(a)



(b)

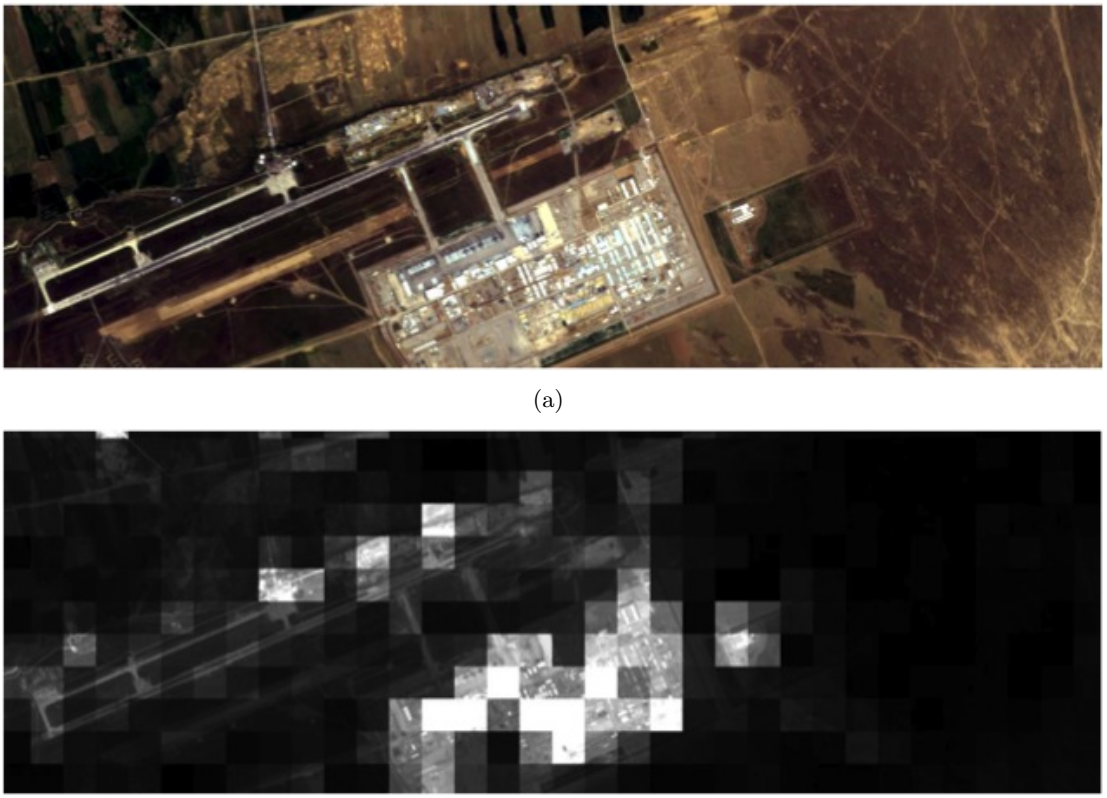
Figure 8.10: Complexity analysis of a 5-band image in Afghanistan processed by the RapidEye satellite. The image is $5k \times 5k$ pixels. (a) RGB of image. (b) Complexity-scaled RGB.

(a)



(b)

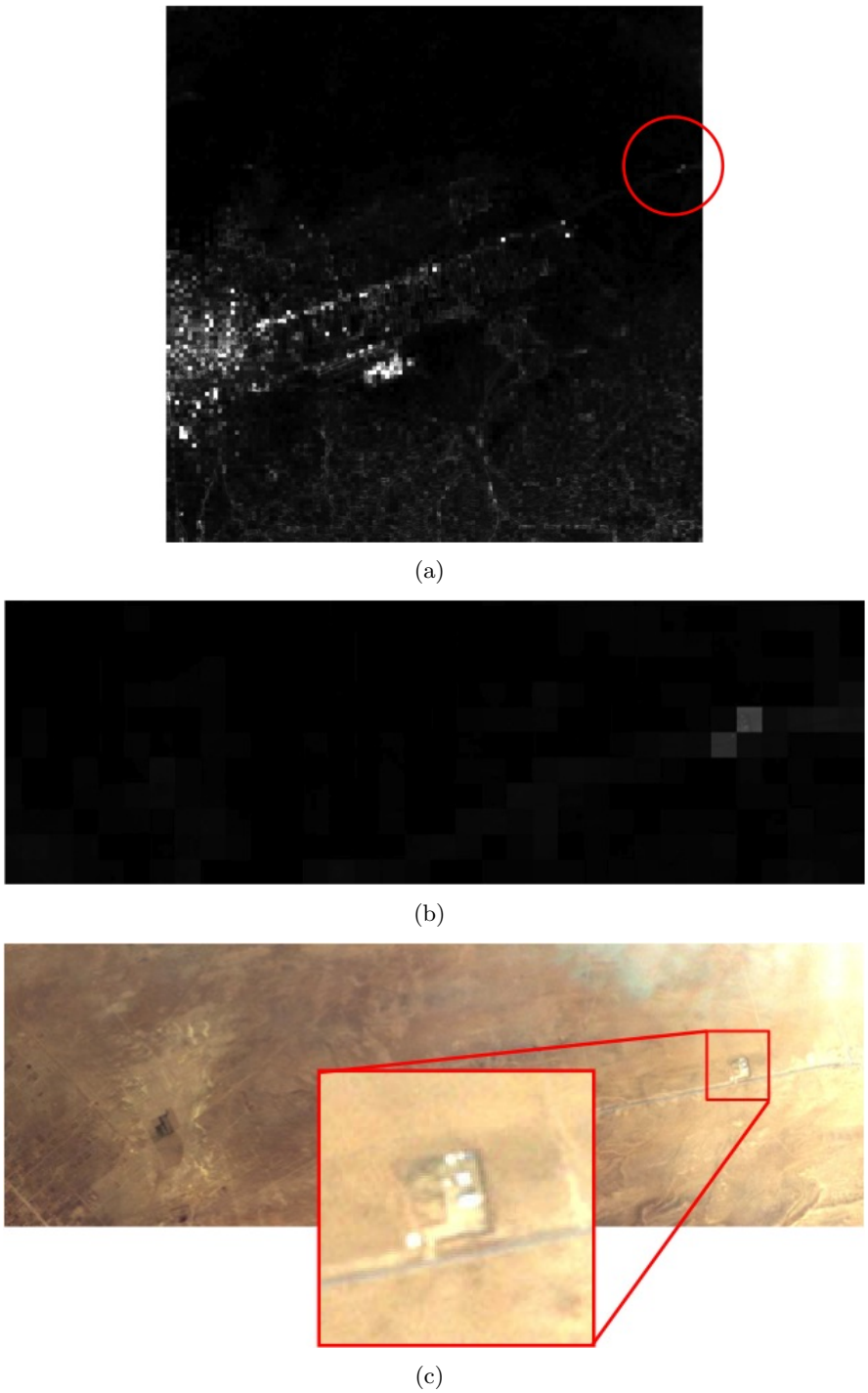
Figure 8.11: Complexity analysis of the city area in the RapidEye Afghanistan image. (a) RGB of the area. (b) Complexity-scaled RGB.



(a)

(b)

Figure 8.12: Complexity analysis of the airport area in the RapidEye Afghanistan image. (a) RGB of the area. (b) Complexity-scaled RGB.



(a)

(b)

(c)

Figure 8.13: Higher-resolution analysis of a cued point far out to the right in the RapidEye Afghanistan image. (a) Complexity-scaled RGB of the image indicating the cued area of interest. (b) Zoomed in cued area, showing that it is the only complex region in this part of the image. (c) Zoomed in RGB of the area showing the presence of a remote desert site.

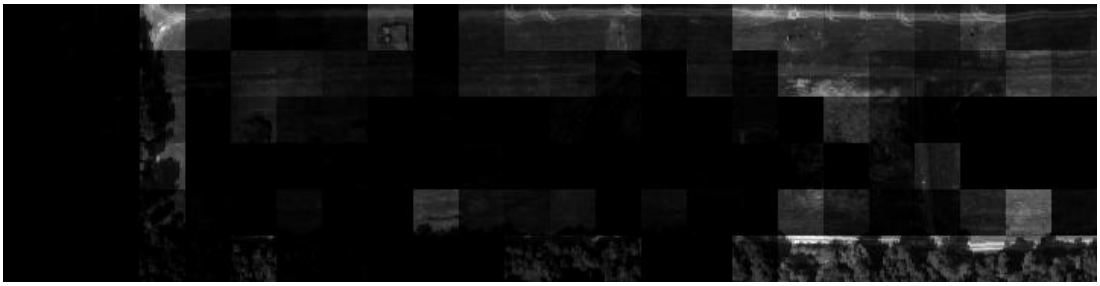
8.2 Change Detection Results

The results from processing the two HYDICE images are presented in Figure 8.14. The collection of images corresponding to a changed scene are shown in parts (a) and (b) of the figure, and the tiled and brightness-scaled result image is shown in Figure 8.14(c). There are several observations to be made about this result. Note that the content in the middle of the image did not change and was correctly identified as such, while the changes along the treeline registered as a small, yet detectable change. Due to the movement of targets as well as the movement of vehicles along the road, the lower right portion of the image registered the largest amount of changes. The changes registered along the left edge of the image can be attributed to the presence of significant shadows in one of the images.

Results from application of the algorithm to the HyMAP imagery are shown in Figure 8.15. Here, we see that the areas in the forest (which occur in the lower right of the image) are significantly darkened in the tiled and brightness-scaled result. This is to be expected as it is a dense forest that is not characterized by high levels of activity. Contrastingly, the most likely changes by far–as indicated by the brightest tiles–are identified as being present in the town as well as the circular field to the right of the town. The circular field contained known changes, and because the town is not a static entity, it is also expected to be filled with detectable changes.

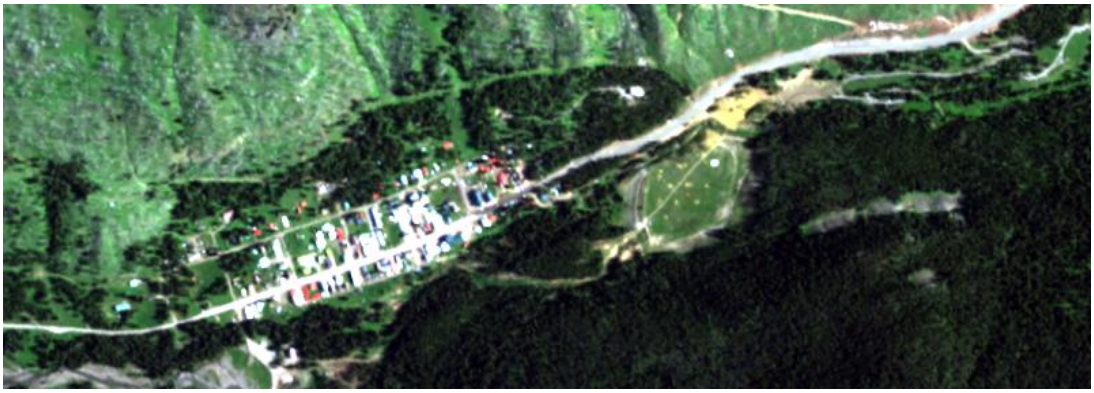
(a)

(b)



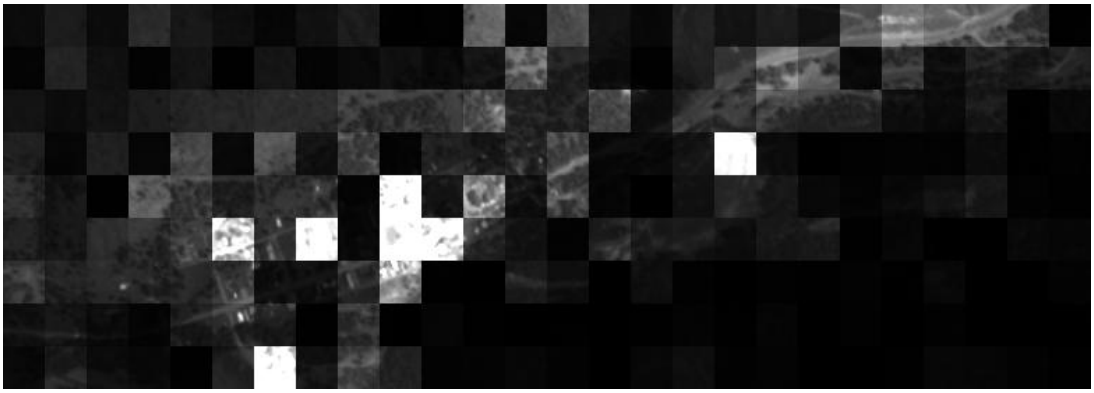
(c)

Figure 8.14: Results from application of the change detection algorithm to the HYDICE imagery.[10] (a) Image without most of the targets present. (b) Image with targets present. (c) Tiled resultant image. Brighter tiles indicate regions that are more likely to have changed.



(a)

(b)



(c)

Figure 8.15: Results from application of the change detection algorithm to the HYMAP imagery.[10] (a) & (b) Images separated by approximately 1 hour with known changes. (c) Tiled resultant image. Brighter tiles indicate regions that are more likely to have changed. Note the prominent change in the circular field to the right of the town.

Chapter 9

Conclusions

A geometric approach to multi and hyperspectral image analysis is desirable, as it does not require that assumptions be made about the distribution of the data. We have an increasing ability to collect images at higher spatial resolutions, and with that comes an increasing inability to make general statistical assumptions about the distribution of the collected data. While those assumptions may certainly be made for data at particular resolutions, they will drastically narrow the scope of application. Because an algorithm's widespread application is critical to its utility, it is not preferable that it require assumptions to be made about the data. This is why an algorithm based upon the geometry of the data is highly advantageous.

In the Simplex Volume Estimation algorithm, the volume of the parallelotope approximated by regions throughout the processed image(s) is exploited to obtain three possible metrics: the peak of the volume function, the area under the curve, and the integer value for the number of endmembers at which the graph went to zero. While only two, the peak and the area, were found to be sensitive enough to changes within an image to be candidates for the metric used, the metric chosen was the peak. It is the easiest value to compute and is also the easiest to observe when different functions are compared.

From the results in Chapter 8, we can see that SVE is a viable tool for change detection and large area search (based on complexity) in both multi and hyperspectral images. Its sensitivity even with 4-band or 8-band images is of particular significance, because there are several government satellites in orbit that process images at those bands. Because SVE has a widespread application at many different spectral and spatial resolutions, it can be used to analyze a multitude of data. This makes SVE an algorithm with extremely high utility. For both change detection and complexity measurements, SVE cues an analyst to areas of interest, where "interest" means areas with changes or areas of higher complexity,

respectively. By significantly scaling down the number of pixels that an analyst should examine, SVE drastically cuts down the time needed to analyze an image.

Chapter 10

Future Work

In multi and hyperspectral imaging, knowing the inherent dimensionality of a dataset is incredibly desirable. With that knowledge, the correct number of endmembers to define the corner of the simplex of data may be obtained. By identifying and obtaining the proper number of endmembers, every member of the dataset can be written as a linear combination of those endmembers. This significantly cuts back on memory and processing time of the set, which is extremely beneficial. However, there currently is no method for accurately determining the proper number of endmembers to use.

As mentioned in Section 4.5, the integer value for the number of endmembers at which the Gram volume function goes to zero is mathematically related to the inherent dimensionality of the data. When the Gram determinant of a set of vectors (in this case, pixels) is zero, that means the set is no longer linearly independent (as proven in Theorem 4). In other words, the number of endmembers that Max-D was asked to compute exceeded the inherent dimensionality of the data. By locating the number of endmembers at which the function effectively first goes to zero (because there is a small bit of noise due to the sensor), we can easily step back by one endmember to find what–mathematically–is the inherent dimensionality of the set. The ability to identify this value is due to the iterative nature of the algorithm. If this does prove, in practical applications, to be indicative of the desired number of endmembers, then it is highly important. This value could be tested against several datasets and the results when used to perform Linear Unmixing.

Bibliography

[1] D. Manolakis, D. Marden, and G. Shaw, "Hyperspectral image processing for automatic target detection applications," *Lincoln Laboratory Journal*, vol. 14, no. 1, pp. 79 – 116, 2003.

[2] D. Gillis, J. Bowles, E. Ientilucci, and D. Messinger, "A generalized linear mixing model for hyperspectral imagery," in *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XIV* (S. Shen, ed.), vol. 6966, SPIE, 2008.

[3] J. Schott, K. Lee, R. Raqueño, G. Hoffman, and G. Healey, "A subpixel target detection technique based on the invariant approach," in *Proceedings of the AVIRIS Workshop*, NASA JPL, 2003.

[4] D. W. Messinger, A. Ziemann, B. Basener, and A. Schlamm, "A metric of spectral image complexity with application to large area search," *to be submitted to the Journal of Applied Remote Sensing*, 2010.

[5] M. Carlotto, "Detection and analysis of change in remotely sensed imagery with applciation to wide area surveillance," *IEEE Transactions on Image Processing*, vol. 6, pp. 189–202, January 1997.

[6] I. S. Reed and X. Yu, "Adaptive multiple-band cfar detection of an optical pattern with unknown spectral distribution," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, pp. 1760–1770, 1990.

[7] G. A. Jennings, *Modern Geometry with Applications*. Springer, 1994.

[8] A. Kostrikin and Y. Manin, *Linear Algebra and Geometry*. Gordon and Breach Science Publishers, 1997.

[9] N. F. Dupuis, *Elements of Synthetic Solid Geometry*. MacMillan and Co., 1893.

[10] A. Ziemann, D. Messinger, and B. Basener, "Iterative convex hull volume estimation in hyperspectral imagery for change detection," in *Algorithms and Technologies for*

Multispectral, Hyperspectral, and Ultraspectral Imagery XVII (S. Shen, ed.), vol. 7695-54, SPIE, 2010.

[11] D. Poole, *Linear Algebra: A Modern Introduction*. Thomson Brooks/Cole, 2nd ed., 2006.

[12] L. Mirsky, *An Introduction to Linear Algebra*. Dover Publications, Inc., 1990.

[13] J. Schott, *Remote Sensing: The Image Chain Approach*. Oxford University Press, 1997.

[14] A. Schlamm, *Detection of Man-Made Material in Large Area Search Using Hyperspectral Imagery*. PhD thesis, Rochester Institute of Technology, 2010.

[15] D. Snyder, J. Kerekes, I. Fairweather, R. Crabtree, J. Shive, and S. Hager, "Development of a web-based application to evaluate target finding algorithms," in *Proceedings of IGARSS 2008*, IEEE, 2008.