Surface Prior Information Reflectance Estimation (SPIRE) Algorithms

by

Herbert Erik Mattias Viggh

S.B., Aeronautics and Astronautics, Massachusetts Institute of Technology, 1985
M.S., Aeronautics and Astronautics, Massachusetts Institute of Technology, 1988
M.S., Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 1988

SUBMITTED IN PARTIAL FULLFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

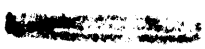
DOCTOR OF PHILOSOPHY
IN ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

August, 2001
[September 2001]

Signature of Author: _____
Department of Electrical Engineering and Computer Science
August 29, 2001

Certified by: _____
David H. Staelin
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by: _____
Arthur C. Smith
Professor of Electrical Engineering and Computer Science
Chairman, Committee on Graduate Students



Surface Prior Information Reflectance Estimation (SPIRE) Algorithms

by

Herbert Erik Mattias Viggh

Submitted to the Department of Electrical Engineering and Computer Science on August 30, 2001, in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Electrical Engineering and Computer Science.

ABSTRACT

In this thesis we address the problem of estimating changes in surface reflectance in hyperspectral image cubes, under unknown multiplicative and additive illumination noise. Rather than using the Empirical Line Method (ELM) or physics-based approaches, we assumed the presence of a prior reflectance image cube and ensembles of typical multiplicative and additive illumination noise vectors, and developed algorithms which estimate reflectance using this prior information. These algorithms were developed under the additional assumptions that the illumination effects were band limited to lower spatial frequencies and that the differences in the surface reflectance from the prior were small in area relative to the scene, and have defined edges. These new algorithms were named Surface Prior Information Reflectance Estimation (SPIRE) algorithms.

Spatial SPIRE algorithms that employ spatial processing were developed for six cases defined by the presence or absence of the additive noise, and by whether or not the noise signals are spatially uniform or varying. These algorithms use high-pass spatial filtering to remove the noise effects. Spectral SPIRE algorithms that employ spectral processing were developed and use zero-padded Principal Components (PC) filtering to remove the illumination noise. Combined SPIRE algorithms that use both spatial and spectral processing were also developed. A Selective SPIRE technique that chooses between Combined and Spectral SPIRE reflectance estimates was developed; it maximizes estimation performance on both modified and unmodified pixels.

The different SPIRE algorithms were tested on HYDICE airborne sensor hyperspectral data, and their reflectance estimates were compared to those from the physics-based ATmospheric REMoval (ATREM) and the Empirical Line Method atmospheric compensation algorithms. SPIRE algorithm performance was found to be nearly identical to the ELM ground-truth based results. SPIRE algorithms performed better than ATREM overall, and significantly better under high clouds and haze. Minimum-distance classification experiments demonstrated SPIRE's superior performance over both ATREM and ELM in cross-image supervised classification applications. The taxonomy of SPIRE algorithms was presented and suggestions were made concerning which SPIRE algorithm is recommended for various applications.

Thesis Supervisor: David H. Staelin
Title: Professor of Electrical Engineering

ACKNOWLEDGEMENTS

I wish to express my deepest gratitude to my thesis supervisor Professor Staelin. He guided and advised me throughout my whole doctoral program, including applying for readmission, preparing for my qualifier and general exams, selecting a thesis topic, performing the research, and navigating the intersecting mazes of requirements for the new doctoral degree program and the Lincoln Scholars Program. He taught me to ask the right questions, avoid ad hoc solutions, and to always make the figures first. He made himself available to meet every week, and was always quick to provide feedback on drafts of papers and thesis chapters. I could not have asked more from a thesis supervisor.

Next, I must acknowledge the financial support of MIT Lincoln Laboratory through the Lincoln Scholars Program. As a research engineer with four children, a mortgage, etc., returning to MIT for a doctorate as a regular graduate student would have involved nearly insurmountable financial obstacles that I avoided with the Lincoln Scholars Program. My summer employment at Lincoln was supported by the Hyperspectral Technology Assessment Program (HTAP), an on going project at MIT Lincoln laboratory sponsored by the Deputy Undersecretary of Defense (Science and Technology).

John Kerekes served as both my Lincoln mentor and as a member of my doctoral committee. He provided me with a wealth of information on remote sensing, the current state of the art, and papers and theses of interest. His help in obtaining the HYDICE data and directing me to other Lincoln staff for assistance and information was invaluable. John read several drafts of my thesis and related papers and always provided valuable suggestions and feedback. Professors Willsky and Tarokh also served on my committee and took the time to read my thesis and provide valuable feedback. They both deserve extra credit for agreeing to do so even though my committee could not be formed until most of the research had already been completed.

Even though I spent most of my time working at Lincoln Laboratory in Bedford, the RLE Remote Sensing group on campus welcomed me and made me feel at home. Bill Blackwell, Vince Leslie, Junhee Lee, Phil Rosenkranz, Fred Chen, and Jay Hancock listened to my ideas, provided suggestions, and patiently explained what their own research was about.

At Lincoln, many people helped me in my day to day research with advice and assistance. Mike Griffin taught me how to run ATREM and MODTRAN. Craig Richard checked my ATREM runs with the AFRL/MODTRAN code. Carolyn Upham provided me with all of the HYDICE data and documentation. My officemate Scott Stuart and fellow Lincoln Scholar was my main IDL and perl consultant while Dan Payne and Tony Ruscitti kept the computers running and backed up. Too many others to list showed interest in my progress, expressed encouragement, and gave advice.

My Lincoln management was also extremely supportive of my doctoral program. My group leader Gerry Banner agreed to pay for my program and provided me with office space and computing resources. Hsiao-hua Burke provided me with summer support, advice, and was a friendly face on the Lincoln Scholars Committee. Grant Stokes provided encouragement, advice, and helped with several administrative issues.

My parents moved our family to this country so that their children would have a better future, leaving behind all their family and friends. If they had not done so, I doubt that I would ever have made it to MIT. During this doctoral program they provided help and support in innumerable ways. Without their financial assistance I would not have been able to complete the

program. My father helped me with problem sets and preparing for exams, while my mother babysat the grandchildren so that I could study and their mom could work.

My four children Steven, Carolyn, Annie, and Erik were patient and understanding about the many adjustments that were needed while Daddy went back to school. Talking and playing with them before bed time each evening were the best stress reducers possible.

Finally, I gratefully acknowledge my wonderful wife Julia, the love of my life and my best friend. Without her support and encouragement I would not have returned for my doctorate and this thesis would never have been written. Though she always felt as if she was not helping enough, she did more to help me complete this work than anyone else. She worked a full time job at home with the kids, a part time job to make ends meet, and was still always there to listen to my ideas, review my results, and dream about what we would do once I was done. Now it is time to enjoy that wonderful life.

Perge!

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 A Simple Experiment

The research in this thesis was inspired by a simple experiment first done over thirty years ago (Land, 1964). Figure 1.1 depicts this experiment. Take adjacent red and blue pieces of paper and illuminate them with a controlled white light. A human observer viewing both pieces of paper simultaneously will see the red paper as red and the blue paper as blue. Measure the optical radiation coming from the red paper in the short, middle, and long wavelength channels of the human eye color receptors (Kaiser and Boynton, 1996, Spillman and Werner, 1990, Dowling, 1987). Next, adjust the source illumination so that the reflected light from the blue paper in these three channels becomes identical to what was previously reflected by the red paper under the white light. Now ask the human observer what color the two pieces of paper have. The observer will still see the blue paper as blue and the red paper as red, despite the fact that the blue paper is now radiating the exact same three channel spectrum which the red paper had radiated previously.

These surprising results (at the time), spurred the development of human vision color perception models and algorithms to emulate them. These "color constancy" algorithms all reduce to a high-pass spatial filtering operation in each wavelength band, that eliminates low spatial frequencies where the illumination effects are most pronounced, while retaining the higher spatial frequency content of the surface edges in the scene (Hurlbert, 1986, Horn, 1986). While low spatial frequencies of the surface are lost, the mean of the surface reflectance is restored from a global average mean of all possible scenes, assumed to be built into humans by

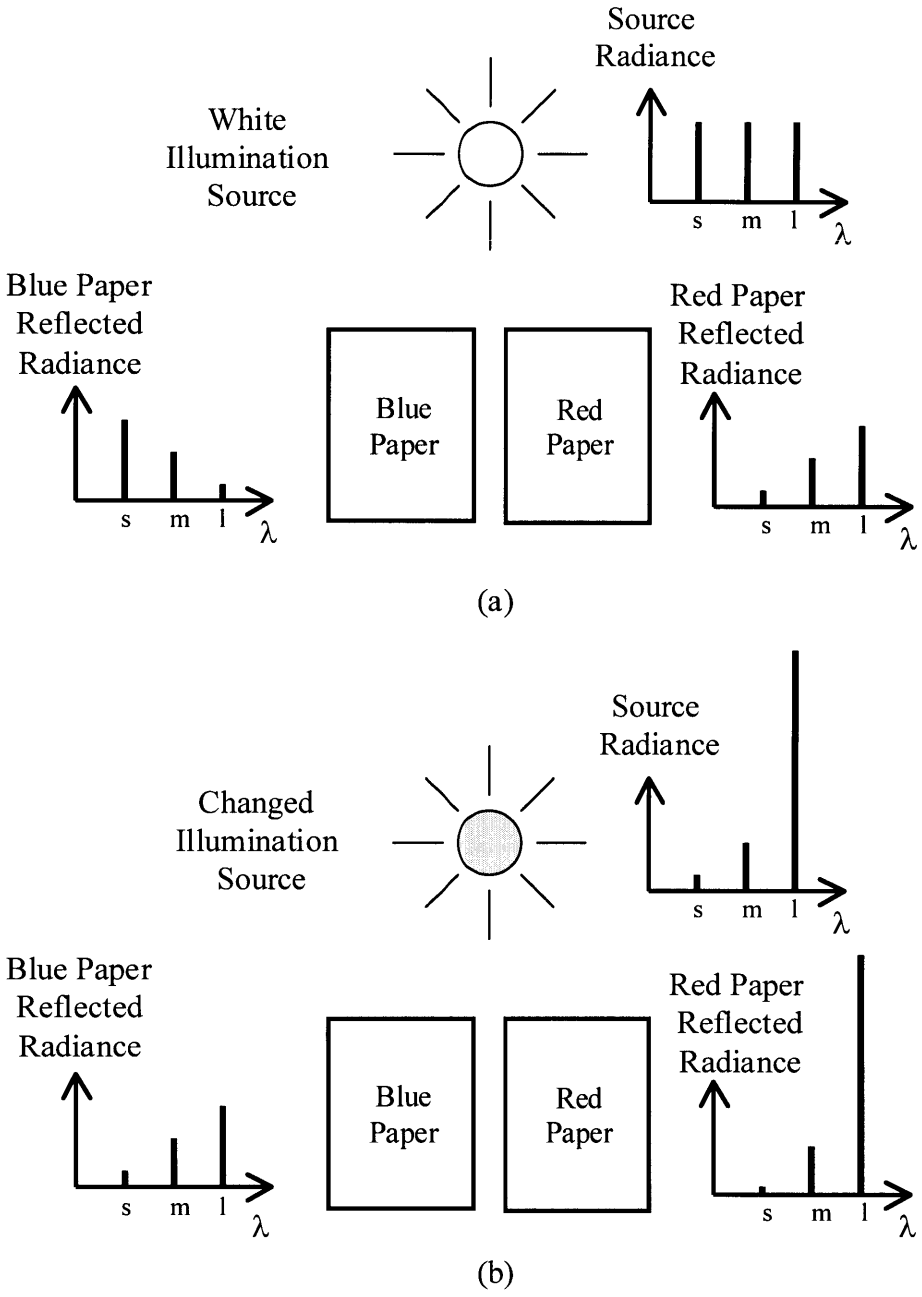


Figure 1.1. *A simple experiment in human color constancy. In (a), a white illumination source irradiates adjacent blue and red pieces of paper in the three human eye spectral channels at short (s), medium (m), and long (l) wavelengths (λ). The reflected radiances seen by the human eye from each piece of paper are depicted next to them. In (b), the source illumination has been changed so that the reflected radiance from the blue paper is the same as that of the red paper in (a). In both (a) and (b), a human observer sees the blue paper as having the color blue, and the red paper as having the color red.*

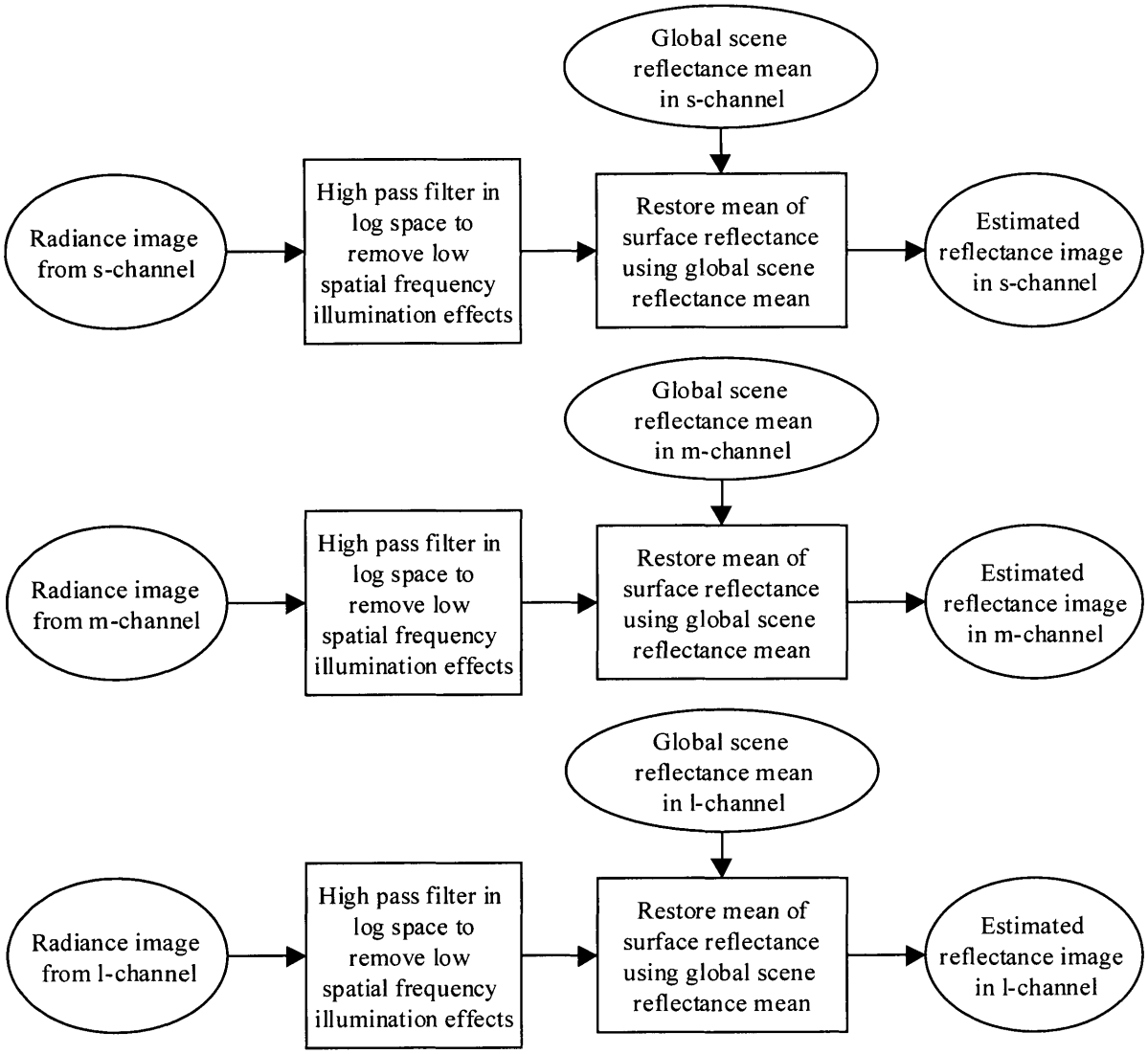


Figure 1.2. *Block diagram of lightness algorithm emulating human color constancy processing. Each of the three short (s), medium (m), and long (l) spectral wavelength channels are processed independently to estimate reflectance. In each channel, the radiance image is spatially high-pass filtered in log space to remove illumination effects which tend to be confined to low spatial frequencies. To restore at least some of the reflectance low frequency information lost in the filtering, the mean (zero spatial frequency) is restored using a global mean for all scenes. At each point in the image, the three channel reflectance values can be mapped to a color space.*

evolution. This is depicted in Figure 1.2. At each point in the scene, the resulting three reflectance values can be mapped to a three dimensional color space to identify the color of each material in the scene.

Such color constancy algorithms have historically been referred to as "lightness" algorithms since they calculate the lightness and darkness of materials relative to each other in each channel. Physically, this lightness is determined by the reflectance, with high reflectance corresponding to high lightness. Intuitively we can see how such algorithms work in Figure 1.1. In both (a) and (b), blue is brighter, or lighter than red in the s-channel, blue and red are equal in the m-channel, and blue is darker than red in the l-channel. These relative lightness ratios are used to identify the colors, independent of the spectral content of the source illumination.

Today, sensors exist which collect images in far more spectral channels than the three of the human eye. These multispectral and hyperspectral sensors are used in many important remote sensing applications in which the surface spectral reflectance must be estimated. Currently, the utility of such sensors is limited because varying illumination conditions, analogous to those in the simple experiment described above, make it difficult to estimate surface reflectance. Color, as perceived by humans, is essentially a three-channel measure of surface spectral reflectance. Therefore, one can pose the question:

Can the principles embodied by color constancy algorithms be used in remote sensing applications to estimate surface reflectance?

We note that color constancy algorithms employ prior knowledge to restore information about the surface reflectance that is lost in the spatial filtering. As more and more spacebased

and airborne sensor systems are deployed in remote sensing applications, we are approaching a time when all parts of the Earth's surface will be routinely imaged. In such a situation, the problem becomes one of detecting and estimating changes in reflectance, with prior information available in the form of previous images and estimates of reflectance based on them. Therefore, a second question can be posed:

How can we best make use of prior estimates of surface reflectance to estimate changes in the current reflectance of a newly sensed image?

Given that the human visual system has only three spectral channels, it is not surprising that our color constancy processing relies on spatially filtering each spectral channel image independently of the others. However, since modern sensors employ hundreds or even thousands of spectral channels, this leads to a third question:

How can we make use of the information in the spectral dimension, along with that in the spatial dimension, to filter out the illumination effects and estimate reflectance?

We note that estimating changes in reflectance in a current image with respect to a prior reflectance image is equivalent to estimating the reflectance of the entire new image and then comparing it to the prior reflectance. Therefore, these three questions can be combined into one central question to be addressed by this thesis:

How can we best estimate the surface reflectance of an image using spatial processing,

spectral processing, and a prior estimate of the surface reflectance?

In the remainder of this introductory chapter, we will briefly review remote sensing applications and sensors, and describe the specific research goals and organization of this thesis.

1.2 Estimation of Surface Spectral Reflectance in Sensed Images

A fundamental problem in many imaging and remote sensing problems is the estimation of the surface reflectance of objects and materials in an imaged scene. The scene radiance measured by a sensor is generated by complex physical interactions between the source illumination, material surfaces, and the intervening atmosphere. The solar angle and often unpredictable atmospheric conditions introduce multiplicative and additive noise effects on top of the reflectance signal of interest. If theses noise effects were always static, then it would be a simple matter to measure them once and use them to solve for the reflectance in all future images. Unfortunately, the temporally varying state of the atmosphere due to changing weather causes these noise effects to vary unpredictably. These illumination effects must therefore be compensated and removed in order to estimate the surface reflectance of the scene.

In remote sensing applications, this problem area is usually referred to as atmospheric compensation ("atmospheric correction" is also used in the literature), since most unpredictable effects are caused by the intervening atmosphere. In other applications, such as mobile robotics, the unpredictable variability of the illumination conditions can be caused by unknown changes in the source illumination intensity, spectral content, or geometry. In applications where temporal changes in surface reflectance are to be identified and measured, one must separate changes in the sensed image caused by varying multiplicative and additive (e.g. due to scattering) illumination effects, from those caused by changes in surface reflectance. Also, the scene

reflectance and illumination can vary both spatially across the scene and across the spectral channels of the sensor.

1.3 Remote Sensing

Remote sensing covers a broad range of applications as well as a broad range of the electromagnetic spectrum from visible to microwave bands, including thermal infrared (Schowengerdt, 1997; Richards, 1993; Lillesand and Kiefer, 1994, Schott, 1997). Remote sensing applications include:

- Monitoring and assessing changes in the Earth's environment such as global warming, ozone depletion, land use, urban growth, etc.
- Monitoring and assessing agricultural crops
- Exploration for non-renewable resources such as oil, minerals, etc.
- Management of renewable natural resources such as forests and wetlands
- Meteorology and weather forecasting
- Military surveillance and reconnaissance

There is often a distinction made between passive and active remote sensing. Passive remote sensing is usually defined as using a sensor that measures natural radiation reflected or emitted by the scene. In active remote sensing, pulsed radar or laser sources are used by the sensor, and the reflected or scattered radiation measured. There are also applications where a steady artificial light source is used by the sensor. Such sensors technically must be considered active sensors since the sensor controls the light source, but the data processing problem is nearly identical to that of passive remote sensing. In this thesis we will consider remote sensing and imaging applications where the sensor measures reflected illumination from a scene, where the

illumination can be either natural or artificial and controlled by the sensor or not, but is steady during the collection of the image, and not pulsed as in radar applications.

This thesis will deal with applications that utilize the visible through short wave infrared (SWIR) wavelengths of 0.4-2.5 μm, where the radiation reaching the sensor from the surface is due mostly to reflected source illumination, as opposed to surface thermal emissions. In most of these applications change detection is of crucial importance. The detection of changes requires the accurate estimation of the surface reflectance over the entire scene so that changes can be detected in comparison to a prior reflectance image. Accurate estimation of the spectral reflectance of the changes is needed to correctly measure, identify, and classify them in downstream processing and analysis.

1.3.1 Airborne Remote Sensors

In airborne sensing applications, a sensor is carried by an aircraft to some altitude above the terrain scene to be imaged. This sensed image can then be processed to estimate surface reflectance and provide knowledge of the materials on the ground. Sensors which include channels covering the atmospheric absorption bands can use these data to extract information about the state of the atmosphere between the sensor and the scene. In airborne remote sensing there is the complication that no information is typically available about the state of the atmosphere above the sensor. This can adversely affect solution approaches that attempt to estimate the state of the atmosphere using information extracted from the sensed image and radiative transfer physics (known as physics-based approaches), or at least restrict them to operating under conditions where the atmospheric state above the sensor is within certain bounds. Existing physics-based techniques do not work well in operational scenarios where an airborne sensor is flown below clouds or haze.

The airborne remote sensing application is therefore the most general and difficult application area and will be the specific problem area of this research.

1.3.2 Satellite Based Remote Sensing

This application is very similar to airborne remote sensing except that the sensor is mounted on a satellite or spacecraft in orbit around the Earth. Since all of the atmosphere is below the sensor, the problem of unknown atmospheric state above the sensor is not present. However, unknown changes in solar illumination at the top of the atmosphere due to small changes in solar output can introduce a similar problem.

1.4 Other Applications

1.4.1 Mobile Robotics

In mobile robot applications, machine vision systems need to identify objects for navigation, inspection, and task completion. Color, a three-channel measure of surface reflectance, is often used to help identify objects and surroundings. While mobile robot technologies are still mainly limited to research environments, envisioned applications involve operations under a variety of different illumination conditions. Many applications would involve operating indoors under artificial lighting which may change due to lights aging, being replaced, turned on and off, and failing. Others could involve outdoor operations under varying natural lighting conditions or at night with artificial illumination where the illumination spectral content or geometry is not known. Inspection and repair operations under hazardous conditions, such as inside a nuclear reactor or power plant after an accident, would involve unpredictable illumination conditions, possibly with smoke or suspended particulates in the air. Unmanned airborne vehicle applications flying at low altitude are similar to the airborne remote sensing

application. There are also many applications for robotic inspection systems in manufacturing or contamination detection in which the illumination conditions may not be well known.

1.4.2 Solar System Remote Sensing

This application includes fly-by and orbiting spaceprobes, and possibly air vehicles (on planets with atmospheres) that perform remote sensing over other planets, moons, or asteroids in the solar system. Mars is currently the focus of several such missions. The problems here are similar to those of airborne and space-based remote sensing of the Earth, but with a different atmosphere and solar illumination.

1.5 Research Goals

The goal of the research described in this thesis was to develop reflectance estimation algorithms that will be applicable to a broad set of problems. In the varied applications discussed above, the illumination effects are caused by different types of source illumination and intervening atmosphere which may be difficult to model. For example, shadows, aerosol scattering, and humidity variations have multiplicative effects while aerosol backscattering introduces additive spectral noise. All of these applications, however, have in common a surface with a spectral surface reflectance function that reflects incident radiation, and the possibility of having a prior estimate of the surface reflectance. In addition, controlled test observations can be used to generate ensembles of the multiplicative and additive effects encountered in a particular application, even if it is difficult to predict the illumination conditions for a particular image.

Therefore, this research concentrated on reflectance estimation algorithms that use prior information about the surface spectral reflectance and a general model of the multiplicative and additive noises that was not wed to any particular source illumination or atmospheric model. For

algorithms that use spectral processing, prior ensembles of multiplicative and additive illumination effects were also used. Since all of the algorithm variants developed rely on prior surface reflectance information, these algorithms were named Surface Prior Information Reflectance Estimation (SPIRE) algorithms. To advance the state of the art, the intent was to develop algorithms that will work under operational conditions in which current state-of-the-art algorithms fail.

1.6 Organization of Thesis

This thesis is organized into six main chapters beyond this introduction. Chapter 2 formulates the problem, describes the current state of the art, and introduces the taxonomy of SPIRE algorithms developed. Chapter 3 describes the development and testing of Spatial SPIRE algorithms which rely on spatial filtering to remove the multiplicative and additive noise effects. Chapter 4 discusses Principal Components analysis and the degrees of freedom in the multiplicative and additive noise ensembles of the test data set used, laying the foundation for the spectral filtering used in later chapters. Chapter 5 covers Spectral SPIRE algorithms that make use of spectral Principal Components filters to remove the noise effects. Chapter 6 covers Combined SPIRE algorithms that utilize both spatial and spectral filtering to take advantage of the strengths of each while overcoming their respective weaknesses. Chapter 7 reviews the SPIRE algorithm taxonomy and gives suggestions for which SPIRE algorithm to use for different applications. Chapter 7 also explores the use of SPIRE algorithms in a simple classification application and presents a Selective SPIRE technique that makes use of Spectral and Combined SPIRE reflectance estimates to achieve the best classification performance. Chapter 8 provides a summary, conclusions, and recommendations for further work. Various appendices supporting these chapters then follow.

Chapter 2

Problem Statement and State of the Art

In this chapter we define the problem to be solved and review the current state of the art. Our goal is to define the problem of reflectance estimation in a simple and generic manner that will be applicable to a large array of imaging applications, independent of the specific illumination and sensor scenarios. We will also discuss a simple classification application as an example of how the reflectance estimates could be used. Our review of the state of the art will be confined to airborne remote sensing since it is the example application explored here. We also present the SPIRE algorithm approach developed in this thesis and a taxonomy of the various SPIRE algorithms developed.

2.1 Image Formation Model

Figure 2.1 depicts a generic imaging scenario with an intervening atmosphere. A sensor is located some distance from the surface, with an optical system that focuses radiation from a portion of the surface onto a detector. Illumination sources radiate upon the scene, such as natural solar illumination or artificial lights. In the imaging scenarios considered in this thesis, all radiation components received at the sensor are due to the illumination sources, and not due to thermal radiation emitted from the surface itself.

The scene radiance measured by the sensor is generated by complex physical interactions between the source illumination, material surfaces, and the intervening atmosphere. Radiation from a source encounters absorption and scattering due to atmospheric molecules and aerosols along the atmospheric path to the surface. Radiation reflected by the surface also encounters absorption and scattering in its path to the sensor. Secondary illumination from scattered light

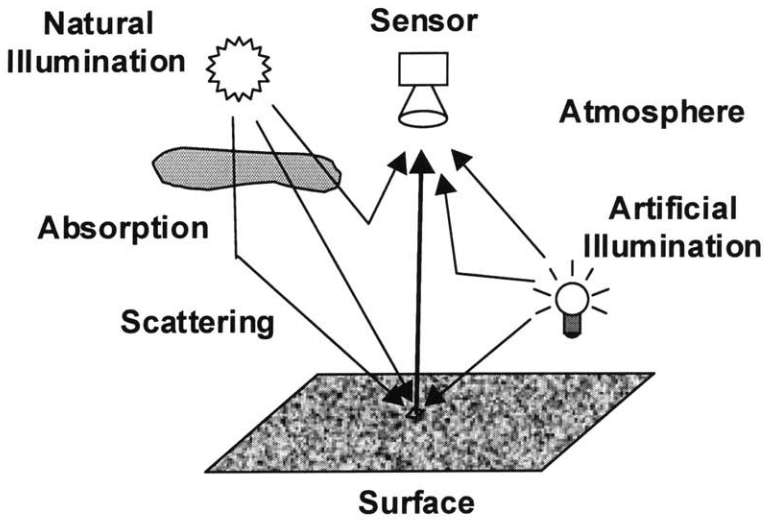


Figure 2.1. *Generic imaging scenario. Radiation from both natural and artificial illumination sources undergo absorption and scattering in the atmosphere. Radiation reaching the surface is reflected towards the sensor, and is affected by absorption and scattering as well. Upscattered radiation that never interacts with the surface also enters the sensor.*

that reaches the surface also encounters absorption and further scattering along its paths to the surface and the sensor. These interactions all have multiplicative effects on the sensed radiation image i generated by the sensor and can be collected into a single multiplicative effect m on the reflectance r.

Light that reaches the sensor without interacting with the surface causes additive effects on the radiation received at the sensor. This includes both radiation scattered into the sensor, as well as illumination from sources within the field of view of the sensor. Note that the additive radiation is also modified from its original source nature by absorption and scattering. Additive sensor noise also affects the received signal. All such additive effects at the sensor can be collected into a single additive effect a. Non-linear effects due to multiple reflections off of ground objects and terrain (Schott, 1997) and non-linear sensor effects are ignored.

In airborne remote sensing applications (Schowendgert, 1997), the source illumination is

the top-of-the-atmosphere solar irradiance which has a characteristic spectral shape of a black body radiator at a temperature of 5,900 degrees Kelvin at a distance of 1 AU. Various atmospheric absorption bands modify this source radiation throughout the visible and IR spectrum, due mainly to absorption by water vapor and carbon dioxide. Water vapor absorption can vary spatially across a scene based on the water conditions of the surface. Variable cloudiness across a scene can have an even larger effect on the spatial variation of the radiance reaching the surface. Up and down scatter are both caused by molecular Rayleigh scattering and aerosol and particulate Mie scattering. Mie scattering can vary across a scene, for example, between rural and urban areas. For an example derivation of how such atmospheric and illumination effects in an airborne remote sensing application can be reduced to a single multiplicative noise m and a single additive noise a, please see Appendix A.

Figure 2.2 represents a simplified, general image formation model consisting of a formation function F which operates on the reflectance r, multiplicative effect m, and additive effect a. The function F for a single sensor detector in the generic imaging scenario of Figure 2.1 is given by:

$$i = rm + a \tag{2.1}$$

Imaging sensors typically image a scene as a two-dimensional array of pixels, and often record multiple, hundreds, or thousands of spectral channels at each spatial pixel location. Therefore, each of the variables in (2.1) is a three dimensional array indexed by two spatial variables and one spectral variable. In this thesis we assume digitized data, and use n_x and n_y as

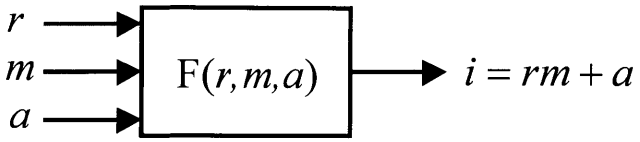


Figure 2.2. *General image formation model. The reflectance r, multiplicative noise m, and additive noise a are input to the model F, which then generates the radiance image i received at the sensor.*

the two discrete spatial variables, and n_λ as the discrete spectral variable. Therefore, at the single pixel array location $[n_{x_o}, n_{y_o}, n_{\lambda_o}]$ (2.1) becomes:

$$i[n_{x_o}, n_{y_o}, n_{\lambda_o}] = r[n_{x_o}, n_{y_o}, n_{\lambda_o}]m[n_{x_o}, n_{y_o}, n_{\lambda_o}] + a[n_{x_o}, n_{y_o}, n_{\lambda_o}] \tag{2.2}$$

All four quantities *i*, *m*, *r*, and *a* are therefore assumed to be three dimensional and can vary both spatially and spectrally. As such, they can be treated as three-dimensional vectors $\mathbf{i}$, $\mathbf{m}$, $\mathbf{r}$, and $\mathbf{a}$. Equation (2.2) then becomes:

$$\mathbf{i} = \mathbf{r} \odot \mathbf{m} + \mathbf{a} \tag{2.3}$$

Where the $\odot$ symbol represents the direct product (also known as the Hadamard product) operation of element-by-element multiplication between two vectors. For the purposes of this thesis, we will use the notation of (2.1) to represent the relationships expressed by (2.2) and (2.3). All italicized variables (*i*, *m*, *r*, and *a*) are assumed to be three dimensional, and products between them are assumed to be direct products. When vector concepts are required, the vector notation of (2.3) will be used.

Our problem, therefore, is to estimate *r*, given *i*. Modern hyperspectral sensors provide

both high spatial and high spectral resolution of r, m, and a, suggesting approaches that involve both spatial and spectral techniques for estimating r.

Two comments regarding nomenclature are required. The first is the use of the terms channel and band. A band is defined as a range of wavelengths or frequencies. A sensor channel will always cover some finite band of spectral frequencies, with a channel response function that weights the individual frequencies of the band. In much of the hyperspectral literature, such a sensor channel is often referred to as a sensor band. In this thesis, we will use the term sensor channel. The term band will be used to refer to ranges of frequencies covered by sensor channels.

The second nomenclature issue is the definition of the word noise. From a signal processing and estimation point of view, noise is most commonly thought of as unwanted stochastic variations that complicate the detection and estimation of the signal of interest. In this thesis we will use a broader definition of noise being any unwanted signal that impedes the estimation of a desired signal. For example, in one application, remote sensing data can be used to estimate surface reflectance. In another, the data can be used to estimate the state of the atmosphere for weather monitoring. In the first application, the surface reflectance is the signal of interest and the atmospheric effects are noise. In the second application, the atmospheric effects are the signal and surface effects are noise. Therefore, the multiplicative and additive illumination effects m and a on the signal of interest r will be referred to as noises in this thesis, even though they may be unknown constant values. Note that random sensor noise is still noise in both applications.

2.2 Classification Applications

In defining our problem of estimating surface reflectance and considering solutions for it,

it is valuable to keep in mind the downstream processing problems for which reflectance estimates will be useful. One such problem is that of classification, in which all of the pixels of an image are to classified into a thematic map, where each pixel is identified as belonging to a distinct class of materials such as grass or road. Supervised classification utilizes the expertise of a human analyst to identify the example, or training pixels, from an image that are used to train a classifier algorithm. Unsupervised classifiers attempt to cluster and partition the pixels autonomously, after which the assignment of the clusters to known material classes is performed by a human analyst (Schowendgert, 1997, Fukunaga, 1990, Jain and Dubes, 1988).

Figure 2.3 depicts a simple supervised classification processes. A human analyst identifies training pixels for the material classes to be identified. Next, a classifier algorithm is trained using the training pixels. The classifier is then applied to the rest of the pixels in the image and the thematic map generated.

Historically, classification has been done using the radiance spectra directly from the image i without estimating reflectance, mostly due to lack of robust techniques for estimating reflectance in single and multispectral channels. Classifying in radiance space suffers from two significant problems. The first is that spatial variations in illumination conditions across the scene can make pixels of the same material type have different radiance spectra at different locations in the scene. Second, scene to scene changes in illumination conditions prevent training in one image and then applying the classifier successfully to other images.

Both of these problems can be solved by estimating reflectance and then training and classifying in reflectance space instead. By removing the in-scene and scene-to-scene illumination variations, these noises no longer impede the classifier. Using reflectance

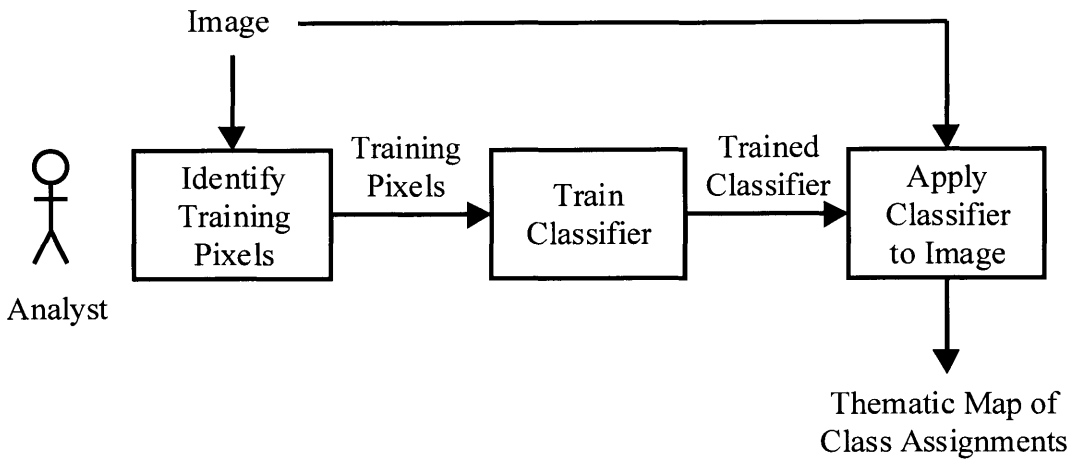
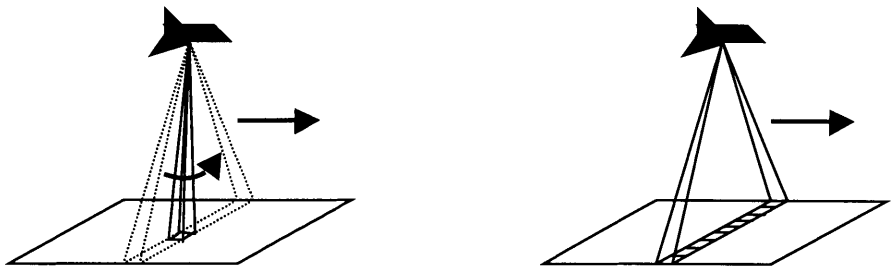


Figure 2.3. *Supervised classification processing. A human analyst identifies training pixels for the material classes to be identified. Next, a classifier algorithm is trained using the training pixels. The classifier is then applied to the rest of the pixels in the image and the thematic map generated.*

spectra also opens the possibility of selecting training spectra from a spectral library without the need for an analyst to identify training pixels in an image. Therefore, classification applications can benefit greatly from robust reflectance estimation algorithms. In Chapter 7 we shall see the superior performance of SPIRE algorithms over a state-of-the-art physics-based algorithm as a preprocessing step in such a classification example.

2.3 Hyperspectral Sensors

There are currently two airborne hyperspectral sensors in use whose data sets are in wide use by the scientific community. The first is the Airborne Visible-InfraRed Imaging Spectrometer (AVIRIS) (Vane, et al., 1984, Vane, 1987, Vane, et al., 1993) which has 224 spectral channel detectors, each with a spectral bandwidth of approximately 0.01 μm, covering the visible and near-infrared from 0.38 μm to 2.5 μm. The AVIRIS sensor is a scanning pushbroom sensor and collects 614 cross- path pixels in each scan. Figure 2.4(a)



(a) (b)

Figure 2.4. *Pushbroom sensors. A scanning pushbroom sensor (a) collects data at a single pixel location at a time. This field of view is scanned side to side to collect a line of pixels perpendicular to the sensor's flight path. As the sensor moves along the flight path, the collected lines form a rectangular image. A non-scanning pushbroom sensor (b) collects data on a line of pixels at the same time. As the sensor moves along the flight path, the collected lines form a rectangular image.*

depicts how a scanning pushbroom sensor collects image data one pixel at a time, scanning perpendicular to the flight path. At each pixel, the sensor collects data in all of the spectral channels, yielding a radiance spectrum at that location. As the sensor moves along the flight path, the successive lines form a rectangular image in each spectral channel, resulting in a hyperspectral image cube. AVIRIS is carried by an ER-2 airplane at a typical altitude of 20 km, which yields a 10-km cross-path swath with 20m-square pixels on the ground.

The second airborne sensor is the HYperspectral Digital Imagery Collection Experiment (HYDICE) sensor (Rickard, et al., 1993, Basedow, et al., 1995). HYDICE has a bi-prism dispersing element and a two-dimensional focal plane detector array which allows it to operate as a non-scanning pushbroom type sensor. Figure 2.4(b) depicts how a non-scanning pushbroom sensor collects image data by collecting a single line of pixels at a time perpendicular to the flight path. The focal plane array has 320 pixels in the spatial cross-path direction and 210 pixels in the spectral dimension, yielding 210 spectral channels that cover 0.4 μm to 2.5 μm. HYDICE

is carried by a CV 580 airplane with an altitude range of 1.5 to 8 km, yielding spatial ground resolutions of 0.8 to 4 meters. Data from the HYDICE airborne sensor was used for the algorithm testing in this thesis due to its availability and higher spatial resolution, since both spatial and spectral processing techniques were to be used. For both HYDICE and AVIRIS data, there exists state of the art physics-based algorithms (see Section 2.5.2) that estimate reflectance, whose results can be compared to the newly developed algorithms described in this thesis.

At the writing of this thesis, only two space-based hyperspectral sensor were operational: the HYPERION sensor on board the Earth Observer - 1 spacecraft (Pearlman, et al., 2000) and the Fourier Transform Hyper-Spectral Imager (FTHSI) onboard the MightySat II.1-SINDRI spacecraft (Otten, et al., 1998). While operational, calibrated data were not yet available for use.

2.4 Solution Approaches

When given i and (2.1), the problem of estimating r is under-constrained since there are three unknowns and only one equation. Ancillary information regarding r, m, and/or a is required to estimate r. Figure 2.5 depicts a general estimator E for r and the four types of ancillary information, in addition to i, that can be used to estimate r. At the top left of the estimator box is $K_p\{r\}$ which denotes prior knowledge of the surface reflectance r. In this notation, K represents knowledge or information, the subscript $_p$ denotes prior information, and the $\{\}$ brackets indicate a set of information that could take several different forms. Examples of possible $K_p\{r\}$ include a complete map of the surface spectral reflectance r_p of the scene from an earlier time, or global statistics on the mean and covariance of a typical scene's reflectance. At the top right is $K_c\{r\}$, current information about the reflectance in the scene, an example of which would be the location and ground truth spectra of known materials in the current scene.

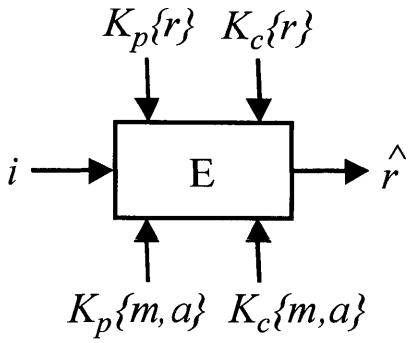


Figure 2.5. *General reflectance estimator. The estimated scene reflectance is found from the sensed image i using ancillary information. Prior knowledge about the reflectance ($K_p\{r\}$) and the multiplicative and additive noise terms ($K_p\{m,a\}$), and current information about the reflectance ($K_c\{r\}$) and the multiplicative and additive noise terms ($K_c\{m,a\}$) may be used individually or in various combinations to estimate reflectance.*

Since m and a are both caused by interactions between the illumination and the atmosphere, knowledge about them both are grouped together. Prior information about m and a, $K_p\{m,a\}$, is at the lower left and could include representative ensembles of m and a, radiative transfer physics, and typical atmospheric profiles, such as temperature, humidity and aerosols concentrations, which can be used in simulations to predict m and a. At the lower right is $K_c\{m,a\}$, current knowledge regarding m and a in the scene, an example of which would be absorption line ratios from current hyperspectral image data that allow the estimation of the water vapor absorption component of m. Table 2.1 lists more possible examples of what would be considered types of current and prior information about r, m, and a.

2.5 State of the Art

Currently, the two main techniques for performing atmospheric compensation are the Empirical Line Method and the atmospheric physics-based approach.

Knowledge Type	Scene Specific	Global
Prior Reflectance: $K_p\{r\}$	Complete r_p (estimated from prior observations or simulations) Stats on r_p: mean, covariance Low spatial frequency content Statistics on change in r since the prior r_p was collected: $\delta r = r - r_p$ Variations due to recent weather conditions and season	Typical scene mean, covar, low spatial frequency power content, and δr stats Spectral reflectance library or ensemble of possible $\{r\}$
Current Reflectance: $K_c\{r\}$	Location and ground truth spectra of known materials Size of δr changes Mean, var, low spatial frequency content from other sensors Weather conditions and season. Known events: landslide, etc.	N/A
Prior Illumination: $K_p\{m,a\}$	Scene statistics on m and a: mean, covariance Scene spatial frequency power of m and a	Ensembles of possible $\{m\}$ and $\{a\}$ spectra Global statistics on m and a: mean, covariance Global spatial frequency power of m and a Radiative transfer physics Atmospheric models Aerosol models Typical visibility Total column water vapor look up table for band ratios Top of atmosphere solar radiation model
Current Illumination: $K_c\{m,a\}$	Groundbased downwelling irradiance sensor data Groundbased visibility meter data Hyperspectral water vapor absorption line measurements Hyperspectral based aerosol and visibility estimates Date and time Viewing geometry Weather and seasonal conditions Knowledge of the presence of clouds, haze, precipitation, smoke, volcanic eruption, etc.	N/A

Table 2.1. *Examples of prior and current information about r, m, and a.*

2.5.1 The Empirical Line Method

The Empirical Line Method (ELM) (Griffin, et al., 1999, Kruse, et al., 1990, *ENVI Users Guide*, 1997) uses ground truth reflectance spectra for materials and the locations of those materials in the current image, to compensate for illumination changes. Therefore, ELM uses

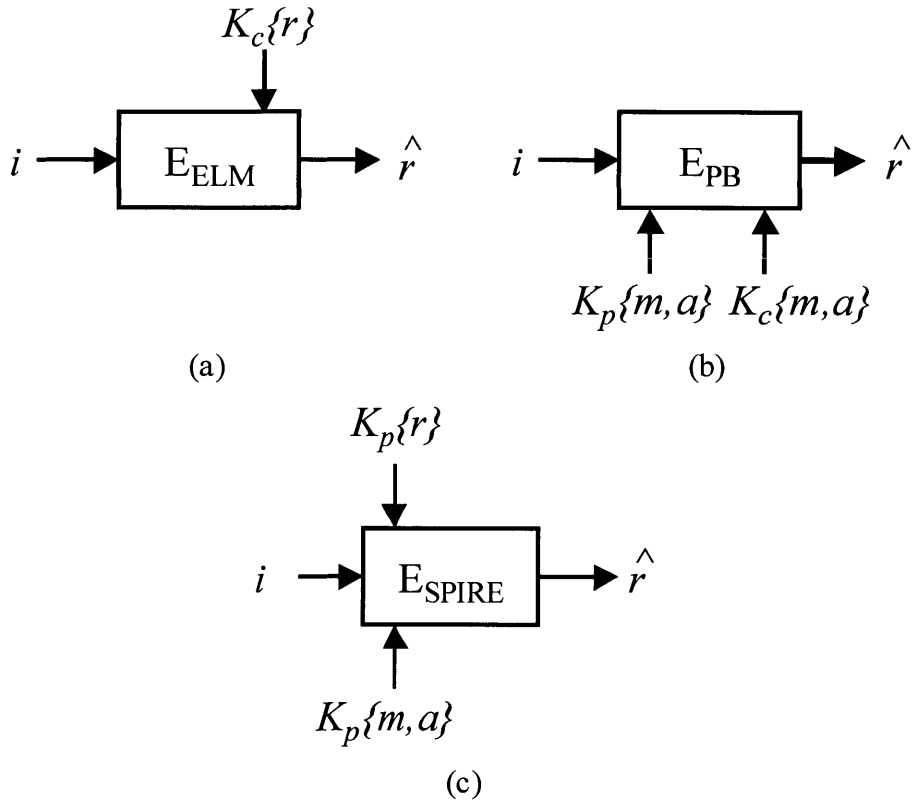


Figure 2.6. *Reflectance estimators for Empirical Line Method (ELM) (a), Physics-Based (PB) approach (b), and Surface Prior Information Reflectance Estimation (SPIRE) approach (c). ELM uses current knowledge about the reflectance r ($K_c\{r\}$) of calibrated test panels in the scene to estimate the entire scene's reflectance from the sensed image i. PB uses both prior and current information ($K_p\{m,a\}$, $K_c\{m,a\}$) about the state of the atmosphere to estimate the multiplicative and additive effects and then compensate for them. SPIRE uses prior information in the form of an earlier estimate of the scene's reflectance, plus prior ensembles of multiplicative and additive effects.*

only current information about the reflectance in the scene $K_c\{r\}$, as depicted in Figure 2.6(a). ELM uses the ground truth reflectance spectra and the corresponding pixel values from the sensed image to estimate the multiplicative and additive noise effects of the atmosphere over the ground truth pixels. Well calibrated, uniform test panels that cover multiple pixels are ideal. The corrections calculated are then applied to the rest of the pixels in the image, which assumes that the multiplicative and additive noises are uniform across the scene. The ELM typically

yields good estimates of the surface reflectance of the image. However, if there are spatially varying effects over parts of the image where there are no ground truth points, then the estimates in these areas will be worse. In other words, the atmospheric compensation will be best over and near the ground truth points, and potentially worse in other parts of the image.

Operationally, the ELM also suffers from the problem that most scenes do not contain well calibrated test panels. Even if a prior reflectance image exists from which "truth" spectra could be used, the random location of changes to r in subsequent images makes it difficult to automatically select which spectra to use. Also, if a ground truth spectrum is misregistered with a point on the ground containing a different material, then the resulting estimates for m and a will be incorrect.

2.5.2 Atmospheric Modeling Physics-Based Approaches

In the atmospheric physics-based (PB) approach, models of the atmosphere and radiation transfer physics, combined with data extracted from the current image, are used to estimate the effects of the atmosphere on the sensed scene and then to compensate for them in order to recover surface reflectance. Therefore, the PB approach uses both prior and current information about m and a: $K_p\{m,a\}$, and $K_c\{m,a\}$, as depicted in Figure 2.6(b). These techniques do not need information regarding the surface reflectance, but they do require information regarding the state of the atmosphere so that its effect on solar radiation can be modeled. Some of the atmospheric state information is extracted from the hyperspectral data itself. For example, the effects of water vapor are estimated by using the ratios of certain channels near and in water absorption bands (Kaufman and Sendra, 1988; Kaufman and Gao, 1992). Typically, other information such as the type of aerosol model to use and the visibility must be input by an analyst, which often requires iteration for best results. Whether or not all needed atmospheric state information can

eventually be extracted from hyperspectral data to fully automate such algorithms is a current area of research. Most atmospheric physics-based codes assume a clear, cloudless day, and their performance degrades on days when there are overcast clouds or haze above the sensor. This problem is an obstacle to making operational physics-based atmospheric compensation algorithms for use with an airborne sensor.

There were two commonly used physics-based atmospheric compensation codes available to the author: the ATmospheric REMoval (ATREM) (ATREM Users Guide, 1997) program, and Air Force Research Lab (AFRL)/MODTRAN code (Adler-Golden, et al., 1998). Both operate in a similar manner and differ mainly in the fidelity of the radiative transfer code used. ATREM uses the 6S (Second Simulation of the Satellite Signal in the Solar Spectrum) (Vermote, et al., 1997) scattering code and ATREM specific radiative transfer modules, while the AFRL code uses the newer MODTRAN (Berk, et al., 1998) radiative transfer code. MODTRAN uses somewhat higher fidelity techniques to estimate water absorption than ATREM and accounts for the effects light scattered by the atmosphere into the pixel from adjacent areas. ATREM was used in this research since it has been more widely used by other researches than the newer AFRL/MODTRAN code.

2.5.3 Other Approaches and Research

In this section we will review possible approaches and techniques other than the ones already discussed. The techniques listed here are not used widely operationally, but represent the thrusts of other research efforts into solving this problem.

Barnes (Barnes, 1995) reviews several techniques for performing atmospheric correction in multispectral imagery. These include ELM, a physics-based approach, and several Dark Object Subtraction (DOS) techniques which include the Histogram Minimum Method, Pairwise

Regression Matrix, and Covariance Matrix method. All of theses DOS techniques rely on the assumption that the darkest material in a scene has 0% reflection, which often is not the case and leads to erroneous results. Barnes also discussed the Shadow and Scene Color Standard Technique which makes use of pixels of the same material in and out of shadow, which is only applicable to images with identifiable shadows.

Healy, et al. (Healy, et al.,1999) have developed a maximum-likelihood (ML) (VanTrees, 1968, Helstrom, 1999, Schowengerdt, 1997) classification approach which classifies each radiance spectra from hyperspectral HYDICE data as one of 498 material reflectance spectra in a standard spectral reflectance library. This ML classifier-based "invariant material-identification" algorithm was trained using spectra obtained by combining MODTRAN simulated illumination effects over many atmospheric conditions with the library material spectra to generate approximately 18,000 radiance spectra. Fairly good results were achieved on HYDICE data under clear sky conditions as compared to a Spectral-Angle Mapper algorithm which attempts to classify pixels based on the angle between spectra. However, this approach cannot estimate the reflectance of unknown materials, and is limited to operating within the atmospheric and illumination conditions used to train the classifier.

Like the ML classifier, other well established non-random vector estimation techniques exist in which one can treat the surface reflectance as a parameter vector to be estimated. In addition, one could also try various other statistical estimation and filtering approaches that rely on the statistical relationships between the reflectance r and the radiance i received at the sensor. The main weakness of these approaches is that they assume that we can effectively predict and model the illumination conditions and know all of the possible materials that will be seen.

In many applications this is not feasible. New materials are continuously being

developed and used, and detection of them is often the goal of remote sensing. Unexpected and unpredictable atmospheric conditions due to smoke, volcanic eruptions, and clouds can be extremely difficult, if not impossible, to model and predict. In mobile robotic applications, unpredictable changes in artificial illumination conditions, or the effects of scatter from terrain, walls, and buildings can be very difficult to model. Even if complex models and simulations for these conditions can be developed, each application would require a different simulation to generate the measurement model, which often involves costly modeling and software development. Indeed, the development of the physics-based radiative transfer code MODTRAN has been a large multi-decade effort.

Given these issues, and given the plausible availability of prior reflectance information in a scene, we chose instead to pursue a new approach, inspired by the color constancy algorithms introduced in Chapter 1. The resulting algorithms require very little information about the illumination conditions, limited to either the maximum spatial variability of the illumination, or ensembles of empirical illumination noise spectra only large enough to capture the few degrees of freedom present in them. In either case, no modeling or simulation of the atmosphere or illumination conditions is required. Instead, these algorithms rely primarily on the prior surface reflectance information to estimate surface reflectance and are therefore called Surface Prior Information Reflectance Estimation (SPIRE) algorithms.

2.6 Surface Prior Information Reflectance Estimation Algorithms

The research described in this thesis had the goal of developing atmospheric compensation algorithms that make use of prior information about the surface reflectance of a scene, and limited prior information about m and a. Figure 2.6(c) depicts this use of only $K_p\{r\}$ and $K_p\{m,a\}$.

We assume that $K_p\{r\}$ consists of a scene-specific prior reflectance map r_p, and global information consisting of an ensemble of all known reflectance spectra **{r}**. The scene prior r_p can be used to derive scene specific prior statistics and spatial frequency content as needed (if the scene specific r_p is not available, then a global average r_p could be used instead, but at the cost of lower performance). We assume that $K_p\{m,a\}$ consists of the global prior ensembles of multiplicative and additive noise vectors **{m}** and **{a}**, where a vector inside curly brackets denotes an ensemble of such vectors. In addition, other ensembles such as $\{\log \mathbf{r}\}$, $\{\log \mathbf{m}\}$, and **{rm}** can be derived. Typical spatial frequency contents of m and a are also assumed to be known.

The motivation for this approach was threefold. First, such an approach would likely work better than an atmospheric physics-based one when using airborne sensors since it is not dependent on the unknown state of the atmosphere above the sensor platform. Second, the repetitive and ever increasing amount of remote sensing data collected and processed around the world promises eventually to provide frequently updated maps of surface reflectance as well as ensembles of r, m, and a that could serve as the source of the needed prior information. Third, approaches to atmospheric compensation that utilize prior information about the surface reflectance have been largely unexplored.

A typical operational use of SPIRE algorithms is depicted in Figure 2.7. First, an initial prior image i_p is collected during the best possible conditions to allow for the estimation of a prior reflectance r_p using a physics-based code, or ELM if ground truth spectra are available. If a physics-based code is used, then this initial image must be collected with a hyperspectral imager. For example, a hyperspectral sensor on board a satellite could be used repeatedly to update a prior reflectance cube on clear days, using a physics-based code to process the data. This then

serves as the source of the needed prior reflectance information, as well as the reference from which changes will be measured. This process may be part of larger effort to build and maintain a larger reflectance map. The ensembles for **{m}** and **{a}** and their typical spatial frequency content have already been determined from previous images using ELM or physics-based codes.

Once this prior information is in place, subsequent images collected at a later time can be processed using SPIRE algorithms, independent of cloud cover or availability of test panels. Continuing our example from the previous paragraph, an airborne sensor could be used on overcast days in between the clear days when the satellite images were collected, and these airborne collected images can be processed using SPIRE algorithms.

Note that some applications of SPIRE algorithms may not require the collection of a prior image to generate the prior reflectance data. For example, in an inspection application, a geometric model of the object to be inspected could be combined with laboratory spectra measurements of materials to generate a prior reflectance map for the object. This prior could then be used for comparison in SPIRE processing to detect either material changes or contamination.

In many applications, one could use a "pseudo" prior reflectance as input to SPIRE without requiring ELM or a physics-based algorithm to generate a prior. One example of a pseudo prior would be to simply scale a prior radiance image using its maximum radiance so that all of the pixel values fall between zero and some number less than or equal to one. This pseudo prior can then be used as the prior reflectance when running SPIRE on the current and subsequent images. The reflectance estimates for these images will be incorrect in an absolute sense, but all images processed using the same pseudo prior will have the same systematic errors. The resulting pseudo-reflectance image cubes can then be used for performing change detection, since the errors introduced by using the pseudo prior would be same in all. More importantly, any

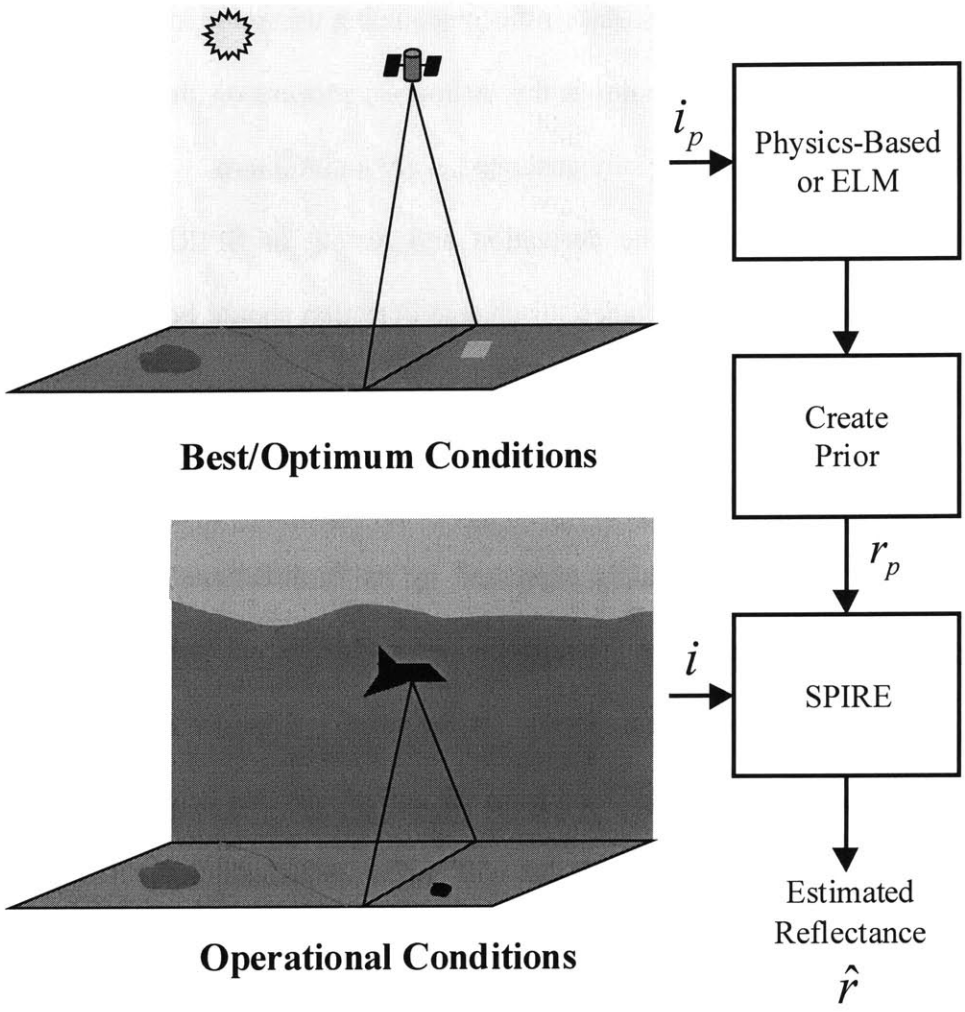


Figure 2.7. *Operational use of SPIRE algorithms in airborne remote sensing applications: At the top, images are routinely collected whenever the best or optimum conditions present themselves for physics-based or Empirical Line Method (ELM) algorithms. The latest estimated reflectance image from this process is then used as a prior for operational conditions where physics-based and ELM techniques do not work, but where SPIRE does.*

classifier trained on one of the image cubes can then be applied to all other image cubes that were processed using the same pseudo prior, allowing successful classification across images in which illumination conditions have changed. This important result will be discussed further in Chapter 7.

Figure 2.8 depicts the research scope of thesis. On the left of the figure are the various

processes and techniques that could be utilized to generate prior information $K_p\{r\}$ and $K_p\{m,a\}$. On the right is the estimation of surface reflectance using this prior information and the current image. This thesis will concentrate on the estimation process on the right of Figure 2.8 and assumes that some process has already generated $K_p\{r\}$ and $K_p\{m,a\}$.

Before continuing on to the derivation and testing of SPIRE algorithms, two issues regarding the use of a prior reflectance for change detection should be addressed. They are the Bi-directional Reflectance Distribution Function and specular reflections.

2.6.1 Bi-directional Reflectance Distribution Function and Prior Reflectance Information

The reflectance of a real material is expressed by its Bi-directional Reflectance Distribution Function (BRDF) which describes the reflectance for different incident illumination and sensor viewing geometries (Schowengerdt, 1997). If the prior reflectance is obtained from a single sensor viewing geometry that is different from the current viewing geometry, then there can be a differences in reflectance due to BRDF that will appear as a changes in reflectance with respect to the prior. Under such circumstances, a BRDF and geometry induced change in reflectance will be indistinguishable from a change in surface material. Such changes affect all reflectance estimation algorithms used in change detection applications, including ELM, PB, and SPIRE.

For the purposes of this thesis, all surfaces were assumed to be Lambertian, i.e. having uniform reflectance in all geometries. In the test data set used in this thesis, the sensor viewing geometries were similar in all images used (including the one used to generate the prior images), but the incident illumination geometry did vary. It is possible that prior reflectance generation techniques could be developed that incorporate BRDF information to take into account the viewing and illumination geometry to generate a prior reflectance that includes the BRDF effect, but they are beyond the scope of this thesis.

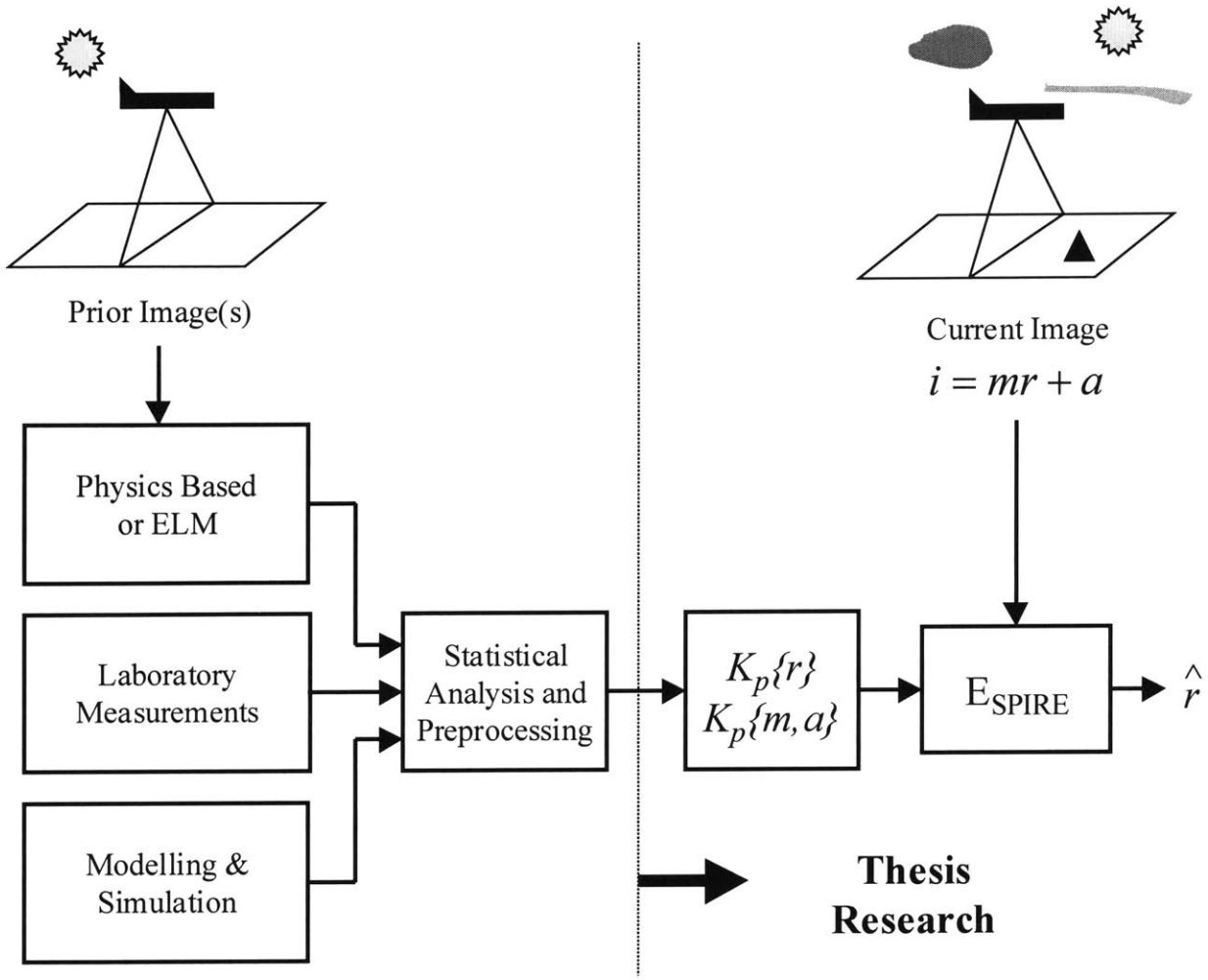


Figure 2.8. *Research scope of thesis. On the left of the figure are the various processes and techniques that could be utilized to generate prior information about the reflectance r and multiplicative and additive noise terms m and a ($K_p\{r\}$, $K_p\{m,a\}$). On the right is the estimation of surface reflectance using this prior information and the current image. This thesis concentrates on the estimation process on the right.*

2.6.2 Specular Reflections and Reflectance Estimation

A specular reflection is a bright mirror like reflection of the incident source illumination that occurs under certain geometries. Similar to changes caused by BRDF and viewing geometry, specular reflections will also appear as changes in surface reflectance. Again, this is

true of ELM, PB, and SPIRE algorithms and must be handled in a post processing step after initial surface reflectance estimation.

2.7 SPIRE Algorithm Taxonomy

SPIRE algorithms were developed in this thesis that use spatial processing (Spatial SPIRE), spectral processing (Spectral SPIRE), and combined spatial-spectral processing (Combined SPIRE). Different SPIRE algorithms were developed for different illumination noise cases. Since no useful signal will reach the sensor without the multiplicative noise m, all cases considered assume that m is present. Given that a may or may not be present, depending on the application, there are six possible permutations of a being present or not, and whether or not m and a are each spatially varying or uniform. 2.2 lists these permutations and the SPIRE algorithm identifications for each case. For Spatial SPIRE, the solution algorithms for these six cases are identified numerically as the Case 1-6 algorithms. For Spectral SPIRE, the solution algorithms for these six cases are identified alphabetically as the Case A-F algorithms. For Combined SPIRE, the solution algorithms are identified with Roman numerals as the Case I-VI algorithms.

Figure 2.9 depicts the SPIRE algorithm taxonomy for the algorithms developed and tested in this thesis, organized by the spatial variability of the multiplicative noise m and additive

m	*a*	Spatial	Spectral	Combined
Uniform	0	Case 1	Case A	Case I
Uniform	Uniform	Case 2	Case B	Case II
Varying	0	Case 3	Case C	Case III
Varying	Uniform	Case 4	Case D	Case IV
Uniform	Varying	Case 5	Case E	Case V
Varying	Varying	Case 6	Case F	Case VI

Table 2.2. *SPIRE algorithm case designations.*

noise *a*. Spatially varying *m* and *a* are assumed to be band-limited to low spatial frequencies as will be discussed in Chapter 3. At the top of the tree in Figure 2.9 is the most general case considered with spatially varying *m* and spatially varying *a*, for which the Spatial SPIRE Case 6 algorithm was developed. No Spectral SPIRE solution for Case F was found, as is discussed in Chapter 5. The Combined Case VI algorithm, while feasible, is left as a suggestion for future work, since *a* can be treated as spatially uniform in most airborne remote sensing applications.

On the left branch are algorithms that work when the additive noise is *a* zero and we have only a spatially varying multiplicative noise, which includes the Spatial Case 3, Spectral Case C, and Combined Case III algorithms. For the case where the multiplicative noise *m* is spatially uniform (and the additive noise *a* is zero), a computationally efficient Spatial Case 1 algorithm was developed that differs significantly from the Spatial Case 3 algorithm. For Spectral and Combined SPIRE, the Case A and Case I algorithms are identical to the Case C and Case III algorithms respectively, so no separate Case A or Case I algorithms were developed. Note that Spatial Case 3 and 6 algorithms can also solve the Case 1 problem, since a uniform *m* is contained within the same low spatial frequency band as a slowly varying *m*.

On the center branch is the condition where the multiplicative noise *m* is spatially uniform but the additive noise *a* is spatially varying. The Spatial Case 5 algorithm solves this case in a computationally efficient manner. No Spectral Case E solution was found. Like the Combined Case VI algorithm, the Combined Case V is feasible, but is left as a suggestion for future work, since *a* can be treated as spatially uniform in most airborne remote sensing applications.

On the right branch are algorithms that work when the multiplicative noise *m* is spatially

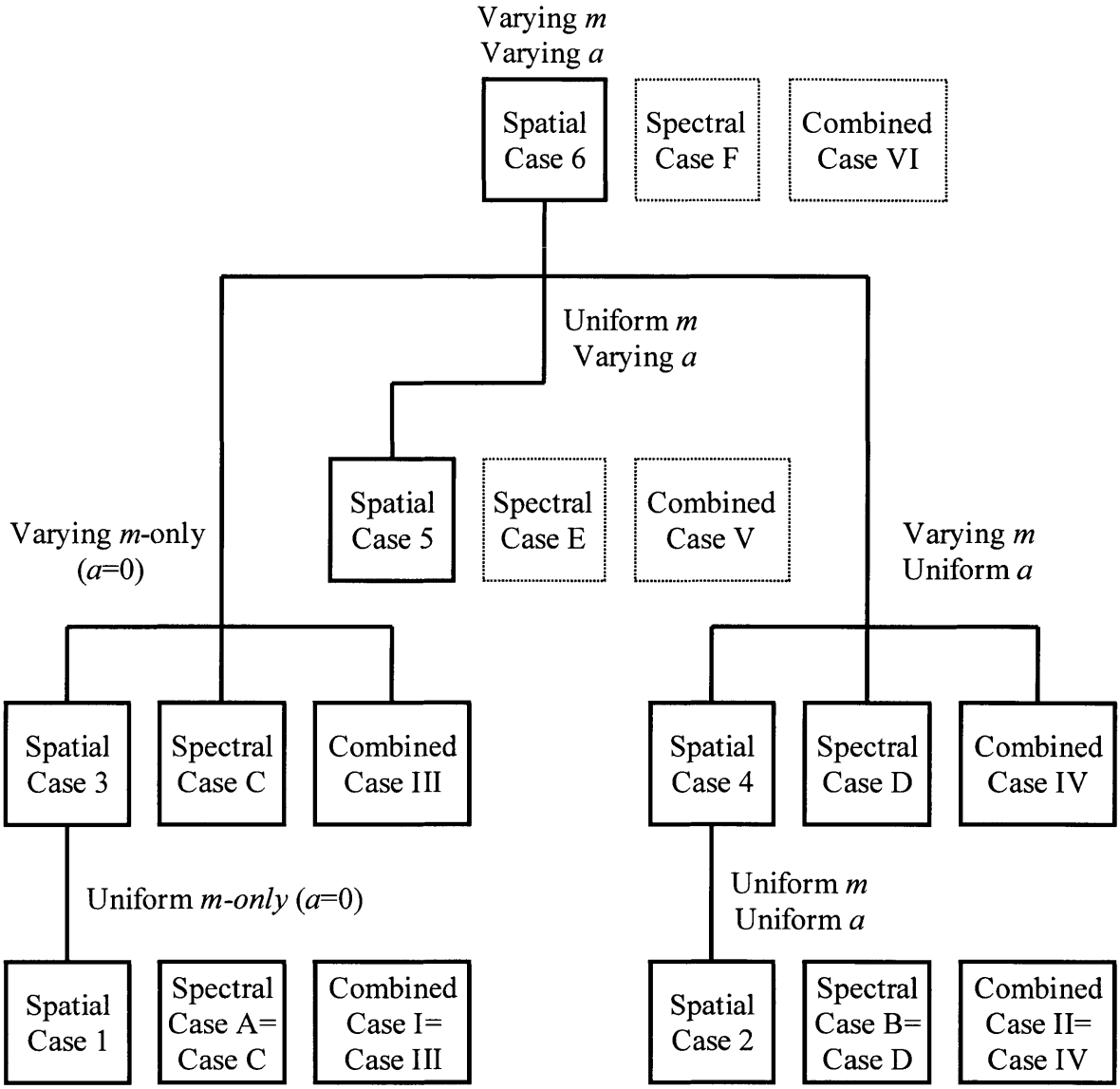


***Figure 2.9**. SPIRE algorithm taxonomy based on the spatial variability of the multiplicative noise m and additive noise a. Spatially varying m and a are assumed to be band-limited to lower spatial frequencies. Algorithms outlined in dashed lines were not developed and are suggestions for further work. For Spectral and Combined, algorithms for cases where m is uniform are identical to those for when m is varying.*

varying and the additive noise a is spatially uniform, which includes the Spatial Case 4, Spectral Case D, and Combined Case IV algorithms. For the case were the multiplicative noise is spatially uniform as well, the computationally efficient Spatial Case 2 algorithm can be used.

For Spectral and Combined SPIRE, the Case B and Case II algorithms are identical to the Case D and Case IV algorithms respectively, so no separate Case B or Case IV algorithms were developed. Again, note that Spatial Case 4 and 6 algorithms can also solve the Case 2 problem, since a uniform m is contained within the same low spatial frequency band as a slowly varying m.

2.8 Summary

In this chapter we defined our problem as that of estimating the surface reflectance r given an image i with multiplicative m and additive a noise terms. This general model is applicable to most remote sensing and imaging applications, since it is independent of the specifics of the physics that generate m and a. The general reflectance estimator for this problem requires either prior or current ancillary information about the reflectance or noise terms in order to sufficiently constrain the problem so that it can be solved. The current state-of-the-art ELM algorithm and physics-based approaches both suffer from problems that limit their operational use. The SPIRE approach, using prior reflectance information and representative ensembles of the multiplicative and additive noise terms, holds the promise of performing well under operational conditions that ELM and physics-based approaches do not.

With our problem defined and an understanding of the strengths and weakness of the existing state of the art in hand, we now proceed to developing SPIRE algorithms. As part of a structured research effort, we will first pursue Spatial SPIRE algorithms that utilize spatial information and processing, then Spectral SPIRE algorithms that utilize spectral information and processing, followed by Combined SPIRE algorithms that use combined spatial-spectral processing. Therefore, in the next chapter we will derive and test Spatial SPIRE algorithms which use spatial filtering to remove illumination noise and estimate surface reflectance.

Chapter 3

Spatial SPIRE Algorithms

3.1 Introduction

In this chapter, we begin our development of SPIRE reflectance estimation algorithms by developing Spatial SPIRE algorithms that make use of spatial filtering to remove the multiplicative and additive noise effects caused by varying illumination and atmospheric scattering. These algorithms operate on a single spectral channel image at a time, making no use of relationships between spectral channels.

3.2 Spatial SPIRE Algorithm Derivation

3.2.1 Algorithmic Approach

For the remainder of this thesis, we assume that the prior surface reflectance information is derived from an earlier observation of the scene. Therefore, our goal is the estimation of local surface spectral reflectance changes from multiple observations under varying multiplicative and additive noise. We make the following assumptions:

1) The first observation, or sensed scene, has been processed to estimate the surface reflectance. This will be referred to as the prior.
2) Any changes or new objects in the scene consist of a few relatively small changes with defined edges, thereby causing changes mostly to the high spatial frequency content of the image.
3) The spatial frequency content of the multiplicative and additive illumination noises

across the scene are band limited to lower spatial frequencies.

In effect we are assuming that the illumination noises and changes in reflectance are separated into distinct spatial frequency bands, but there is no constraint on the frequency content of the image reflectance itself. In other words, both the illumination noises and the surface reflectance function can have overlapping spatial frequency content, but the illumination noises and reflectance changes cannot.

A subsequent observation is made on the scene after the changes have occurred. The sensed image formation model is given by:

$$i[n_x, n_y, n_\lambda] = r[n_x, n_y, n_\lambda] m[n_x, n_y, n_\lambda] + a[n_x, n_y, n_\lambda] \qquad (3.1)$$

Where $i[n_x, n_y, n_\lambda]$ is the sensed image, $r[n_x, n_y, n_\lambda]$ is the current surface reflectance and signal of interest, $m[n_x, n_y, n_\lambda]$ is the multiplicative noise, and $a[n_x, n_y, n_\lambda]$ is the additive noise. Each of these terms is a three dimensional array (often referred to as an image "cube" in the literature, though it is typically the shape of a brick) where n_x and n_y are spatial dimension indices and n_λ is the spectral dimension index. All data are assumed to be discrete in space and wavelength.

Our problem is to recover an estimate of $r[n_x, n_y, n_\lambda]$ given the sensed image $i[n_x, n_y, n_\lambda]$ and the prior reflectance $r_p[n_x, n_y, n_\lambda]$. Six distinct cases will be considered. In all six cases, the multiplicative noise will be present. The six permutations are formed by including the additive noise or not, and by allowing the multiplicative and additive noise terms to be either spatially varying or constant.

Table 3.1 summarizes the six cases and their solution techniques. Using the notation of

$i = i[n_x, n_y, n_\lambda]$, we denote that all variables are assumed to be varying in all three dimensions, including the prior r_p. The only exceptions are spatially uniform variables that only vary in the spectral dimension, which are denoted using a subscripted naught: $c_o = c_o[n_\lambda]$. All solution derivations are done for a single spectral channel image and will be independently applied to each spectral channel of the image cube. Therefore, operations such as mean and variance refer to operations done spatially across the image. The h refers to the impulse response of a linear low pass filter, which is also often referred to as h_{LPF} in the text. The est function is an estimator described for Cases 4 and 6 in their corresponding sections below.

Note that Case 1 is a subset of Case 3 in that both have only multiplicative noise present, so that Case 1 can be considered to be Case 3 with a noise that has only a zero-frequency (DC) component. Similarly, Cases 2, 4, and 5 are simplifications of Case 6. Rather than first discussing Case 6 and Case 3 first, and then discussing the other cases, we discuss them all in order of increasing complexity, since the solution algorithms for the more complex cases tend to build upon the solutions to the simpler cases, yielding an easier to understand progression.

3.2.2 Case 1: Spatially Uniform Multiplicative Noise Only

In this first case the additive noise a is zero and the multiplicative noise m_o is constant across the scene:

$$i = rm_o \tag{3.2}$$

Our problem, therefore, is to estimate m_o and divide the sensed image i by it to recover an estimate of the surface reflectance:

$$\hat{r} = i\frac{1}{\hat{m}_o} = rm_o\frac{1}{\hat{m}_o} \cong r \tag{3.3}$$

The simplest solution to (3.2) is to estimate m_o as the ratio of the mean of the sensed image to the mean of the prior image:

$$\hat{r} = i\frac{\text{mean } r_p}{\text{mean } i} \tag{3.4}$$

This relies on the assumption that any changes to the scene are small enough to not change the mean of r appreciably from that of the prior r_p. Formally, we can derive this solution by first moving (3.2) into log space to linearize the problem:

$$\log i = \log r + \log m_o \tag{3.5}$$

Next we subtract the log of the mean of i from both sides of (3.5):

$$\begin{aligned} \log i - \log(\text{mean } i) = \log r + \log m_o - \log(\text{mean } r) \\ - \log(\text{mean } m_o) = \log r - \log(\text{mean } r) \end{aligned} \tag{3.6}$$

We see in (3.6) that, since m_o is a constant, it is eliminated from the equation, but with the complication that the log of the mean of the surface reflectance is also subtracted. Under the assumption that any changes to the image are small and have a negligible effect on the mean, we can add back in the mean of the prior reflectance image to both sides, take the exponential, and solve for an estimate of the reflectance, obtaining (3.4). Alternatively, we can reverse the order of the spatial mean and log functions in the subtracted term in (3.6). This results in a similar, but more computationally complex solution:

$$\hat{r} = i\exp((\text{mean}(\log r_p) - \text{mean}(\log i)) \tag{3.7}$$

Case	m	a	Problem	Eqn	Solution	Eqn
1	Uniform	None	$i = rm_o$	(3.2)	$\hat{r} = i\frac{\text{mean } r_p}{\text{mean } i}$	(3.4)
2	Uniform	Uniform	$i = rm_o + a_o$	(3.8)	$\hat{r} = \sqrt{\frac{\text{var } r_p}{\text{var } i}}(i - \text{mean } i) + \text{mean } r_p$	(3.11)
3	Varying	None	$i = rm$	(3.13)	$\hat{r} = \exp(\log i - \log i * h + \log r_p * h)$	(3.15)
4	Varying	Uniform	$i = rm + a_o$	(3.17)	$\hat{r} = \exp\begin{pmatrix}\log(i - \text{mean } i + \text{est}(mean(rm))) \\ -\log(i - \text{mean } i + \text{est}(mean(rm))) * h \\ +\log r_p * h\end{pmatrix}$	(3.19)
5	Uniform	Varying	$i = rm_o + a$	(3.20)	$\hat{r} = \frac{i - h * i}{\sqrt{\frac{\text{var}(i - h * i)}{\text{var}(r_p - h * r_p)}}} + h * r_p$	(3.23)
6	Varying	Varying	$i = rm + a$	(3.24)	$\hat{r} = \exp\begin{pmatrix}\log(i - h_1 * i + \text{est}(h_1 * (rm))) \\ -\log(i - h_1 * i + \text{est}(h_1 * (rm))) * h_2 \\ +\log r_p * h_2\end{pmatrix}$	(3.28)

Table 3.1. *Six cases considered with the problem equations to solve and the solution equations.*

3.2.3 Case 2: Spatially Uniform Multiplicative and Spatially Uniform Additive Noise

In this case, both the additive noise a_o and the multiplicative noise m_o are constant across the scene:

$$i = rm_o + a_o \tag{3.8}$$

We begin by subtracting the spatial mean of the sensed image from the image itself:

$$i - \text{mean}(i) = rm_o + a_o - \text{mean}(rm_o + a_o) = m_o(r - \text{mean } r) \tag{3.9}$$

This eliminates the constant additive noise since its mean is equal to itself. By taking the

spatial variance of (3.9) and recognizing that the variance does not change if the mean is subtracted before calculating the variance, we can estimate m_o using:

$$\hat{m}_o = \sqrt{\frac{\text{var}\, i}{\text{var}\, r_p}} \tag{3.10}$$

Substituting this back into (3.9), replacing the mean of r and with that of r_p as in Case 1, and solving for the estimate of the reflectance, we obtain:

$$\hat{r} = \sqrt{\frac{\text{var}\, r_p}{\text{var}\, i}}\left(i - \text{mean}\, i\right) + \text{mean}\, r_p \tag{3.11}$$

In effect, we are scaling the zero mean sensed image to achieve the variance of the prior reflectance image, and then adding in the prior reflectance mean. As expected, this solution also works for Case 1. If we assume that $a_o = 0$ and that the ratio $m/m_o = 1$, then (3.11) can be reduced to the same form of equation as (3.4) and (3.7), where (3.10) is the estimate of m_o:

$$\hat{r} = i\sqrt{\frac{\text{var}\, r_p}{\text{var}\, i}} \tag{3.12}$$

3.2.4 Case 3: Spatially Varying Multiplicative Noise Only

In this third case, the additive noise a is zero but we have a multiplicative noise varying spatially across the scene:

$$i = rm \tag{3.13}$$

As in Case 1, we first move to log space to linearize the problem as in (3.5). In Case 1,

we were able to remove the constant log m_o by subtracting the mean of the sensed image from itself. In this case, log m is not a constant across the scene, but a spatially varying term confined to low spatial frequencies. By analogy, we can eliminate it by subtracting a spatially low pass filtered version of the sensed image from itself, where the filter h has support over the same bandwidth as log m:

$$\log i - h * \log i = \log r - h * \log r + \log m - h * \log m = \log r - h * \log r \qquad (3.14)$$

This approach is similar to Case 1 since the mean function can be viewed as a filtering operation. The subtracted filtered version of log m cancels the unfiltered log m, but at the expense of subtracting the low frequency components of log r in the process. This is equivalent to a high pass filtering operation. Given our assumption that any small changes present in the scene have a negligible effect on the low frequency content of the image, we can recover these lost low frequency components from the prior by adding back a low pass filtered version of the prior image to both sides. We then take the exponential and solve for an estimate of the surface reflectance, resulting in:

$$\hat{r} = \exp\left(\log i - \log i * h + \log r_p * h\right) \qquad (3.15)$$

To illustrate this processing, we take the Discrete Time Fourier Transform (DTFT) (Oppenheim, et al., 1997; 1999) of the log of (3.15) to obtain (3.16), in which the subscript $_{log}$ denotes that the term is the DTFT of the log of the variable, and $\phi=[\phi_x, \phi_y]$:

$$\tilde{\hat{R}}_{\log}(\phi) = \tilde{I}_{\log}(\phi) - \tilde{H}(\phi)\tilde{I}_{\log}(\phi) + \tilde{H}(\phi)\tilde{R}_{p,\log}(\phi) \qquad (3.16)$$

Figure 3.1. depicts a one dimensional example of Case 3 processing in the spatial

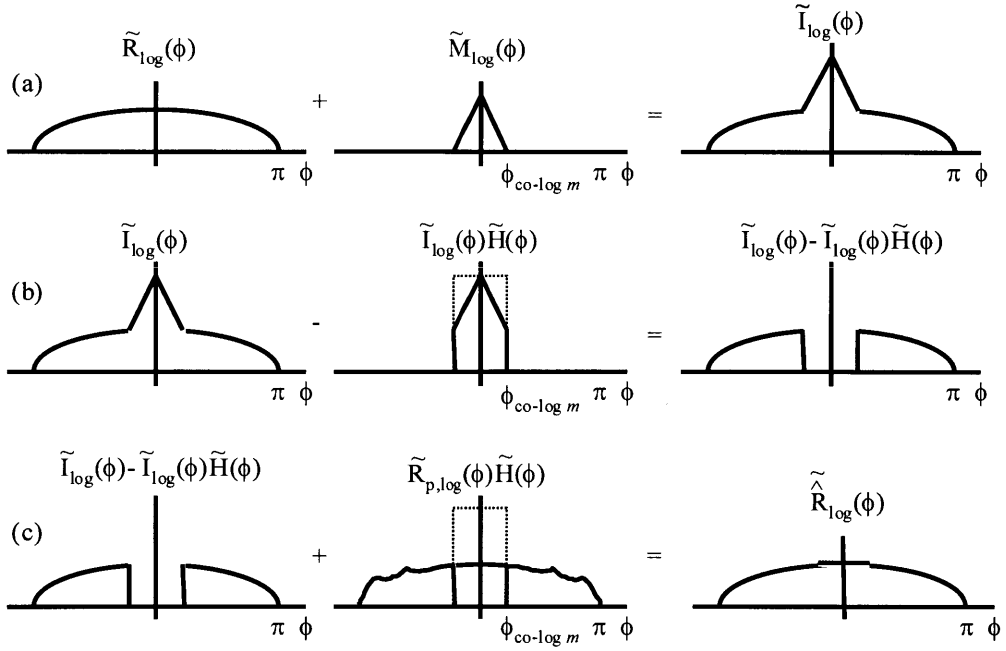


Figure 3.1. *Spatial Case 3 problem and algorithm depicted in the log spatial frequency domain. In (a) the sensed image* $\tilde{I}_{log}(\phi)$ *is formed by adding the Discrete Time Fourier Transform (DTFT) of the log of reflectance* $\tilde{R}_{log}(\phi)$ *and DTFT of log of the multiplicative effect* $\tilde{M}_{log}(\phi)$ *which is band limited to spatial frequencies below* $\phi_{co-log\,m}$. *In (b), the log image* $\tilde{I}_{log}(\phi)$ *is high pass filtered to remove the low frequency components containing the multiplicative effect* $\tilde{M}_{log}(\phi)$. *In (c), the low frequencies of the reflectance lost in the filtering operation are estimated and restored by low pass filtering the prior* $\tilde{R}_{p,log}(\phi)$ *and adding it to the result of (b), resulting in the estimated current reflectance.*

frequency domain. In Figure 3.1(a), the sensed image is shown as the sum (in log space) of the reflectance and the multiplicative noise which is band limited to frequencies below the cut-off frequency $\phi_{co\text{-}log\ m}$. In Figure 1(b), the log of the sensed image is high pass filtered (by subtracting a low-pass filtered version of itself). In Figure 1(c), the missing low frequencies of the reflectance are restored using the low frequency content of the log of the prior, with a small difference due to the effect of any changes in the image. Note that high-frequency differences between the prior reflectance and the new reflectance do not affect the processing.

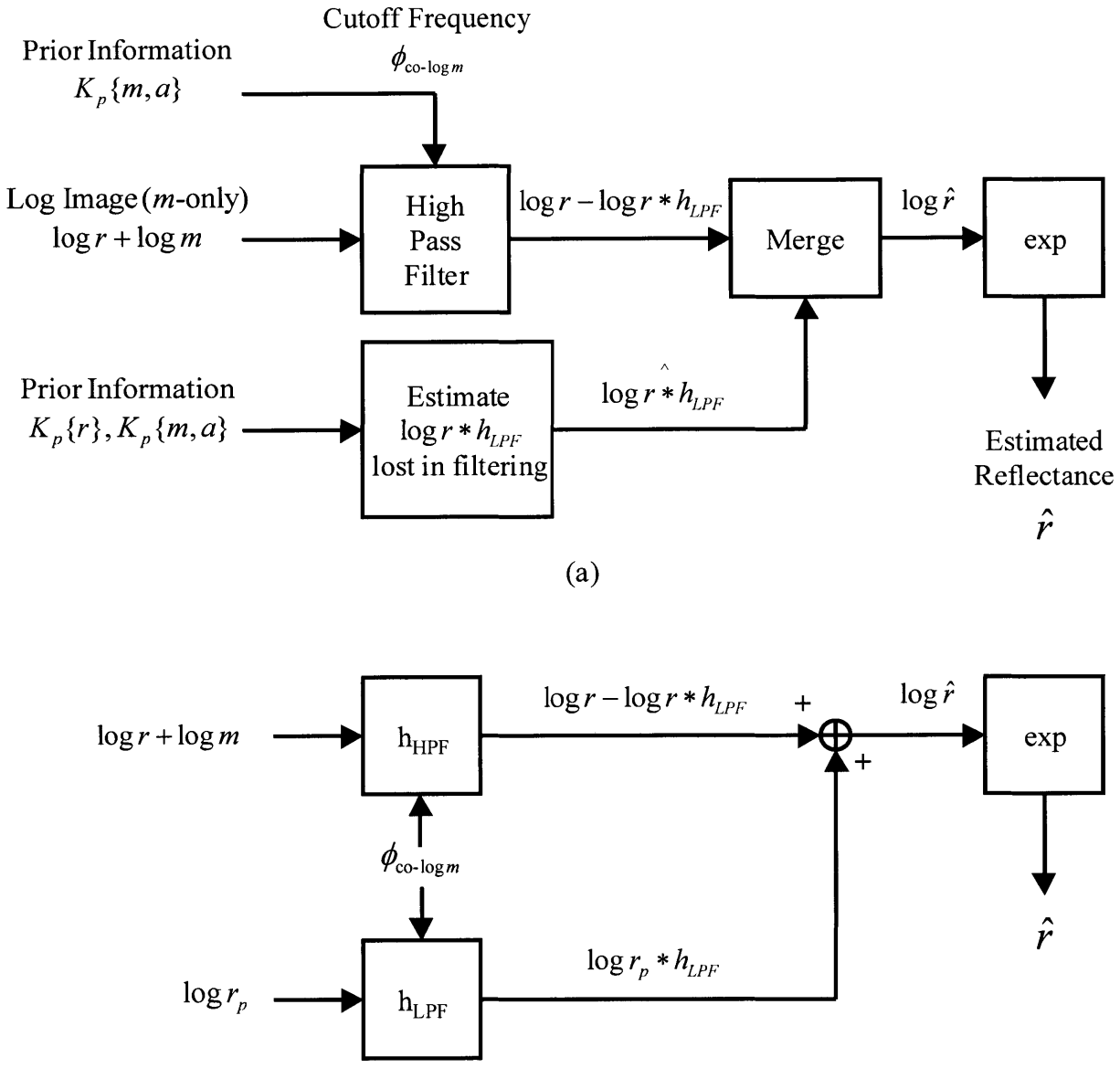


Figure 3.2. *Spatial Case 3 generalized processing block diagram (a) and specific implementation tested (b). In (a), the m-only image is high-pass filtered using the $\phi_{co\text{-}logm}$ spatial cutoff frequency from the prior information about the multiplicative noise m ($K_p\{m,a\}$). The low-pass filtered* log *r lost in the filtering operation is then estimated using prior information $K_p\{r\}$ and $K_p\{m,a\}$ and merged with the filtered signal. Finally, the exponential is taken to estimate reflectance. In (b), the lost low pass filtered* log *r is estimated by low pass filtering the log of the prior reflectance* log r_p, *and the merge operation is a simple addition.*

Figure 3.2(a) depicts the generalized processing block diagram for the Case 3 solution algorithm, while Figure 3.2(b) depicts the specific algorithm implemented and tested in this thesis. Figure 3.2(a) depicts a generic block for estimating $\log r * h_{LPF}$ based on prior information, since there may be other approaches than the low-pass filtering of the prior used in Figure 3.2(b). Similarly, a generic merge block is used in place of the addition used in this thesis, since strategies that weight the high-passed image and the estimate of $\log r * h_{LPF}$ differently may be useful in some applications.

3.2.5 Case 4: Spatially Varying Multiplicative and Spatially Uniform Additive Noise Case

In this case, we have multiplicative noise that varies spatially across the scene, plus additive noise that is uniform across the scene:

$$i = rm + a_o \tag{3.17}$$

This case may be the most important one for solving practical remote sensing problems, since the additive upscatter radiation can usually be assumed to be uniform across a scene. This case is similar to Case 3, but with a constant additive noise term added to the image. Since we no longer have a purely multiplicative situation, we cannot immediately move to the log domain to linearize the problem. The solution approach involves removing the additive noise a_o as we did in Case 2, estimating and adding back in the mean(rm) lost in the process, and then using the algorithm for Case 3 to complete the solution. We begin by subtracting the global mean of the sensed image from itself, whereby the additive noise a_o is removed since it is constant:

$$i - \text{mean}\, i = rm + a_o - \text{mean}(rm + a_o) = rm - \text{mean}(rm) \tag{3.18}$$

This, however, is not the form that we need to utilize the algorithm for Case 3. In order to do so, we must estimate and add back in the single number mean(*rm*). In order to estimate this number, we first use the sensed image values to place bounds on its possible values, making use of the fact that reflectance values only range between 0 and 1. We then step through the range of possible values for mean(*rm*), adding them back into (3.18), and then estimating the reflectance using the Case 3 algorithm. The final value of mean(*rm*) used is the one that minimizes the mean square error (MSE) between the prior reflectance and the estimated reflectance. The block diagram for this algorithm is depicted in Figure 3.3. Once we have found the best estimate for mean(*rm*), we then use the Case 3 algorithm with that value to estimate the surface reflectance. The equation for this final estimate is given by:

$$\hat{r} = \exp\begin{pmatrix} \log\left(i - \text{mean}\, i + \text{est}(mean(rm))\right) \\ -\log\left(i - \text{mean}\, i + \text{est}(mean(rm))\right) * h + \log r_p * h \end{pmatrix} \tag{3.19}$$

3.2.6 Case 5: Spatially Uniform Multiplicative and Spatially Varying Additive Noise Case

In this case, we have a uniform multiplicative noise across the scene, plus additive noise that is spatially varying across the scene:

$$i = rm_o + a \tag{3.20}$$

To derive the solution algorithm for this case, we begin by subtracting a low-pass filtered version of the image from itself:

$$i - h * i = rm_o - h * (rm_o) + a - h * a = m_o(r - h * r) \tag{3.21}$$

Using the same arguments as in Case 3 (except now we are not in the log domain) that

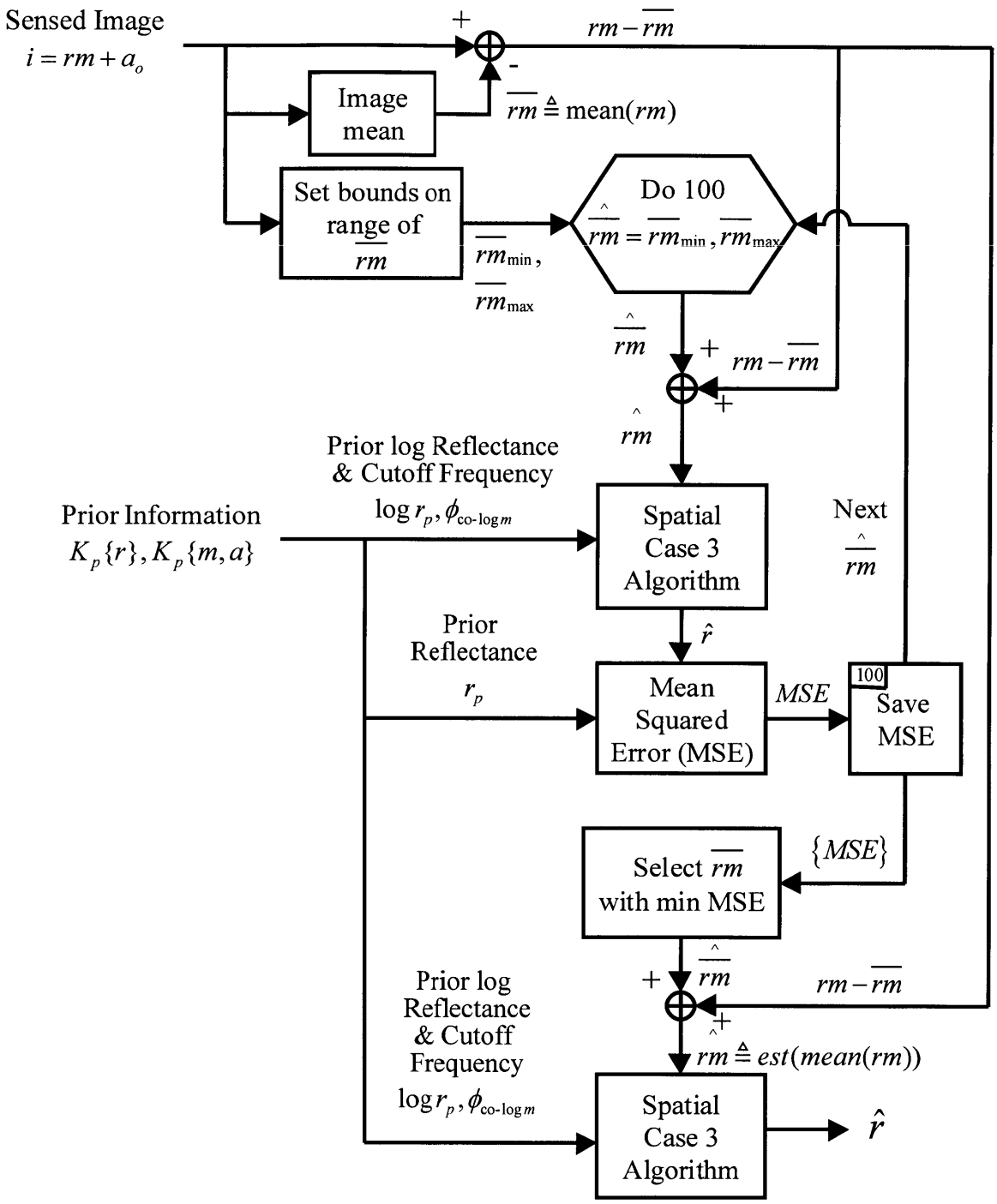


Figure 3.3. *Case 4 processing block diagram. The mean of the sensed image is subtracted to remove the additive noise* a_o. *A loop is then entered to estimate the* mean(*rm*) *lost, with the sensed image used to set bounds on the range of* mean(*rm*). *As different values of* mean(*rm*) *are stepped through, they are added back in and the Spatial Case 3 algorithm run to estimate reflectance. This reflectance is then compared to the prior reflectance and the mean squared error (MSE) calculated. The estimate of* mean(*rm*) *with the minimum MSE is then used to calculate the final estimated reflectance.*

the additive noise is band limited to low frequencies over which the low-pass filter has support, this operation eliminates the additive noise, while also subtracting the term $h^*(rm)$. However, unlike Case 4, we do not have to estimate this term and add it back in. Instead, we can use a similar technique as in Case 2. Taking the variance of both sides of (3.21) and using the prior image as the argument of the filtering operation (since it has the same low frequency content as the new reflectance) allows us to estimate m_o as:

$$\hat{m}_o = \sqrt{\frac{\mathrm{var}(i - h*i)}{\mathrm{var}(r_p - h*r_p)}} \tag{3.22}$$

Substituting (3.22) into (3.21) and solving for an estimate of the reflectance yields:

$$\hat{r} = \frac{i - h*i}{\sqrt{\frac{\mathrm{var}(i - h*i)}{\mathrm{var}(r_p - h*r_p)}}} + h*r_p \tag{3.23}$$

3.2.7 Case 6: Spatially Varying Multiplicative and Spatially Varying Additive Noise

In this case, we have multiplicative noise varying across the scene, plus additive noise that is also varying spatially across the scene:

$$i = rm + a \tag{3.24}$$

This is the most general case, and each of Case 1-Case 5 represents a simplification of this one. We begin with the same approach as in Case 5: subtracting a low pass filtered version of the image from itself. The low pass filter used at this stage will be referred to as h_1. As in Case 5, this eliminates the additive noise at the expense of subtracting the term $h_1^*(rm)$, a low

pass filtered version of the reflectance times the multiplicative noise.

$$i - h_1 * i = rm + a - h_1 * (rm + a) = rm - h_1 * (rm) \tag{3.25}$$

At this point we have a situation similar to Case 4 equation (3.18), in that we must estimate the term $rm * h_1$ and add it back in to (3.25). Once this is accomplished, we will have a new problem equivalent to Case 3, which can then be solved using the Case 3 algorithm, just as was done in Case 4. In Case 4, we estimated a single number to add back in, finding the number that minimized the mean squared error of the resulting solution when compared to the prior. In this case, we must estimate a two-dimensional function across the image. By describing this two-dimensional spatial function as a summation of weighted orthogonal basis functions, we can repeatedly use the same technique as in Case 4 to find the weighting coefficient $C_E(k_x, k_y)$ for each basis function, one at a time. For this paper, the Discrete Cosine Transform (DCT) (Lim, 1990) was used; other possibilities exist, such as Lapped Orthogonal Transforms (LOT) (Malvar, 1992). We express the function to add back in as a DCT as follows:

$$\text{est}(h_1 * (rm)) = \begin{cases} \dfrac{1}{N_x N_y} \displaystyle\sum_{k_x=0}^{N_x-1} \sum_{k_y=0}^{N_y-1} w_x(k_x) w_y(k_y) C_E(k_x, k_y) \cos\frac{\pi}{2N_x} k_x (2n_x + 1) \cos\frac{\pi}{2N_y} k_y (2n_y + 1), \\ \qquad \text{for } 0 \le n_x \le N_x - 1,\ 0 \le n_y \le N_y - 1 \\ 0, \qquad \text{otherwise} \end{cases} \tag{3.26}$$

Where:

$$w_x(k_x) = \begin{cases} \frac{1}{2}, & k_x = 0 \\ 1, & 1 \le k_x \le N_x - 1 \end{cases}, \quad w_y(k_y) = \begin{cases} \frac{1}{2}, & k_y = 0 \\ 1, & 1 \le k_y \le N_y - 1 \end{cases} \tag{3.27}$$

The estimate of the surface reflectance associated with the $\text{est}(rm * h_1)$ that minimizes the mean square error between $\hat{r}$ and r_p can then be written in the form of the solution to Case 3, where a second low pass filter h_2 is used in the log domain, and is given by:

$$\hat{r} = \exp\begin{pmatrix} \log\left(i - h_1 * i + \text{est}(h_1 * (rm))\right) \\ -\log\left(i - h_1 * i + \text{est}(h_1 * (rm))\right) * h_2 + \log r_p * h_2 \end{pmatrix} \tag{3.28}$$

Figure 3.4 depicts the flow diagram for this algorithm. Since $\text{est}(rm * h_1)$ is restricted to lower spatial frequencies, only those DCT coefficients passed by h_1 need be used in the estimate.

3.2.8 Relationship of Spatial SPIRE to Homomorphic, Retinex, and Lightness Algorithms

Homomorphic algorithms have been used to achieve simultaneous contrast enhancement and dynamic range compression in image processing applications (Oppenheim, et al., 1975, Lim, 1990). Under the same assumptions as Case 3 of a slowly varying illuminant (low spatial frequency) and a rapidly varying reflectance (high spatial frequency), these algorithms separate the two components in log space using high and low pass filters in the spatial frequency domain. The two components are recombined after attenuating the log illuminant and enhancing the log reflectance. Any low frequency reflectance content is also attenuated. Versions of this algorithm that handle a limited amount of backscatter from clouds are able to enhance images using an adaptive technique that enhances the local contrast using the local mean (Lim, 1990). These algorithms are aimed at image enhancement for viewing and not for estimating reflectance.

Edwin Land developed the Retinex "lightness" algorithm that simulates the color constancy processing of the human visual system (Land, 1964; Land, 1986; Land, 1986). Color

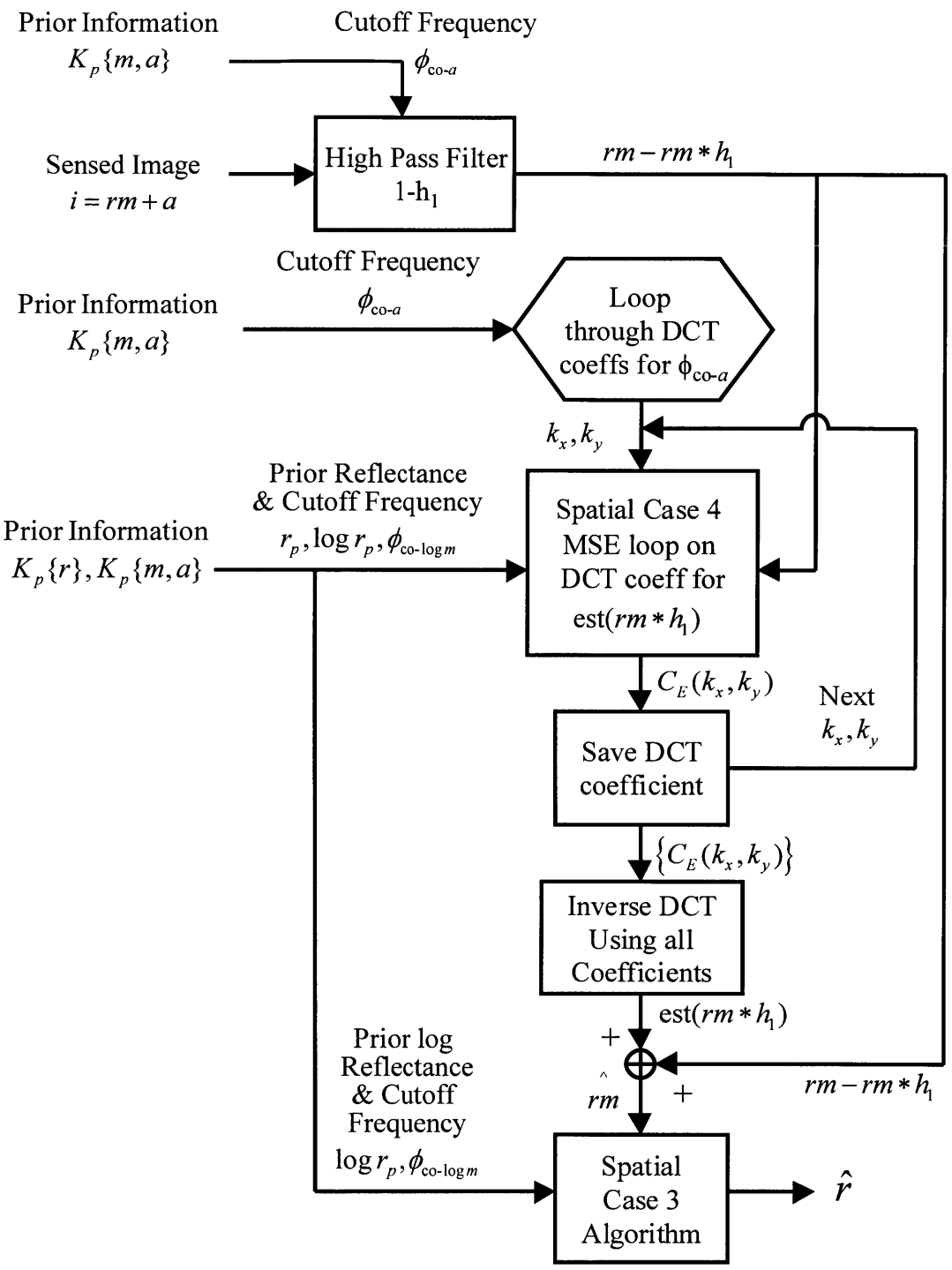


Figure 3.4. *Case 6 processing block diagram. The sensed image is first high pass filtered to remove the additive noise a. A loop is then entered to estimate the $rm * h_1$ lost using a Discrete Cosine Transform (DCT). The number of DCT coefficients needed is derived from the band limit of a $\phi_{co\text{-}a}$. For each DCT coefficient, a Case 4 MSE loop is used to estimate the correct value (as in Figure 3.3). Once all the coefficients are estimated, the estimate of $rm * h_1$ is added back in and the Spatial Case 3 algorithm is run to obtain the final estimated reflectance.*

constancy can be interpreted as estimating the reflectance in three spectral channels which, when combined, define a vector color in a Red-Green-Blue type space which is consistent under varying illumination conditions. Lightness algorithms process images in the three channels corresponding to the short, middle, and long wavelength cone receptors of the human eye, to yield a lightness value for each point in the scene in each channel. These three lightness values then serve as coordinates in a three dimensional color space. The term lightness is used because in each channel, the relative lightness or gray level of each pixel, when compared to all other pixels in the image, is calculated. The color constancy of this algorithm, as well as similar lightness algorithms subsequently developed, did approach that of human vision. Hurlbert (Hurlbert, 1986, Hurlbert, 1989) reviewed and drew formal mathematical connections between Land's lightness algorithms and those developed by other researchers. However, these lightness algorithms are all dependent on three assumptions regarding the image being processed.

Lightness Assumption 1: The imaged area is a so called "Mondrian" scene, meaning that it is composed of flat patches of uniform reflectance. A typical Mondrian scene would be a collage of different colored rectangles of random sizes that may overlap. The different colors are typically randomly distributed across the scene.

Lightness Assumption 2: That the effective irradiance varies slowly across the scene and is everywhere independent of the viewer's position.

Lightness Assumption 3: That the surface reflectance averages to a gray value in each wavelength channel, which is the same for all scenes. This is often

referred to as the gray world assumption.

All lightness algorithms attempt to solve the problem presented in Case 3, where there is no additive illumination noise and only a multiplicative illumination effect on the surface reflectance signal of interest. Lightness algorithms cannot solve Cases 2, 4, 5, and 6 which involve an additive term. Lightness Assumption 2 is the same as the Spatial SPIRE assumption that the illumination is band limited to low spatial frequencies. Like the SPIRE algorithm for Case 3, lightness algorithms move into log space and effectively use a high-pass filter to eliminate the illumination. However, lightness algorithms differ from Spatial SPIRE algorithms in that the solution of (3.13) requires being able to estimate $h^*\log r$ and adding it back in. In the Case-3 Spatial SPIRE algorithm, this estimate is found by simply filtering the log of the prior reflectance. In lightness algorithms, there is no prior information available, other than that implied by Lightness Assumption 3. Effectively, Lightness Assumption 3 allows one to add back in an estimate of only the DC component, or mean, of log r. However, this estimate is fixed ahead of time as the average surface reflectance for all images. If the particular image being processed doesn't contain a statistically similar distribution of surface reflectances as that of the rest of the world, then the restored DC component will be incorrect. Therefore, in the best case, the lightness algorithms can restore only the DC component of the image, losing lower spatial frequency information that occupies the same spatial frequency band as the spatially varying illumination. If the material content of the scene is not similar to the whole world, then the estimated DC component will be incorrect as well. Lightness algorithms work well on Mondrian images because the random distribution of color patches across the image assured by Lightness Assumption 1 reduces the amount of low-frequency power in the image and forces the

mean to be similar in all images generated using the same probability distribution.

3.3 Algorithm Testing and Results

3.3.1 Test Data Description

The Spatial SPIRE algorithms were tested on three types of image data: simulated single channel, simulated hyperspectral, and real hyperspectral. The use of simulated data allowed for complete control of both the surface reflectance and the multiplicative and additive illumination effects, thereby eliminating uncertainties in the data when measuring the absolute performance of the algorithms. The use of real data demonstrated the true validity of the algorithms by ensuring that no real world effects not included in the simulated data adversely affect the algorithms. The use of real hyperspectral data also allowed the comparison of Spatial SPIRE algorithm performance to physics-based atmospheric compensation algorithms which require hyperspectral data.

Data from the widely used HYDICE airborne hyperspectral sensor (Basedow, et al., 1995) was selected for use in this thesis. The Department of Energy (DOE) operates the Southern Great Plains Site (SGPS) Atmospheric Radiation Measurement (ARM) and Cloud and Radiation Testbed (CART) facility located southeast of Lamont, Oklahoma. HYDICE data was collected over this site during 23 to 28 June 1997 as part of the Atmospheric Compensation Investigation (ACI) data collection (Lockheed-Martin, 1997). The ACI data collection over the ARM Site was carried out for the express purpose of collecting a set of hyperspectral data that could then be used for the development and testing of atmospheric compensation algorithms. Various test panels were in place on the ground, and ground truth data was collected over these panels using a spectroradiometer. Various other instruments were used to measure meteorological conditions,

including all-sky photographs (Lockheed-Martin, 1997).

3.3.2 Single Channel Image Testing

Since the SPIRE algorithms work on a single channel basis, their performance can be tested using single channel images. A single-channel sub-image 256 pixels square was extracted from HYDICE ARM site data from Run 07 of 24 June 1997 from the 0.468 μm channel. The selected sub-image contains a variety of image features, including test panels (near the top), grass, mowed grass, roads, buildings, and vehicles. This single radiance image was then scaled to bring all pixel values to between 0 and 1 to simulate a reflectance image. This approach ensured that the image used contained typical ground features and pixel value variability found in real data.

Since the SPIRE algorithms are intended for use in applications where small changes in surface reflectance have occurred since a prior image collection and reduction to reflectance, a second single-channel test image was generated from the original by making a small modification to it. The modified reflectance image is depicted in Figure 3.5(a). The modification made was to replace a 4x5 pixel area in the grass near image coordinates [152,218] with a 4x5 pixel area copied from the road in the lower left of the image. A close-up of the modification is shown in Figure 3.5(b). Figure 3.5(c) shows a vertical profile along the dashed line at n_x=152, where the modification is centered around the gray line at n_y=218. Test panels, road, parking lot, and grass are all evident in the profile.

To test each of the algorithms developed for the six cases described in Section 3.2, the modified test image was subjected to multiplicative and additive effects appropriate to each case. Figure 3.5(d) shows the same profile as Figure 3.5(c), but through the Case-6 test image which was generated from the modified reflectance image by multiplying it with a horizontal ramp

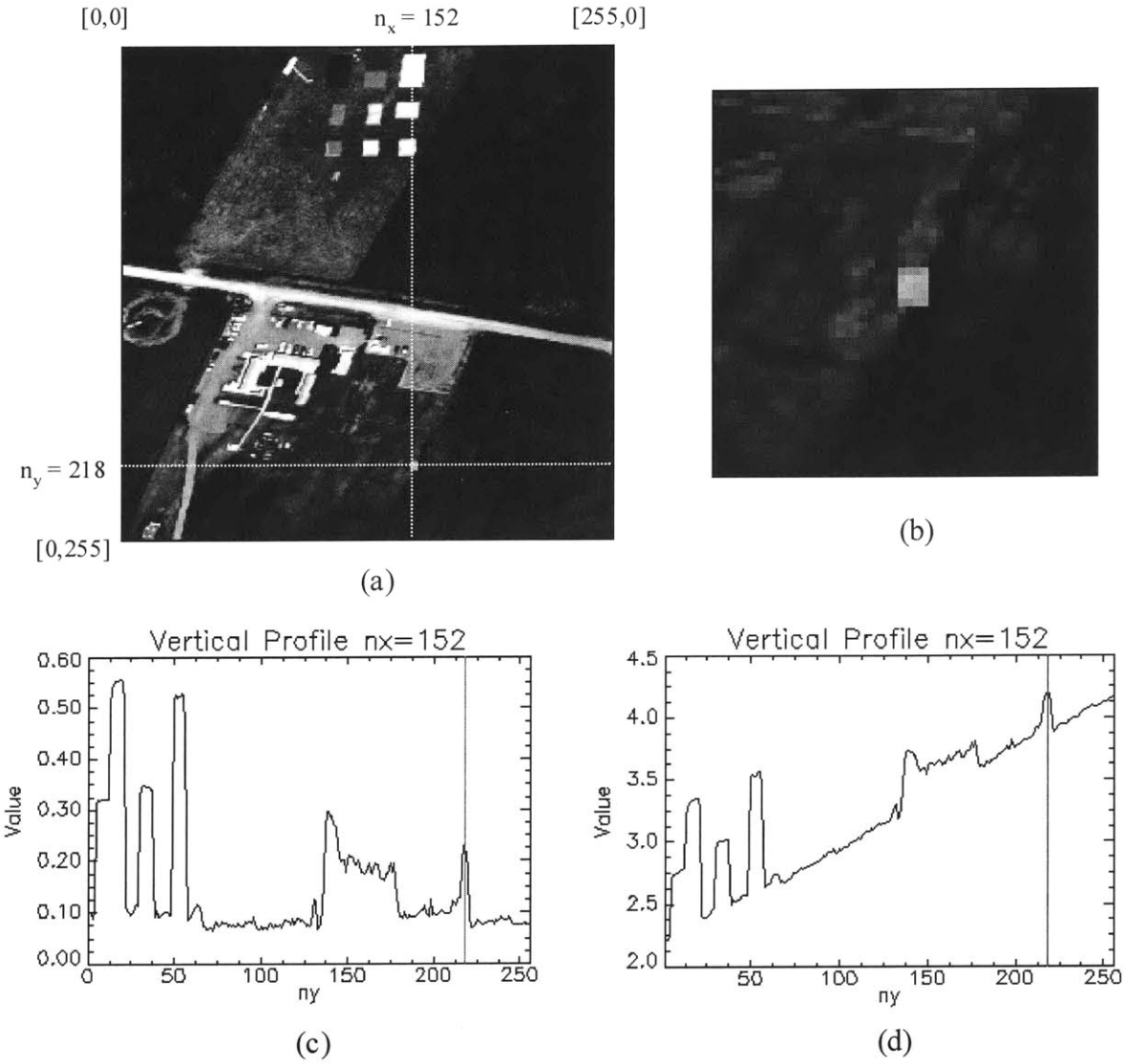


Figure 3.5. *Modified surface reflectance single channel test image (a) with closeup of modification made in area surrounding [152,218] (b), with vertical (c) profile through n_x=152 plotted. The modification simulates a change in the image of a material similar to the road being placed in the grass field. For testing the Case 6 algorithm, the reflectance image of (a) was multiplied by a horizontal ramp image going from 1.0 to 3.0 across the image, and then a vertical ramp going from 2.0 to 4.0 was added. The vertical profile through this test image is plotted in (d).*

image ranging in value from 1 to 3, and then adding a vertical ramp image ranging in value from 2 to 4. Table 3.2 describes the multiplicative and additive images applied to the modified

reflectance image to generate the test images for Cases 1-6.

Case	Multiplicative Image	Additive Image
Case 1	Constant=5.0	None
Case 2	Constant=5.0	Constant=3.0
Case 3	Horizontal Ramp=1.0-3.0	None
Case 4	Horizontal Ramp=1.0-3.0	Constant=3.0
Case 5	Constant=5.0	Vertical Ramp=2.0-4.0
Case 6	Horizontal Ramp=1.0-3.0	Vertical Ramp=2.0-4.0

Table 3.2. Multiplicative and additive images applied to modified reflectance image to generate single channel test images.

3.3.3 Single Channel Image Processing Results

The resulting test images for Cases 1-6 were processed through their respective algorithms described in Section 3.2. In all cases, the original, unmodified reflectance image was used as the prior reflectance image from which prior spatial information needed by the algorithms was extracted. For each case, the absolute percent error image between the estimated reflectance image and the modified reflectance image used to generate the test image was calculated using:

$$absolute\ percent\ error = 100 * \left| r - \hat{r} \right| / r \tag{3.29}$$

Where r represents the modified reflectance image used to generate the case test image and $\hat{r}$ represents the estimated reflectance recovered from the test image. A 32x32 square low-pass filter kernel was used for all cases where the algorithm performs spatial filtering.

Figure 3.6 depicts the profiles at n_x=152 through the absolute percent error image for each of the six cases. The profiles exclude a 16-pixel border lost to edge effects as described below. In Case 1, the percent error is very small and uniform across the image. No extra error is

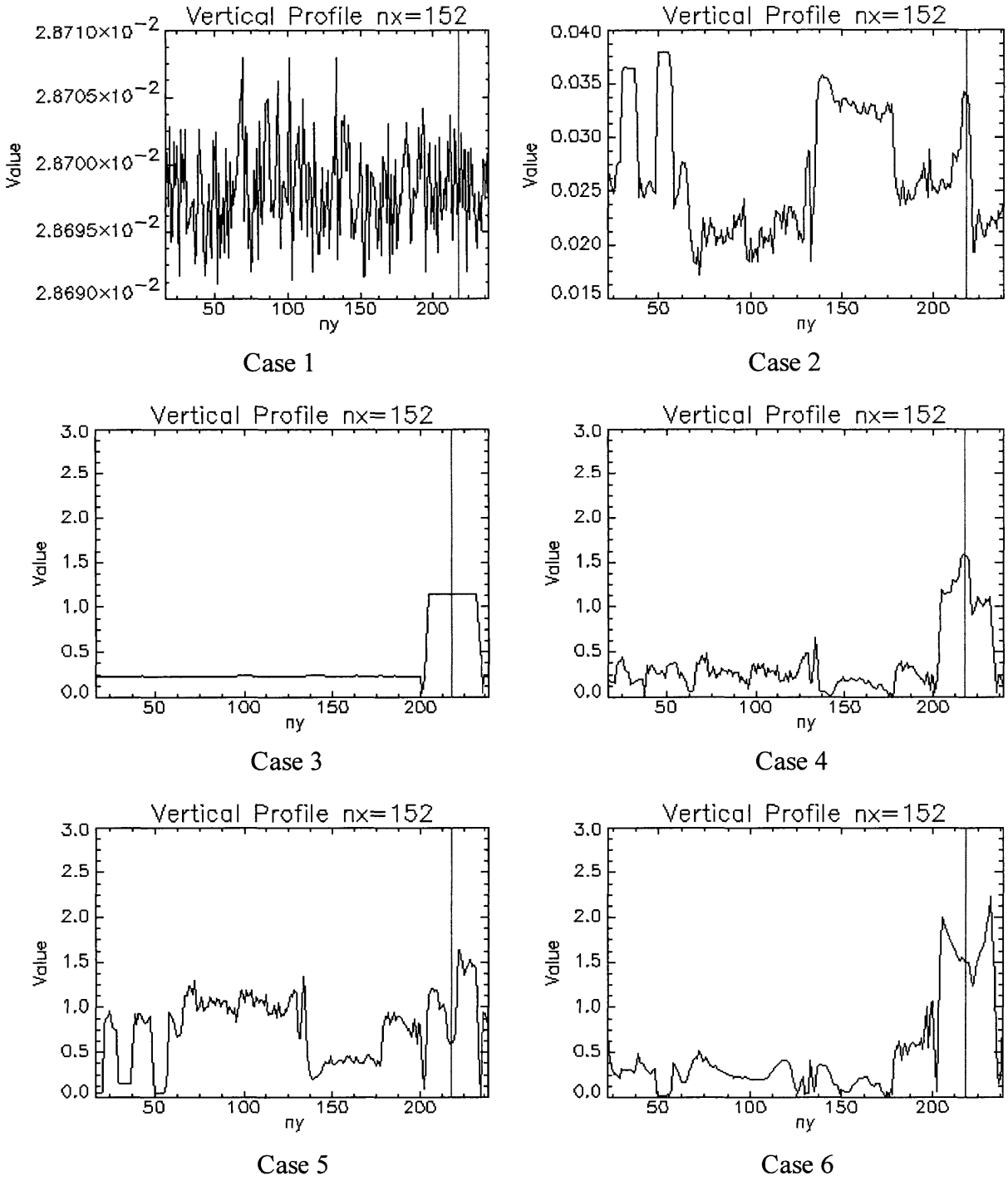


Figure 3.6. *Single channel image test results for Cases 1-6. These vertical image profiles plot the absolute percent error of the estimated surface reflectance compared to the actual reflectance image at n_x=152 , for each of the six cases.*

induced by the modification. In Case 2, the percent error is still quite small, with higher error over brighter pixels. Again, no extra error is induced by the modification. In case 3, the error is less than 0.2 percent for most of the image, but is larger (1.1 percent at its maximum) in the area near the modification where the filter kernel overlaps it. Case 4 has similar performance to Case 3, but with slightly higher errors (a few tenths of one percent higher), and more variability. Case 5 has overall more percent error than the previous cases, but with errors of 1.6 percent or less. Brighter pixels tend to have less error than dimmer ones. Finally, Case 6 has errors similar to Case 4, though slightly higher around the modified portion of the image. In all cases, errors are less than 2.2 percent, indicative of good performance. Testing was repeated in the face of additive Gaussian noise (AGN) of different variances and performance was found to be robust, with errors being dominated by the AGN as its variance dominated the Spatial SPIRE algorithms errors described above.

When implementing and applying the Spatial SPIRE algorithms that involve applying a low-pass filter, one will encounter two types of edge effects. The first is from the filtering operation itself, which can be viewed in the spatial domain as being caused by the convolution kernel extending beyond the edges of the image, or in the spectral domain due to the saw tooth effect that occurs when using FFTs to implement convolution, due to repeating the image as a two-dimensional discrete time Fourier series that will tend to have discontinuities at the edges between periods. This first edge effect can be minimized by simply extending or mirroring the edge pixels in a border around the image.

The second edge effect comes from the window function that cuts the sensed image from the observed scene. A slowly varying illumination across the scene will gain high frequencies components when windowed (Oppenheim, et al., 1999). Since the SPIRE algorithms in effect

use only low spatial frequencies to estimate the illumination, the lack of higher frequencies to meet the window edges will cause higher errors at the edges of the image. The solution to this is to collect an image larger than needed and then only use the portion of the estimated reflectance image inside an appropriate border (based on the size of the image and the bandwidth of the low pass filter) as the final result.

3.3.4 Comparison of ELM, ATREM, and Spatial SPIRE Algorithms

A third set of experiments was carried out to compare the performance of Spatial SPIRE algorithms to the ELM and ATREM atmospheric compensation algorithms. Since pixels of known reflectance are typically not available under operational conditions, this comparison is mainly aimed at comparing SPIRE to ATREM, with ELM used as a ground truth derived baseline. While SPIRE can work under conditions that ATREM cannot (single or multiple channels, cloudy conditions above sensor), the author felt it important to compare the Spatial SPIRE algorithm performance to that of an established technique known to the community. The ARM Site HYDICE radiance dataset was used directly for these experiments. Six HYDICE ARM Site data runs were selected from three different days of the ARM Site data collect.

Run	Date	Time (Local)	Altitude
07	6/24/97	12:26-12:28	6,087'
13	6/24/97	13:13-13:16	11,433'
26	6/24/97	14:24-14:27	11,410'
06	6/26/97	11:43-11:46	5,994'
22	6/27/97	07:37-07:39	6,077'
31	6/27/97	08:19-08:22	11,333'

Table 3.3. *ARM Site runs selected.*

Table 3.3 shows the times and altitudes of the six runs. All runs covered the area containing the test panels and buildings visible in Figure 3.4(a). Figure 3.7 shows single channel

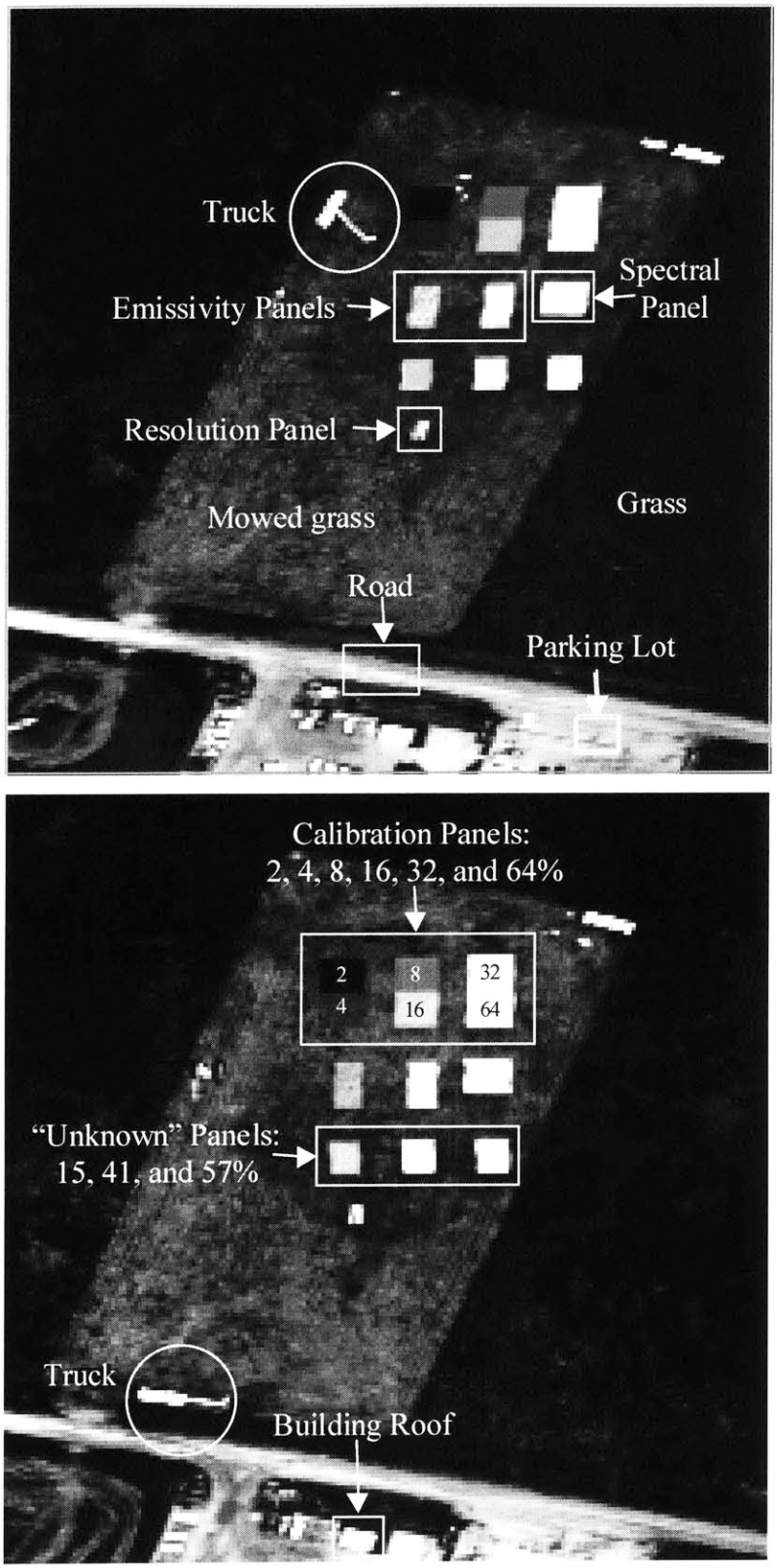


Figure 3.7. *Location of panels and other objects at ARM Site. Single channel images from runs on two different days are used to show the movement of the truck and other objects.*

images from Runs 07 and 06, with the locations of the various panels and materials of interest identified. Note that runs were done at different times of day including morning, noon, and mid-afternoon, which introduced shadows and different solar illumination angles in the data. The different altitudes of the runs caused image scale differences of a factor of two between some images. Also, the flight paths differed so that the images were rotated with respect to each other. In addition, the truck carrying the spectroradiometer appeared in different locations in each run, which introduced small changes from the prior, making this a suitable dataset to test SPIRE algorithms on. Some runs contained shadows due to low sun angle. While estimation of reflectance within the shadows was not tested, their effect on estimating the reflectance of non-shadowed areas was. Appendix B contains band images of each run.

3.3.4.1 ELM Processing of Test Data Set

ELM processing requires known spectra for pixels in the image. Ground truth measurements of spectra over the test panels were available with the ARM site image data. Ground truth spectra for the 2 percent and 64 percent panels, collected on 24 June 1997, were used to perform the ELM calibration. Sub-image cubes similar in spatial extent to the image in Figure 3.5(a) were extracted from the original radiance cubes. The sub-cubes were pre-processed to remove artifacts involving integer wrapping of some bright pixels into negative numbers. For each calibration panel, all interior pixels were selected and input to ENVI's Empirical Line Calibration (ELM) routine, which performs a linear regression to minimize the effects of noise across the uniform panels (ENVI User's Guide, 1997). All six run cubes were processed to estimate reflectance. Channels affected by water vapor absorption and negative radiance values were removed leaving a total of 73 spectral channels, as described in Chapter 4.

3.3.4.2 ATREM Processing of Test Data Set

ATREM Version 3.0 was used to estimate reflectance from each of the six run cubes. For each run, the entire image cube was processed with all 210 original HYDICE sensor channels included. The atmospheric parameters used in the input files for all six runs were:

```
1                          ! channel ratio parameters (0 or 1)
0.8650 3 1.030 3 0.940 7   ! .94 um water vapor band ratio parameters
1.050 3 1.235 3 1.1375 7   ! 1.14 um water vapor band ratio parameters
2                          ! atmospheric model (2=midlat sum)
1 1 1 1 1 1 1              ! gas selectors
0.34                       ! total column ozone amount (atm-cm)
1 23                       ! aerosol model and visibility (km)
```

After the estimated reflectance cubes were generated, sub-cubes matching the same spatial area of the ELM sub-cubes were extracted, and the channels affected by water vapor absorption and negative radiance values were removed as described in Chapter 4, leaving 73 channels.

3.3.4.3 Spatial SPIRE Processing of Test Data Set

Spatial SPIRE processing was preformed on the same sub-cubes extracted and integer wrap undone during the ELM processing. Channels affected by water vapor absorption and negative radiance values were removed before SPIRE processing, leaving a total of 73 spectral channels as described in Chapter 4. In addition, any negative radiance values were set to the minimum positive value in that channel.

Analysis of the ELM results from different days showed that both additive and multiplicative effects were essentially spatially uniform. This analysis involved measuring the differences between the low frequency components of the ELM images, as well as registering the images and measuring differences between them. Since ELM assumes uniform illumination conditions, any deviations from this assumption would show up as spatially varying differences

in the ELM images of the same scene. The differences found were very small, meeting Case 2 conditions. However, the Case 4 algorithm was used with a mean filter kernel instead of the non-iterating Case 2 algorithm, to allow comparison against similar iterative Spectral and Combined SPIRE algorithms in subsequent chapters.

A full-size-run 07 ELM generated reflectance cube was used as the source of prior information for the Case 4 SPIRE algorithm. To generate the actual prior cubes needed to process each of the six runs, ENVI's image-to-image warping routine using ground control points was used (*ENVI Users Guide*, 1997), utilizing the RST (Rotation, Scaling, and Translation) warping method with nearest neighbor re-sampling. Prior sub-cubes were then cut from the warped full-size-run 07 ELM cube, covering the same pixel values as the other run sub-cubes. No warping of the prior was needed for run 07. Appendix B contains channel images from each of the prior cubes.

3.3.4.4 Processing Results and Comparisons

Having run ELM, ATREM, and Spatial SPIRE on each of the six test data cubes from the six runs, we can now compare the results for individual pixels from each image of the same material. Figure 3.8(a)-(r) shows the estimated reflectance spectra generated by the ELM, ATREM, and Spatial SPIRE algorithms for 19 different types of pixels. Many of these pixels were selected from uniform materials such as panels, others are from materials such as grass and road which are fairly uniform but do display some mottled variation that could influence the run-to-run repeatability of reflectance estimates, since the pixels from each image are not registered exactly and have different ground coverage due to altitude variations.

For each pixel type, ELM reflectance estimates for the six runs are plotted at the top, with ATREM's estimates in the middle, and Spatial SPIRE at the bottom. Note that the vertical axis

of each reflectance plot is different from pixel to pixel, and was chosen to best highlight the variance between runs for that pixel and to compare the performance of the three algorithms. Similar plots with the vertical axis scaled from 0 to 1 can be found in Appendix C as Figure C.1, and provide a sense of how the performance on different pixels compared to each other. In each plot, five vertical dashed lines are drawn where channels were removed due to water absorption and other effects discussed in Chapter 4. The bands of channels between the vertical lines will be referred to as Bands 1-6, as denoted in Figure 3.8 (a).

Figure 3.8 (a)-(f) are of the six spectroradiometric calibration panels. These panels were intended to have spectrally uniform reflectance of 2, 4, 8, 16, 32, and 64 percent. Each panel consisted of a Dacron^{TM} substrate with an acrylic top coating. Both field and laboratory spectral reflectance data were collected from these panels. Immediately following deployment, the panels were cleared of accumulated dust using a leaf blower. Prior to the start of the 26 June overflights the panels were cleaned, following a thunderstorm which occurred on 25 June 1997. Cleaning was accomplished using a pressure washer provided by the ARM Site, and accumulated water was removed using a leaf blower.

The actual percent reflectance of these panels are often not the exact value of their name. For example, the 2 percent panel's actual reflectivity is approximately 5 percent. This is also true for the 15, 41, and 57 percent panels described below. These discrepancies are due to either inaccurate manufacturing or inaccurate ground truth collection by the spectroradiometer, and are not an artifact of the reflectance estimation algorithms used. The ELM estimates of Run 07 come closest to matching the ground truth spectra for these panels since the ground truth was collected close in time to the Run 07 data collection.

Figure 3.8 (g) depicts the results for the spectral panel, which was designed to provide

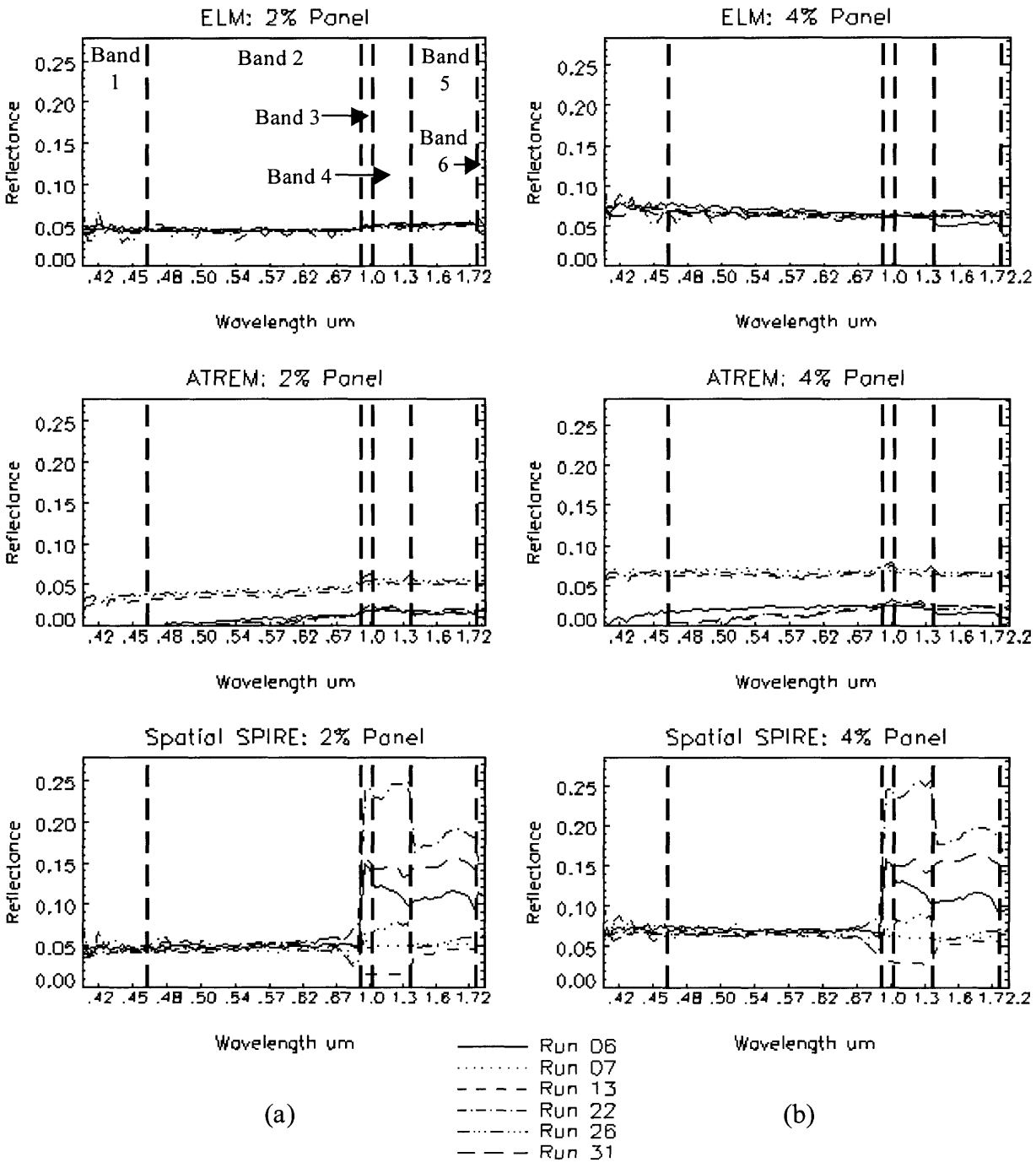


Figure 3.8 (a) and (b). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the 2 percent panel (a) and the 4 percent panel (b). There are six bands of contiguous channels left after dropping problem channels, defined as Bands 1-6 as depicted in the upper left plot of ELM 2 percent panel reflectance. Spatial SPIRE's poor performance at longer wavelengths is due to errors in estimating the additive noise a under low SNR conditions.*

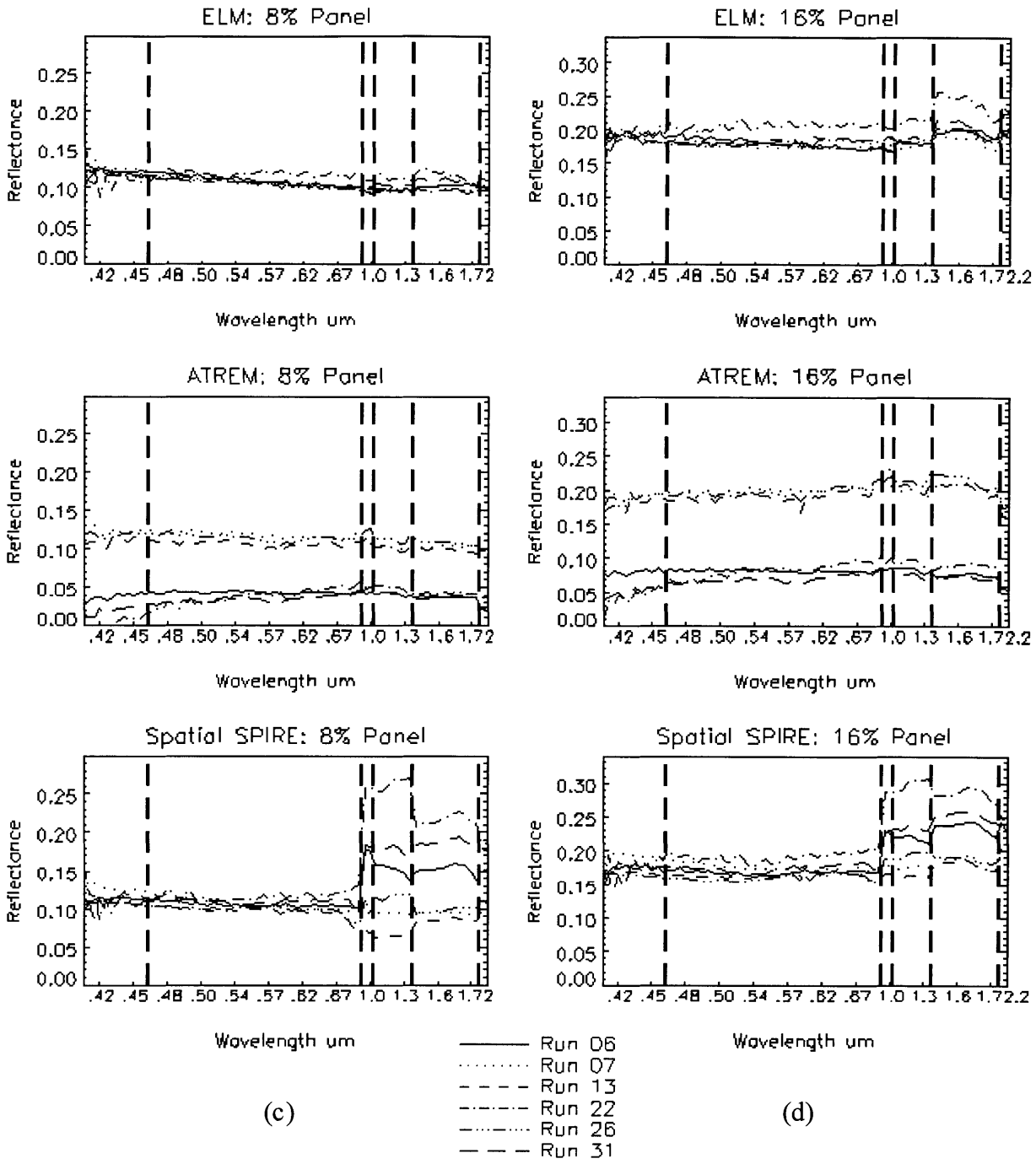


Figure 3.8 (c) and (d). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the 8 percent panel (c) and the 16 percent panel (d).*

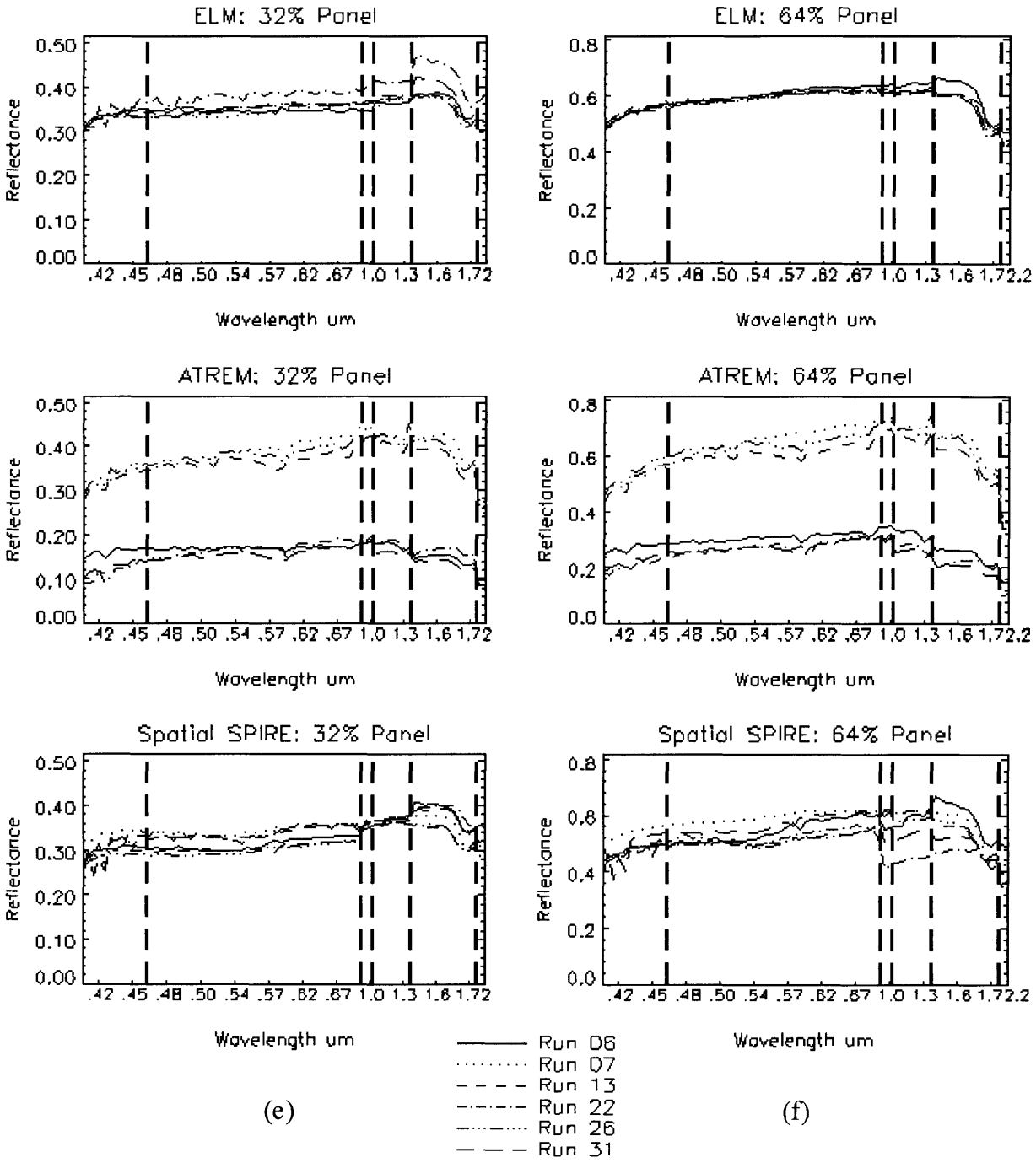


Figure 3.8 (e) and (f). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the 32 percent panel (e) and the 64 percent panel (f).*

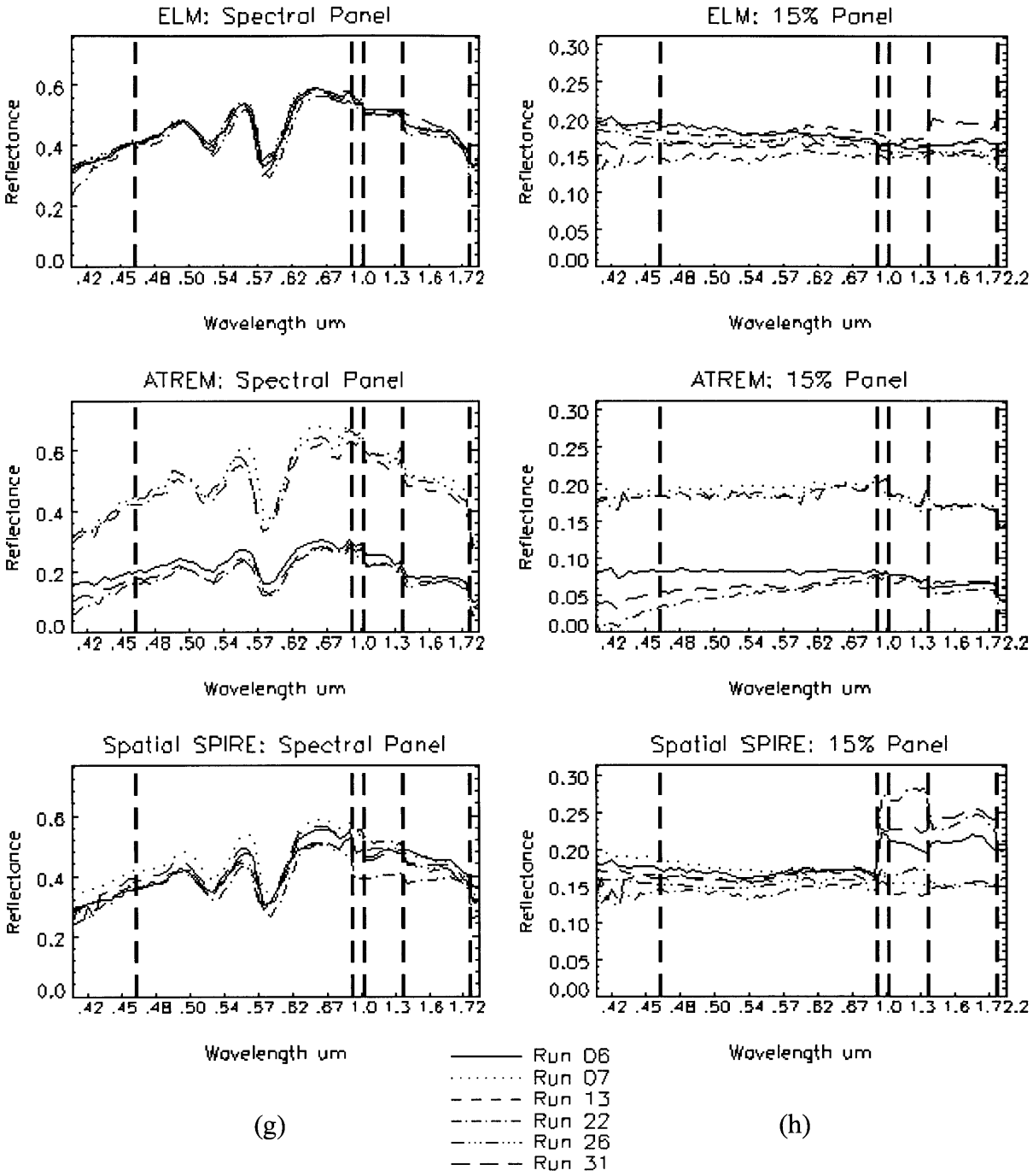


Figure 3.8 (g) and (h). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the spectral panel (g) and the 15 percent panel (h).*

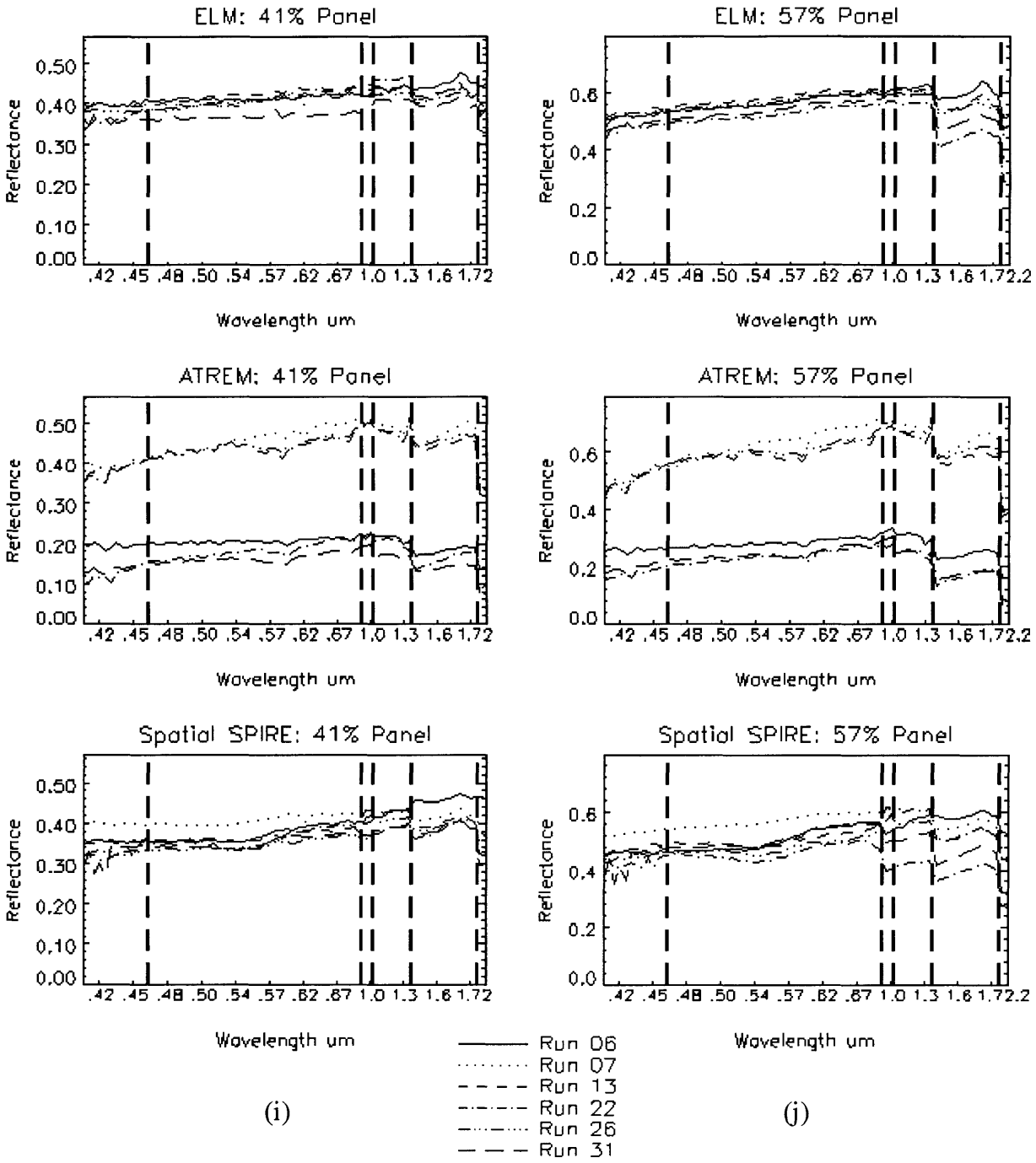


Figure 3.8 (i) and (j). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the 41 percent panel (i) and the 57 percent panel (j).*

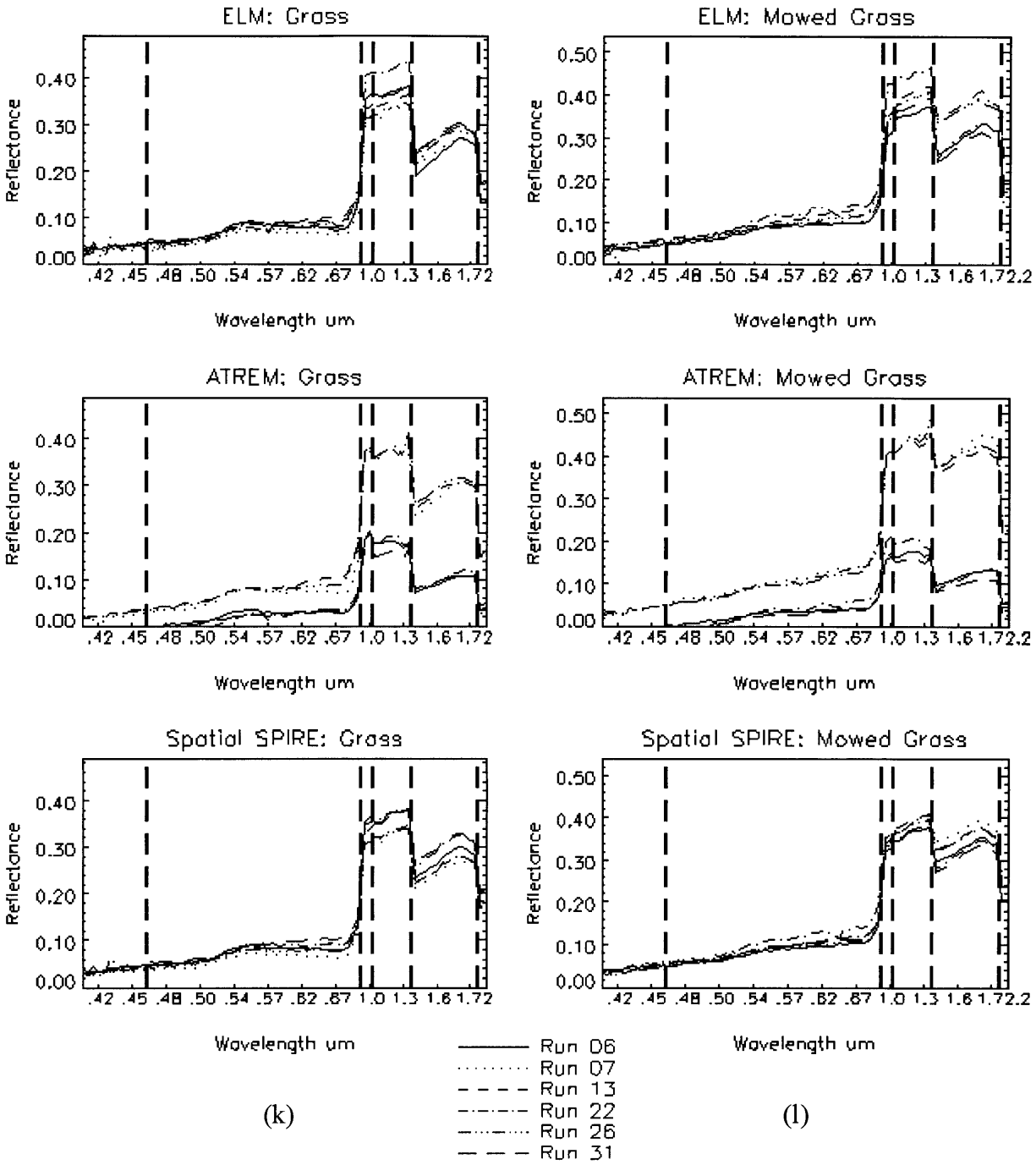


Figure 3.8 (k) and (l). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on grass (k) and mowed grass (l).*

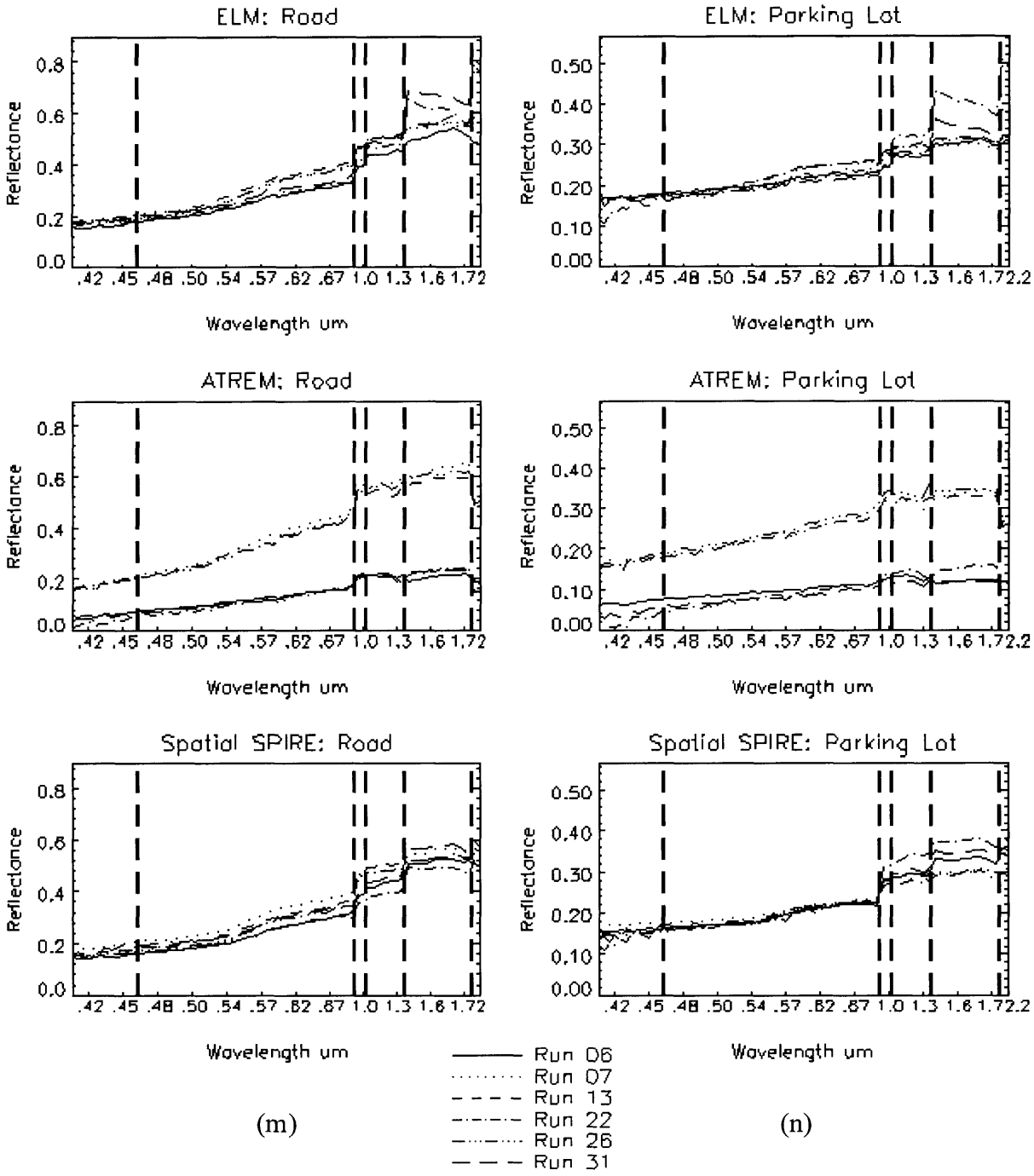


Figure 3.8 (m) and (n). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the road (m) and the parking lot (n).*

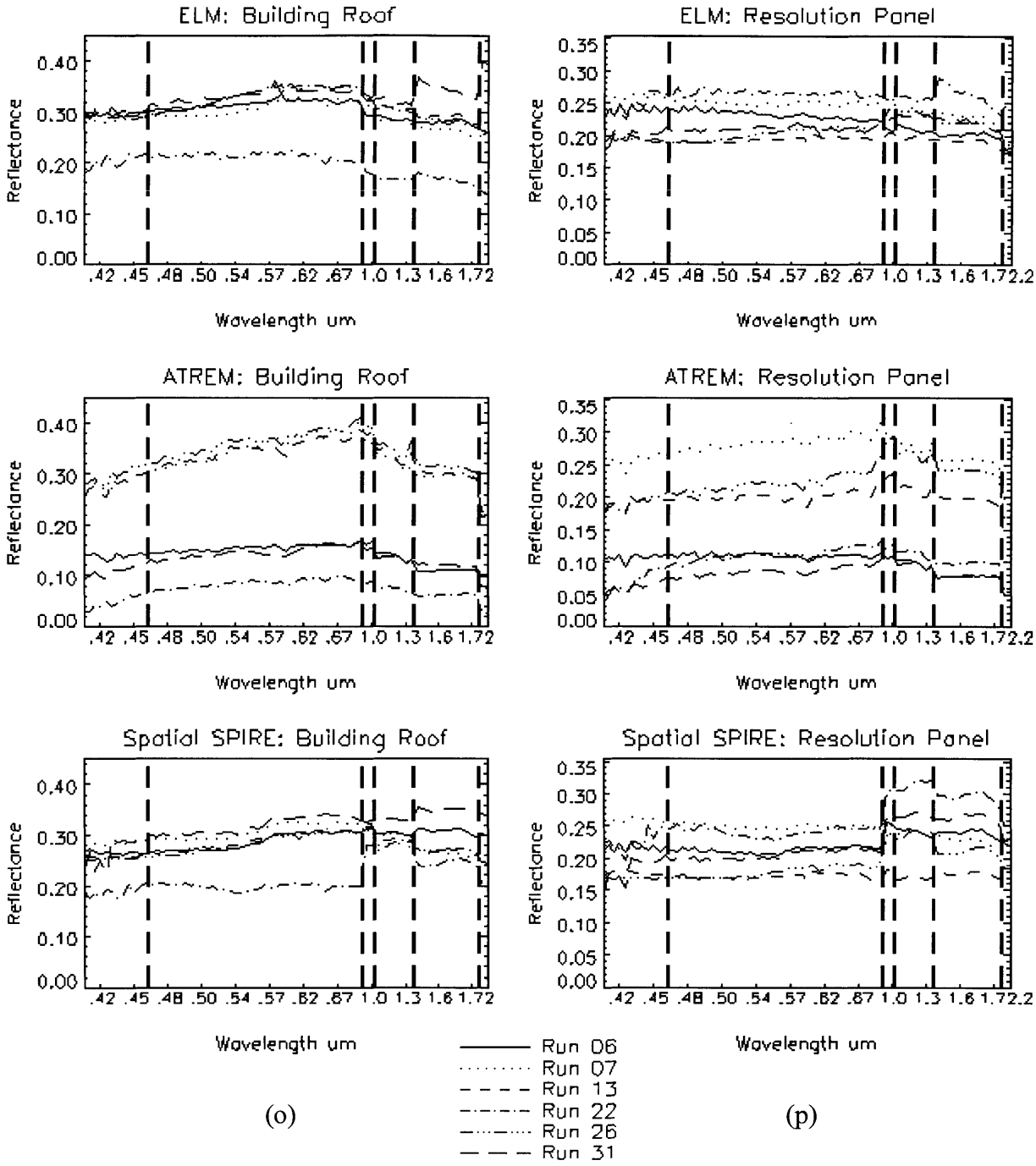


Figure 3.8 (o) and (p). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the building roof (o) and the resolution panel (p).*

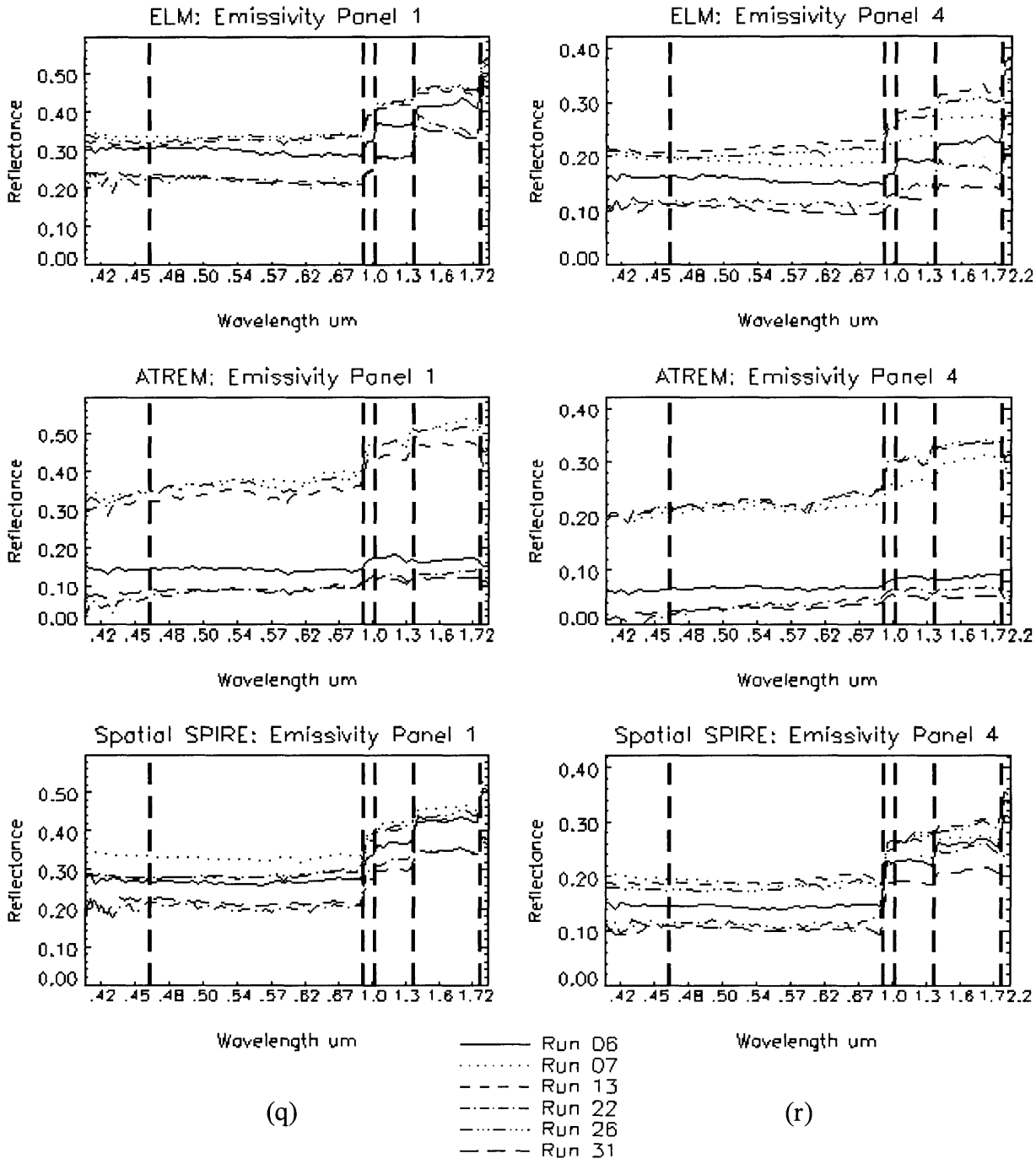
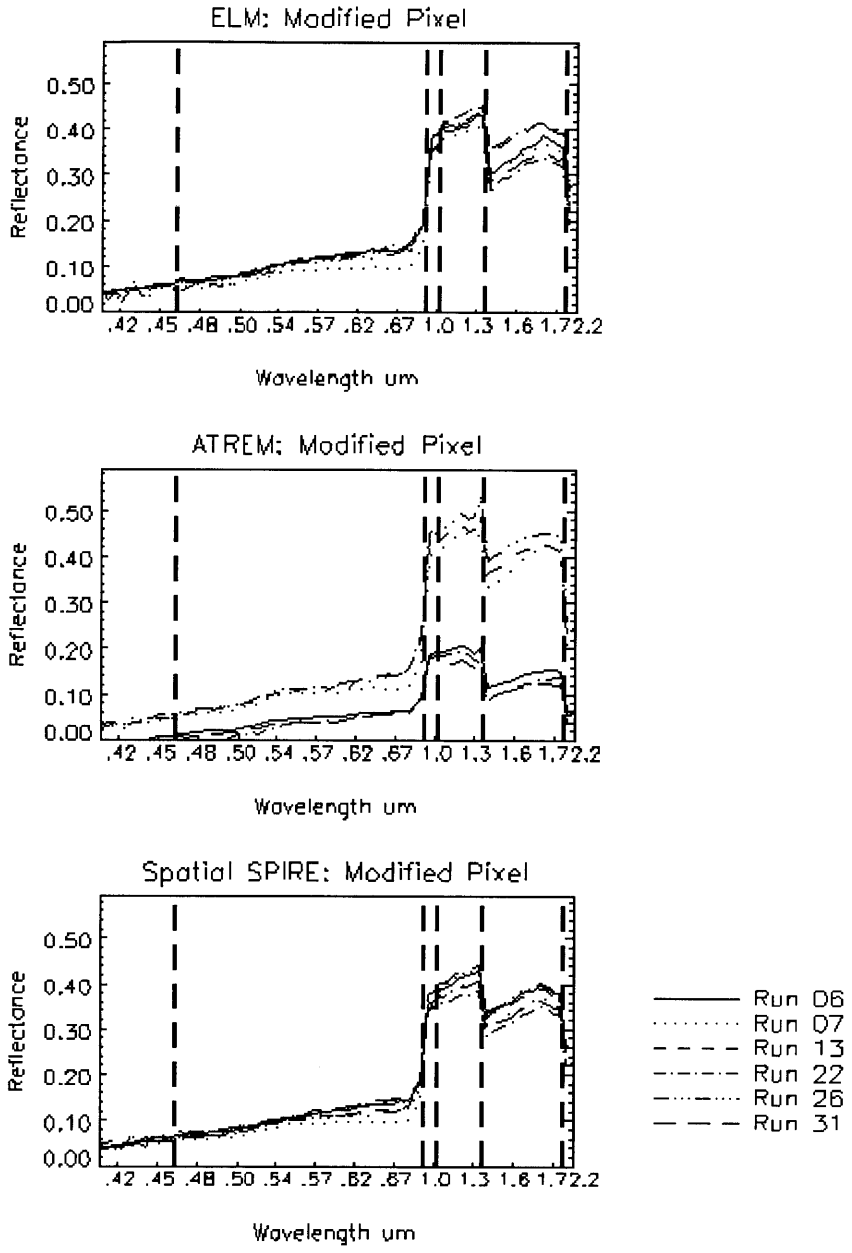


Figure 3.8 (q) and (r). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on emissivity panel 1 (q) and emissivity panel 4 (r).*



(s)

Figure 3.8(s). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for the modified pixel. The modified pixel was a truck in Run 07, and changed to mowed grass in all subsequent Runs. For Run 07, a similar mowed grass pixel is plotted for comparison.*

several sharp absorption features at known wavelengths for testing hyperspectral sensors. The panel was comprised of a natural polyester substrate with a nominally 64 percent reflective pigmented acrylic underlayer, and a non-pigmented acrylic top coat containing a suspended rare earth compound. This panel was also cleaned with a leaf blower following its initial deployment on 23 June 1997. The panel was cleaned with a pressure washer following the thunderstorm that occurred the evening of 25 June 1997.

Figure 3.8 (h)-(j) are of the three "unknown" reflectance panels, which were estimated to have uniform reflectances of 15, 41, and 57 percent. These panels were composed of an acrylic coating on a canvas fabric substrate that provides a uniform reflectance over the near ultraviolet to the short-wave infrared spectral region. The three panels were initially deployed on 23 June 1997 and remained throughout the collection period. Following the thunderstorm on 25 June 1997, each panel was pressure washed to remove accumulated dirt, then dried using a leaf blower and cotton rags.

Figure 3.8 (k) and (l) depict grass and mowed grass. Figure 3.8 (m)-(p) depict estimates for pixels on the gravel road, a parking lot, a building roof, and the resolution panel, which was a small panel intended to assess a sensor's spatial resolution capabilities. Figure 3.8 (q) and (r) are of the two emissivity panels. These panels were manufactured from materials that vary in emissivity and provide a flat spectral response over the 1-to 14-micrometer region. The basic coating is acrylic, over a canvas substrate. Nominal emissivity ranges from 0.6 to 0.9. The emissivity panels were cleaned using a pressure washer on 26 June 1997, with drying accomplished using a leaf blower and cotton rags. Finally, Figure 3.8 (s) depicts a pixel where a truck was present in the Run 07 image used to generate the priors, but which moved and left mowed grass in its place. This pixel will be referred to as the "modified" pixel.

The relative performance of ELM, ATREM, and Spatial SPIRE can best be measured in two ways. The first is to measure the variance or standard deviation of the estimates at each pixel for the six runs. This provides insight into the repeatability of the estimates. Second, if we treat the ELM results as ground truth, we can measure the error between the mean of the six ELM estimates and the mean of the six estimates from either ATREM or Spatial SPIRE to obtain insight into any bias or mean error in their estimates.

Figure 3.9 shows the scatter plot of mean and standard deviation performance of Spatial SPIRE, ELM, and ATREM for estimation of surface spectral reflectance for all of the 19 pixel types selected. The horizontal axis represents the average standard deviation over all the spectral channels, where the standard deviation in each channel was calculated over the reflectance estimates of all six runs (Runs 06-31). The vertical axis is the root mean squared (RMS) error over all the spectral channels for the mean reflectance estimate minus the mean reflectance estimate of ELM as ground truth, which is why ELM has zero RMS error. We see that in general, Spatial SPIRE has better standard deviation and better RMS error performance than ATREM. Spatial SPIRE also has about the same standard deviation performance as ELM. In Appendix C, Table C.1 lists the average channel standard deviation over the six runs in each band for each algorithm and Table C.2 lists the same values but as a percentage of the mean estimated reflectance over the six runs.

Several interesting results are also demonstrated by the plots in Figure 3.8(a)-(s). For nearly all of the pixels, including the modified pixel, the Spatial SPIRE reflectance estimates agree much better with the ELM than do the ATREM results. If we consider ELM as ground truth, then Spatial SPIRE consistently demonstrates better performance than ATREM. Note that the ATREM estimates tend to group into two clusters with the three spectra from runs 06, 22,

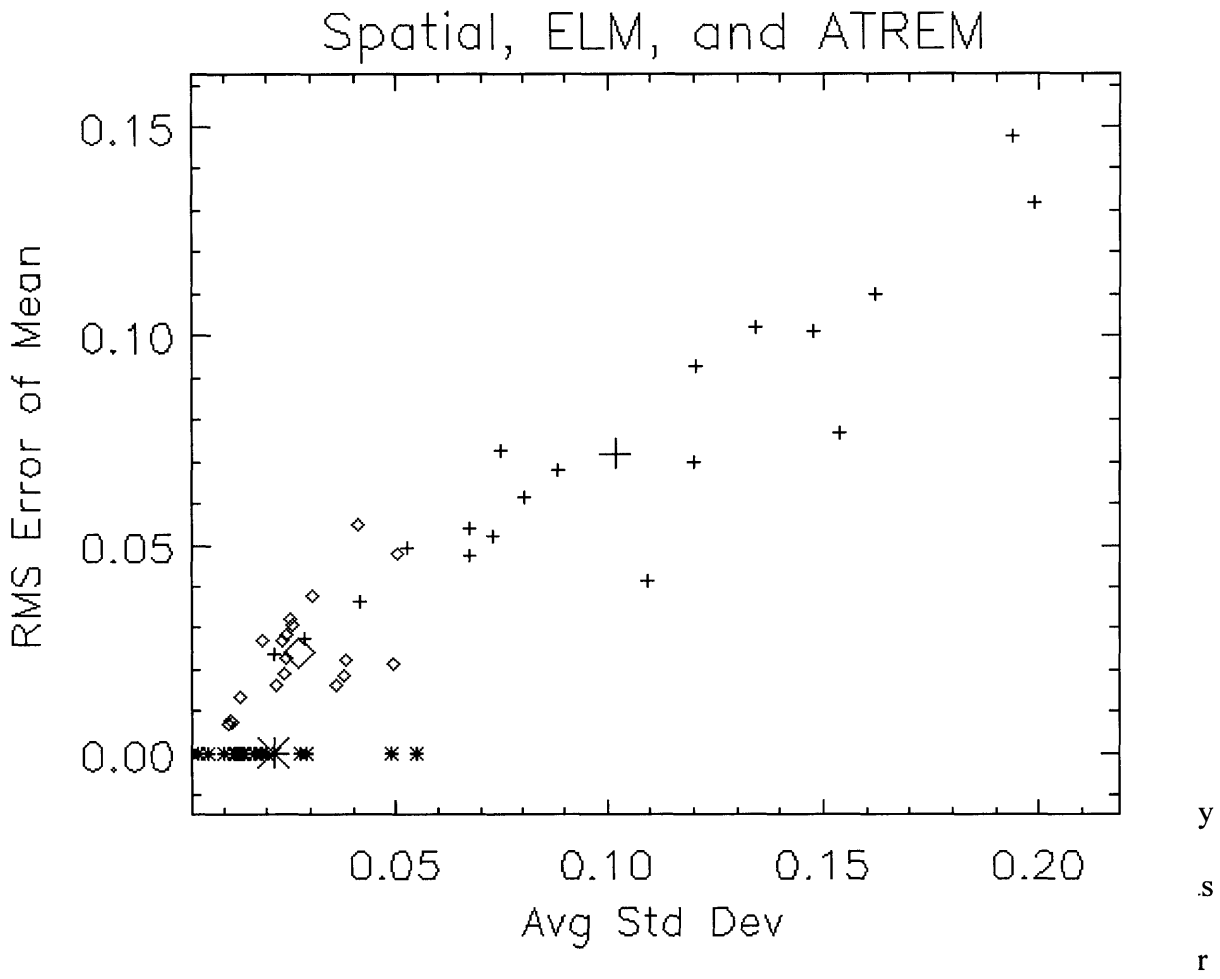


Figure 3.9. *Scatter plot of mean and standard deviation performance of Spatial SPIRE (diamonds), ELM (*), and ATREM (+) for estimation of surface spectral reflectance for all of the 19 pixel types selected. The horizontal axis represents the average standard deviation over all the spectral channels, where the standard deviation in each channel was calculated over the reflectance estimates of all six runs (Runs 06, 07, 13, 22, 26, and 31). The vertical axis is the RMS error over all the spectral channels for the mean reflectance estimate minus the mean reflectance estimate of ELM as ground truth, which is why ELM has zero RMS error. The larger symbols represent the mean of the points plotted with that symbol. We see that Spatial SPIRE has better standard deviation and RMS error performance than ATREM.*

and 31 significantly lower than the other three.

The performance of ATREM caused some concern so the same runs were processed using the AFRL/MODTRAN code, which confirmed the ATREM results. All-sky photographs ATREM reflectance estimates on these days due to the reduced illumination reaching the ground.

This also underscores the problems physics-based codes can have with overcast clouds, which do not affect the SPIRE algorithms. Similar problems with ATREM were noted in (Sanders, et al., 1998).

The second result of interest is apparent in the plots of the low reflectivity panels such as the 2, 4, and 8 percent panels. While there is excellent agreement between ELM and Spatial SPIRE in Bands 1 and 2, Spatial SPIRE has high variance in Bands 3-6. For pixels with higher reflectance, this is not true. This variance in the Spatial SPIRE results are due to errors in estimating a by the Case 4 algorithm that disproportionally affect low signal values. This is discussed further in section 3.4.4.

3.4 Noise Analysis

HYDICE sensor data contains noise from several different sources (Nischan, et al. 1999). These include detector noise, periodic background "heartbeat", calibration errors, channel center wavelength (spectral) drift and jitter, and interaction between the spectral jitter and the HYDICE thermal background suppression filter. In addition, noise in the prior and due to registration errors, deviations from the assumptions of small area reflectance changes, and slowly varying multiplicative and additive noises also affect SPIRE algorithms. A detailed analysis of each type of noise and it's effects on Spatial SPIRE algorithms is beyond the scope of this thesis.

Instead our goal in this section is to understand which general types of noise (sensor, calibration, prior, etc.) affect Spatial SPIRE reflectance estimates, and for those that do, understand the behavior well enough to make recommendations for overcoming or minimizing the effects. For those noises that do have a significant effects, we will assume simple models for the noise sources and derive equations for reflectance estimation errors based on these models. When appropriate, we will generate numerical results from these equations to demonstrate the

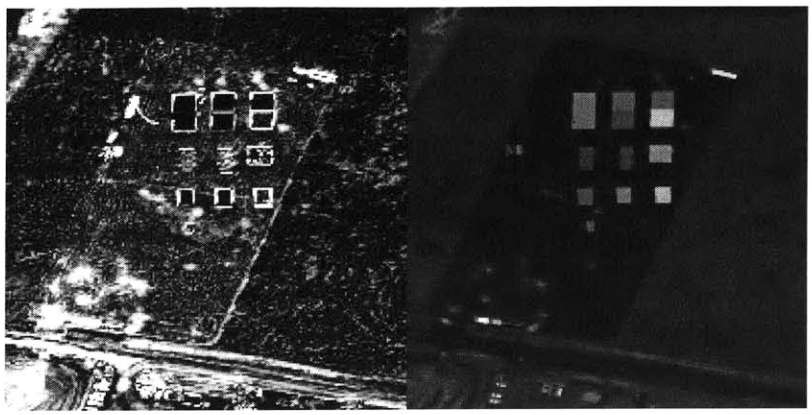
effects on Spatial SPIRE algorithm performance. These error equations and results provide guidance for how to perform specific noise analyses for different sensors and applications. We will also gauge the effect of the actual total noise in the HYDICE data and priors by presenting results from running the Spatial SPIRE with the normal (noisy) prior and a perfect prior based on ELM results.

3.4.1 Registration Noise

Imperfect registration of the image and its prior introduces noise that can effect reflectance estimation. If we view the spectrum at each pixel as a vector in a normed vector space (one in which a notion of length is defined for the elements of the space), then we can define the distance between the pixels at $[n_{x_o}, n_{y_o}]$ from two different images as the length of the vector difference between the two pixel vectors. For two reflectance images r_1 and r_2, the distance $d[n_{x_o}, n_{y_o}]$ between the pixel spectra $\mathbf{r}_1[n_{x_o}, n_{y_o}, n_\lambda]$ and $\mathbf{r}_2[n_{x_o}, n_{y_o}, n_\lambda]$ is defined as:

$$d[n_{x_o}, n_{y_o}] = \sqrt{\sum_{n_\lambda=1}^{N} \left(\mathbf{r}_1[n_{x_o}, n_{y_o}, n_\lambda] - \mathbf{r}_2[n_{x_o}, n_{y_o}, n_\lambda]\right)^2} \tag{3.30}$$

Performing this calculation at each $[n_{x_o}, n_{y_o}]$ results in a distance image. In Figure 3.10(a), we see the distance image between the prior used for Run 06 and the ELM estimated reflectance for Run 06. Brighter pixels correspond to large distances, we see that misregistration errors generate larger distances at the edges of the test panels. In addition, bright differences are apparent where the truck to the left of the calibration panels has moved, and in some of the grass and road areas, that cause of which is not understood due to lack of ground truth at those locations. If the ELM reflectance estimates are regarded as ground truth, then this distance



(a) (b)

Figure 3.10. *Distance images of prior reflectance with respect to ELM estimated reflectance (a) for Run 06. The distance value at each pixel is equal to the length of the vector difference between the pixel spectra of the two images at that pixel. Brighter distance image pixels have longer distances and highlight changes since the prior and registration errors in (a). The distance image between Spatial SPIRE and ELM estimated r for the same run is shown in (b), where misregistration errors did not affect the results.*

image represents the registration errors present in the prior reflectance.

In Figure 3.10(b) is depicted the distance between the Spatial SPIRE reflectance estimates and ELM estimated reflectance. We see that the misregistration noise has not affected the Spatial SPIRE results. This is because misregistration noise will contain mostly high spatial frequencies and be localized to object edges. Since only low spatial frequencies are used from the prior, misregistration noise will not affect Spatial SPIRE algorithms, so long as the misregistration does not cause major changes in the low spatial frequency content of the prior compared to the current scene. Since misregistration noise does not have a significant impact on Spatial SPIRE results, there is no need to develop equations for the errors involved.

3.4.2 Prior, Sensor, and Calibration Noise

In this section we derive the effects of prior and sensor noise sources and give some

examples of their effects. We are working in a single channel image, so all variables are a function of only n_x and n_y. We assume that there are three main types of noise: sensor, calibration, and prior. A single noise free, calibrated radiance image i is given by:

$$i = rm + a \tag{3.31}$$

Where r is surface reflectance and m and a are the multiplicative and additive illumination effects. We assume that there is an additive sensor noise s which is typically the sum of several noise sources. We will assume that s is a random Gaussian variable across n_x and n_y with mean μ_s and variance σ_s^2 across the image: $s \sim N(\mu_s, \sigma_s^2)$. Calibration noise is assumed to consist of an additive offset noise, or error, c_{offset} and a multiplicative gain noise c_{gain}. With calibration and sensor noise sources taken into account, our single calibrated radiance image can be viewed as:

$$i = c_{gain}(rm + a + s) + c_{offset} \tag{3.32}$$

Typically, a single gain and offset correction for each detector is applied to each individual image. In a single detector (per spectral channel), scanning push-broom sensor, c_{gain} and c_{offset} will tend to be constant across the image, with slowly varying changes possible due to drifts away from any in flight calibration data collection.

In a push-broom sensor with a linear array, c_{gain} and c_{offset} are constant or slowly varying in the in-flight spatial dimension by the same arguments as above, but could vary in the cross track due to individual detector differences. Assuming that such differences can be well compensated for using laboratory measurements, any calibration noise across a linear array will

tend to be constant, or slowly varying if changing thermal conditions cause optical variations or geometric effects.

Since the Spatial SPIRE algorithms employ spatial filtering that removes low spatial frequencies, we can discuss just the case where m, a, c_{gain}, and c_{offset} are constant across the image without loss of generality (so long as the low spatial frequency support of c_{gain} and c_{offset} is within that of m and a). Therefore, we will assume that both c_{gain} and c_{offset} are unknowns that are constant across the image.

The removal of a is accomplished by subtracting the mean of i from itself (high pass filtering):

$$\begin{aligned} i - mean(i) &= c_{gain}(rm + a + s) + c_{offset} \\ &\quad - mean(c_{gain}(rm + a + s)) - mean(c_{offset}) \\ &= c_{gain}(rm + s_{zm} - mean(rm)) \end{aligned} \tag{3.33}$$

Where $s_{zm} \sim N(0, \sigma_s^2)$ is simply a zero mean version of s. Assuming that we have found the best value of $mean(rm)$ to add back in (using the iterative Case 4 algorithm), we proceed to take the log and use the Case 3 algorithm to solve the multiplicative-only case. If there is an error in our estimation of $mean(rm)$, then this adds a constant error e_a to the resulting estimated rm image:

$$\begin{aligned} \hat{rm} &= i - mean(i) + est(mean(rm)) \\ &= c_{gain}(rm + s_{zm} + e_a) \end{aligned} \tag{3.34}$$

Next we move into log space in an attempt to separate r and m:

$$\log \hat{rm} = \log c_{gain} + \log(rm + s_{zm} + e_a) \approx \log r + \log m \tag{3.35}$$

The addition of s_{zm} and e_o to rm causes a non-linear effect on the image in log space that can be treated as an added noise $\log e_{sanl}$:

$$\log e_{sanl} = \log(rm + s_{zm} + e_a) - \log rm \tag{3.36}$$

So that:

$$\log(rm + s_{zm} + e_a) = \log rm + \log e_{sanl} \tag{3.37}$$

Substituting (3.37) into (3.35) yields:

$$\begin{aligned} \log \hat{rm} &= \log c_{gain} + \log rm + \log e_{sanl} \\ &= \log r + \log m + \log c_{gain} + \log e_{sanl} \end{aligned} \tag{3.38}$$

We now high pass filter in log space to remove the $\log m$ term, which is equivalent to subtracting the mean:

$$\begin{aligned} \log \hat{rm} - mean(\log \hat{rm}) &= \log r + \log m + \log c_{gain} + \log e_{sanl} \\ &\quad - mean(\log r + \log m + \log c_{gain} + \log e_{sanl}) \\ &= \log r - mean(\log r) + \log e_{sanl} - mean(\log e_{sanl}) \end{aligned} \tag{3.39}$$

Since $\log m$ and $\log c_{gain}$ are both constant, they are eliminated by this filtering step. At this point it is noteworthy that all of the calibration noise, both gain and offset, have been removed. Therefore, calibration noise does not affect the Spatial-Only SPIRE algorithms single image reflectance estimation, a trait shared with the Empirical Line Method (ELM). Physics-

based codes such as ATREM require radiometrically calibrated data and are therefore sensitive to such calibration noises.

Note that in (3.39), the last two terms on the right are equivalent to a zero-mean version of $\log e_{sanl}$. We therefore define a new error term:

$$\log e_{sanl-zm} = \log e_{sanl} - mean(\log e_{sanl}) \tag{3.40}$$

And (3.39) becomes:

$$\log \hat{rm} - mean(\log \hat{rm}) = \log r - mean(\log r) + \log e_{sanl-zm} \tag{3.41}$$

The next step is to restore our best estimate of $mean(\log r)$ by adding $mean(\log r_p)$ to (3.41), where r_p is the prior reflectance. If we assume that there have been no changes in the imaged scene since obtaining r_p, then the prior r_p will have some noise, or error, e_p with respect to the true reflectance:

$$e_p = r_p - r \tag{3.42}$$

We therefore are restoring:

$$mean(\log r_p) = mean(\log(r + e_p)) \tag{3.43}$$

Again, the subtraction of e_p will have a nonlinear effect on the term $mean(\log r_p)$ which we define as a second nonlinear noise which is spatially constant:

$$\log e_{pnl} = mean(\log(r + e_p)) - mean(\log r) \tag{3.44}$$

So that:

$$mean(\log r_p) = mean(\log(r + e_p)) = mean(\log r) + \log e_{pnl} \tag{3.45}$$

Adding (3.45) to (3.41) yields:

$$\begin{aligned} &\log \hat{rm} - mean(\log \hat{rm}) + mean(\log r_p) \\ &\quad = \log r - mean(\log r) + \log e_{sanl-zm} + mean(\log r_p) \\ &\quad = \log r - mean(\log r) + \log e_{sanl-zm} + mean(\log r) + \log e_{pnl} \\ &\quad = \log r + \log e_{sanl-zm} + \log e_{pnl} \end{aligned} \tag{3.46}$$

After taking the exponential, we solve for reflectance, which as expected, is the exponential of the log of r, perturbed by two multiplicative noise terms:

$$\begin{aligned} \hat{r} &= \exp(\log r + \log e_{sanl-zm} + \log e_{pnl}) \\ &= r e_{sanl-zm} e_{pnl} \end{aligned} \tag{3.47}$$

Expressing the reflectance error as an additive noise, we have:

$$e_{\hat{r}} = \hat{r} - r = r e_{sanl-zm} e_{pnl} - r = r\left(e_{sanl-zm} e_{pnl} - 1\right) \tag{3.48}$$

Where the two noise terms are defined as:

$$\begin{aligned} e_{sanl-zm} &= \exp(\log e_{sanl} - mean(\log e_{sanl})) \\ &= \exp(\log(rm + s_{zm} + e_a) - \log rm \\ &\qquad - mean(\log(rm + s_{zm} + e_a) - \log rm) \end{aligned} \tag{3.49}$$

$$e_{pnl} = \exp\left(mean(\log(r + e_p)) - mean(\log r)\right) \tag{3.50}$$

Both $e_{sanl-zm}$ and e_{pnl} cause no error in $\hat{r}$ if they are equal to 1 (0 in log space). Figure

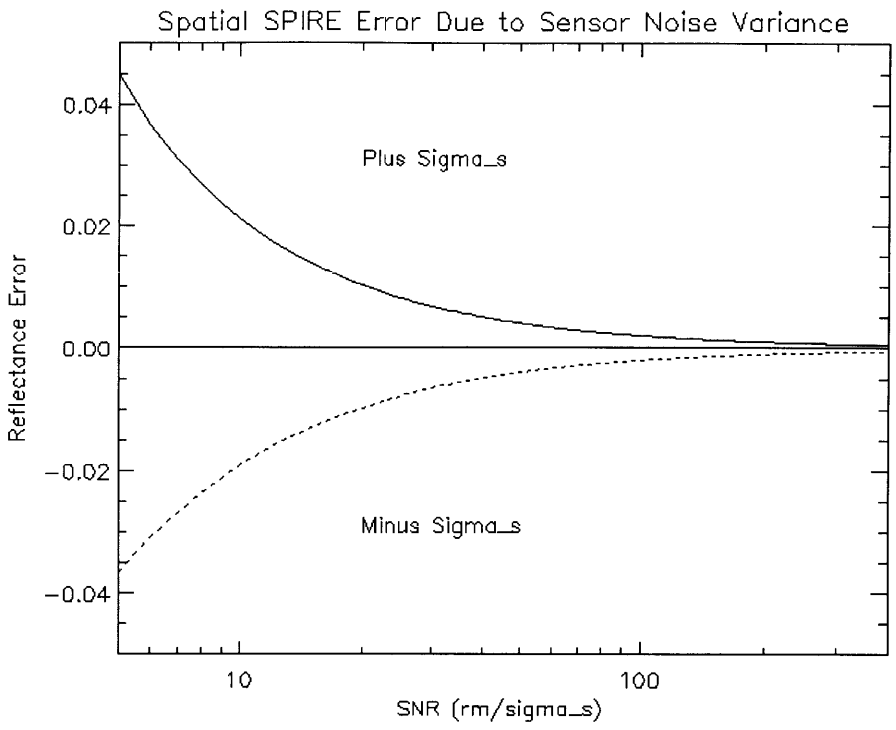


Figure 3.11. *Spatial SPIRE reflectance error at a single pixel of reflectance r=0.20 caused by sensor noise standard deviation σ_s (sigma_s) versus signal-to-noise ratio (SNR=rm/σ_s) in the multiplicative noise only case (a=0). The effect of the sensor noise is non-linear and plots for the noise s=+σ_s and s=-σ_s are shown. A value of m=600 Watts/m^2/sr/μm was used at all pixels.*

3.11 depicts $e_{\hat{r}}$ as a function of $SNR = rm/\sigma_s$ when $e_{pnl} = 1$, so that $e_{\hat{r}}$ is caused purely by sensor noise s via $e_{sanl-zm}$. We have assumed that $a = 0$ and therefore $e_a = 0$, and that $r = 0.20$ and $m = 600$ Watts/m^2/sr/μm for all pixels in the image. The two curves represent the effect on a single pixel at $[n_x, n_y]$ of $s[n_x, n_y] = +\sigma_s$ and $s[n_x, n_y] = -\sigma_s$ since these will have different nonlinear effects on $e_{\hat{r}}$. We see that errors cause by sensor noise can be overcome by increasing SNR. For the Run 06, SNR (see Figure 3.15) is above 3000 for most channels in Bands 1 and 2, above 1000 in Bands 3 and 4, and above 100 in Bands 5 and 6, resulting in reflectance errors less than 0.005.

Figure 3.12 plots the effect of the prior noise e_p on $e_{\hat{r}}$ when $e_{sanl-zm} = 1$. Again, $r = 0.20$

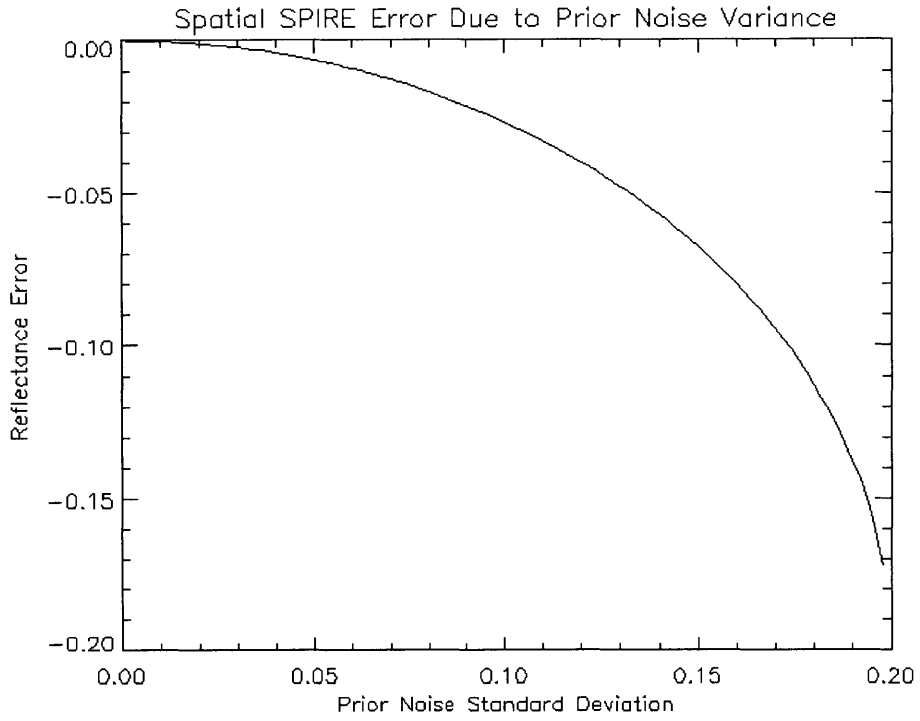


Figure 3.12. *Reflectance error at a single pixel of reflectance r=0.20 caused by prior noise variance σ_p in the multiplicative noise only case (a=0). A value of m=600 Watts/m^2/sr/μm was used at all pixels and the prior noise was assumed to be Gaussian with zero mean.*

and $m = 600$ Watts/m^2/sr/μm for all pixels in the image. In Figure 3.12 we assume that $\mu_p = 0$ so that $e_p \sim N(0, \sigma_p^2)$. Since $r_p = r + e_p$ would have been forced to have values between 0 and 1, our range of standard deviations considered is $\sigma_p = 0$ to 0.8. For the priors used in this thesis, $\sigma_p \simeq 0.005$, causing very small reflectance errors. For the case when a mean error μ_p exist in r_p, this results in the same error being introduced into $\hat{r}$.

3.4.3 Violation of Spatial SPIRE Assumptions

The two basic assumptions that Spatial SPIRE relies on are that any changes since the prior are small in area compared to the scene and that the multiplicative and additive noise effects are band limited to lower spatial frequencies. Together, these two assumptions are

equivalent to having no overlap between the spatial frequency content of the illumination noise and reflectance changes in the image.

Figure 3.13 shows the effect of violating these assumptions by having increasingly larger and larger reflectance changes until they effect the mean of the image. A simulated single channel prior image of uniform 0.30 reflectance was modified with a new reflectance of value 0.40. The size of this modification was varied from 0-100 percent of the image area to generate 101 test images. This was then repeated with modification reflectances values of 0.50 and 0.60. A uniform multiplicative noise of m=5 was then applied to all of the test images. The resulting simulated radiance images were processed using the Case 1 algorithm with the original uniform 0.30 reflectance image as the prior image. The absolute percent error in the estimation of one of the modified reflectance pixels is plotted in Figure 3.13 versus the percent area of the change. We see that the percentage error quickly grows to several percent as the area covered by the change grows to 5-10 percent of the scene area.

3.4.4 Perfect versus Normal Priors

To gain insight into the overall effects the noise sources present in the prior on the results of Figure 3.8, the Spatial SPIRE processing of Section 3.3.4.3 was redone using perfect priors consisting of the ELM reflectance estimate for each test cube. In addition, the ELM-estimated additive a vectors were subtracted to create "m-only" test cubes which were also processed with perfect priors and those generated from Run 07. (In the multiplicative-noise-only case, Spatial SPIRE duplicates ELM performance exactly when run with a perfect prior.) Figure 3.14(a) shows these four results for the 2 percent panel. Here we see direct confirmation that the excessive variance in Bands 3-6 for low reflectance materials is directly related to the presence of prior noise, and that the problem is introduced when estimating a.

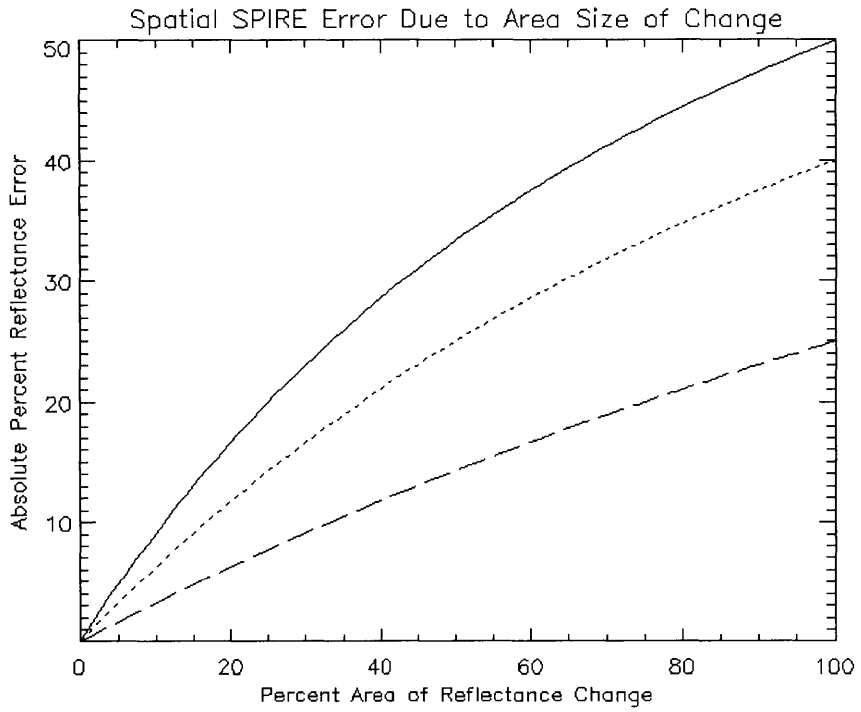


Figure 3.13. *Absolute percent error in estimated reflectance for a change from 0.3 reflectance to 0.4 (---), 0.5 (¨), and 0.6 (—) versus percentage change in area. The unmodified image had a uniform reflectance of 0.3 for all pixels.*

Any error in estimating *a* has a larger percentage effect on a low reflectivity material than a higher one. Upon moving into log space, this will have a much larger effect on a low reflectivity material's log reflectance value. In channels with very low signal-to-noise ratio (SNR), prior noise will have a large effect on the log of the prior reflectance as well. These two effects combine to create the large errors seen in the longer wavelength channels for low reflectivity materials. Figure 3.15 shows the SNR of each channel as measured over the 64 percent panel on Run 06. We see that the SNR is lower in Bands 3-6 than in Bands 1 and 2, and will be especially low for a low reflectance material like the 2 percent panel. The fact that the effect is worse on the runs with cloud and haze indicates that the problem is caused by low SNR, which can be overcome by a sensor design that achieves higher SNR. In Figure 3.14(b) the

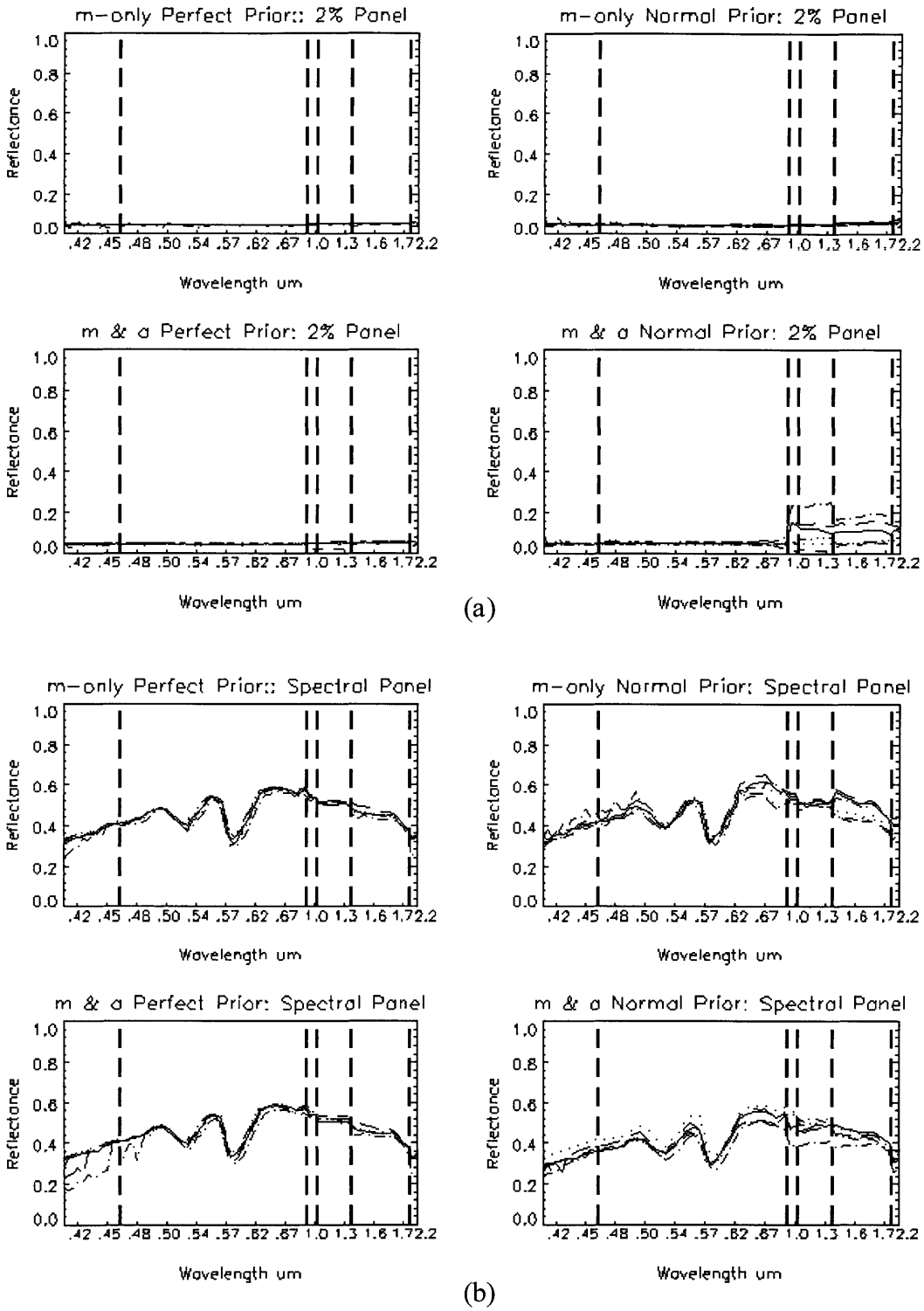


Figure 3.14. *Spatial SPIRE Case 3 (m-only) and Case 4 (m & a) reflectance estimates for all six runs using a perfect prior and a normal (noisy) prior, for the 2 percent panel (a) and the spectral panel (b). The longer wavelength errors on the 2 percent panel are caused by errors in estimating the additive noise a under low SNR conditions.*

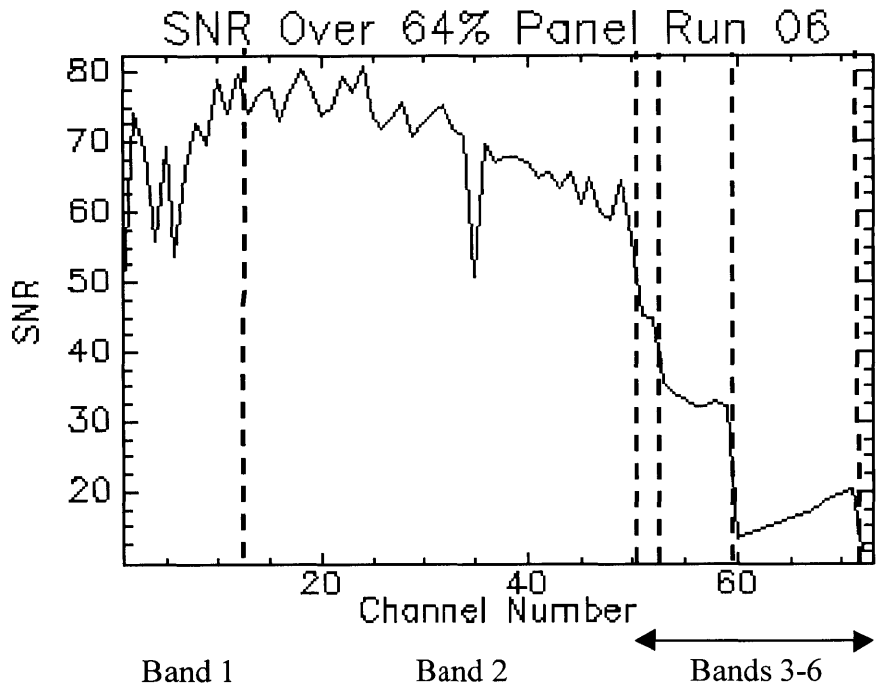


Figure 3.15. *Signal-to-noise ratio (SNR) over the 64 percent spectroradiometric calibration panel during Run 06. The SNR was calculated in each channel as the mean divided by the standard deviation over all pure (non-edge) pixels on the panel. The SNR is lower in the longer wavelengths of Bands 3-6 than in Bands 1 and 2.*

same plots are shown for the spectral panel. We see an increase in variance overall due to a noisy prior, but it is not related to the estimation of a.

3.4.5 Multiple Images

For this section we assume that several images have been collected over the same scene, and that no changes have occurred in r between images and since the generation of the prior. Our goal is to determine if multiple such images can be used to reduce the effects of noise.

In Section 3.4.2 we established that calibration noise does not affect $\hat{r}$ in the single image case, so it is not an issue in the multiple images case.

If we assume that the same prior is used when processing all of the multiple images, and

we assume perfect registration, then each $\hat{r}$ image pixel will have the same prior noise induced error, and therefore can not be reduced by processing multiple images.

The only noise effect that could be reduced by having multiple images is the one due to sensor noise. In Figure 3.11 it is apparent that the reflectance error caused by a positive one-sigma noise sample is nearly the same in magnitude as the negative error caused by a negative one-sigma noise sample. Therefore, averaging the estimated reflectance images will eliminate most of the sensor noise induced reflectance errors. For SNR>150, the reflectance error is already less than 0.01, and averaging would reduce it by approximately a factor of 10.

3.5 Summary and Conclusions

In this chapter we derived Spatial SPIRE algorithms that estimate surface reflectance using a prior reflectance image and spatial filtering to remove the multiplicative and additive illumination noise effects. Unlike traditional lightness algorithms which can only compensate for multiplicative noise, we demonstrated that SPIRE algorithms compensate for both multiplicative and additive noise. We compared the performance of the Spatial SPIRE algorithms to ELM and ATREM on six HYDICE airborne sensing hyperspectral image cubes from the ARM Site data collect. Based on these experiments we can make the following conclusions:

- Performance of Spatial SPIRE was very similar ELM.
- Performance of Spatial SPIRE was consistently far better than ATREM, mostly due to clouds affecting ATREM's performance on three of the runs.
- Spatial SPIRE algorithms are insensitive to calibration noise and can therefore be applied to uncalibrated sensor data.

- Spatial SPIRE algorithms are insensitive to misregistration noise between the prior and current image.
- The effects of sensor and prior noise can be overcome with adequate SNR.
- Non-iterating, computationally efficient algorithms were developed for Cases 1, 2, and 5 and can be used if the spatial uniformity conditions on the noises are met.
- The Spatial SPIRE Case 4 algorithm performs poorly on low reflectivity materials under low SNR conditions.

In the next chapter, we develop Spectral SPIRE algorithms which use filtering techniques similar to Spatial SPIRE, but in the spectral dimension.

Chapter 4

Principal Components Analysis of Multiplicative and Additive Noise Ensembles

4.1 Introduction

In this chapter, we lay the groundwork for the Spectral SPIRE algorithms developed in Chapter 5 and the spectral processing used in the Combined SPIRE algorithms developed in Chapter 6. We begin by discussing Principal Components Analysis (PCA) and then apply it to ensembles of HYDICE noise and reflectance pixel spectral vectors to determine the number of degrees of freedom in r, m, and a in the HYDICE test data set presented in Chapter 3. We next develop two techniques that make use of PCA to "focus", or collect the majority of the multiplicative and additive noise into a small number of principal components (PCs) called Abutted Principal Components (APC) analysis and Zero-Padded Principal Components (ZPC) analysis. These focused PCA techniques are required for effective Spectral and Combined SPIRE algorithms. We end the chapter with a discussion of the HYDICE channels dropped due to negative radiance values, water vapor absorption, and spectral channel center wavelength drift at edges of water absorption bands, since problems in these bands have a detrimental effect on PCA and are generally not of use by applications using surface reflectance.

4.2 Principal Components Analysis

Principal Components Analysis (PCA) utilizes the Principal Components Transform (PCT) to remove spectral redundancy, or correlation between spectral channels (Schowendgert, 1997, Richards, 1993, Lillesand and Kiefer, 1994). The PCT is mathematically equivalent to the Karhunen-Loéve Transform (Van Trees, 1968, Papoulis, 1991) and the Hotelling Transform

(Richards, 1993) for discrete data. By viewing each pixel spectrum as a vector in an orthogonal N-dimensional space, where N is the number of spectral channels, then the PCT can be understood as a rotation into a new set of N-dimensional coordinate axes in which all of the data are uncorrelated. Since independent variables in the data will be uncorrelated, decorrelating the data with a PCT will tend to separate independent variables and collect them into different PC dimensions.

For discrete data, the PCT can be defined mathematically as follows. If the original ensemble of pixel vectors is denoted by $\{\mathbf{x}\}$, then we seek a rotation matrix $\mathbf{\Phi}$ such that the covariance matrix of the rotated ensemble $\{\mathbf{y}\}$ is diagonal, indicating that $\{\mathbf{y}\}$ are uncorrelated:

$$\begin{aligned} &\mathbf{y} = \mathbf{\Phi x} \\ &\mathbf{\Lambda}_y = \mathbf{\Phi \Lambda}_x \mathbf{\Phi}^T \Rightarrow \mathbf{\Lambda}_y \textit{ is diagonal} \end{aligned} \tag{1.1}$$

In such a PCT, the rows of the matrix $\mathbf{\Phi}$ are the eigenvectors $\boldsymbol{\varphi}_i$ of the covariance matrix $\mathbf{\Lambda}_x$ and the diagonal elements of $\mathbf{\Lambda}_y$ are the corresponding eigenvalues λ_i. The eigenvectors and eigenvalues satisfy (Strang, 1998):

$$\mathbf{\Lambda}_x \boldsymbol{\varphi}_i = \lambda_i \boldsymbol{\varphi}_i \tag{1.2}$$

By convention, the new N-dimensional vector's dimensions, referred to as Principal Components (PCs), are ordered in descending eigenvalue magnitude. In this thesis we will use the convention that the first PC, corresponding to the largest eigenvalue, will be PC number 1. Thereby, the first, or lowest, principal component will have the largest variance, while the last, or highest PCs will typically contain only low variance random noise, especially if there are far fewer total degrees of freedom in the data than there are dimensions. This typically happens in

sensor datasets where spectral channels are correlated. In datasets containing linear combinations of independent variables (degrees of freedom), the PCT will tend to collect these degrees of freedom into separate PCs to the extent that they are uncorrelated. While this separation is seldom perfect, it can be a powerful tool in signal analysis. Finding the PCT for an ensemble is often referred to as Principal Components Analysis (PCA)

4.2.1 PCT Implementation

Since the PCT is defined using an ensemble's covariance matrix, the mean vector $\boldsymbol{\mu}_x$ can be subtracted from the ensemble before PCA without affecting the rotation. This is due to the fact that the covariance is defined as:

$$\mathrm{cov}(\mathbf{x},\mathbf{x}) = E[(\mathbf{x}-\boldsymbol{\mu}_x)(\mathbf{x}-\boldsymbol{\mu}_x)^T] \tag{1.3}$$

In many applications of PCA, the mean vector is not only subtracted before calculating the $\boldsymbol{\Phi}$ rotation matrix, but it is also done prior to rotating any vector into PC space. In this thesis, we will eventually be dealing with combined spatial and spectral filtering techniques that filter out low spatial frequency components from specific PCs, and then restore them from a prior. Therefore, we need to keep the mean (zero spatial frequency) information in the data, so our PCA will be done without subtracting the mean.

The estimation of the $\boldsymbol{\Phi}$ and $\boldsymbol{\Lambda}_y$ from a vector ensemble in this thesis was done by using the *envi_stats_doit()* ENVI routine (ENVI User's Guide, 1997) which calculates the covariance matrix $\boldsymbol{\Lambda}_x$, its eigenvectors $\boldsymbol{\varphi}_i$, and its eignevalues. Treating the eigenvectors as column vectors, they were then used to construct the PC rotation matrix $\boldsymbol{\Phi}$ as follows:

$$\Phi = \begin{bmatrix} \leftarrow \varphi_1^T \rightarrow \\ \leftarrow \varphi_2^T \rightarrow \\ \vdots \\ \leftarrow \varphi_N^T \rightarrow \end{bmatrix} \tag{1.4}$$

4.3 Degrees of Freedom of m, a, and r in HYDICE Data Set

In this thesis we are using data from the HYDICE airborne hyperspectral sensor which has 210 spectral channels. We can use PCA to analyze this dataset to determine the approximate number of degrees of freedom present in the multiplicative and additive noises, as well as in the reflectance signal of interest. This information will be useful when designing our Spectral and Combined SPIRE algorithms, which rely on PCTs that collect noise terms into a few PCs, effectively band limiting them to a few PCs so that "spectral" band pass filters can be applied in PC space.

4.3.1 Empirical Ensembles of m and a

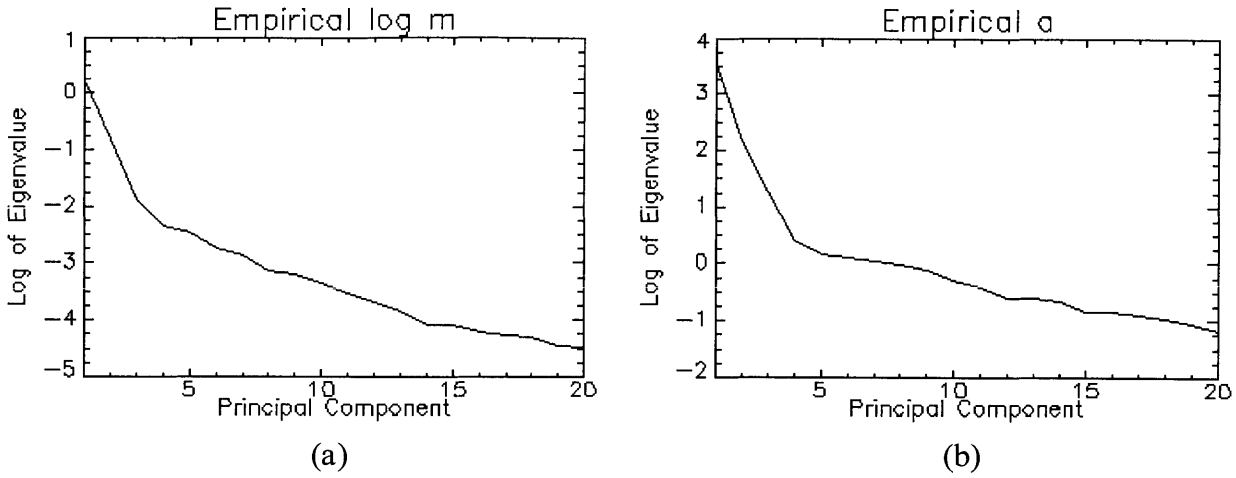
Ensembles of additive and multiplicative *m* and *a* vectors were required to define the PC rotations needed by the Spectral and Combined SPIRE algorithms to concentrate the *a* and log *m*. noise terms into a few low PCs for removal. We will show in Chapters 5 and 6 that Spectral and Combined SPIRE algorithms will be insensitive to calibration noise if ensembles derived from calibrated sensor data are used to define the PC rotations, since any calibration noise will also be collected into low PCs and removed along with the multiplicative and additive noises caused by illumination effects. Therefore, calibrated HYDICE data were used as the source of the ensembles of *m* and *a*.

4.3.1.1 Pre-computed ELM Gain and Offset Files

Several HYDICE data sets were available in addition to the 1997 ARM Site data set discussed in Chapter 3. For each of these data sets, ELM and ground truth measurement of calibration panels had been used to generate gain (multiplicative noise **m**) and offset (additive noise **a**) vectors for each image. Twenty five pairs of such **m** and **a** vectors were obtained from the HYDICE Forest Radiance I, Urban Radiance I, and Desert Radiance II campaigns. Such pre-computed vectors were not available for the 1997 ARM Site campaign.

4.3.1.2 ELM m and a Vectors from ARM Site Images

Since we are now dealing with pixel vectors, we will use bold variable names to denote vector quantities. To obtain **m** and **a** vectors from the ARM Site dataset, 8 of the 14 image cubes in the ARM Site data collect were processed using ELM, excluding the 6 test cubes used in Chapter 3 for testing Spatial SPIRE, so that the PCT would not be defined by the blind data it would be tested on. For each image cube, ground truth spectra for the 64% and 2% panels were re-sampled into the 210 HYDICE channels defined by the channel center wavelength file for that cube. Radiance values from all non-edge pixels on the 64% and 2% panels were input to the ELM algorithm, along with the ground truth spectra, generating an **m** and an **a** vector for each ARM Site image cube. These were combined with the gain and offset vectors from the other HYDICE campaigns, to create $\{\mathbf{m}\}$ and $\{\mathbf{a}\}$ ensembles of 33 vectors each. The log was then taken of the $\{\mathbf{m}\}$ ensemble to generate a $\{\log \mathbf{m}\}$ ensemble, which will be used for PC based filtering in log space. Spectral channels were dropped as described later in this chapter to remove channels affected by water vapor absorption and negative radiance values, and then PCA was performed on the $\{\log \mathbf{m}\}$ and $\{\mathbf{a}\}$ ensembles.



(a) (b)

Figure 4.1. *Principal component eigenvalue plots for empirical* log **m** *(a) and* **a** *(b) ensembles. Each ensemble contains 33 vectors obtained from ELM processing of calibrated HYDICE data. Using the Lee and Staelin scree-plot technique, both* {log **m**} *and {***a***} empirical ensembles are estimated to contain 3 degrees of freedom each.*

Figure 4.1 shows PC eigenvalue plots for $\{\log \mathbf{m}\}$ and $\{\mathbf{a}\}$. Using these data we can estimate the number of degrees of freedom (DOF) present in the data. The "scree-plot" technique (Lee and Staelin, 2001, Lee, 2000) was used to estimate the number of degrees of freedom. In geophysical terms, a scree is loose rock that slopes up to the base of a mountain and is the junction between a mountain and a plain. PCs containing mostly noise will tend to increase gradually in eigenvalue (a scree) until the PCs containing degrees of freedom are reached, at which point the eigenvalues increase faster (a mountain). The scree-plot technique developed by Lee and Staelin to estimate the number of DOF uses a least squares linear regression to estimate the slope and intercept of a line that runs through the scree of noise dominated PCs when the eigenvalues are plotted on a log axis. Where the scree-plot deviates from this line is the start of the "mountain" and corresponds to the PC of the last DOF. Using this technique, the empirical $\{\log \mathbf{m}\}$ and $\{\mathbf{a}\}$ ensembles were each estimated to contain 3 DOF.

4.3.2 MODTRAN Simulated Ensembles

To verify that the empirically derived ensembles were representative and contained reasonable degrees of freedom, simulated ensembles were generated using the MODTRAN 4.0 atmospheric radiation transfer and simulation code (Adler-Goldem, et al., 1998).

First, 5 sets of 8 input runs were done to generate **m** and **a** vectors similar to those encountered in the ARM Site data set. All were run with the Mid-Latitude Summer atmospheric model, a RURAL aerosol extinction model with 23 km visibility, multiple scattering, and surface spectral reflectance set to one for all wavelengths. The five sets each stepped through eight runs of:

- Sensor altitudes ranging from 0.5 to 4.0 km
- Solar zenith angles ranging from 10 to 80 degrees
- Column water vapor scaling factors of 0.15 to 1.90 times the default value in the model
- Day of year during the months appropriate for the model ranging from day 151 to 242
- Random combinations of the above four variables

In addition, three additional sets of eight runs were done of the random combinations but with:

- Mid-Latitude Summer atmospheric model and URBAN aerosol model with 5 km visibility
- Mid-Latitude Winter atmospheric model and RURAL aerosol model with 23 km visibility (with appropriate winter days)
- Mid-Latitude Winter atmospheric model and URBAN aerosol model with 5 km visibility (with appropriate winter days)

The above simulation runs yielded a total of 64 **m** and 64 **a** vectors. Each **a** vector was calculated as the sum of the PTH THERM (path thermal) and SING SCAT (single scattering) outputs. Each **m** vector was calculated as the TOTAL RAD (total radiation) output minus the calculated **a** vector. PCA analysis was performed on the simulated ensembles of $\{\log \mathbf{m}\}$ and $\{\mathbf{a}\}$. Spectral channels were dropped as described later in this chapter to remove channels affected by water vapor absorption and negative radiance values, and PCA was performed on the resulting reduced channel $\{\log \mathbf{m}\}$ and $\{\mathbf{a}\}$ ensembles.

Figure 4.2 shows PC eigenvalue plots for the simulated $\{\log \mathbf{m}\}$ and $\{\mathbf{a}\}$. Using the scree-plot technique to estimate the number of DOF, the $\{\log \mathbf{m}\}$ ensemble was estimated to contain 4 DOF, and $\{\mathbf{a}\}$ was estimated to contain 2. This is consistent with the fact that four variables were varied, and that the day of year and column water vapor had very little effect on the **a** vectors.

Figure 4.3 plots the PC eigenvalues for both the empirical and simulated ensembles, simultaneously plotting the data from Figures 4.1 and 4.2. The disparity of the empirical and MODTRAN plots at higher PCs is due to sensor noise in the empirical data not present in the MODTRAN simulation. While the $\log \mathbf{m}$ plots match well at lower PCs, the difference between the two **a** plots at lower PCs is most likely due to limitations of the MODTRAN aerosol models.

Based on these plots and the scree-plot analysis, we conclude that the $\{\log \mathbf{m}\}$ ensemble contains 3-4 DOF and the $\{\mathbf{a}\}$ contains 2-3. This is plausible given the physics involved. Absorption related multiplicative effects will vary with altitude and solar zenith angle as the amount of atmosphere to be traversed changes. Varying water vapor absorption will also vary the multiplicative noise. Also, varying the day of year varies the amount of top-of-the-

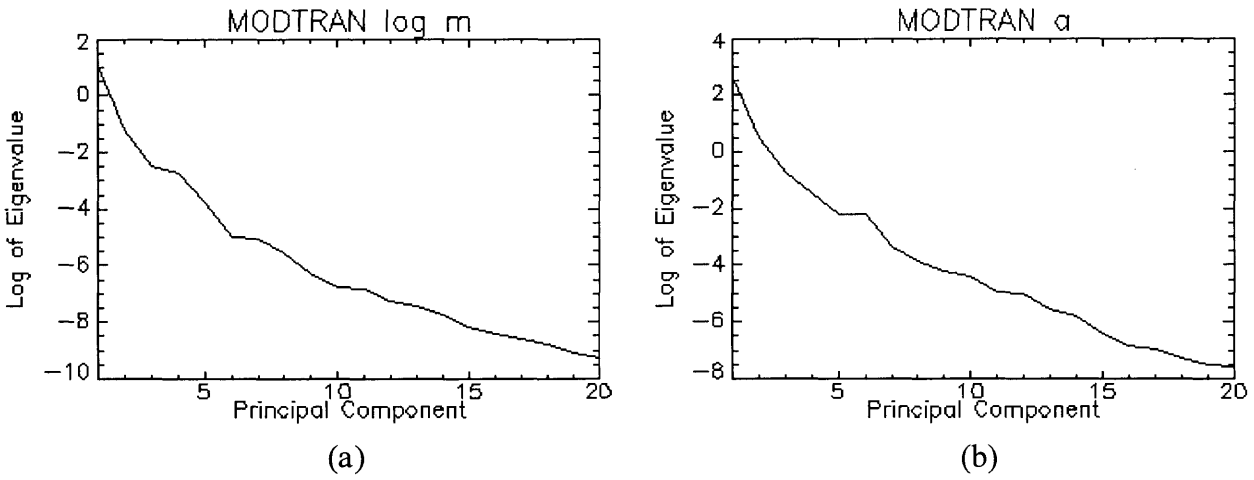


Figure 4.2. *Principal component eigenvalue plots for MODTRAN simulated* log **m** *(a) and* **a** *(b) ensembles. Each ensemble contains 33 vectors obtained from ELM processing of calibrated HYDICE data. Using the Lee and Staelin scree-plot technique, the* {log **m**} *MODTRAN ensemble is estimated to contain 4 degrees of freedom while the* {**a**} modtran *ensemble is estimated to contain 2 degrees of freedom.*

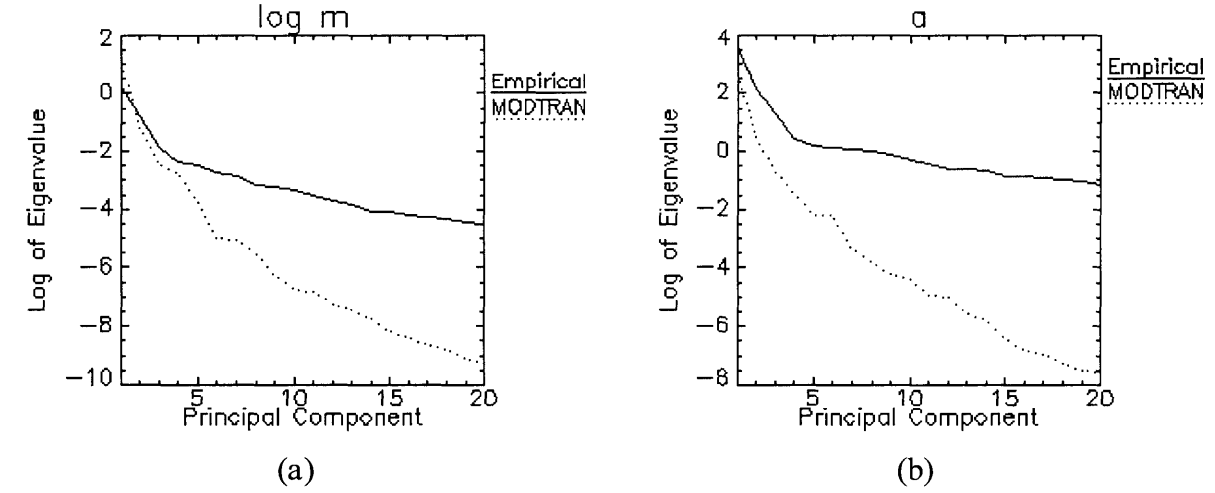


Figure 4.3. *Principal component eigenvalue plots for both empirical and MODTRAN simulated* {log **m**} *(a) and* {**a**} *(b) ensembles. These plots combine the plots from Figures 4.1 and 4.2. The disparity of the empirical and MODTRAN plots at higher PCs is due to sensor noise not present in the MODTRAN simulation. While the* log **m** *plots match well at lower PCs, the difference between the two* **a** *plots at lower PCs is most likely due to limitations of the MODTRAN aerosol models.*

atmosphere solar radiation due the non-circular orbit of the earth about the sun, which will have a multiplicative effect. Additive noise is affected mostly by variations in aerosol scattering, which are most affected by altitude and solar zenith angle which vary the length of the radiative path through the atmosphere and thereby the amount of scattering constituents encountered. The additive noise is also affected a lesser extent by the type of aerosols present over different land types.

4.3.3 Empirical Ensemble of Reflectance

To get a sense of the number of degrees of freedom in the surface reflectance of the ARM Site data, the ELM reflectance estimate for Run 07 was analyzed using PCA (after removing water vapor and other bad channels) by treating the pixel vectors as an ensemble of reflectance spectra $\{\mathbf{r}\}$. Figure 4.4 shows the PC eigenvalue plot this $\{\mathbf{r}\}$. Using the scree-plot technique, the $\{\mathbf{r}\}$ ensemble is estimated to have 6 degrees of freedom.

4.4 Abutted Principal Components Analysis

In a normal PCA, the mean of the ensemble of vectors being analyzed does not affect the PC rotation since the mean does not affect the covariance. Therefore, normal PCA will not collect all of the mean or DC component into the low PCs (though the lower PCs with high variances often tend to have higher means than other PCs). If all of a quantity such as $\log \mathbf{m}$ needs to be removed, including its mean, then normal PCA will not achieve the desired goal of collecting all of $\log \mathbf{m}$ into a few low PCs to band limit $\log \mathbf{m}$ in PC space, so that they can be spectrally filtered out.

Two techniques were used in this thesis to "focus", or collect the mean of a noise ensemble into low PCs along with the rest of the ensemble. The first is called Abutted Principal

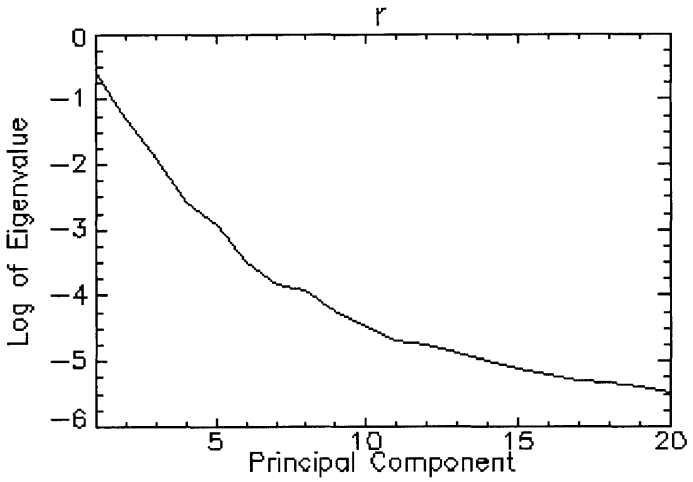


Figure 4.4. *Principal component eigenvalue plot for an ensemble of ELM estimates of r vectors in the Run 07 HYDICE image. Using the Lee and Staelin scree-plot technique, this* $\{\mathbf{r}\}$ *ensemble is estimated to contain 6 degrees of freedom.*

Components (APC) analysis and will be developed in this section.

We begin with our vector notation, multiplicative-noise-only image formation model:

$$\mathbf{i} = \mathbf{r} \odot \mathbf{m} \tag{1.5}$$

We then move into log space:

$$\log \mathbf{i} = \log \mathbf{r} + \log \mathbf{m} \tag{1.6}$$

Next we spatially abut the log image cube and the log prior reflectance cube, which is simply a union of the two ensembles, and is shown in Figure 4.5:

$$\{\mathbf{j}\} = abut(\{\log \mathbf{i}\}, \{\log \mathbf{r}_p\}) = \{\log \mathbf{i}\} \cup \{\log \mathbf{r}_p\} \tag{1.7}$$

Note that the spatial locations of these pixels do not matter for the analysis. In other words, we could now spatially scramble all of the pixel vectors and the principal components

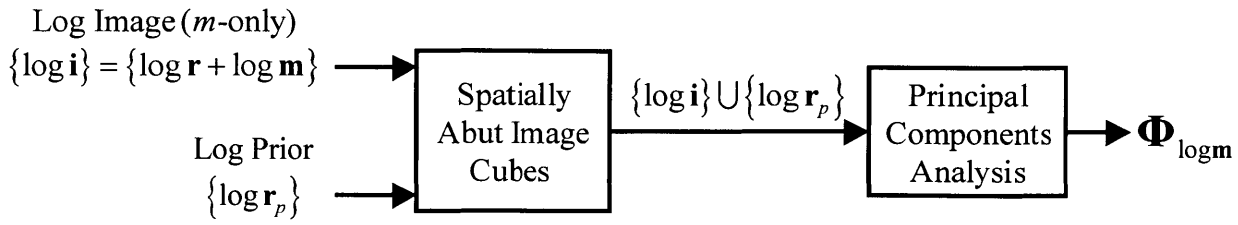


Figure 4.5. *Abutted Principal Components (APC) flow diagram. The log radiance image is spatially abutted with the log prior reflectance. This union of the two pixel ensembles is then subjected to Principal Components analysis (PCA). Note that spatial abutting is being used here to make the results of the PCA clearer in Figure 4.6. Any union of the ensembles will result in the same PC rotation from the PCA. In other words, the spatial positions of the pixels relative to one another do not affect the PCA and resulting* $\mathbf{\Phi}_{\log \mathbf{m}}$.

analysis and resulting rotation matrix would be the same. We use the spatial abutting approach because it makes it easier to understand the APC technique as depicted in Figure 4.6, and because it was easy to implement algorithmically. The final step is to calculate the rotation matrix $\mathbf{\Phi}_j$ that decorrelates the abutted ensemble **{j}** per (1.1)-(1.4).

In Figure 4.6(a) the log prior reflectance log $\mathbf{r}_p$ is spatially abutted with the log image log **i**. Each single channel abutted image has only log $\mathbf{r}_p$ on the left but both log **m** and log **r** on the right half. In Figure 4.6(b) we have performed the PC rotation, collecting the $\log \mathbf{m}$ into the low PCs, along with a small amount of log $\mathbf{r}_p$ and log **r**, leaving the rest in the higher PCs. In effect, the presence or absence of $\log \mathbf{m}$ has been turned into a degree of freedom in the data. Reflectance values vary between 0 and 1. By selecting a larger numerical value for the scale of the illumination, the variance of $\log \mathbf{m}$ across the abutted image will be much higher than the variance of reflectance within the ensemble. This will force the degrees of freedom in the data associated with $\log \mathbf{m}$ to be collected in the lowest PCs, along with whatever portion of $\log \mathbf{r}$ that happens to be randomly correlated with them.

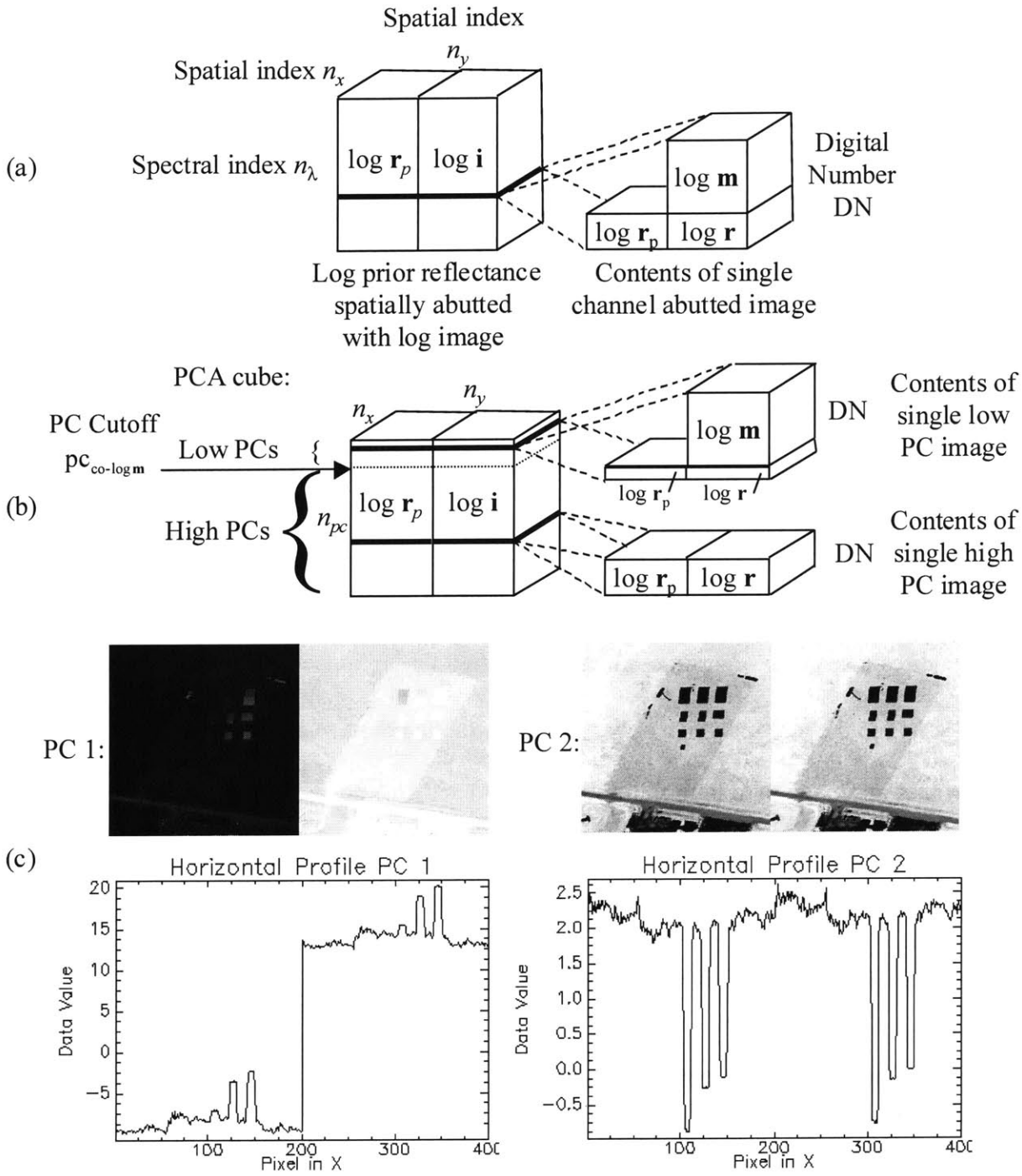


Figure 4.6. *Abutted Principal Components (APC) details. In (a), the log prior reflectance* log $\mathbf{r}_p$ *is spatially abutted with the log image* log $\mathbf{i}$. *Each single channel abutted image has only* log $\mathbf{r}_p$ *on the left but both* log $\mathbf{m}$ *and* log $\mathbf{r}$ *on the right half. In (b) we have performed the PC analysis, collecting the* log $\mathbf{m}$ *into the low PCs, along with a small amount of* log $\mathbf{r}_p$ *and* log $\mathbf{r}$, *leaving the rest in the higher PCs. In effect, the presence or absence of* log $\mathbf{m}$ *has been turned into a degree of freedom in the data. Since* log $\mathbf{m}$*'s magnitude is much larger than the variance in* log $\mathbf{r}$, *it ends up in the lowest PCs. In (c), an example cube with uniform* log $\mathbf{m}$ *has had nearly all* log $\mathbf{m}$ *collected into PC 1, with almost none remaining in PC 2 and higher.*

In Figure 4.6(c), an example cube with uniform $\log \mathbf{m}$ has been analyzed using APC. Since $\log \mathbf{m}$ is uniform, it represents only a single degree of freedom, and is essentially all collected into PC 1, with none remaining in PC 2 and higher. We see in the PC images and horizontal profiles that PC 1 has a large horizontal step corresponding to the absence and presence of log m in the original abutted cube of Figure 4.6(a), while this step is essentially not present in PC 2. With normal PCA, a uniform $\log \mathbf{m}$ would be completely missed since it is all in the mean.

One major strength of APC is that it does not require any knowledge about $\log \mathbf{m}$. No prior ensembles are needed of $\log \mathbf{m}$ for this to work. Also, the spatial variability of $\log \mathbf{m}$ is not an issue, it will all be collected into low PCs. Deciding where the PC cutoff is for significant $\log \mathbf{m}$ requires a separate algorithm and this is addressed in Appendix D. For our application, APC is limited to the multiplicative only case, since we do not have a prior ensemble of $\mathbf{rm}$ with which to abut an image with both multiplicative and additive noise. Therefore, APC is of little use in remote sensing applications where both $\mathbf{m}$ and $\mathbf{a}$ are present. In the next section, we develop a related technique called Zero-padded Principal Components, which is applicable to removing both $\mathbf{a}$ and $\log \mathbf{m}$, but requires prior ensembles of them..

4.5 Zero-Padded Principal Components Analysis

Figure 4.7 shows the flow diagram for performing Zero-padded Principal Components (ZPC). We assume that we have ensembles of noise vectors such as $\{\log \mathbf{m}\}$. If we were to perform PCA on them, then the mean would not be collected into the low PCs. To make the mean a degree of freedom, we pad the noise ensemble of $\log \mathbf{m}$ with an image of zero vectors by spatially abutting them:

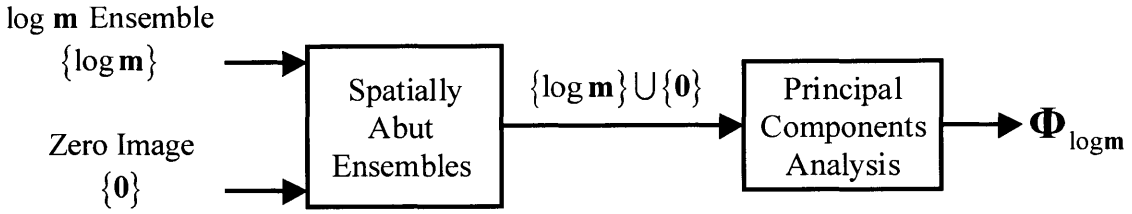


Figure 4.7. *Zero-padded Principal Components (ZPC) flow diagram. The* log **m** *ensemble is spatially abutted with an image cube of zero vectors. This union of the two pixel ensembles is then subjected to Principal Components analysis (PCA). This is similar to Abutted Principal Components of Figures 4.5 and 4.6, but uses a zero vector image cube abutted to the* log **m** *ensemble.*

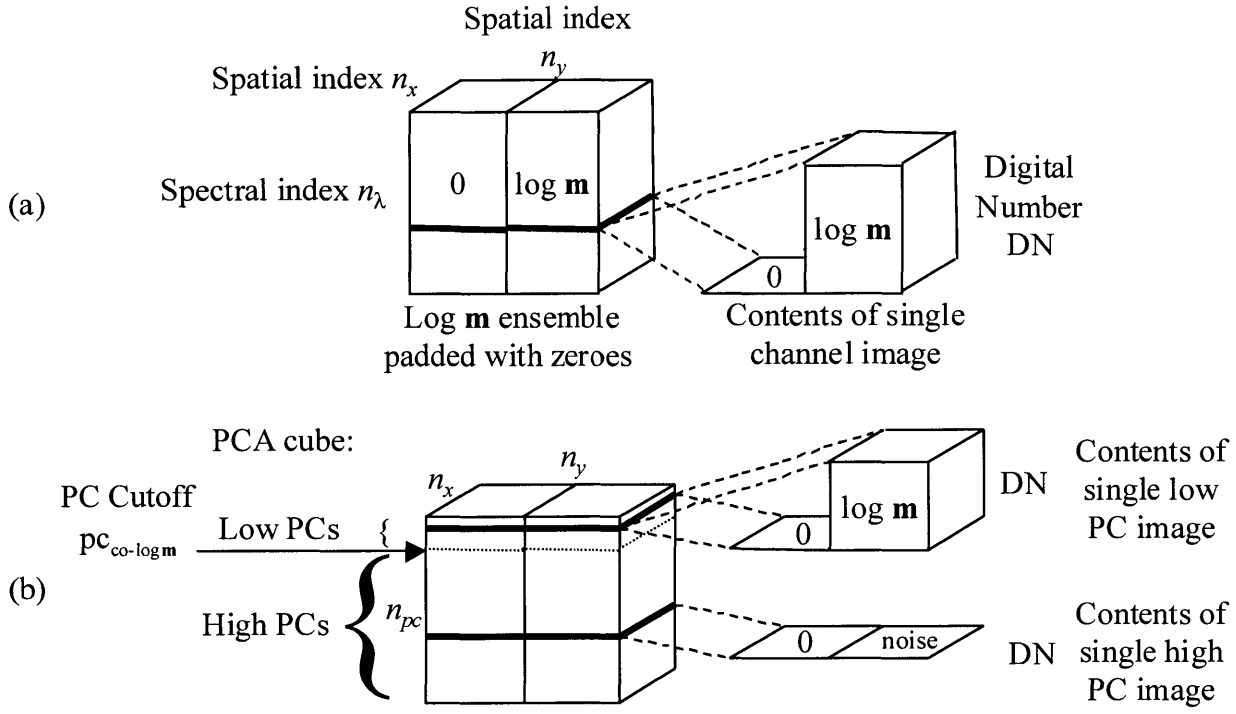


Figure 4.8. *Zero-padded Principal Components (ZPC) details. In (a), the log multiplicative noise ensemble of* log **m** *is padded with zero vectors. Each single channel image has zero values on the left and* log **m** *on the right half. In (b) we have performed the PC analysis, collecting the log* ***m*** *into the low PCs, leaving noise in the high PCs. In effect, the presence or absence of* log **m** *has been turned into a degree of freedom in the data, which turns the zero spatial frequency (DC) of* log **m** *into a DOF, which it is not in normal PCA.*

$$\{z\} = about(\{\log \mathbf{m}\}, \{\mathbf{0}\}) = \{\log \mathbf{m}\} \cup \{\mathbf{0}\} \tag{1.8}$$

We then perform PCA on this zero padded image. In Figure 4.8(a) we see this graphically depicted. Each single channel image now has zero values on the left and $\log \mathbf{m}$ on the right half. In Figure 4.8(b) we have performed a normal PC analysis, collecting the $\log \mathbf{m}$ into the low PCs, leaving noise in the high PCs. This technique is applicable to both the $\{\log \mathbf{m}\}$ and $\{\mathbf{a}\}$ noise ensembles, and can be pre-computed from prior noise ensembles.

4.6 Comparison of PCA, APC, and ZPC

Figure 4.9 plots the mean of the $\log \mathbf{m}$ ensemble in PC space after rotating with normal Principal Components analysis (PCA) (a), Abutted Principal Components (APC) analysis (b), and Zero-padded Principal Components (ZPC) analysis (c). The APC rotation was defined using the **m**-only image for Run 07 after subtracting the ELM derived **a** vector for the image. ZPC is clearly better than PCA for concentrating mean $\log \mathbf{m}$ noise into the lowest PCs. The APC plot is of the mean of the whole ensemble, which is not a fair comparison since APC is focused only on the Run 07 image for which it was defined, and not the entire ensemble. All three means are plotted together in (d)-(f), with the first 20 PCs plotted in (f) with full value scale, and all the PCs plotted in (e) with the value scale zoomed in near the origin.

In Figure 4.10 is plotted the ELM calculated log **m** spectrum in PC space for the Run 07 image after rotating with normal Principal Components analysis (PCA) (a), Abutted Principal Components (APC) analysis (b), and Zero-padded Principal Components (ZPC) analysis (c). ZPC is again clearly better than PCA at focusing the mean $\log \mathbf{m}$ signal into the low PCs. APC is the best, since it was defined on the Run 07 image. However, the PC rotation and PC cutoff

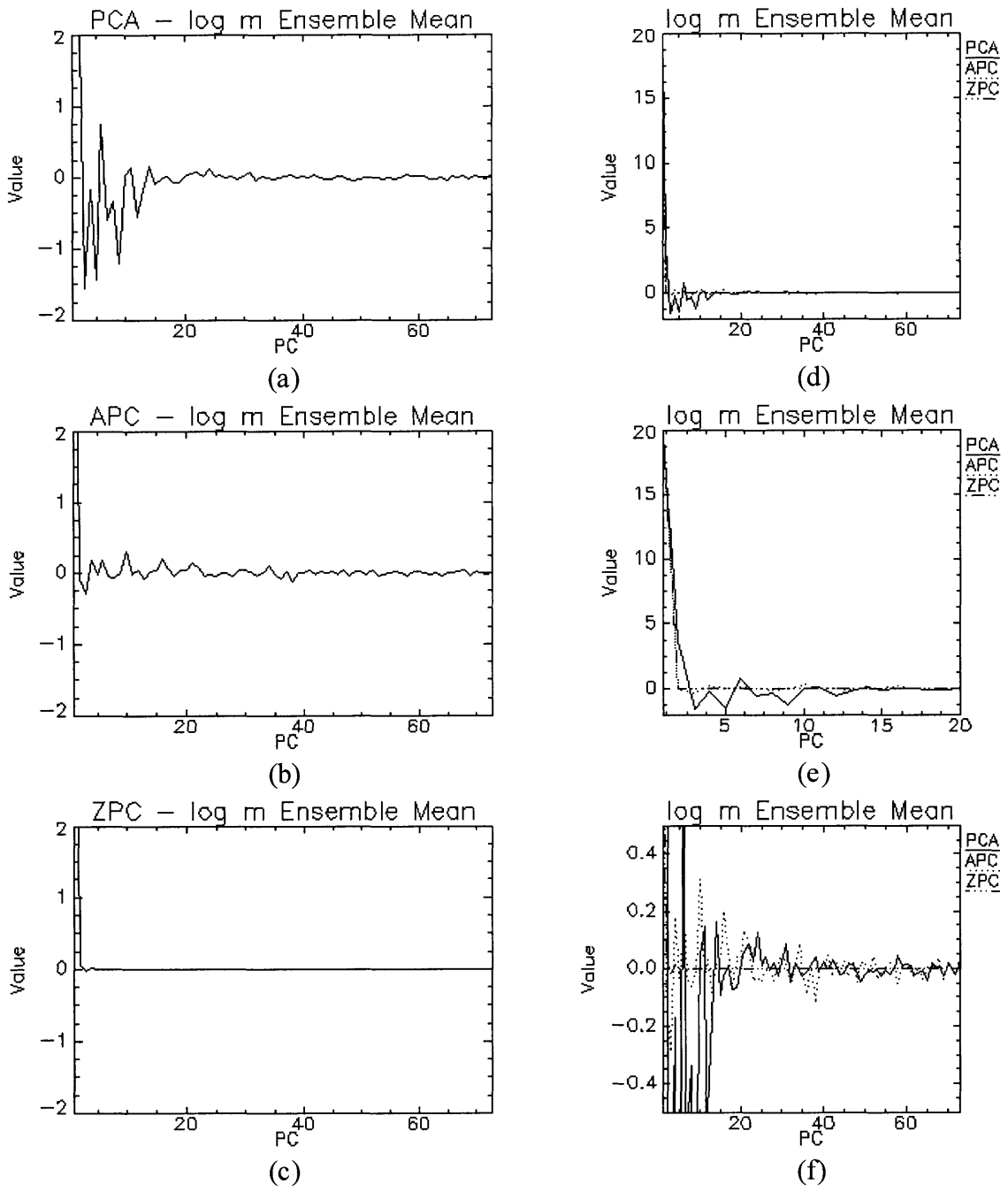


Figure 4.9. *Mean of* log **m** *ensemble in PC space after rotating with normal Principal Components analysis (PCA) (a), Abutted Principal Components (APC) analysis (b), and Zero-padded Principal Components (ZPC) analysis (c). ZPC is clearly better than PCA for concentrating mean* log **m** *noise into the lowest PCs. The APC plot is of the mean of the whole* log **m** *ensemble, which is not a fair comparison since the APC rotation matrix is intended to be used only on the pixels of the image for which it was defined, which is Run 07 in this case. All three means are plotted together in (d)-(f), with the first 20 PCs plotted in (e) with full value scale, and all the PCs plotted in (f) with the value scale zoomed in near the origin. See Figure 4.10 for plots of performance on just Run 07.*

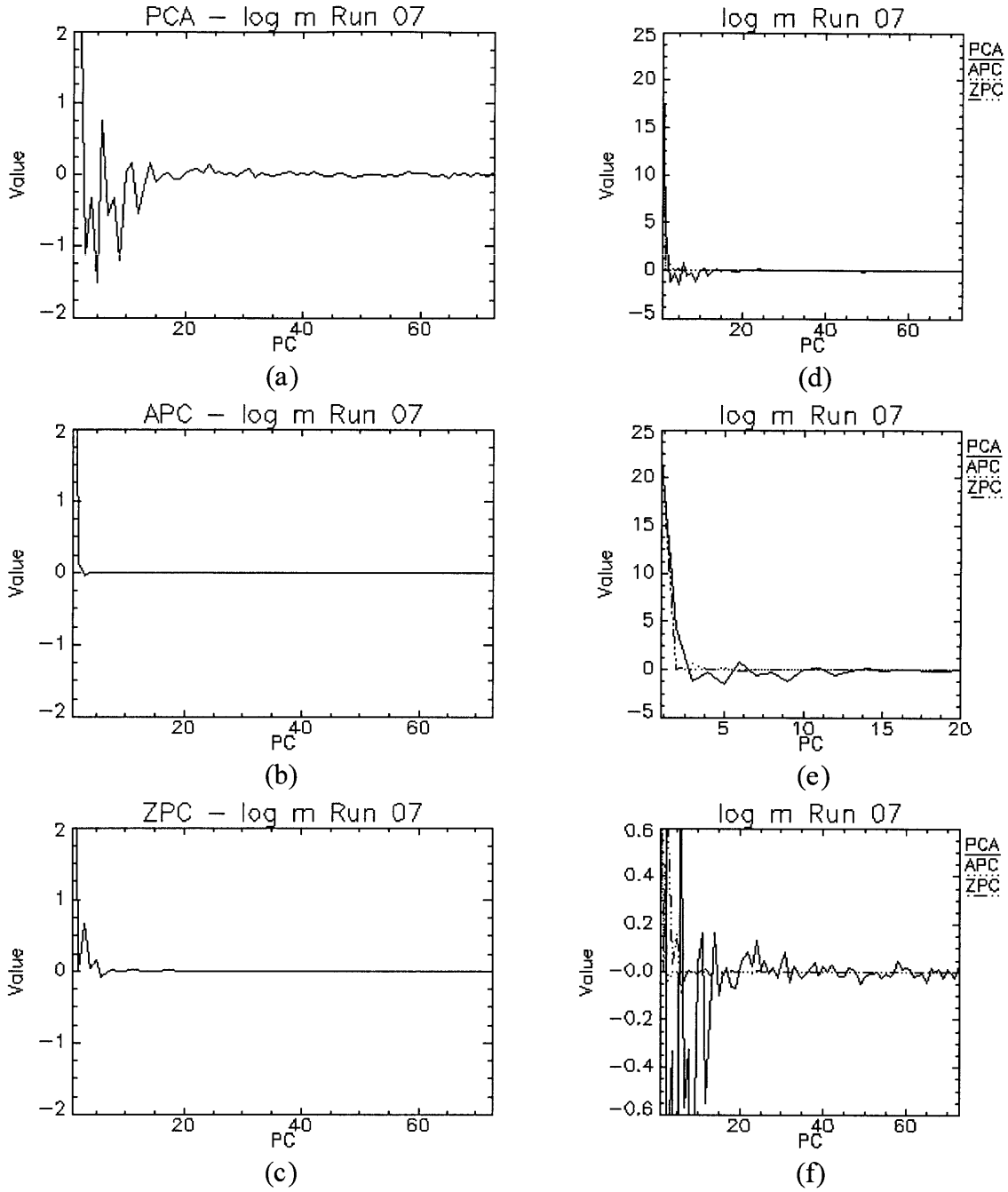


Figure 4.10. *ELM calculated* log **m** *spectrum for Run 07 in PC space after rotating with normal Principal Components analysis (PCA) (a), Abutted Principal Components (APC) analysis (b), and Zero-padded Principal Components (ZPC) analysis (c). ZPC is clearly better than PCA at focusing the log* ***m*** *signal into the low PCs. APC is the best, since it was defined on the Run 07 image. However, the ZPC rotation and PC cutoff can be pre-computed from a prior ensemble of log* ***m****, while both must be computed for each image using APC. We see that 5-6 PCs in ZPC contain significant mean signal, vs. less than 5 in APC. All three spectra are plotted together in (d)-(f), with the first 20 PCs plotted in (e) with full value scale, and all the PCs plotted in (f) with the value scale zoomed in near the origin.*

can be pre-computed for ZPC from a prior ensemble of $\log \mathbf{m}$, while both must be computed for each current image using APC. We see that approximately 5 PCs in ZPC contain significant mean signal, vs. less than 5 in APC. All three spectra are plotted together in (d)-(f), with the first 20 PCs plotted in (e) with full value scale, and all the PCs plotted in (f) with the value scale zoomed in near the origin.

Figures 4.9 and 4.10 tell us that APC will do the best at focusing the $\log \mathbf{m}$ noise into a few PCs for the image from which it was defined, does not require an ensemble of $\log \mathbf{m}$ vectors, but will not work well on other images or for cases with additive noise. ZPC will not work quite as well as APC for a particular image, but ZPC will work well enough on all images, can be precomputed, and works for cases with additive noise, though it requires ensembles of noise vectors. Since we do have cases with additive noise and we have ensembles of noise vectors, we will use ZPC in the rest of this thesis. Also, the 5 PCs containing significant signal in Figure 4.10(c) is an independent confirmation of the scree-plot analysis result of 4 DOFs in the $\mathbf{m}$ ensemble, since ZPC theory that predicts that we will add a DOF by zero-padding and need to process 4+1=5 DOF to remove the multiplicative noise.

Figure 4.11 shows the effect on the eigenvalue plot for the $\log \mathbf{m}$ and $\mathbf{a}$ ensembles for going from normal PCA to ZPC. We see that in the low PCs there is very little difference. The noise floor of the higher PCs drops under ZPC, indicating that more significant DOF signal has been transferred to the lower PCs as well. Using the scree-plot technique, the zero-padded ensembles were found to contain one more DOF than the original empirical ensembles, serving as an additional confirmation that we correctly understand the number of DOF in our noise ensembles.

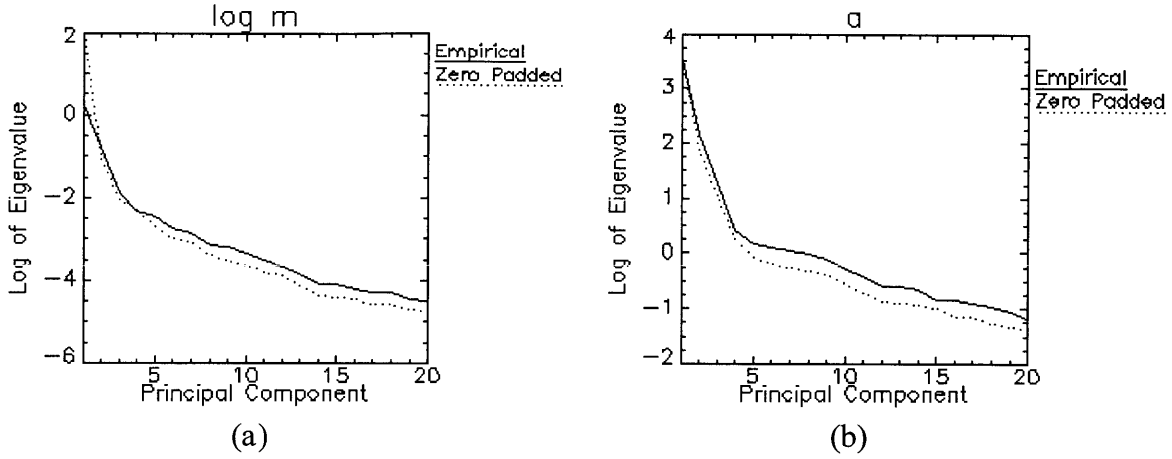


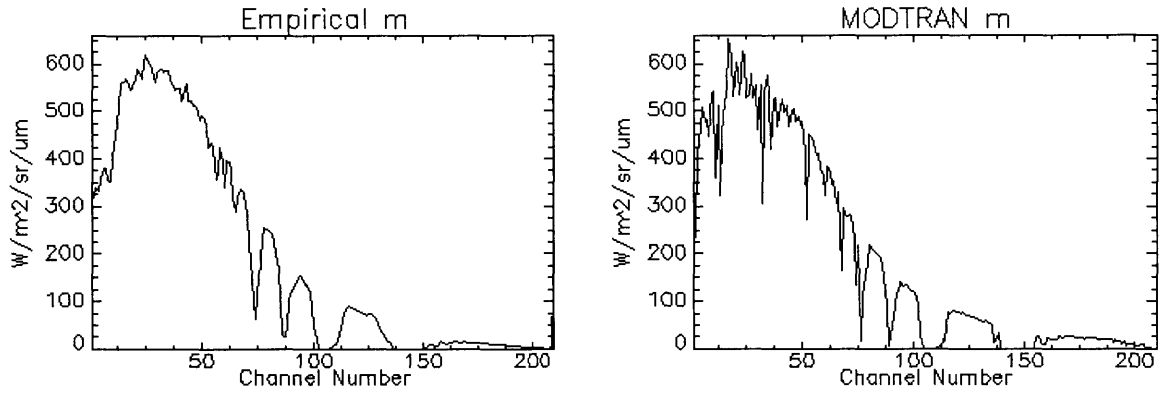
Figure 4.11. *Principal component eigenvalue plots for empirical and zero padded empirical* log **m** *(a) and* **a** *(b) ensembles. Using the Lee and Staelin scree-plot technique, zero padding was found to add one degree of freedom to both ensembles, as would be expected.*

4.7 HYDICE Spectral Channels Used

Not all of the HYDICE sensors 210 channels provide useful data for estimating spectral reflectance of the ground using SPIRE. Many channels are in water vapor absorption bands and do not receive any reflected radiation from the ground. Other channels suffer for sensor non-linearities at low signals which can generate physically impossible negative calibrated radiance values. These channels had to be removed to ensure that there was reflectance information in each channel used, and that no negative numbers would affect the results.

Figure 4.12 shows typical multiplicative noise **m** vectors from both empirical ELM results (a) and MODTRAN simulation (b). The water vapor absorption lines are apparent near channels 70, 90, 100-110, and 140-150. Analysis of the ELM gain and offset vector available from non-ARM Site HYDICE campaigns showed additional bands which consistently did not have valid ELM solutions due to these effects. These channels had to be removed since the lack of numbers in these channels would affect Principal Components analyses.

The HYDICE sensor suffers from non-linearities at low SNR for some of its channels,



(a) (b)

Figure 4.12. *Typical* **m** *vectors from empirical ELM results (a) and from MODTRAN simulation results (b). Water vapor absorption bands are apparent near channels 70, 90, 100-110, and 140-150. Such channels were removed from the data before testing SPIRE algorithms. This was required for techniques that make use of spectral processing, since empirical vectors of* **m** *and* **a** *had undefined values in these channels.*

especially the longer wavelength ones. This can result in negative radiance values in these channels. In the six ARM test image cubes used for SPIRE algorithm testing, Runs 13, 26, and 31 had very few negative numbers in any channels. In Figure 4.13(a) are plotted the percent of pixels with negative radiance in each channel for Runs 06, 07, and 22. In Figure 4.13(b), we have selected on Run 22 since it has the most negative values. The two central peaks near channels 110 and 150 correspond to water vapor absorption bands. The peaks on the left and right extremes are due to sensor non-linearities at low signal-to-noise ratios (SNR). All of the channels with over 2% of pixels less than zero in Run 22 were dropped, leaving two channels near 170. Any negative pixels remaining in the data were set to the smallest positive pixel value in its spectral channel.

The HYDICE sensor also suffers from a slow drift in the center wavelength of all of its spectral channels, which causes each channel's center wavelengths to vary from image to image. (A faster spectral jitter of the center wavelengths also occurs during a single image collection

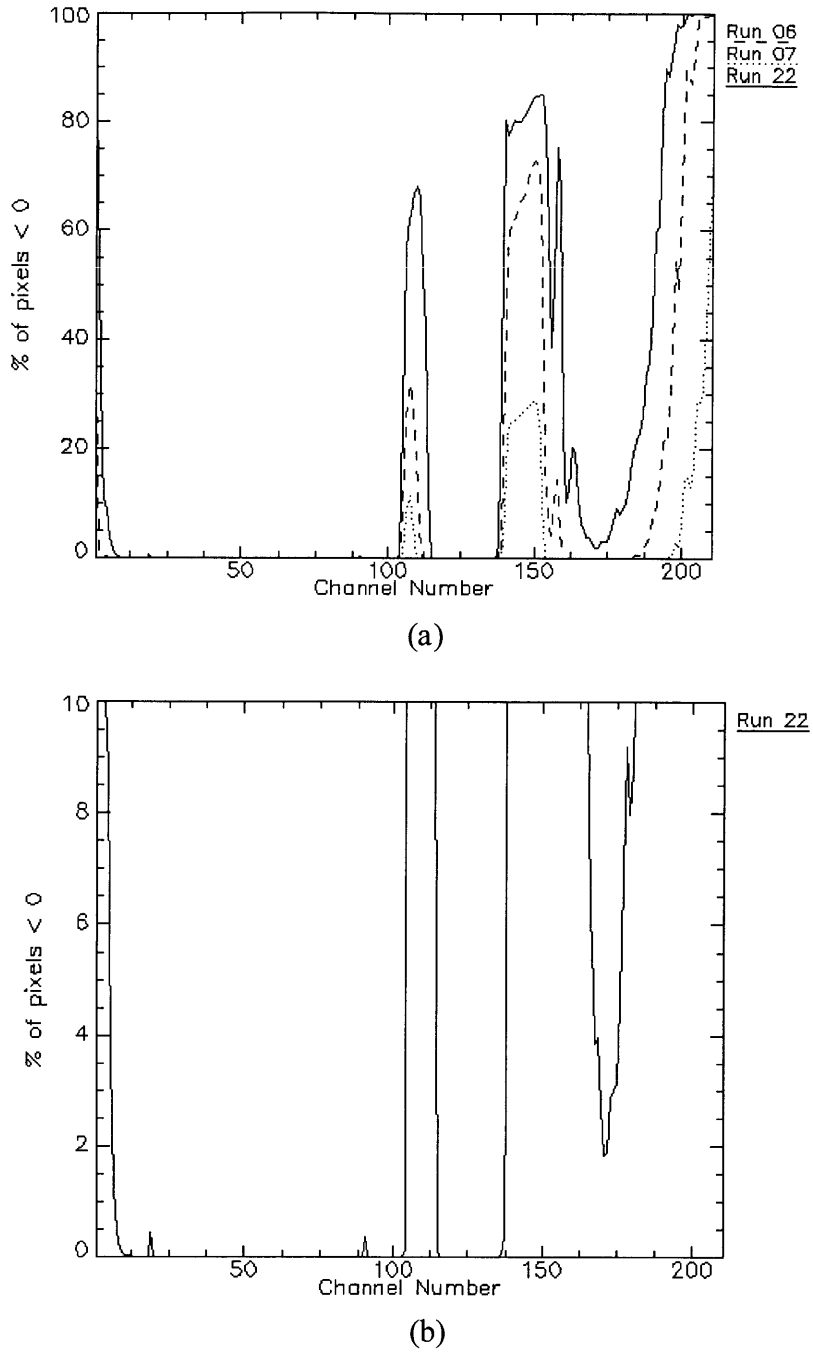
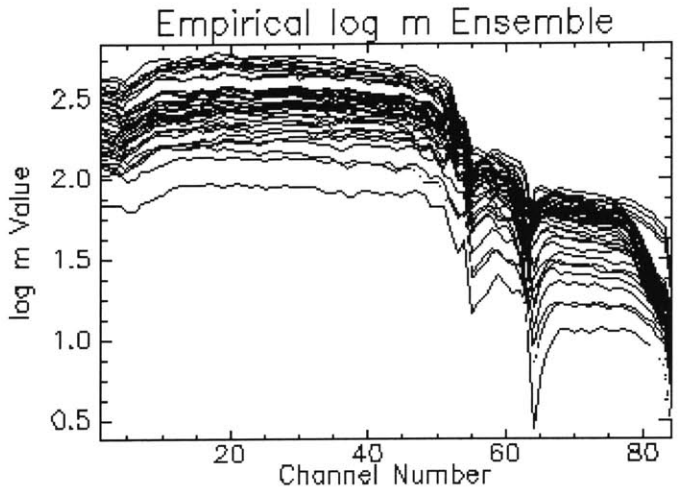
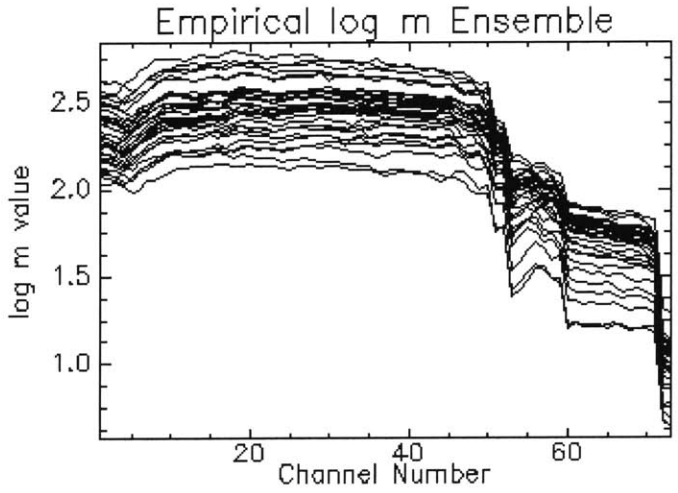


Figure 4.13. *Percent of pixels less than zero in the radiance images from Runs 06, 07, and 22 (a). In (b), we have selected Run 22 since it has the most negative values. The two central peaks near channels 110 and 150 correspond to water vapor absorption bands. The peaks on the left and right extremes are due to sensor non-linearities at low signal-to-noise ratios (SNR). All of the channels with over 2% of pixels less than zero in Run 22 were dropped, leaving two channels near 170.*



(a)



(b)

Figure 4.14. *Empirical log* **m** *ensembles before (a) and after (b) removing channels at the edges of water vapor absorption bands. In (a), center channel wavelength drift was causing spiking near channels 53 and 65 where drift in the center wavelength caused channel radiance values to rise or fall as the channels moved in and out of the water vapor absorption bands. In (b), these channels have been removed. After removal there are still discontinuous transitions where channels have been removed, but the transitions do not vary in size with center wavelength drift as they did in (a).*

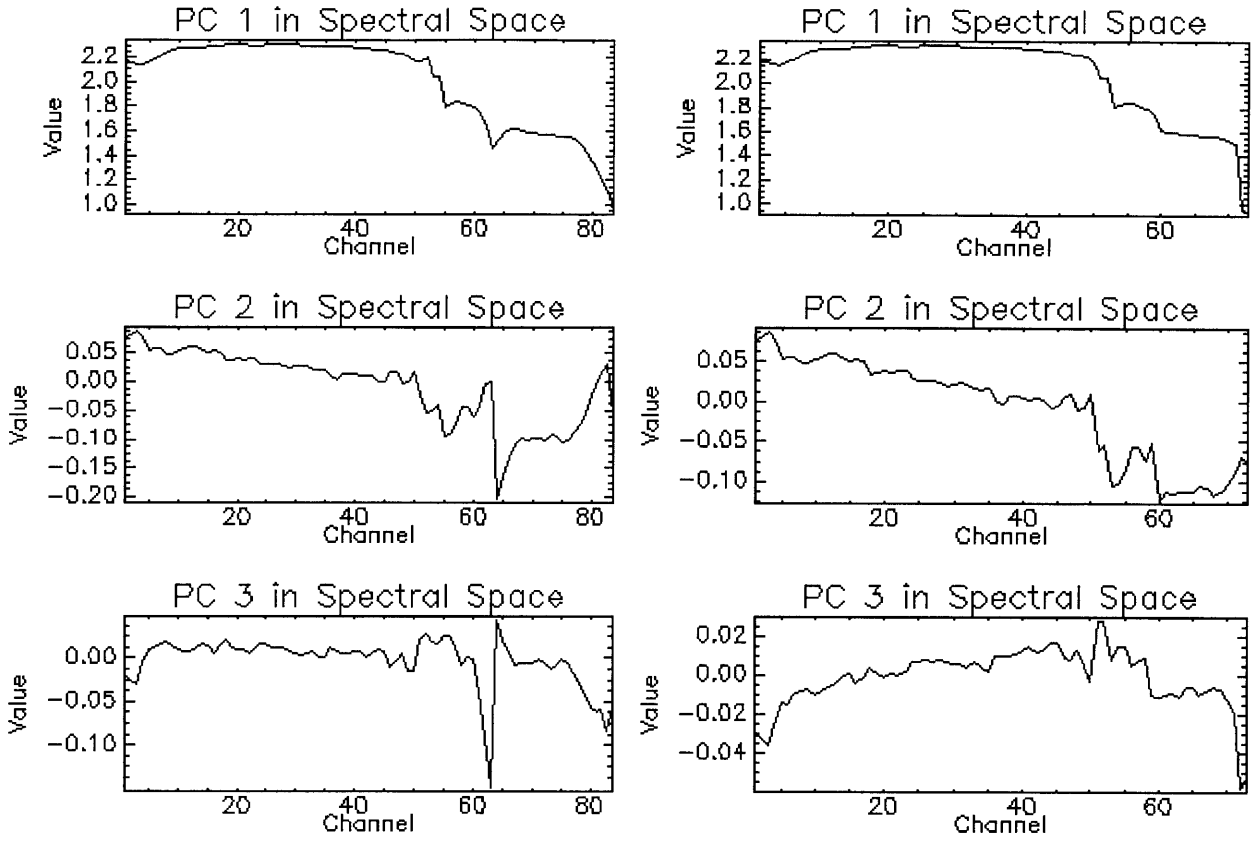
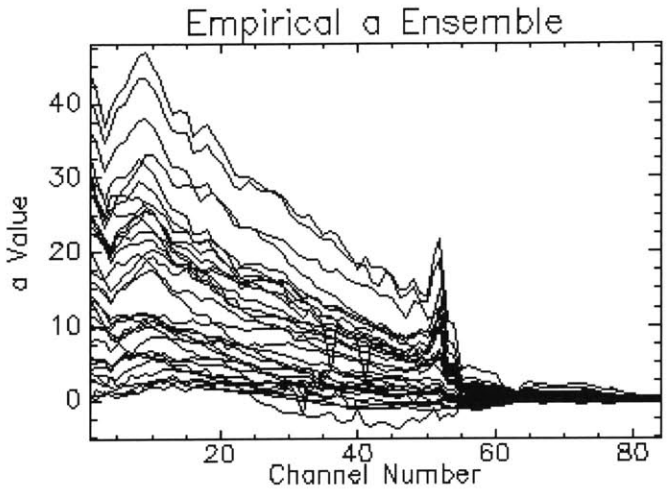


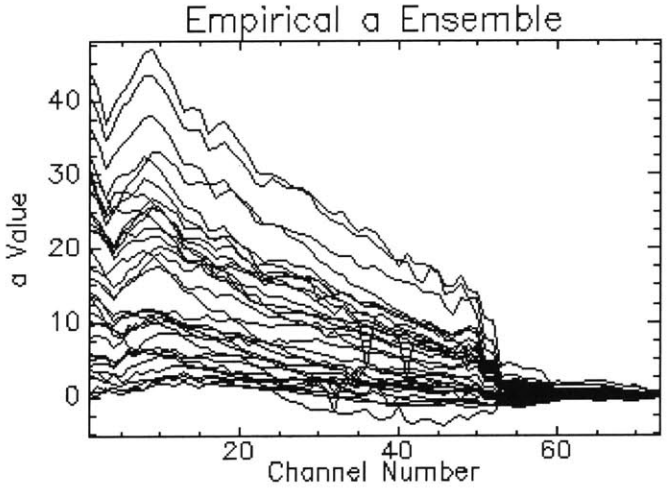
Figure 4.15. *First three PCs of empirical log* **m** *ensembles plotted in spectral space, before (a) and after (b) removing channels at the edges of water vapor absorption bands. In (a), center channel wavelength drift was causing spiking near channels 53 and 65 where drift in the center wavelength caused channel radiance values to rise or fall as the channels moved in and out of the water vapor absorption bands. In (b), these channels have been removed. After removal the spiking is reduced, especially in PCs 2 and 3.*

contributing to sensor noise). Near the edges of water absorption bands, this drift can make the measured radiances fall and rise as the center wavelength drifts in and out of the absorption band. This introduces a false degree of freedom into the noise ensembles which affect Principal Components analyses.

Figure 4.14 shows the empirical log **m** ensembles before (a) and after (b) removing channels at the edges of water vapor absorption bands. In Figure 4.14(a), center channel



(a)



(b)

Figure 4.16. *Empirical* **a** *ensembles before (a) and after (b) removing channels at the edges of water vapor absorption bands. In (a), center channel wavelength drift was causing spiking near channel 53 where drift in the center wavelength caused channel radiance values to rise or fall as the channels moved in and out of the water vapor absorption bands. In (b), these channels have been removed. After removal there are still discontinuous transitions where channels have been removed, but the transitions do not vary in size with center wavelength drift as they did in (a).*

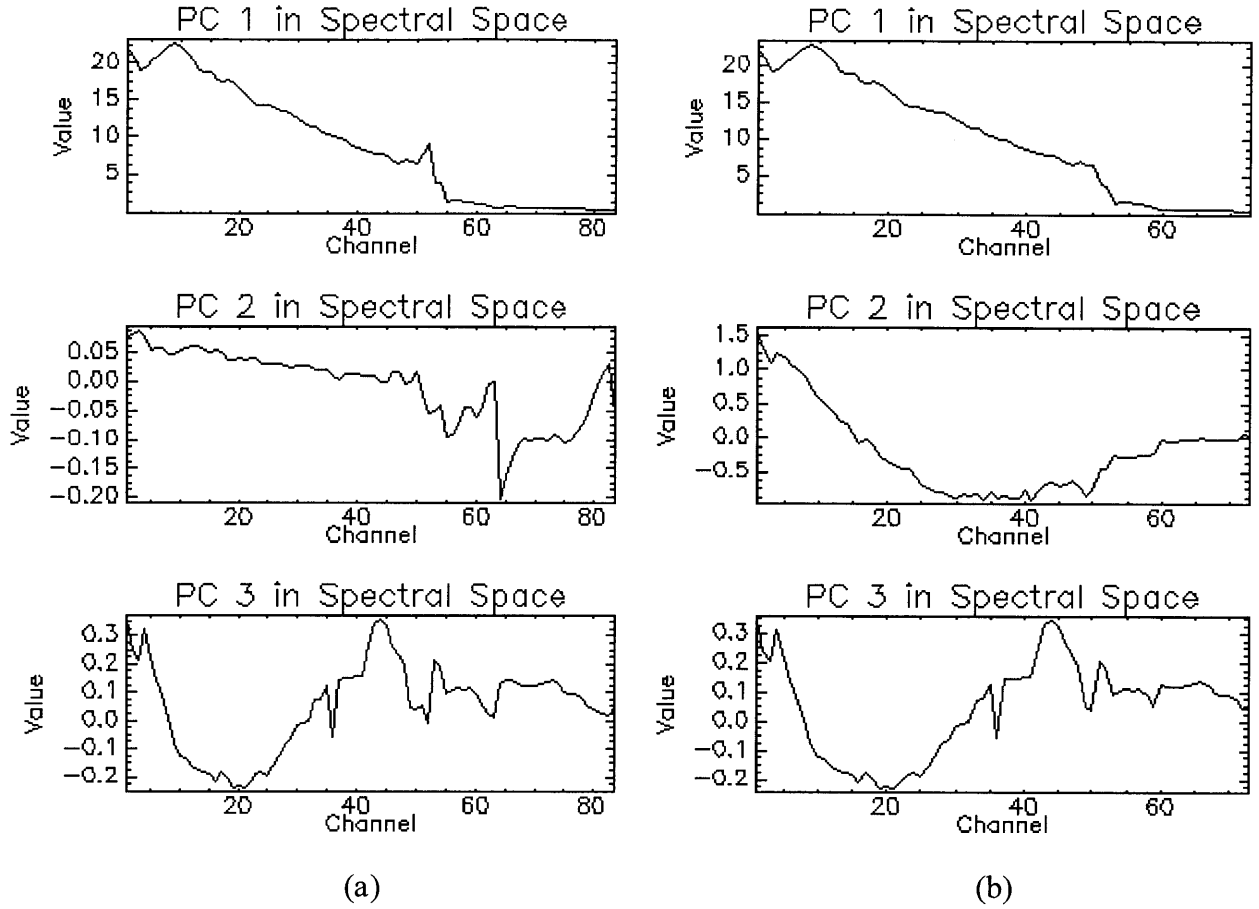


Figure 4.17. *First three PCs of empirical* **a** *ensembles plotted in spectral space, before (a) and after (b) removing channels at the edges of water vapor absorption bands. In (a), center channel wavelength drift was causing spiking near channels 53 and 65 where drift in the center wavelength caused channel radiance values to rise or fall as the channels moved in and out of the water vapor absorption bands. In (b), these channels have been removed. After removal the spiking is reduced, especially in PCs 1 and 2.*

wavelength drift was causing spiking near channels 53 and 65. In (b), these channels have been removed. After removal there are still discontinuous transitions where channels have been removed, but the transitions do not vary in size with center wavelength drift as they did in (a). In Figure 4.15 are plotted the first three PCs of the empirical log **m** ensembles plotted in spectral space, before (a) and after (b) removing channels at the edges of water vapor absorption bands.

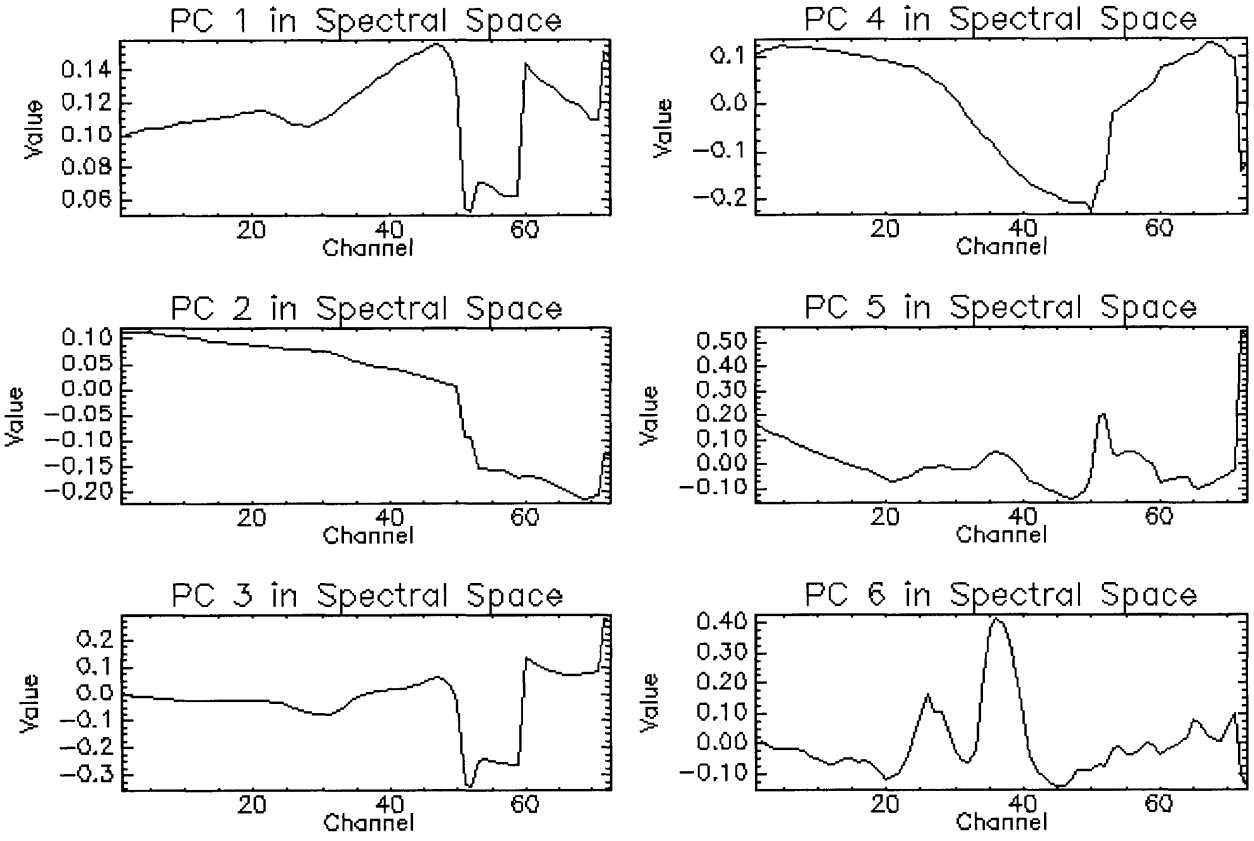


Figure 4.18. *First six PCs of the Run 07 ELM estimated reflectance cube. A value of one in each PC was rotated back into spectral space using the PC rotation matrix derived from a PCA of the ELM estimated reflectance cube. Note that PCs 1 and 3 have features (inverted) related to the grass that dominates the scene.*

In Figure 4.15(a), center channel wavelength drift was causing spiking near channels 53 and 65 where drift in the center wavelength caused channel radiance values to rise or fall as the channels moved in and out of the water vapor absorption bands. In Figure 4.15(b), these channels have been removed. After removal the spiking is reduced, especially in PCs 2 and 3. Figures 4.16 and 4.17 show similar plots for the empirical **a** ensembles before and after removing channels at the edges of water vapor absorption bands.

The water absorption and edge channels, channels with greater than 2% of pixels with

negative numbers in Run 22, additional channels that did not have valid ELM solutions in the non-ARM Site HYDICE campaigns were dropped, and one channel that exhibited a striping artifact (channel 19) were removed from the data: 1-6, 19, 60-79, 82-92, 100-117, 130-170, 173-210, leaving 73 spectral channels.

Figure 4.18 plots the first six PCs of reflectance in spectral space after removal of the bands described above. For each PC, the value of one in the PC and 0 in all others was rotated back into spectral space using the PC rotation matrix generated by a standard PCA of the Run 07 ELM estimated reflectance. Note that PCs 1 and 3 have features (inverted) related to the grass that dominates the scene.

4.8 Summary and Conclusions

In this chapter we laid the ground work for developing Spectral and Combined SPIRE algorithms which rely on Principal Components Analysis. We developed two techniques for focusing noise signal mean into low principal components. The first, Abutted Principal Components, uses the prior reflectance to turn the log multiplicative noise mean into a degree of freedom. Second, Zero-padded Principal Components uses zero vectors in a similar way to force the mean of a noise ensemble into low principal components. We also discussed the HYDICE channels dropped from the data due to water vapor absorption, negative radiance values, and channel center wavelength drift.

From this chapter we draw the following conclusions:

- APC has slightly better performance in a given image than ZPC and requires no prior knowledge about the illumination noise.
- However, APC only works for the multiplicative-noise-only case, and a new PC rotation matrix must be calculated for each image.

- ZPC works almost as well as APC, and the PC rotation can be precomputed from the required prior noise ensembles.
- ZPC works for multiplicative and additive noise cases. Therefore, ZPC is most appropriate for airborne remote sensing applications and will be used in this thesis.
- The HYDICE multiplicative noise **m** ensemble contains 3-4 degrees of freedom while the additive noise **a** ensemble contains 2-3 degrees of freedom. Approximately 6 degrees of freedom are present in the reflectance **r** vectors in the HYDICE test data set.
- Using ZPC, approximately 5 log **m**-PCs and 4 **a**-PCs should be filtered to eliminate the illumination noise effects.

We are now prepared to move on to the development of Spectral SPIRE algorithms in the next chapter.

Chapter 5

Spectral SPIRE Algorithms

5.1 Introduction

In this chapter, we develop Spectral SPIRE reflectance estimation algorithms that filter in the spectral dimension, operating on a single pixel at a time and making no use of spatial relationships between pixels. In Chapter 3 we developed Spatial SPIRE algorithms under the assumptions that the multiplicative and additive noises were band limited to lower spatial frequencies, and that any changes since the prior were band limited to higher spatial frequency. This led to a spatial frequency filtering strategy that removed the lower frequency noise terms without removing the higher frequency changes. In this chapter we develop analogous spectral Principal Component filtering techniques to remove the multiplicative and additive noise.

5.2 Spectral Principal Component Filtering

The multiplicative and additive noise vectors are not limited to any specific spectral bands or sensor channels. Therefore, we cannot directly perform a spectral filtering operation analogous to the high-pass spatial filtering done in Chapter 3 for Spatial SPIRE. Instead we must use the zero-padded principal components (ZPC) or abutted principal components (APC) to collect the multiplicative or added noise into a few low PCs. This has the effect of band limiting the noise in the rotated spectral PC space. Once this is done, we can perform the desired filtering analogous to that done in Spatial SPIRE.

To perform this spectral filtering, we define the concept of a Principal Components Filter (PC filter or PCF). Figure 5.1 defines a high pass PCF denoted by PCF_H, using the **m**-only case

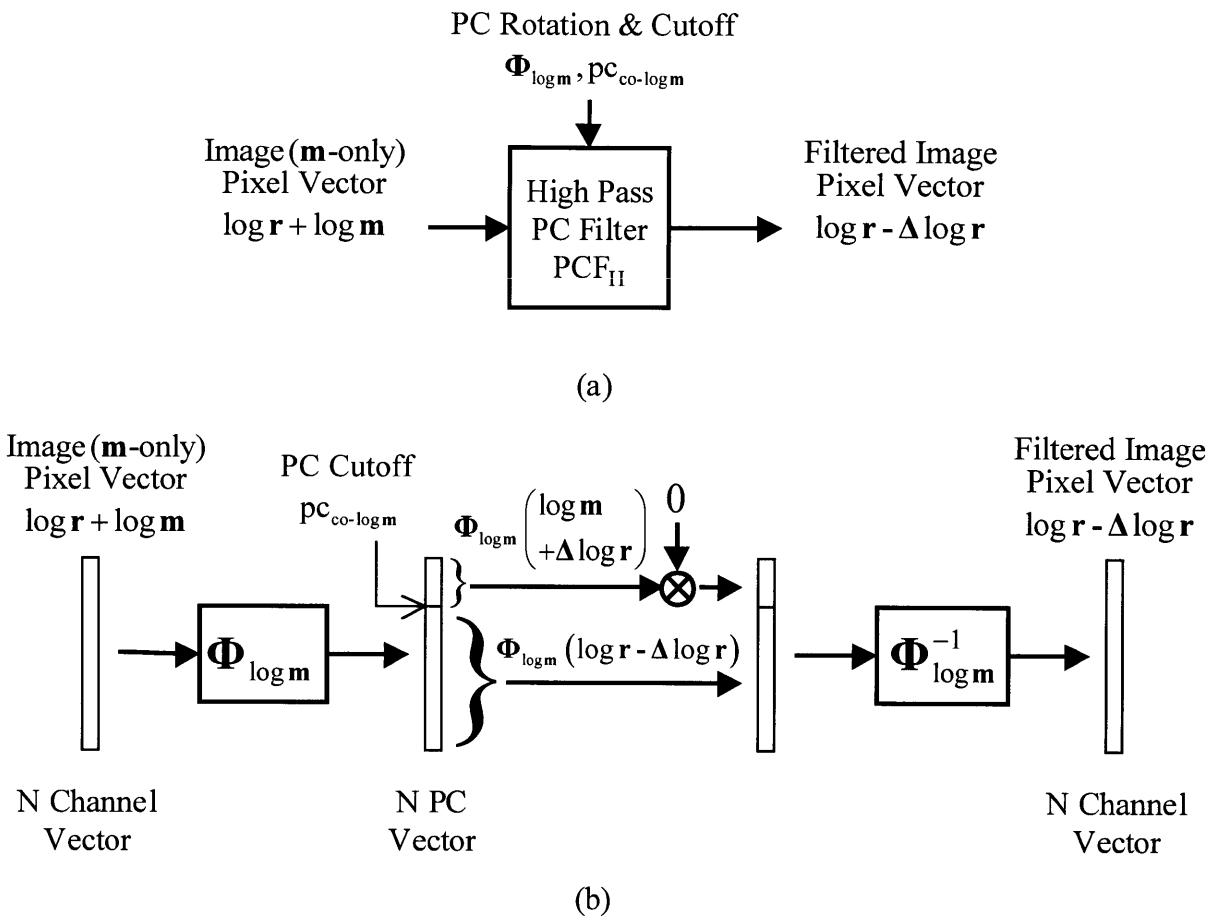


Figure 5.1. *High pass PC filter: PCF_H input and output definition (a) and detailed processing block diagram (b). The image pixel vector is first rotated into PC space using the* $\Phi_{\log \mathbf{m}}$ *PC rotation matrix. The low PCs numbered 1 through the* $\mathbf{pc}_{co\text{-}\log \mathbf{m}}$ *PC cutoff index contain* $\log \mathbf{m}$ *and are zeroed (rejected), removing* $\log \mathbf{m}$ *along with a portion of* $\log \mathbf{r} : \Delta \log \mathbf{r}$. *The higher PCs are passed through and finally rotated back into the original spectral space.*

were we have already moved to log space so that our image pixel vector consists of:

$$\log \mathbf{i} = \log \mathbf{r} + \log \mathbf{m} \tag{5.1}$$

The PC rotation matrix $\Phi_{\log \mathbf{m}}$ and PC filter cutoff $pc_{co\text{-}\log \mathbf{m}}$ comes from a ZPC analysis of $\{\log \mathbf{m}\}$ or APC analysis of the abutted log image and log prior $\log \mathbf{i} \cup \log \mathbf{r}_p$ as described in Chapter 4. In the PCF the original N-channel vector is first rotated into the $\log \mathbf{m}$ PC space:

$$\mathbf{\Phi}_{\log \mathbf{m}} \log \mathbf{i} = \mathbf{\Phi}_{\log \mathbf{m}} (\log \mathbf{r} + \log \mathbf{m}) = \mathbf{\Phi}_{\log \mathbf{m}} \log \mathbf{r} + \mathbf{\Phi}_{\log \mathbf{m}} \log \mathbf{m} \qquad (5.2)$$

We assume that this perfectly collects all of the rotated $\mathbf{\Phi}_{\log \mathbf{m}} \log \mathbf{m}$ noise into the lower PCs (PCs 1-$pc_{co\text{-}\log \mathbf{m}}$) which are shown in Figure 5.1 as being at the top of the N PC vector. Some portion of $\mathbf{\Phi}_{\log \mathbf{m}} \log \mathbf{r}$, denoted as $\mathbf{\Phi}_{\log \mathbf{m}} (\Delta \log \mathbf{r})$, also ends up in the $\log \mathbf{m}$ PCs (1-$pc_{co\text{-}\log \mathbf{m}}$). A boxcar high pass PC filter is now applied which zeroes out the lower PCs containing $\mathbf{\Phi}_{\log \mathbf{m}} (\log \mathbf{m} + \Delta \log \mathbf{r})$, and passes the higher PCs containing $\mathbf{\Phi}_{\log \mathbf{m}} (\log \mathbf{r} - \Delta \log \mathbf{r})$. This filtered PC vector is then rotated back into the original spectral space. Note that the rotated $\mathbf{\Phi}_{\log \mathbf{m}} \log \mathbf{m}$ signal will typically not all be collected into the lower numbered PCs, so a term $\Delta \log \mathbf{m}$ will be present in the final filtered vector in real applications. Figure 5.2 defines a low pass PC filter, PCF_L, in a similar manner. The final filtered vector is assumed to contain all of the $\log \mathbf{m}$ signal and $\Delta \log \mathbf{r}$, but will be missing a small $\Delta \log \mathbf{m}$ in real applications.

We note at this point, to be perfectly analogous to the processing done in Spatial SPIRE, we need to assume that any changes in $\log \mathbf{r}$ since the prior will end up in the higher number PCs above $pc_{co\text{-}\log \mathbf{m}}$) so that a high pass PCF does not eliminate them. If this is true, then we can perform a filtering operation that removes the band limited noise, yet does not remove the changes in reflectance. For now, we will proceed on the assumption that this is true, or if violated, only a small amount of the changes in $\log \mathbf{r}$ are lost so that it has a small effect on the estimated reflectance.

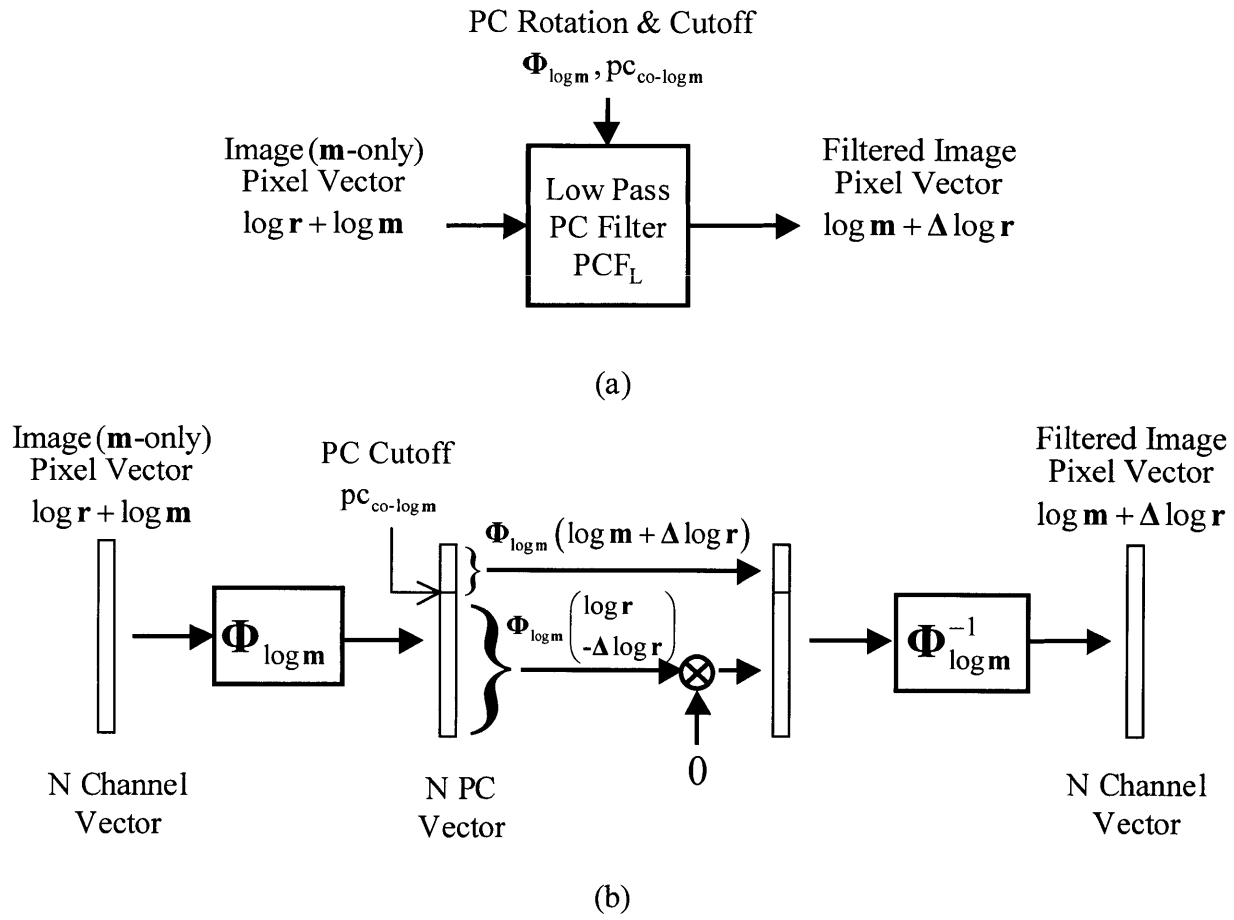


Figure 5.2. *Low pass PC filter: PCF_L input and output definition (a) and detailed processing block diagram (b). The image pixel vector is first rotated into PC space using the $\Phi_{\log \mathbf{m}}$ PC rotation matrix. The low PCs numbered 1 through the* $\mathbf{pc}_{\text{co-log}\mathbf{m}}$ *PC cutoff index contain* $\log \mathbf{m}$ *and are passed, along with a portion of* $\log \mathbf{r} : \Delta \log \mathbf{r}$. *The higher PCs containing the rest of* $\log \mathbf{r}$ *are zeroed (rejected). The filtered PCs are then rotated back into the original spectral space.*

5.3 Spectral SPIRE Algorithm Derivation

5.3.1 Algorithmic Approach and Issues

Our overall approach is very similar to that described in Chapter 3 for Spatial SPIRE algorithms. The same assumption of an existing prior is still in force. The Spatial SPIRE assumptions that the multiplicative noise is slowly spatially varying and that changes are small in

area with respect to the scene, are not required for the solution of the Spectral SPIRE **m**-only case. However, for the solution of the multiplicative and additive noise case, Spectral SPIRE algorithms are limited to cases where reflectance changes are small relative to the scene, since we remove the additive noise **a** with a technique similar to the Spatial SPIRE Case 4 and 6 algorithms, where the minimum mean square error (MSE) of the estimated reflectance with respect to the prior is used. This also limits us to cases where the additive noise **a** is the same for all pixels in the image ensemble, (equivalent to **a** being spatially uniform) since the minimum MSE technique can only estimate a single value at a time.

We begin with the image formation equation (2.2) presented in Chapter 2 expressed in vector notation:

$$\mathbf{i} = \mathbf{r} \odot \mathbf{m} + \mathbf{a} \tag{5.3}$$

We shall again consider two cases, one where **a** is present, the other where it is not. Since the spatial variability of **m** is not an issue, we shall not differentiate between the Spectral SPIRE Cases A and C (spatially uniform and slowly varying *m*, respectively) and simply refer to both **m**-only cases, and also ones in which **m** is quickly or arbitrarily spatially varying, as Case C. We shall consider only one case with both **m** and **a**, in which **m** can be arbitrarily varying while **a** is the same for all pixels. This constraint on **a**, along with the assumption that reflectance changes are small in area, are necessary to employ the iterative MSE algorithmic approach developed in Chapter 3 for Spatial SPIRE Cases 4 and 6. This will be referred to as Case D. Case D is similar to Spatial SPIRE Case 4, but **m** can be arbitrarily varying. No Spectral SPIRE solutions have yet been developed for cases where both **m** and **a** are spatially varying as is discussed in the Section 5.7.

5.3.2 Case C: Multiplicative Noise Only

The Spectral SPIRE Case C algorithm is now derived. We note that the Spectral Case A algorithm would be identical to the Spectral Case C, since the spatial variability of **m** does not affect the spectral processing for either case. In this case the additive noise **a** is zero. Our image formation model (5.3) then becomes:

$$\mathbf{i} = \mathbf{r} \odot \mathbf{m} \tag{5.4}$$

We first move to log space to linearize the problem:

$$\log \mathbf{i} = \log \mathbf{r} + \log \mathbf{m} \tag{5.5}$$

Figure 5.3(a) depicts the generalized processing block diagram for the Case C solution algorithm. This algorithm is analogous to the Case 3 general Spatial SPIRE algorithm of Figure 3.2(a). Figure 5.3(b) depicts the specific algorithm implemented in this thesis and tested upon the same HYDICE test data set as the Spatial SPIRE algorithms. Note the similarity between Figure 5.3 and Figure 3.2.

In Spatial SPIRE each spectral channel image was processed independently. In Spectral SPIRE, each image pixel is processed independently. Each pixel is passed through the high-pass PC filter PCF_H to zero the $\log \mathbf{m}$ PCs and thereby eliminate $\log \mathbf{m}$. We denote such a PC filtering operation as $PCF_{H/L\text{-}var}(\ )$, so that:

$$\log \mathbf{r} - \Delta \log \mathbf{r} = PCF_{H-\log \mathbf{m}}(\log \mathbf{r} + \log \mathbf{m}) \tag{5.6}$$

Where $PCF_{H\text{-}\log \mathbf{m}}(\ )$ performs a low pass PCF using the $\mathbf{\Phi}_{\log \mathbf{m}}$ rotation matrix and the PC cut off $pc_{co\text{-}\log \mathbf{m}}$. Note that by using this notation we do not need $\mathbf{\Phi}_{\log \mathbf{m}}$ or $pc_{co\text{-}\log \mathbf{m}}$ explicitly in

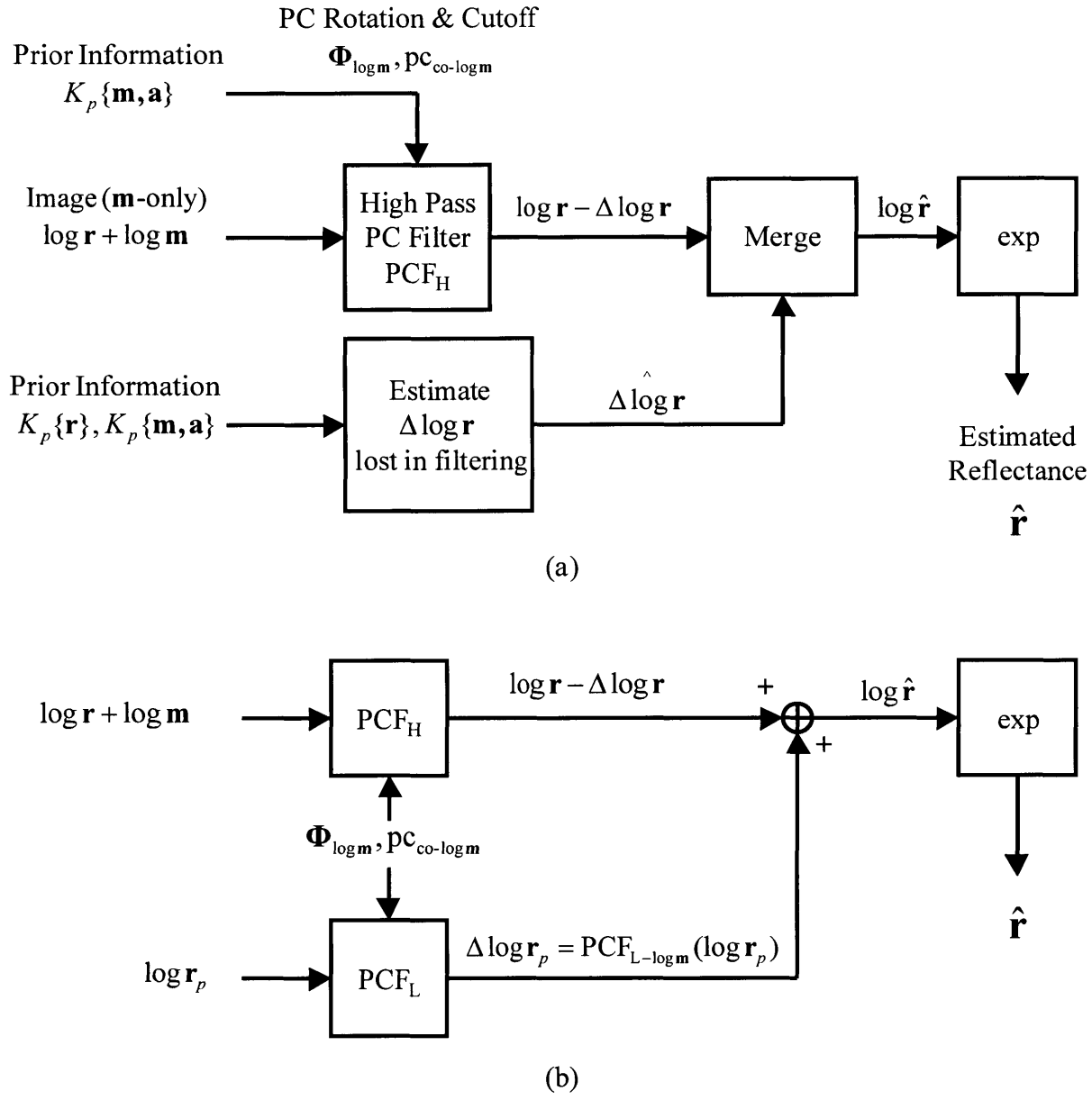


Figure 5.3. *Spectral Case C generalized processing block diagram (a) and specific implementation tested (b). In (a), the* **m***-only image is high-pass PC filtered using the* $\Phi_{\log \mathbf{m}}$ *PC rotation and* $pc_{\text{co-log}\mathbf{m}}$ *PC cutoff index from the prior information about the multiplicative noise* **m** *($K_p\{\mathbf{m},\mathbf{a}\}$). The low-pass PC filtered* log **r** *lost in the filtering operation is then estimated using prior information $K_p\{\mathbf{r}\}$ and $K_p\{\mathbf{m},\mathbf{a}\}$ and merged with the filtered signal. Finally, the exponential is taken to estimate reflectance. In (b), the lost low pass PC filtered* log **r** *is estimated by low-pass filtering the log of the prior reflectance* $\log \mathbf{r}_p$*, and the merge operation is a simple addition.*

our equations. We must now estimate $\Delta \log \mathbf{r}$, which we do by low pass PC filtering the prior $\log \mathbf{r}_p$:

$$\hat{\Delta \log \mathbf{r}} = \Delta \log \mathbf{r}_p = \mathrm{PCF}_{\mathrm{L}-\log \mathbf{m}}(\log \mathbf{r}_p) \tag{5.7}$$

This estimate of $\Delta \log \mathbf{r}$ is then merged with $\log \mathbf{r} - \Delta \log \mathbf{r}$ through a simple addition to estimate $\log \mathbf{r}$:

$$\begin{aligned} \log \hat{\mathbf{r}} &= \log \mathbf{r} - \Delta \log \mathbf{r} + \hat{\Delta \log \mathbf{r}} \\ &= \mathrm{PCF}_{\mathrm{H}-\log \mathbf{m}}(\log \mathbf{r} + \log \mathbf{m}) + \mathrm{PCF}_{\mathrm{L}-\log \mathbf{m}}(\log \mathbf{r}_p) \end{aligned} \tag{5.8}$$

And finally take the exponential to estimate reflectance:

$$\hat{\mathbf{r}} = \exp(\log \hat{\mathbf{r}}) \tag{5.9}$$

5.3.2.1 Matching Against A Prior Spectral Library to Restore $\Delta \log \mathbf{r}$

There is an alternative technique for restoring the $\Delta \log \mathbf{r}$ lost in the high pass PC filtering in the Case C Spectral SPIRE algorithm other than low pass PC filtering $\log \mathbf{r}_p$ and adding it back in. Figure 5.4 depicts this alternative algorithm. If a prior spectral library of known materials $\{\log \mathbf{r}_l\}$ is available, then it can be low pass PC filtered instead of $\log \mathbf{r}_p$. Then, each $\log \mathbf{r} - \Delta \log \mathbf{r}$ vector can be matched against this filtered library using a minimum vector distance criteria. Once the closest match is found, the original library spectrum $\log \mathbf{r}_l$ is then used to replace the $\log \mathbf{r} - \Delta \log \mathbf{r}$ vector, effectively restoring the lost $\Delta \log \mathbf{r}$. An additional benefit of such an algorithm is that it can handle large area changes.

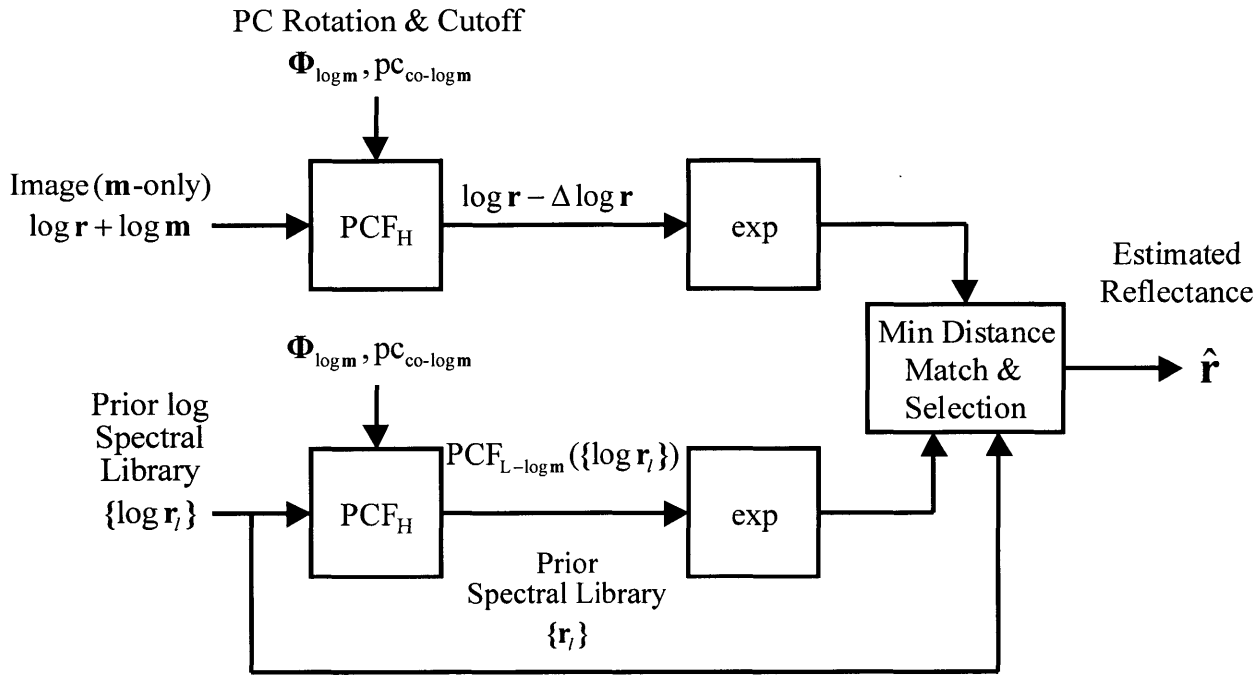


Figure 5.4. *Spectral Case C implementation using matching against a prior spectral library to restore lost* $\Delta \log \mathbf{r}$. *Both the image and the spectral library of known materials* $\{\log \mathbf{r}_l\}$ *are high pass PC filtered and exponentiated. Matching is then done using a minimum distance criteria where distance is measured as the length of the difference vector between the filtered image pixel vector and each filtered library spectrum. Once the closest match is identified, the unfiltered library spectrum is then used as the estimate of reflectance at that pixel.*

Such an approach can work well, but only if spectral reflectance vectors of all materials, including mixed vectors of materials that may be in the same pixel, are in the spectral library. If a material is not present in the spectral library, then large errors can result. Also, such an approach can be computationally intractable, especially if the number of materials in the library is large and there are many different percentages and permutations of mixing between them.

Figure 5.5 depicts the results from a test of this algorithm using simulated data. A simulated reflectance cube was generated with large area changes, including one that is of an "unknown" material with 50 percent uniform spectral reflectance. A multiplicative noise containing high spatial frequency steps and ramps between the steps was simulated and applied

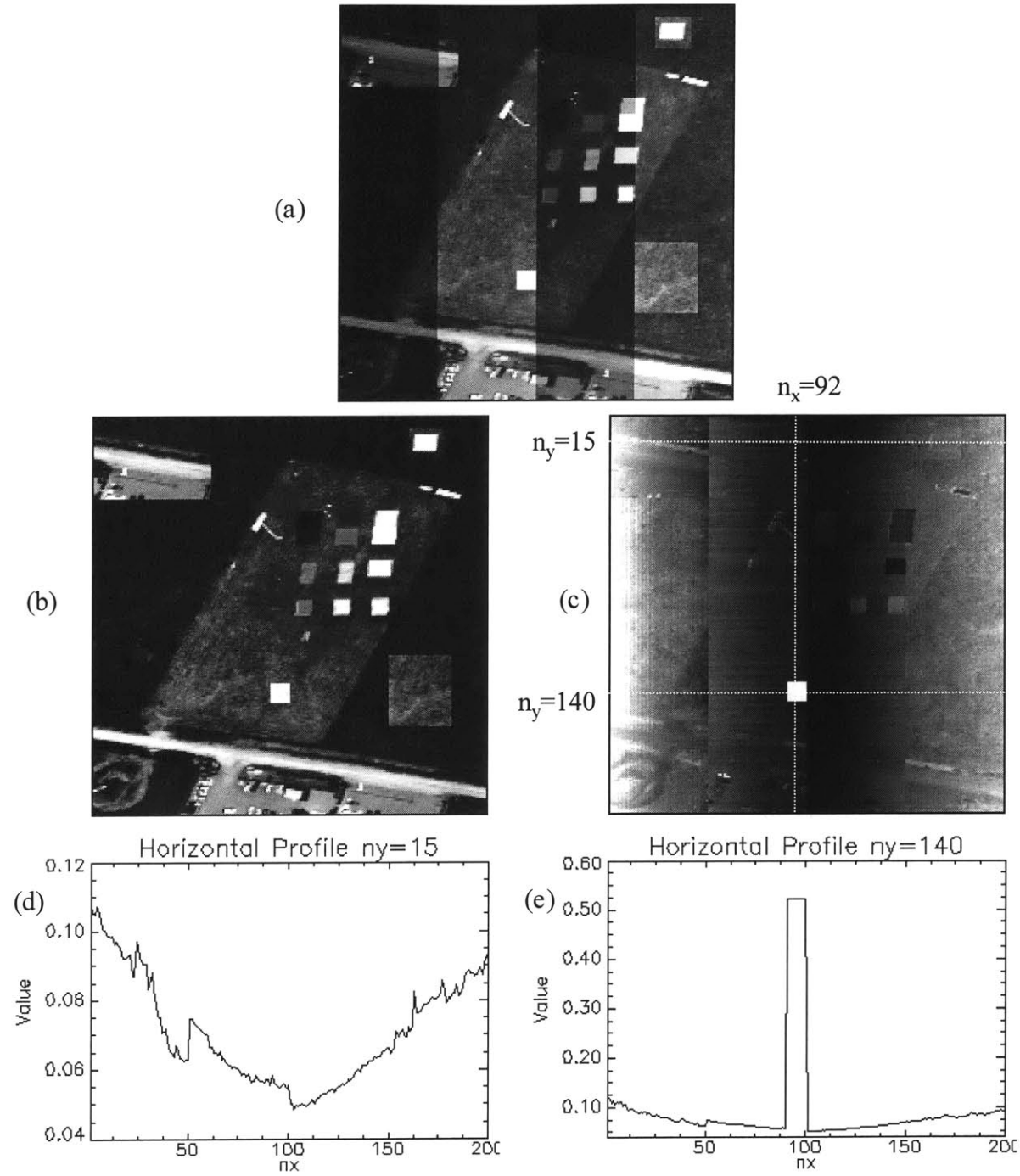


Figure 5.5. *Example of Spectral SPIRE Case C algorithm using spectral library matching and abutted-PC analysis. (a) Modified reflectance test image cube with quickly varying multiplicative noise. Estimated reflectance (b) has perfect performance using unmodified reflectance cube as spectral library, except at unknown panel at [92,140]. Matching distance image (c) shows the distance to the best match in the spectra library for each pixel, with horizontal profiles through n_y=15 and n_y=140 (d) and (e). Variation in (d) is due to small amounts of log* ***m*** *noise in higher PCs, but matching overcomes this noise. The unknown panel has highest distance and large error since its material is not present in the spectral library.*

to the reflectance image as shown in Figure 5.5(a).

Abutted PC analysis was used to generated the PC rotation matrix $\mathbf{\Phi}_{\log \mathbf{m}}$. The estimated reflectance image shown in Figure 5.5(b) was generated using the prior reflectance cube as the prior spectral library. It matches perfectly the true reflectance of the scene, except for the unknown material that was not present in the library. Figure 5.5(c) shows the minimum distance image, which displays the distance to the best match in the spectral library, where brighter color indicates longer distance. Figure 5.5(d) and (e) show profile plots through the distance image. Note the large distance for the unknown material at $[n_x,n_y]=[100,140]$. Note that there was no noise present in the spectral library, which is an idealized condition. Further details about this experiment can be found in Appendix D.

One can consider such a matching against a spectral library part of the post processing that is done after estimating reflectance. In many remote sensing applications, the estimates of reflectance are run through a classifier to identify material types. In this case, our classifier is using each spectrum in the spectral library as a class, and we are using only the information in the higher PCs to do the classification. One could also run the results of Spatial SPIRE through such a classifier as well, to fine tune the reflectance estimates of known materials.

Since such an approach can be regarded as post-processing classification, it will not be further pursued in this thesis. We recognize that some classifiers can work on the subset of non-log **m** PCs created by a high-pass PC filter, and their performance would need to be compared against the results of Spatial SPIRE for classification. Such efforts are beyond the scope of this thesis but fall under possible areas of further research.

5.3.3 Case D: Multiplicative and Ensemble Uniform Additive Noise

The Spectral SPIRE Case D algorithm is now derived. We note that the Spectral Case B algorithm would be identical to the Spectral Case D, since the spatial variability of **m** does not affect the spectral processing for either case. To solve this case we continue with our approach of developing spectral PC filtering techniques analogous to the Spatial SPIRE techniques. The solution technique derived in this section is similar to the Spatial SPIRE Case 4 algorithm, which estimate the amount of *rm* to restore after filtering out *a* by selecting the *rm* amount to minimize the mean squared error between the estimated reflectance and the prior reflectance.

Since we make no use of spatial relationships between pixels, we are limited to using information within the single image pixel being processed, and information that can be derived from the ensemble of image pixels as a whole, independent of their spatial relationships.

We begin with the image formation equation (5.3), but with an additive noise that is the same for all pixels in the image, which will be referred to as ensemble uniform, (which is equivalent to being spatially uniform):

$$\mathbf{i} = \mathbf{r} \odot \mathbf{m} + \mathbf{a} \tag{5.10}$$

The solution algorithm for this case is depicted in Figure 5.6. The full image ensemble of pixels is first rotated into **a**-PC space using $\mathbf{\Phi}_\mathbf{a}$:

$$\mathbf{\Phi}_\mathbf{a}\mathbf{i} = \mathbf{\Phi}_\mathbf{a}\left(\mathbf{r} \odot \mathbf{m} + \mathbf{a}\right) = \mathbf{\Phi}_\mathbf{a}\left(\mathbf{r} \odot \mathbf{m}\right) + \mathbf{\Phi}_\mathbf{a}\mathbf{a} \tag{5.11}$$

Then, each of the **a**-PCs from 1 to $pc_{co\text{-}a}$ is stepped through one at a time, the current PC being denoted by a subscripted *n* (PC_n). In Figure 5.6 the Do loops are denoted using standard FORTRAN flow chart notation where "Do NNN" sets up a loop down to the step with the label

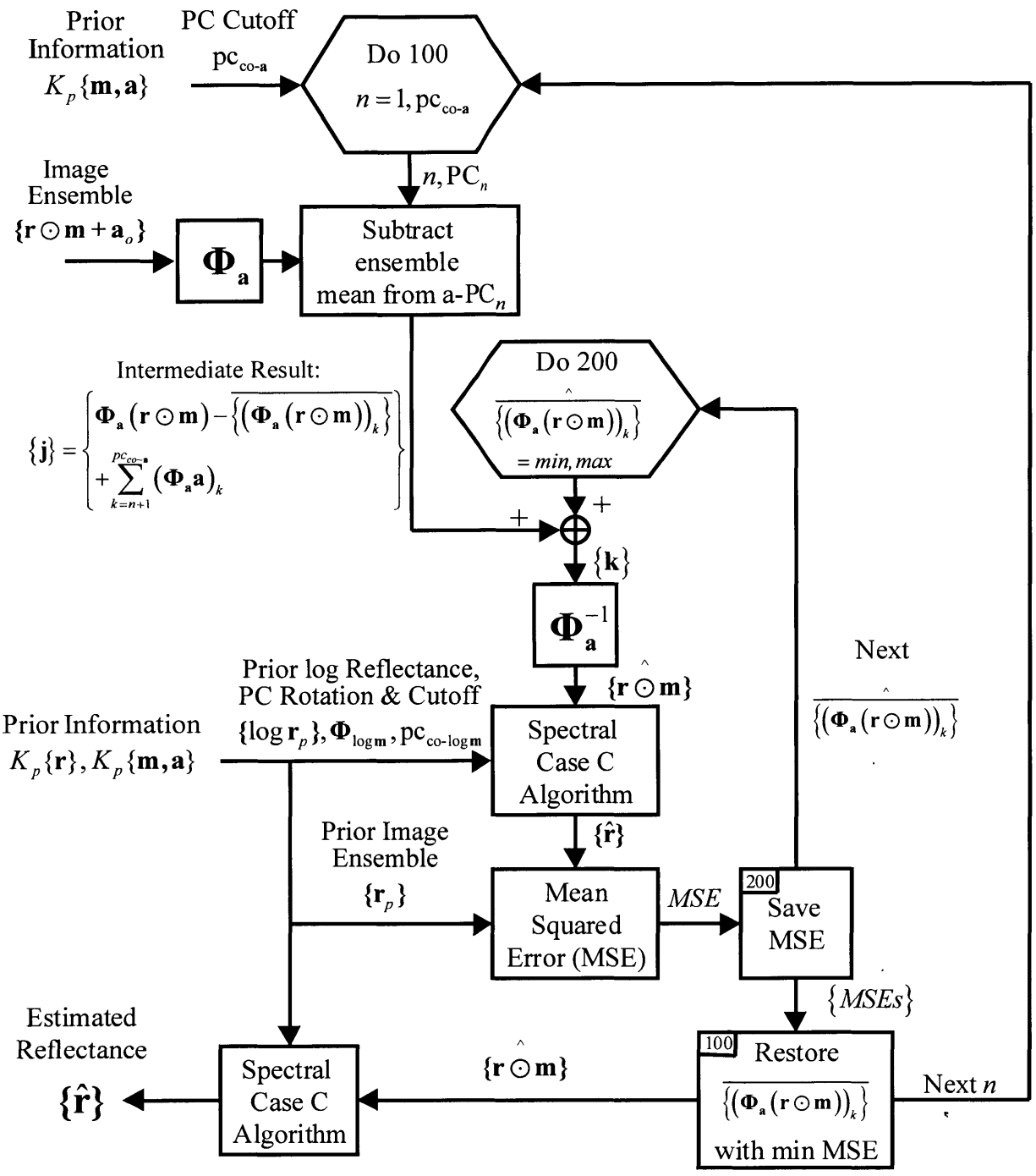


Figure 5.6. *Spectral Case D processing block diagram. Based on the additive noise* **a** *PC cutoff* $pc_{co\text{-}a}$*, all of the* **a***-PCs are looped through. For each* **a***-*PC_n*, the ensemble mean of the the PC is subtracted to remove the* **a** *in that PC. An inner loop then steps through the possible values for the lost mean of* $\mathbf{r}\odot\mathbf{m}$ *in* PC_n*, using the Spectral Case C algorithm and MSE minimization as in the Spatial Case 4 algorithm. Once all of the* **a***-PCs have been processed, then the Case C algorithm is run one last time to estimate the reflectance.*

NNN. For each **a**-PC, the mean of the PC over the ensemble is subtracted to eliminate the rotated constant **a** term in that PC, which also removes the ensemble mean of $\mathbf{r} \odot \mathbf{m}$ from that PC. In addition, estimates of the ensemble mean of $\mathbf{r} \odot \mathbf{m}$ from previous PCs have been restored, resulting in the following intermediate term at this point:

$$\{\mathbf{j}\} = \left\{ \begin{array}{l} \mathbf{\Phi}_{\mathbf{a}} (\mathbf{r} \odot \mathbf{m} + \mathbf{a}) - \sum_{k=1}^{n} \text{mean}\left\{ \left(\mathbf{\Phi}_{\mathbf{a}} (\mathbf{r} \odot \mathbf{m} + \mathbf{a}) \right)_k \right\} \\ + \sum_{k=1}^{n-1} \text{est}\left(\text{mean}\left\{ \left(\mathbf{\Phi}_{\mathbf{a}} (\mathbf{r} \odot \mathbf{m}) \right)_k \right\} \right) \end{array} \right\} \tag{5.12}$$

Where {} denotes that we are dealing with an ensemble of pixels and the following notation is used:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_{k-1} \\ x_k \\ x_{k+1} \\ \vdots \\ x_N \end{bmatrix}, \quad \mathbf{x}_k = \begin{bmatrix} 0 & & \cdots & & 0 \\ 0 & \ddots & & & \\ & & 0 & & \\ \vdots & & & 1 & \vdots \\ & & & & 0 \\ & & & & & \ddots \\ 0 & & \cdots & & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_{k-1} \\ x_k \\ x_{k+1} \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ x_k \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{5.13}$$

Which defines the vector $\mathbf{x}_k$ as a vector with all zero elements except for the k-th element, which is equal to the k-th element of **x**. Based on the assumption that all of the rotated **a** is collected into the lowest $pc_{\text{co-}\mathbf{a}}$ PCs and none is present in the higher PCs we can write:

$$\sum_{k=pc_{\text{co-}\mathbf{a}}+1}^{N} \left(\mathbf{\Phi}_{\mathbf{a}} \mathbf{a} \right)_k = 0 \tag{5.14}$$

So that:

$$\begin{aligned}\boldsymbol{\Phi}_{\mathbf{a}}\mathbf{a} &= \sum_{k=1}^{n}\left(\boldsymbol{\Phi}_{\mathbf{a}}\mathbf{a}\right)_k + \sum_{k=n+1}^{N}\left(\boldsymbol{\Phi}_{\mathbf{a}}\mathbf{a}\right)_k \\ &= \sum_{k=1}^{n}\left(\boldsymbol{\Phi}_{\mathbf{a}}\mathbf{a}\right)_k + \sum_{k=n+1}^{pc_{co-\mathbf{a}}}\left(\boldsymbol{\Phi}_{\mathbf{a}}\mathbf{a}\right)_k\end{aligned} \tag{5.15}$$

We therefore can rewrite (5.12) as:

$$\{\mathbf{j}\} = \left\{\begin{array}{l}\boldsymbol{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right) + \sum_{k=n+1}^{pc_{co-\mathbf{a}}}\left(\boldsymbol{\Phi}_{\mathbf{a}}\mathbf{a}\right)_k - \sum_{k=1}^{n}\text{mean}\left\{\left(\boldsymbol{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right)\right)_k\right\} \\ + \sum_{k=1}^{n-1}\text{est}\left(\text{mean}\left\{\left(\boldsymbol{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right)\right)_k\right\}\right)\end{array}\right\} \tag{5.16}$$

Assuming that the restored estimates of the previous PCs means of $\mathbf{r}\odot\mathbf{m}$ are correct, we can simplify (5.16) to:

$$\{\mathbf{j}\} = \left\{\boldsymbol{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right) - \text{mean}\left\{\left(\boldsymbol{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right)\right)_k\right\} + \sum_{k=n+1}^{pc_{co-\mathbf{a}}}\left(\boldsymbol{\Phi}_{\mathbf{a}}\mathbf{a}\right)_k\right\} \tag{5.17}$$

Where the second term is the lost rotated $\mathbf{r}\odot\mathbf{m}$ to be estimated and restored in the current PC and the third term is simply the rotated $\mathbf{a}$ remaining in the processed PCs not yet processed. This is the "Intermediate Result" in Figure 5.6, in which an over bar signifies the ensemble mean:

$$\overline{\left\{\left(\boldsymbol{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right)\right)_k\right\}} = \text{mean}\left\{\left(\boldsymbol{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right)\right)_k\right\} \tag{5.18}$$

Each of the values represented by (5.18) is a single number that must be estimated and restored for each PC processed:

$$b = \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right)\right)_k\right\} \tag{5.19}$$

We therefore use an algorithm very similar to the Spatial SPIRE Case 4 algorithm to restore each b, in which we step through a range of possible values for the number. For each candidate value of b, we add the number back into all the pixels of PC_n, rotate the pixel ensemble back into the original spectral space, and run the Spectral SPIRE Case C algorithm described in Section 5.3.2. As in the Spatial Case 4 algorithm, we then calculate the MSE with respect to the prior reflectance and eventually select the value of b that minimizes the MSE and use it as the estimate of b. This results in:

$$\left\{\mathbf{k}\right\} = \left\{\begin{array}{l} \mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right) - \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right)\right)_k\right\} \\ + \sum_{k=n+1}^{pc_{co\text{-}\mathbf{a}}}\left(\mathbf{\Phi}_{\mathbf{a}}\mathbf{a}\right)_k + \text{est}\left(\text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right)\right)_k\right\}\right) \end{array}\right\} \tag{5.20}$$

Assuming that this is an accurate estimate of b, the net result of processing the nth PC is:

$$\left\{\mathbf{k}\right\} = \left\{\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right) + \sum_{k=n+1}^{pc_{co\text{-}\mathbf{a}}}\left(\mathbf{\Phi}_{\mathbf{a}}\mathbf{a}\right)_k\right\} \tag{5.21}$$

In which we have eliminated **a** from the PCs processed so far. This is repeated for each of the $pc_{co\text{-}\mathbf{a}}$ PCs to be processed. After processing all of the $pc_{co\text{-}\mathbf{a}}$ **a**-PCs, the third term is eliminated. After restoring the last estimate of the second term, and assuming that such estimates are accurate, we end up with:

$$\left\{\mathbf{k}\right\} = \left\{\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{r}\odot\mathbf{m}\right)\right\} \tag{5.22}$$

We then rotate this back into spectral space as:

$$\{\mathbf{r} \hat{\odot} \mathbf{m}\} = \mathbf{\Phi}_a^{-1}\{\mathbf{k}\} = \mathbf{\Phi}_a^{-1}\{\mathbf{\Phi}_a(\mathbf{r} \odot \mathbf{m})\} \tag{5.23}$$

Finally we run the Spectral Case C algorithm one last time to estimate reflectance.

5.3.4 Spectral SPIRE Case E and F Algorithms

We did not develop Spectral SPIRE Case E and F algorithms that could handle spatially varying additive **a** noise. By analogy to the Spatial Case 5 and 6 algorithms, we would begin by high-pass PC filtering **i** to remove **a**. This would leave us with the task of estimating and restoring the lost $\Delta\mathbf{rm}$ at every pixel. Since we do not know **m**, we do not have a prior **rm** from which to restore this information. While this is certainly an area for further work, the results of this thesis provide no obvious direction for proceeding.

5.4 Algorithm Testing and Results

The test data set of six ARM Site image cubes used for testing Spatial SPIRE in Chapter 3 was used to test the Case 4 Spectral SPIRE algorithm developed in this chapter. The same ELM and ATREM processing results from Chapter 3 were also compared with the Spectral SPIRE results. The same pixels from the images were used as well. Please refer to Section 3.3.4.4 for a description of the different pixel types used. Based on the results of Chapter 4 and preliminary experiments, 5 $\log\mathbf{m}-$PCs and 5 **a**-PCs were processed ($pc_{co\text{-}log\mathbf{m}}$=$pc_{co\text{-}\mathbf{a}}$=5). Processing fewer PCs resulted in noticeable errors in a few spectral channels, and processing more showed little improvement.

Figure 5.7 depicts the scatter plot of mean and standard deviation performance of Spectral SPIRE, ELM, and ATREM for estimation of surface spectral reflectance for all of the

19 pixel types selected. This figure is directly analogous to Figure 3.9 for Spatial SPIRE. The horizontal axis represents the average standard deviation over all the spectral channels, where the standard deviation in each channel was calculated over the reflectance estimates of all six Runs (06-31). The vertical axis is the RMS error over all the spectral channels for the mean reflectance estimate minus the mean reflectance estimate of ELM as ground truth, which is why ELM has zero RMS error. We see that Spectral SPIRE has better standard deviation and RMS error performance than ATREM, except for a single pixel which has far worse performance than ATREM, denoted by the triangle in the upper right. This data point corresponds to the modified pixel. Spectral SPIRE also has about the same standard deviation performance as ELM.

Figure 5.8 (a)-(s) is directly analogous to Figure 3.8 (a)-(s) for comparing Spectral SPIRE performance against that of ELM and ATREM. Similar plots with the vertical axis scaled from 0 to 1 can be found in Appendix E as Figure E.1. Appendix E also contains Tables E.1 and E.2 which are analogous to Tables C.1 and C.2 for Spatial SPIRE and present the average channel standard deviation values in each band for the ELM, ATREM, and Spectral SPIRE results plotted in Figure 5.8.

Again we note the clustering of the ATREM spectra into two distinct sets due to clouds and haze on three of the runs, as was discussed in Chapter 3. We also note that for all but the modified pixel, Spectral SPIRE performs better than ATREM and very similar to ELM. In fact, the performance of Spectral SPIRE is better than that of Spatial SPIRE on the unmodified pixels. In addition, Spectral SPIRE performs better than Spatial in the longer wavelength channels for the 2, 4, and 8 percent panels. Also, Spectral SPIRE often has lower variance than ELM for pixels like the 15 percent panel.

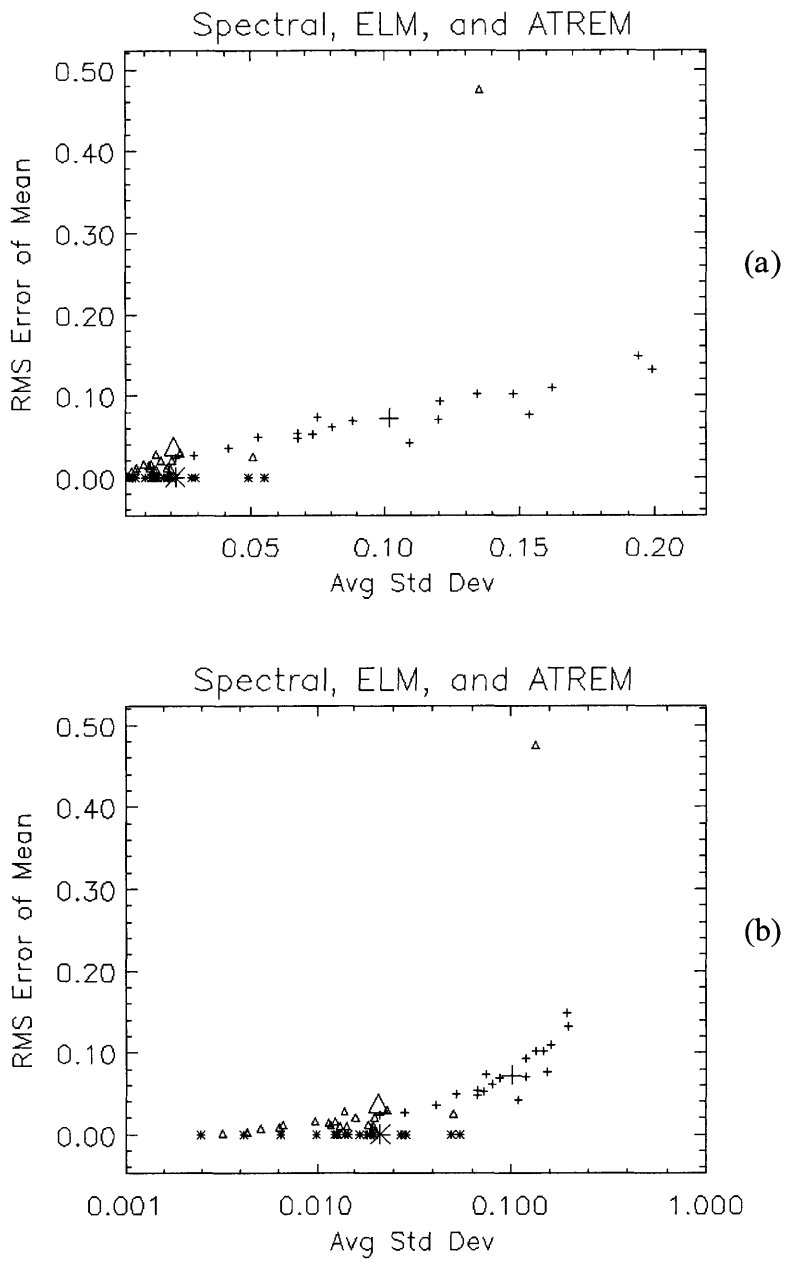


Figure 5.7. *Scatter plots of mean and standard deviation performance of Spectral SPIRE (triangles), ELM (*), and ATREM (+) for estimation of surface spectral reflectance for 19 pixel types. In both (a) and (b) the horizontal axis represents the average standard deviation over all the spectral channels, where the standard deviation in each channel was calculated over the reflectance estimates of all six Runs (Runs 06-31). The vertical axis is the RMS error over all spectral channels for the mean reflectance estimate minus the ELM mean reflectance estimate, which is why ELM has zero RMS error. The triangle in the upper right corresponds to the modified pixel. (b) is the same as (a), but with a log horizontal axis. The larger symbols represent the mean of the points plotted with that symbol. We see that Spectral SPIRE performs similarly to ELM in standard deviation and better than ATREM in both RMS and standard deviation performance.*

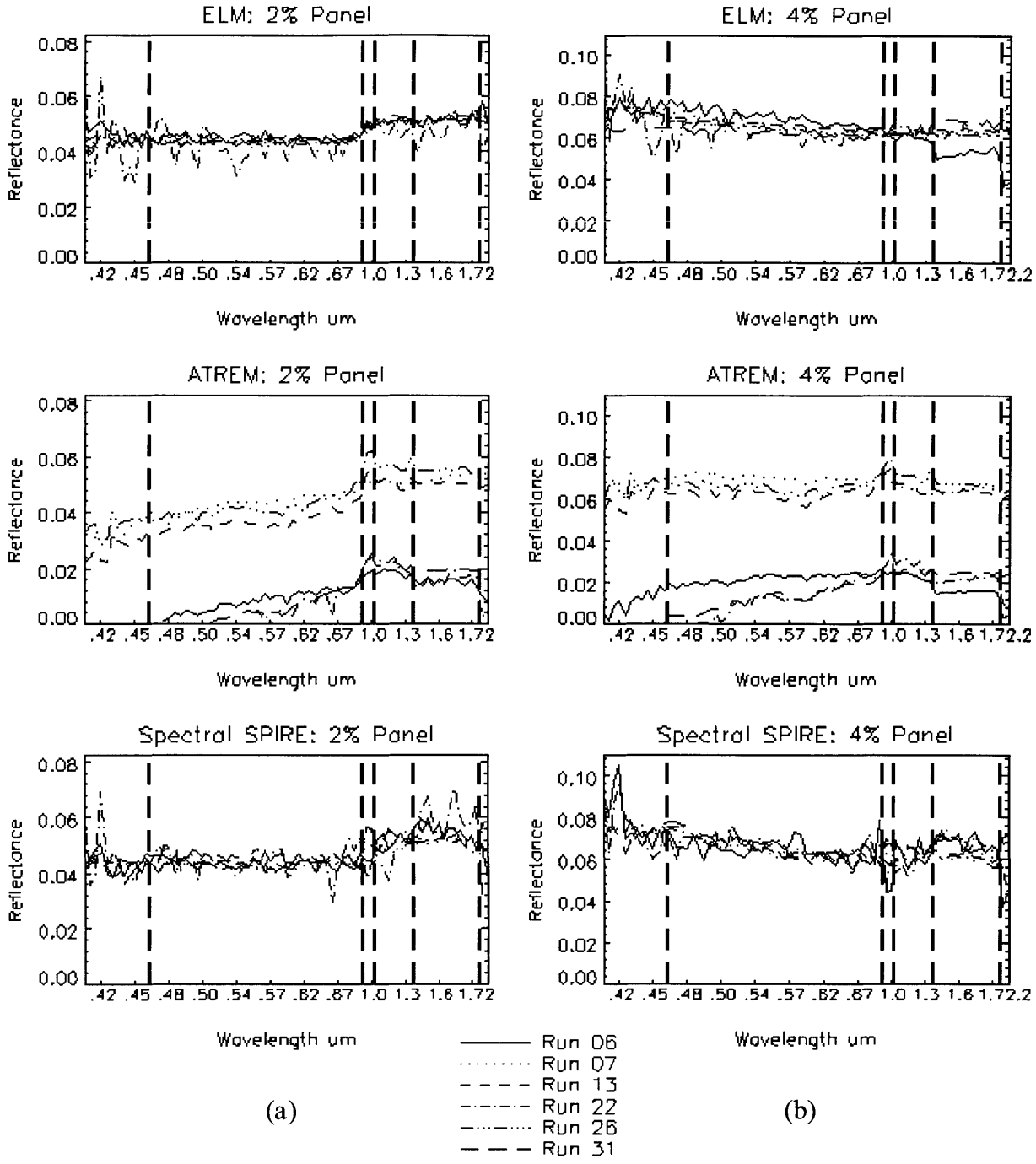


Figure 5.8 (a) and (b). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the 2 percent panel (a) and the 4 percent panel (b).*

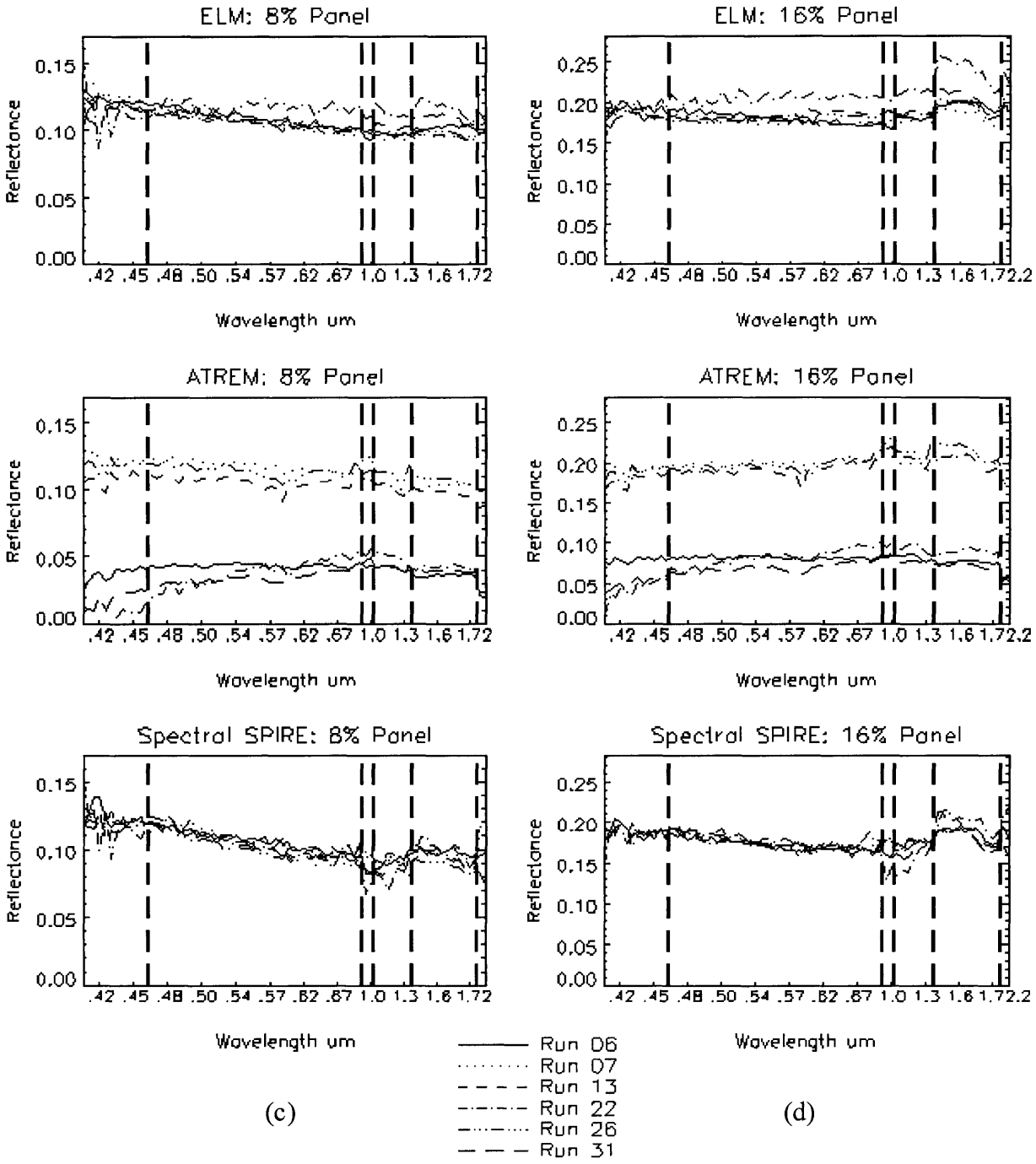


Figure 5.8 (c) and (d). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the 8 percent panel (c) and the 16 percent panel (d).*

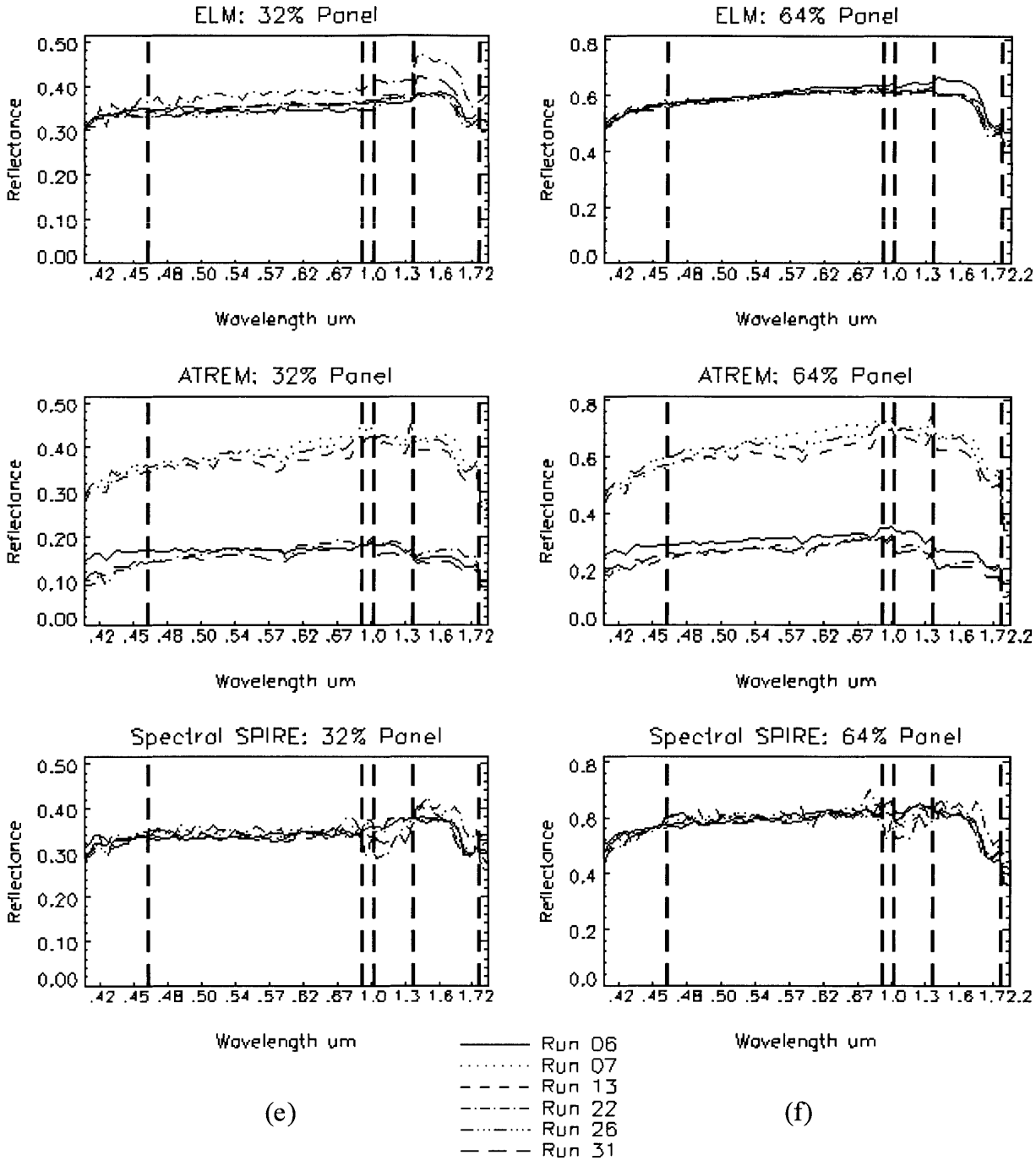


Figure 5.8 (e) and (f). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the 32 percent panel (e) and the 64 percent panel (f).*

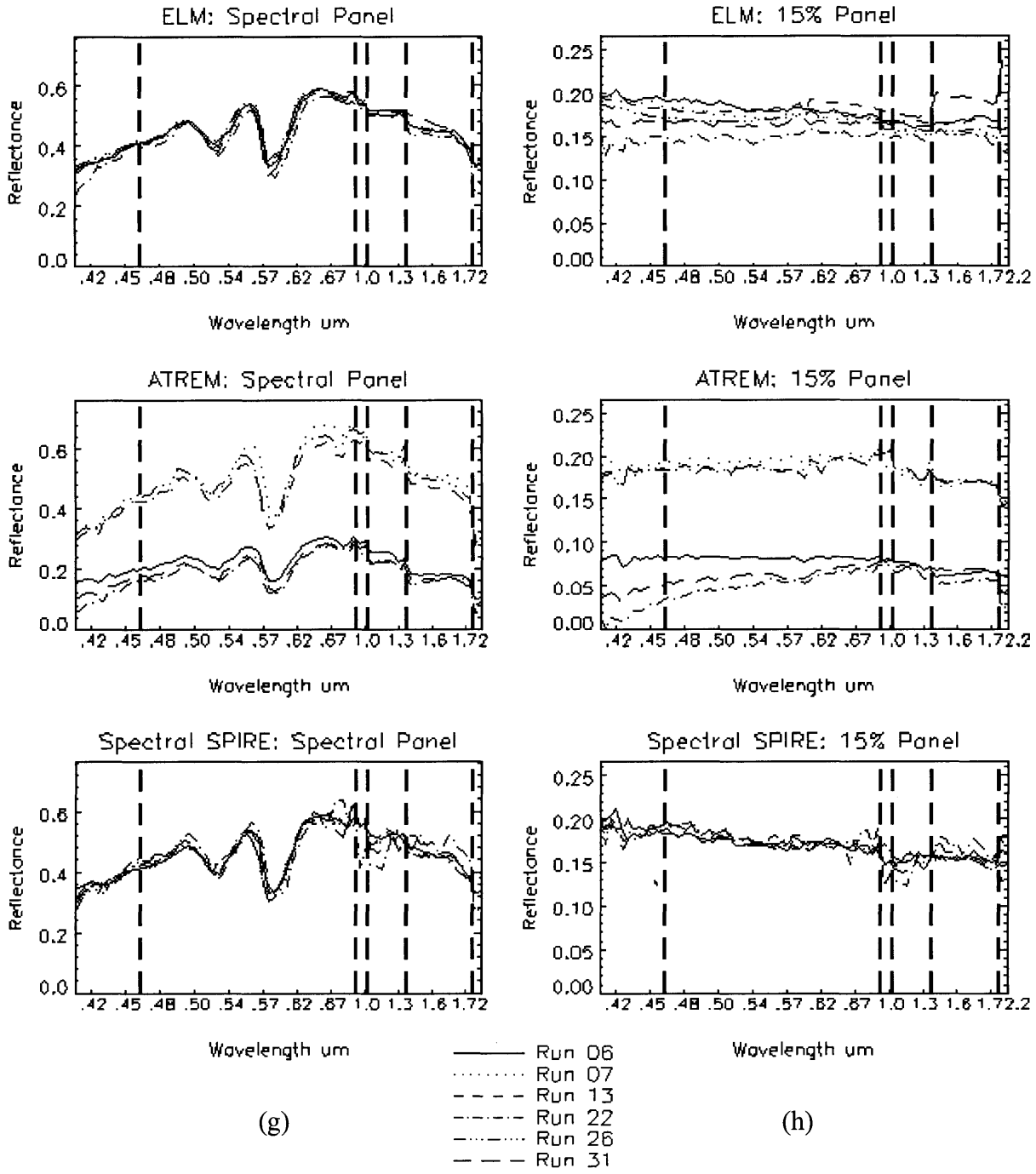


Figure 5.8 (g) and (h). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the spectral panel (g) and the 15 percent panel (h).*

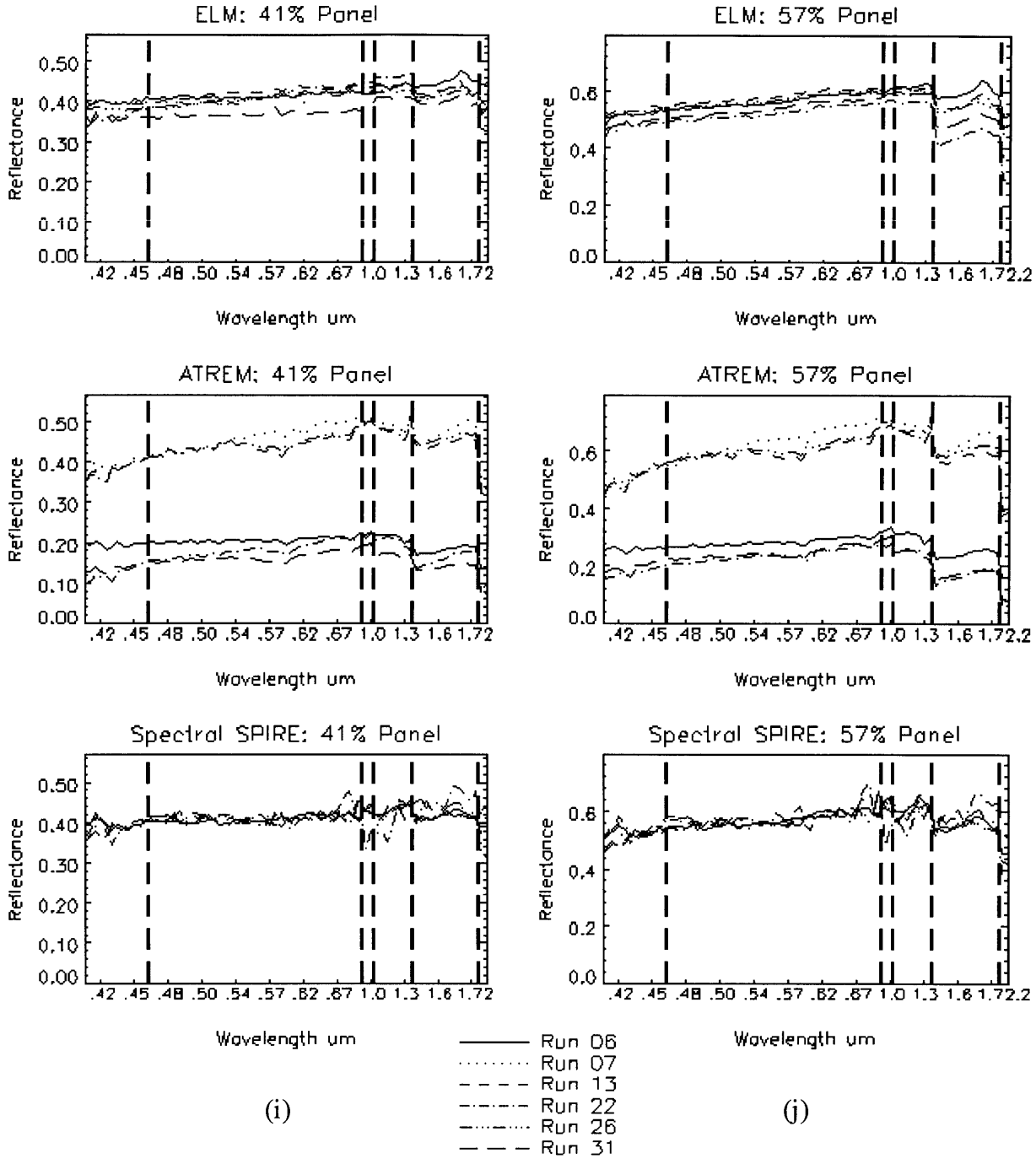


Figure 5.8 (i) and (j). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the 41 percent panel (i) and the 57 percent panel (j).*

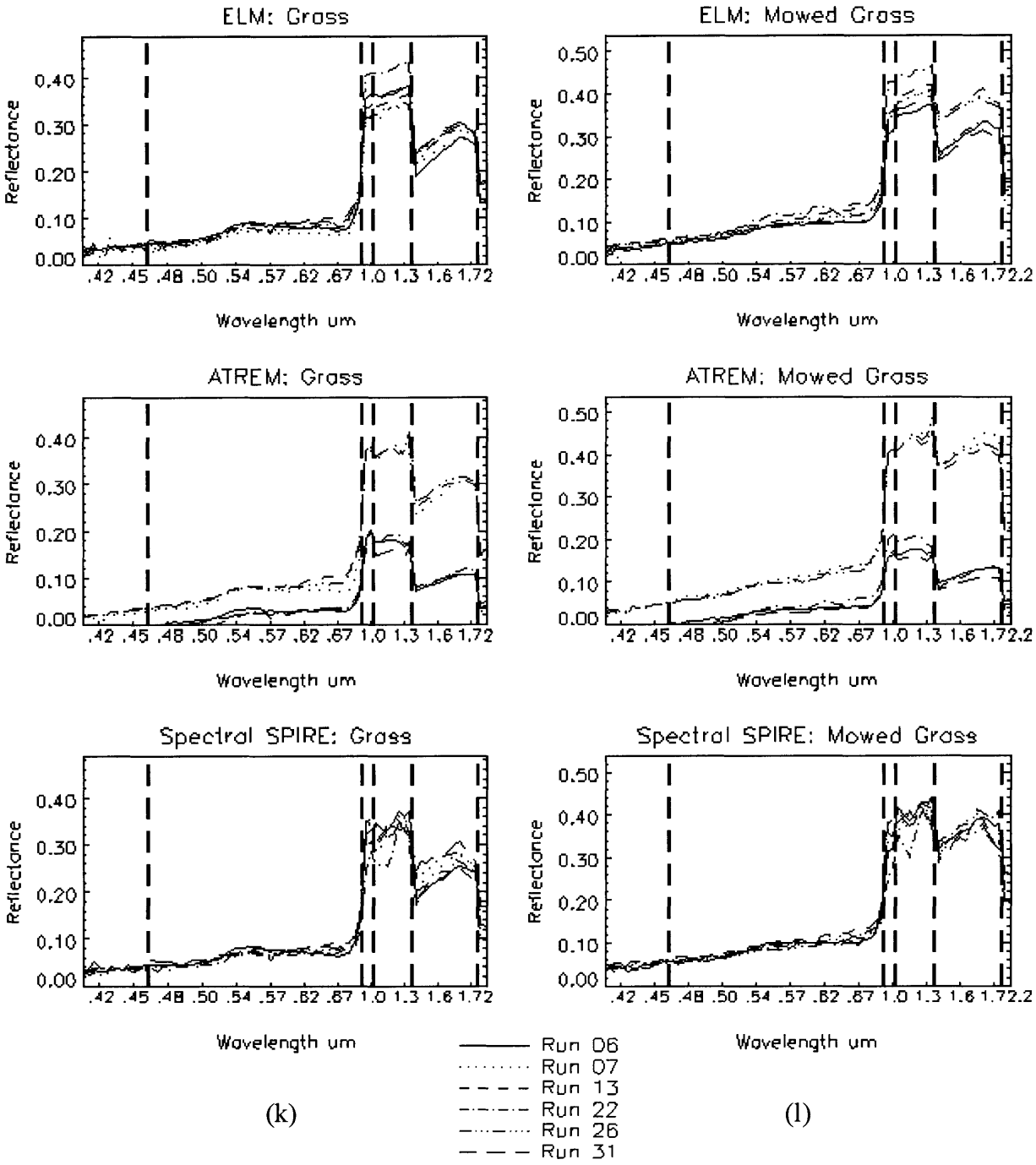


Figure 5.8 (k) and (l). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on grass (k) and mowed grass (l).*

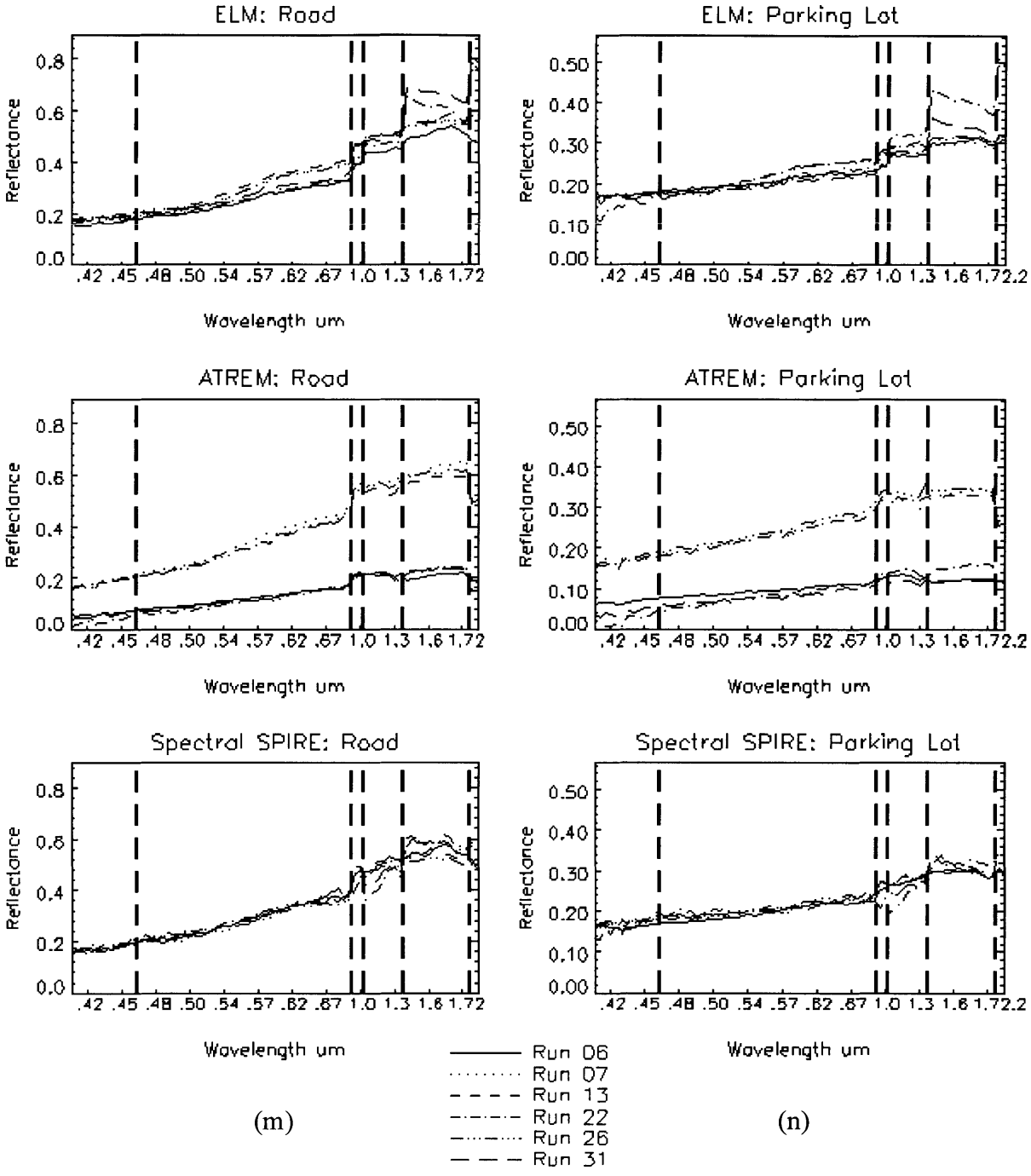


Figure 5.8 (m) and (n). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the road (m) and the parking lot (n).*

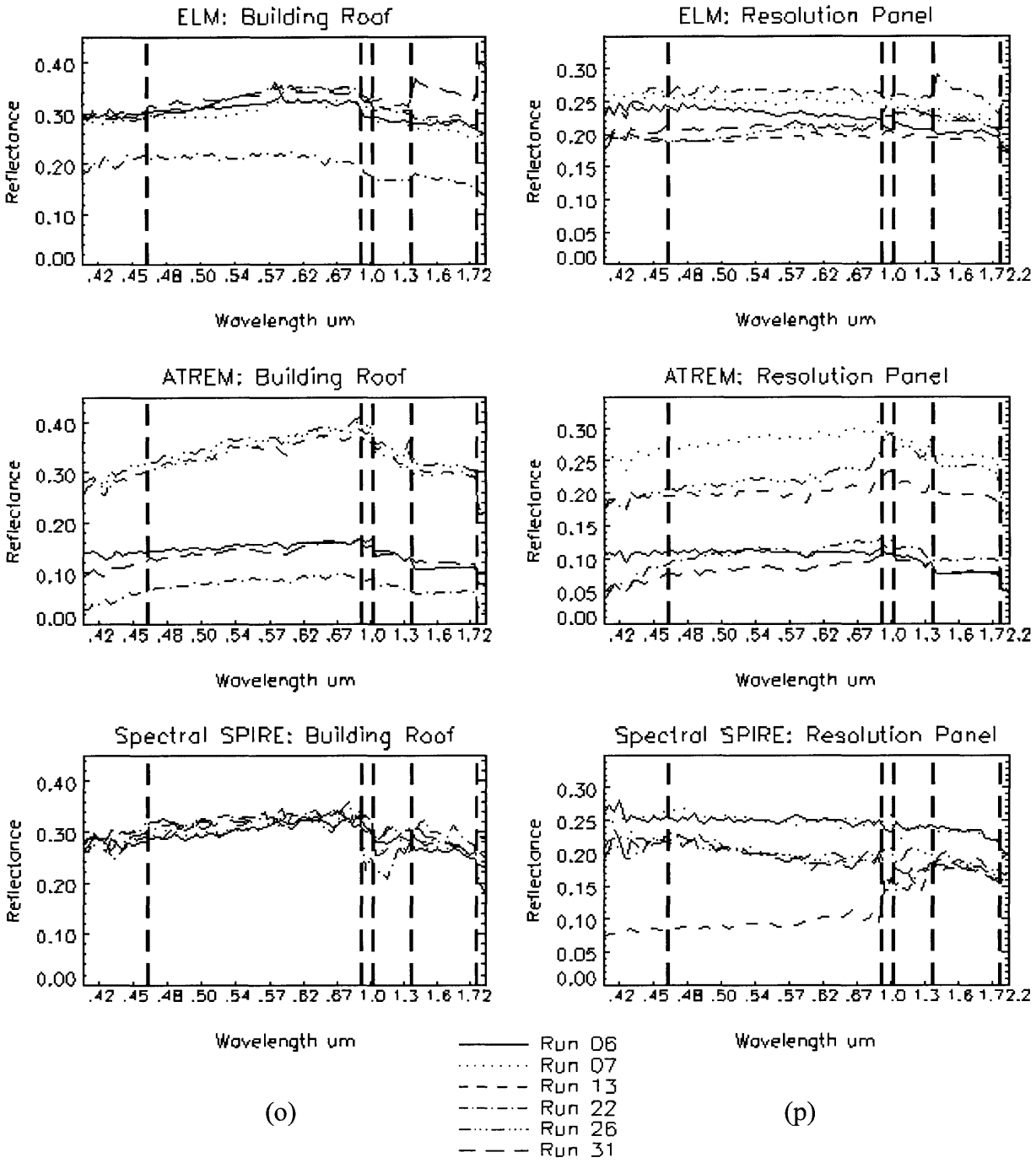


Figure 5.8 (o) and (p). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the building roof (o) and the resolution panel (p).*

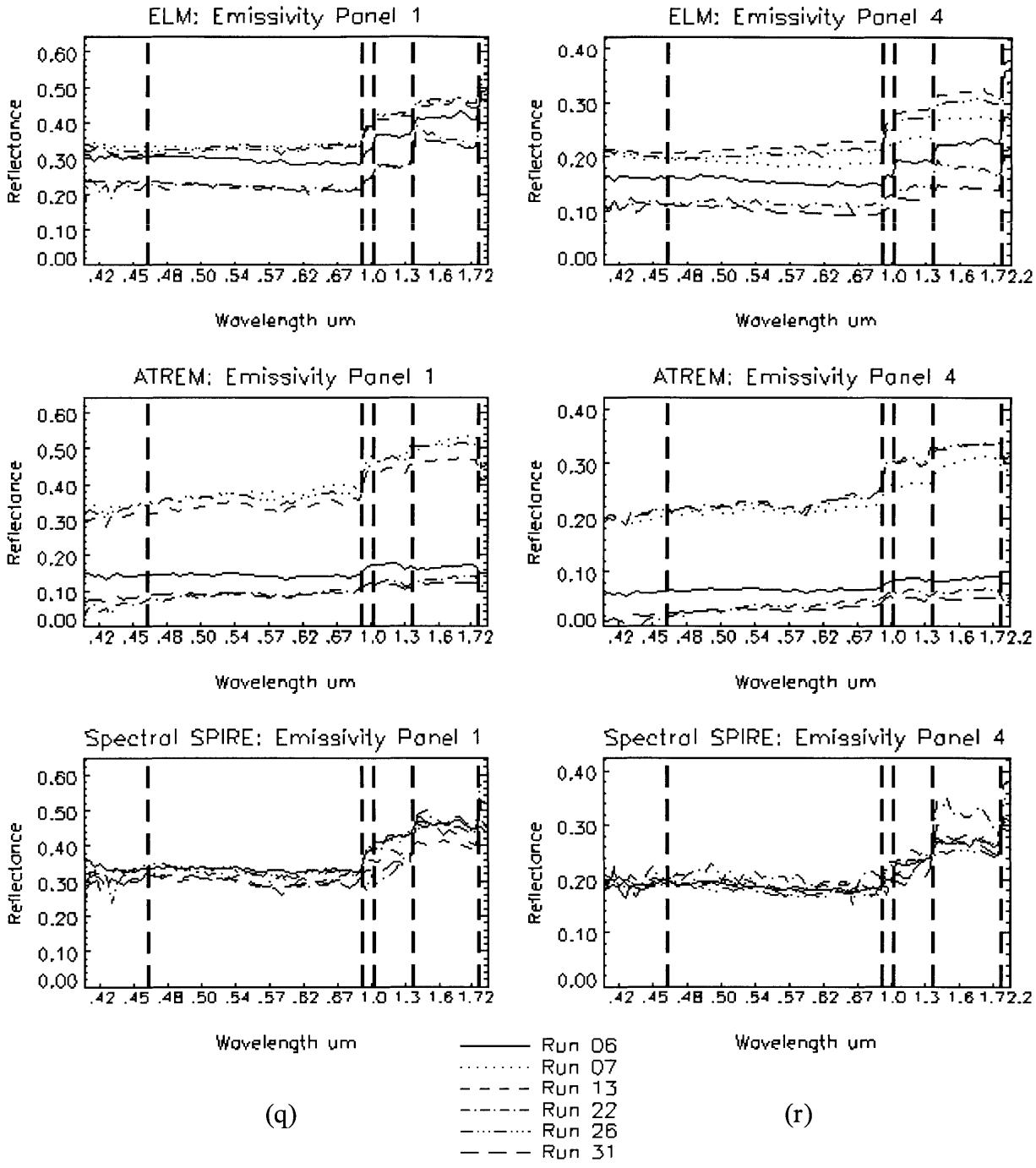
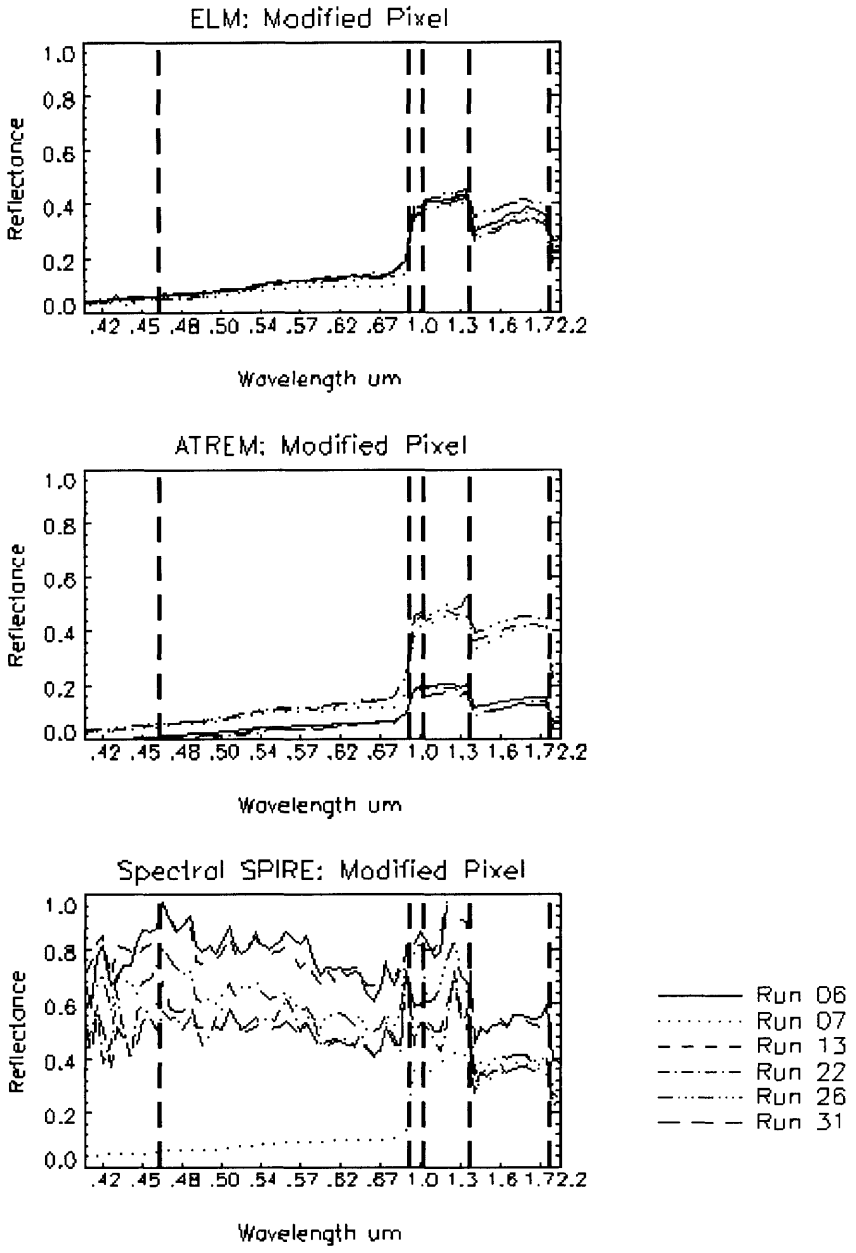


Figure 5.8 (q) and (r). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on emissivity panel 1 (q) and emissivity panel 4 (r).*



(s)

Figure 5.8(s). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six Runs for the modified pixel. The modified pixel was a truck in Run 07, and changed to mowed grass in all subsequent Runs. For Run 07, a similar mowed grass pixel is plotted for comparison.*

These last two effects are caused by the fact that the PC information replaced from the prior reflectance cube PCs tends to drive the Spectral SPIRE solution towards replacing the current image pixel with that of the prior. We are only replacing some PCs, but there is enough energy in these PCs to drive down the variation in the Spectral SPIRE solution to less than that of ELM, and to overcome the problems with low SNR at longer wavelengths. If we were to process all PCs, then the spectral solution would be equal to the prior and have the same variation as the prior across runs. If the prior reflectance significantly deviates from the truth, then this would introduce a bias in the estimates.

Another way to understand why the variance is less than that of ELM, is that for unmodified pixels, we are removing sensor and registration noise (along with the current reflectance) from the low PCs of each image and replacing them with the sensor noise sequence from the prior in those PCs (along with the prior reflectance). Since the same prior is used for all of the images, the variance is reduced.

Figure 5.8(s) shows the reflectance estimates for the modified pixel. We see that Spectral SPIRE performance on this pixel is far worse than ATREM and ELM. The Spectral SPIRE algorithm combines the top PCs from the prior that are derived from a pixel with a truck in it, with lower PCs from the current image with PCs from mowed grass. The result is a very poor estimate of the current reflectance.

The main reason for this poor performance on modified pixels is that changes in log **r**, when rotated into log **m** PC space, overlap the low log **m** PCs that are removed in the PC filtering. This violates our original assumption that no significant amount of the changes in log **r** ends up in the log m PCs. In Spatial SPIRE algorithms, the assumptions of log m noise being band limited to lower spatial frequencies and changes in log r being limited to high spatial

frequencies, allowed us to remove the noise without removing any of log r. However, in Spectral SPIRE, the change in log **r** since the prior is not naturally excluded from the low log **m** PCs, so the low pass PC filtering removes part of the change in log **r** which cannot be restored from the prior. While this is a major problem in using Spectral SPIRE algorithms, we shall see in Chapter 6 that this can be overcome using combined spatial-spectral filtering.

Figure 5.9 demonstrates how much signal energy in the change in log **r** associated with the modified pixel goes into the low PCs with log **m**. In Figure 5.9(a), the log of the ELM estimate of a truck pixel reflectance spectrum from the Run 07 image is plotted rotated into log **m** PC space as defined by Zero-padded Principal Components. In (b) is plotted the same type of spectrum for mowed grass. The difference between (a) and (b) is plotted in (c) and corresponds to the change in log **r** experienced by the modified pixel of Figure 5.8(s). We see in (c) that much of this change falls in the lowest 5 PCs processed by Spectral SPIRE, demonstrating that there is not a clean separation in PC space of log **m** illumination noise and changes in log **r**. In (d) is plotted the mean log **r** of the ELM reflectance estimate for Run 07, showing that much of the zero spatial frequency component of log **r** ends up in the lowest 5 PCs as well and must be restored from the prior log reflectance. Note the similarity between (b) and (d), which is caused by the fact that most of the image is grass and mowed grass, so that the mean closely resembles these materials.

While a study of all possible material changes may show that some changes end up mainly in the higher non-log **m** PCs so that Spectral SPIRE will correctly estimate them, we must conclude that the PC rotations that collect log **m** also collects a significant portion of any

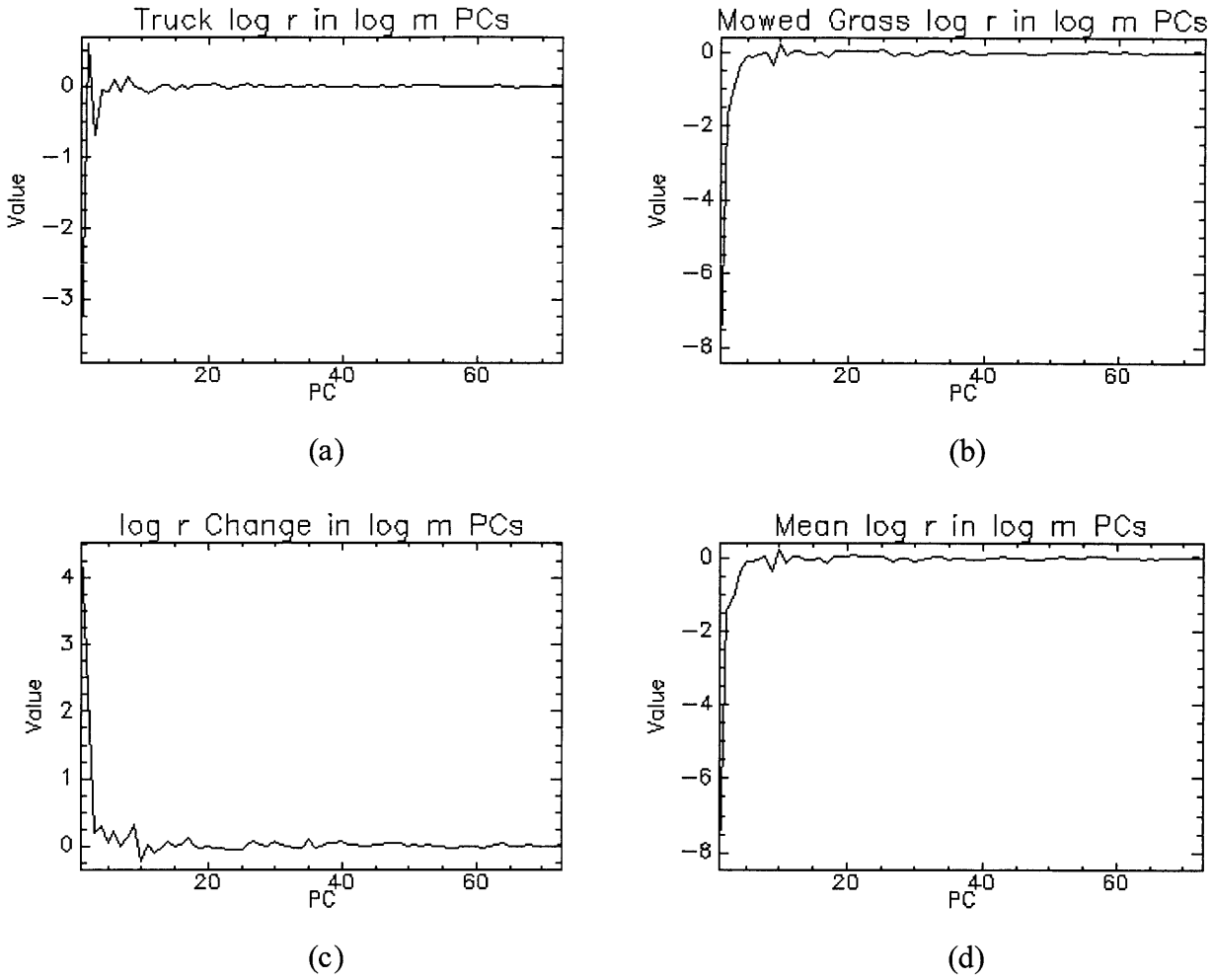


Figure 5.9. *Changes in* log **r** *plotted in* log **m** *PCs. In (a), the log of the ELM estimate of a truck pixel reflectance spectrum from the Run 07 image is plotted rotated into* log **m** *PC space as defined by Zero-padded Principal Components. In (b) is plotted the same type of spectrum for mowed grass. The difference between (a) and (b) is plotted in (c) and corresponds to the change in* log **r** *experienced by the modified pixel of Figure 5.8(s). We see in (c) that much of this change falls in the lowest 5 PCs processed by Spectral SPIRE, demonstrating that there is not a clean separation in PC space of* log **m** *illumination noise and changes in* log **r**. *In (d) is plotted the mean* log **r** *of the ELM reflectance estimate for Run 07, showing that much of the zero spatial frequency component of* log **r** *ends up in the lowest 5 PCs as well and must be restored from the prior log reflectance. Note the similarity between (b) and (d), which is caused by the fact that most of the image is grass and mowed grass, so that the mean closely resembles these materials.*

changes in log **r**. Further research may develop other spectral filtering techniques that better separate these two quantities, but the results of this Chapter indicate that the two quantities inherently overlap in spectral space.

5.5 Computational Cost of Spectral and Spatial SPIRE

While Spectral SPIRE performs worse on modified pixels (changes in reflectance) than Spatial SPIRE, Spectral SPIRE is far more computationally efficient. If we have an (*NxN xnb*) image cube to be processed and only multiplicative noise, then the following computational cost estimates apply:

5.5.1 Spatial SPIRE Computational Cost Estimate

We assume that all spatial filtering is done using Fast Fourier Transforms (FFT) (Oppenheim, et al., 1999) and that the filter kernels are pre-computed. We assume that a complex, length-N FFT requires $6*N*\log_2(N)$ floating point operations (FLOPS) (Oppenheim, et al., 1999). Then each image filtering operation requires one FFT of the image, one product of the NxN image FFT with the NxN kernel FFT, and one inverse FFT of the product. Therefore, each two-dimensional spatial filtering operation requires:

$$(2)(6)(N^2)\log_2(N^2)+N^2 \qquad \text{FLOPS} \tag{5.24}$$

In the Spatial SPIRE Case 3 algorithm, we first move to log space. Empirical experiments indicate that a log or an exponential operation takes approximately 2 FLOPS. Therefore, the following operations are required to process a single channel image using the Spatial Case 3 algorithm:

$$
\begin{aligned}
&rm \rightarrow \log rm: \quad 2N^2 \\
&r_p \rightarrow \log r_p: \quad 2N^2 \\
&\log rm * h_{HPF}: \quad 12N^2 \log_2\left(N^2\right) + N^2 \\
&\log r_p * h_{LPF}: \quad 12N^2 \log_2\left(N^2\right) + N^2 \\
&\log rm * h_{HPF} + \log r_p * h_{LPF}: \quad N^2 \\
&\exp(\log \hat{r}): \quad 2N^2
\end{aligned}
\tag{5.25}
$$

Performing this processing on an image cube with *nb* spectral channels, yields a total cost function of:

$$
\begin{aligned}
\text{Spatial SPIRE Cost} &= \left(9N^2 + 24N^2 \log_2\left(N^2\right)\right) nb \\
&= N^2\left(9 + 24\log_2\left(N^2\right)\right) nb \\
&\simeq 48(nb) N^2 \log_2(N)
\end{aligned}
\tag{5.26}
$$

For an (*N*x*N*x*nb*) = (100x100x100) (*N*=100, *nb*=100) image cube, the Spatial SPIRE cost would be $3.28\text{x}10^8$ FLOPS

5.5.2 Spectral SPIRE Computational Cost Estimate

We assume that the $\mathbf{\Phi}_{\log-\mathbf{rm}}$ rotation matrix is pre-computed and that the rotation of each pixel requires $2(nb)$ FLOPS, so rotating a whole image cube requires $2(nb)N^2$ FLOPS. We assume that replacing a PC requires N^2 FLOPS. Using the same assumptions as for Spatial SPIRE above, the following operations are required:

$$
\begin{aligned}
&\{\mathbf{rm}\} \rightarrow \{\log \mathbf{rm}\}: && 2N^2 \\
&\{\mathbf{r}_p\} \rightarrow \log \mathbf{r}_p: && 2N^2 \\
&\{\Phi_{\log \mathbf{rm}} \log \mathbf{rm}\}: && 2(nb)N^2 \\
&\{\Phi_{\log \mathbf{rm}} \log \mathbf{r}_p\}: && 2(nb)N^2 \\
&\text{Replace } pc_{co-\log \mathbf{m}} \text{ PCs}: && \left(pc_{co-\log \mathbf{m}}\right)N^2 \\
&\{\Phi^{-1}_{\log \mathbf{rm}} \log \hat{\mathbf{r}}\}: && 2(nb)N^2 \\
&\exp(\{\log \hat{r}\}): && 2N^2
\end{aligned}
\tag{5.27}
$$

Yielding a total cost function of:

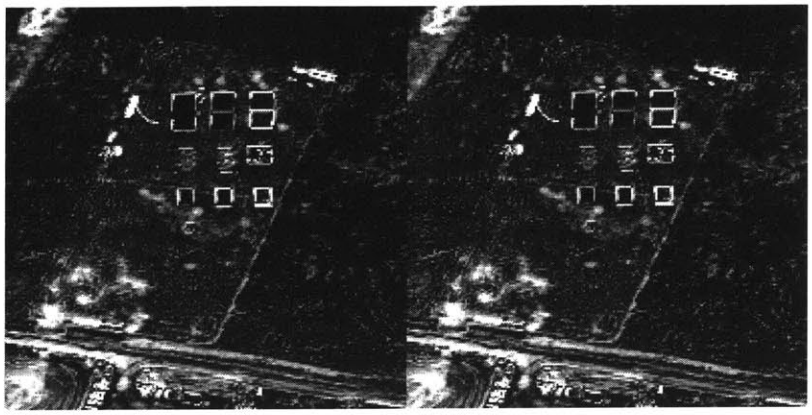
$$
\begin{aligned}
\text{Spectral SPIRE Cost} &= 6N^2 + 6(nb)N^2 + \left(pc_{co-\log \mathbf{m}}\right)N^2 \\
&= N^2\left(6(1+nb) + pc_{co-\log \mathbf{m}}\right) \\
&\simeq 6(nb)N^2
\end{aligned}
\tag{5.28}
$$

For a (100x100x100) image cube with $\mathrm{pc}_{co-\log \mathbf{m}} = 5$, the Spectral SPIRE cost would be: $6.11\text{x}10^6$ FLOPS, which makes Spatial SPIRE over 50 times more computationally expensive than Spectral SPIRE.

Comparing Spatial SPIRE Case 4 algorithm to Spectral SPIRE Case C where both multiplicative and additive noise are present, the disparity is even greater. Assuming that the same number of iteration steps are required for each spatial band or PC to be processed to remove the additive noise, and assuming that 5 a-PCs must be processed versus all channels in Spatial SPIRE, then there is an additional factor of 20 favoring Spectral SPIRE, making the Case 4 algorithm over 1000 times slower than the Case C.

5.6 Noise Analysis

In this section we perform similar noise analyses as was done in Chapter 3 for Spatial



(a) (b)

Figure 5.10. *Distance images of prior reflectance with respect to ELM estimated reflectance (a) for Run 06. The distance value at each pixel is equal to the length of the vector difference between the pixel spectra of the two images at that pixel. Brighter distance image pixels have longer distances and highlight changes since the prior and registration errors in (a). The distance image between Spectral SPIRE and ELM estimated r for the same run is shown in (b), where mis-registration errors do affect the results.*

SPIRE algorithms.

5.6.1 Registration Noise

Imperfect registration of the image and its prior introduces noise that can effect reflectance estimation. Figure 5.10 is analogous to Figure 3.9 and shows distance images that measure the magnitude of the vector difference between the corresponding pixels of two images. For two reflectance images r_1 and r_2, the distance $d[n_{x_o}, n_{y_o}]$ between the pixel spectra $\mathbf{r}_1[n_{x_o}, n_{y_o}, n_\lambda]$ and $\mathbf{r}_2[n_{x_o}, n_{y_o}, n_\lambda]$ is defined as:

$$d[n_{x_o}, n_{y_o}] = \sqrt{\sum_{n_\lambda=1}^{N} \left(\mathbf{r}_1[n_{x_o}, n_{y_o}, n_\lambda] - \mathbf{r}_2[n_{x_o}, n_{y_o}, n_\lambda]\right)^2} \tag{5.29}$$

In Figure 5.10(a), we see the distance image between the prior used for Run 06 and the ELM estimated reflectance for Run 06, just as in Figure 3.9(a). In Figure 5.10(b) is depicted the

distance between the Spectral SPIRE reflectance estimates and ELM estimated reflectance. We note that this distance image is nearly identical to the one in (a). This is because much of the prior reflectance has been incorporated into the Spectral SPIRE reflectance estimate.

For true changes in the image, such as where the truck moves to and from, the large distances are due to Spectral SPIRE's poor performance on modified pixels and not due to registration noise. True registration noise, as seen at the edges of panels, does degrade the Spectral SPIRE reflectance estimates at these locations, more so than they affect Spatial SPIRE. The remaining differences are due to a variety of other noise sources which ELM does not handle perfectly (such as sensor noise and BRDF issues) and makes the ELM reflectance estimates deviate from true ground truth.

The actual effect of misregistration noise in Spectral SPIRE results will depend on several factors. Such noise will only affect pixels at the edges of uniform materials, or pixels over mottled surfaces, unless the misregistration errors are many pixel widths in magnitude. The magnitude of the reflectance error introduced will be dependent on the different materials that are shifted into and out of the pixel in question. This makes it unfeasible to develop a single equation that predicts the Spectral SPIRE estimated reflectance errors based on a given misregistration amount. However, registration can typically be done to within a fraction of a pixel, which will limit errors due to misregistration to edge pixels. Using sensors with adequate spatial resolution to resolve objects of interest so that they have at least one spectrally pure pixels between edge pixels will allow classification and spectral analysis to be done on the pure pixels free of any effects from misregistration.

5.6.2 Prior, Sensor, and Calibration Noise

We shall now derive the effects of sensor, calibration, and prior noise sources on the

Spectral SPIRE Case D algorithm and give some examples of their effects. Since we are dealing with Spectal SPIRE, we will concentrate on the effects at a single pixel. Again, we have restricted ourselves to dealing with additive noise that is the same for all pixels in the image. We assume that there are three main types of noise: sensor, calibration, and prior.

In spectral processing we use spectral Principal Component Analysis to define a rotation that collects all of $\mathbf{a}$ or $\log \mathbf{m}$ in a few low PCs so that only those PCs need to be processed. Such rotations will not tend to collect sensor noise into the low PCs, since it will tend to be uncorrelated with $\mathbf{a}$ or $\log \mathbf{m}$. However, it also will not tend to reject it from these PCs either. Since only the low $\log \mathbf{m}$ PCs have information from the prior restored to them, prior noise is only introduced to these PCs. This noise will then be spread out into the spectral channels when the inverse PC rotation is applied.

If the ensembles of $\mathbf{a}$ and $\log \mathbf{m}$ used to define rotations contain representative calibration noise, then calibration noise will be collected along with $\mathbf{a}$ and $\log \mathbf{m}$ into the top PCs that will be spatially filtered. Therefore, it is important to use $\mathbf{a}$ and $\log \mathbf{m}$ ensembles from empirical data that have been through the calibration process so that they contain representative calibration noise. If this is done, then multiplicative and additive calibration noise will be collected into the low PCs along with $\mathbf{a}$ and $\log \mathbf{m}$, and will then be removed by the Spectral SPIRE algorithm. With these comments regarding calibration noise in mind, we now derive expressions for reflectance estimation errors caused by noise, similar to those derived in Chapter 3 for Spatial SPIRE.

A noise free, calibrated radiance image $\mathbf{i}$ is given by:

$$\mathbf{i} = \mathbf{r} \odot \mathbf{m} + \mathbf{a} \tag{5.30}$$

Where **r** is the surface reflectance and **m** and **a** are the multiplicative and additive illumination noise effects. We assume that there is an additive sensor noise vector **s** which is typically the sum of several noise sources. We will assume that **s** is a random Gaussian vector with mean vector $\boldsymbol{\mu}_s$ and covariance $\boldsymbol{\Lambda}_s$: $\mathbf{s} \sim N(\boldsymbol{\mu}_s, \boldsymbol{\Lambda}_s)$. Calibration noise is assumed to consist of an additive offset noise vector, or error, $\mathbf{c}_{offset}$ and a multiplicative gain noise vector $\mathbf{c}_{gain}$. As in Chapter 3, we will assume that $\mathbf{c}_{offset}$ and $\mathbf{c}_{gain}$ are uniform across the image and therefore the same for all pixels. With calibration and sensor noise sources taken into account, our single calibrated radiance pixel can be viewed as:

$$\mathbf{i} = \mathbf{c}_{gain} \odot (\mathbf{r} \odot \mathbf{m} + \mathbf{a} + \mathbf{s}) + \mathbf{c}_{offset} \tag{5.31}$$

The removal of **a** is accomplished in the Case D algorithm by first rotating **i** into **a**-PC space using the PC rotation matrix $\boldsymbol{\Phi}_\mathbf{a}$:

$$\boldsymbol{\Phi}_\mathbf{a}\mathbf{i} = \boldsymbol{\Phi}_\mathbf{a}\left(\mathbf{c}_{gain} \odot (\mathbf{r} \odot \mathbf{m} + \mathbf{a} + \mathbf{s})\right) + \boldsymbol{\Phi}_\mathbf{a}\mathbf{c}_{offset} \tag{5.32}$$

To remove the additive noise **a** we subtract the mean of the pixel ensemble $\{\boldsymbol{\Phi}_\mathbf{a}\mathbf{i}\}$ from itself in the lowest $pc_{co-\mathbf{a}}$ PCs. If we assume that all of **a** was collected into the low **a**-PCs processed, then using the notation defined in (5.13) we have:

$$\begin{aligned} \boldsymbol{\Phi}_\mathbf{a}\mathbf{i} - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{(\boldsymbol{\Phi}_\mathbf{a}\mathbf{i})_k\right\} &= \boldsymbol{\Phi}_\mathbf{a}\left(\mathbf{c}_{gain} \odot (\mathbf{r} \odot \mathbf{m} + \mathbf{a} + \mathbf{s})\right) + \boldsymbol{\Phi}_\mathbf{a}\mathbf{c}_{offset} \\ &\quad - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\boldsymbol{\Phi}_\mathbf{a}\left(\mathbf{c}_{gain} \odot (\mathbf{r} \odot \mathbf{m} + \mathbf{a} + \mathbf{s})\right)\right)_k\right\} \\ &\quad - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\boldsymbol{\Phi}_\mathbf{a}\mathbf{c}_{offset}\right)_k\right\} \end{aligned} \tag{5.33}$$

If empirical ensembles were used as described above to determine $\mathbf{\Phi}_{\mathbf{a}}$, then we can assume that all of the constant $\mathbf{c}_{offset}$ will be collected into the lowest $pc_{co-\mathbf{a}}$ PCs along with **a**, so that:

$$\mathbf{\Phi}_{\mathbf{a}}\mathbf{c}_{offset} = \sum_{k=1}^{pc_{co-\mathbf{a}}} \left(\mathbf{\Phi}_{\mathbf{a}}\mathbf{c}_{offset}\right)_k = \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\mathbf{c}_{offset}\right)_k\right\} \tag{5.34}$$

Giving us:

$$\begin{aligned}\mathbf{\Phi}_{\mathbf{a}}\mathbf{i} - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\mathbf{i}\right)_k\right\} &= \mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \left(\mathbf{r} \odot \mathbf{m} + \mathbf{a} + \mathbf{s}\right)\right) \\ &\quad - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \left(\mathbf{r} \odot \mathbf{m} + \mathbf{a} + \mathbf{s}\right)\right)\right)_k\right\}\end{aligned} \tag{5.35}$$

Which can be expanded to:

$$\begin{aligned}\mathbf{\Phi}_{\mathbf{a}}\mathbf{i} - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\mathbf{i}\right)_k\right\} &= \mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \mathbf{r} \odot \mathbf{m}\right) + \mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \mathbf{a}\right) + \mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \mathbf{s}\right) \\ &\quad - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \mathbf{r} \odot \mathbf{m}\right)\right)_k\right\} \\ &\quad - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \mathbf{a}\right)\right)_k\right\} \\ &\quad - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \mathbf{s}\right)\right)_k\right\}\end{aligned} \tag{5.36}$$

Using the same argument that all of **a** is collected into the **a**-PCs, this can be reduced to:

$$\begin{aligned}\mathbf{\Phi}_{\mathbf{a}}\mathbf{i} - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\mathbf{i}\right)_k\right\} &= \mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \mathbf{r} \odot \mathbf{m}\right) - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \mathbf{r} \odot \mathbf{m}\right)\right)_k\right\} \\ &\quad + \mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \mathbf{s}\right) - \sum_{k=1}^{pc_{co-\mathbf{a}}} \text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain} \odot \mathbf{s}\right)\right)_k\right\}\end{aligned} \tag{5.37}$$

Note that if the sensor noise $\mathbf{s}$ is zero mean, then the last term in (5.37) equals zero. Using the iterative Case D algorithm, we estimate the lost means of $\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\mathbf{r}\odot\mathbf{m}\right)\right)_k\right\}$ and add them back in, undo the PC rotation, and estimate $\mathbf{r}\odot\mathbf{m}$:

$$\hat{\mathbf{r}\odot\mathbf{m}}=\mathbf{\Phi}_{\mathbf{a}}^{-1}\left(\mathbf{\Phi}_{\mathbf{a}}\mathbf{i}-\sum_{k=1}^{pc_{co-\mathbf{a}}}\text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\mathbf{i}\right)_k\right\}+\sum_{k=1}^{pc_{co-\mathbf{a}}}\text{est}\left(\text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\mathbf{r}\odot\mathbf{m}\right)\right)_k\right\}\right)\right) \quad (5.38)$$

Substituting in (5.37) yields:

$$\hat{\mathbf{r}\odot\mathbf{m}}=\mathbf{\Phi}_{\mathbf{a}}^{-1}\left(\begin{array}{c}\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\left(\mathbf{r}\odot\mathbf{m}\right)\right)-\sum_{k=1}^{pc_{co-\mathbf{a}}}\text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\left(\mathbf{r}\odot\mathbf{m}\right)\right)\right)_k\right\}\\+\sum_{k=1}^{pc_{co-\mathbf{a}}}\text{est}\left(\text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\left(\mathbf{r}\odot\mathbf{m}\right)\right)\right)_k\right\}\right)\\+\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\mathbf{s}\right)-\sum_{k=1}^{pc_{co-\mathbf{a}}}\text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\mathbf{s}\right)\right)_k\right\}\end{array}\right) \quad (5.39)$$

Let us define the modified sensor noise represented by the last two terms in (5.38) as:

$$\mathbf{s}_{\mathbf{mod}}=\mathbf{\Phi}_{\mathbf{a}}^{-1}\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\mathbf{s}\right)-\sum_{k=1}^{pc_{co-\mathbf{a}}}\text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\mathbf{s}\right)\right)_k\right\}\right) \quad (5.40)$$

Equation (5.40) only modifies the mean vector of the sensor noise, so its covariance is unaffected. If the mean vector $\boldsymbol{\mu}_{\mathbf{s}}$ of the original sensor noise $\mathbf{s}$ equals the zero vector, then the operation defined by (5.40) has no affect on $\mathbf{s}$, then $\mathbf{s}_{\mathbf{mod}}=\mathbf{s}$.

Assuming that we have found the best value of the lost means of $\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\left(\mathbf{r}\odot\mathbf{m}\right)\right)\right)_k$ to add back in, we proceed to take the log and use the Case C algorithm to solve the multiplicative-noise-only case. If there are errors in our estimates of the lost means of

$\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\left(\mathbf{r}\odot\mathbf{m}\right)\right)\right)_k\right\}$, then this adds an error vector $\mathbf{e}_{\mathbf{a}}$ defined as:

$$\mathbf{e}_{\mathbf{a}}=\mathbf{\Phi}_{\mathbf{a}}^{-1}\left(\begin{array}{l}\sum_{k=1}^{pc_{co-\mathbf{a}}}\text{est}\left(\text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\left(\mathbf{r}\odot\mathbf{m}\right)\right)\right)_k\right\}\right)\\ -\sum_{k=1}^{pc_{co-\mathbf{a}}}\text{mean}\left\{\left(\mathbf{\Phi}_{\mathbf{a}}\left(\mathbf{c}_{gain}\odot\left(\mathbf{r}\odot\mathbf{m}\right)\right)\right)_k\right\}\end{array}\right) \tag{5.41}$$

Substituting (5.41) and (5.40) into (5.39) gives us the following estimate of $\mathbf{r}\odot\mathbf{m}$:

$$\hat{\mathbf{r}\odot\mathbf{m}}=\mathbf{c}_{gain}\odot\left(\mathbf{r}\odot\mathbf{m}\right)+\mathbf{e}_{\mathbf{a}}+\mathbf{s}_{\mathbf{mod}} \tag{5.42}$$

Next we move into log space in an attempt to separate $\mathbf{r}$ and $\mathbf{m}$:

$$\log\hat{\mathbf{r}\odot\mathbf{m}}=\log\left(\mathbf{c}_{gain}\odot\mathbf{r}\odot\mathbf{m}+\mathbf{e}_{\mathbf{a}}+\mathbf{s}_{\mathbf{mod}}\right)\approx\log\mathbf{r}+\log\mathbf{m} \tag{5.43}$$

If $\mathbf{s}_{\mathbf{mod}}$ and $\mathbf{e}_{\mathbf{a}}$ are both small and can be ignored, then $\mathbf{c}_{gain}$ causes a constant offset effect in log space:

$$\log\hat{\mathbf{r}\odot\mathbf{m}}\cong\log\left(\mathbf{c}_{gain}\odot\mathbf{r}\odot\mathbf{m}\right)=\log\mathbf{c}_{gain}+\log\mathbf{r}+\log\mathbf{m} \tag{5.44}$$

If $\mathbf{s}_{\mathbf{mod}}$ and $\mathbf{e}_{\mathbf{a}}$ cannot be ignored, then the addition of $\mathbf{s}_{\mathbf{mod}}$ and $\mathbf{e}_{\mathbf{a}}$ to $\mathbf{c}_{gain}\odot\mathbf{r}\odot\mathbf{m}$ causes non-linear effects on the pixel vector in log space that can be treated as an added noise $\log\mathbf{e}_{\mathbf{sanl}}$:

$$\begin{aligned}\log\mathbf{e}_{\mathbf{sanl}}&=\log\left(\mathbf{c}_{gain}\odot\mathbf{r}\odot\mathbf{m}+\mathbf{e}_{\mathbf{a}}+\mathbf{s}_{\mathbf{mod}}\right)-\log\left(\mathbf{c}_{gain}\odot\mathbf{r}\odot\mathbf{m}\right)\\&=\log\left(\mathbf{c}_{gain}\odot\mathbf{r}\odot\mathbf{m}+\mathbf{e}_{\mathbf{a}}+\mathbf{s}_{\mathbf{mod}}\right)-\log\mathbf{c}_{gain}+\log\mathbf{r}+\log\mathbf{m}\end{aligned} \tag{5.45}$$

Substituting (5.44) and (5.45) into(5.43) yields:

$$\log\left(\mathbf{r}\hat{\odot}\mathbf{m}\right) = \log\mathbf{r} + \log\mathbf{m} + \log\mathbf{c}_{gain} + \log\mathbf{e}_{\text{sanl}} \tag{5.46}$$

We now apply the rest of the Case C algorithm to (5.46). We first low-pass PC filter the estimate of $\log\left(\mathbf{r}\hat{\odot}\mathbf{m}\right)$ to remove $\log\mathbf{m}$. If empirical ensembles were used to determine $\mathbf{\Phi}_{\log\mathbf{m}}$, then we can assume that all of the constant $\log\mathbf{c}_{gain}$ will be collected into the lowest $\text{pc}_{co-\log\mathbf{m}}$ PCs along with $\log\mathbf{m}$ so that the $\text{PCF}_{\text{L-log}\mathbf{m}}$ filtering operation will also remove the $\log\mathbf{c}_{gain}$ term. The result of this filtering operation is then:

$$\text{PCF}_{\text{H}-\log\mathbf{m}}\left(\log\left(\mathbf{r}\hat{\odot}\mathbf{m}\right)\right) = \log\mathbf{r} - \Delta\log\mathbf{r} + \log\mathbf{e}_{\text{sanl-HPC}} \tag{5.47}$$

Where we have defined a new noise $\log\mathbf{e}_{\text{sanl-HPC}}$ which is the high-pass PC filtered version of $\log\mathbf{e}_{\text{sanl}}$:

$$\log\mathbf{e}_{\text{sanl-HPC}} = \text{PCF}_{\text{H}-\log\mathbf{m}}\left(\log\mathbf{e}_{\text{sanl}}\right) \tag{5.48}$$

The nest step is to low-pass PC filter the rotated log prior reflectance to estimate and restore $\Delta\log\mathbf{r}$. We assume that the prior has a random Gaussian noise vector $\mathbf{e}_{\text{p}}$ with mean vector $\boldsymbol{\mu}_{\text{p}}$ and covariance $\boldsymbol{\Lambda}_{\text{p}}$: $\mathbf{e}_{\text{p}} \sim N(\boldsymbol{\mu}_{\text{p}}, \boldsymbol{\Lambda}_{\text{p}})$. This noise vector is also low pass PC filtered:

$$\hat{\Delta\log\mathbf{r}} = \text{PCF}_{\text{L}-\log\mathbf{m}}\left(\log\mathbf{r}_{p} + \log\mathbf{e}_{\text{pnl}}\right) \tag{5.49}$$

Where the log operation forces us to define a second non-linear noise term:

$$\log \mathbf{e}_{\text{pnl}} = \log\left(\mathbf{r}_p + \mathbf{e}_{\text{p}}\right) - \log \mathbf{r}_p \tag{5.50}$$

So that:

$$\Delta \hat{\log} \mathbf{r} = \text{PCF}_{\text{L}-\log \text{m}}\left(\log \mathbf{r}_p + \log \mathbf{e}_{\text{pnl}}\right) = \Delta \log \mathbf{r}_p + \Delta \log \mathbf{e}_{\text{pnl}} \tag{5.51}$$

Adding (5.51) to (5.47) to estimate $\log \mathbf{r}$, and assuming that $\Delta \log \mathbf{r}_p = \Delta \log \mathbf{r}$ we obtain:

$$\hat{\log} \mathbf{r} = \log \mathbf{r} + \log \mathbf{e}_{\text{sanl-HPC}} + \Delta \log \mathbf{e}_{\text{pnl}} \tag{5.52}$$

Utilizing the same notation as we did for $\log \mathbf{e}_{\text{sanl-HPC}}$, we can write that $\log \mathbf{e}_{\text{pnl-LPC}} = \Delta \log \mathbf{e}_{\text{pnl}}$ and (5.52) becomes:

$$\hat{\log} \mathbf{r} = \log \mathbf{r} + \log \mathbf{e}_{\text{sanl-HPC}} + \log \mathbf{e}_{\text{pnl-LPC}} \tag{5.53}$$

Taking the exponential yields:

$$\begin{aligned} \hat{\mathbf{r}} &= \exp\left(\log \mathbf{r} + \log \mathbf{e}_{\text{sanl-HPC}} + \log \mathbf{e}_{\text{pnl-LPC}}\right) \\ &= \mathbf{r} \odot \mathbf{e}_{\text{sanl-HPC}} \odot \mathbf{e}_{\text{pnl-LPC}} \end{aligned} \tag{5.54}$$

Which is directly analogous to Equation (3.47). We can define an overall error vector as follows:

$$\mathbf{e}_{\hat{\mathbf{r}}} = \hat{\mathbf{r}} - \mathbf{r} = \mathbf{r} \odot \mathbf{e}_{\text{sanl-HPC}} \odot \mathbf{e}_{\text{pnl-LPC}} - \mathbf{r} = \mathbf{r} \odot \left(\mathbf{e}_{\text{sanl-HPC}} \odot \mathbf{e}_{\text{pnl-LPC}} - 1\right) \tag{5.55}$$

Where the two noise terms are defined as:

$$\begin{aligned}\mathbf{e}_{\text{sanl-HPC}} &= \exp\left(\text{PCF}_{\text{H}-\log\mathbf{m}}\left(\log\mathbf{e}_{\text{sanl}}\right)\right) \\ &= \exp\left(\text{PCF}_{\text{H}-\log\mathbf{m}}\left(\log\left(\mathbf{c}_{gain}\odot\mathbf{r}\odot\mathbf{m}+\mathbf{e}_{\mathbf{a}}+\mathbf{s}_{\mathbf{mod}}\right)-\log\mathbf{c}_{gain}+\log\mathbf{r}+\log\mathbf{m}\right)\right)\end{aligned} \tag{5.56}$$

$$\begin{aligned}\mathbf{e}_{\text{pnl-LPC}} &= \exp\left(\text{PCF}_{\text{L}-\log\mathbf{m}}\left(\log\mathbf{r}_p+\log\mathbf{e}_{\text{pnl}}\right)-\Delta\log\mathbf{r}_p\right) \\ &= \exp\left(\text{PCF}_{\text{L}-\log\mathbf{m}}\left(\log\mathbf{r}_p+\log\mathbf{e}_{\text{pnl}}\right)-\text{PCF}_{\text{L}-\log\mathbf{m}}\left(\log\mathbf{r}_p\right)\right)\end{aligned} \tag{5.57}$$

We see that the use of the prior to replace the reflectance information in the low log $\mathbf{m}$ PCs also replaces the $\mathbf{\Phi}_{\log\mathbf{m}}\log\mathbf{e}_{\text{sanl}}$ in those PCs with the $\mathbf{\Phi}_{\log\mathbf{m}}\log\mathbf{e}_{\text{pnl}}$ in those PCs. We also note that $\mathbf{e}_{\text{sanl-HPC}}$ is independent of calibration noise $\mathbf{c}_{gain}$ even though it appears in (5.56), since the high pass PC filtering operation removes $\mathbf{c}_{gain}$.

Both $\mathbf{e}_{\text{sanl-HPC}}$ and $\mathbf{e}_{\text{pnl-LPC}}$ cause no error in $\hat{\mathbf{r}}$ if they are equal to 1 (0 in log space). Figure 5.11 depicts $\mathbf{e}_{\hat{\mathbf{r}}}$ as a function of $SNR = \mathbf{rm}/\boldsymbol{\sigma}_{\mathbf{s}}$ at a single pixel in the first spectral channel (~0.41 μm) when $\mathbf{e}_{\text{pnl-LPC}} = 1$, so that $\mathbf{e}_{\hat{\mathbf{r}}}$ is caused purely by sensor noise s via $\mathbf{e}_{\text{sanl-HPC}}$. We have assumed that $\mathbf{a} = 0$ and that $\mathbf{r}[1] = 0.20$ and $\mathbf{m}[n_\lambda] = 600$ Watts/m^2/sr/μm for all pixels in the image. The two curves represent the effect on a single pixel for $s[1] = +\sigma_s$ and $s[1] = -\sigma_s$ since these will have different nonlinear effects on $\mathbf{e}_{\hat{\mathbf{r}}}$. We see that errors cause by sensor noise can be overcome by increasing SNR. For the Run 06, SNR (see Figure 3.15) is above 65 for most channels in Bands 1 and 2, above 30 in Bands 3 and 4, and above 5 in Bands 5 and 6, resulting in absolute reflectance errors ranging between 0.005 in Bands 1 & 2 and 0.04 in bands 5 and 6.

Figure 5.12 plots the effect of the prior noise $\mathbf{e}_{\text{pnl-LPC}}$ on $\mathbf{e}_{\hat{\mathbf{r}}}$ when $\mathbf{e}_{\text{sanl-HPC}} = 1$. Again, $\mathbf{r}[1] = 0.20$ and $\mathbf{m}[n_\lambda] = 600$ Watts/m^2/sr/μm for all pixels in the image. In Figure 5.12 we assume that $\boldsymbol{\mu}_{\mathbf{p}} = 0$ so that $\mathbf{e}_{\mathbf{p}} \sim N(0, \boldsymbol{\Lambda}_{\mathbf{p}})$. Since $\mathbf{r}_{\mathbf{p}} = \mathbf{r} + \mathbf{e}_{\mathbf{p}}$ would have been forced to have

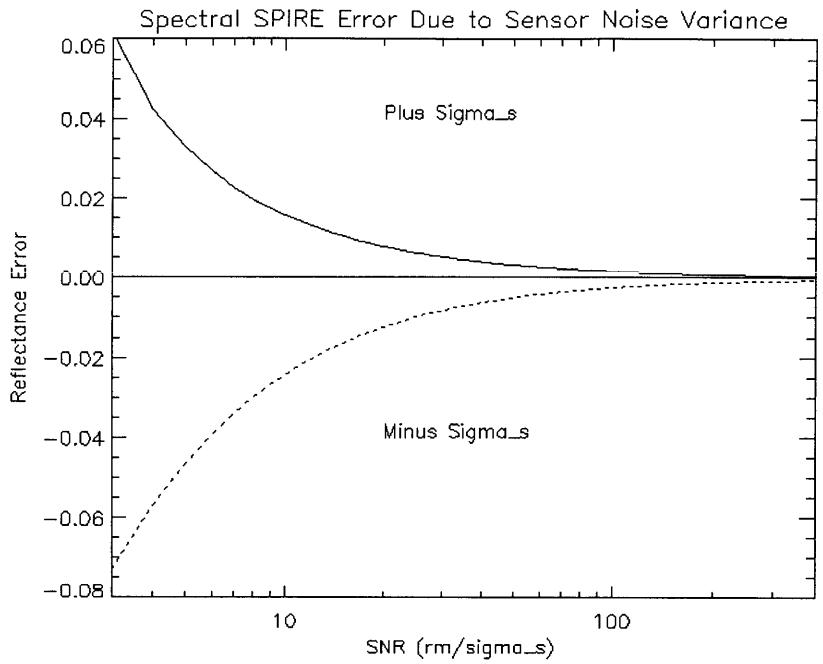


Figure 5.11. *Spectral SPIRE reflectance error at a single pixel of reflectance r=0.20 caused by sensor noise standard deviation σ_s (sigma_s) versus signal-to-noise ratio (SNR=rm/σ_s) in the multiplicative noise only case (**a**=0). The effect of the sensor noise is non-linear and plots for the noise s=+σ_s and s=-σ_s are shown. A value of m=600 Watts/m^2/sr/μm was used at all pixels. Effects of sensor noise can be overcome by increasing SNR.*

values between 0 and 1, our range of standard deviations considered is $\sigma_p = 0$ to 0.008. For the priors used in this thesis, $\sigma_p \simeq 0.005$, causing very small reflectance errors. Since the priors used in the algorithm testing in this thesis are derived from ELM estimates that we can consider ground truth, the prior noise means are all effectively zero and do not cause any reflectance error.

5.6.3 Perfect versus Normal Priors

To gain insight into the overall effects of these noise sources on the results of Figure 5.8, the Spectral SPIRE processing of Section 5.4 was redone using a "perfect prior" consisting of the ELM reflectance estimate for each test cube. In addition, the ELM-estimated additive **a** vectors were subtracted to create "**m**-only" test cubes which were also processed with perfect

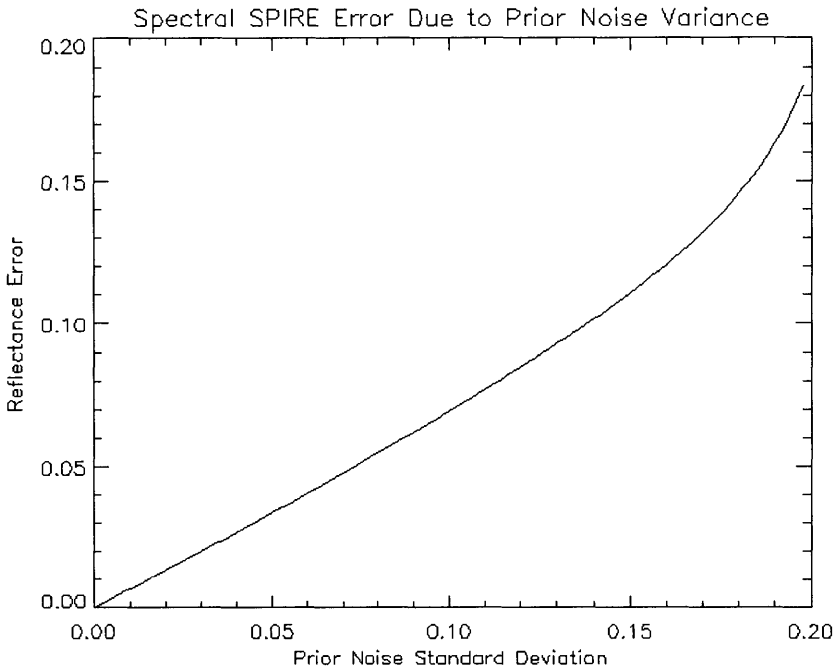


Figure 5.12. *Reflectance error at a single pixel of reflectance r=0.50 caused by prior noise variance σ_p in the multiplicative noise only case (***a***=0). A value of m=600 Watts/m^2/sr/μm was used at all pixels and the prior noise was assumed to be Gaussian with zero mean.*

priors and those generated from Run 07. Figure 5.13(a) shows these four results for the 2 percent panel while 5.13(b) shows these results for the spectral panel. Here we see that the prior noise on these unmodified pixels does not affect the Spectral SPIRE results.

5.7 Summary and Conclusions

In this chapter we developed Spectral SPIRE algorithms that estimate surface reflectance using a prior reflectance image, prior ensembles of **m** and **a**, and PC filtering to remove the multiplicative and additive noise effects. We developed them for the multiplicative-noise-only case, and for the case where a constant additive noise term is present. We compared the performance of the Spectral SPIRE algorithms to ELM and ATREM on six HYDICE hyperspectral image cubes from the ARM Site data collect.

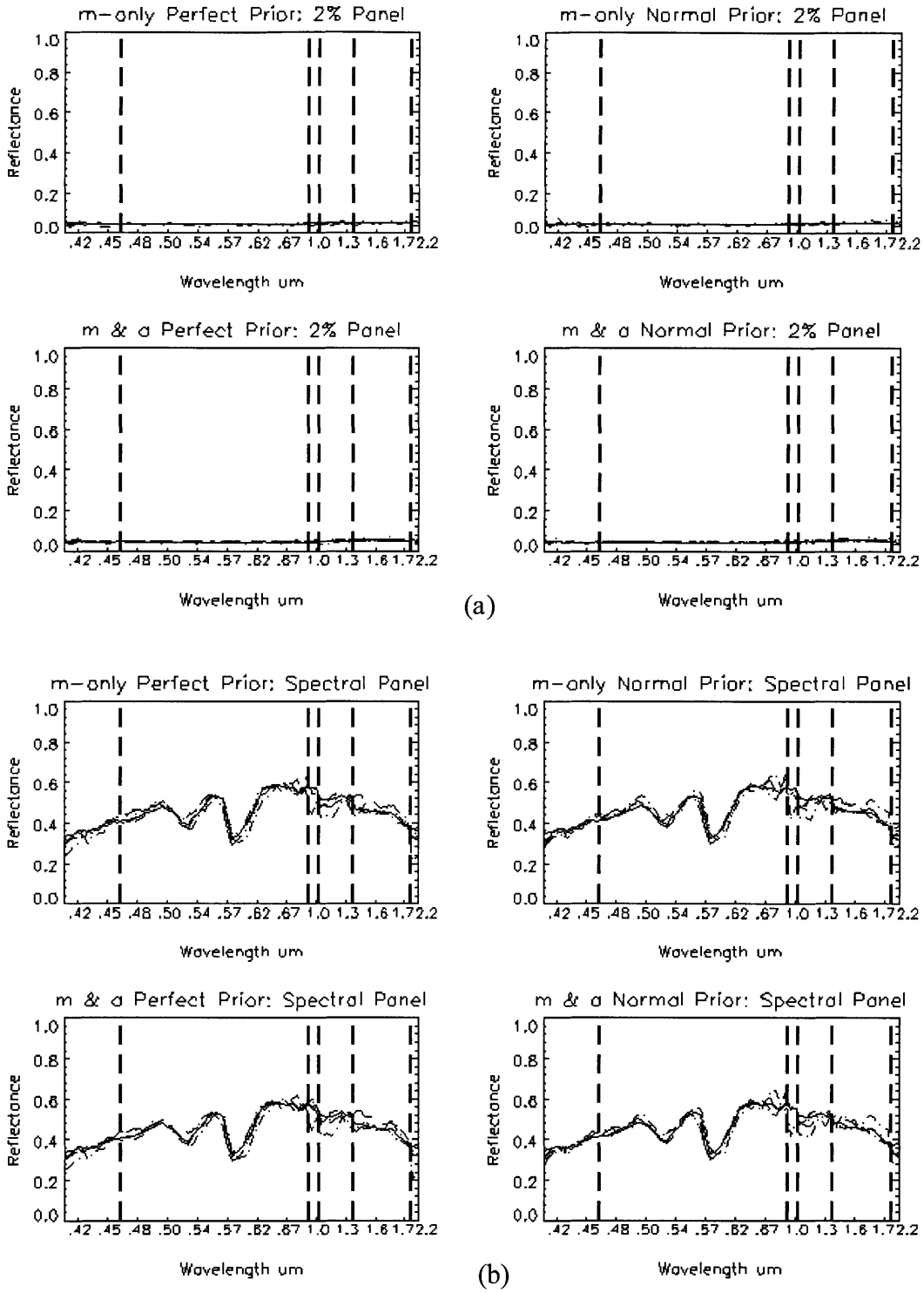


Figure 5.13. *Spectral SPIRE Case C (m-only) and Case D (m & a) reflectance estimates for all six runs using a perfect prior and a normal (noisy) prior, for the 2 percent panel (a) and the spectral panel (b).*

From these experiments we can draw the following conclusions:

- Performance of Spectral SPIRE was very similar to that of ELM on unmodified pixels.
- Performance of Spectral SPIRE was consistently far better than ATREM on unmodified pixels.
- Performance of Spectral SPIRE was better than Spatial SPIRE on unmodified pixels.
- Spectral SPIRE performance does not suffer on unmodified low reflectivity materials during low SNR conditions like Spatial SPIRE does.
- Spectral SPIRE algorithms are computationally less costly than Spatial SPIRE.
- Like Spatial SPIRE, Spectral SPIRE algorithms are insensitive to calibration noise if the ensembles of **m** and **a** used are derived empirically from real sensor data.
- The effects of sensor and prior noise can be overcome with adequate SNR.
- However, Spectral SPIRE performed far worse than either Spatial SPIRE, ELM, or ATREM on modified pixels.
- Also, Spectral SPIRE algorithms are sensitive to prior misregistration noise.

We now move on to Combined SPIRE algorithms that utilize combined spatial-spectral processing. Our challenge is to combine the Spatial SPIRE algorithms of Chapter 3 with the Spectral SPIRE algorithms of this chapter, to take advantage of 1) Spectral SPIRE's low computational cost and superior performance on unmodified pixels and 2) Spatial SPIRE's superior performance on modified pixels.

Chapter 6

Combined SPIRE Algorithms

6.1 Introduction

In this chapter, we develop SPIRE reflectance estimation algorithms that make use of both spatial and spectral filtering. The goal is to merge the superior performance of Spatial SPIRE algorithms for estimating reflectance of modified pixels, with the superior performance on unmodified pixels and computational efficiency of Spectral SPIRE algorithms, to develop an algorithm with the positive attributes of each.

6.2 Combining Spatial and Spectral Principal Component Filtering

There are many ways to combine the spatial and spectral filtering used in Chapters 3 and 5. Given our goal of combining Spatial SPIRE's ability to estimate the reflectance of modified pixels and Spectral SPIRE's performance and computational efficiency, one approach stands out. Using the Spectral SPIRE technique of collecting $\log \mathbf{m}$ into a few low PCs, we can spatially filter only those PCs, dramatically reducing the number of channels to spatially filter. This should also eliminate the problem Spectral SPIRE has with estimating the reflectance of modified pixels, since we would again be replacing only low spatial frequency information from the prior reflectance.

To perform this combined filtering, we define the concept of a combined spectral-PC, spatial-frequency filter (CF). Given that we want first to collect the illumination noise into a few PCs, we will restrict our attention to those combined filters that perform the PC rotation first. Although only certain variations are of interest here, the general operation of such a CF is to:

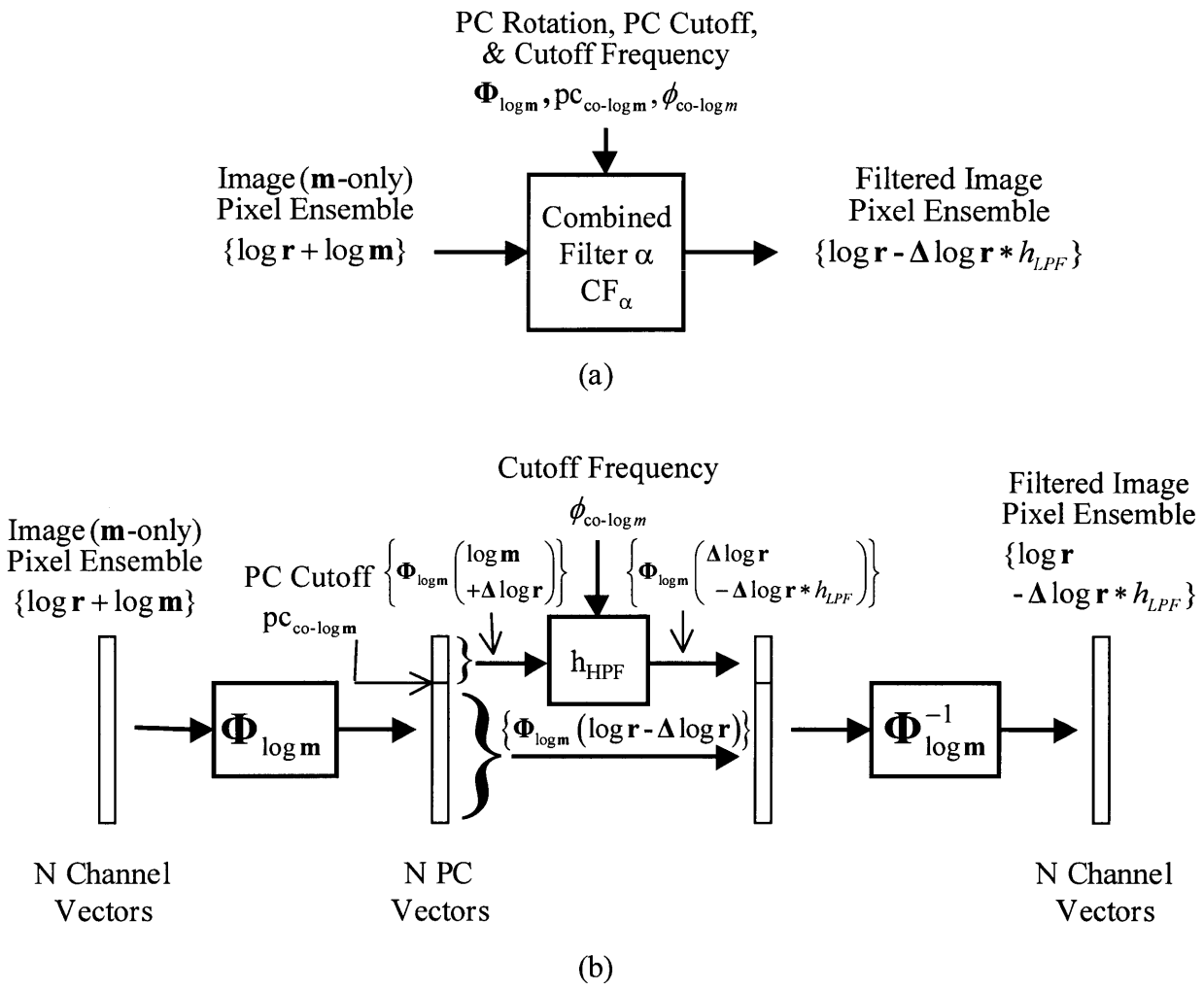


Figure 6.1. *Combined filter type* α (CF_α)*: high PC pass, low PC spatial high-pass filter. At the top (a) is the input and output definition, while (b) is the detailed processing block diagram. The net effect of this filter is to collect the* log **m** *noise into a few low PCs, an then remove it by spatially filtering only those PCs. This filter is similar to the high-pass PC Filter* PCF_H *of Figure 5.1, except that the low PCs are spatially high-pass filtered instead of being zeroed.*

- Separate an ensemble of vectors ({**i**} or $\{\log \mathbf{rm}\}$) in PC space (**a** or $\log \mathbf{m}$) into high and low PCs
- Reject (multiply by zero) or pass either the low or high PCs
- Perform spatial high or low pass filtering on the PCs that were not rejected

There are many possible permutations of the above three operations, and the specific options selected will depend on the objective of the filtering operation. Two permutations will be useful to our problem. The first type of combined filter will be referred to as CF_α, and is defined in Figure 6.1. The CF_α filter can be described as a high PC pass, low PC spatial low-pass filter. Figure 6.1(a) depicts the input and output definition, while Figure 6.1(b) depicts the detailed processing block diagram. Essentially, this filter is similar to the high-pass PCF of Figure 5.1, but instead of completely rejecting the low PCs, they are spatially high-pass filtered to reject just the low spatial frequencies containing illumination noise in these PCs. Note that unlike Spectral SPIRE where each pixel can be processed independently, the entire ensemble must be PC filtered before the spatial filtering can be applied, hence the use of curly brackets to denote ensembles.

The second type of combined filter will be referred to as CF_β, and is defined in Figure 6.2. The CF_β filter can be described as a high PC reject, low PC spatial low-pass filter. Figure 6.2(a) depicts the input and output definition, while Figure 6.2(b) depicts the detailed processing block diagram. Essentially, this filter is similar to the low-pass PCF of Figure 5.2, but the low PCs are spatially low-pass filtered. Also, this filter is applied to the prior reflectance ensemble, while the filter of Figure 5.2 is applied to a log **rm** pixel.

The subscripted $_\alpha$ and $_\beta$ have no significance other than to distinguish between the two types of combined filters employed in this thesis, and were chosen so that there designations would not be confused with the various SPIRE Case identifiers.

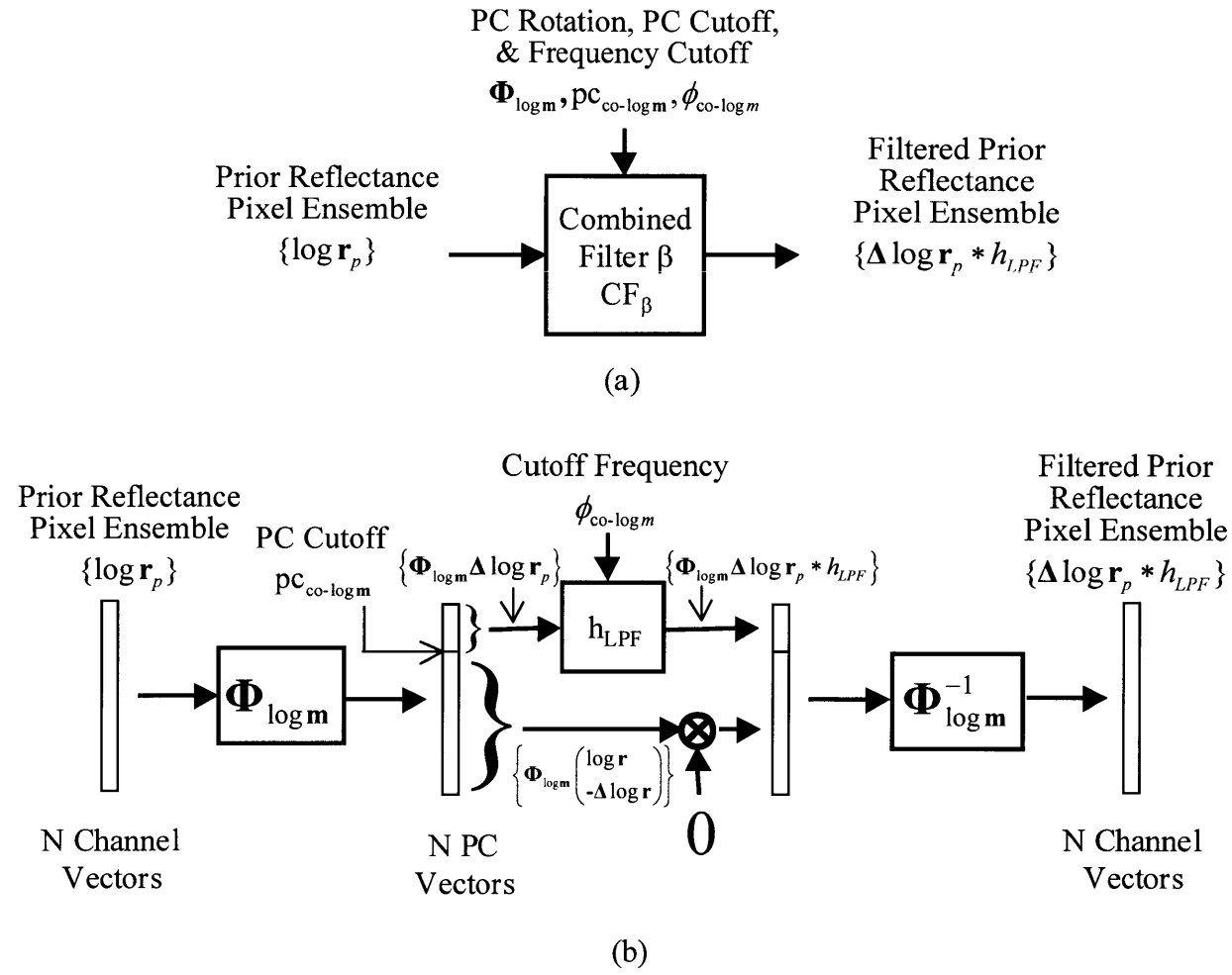


Figure 6.2. *Combined filter type* β (CF_β)*: high PC reject, low PC spatial low-pass filter. At the top (a) is the input and output definition, while (b) is the detailed processing block diagram. The net effect of this filter as shown is to pass only the high spatial frequencies of the input image (prior reflectance is used in this example) in the low* log **m** *PCs. This filter is similar to the low-pass PC Filter* PCF_L *of Figure 5.2, except that the low PCs are spatially low-pass filtered instead of simply being passed. Also, this filter is applied to the prior reflectance ensemble, while the filter of Figure 5.2 is applied to a* log **rm** *pixel.*

6.3 Combined SPIRE Algorithm Derivation

6.3.1 Algorithmic Approach and Issues

Our overall approach is a combination of Spectral SPIRE and Spatial SPIRE approaches.

Consider the multiplicative-only case. Our goal is to use a PC rotation to concentrate the $\log \mathbf{m}$ noise into a few PCs, and then spatially high-pass filter them to remove the $\log \mathbf{m}$, rather than simply setting the PCs to zero. This should preserve any reflectance changes in $\Delta \log \mathbf{r}$ and improve performance on modified pixels, while avoiding the higher computational costs of spatially filtering all bands. We then spatially low-pass filter the same low PCs from the prior, to restore what is lost from filtering $\Delta \log \mathbf{r}$.

The same assumption of an existing prior is still in force. The assumptions regarding the slow spatial variation of the multiplicative noise and that changes are small in area with respect to the scene, must be used as well since we are now employing spatial filtering. Since the two variants of combined filters applicable to our problem both perform the PC filtering first, we will begin with the image formation equation expressed in vector notation:

$$\mathbf{i} = \mathbf{r} \odot \mathbf{m} + \mathbf{a} \tag{6.1}$$

We shall again consider two cases, one where $\mathbf{a}$ is present, the other where it is not. These cases will be analogous to Cases C and D for Spectral SPIRE, but with the added constraint that the multiplicative noise $\mathbf{m}$ is spatially slowly varying, while $\mathbf{a}$ is still spatially uniform. This makes these two new cases identical to Spatial SPIRE Cases 3 and 4. However, to preserve our ability to refer to different algorithms based on their case designations, the combined algorithms for solving Spatial SPIRE Cases 3 and 4 will be referred to as Combined SPIRE Cases III and IV, the Roman numerals denoting combined processing. Since there is no Combined SPIRE unique solution techniques for Cases 1 and 2 as there was for Spatial SPIRE, these are included into Cases III and IV. Combined SPIRE for Cases VI (spatially varying additive noise, with Case V as a subset) will not be addressed in this thesis, since it cannot be

compared to both Spatial and Spectral SPIRE algorithms, since no Spectral SPIRE algorithm was developed for that case. Also, the Case IV algorithm will be the most useful algorithm for remotes sensing, since the additive noise **a** can be assumed to be spatially uniform in most applications.

6.3.2 Case III: Slowly Spatially Varying Multiplicative Noise Only

The Combined SPIRE Case III algorithm is now derived. In this case the additive noise **a** is zero. Our image formation model (6.1) then becomes:

$$\mathbf{i} = \mathbf{r} \odot \mathbf{m} \tag{6.2}$$

We first move to log space to linearize the problem:

$$\log \mathbf{i} = \log \mathbf{r} + \log \mathbf{m} \tag{6.3}$$

Figure 6.3(a) depicts the generalized processing block diagram for the Case III solution algorithm. This algorithm is analogous to the Case 3 general Spatial SPIRE algorithm of Figure 3.2(a) and Case C general Spectral SPIRE algorithm of Figure 5.3(a). Figure 6.3(b) depicts the specific algorithm implemented in this thesis and tested upon the same HYDICE test data set as the Spatial and Spectral SPIRE algorithms. Note the similarity between Figures 6.3, 5.3, and 3.2. The Case III algorithm is very similar to the Spectral Case C algorithm of Figure 5.3, the main difference being that the low PCs of the image are high-pass filtered rather than zeroed to remove $\log \mathbf{m}$.

In Combined SPIRE, the entire image pixel ensemble is processed together. All pixels are passed through the combined filter CF_α to spatially high-pass filter the $\log \mathbf{m}$ PCs and

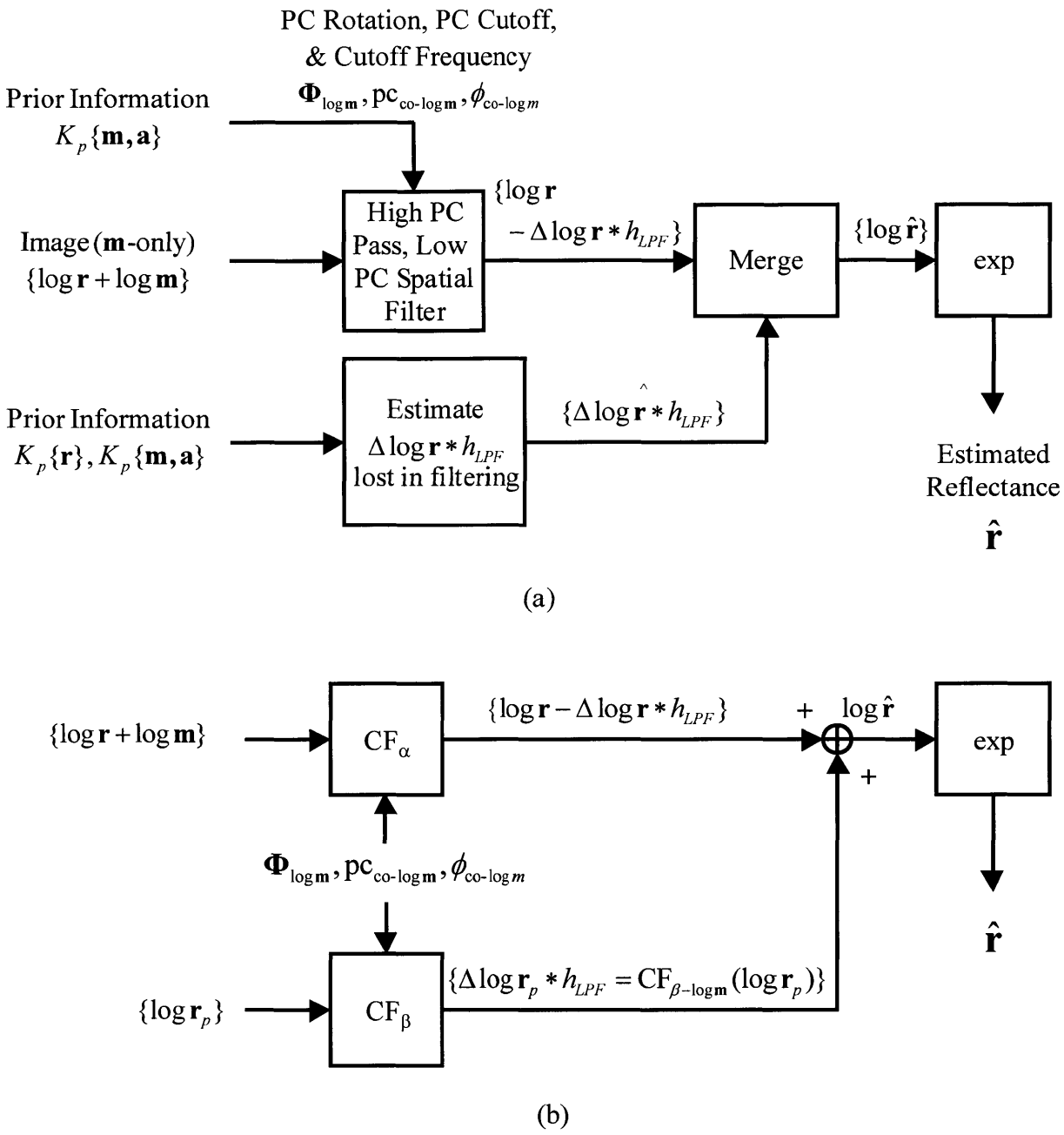


Figure 6.3. *Combined Case III generalized processing block diagram (a) and specific implementation tested (b). In (a), the* **m***-only image is combined* α *filtered using the* $\Phi_{\log \mathbf{m}}$ *PC rotation,* $\mathbf{pc}_{\text{co-log}m}$ *PC cutoff index, and the* $\phi_{\text{co-log}m}$ *spatial cutoff frequency from the prior information about the multiplicative noise* **m** ($K_p\{\mathbf{m},\mathbf{a}\}$). *The filtered* $\{\log \mathbf{r}\}$ *lost in the operation is then estimated using prior information* $K_p\{\mathbf{r}\}$ *and* $K_p\{\mathbf{m},\mathbf{a}\}$ *and merged with the filtered signal. Finally, the exponential is taken to estimate reflectance. In (b), the lost combined filtered* $\log \mathbf{r}$ *is estimated by combined* β *filtering the log of the prior reflectance* $\{\log \mathbf{r}_p\}$, *and the merge operation is a simple addition.*

the subscripts α and β define the type of combined filter as defined in Section 6.2, and the $_{\text{-var}}$ thereby eliminate $\log\mathbf{m}$. We denote such a combined filtering operation as $\text{CF}_{\alpha/\beta\text{-var}}(\ )$, where subscript describes the variable from which the PC rotation was defined which is used in the combined filter. For example, $\text{CF}_{\alpha\text{-log}\mathbf{m}}(\ )$ is a CF_{α} combined filter that uses the PC rotation matrix $\mathbf{\Phi}_{\log\mathbf{m}}$ and PC cutoff $\text{pc}_{\text{co-log}\mathbf{m}}$. Applying such a filter to $\log\mathbf{i}$ yields:

$$\text{CF}_{\alpha-\log\mathbf{m}}(\{\log\mathbf{i}\}) = \text{CF}_{\alpha-\log\mathbf{m}}(\{\log\mathbf{r}+\log\mathbf{m}\}) = \{\log\mathbf{r} - \Delta\log\mathbf{r} * h_{LPF}\} \tag{6.4}$$

We must now estimate the lost $\{\Delta\log\mathbf{r} * h_{LPF}\}$, by applying a CF_{β} filter to the prior $\{\log\mathbf{r}_p\}$:

$$\{\Delta\log\hat{\mathbf{r}} * h_{LPF}\} = \{\Delta\log\mathbf{r}_p * h_{LPF}\} = \text{CF}_{\beta-\log\mathbf{m}}(\{\log\mathbf{r}_p\}) \tag{6.5}$$

This estimate of $\{\Delta\log\mathbf{r} * h_{LPF}\}$ is then added to $\{\log\mathbf{r} - \Delta\log\mathbf{r} * h_{LPF}\}$ to estimate $\{\log\mathbf{r}\}$:

$$\begin{aligned}\{\log\hat{\mathbf{r}}\} &= \{\log\mathbf{r}\} - \{\Delta\log\mathbf{r} * h_{LPF}\} + \{\Delta\log\hat{\mathbf{r}} * h_{LPF}\} \\ &= \text{PCF}_{\alpha-\log\mathbf{m}}(\{\log\mathbf{r}+\log\mathbf{m}\}) + \text{PCF}_{\beta-\log\mathbf{m}}(\{\log\mathbf{r}_p\})\end{aligned} \tag{6.6}$$

Lastly, we take the exponential to estimate reflectance:

$$\{\hat{\mathbf{r}}\} = \exp(\{\log\hat{\mathbf{r}}\}) \tag{6.7}$$

We note that the Combined Case I algorithm would be identical to the Combined Case III algorithm, except that a spatial mean filter could be used in place of a low-pass filter if desired.

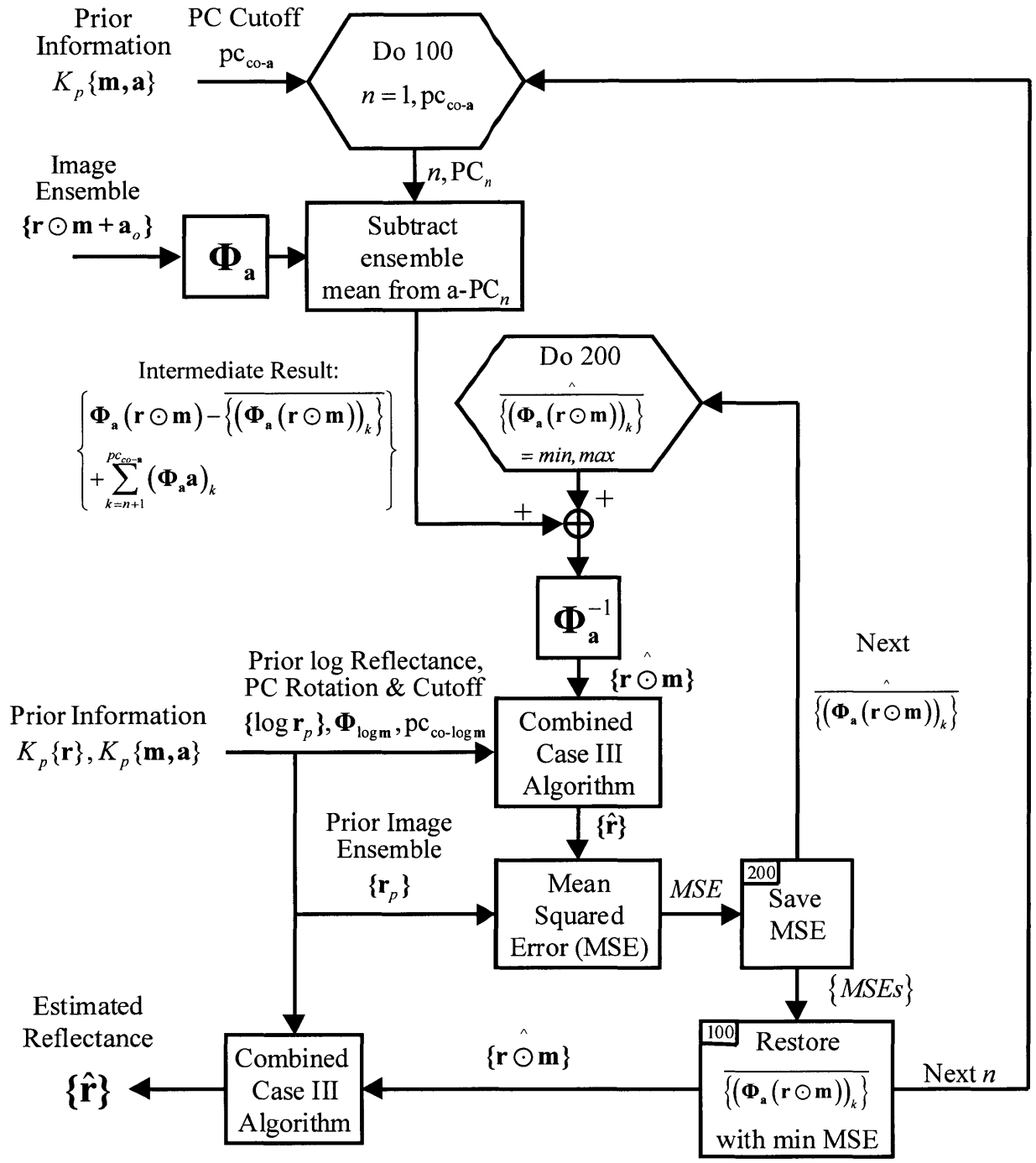


Figure 6.4. *Combined Case IV processing block diagram. Based on the additive noise* **a***-PC cutoff* $pc_{co\text{-}a}$*, all of the* **a***-PCs are looped through. For each* **a***-PC_n, the ensemble mean of the PC is subtracted to remove the a in that PC. An inner loop then steps through the possible values for the lost mean of* $\mathbf{r}\odot\mathbf{m}$ *in PC_n, using the Case C algorithm and MSE minimization as in the Spatial Case 4 algorithm. Once all of the* **a***-PCs have been processed, then the Combined Case III algorithm is run one last time to estimate the reflectance. This algorithm is similar to the Spectral Case D algorithm (Figure 5.6) except that the Combined Case III algorithm (Figure 6.3) is used in place of the Spectral Case C (Figure 5.3).*

6.3.3 Case IV: Slowly Spatially Varying Multiplicative and Spatially Uniform Additive Noise

The solution algorithm for this case is depicted in Figure 6.4. This combined Case IV algorithm and Figure 6.4 are nearly identical to the Spectral SPIRE Case B algorithm depicted in Figure 5.6. The only difference is that the Combined Case III algorithm is used in place of the spectral Case C algorithm. Please refer to Section 5.3.3 for the detailed description of this algorithm. We note that the Combined Case II algorithm would be identical to the Combined Case IV, except that a spatial mean filter could be used in place of a low-pass filter to remove log **m** noise in the Case III algorithm.

6.4 Algorithm Testing and Results

The test data set of six ARM Site image cubes used for testing Spatial SPIRE in Chapter 3 and Spectral SPIRE in Chapter 5 was used to test the Combined SPIRE Case IV algorithm developed in this chapter. The same ELM and ATREM processing results from Chapter 3 were also compared with the Combined SPIRE results against. The same pixels from the images were used as well. Please refer to Section 3.3.4.4 for a description of the different pixel types used. As in the Spectral SPIRE Case D processing in Chapter 5, 5 $\log \mathbf{m} - $PCs and 5 **a**-PCs were processed ($pc_{co\text{-}log\mathbf{m}}=pc_{co\text{-}\mathbf{a}}=5$) in the Combined SPIRE Case IV processing.

Figure 6.5 is a scatter plot of mean and standard deviation performance of Combined SPIRE, ELM, and ATREM for estimation of surface spectral reflectance for all of the 19 pixel types selected. This figure is directly analogous to Figure 3.9 for Spatial SPIRE and Figure 5.7 for Spectral SPIRE. The horizontal axis represents the average standard deviation over all the spectral channels, where the standard deviation in each channel was calculated over the reflectance estimates of all six Runs (06-31). The vertical axis is the RMS error over all the

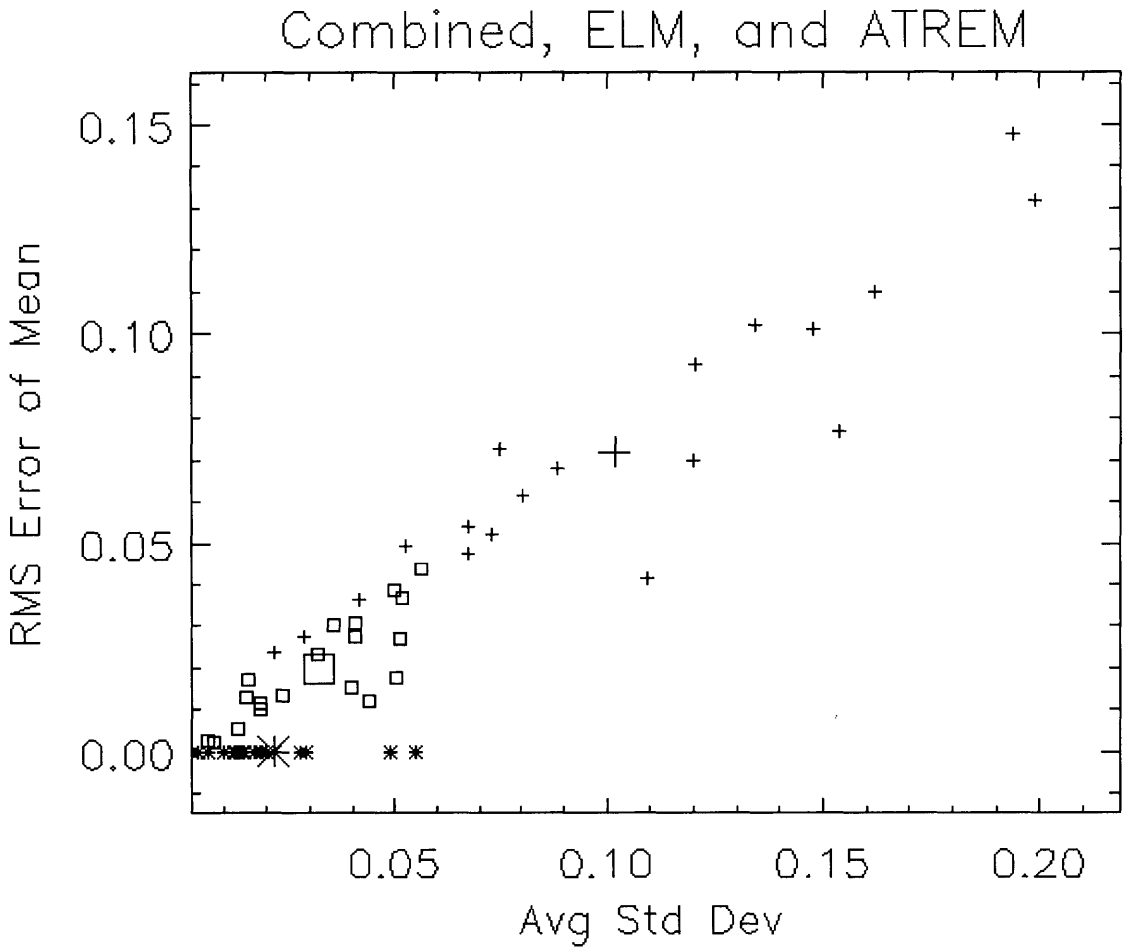


Figure 6.5. *Scatter plot of mean and standard deviation performance of Combined SPIRE (squares), ELM (*), and ATREM (+) for estimation of surface spectral reflectance for all of the 19 pixel types selected. The horizontal axis represents the average standard deviation over all the spectral channels, where the standard deviation in each channel was calculated over the reflectance estimates of all six Runs (06-31). The vertical axis is the RMS error over all the spectral channels for the mean reflectance estimate minus the mean ELM reflectance estimate, which is why ELM has zero RMS error. The larger symbols represent the mean of the points plotted with that symbol. We see that Combined SPIRE has better standard deviation and RMS error performance than ATREM.*

spectral channels for the mean reflectance estimate minus the mean reflectance estimate of ELM as ground truth, which is why ELM has zero RMS error. We see that Combined SPIRE has better standard deviation and RMS error performance than ATREM, and nearly as good standard deviation as ELM.

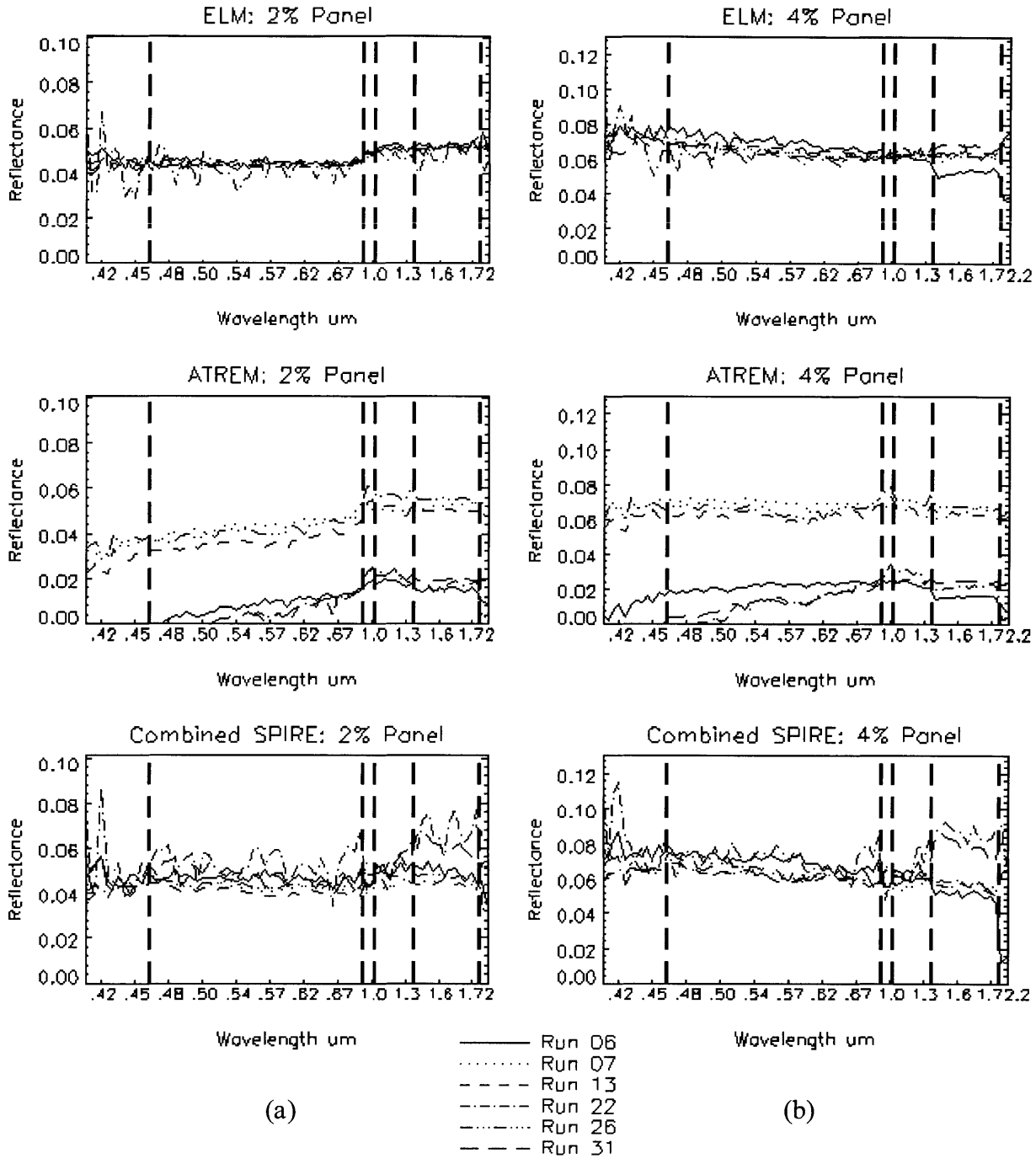


Figure 6.6 (a) and (b). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the 2 percent panel (a) and the 4 percent panel (b).*

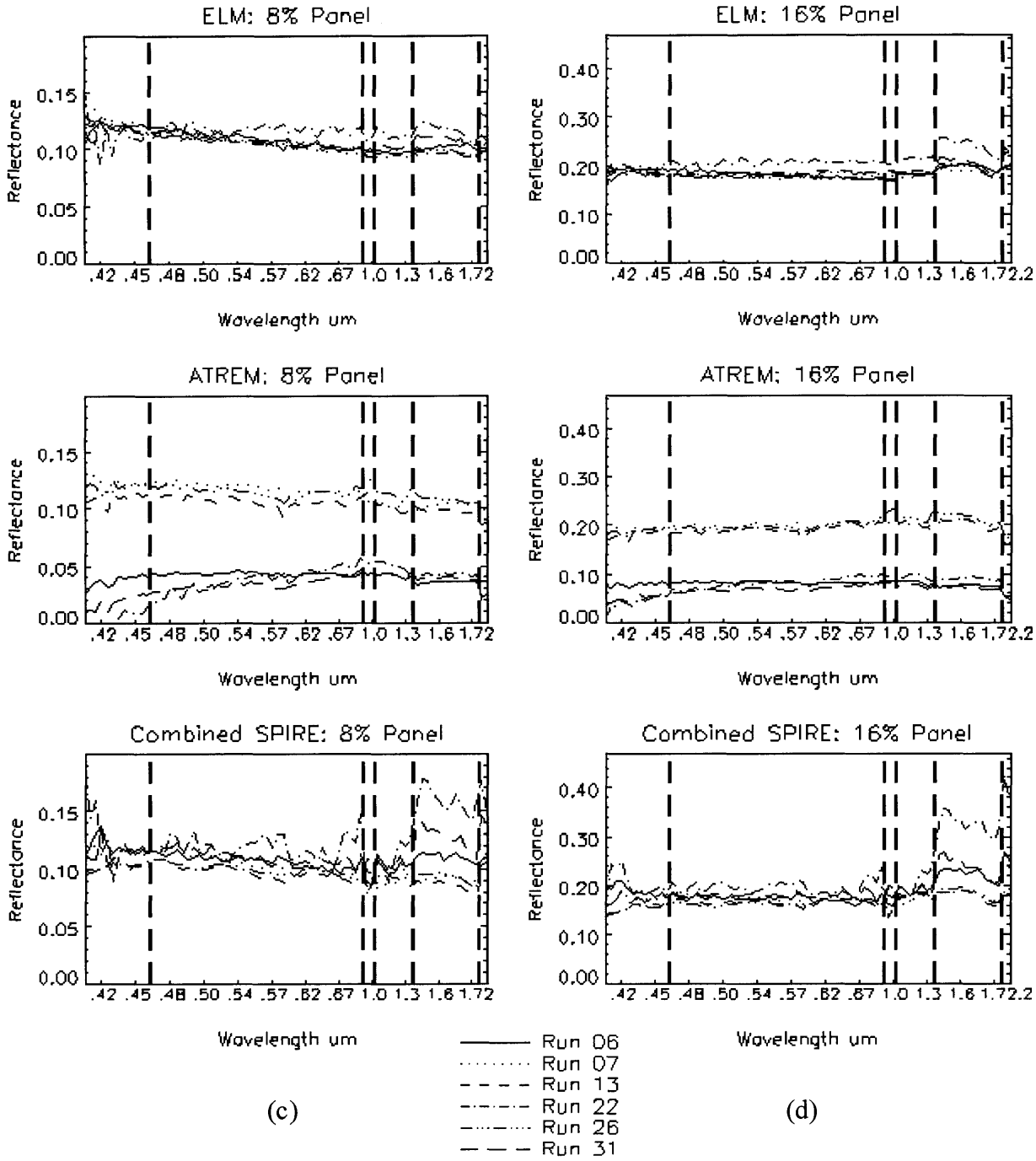


Figure 6.6 (c) and (d). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the 8 percent panel (c) and the 16 percent panel (d).*

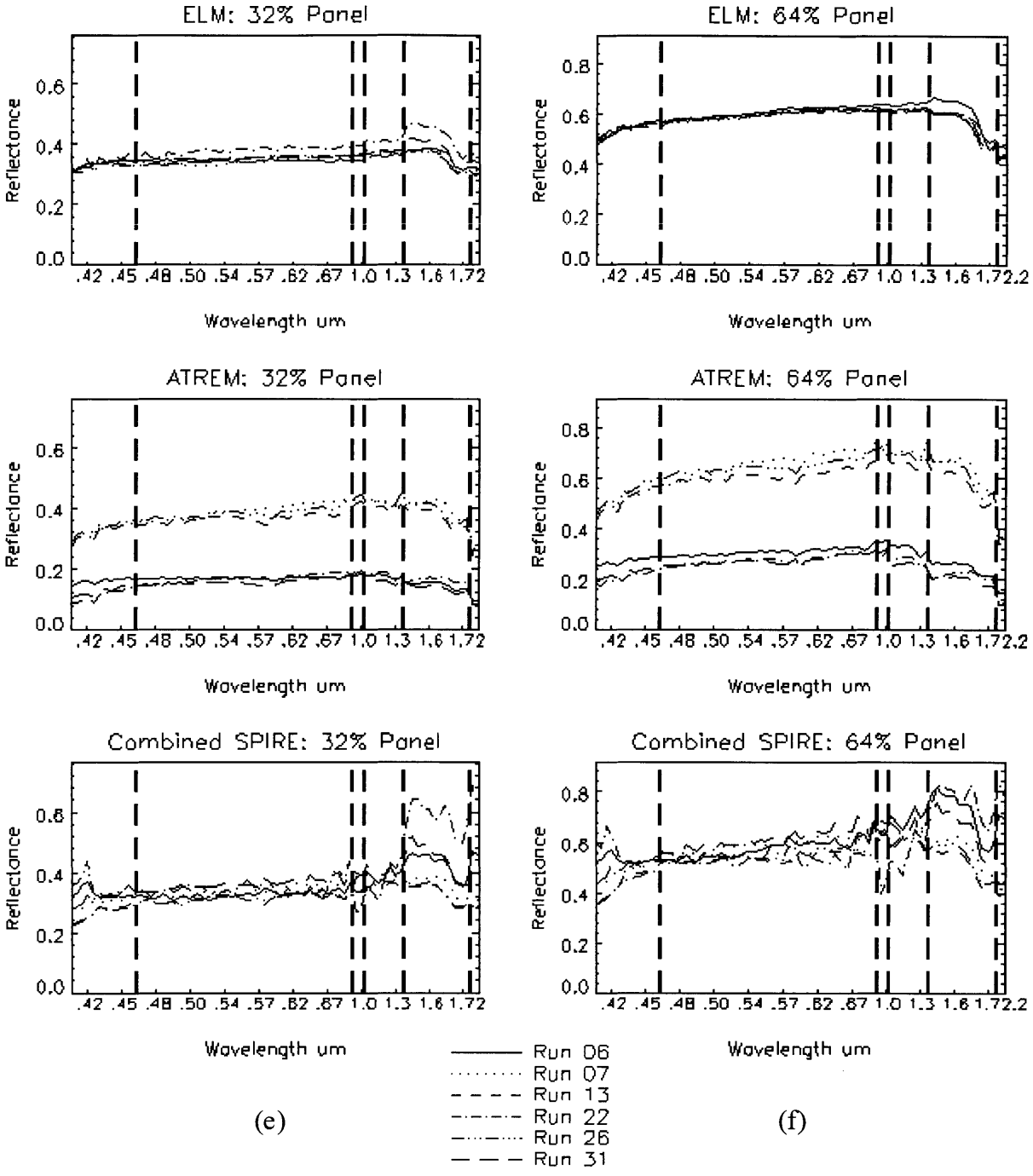


Figure 6.6 (e) and (f). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the 32 percent panel (e) and the 64 percent panel (f).*

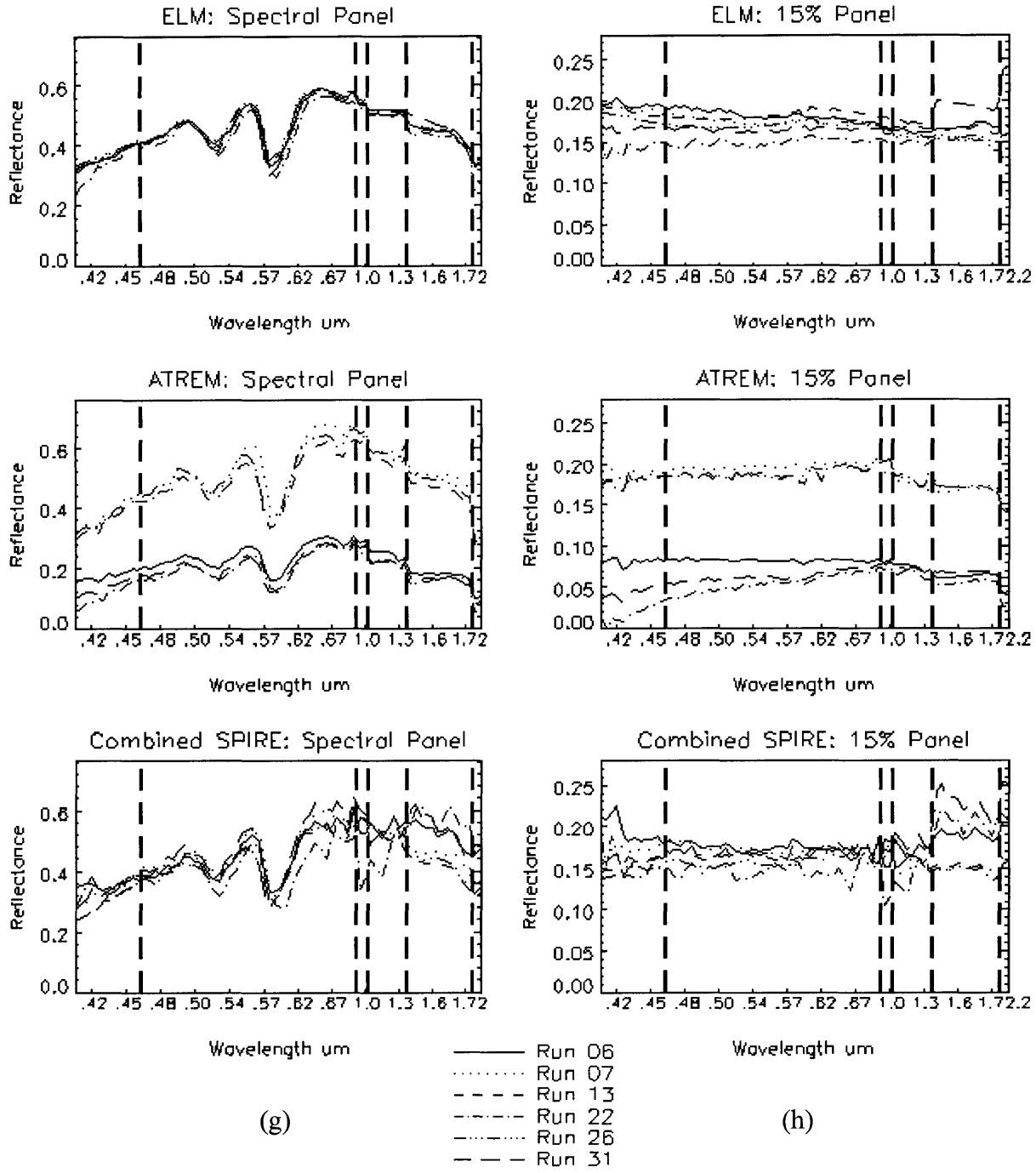


Figure 6.6 (g) and (h). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the spectral panel (g) and the 15 percent panel (h).*

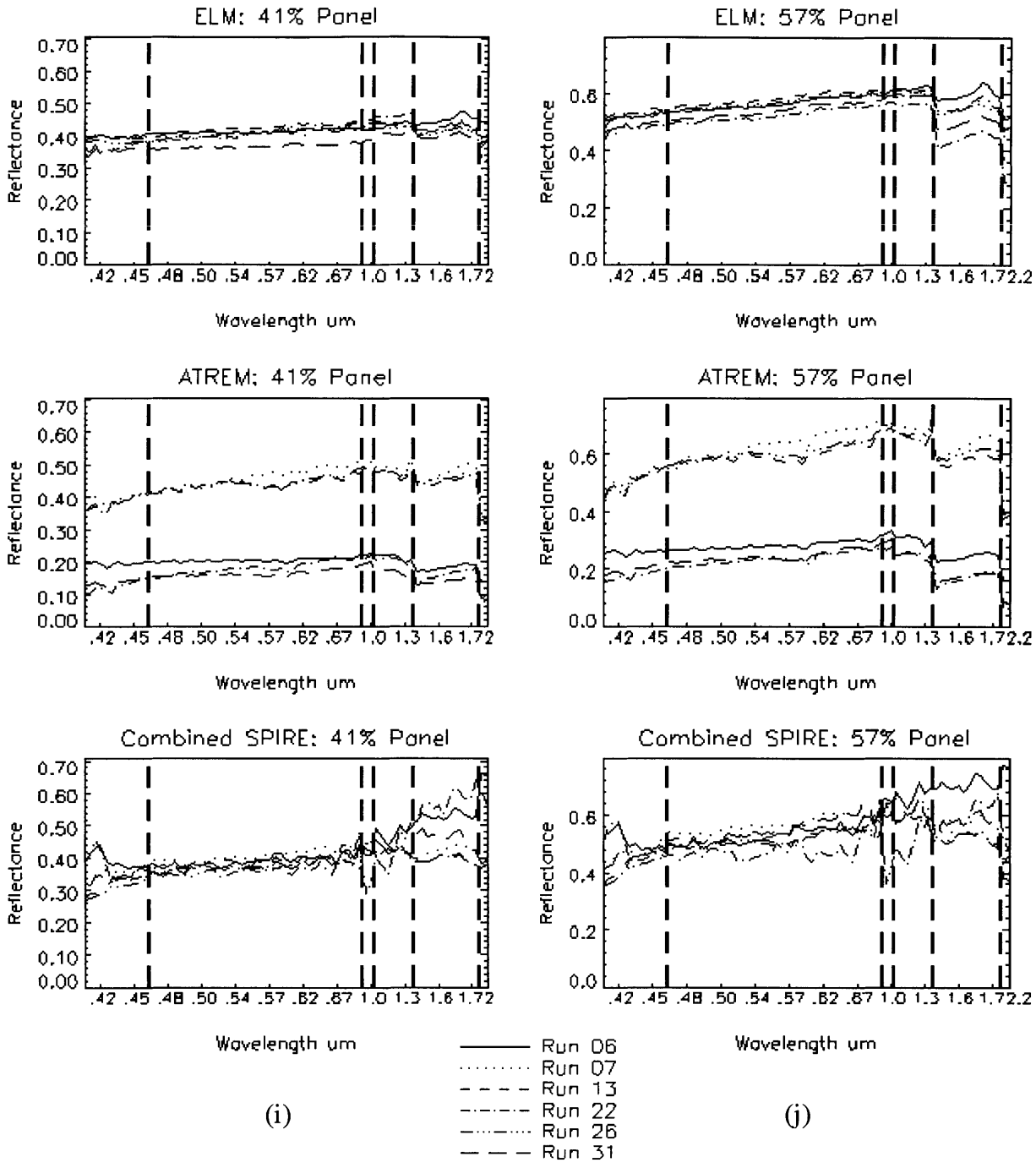


Figure 6.6 (i) and (j). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the 41 percent panel (i) and the 57 percent panel (j).*

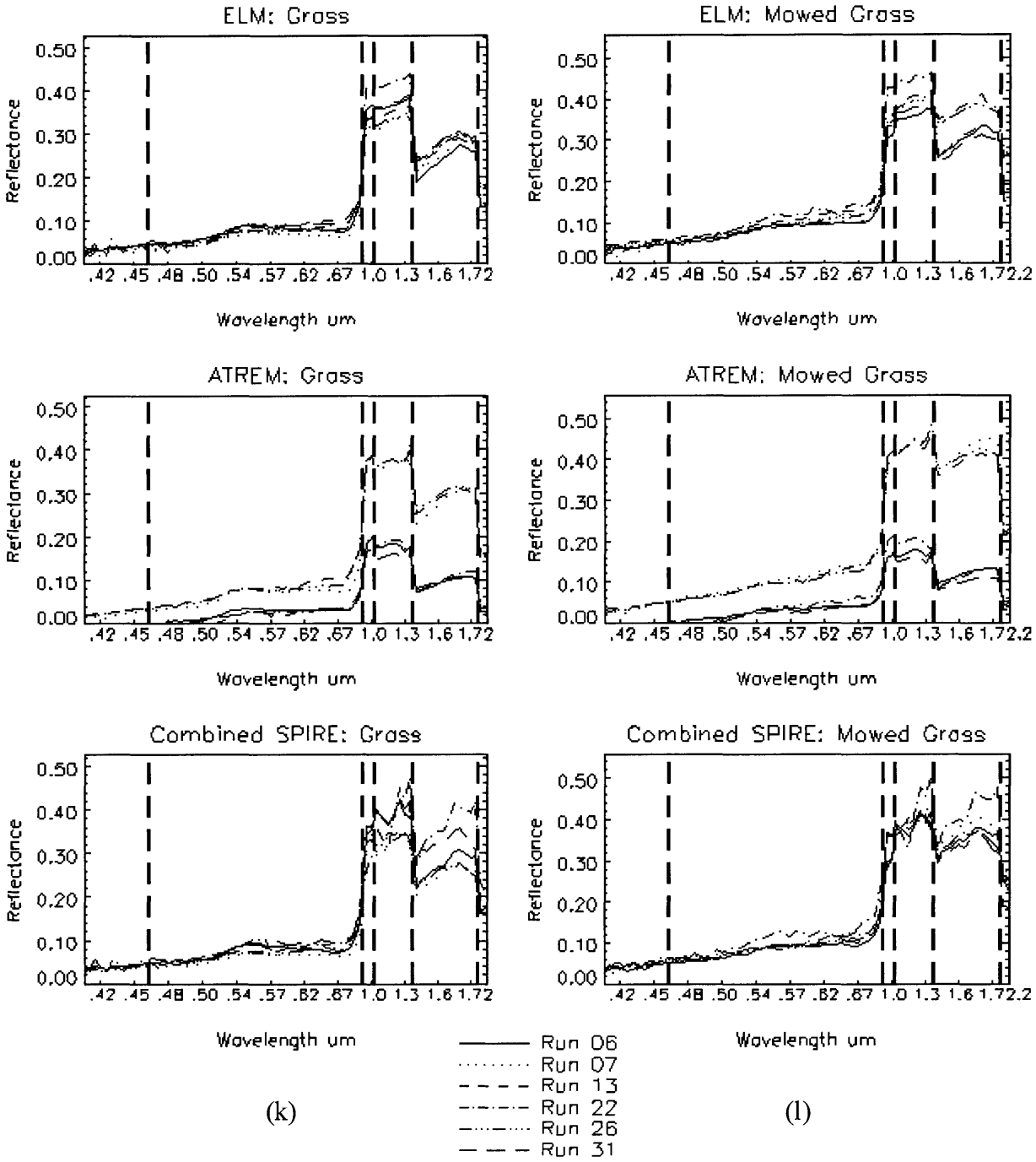


Figure 6.6 (k) and (l). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on grass (k) and mowed grass (l).*

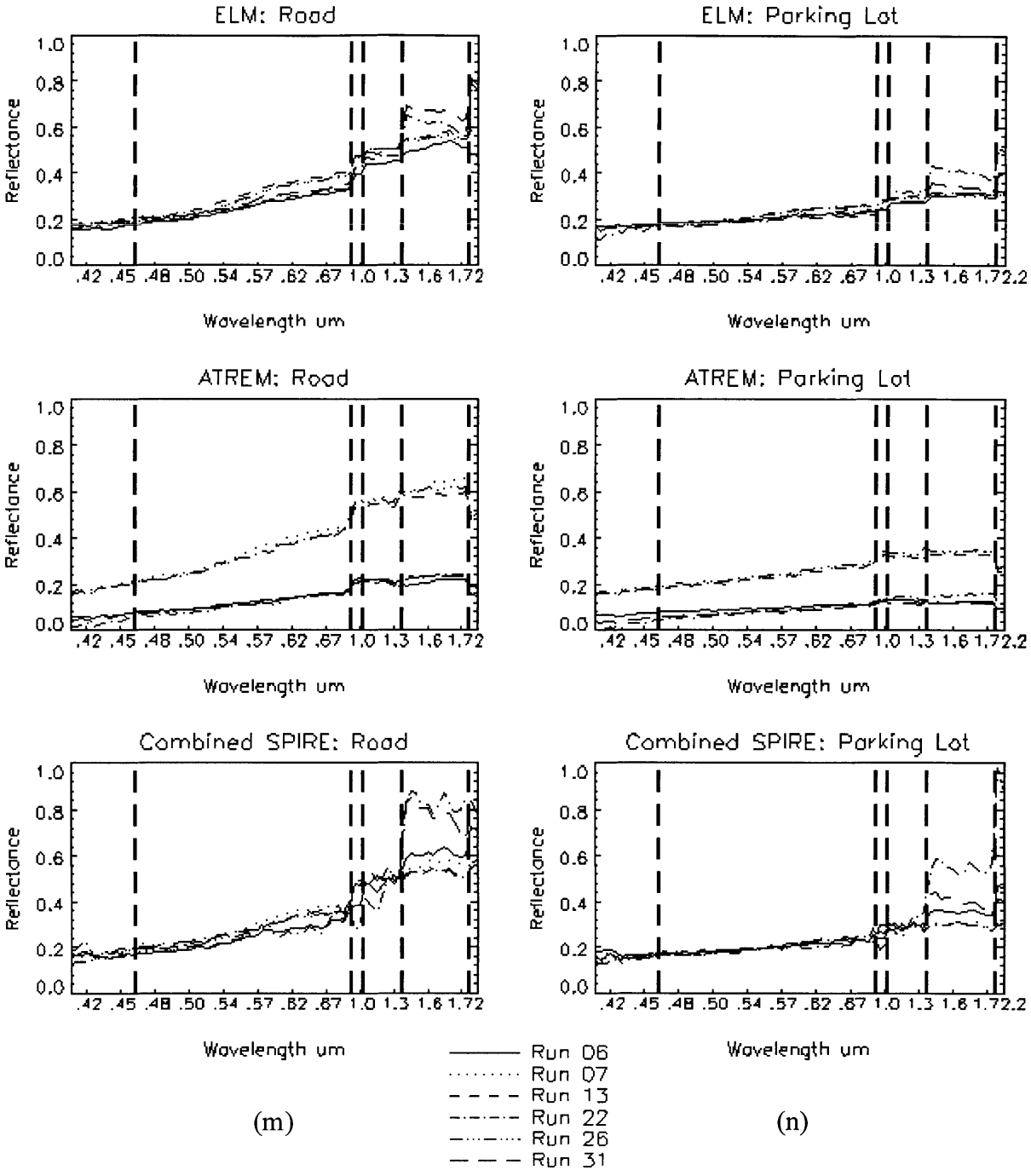


Figure 6.6 (m) and (n). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the road (m) and the parking lot (n).*

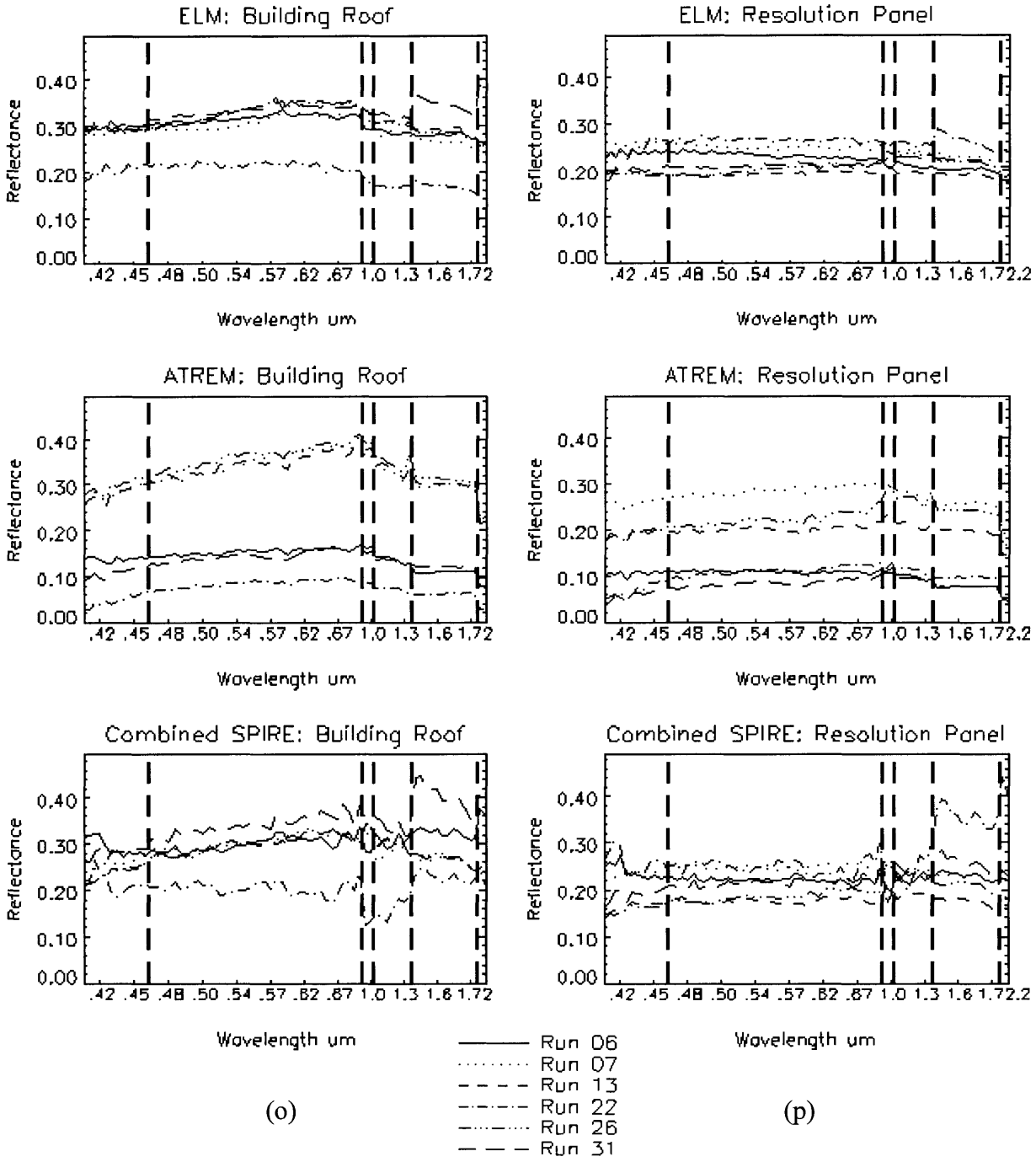


Figure 6.6 (o) and (p). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the building roof (o) and the resolution panel (p)..*

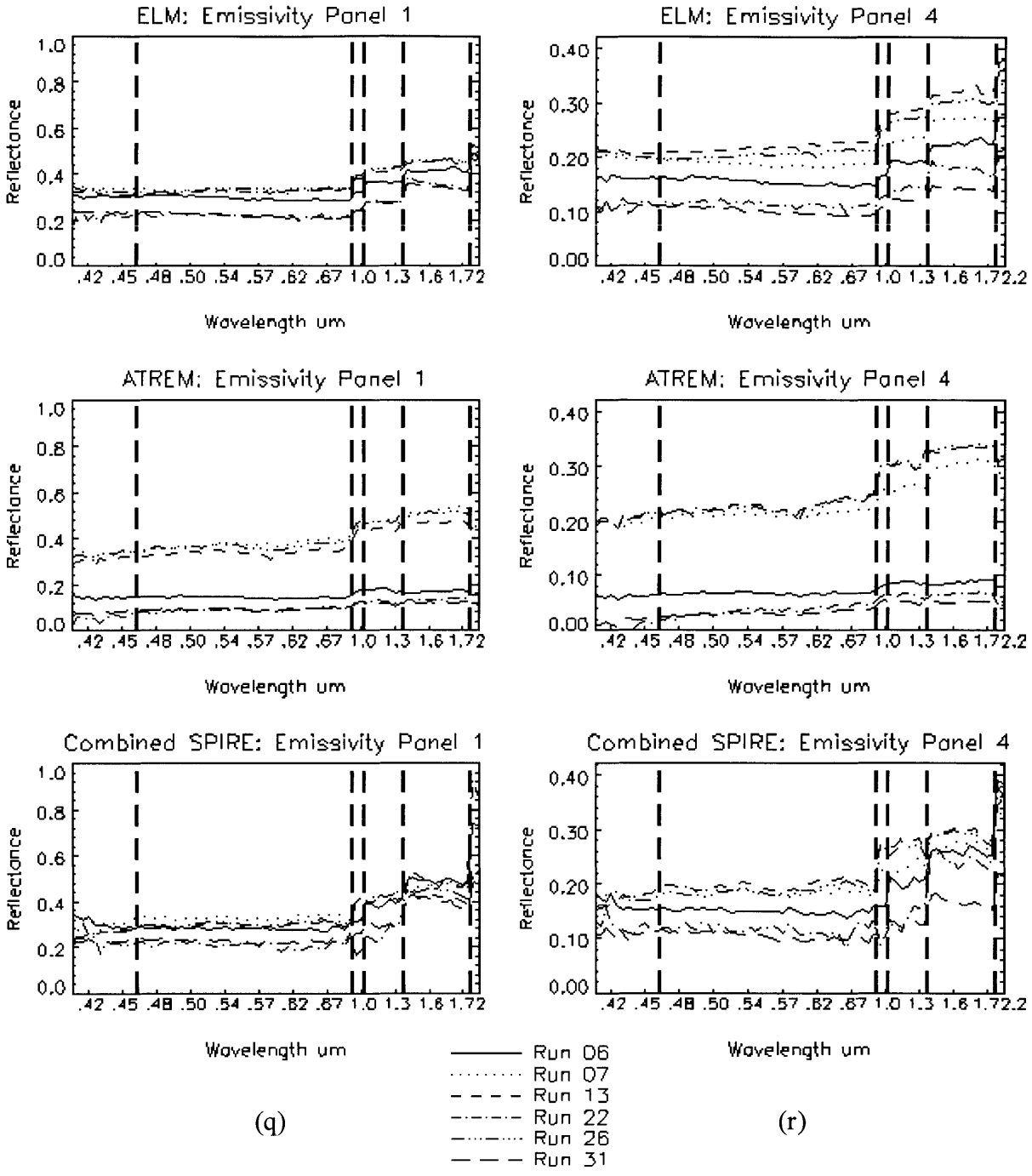
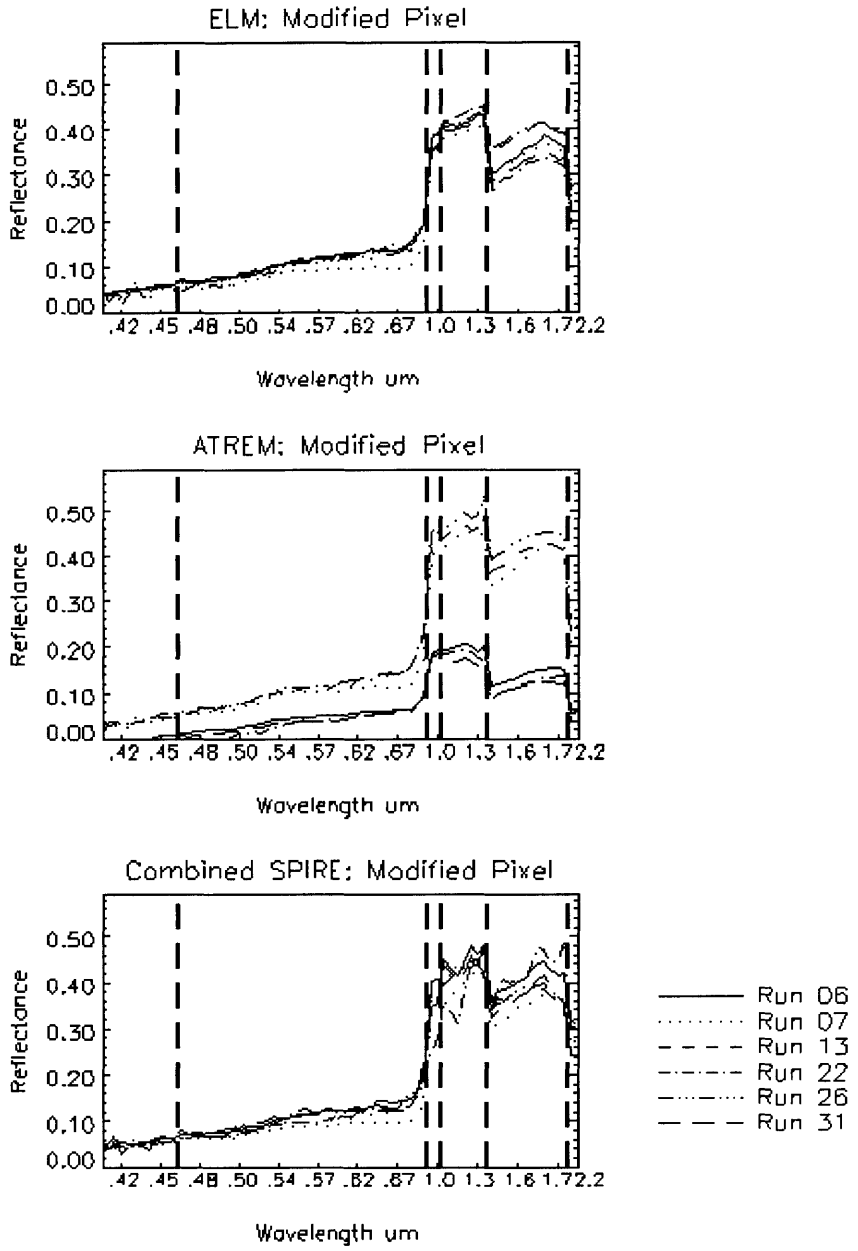


Figure 6.6 (q) and (r). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on emissivity panel 1 (q) and emissivity panel 4 (r).*



(s)

Figure 6.6(s). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the modified pixel. The modified pixel was a truck in Run 07, and changed to mowed grass in all subsequent Runs. For Run 07, a similar mowed grass pixel is plotted for comparison.*

Figure 6.6(a)-(s) is directly analogous to Figure 3.8(a)-(s) and Figure 5.8(a)-(s) for comparing Combined SPIRE performance against that of ELM and ATREM. Similar plots with the vertical axis scaled from 0 to 1 can be found in Appendix F as Figure F.1. Appendix F also contains Tables F.1 and F.2 which are analogous to Tables C.1 and C.2 for Spatial SPIRE and Tables E.2 and E.2 for Spectral SPIRE and present the average channel standard deviation values in each band for the ELM, ATREM, and Spectral SPIRE results plotted in Figure 6.6.

Again we note the clustering of the ATREM spectra into two distinct sets due to clouds and haze on three of the runs, as was discussed in Chapter 3. We also note that for all pixels, including the modified pixel, Combined SPIRE performs better than ATREM and very similar to ELM. We shall compare the relative performance of Spatial, Spectral, and Combined SPIRE in more detail in Chapter 7, but we note here that unlike Spectral SPIRE, Combined SPIRE has good performance on both modified and unmodified pixels. The addition of spatial filtering to the Spectral SPIRE algorithm dramatically improved the performance on the modified pixel, as can be seen in Figure 6.6(s). In addition, Combined SPIRE has better performance than Spatial SPIRE on the low reflectance panels, which can best be seen in Figure F.1(a) and (b).

6.5 Computational Cost of Spatial, Spectral, and Combined SPIRE

6.5.1 Combined SPIRE Computational Costs Estimate

Here we make the same assumptions as in Section 5.5, where we have a spatially square image cube with N spatial samples and nb spectral channels. We therefore have an NxNxnb image cube to be processed and only multiplicative noise present. The computational cost estimate for the Combined SPIRE Case III algorithm is:

$$
\begin{aligned}
&\{\mathbf{rm}\} \rightarrow \{\log \mathbf{rm}\}: && 2N^2 \\
&\{\mathbf{r}_p\} \rightarrow \log \mathbf{r}_p: && 2N^2 \\
&\{\Phi_{\log \mathbf{rm}} \log \mathbf{rm}\}: && 2(nb)N^2 \\
&\{\Phi_{\log \mathbf{rm}} \log \mathbf{r}_p\}: && 2(nb)N^2 \\
&\{\Delta \log \mathbf{rm} * h_{HPF}\}: && 12(pc_{co-\log \mathbf{m}})N^2 \log_2(N^2) + N^2 \\
&\{\Delta \log \mathbf{r}_p * h_{LPF}\}: && 12(pc_{co-\log \mathbf{m}})N^2 \log_2(N^2) + N^2 \\
&\{\Delta \log \mathbf{rm} * h_{HPF}\} + \{\Delta \log \mathbf{r}_p * h_{LPF}\}: && (pc_{co-\log \mathbf{m}})N^2 \\
&\{\Phi^{-1}_{\log \mathbf{rm}} \log \hat{\mathbf{r}}\}: && 2(nb)N^2 \\
&\exp(\{\log \hat{r}\}): && 2N^2
\end{aligned}
\tag{6.8}
$$

Yielding a total cost function of:

$$
\begin{aligned}
\text{Combined SPIRE Cost} &= 6N^2 + 6(nb)N^2 + (pc_{co-\log \mathbf{m}})N^2 \\
&\quad + (pc_{co-\log \mathbf{m}})[24N^2 \log_2(N^2) + 2N^2] \\
&= \text{Spectral SPIRE Cost} + (pc_{co-\log \mathbf{m}})24N^2 \log_2(N^2) + 2N^2] \\
&= N^2\left(6 + 6(nb) + (pc_{co-\log \mathbf{m}})[24\log_2(N^2) + 3]\right) \\
&\simeq 6(nb)N^2 + 240N^2 \log_2(N)
\end{aligned}
\tag{6.9}
$$

Where the total costs functions for Spatial and Spectral SPIRE were:

$$
\begin{aligned}
\text{Spatial SPIRE Cost} &= \left(9N^2 + 24N^2 \log_2\left(N^2\right)\right)nb \\
&= N^2\left(9 + 24\log_2\left(N^2\right)\right)nb \\
&\simeq 48(nb)N^2 \log_2(N)
\end{aligned}
\tag{6.10}
$$

$$
\begin{aligned}
\text{Spectral SPIRE Cost} &= 6N^2 + 6(nb)N^2 + \left(pc_{co-\log \mathbf{m}}\right)N^2 \\
&= N^2\left(6(1 + nb) + pc_{co-\log \mathbf{m}}\right) \\
&\simeq 6(nb)N^2
\end{aligned}
\tag{6.11}
$$

For a (100x100x100) image cube with $pc_{co-\log \mathbf{m}} = 5$, the Combined SPIRE cost would be:

2.22×10^7 FLOPS, as compared to 6.11×10^6 FLOPS for Spectral SPIRE and 3.28×10^8 FLOPS for Spatial SPIRE. This makes Combined SPIRE 3.6 times slower than Spectral SPIRE while Spatial SPIRE is 50 times slower than Spectral SPIRE for **m**-only. This makes Spatial SPIRE 15 times slower than Combined SPIRE as well. For Spatial Cases 4 and Combined Case IV where **a** is present, and assuming a 20 times difference in iteration time as in Section 5.4.2, Spatial SPIRE is 300 times slower than Combined SPIRE.

Comparing Spatial SPIRE Case 4 algorithm to Spectral SPIRE Case C where both multiplicative and additive noise are present, the disparity is even greater. Assuming that the same number of iteration steps are required for each spatial band or PC to be processed to remove the additive noise, and assuming that 5 a-PCs must be processed versus all channels in Spatial SPIRE, then there is an additional factor of 20 favoring Spectral SPIRE, making the Case C algorithm 200 times faster than the Case 4.

Figure 6.7 shows a scatter plot of the average channel standard deviation across all runs for the modified pixel versus the computational load for each SPIRE algorithm. We see that Spatial SPIRE has high computational cost but low RMS error, Spectral SPIRE has low cost but high RMS error, and Combined SPIRE has nearly as low cost as spectral with nearly as low error as Spatial. When we also take into account Combined SPIRE's improved performance on low reflectivity materials under low SNR conditions, Combined SPIRE stands out as the best all around individual SPIRE algorithm, combining high performance on all pixel types with low computational cost.

6.5.2 Computational Costs of Spatial, Spectral, and Combined SPIRE Versus Image Cube Size

Computer processing power has been steadily increasing over time. As technology

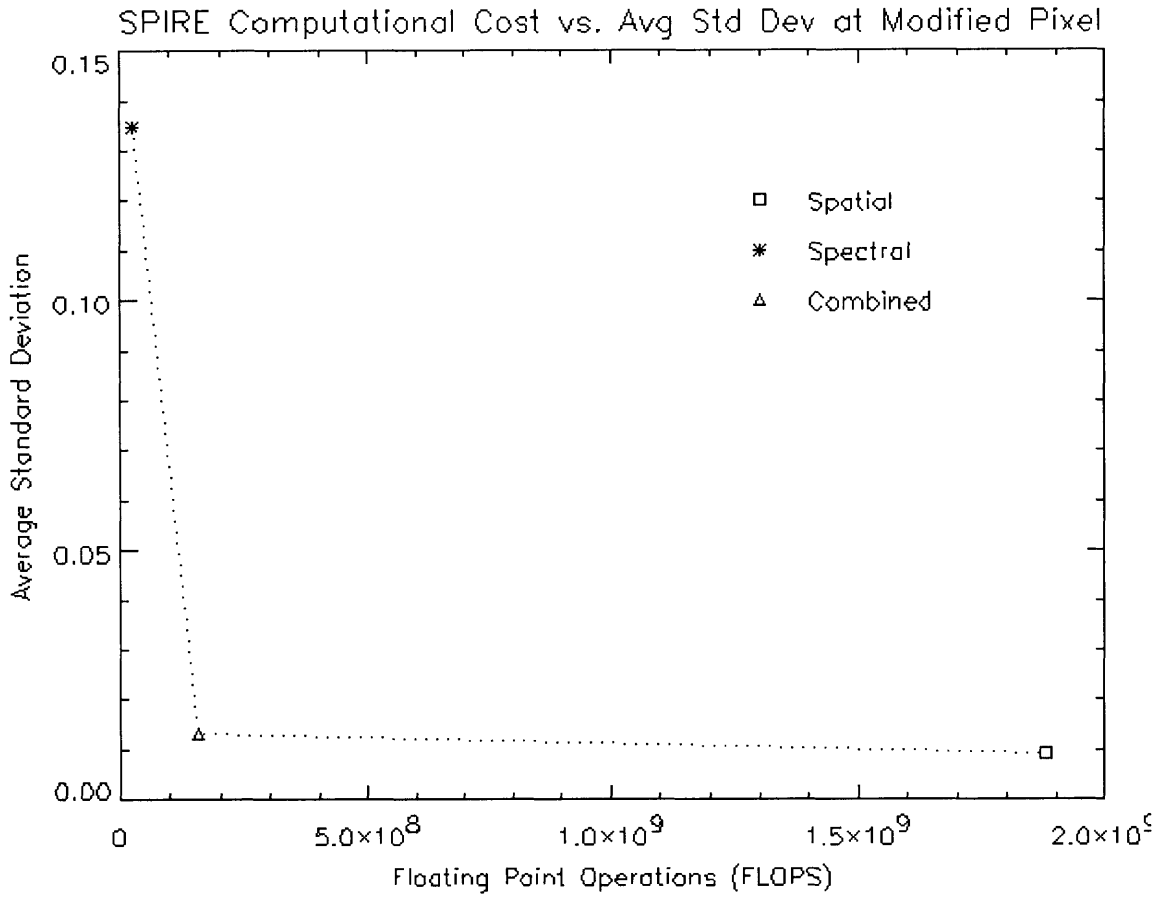


Figure 6.7. *Scatter plot of standard deviation performance and computational cost of Spatial, Spectral, and Combined SPIRE on the modified pixel. The horizontal axis represents the computational costs of the multiplicative noise only algorithms (Spatial SPIRE Case 3, Spectral SPIRE Case A, and Combined SPIRE Case III) in Floating Point Operations (FLOPS). The vertical axis represents the average standard deviation over all the spectral channels, where the standard deviation in each channel was calculated over the reflectance estimates of all six Runs (06-31). We see that Spectral SPIRE has low computational cost but high standard deviation, Spatial SPIRE has high computational cost and low standard deviation, and Combined SPIRE has nearly as low computational costs as Spectral SPIRE and almost as good standard deviation performance as Spatial SPIRE.*

advances, so has the spatial resolution, spectral resolution, and image size of hyperspectral image cubes. As the dimensions of a hyperspectral image cube grow, so do the computational costs of each SPIRE algorithm. If we are to estimate the reflectance of modified pixels well, and we are dealing with high SNR applications where Spatial SPIRE does well on low reflectivity materials,

then our choices are between Spatial and Combined SPIRE. If computational costs are a factor in the choice, then we must trade off the slightly better performance of Spatial SPIRE on modified pixels versus its computational cost compared to Combined SPIRE.

Figure 6.8 plots the computational costs for processing an *N*x*N*x100 pixel image cube with **m**-only Spatial (Case 3), Spectral (Case C), and Combined (Case III) SPIRE algorithms, as the spatial dimension N increases. From this plot we see that Spatial SPIRE's computational costs grow much faster than either Spectral or Combined SPIRE as the spatial dimension increases. This is due to the fact that Spatial SPIRE costs grow at approximately $4800N^2\log_2(N)$, versus approximately $600N^2$ for Spectral and $600N^2+240N^2\log_2(N)$ for Combined SPIRE.

Figures 6.9 plots a similar cost curve but for a 100x100x *nb* image cube as the spectral dimension *nb* increases. As expected, Spatial SPIRE costs increase linearly with *nb* with the slope being determined by the cost of processing a single (100x100) channel image: $48\left(100^2\right)\log_2(100)nb$. Spectral and Combined costs also increase nearly linearly, their slope equal to 60000 as *nb* gets large.

Figure 6.10(a) plots a similar cost curve but for an *N*x*N*x*N* image cube as all three dimension increase at the same time. Figure 6.10(a) plots the same curve as in (a) but with a log computational cost axis to make the behavior of Spectral and Combined SPIRE more apparent. These plots combine the effects plotted in Figures 6.8 and 6.9. As all dimensions increase simultaneously, Spectral and Combined both increase at approximately $6N^3$ while Spatial increases even faster, at approximately $48N^3\log_2(N)$.

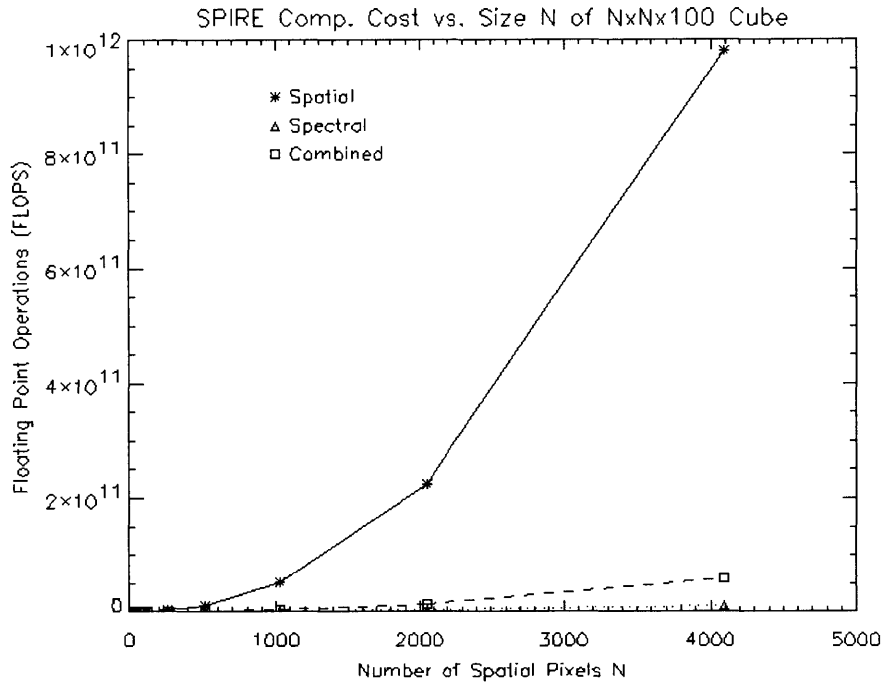


Figure 6.8. *Computational costs for processing an (NxNx100) pixel image cube with* ***m****-only Spatial, Spectral, and Combined SPIRE algorithms, as the spatial dimension N increases.*

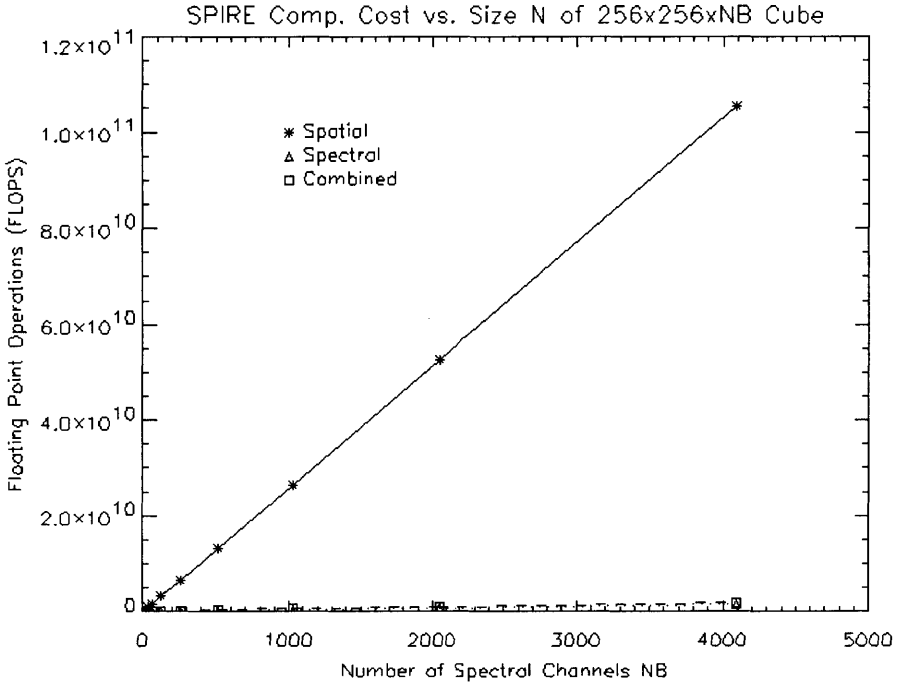
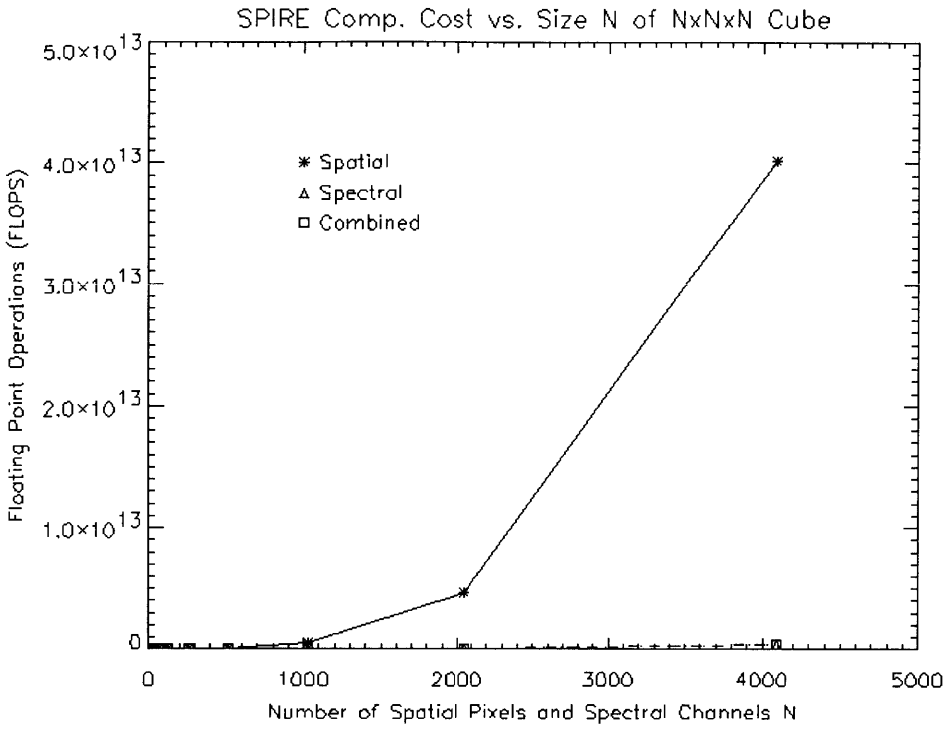
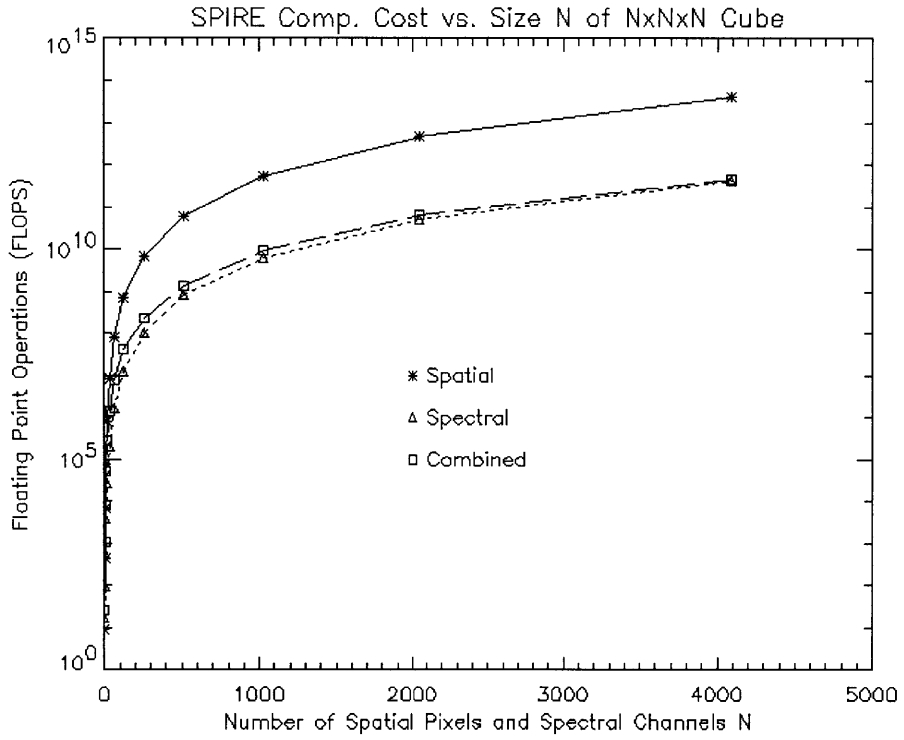


Figure 6.9. *Computational costs for processing a (100x100xnb) pixel image cube with m-only Spatial, Spectral, and Combined SPIRE algorithms, as the spectral dimension nb increases.*



(a)



(b)

Figure 6.10. *Computational costs for processing an (NxNxN) pixel image cube with* ***m****-only Spatial, Spectral, and Combined SPIRE algorithms, as the all dimensions N increase simultaneously (a) and with log axis for computational cost (b).*

6.6 Noise Analysis

In this section we perform similar noise analyses as was done in Chapter 5 for Spectral SPIRE algorithms.

6.6.1 Registration Noise

Imperfect registration of the image and its prior introduces noise that can effect reflectance estimation. Figure 6.11 is analogous to Figures 3.10 and 5.10 and shows distance images that measure the magnitude of the vector difference between the corresponding pixels of two images. For two reflectance images r_1 and r_2, the distance $d[n_{x_o}, n_{y_o}]$ between the pixel spectra $\mathbf{r}_1[n_{x_o}, n_{y_o}, n_\lambda]$ and $\mathbf{r}_2[n_{x_o}, n_{y_o}, n_\lambda]$ is defined as:

$$d[n_{x_o}, n_{y_o}] = \sqrt{\sum_{n_\lambda=1}^{N} \left(\mathbf{r}_1[n_{x_o}, n_{y_o}, n_\lambda] - \mathbf{r}_2[n_{x_o}, n_{y_o}, n_\lambda]\right)^2} \tag{6.12}$$

In Figure 6.11(a), we see the distance image between the prior used for Run 06 and the ELM estimated reflectance for Run 06, just as in Figures 3.9(a) and 5.8(a). In Figure 6.11(b) is depicted the distance between the Combined SPIRE reflectance estimates and ELM estimated reflectance. We see that the misregistration noise has not carried over and affected the Combined SPIRE results, just as in Spatial SPIRE. Therefore, the introduction of the spatial filtering has eliminated the effects of misregistration noise that caused errors in Spectral SPIRE. This is because only low spatial frequencies are restored from the prior, so that no high frequency prior misregistration noise is introduced into the reflectance estimate. Since misregistration noise does not have a significant impact on Spatial SPIRE results, there is no need to develop equations for the errors involved.

(a) (b)

Figure 6.11. *Distance images of prior reflectance with respect to ELM estimated reflectance (a) for Run 06. The distance value at each pixel is equal to the length of the vector difference between the pixel spectra of the two images at that pixel. Brighter distance image pixels have longer distances and highlight changes since the prior and registration errors in (a). The distance image between Combined SPIRE and ELM estimated* ***r*** *for the same run is shown in (b), where misregistration noise did not affect the results.*

6.6.2 Prior, Sensor, and Calibration Noise

We shall now derive the effects of sensor, calibration, and prior noise sources on the Combined SPIRE Case IV algorithm and give some examples of their effects. The Combined SPIRE Case IV algorithm is nearly identical to the Spectral SPIRE Case B algorithm with the exception that the Spatial SPIRE Case III algorithm is used to eliminate the log **m** noise instead the Spectral SPIRE Case C. In other words, both algorithms use the same technique for eliminating the additive noise **a**. Because of this, the derivation of the effects of sensor, calibration, and prior noise for the Combined Case IV algorithm is identical to that for the Spectral Case D algorithm as developed in Section 5.3.3, up to the point where the Combined Case III algorithm is about to be applied to remove the log **m** noise rather than the Spectral Case C. The remaining derivation is similar to the last portion of the same derivation for Spatial

SPIRE in Section 3.4.2, and we will again limit ourselves to assuming that the spatial filtering takes the form of a mean filter.

We begin with equation (6.13) which corresponds to (5.46):

$$\log\left(\mathbf{r} \odot \hat{\mathbf{m}}\right) = \log \mathbf{r} + \log \mathbf{m} + \log \mathbf{c}_{gain} + \log \mathbf{e}_{\text{sanl}} \tag{6.13}$$

We now apply the Combined filter $\text{CF}_{\alpha\text{-log}\mathbf{m}}$ to (6.13), which begins by rotating the estimate of $\log\left(\mathbf{r} \odot \hat{\mathbf{m}}\right)$ to into $\log \mathbf{m}$ PC space. If empirical ensembles were used to determine $\mathbf{\Phi}_{\log \mathbf{m}}$, then we can assume that all of the constant $\log \mathbf{c}_{gain}$ will be collected into the lowest $pc_{co-\log \mathbf{m}}$ PCs along with $\log \mathbf{m}$. Next we apply a high-pass spatial filtering operation (subtraction of the mean) to remove $\log \mathbf{m}$ and which will also remove the $\log \mathbf{c}_{gain}$ term. The result of this filtering operation is:

$$\text{CF}_{\alpha-\log \mathbf{m}}\left(\log\left(\mathbf{r} \odot \hat{\mathbf{m}}\right)\right) = \log \mathbf{r} - \text{mean}\left\{\Delta \log \mathbf{r}\right\} + \log \mathbf{e}_{\text{sanl-HPC}} - \text{mean}\left\{\log \mathbf{e}_{\text{sanl-LPC}}\right\} \tag{6.14}$$

Where noise $\log \mathbf{e}_{\text{sanl-HPC}}$ was present only in the high log $\mathbf{m}$ PCs, while $\log \mathbf{e}_{\text{sanl-LPC}}$ was present only in the low log $\mathbf{m}$ PCs so that:

$$\log \mathbf{e}_{\text{sanl-HPC}} = \text{PCF}_{\text{H}-\log \mathbf{m}}\left(\log \mathbf{e}_{\text{sanl}}\right) \tag{6.15}$$

$$\log \mathbf{e}_{\text{sanl-LPC}} = \text{PCF}_{\text{L}-\log \mathbf{m}}\left(\log \mathbf{e}_{\text{sanl}}\right) \tag{6.16}$$

As in Section 5.6.2, we assume that the prior has a random Gaussian noise vector $\mathbf{e}_{\text{p}}$ with mean vector $\boldsymbol{\mu}_{\text{p}}$ and covariance $\boldsymbol{\Lambda}_{\text{p}}$: $\mathbf{e}_{\text{p}} \sim N(\boldsymbol{\mu}_{\text{p}}, \boldsymbol{\Lambda}_{\text{p}})$. We need to restore $\text{mean}\left\{\Delta \log \mathbf{r}\right\}$ and do so by applying the $\text{CF}_{\alpha\text{-log}\mathbf{m}}$ filter to the log prior reflectance and its associated prior noise:

$$\text{mean}\left\{\hat{\Delta}\log \mathbf{r}\right\} = \text{CF}_{\beta-\log \mathbf{m}}\left(\log \mathbf{r}_p + \log \mathbf{e}_{\mathbf{pnl}}\right) \tag{6.17}$$

Where the log operation forces us to define a second non-linear noise term:

$$\log \mathbf{e}_{\mathbf{pnl}} = \log\left(\mathbf{r}_p + \mathbf{e}_{\mathbf{p}}\right) - \log \mathbf{r}_p \tag{6.18}$$

So that:

$$\begin{aligned} \text{mean}\left\{\hat{\Delta}\log \mathbf{r}\right\} &= \text{mean}\left\{\Delta \log \mathbf{r}_p\right\} + \text{mean}\left\{\Delta \log \mathbf{e}_{\mathbf{pnl}}\right\} \\ &= \text{mean}\left\{\Delta \log \mathbf{r}_p\right\} + \text{mean}\left\{\log \mathbf{e}_{\mathbf{pnl}\text{-LPC}}\right\} \end{aligned} \tag{6.19}$$

Adding (6.19) to (6.14) to estimate $\log \mathbf{r}$ we obtain:

$$\begin{aligned} \hat{\log \mathbf{r}} = \log \mathbf{r} &- \text{mean}\left\{\Delta \log \mathbf{r}\right\} + \log \mathbf{e}_{\mathbf{sanl}\text{-HPC}} - \text{mean}\left\{\log \mathbf{e}_{\mathbf{sanl}\text{-LPC}}\right\} \\ &+ \text{mean}\left\{\Delta \log \mathbf{r}_p\right\} + \text{mean}\left\{\log \mathbf{e}_{\mathbf{pnl}\text{-LPC}}\right\} \end{aligned} \tag{6.20}$$

And assuming that $\text{mean}\left\{\Delta \log \mathbf{r}_p\right\} = \text{mean}\left\{\Delta \log \mathbf{r}\right\}$ we obtain:

$$\hat{\log \mathbf{r}} = \log \mathbf{r} + \log \mathbf{e}_{\mathbf{sanl}\text{-HPC}} - \text{mean}\left\{\log \mathbf{e}_{\mathbf{sanl}\text{-LPC}}\right\} + \text{mean}\left\{\log \mathbf{e}_{\mathbf{pnl}\text{-LPC}}\right\} \tag{6.21}$$

We can now define a new noise term:

$$\log \mathbf{e}_{\mathbf{sanl}\text{-mod}} = \log \mathbf{e}_{\mathbf{sanl}\text{-HPC}} - \text{mean}\left\{\log \mathbf{e}_{\mathbf{sanl}\text{-LPC}}\right\} + \text{mean}\left\{\log \mathbf{e}_{\mathbf{pnl}\text{-LPC}}\right\} \tag{6.22}$$

Which is simply $\log \mathbf{e}_{\mathbf{sanl}}$ but with its mean in the low log $\mathbf{m}$ PCs replaced by those of

$\log \mathbf{e}_{\text{pnl}}$ in those same PCs. Equation (6.21) becomes:

$$\hat{\log \mathbf{r}} = \log \mathbf{r} + \log \mathbf{e}_{\text{sanl-mod}} \tag{6.23}$$

Taking the exponential yields:

$$\begin{aligned} \hat{\mathbf{r}} &= \exp\left(\log \mathbf{r} + \log \mathbf{e}_{\text{sanl-mod}}\right) \\ &= \mathbf{r} \odot \mathbf{e}_{\text{sanl-mod}} \end{aligned} \tag{6.24}$$

We can define an overall error vector as follows:

$$\mathbf{e}_{\hat{\mathbf{r}}} = \hat{\mathbf{r}} - \mathbf{r} = \mathbf{r} \odot \mathbf{e}_{\text{sanl-mod}} - \mathbf{r} = \mathbf{r} \odot \left(\mathbf{e}_{\text{sanl-mod}} - 1\right) \tag{6.25}$$

Where the noise term is defined as:

$$\begin{aligned} \mathbf{e}_{\text{sanl-mod}} &= \exp\left(\log \mathbf{e}_{\text{sanl-HPC}} - \text{mean}\left\{\log \mathbf{e}_{\text{sanl-LPC}}\right\} + \text{mean}\left\{\log \mathbf{e}_{\text{pnl-LPC}}\right\}\right) \\ &= \exp\left(\begin{array}{l} \text{PCF}_{\text{H-log}\mathbf{m}}\left(\log \mathbf{e}_{\text{sanl}}\right) - \text{mean}\left\{\text{PCF}_{\text{L-log}\mathbf{m}}\left(\log \mathbf{e}_{\text{sanl}}\right)\right\} \\ + \text{mean}\left\{\text{PCF}_{\text{L-log}\mathbf{m}}\left(\log \mathbf{e}_{\text{pnl}}\right)\right\} \end{array}\right) \\ &= \exp\left(\begin{array}{l} \text{PCF}_{\text{H-log}\mathbf{m}}\left(\left(\begin{array}{l} \log\left(\mathbf{c}_{gain} \odot \mathbf{r} \odot \mathbf{m} + \mathbf{e}_{\mathbf{a}} + \mathbf{s}_{\text{mod}}\right) \\ -\log \mathbf{c}_{gain} + \log \mathbf{r} + \log \mathbf{m} \end{array}\right)\right) \\ -\text{mean}\left\{\text{PCF}_{\text{L-log}\mathbf{m}}\left(\left(\begin{array}{l} \log\left(\mathbf{c}_{gain} \odot \mathbf{r} \odot \mathbf{m} + \mathbf{e}_{\mathbf{a}} + \mathbf{s}_{\text{mod}}\right) \\ -\log \mathbf{c}_{gain} + \log \mathbf{r} + \log \mathbf{m} \end{array}\right)\right)\right\} \\ + \text{mean}\left\{\text{PCF}_{\text{L-log}\mathbf{m}}\left(\log \mathbf{e}_{\text{pnl}}\right)\right\} \end{array}\right) \end{aligned} \tag{6.26}$$

While (6.26) appears formidable, its behavior is very similar to effects of the two noise terms derived for Spectral SPIRE. The reflectance error due to sensor noise with respect to SNR

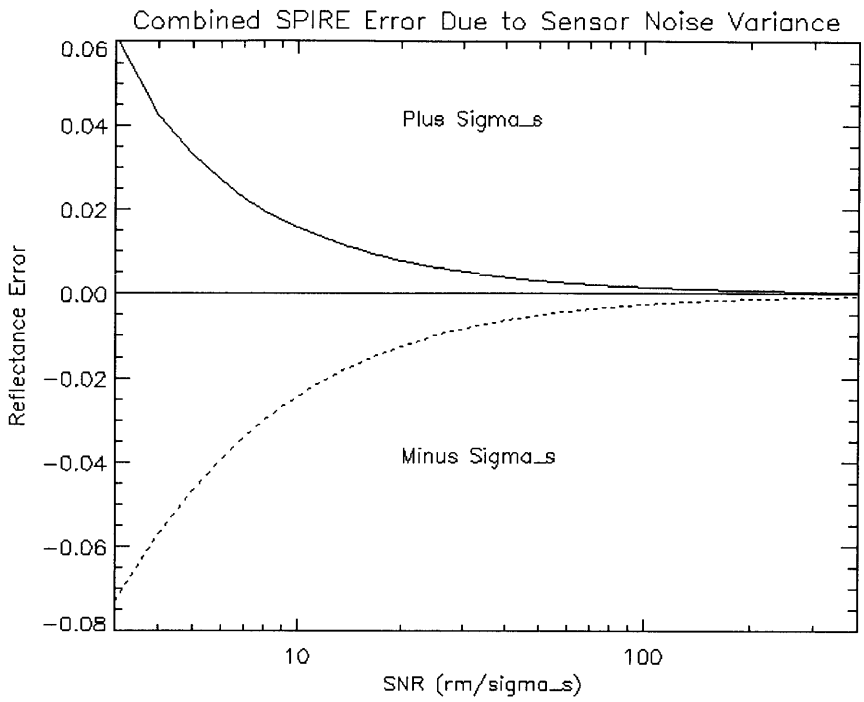


Figure 6.12. *Combined SPIRE reflectance error at a single pixel of reflectance r=0.20 caused by sensor noise standard deviation σ_s (sigma_s) versus signal-to-noise ratio (SNR=rm/σ_s) in the multiplicative noise only case (***a****=0). The effect of the sensor noise is non-linear and plots for the noise s=+σ_s and s=-σ_s are shown. A value of m=600 Watts/m^2/sr/μm was used at all pixels. Effects of sensor noise can be overcome by increasing SNR.*

behaves nearly identically to the Spectral SPIRE sensor error term plotted in Figure in 5.11 and is plotted in Figure 6.12. The plot is for the error at a single pixel of reflectance r=0.20 caused by sensor noise standard deviation σ_s (sigma_s) versus signal-to-noise ratio (SNR=rm/σ_s) in the multiplicative noise only case (**a**=0). A value of m=600 Watts/m^2/sr/μm was used at all pixels, and we see that the effects of sensor noise can be overcome by increasing SNR.

The reflectance error due to prior noise variance behaves similarly to the prior noise error for the Spectral SPIRE prior noise term plotted in Figure 5.12, but is dependent on only the variance which is passed by the low pass filtering operation. Since we have assumed mean filtering in this section, the Combined SPIRE reflectance error due to prior noise is only due to

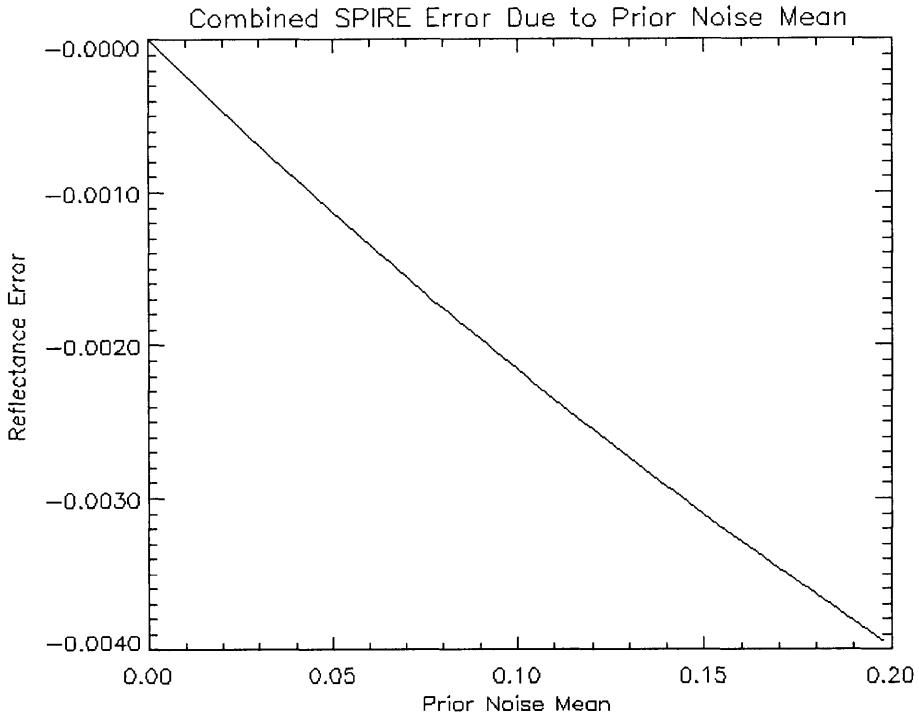
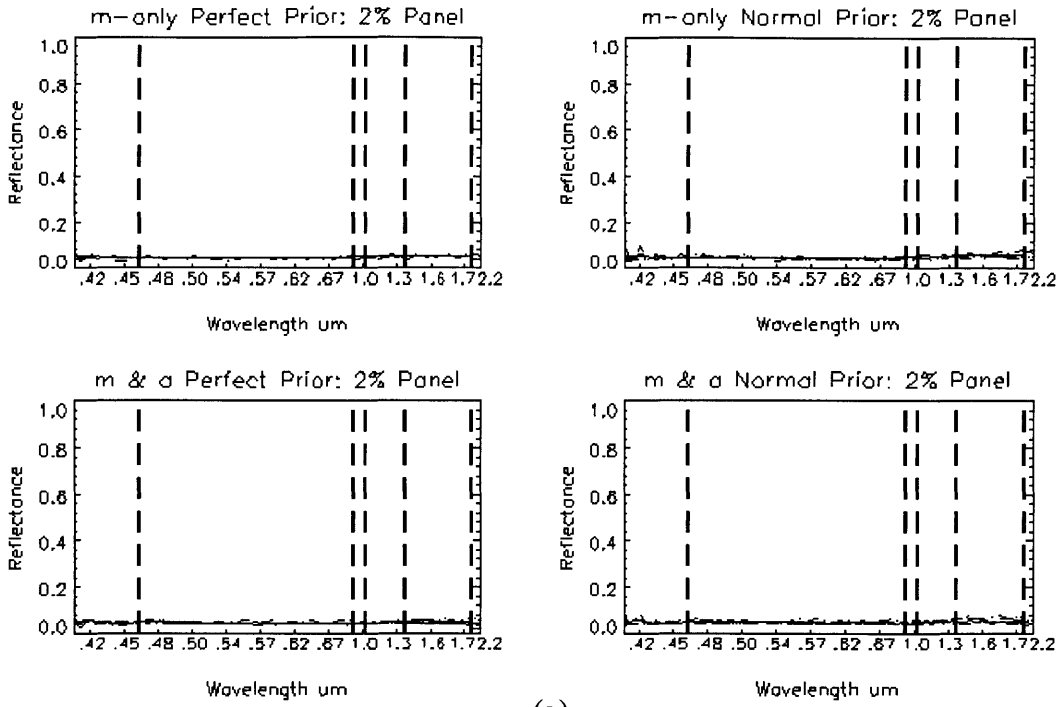


Figure 6.13. *Reflectance error at a single pixel of reflectance r=0.50 caused by prior noise mean μ_p in the multiplicative noise only case (**a**=0). A value of m=600 Watts/m^2/sr/μm was used at all pixels and the prior noise was assumed to be Gaussian with zero mean.*

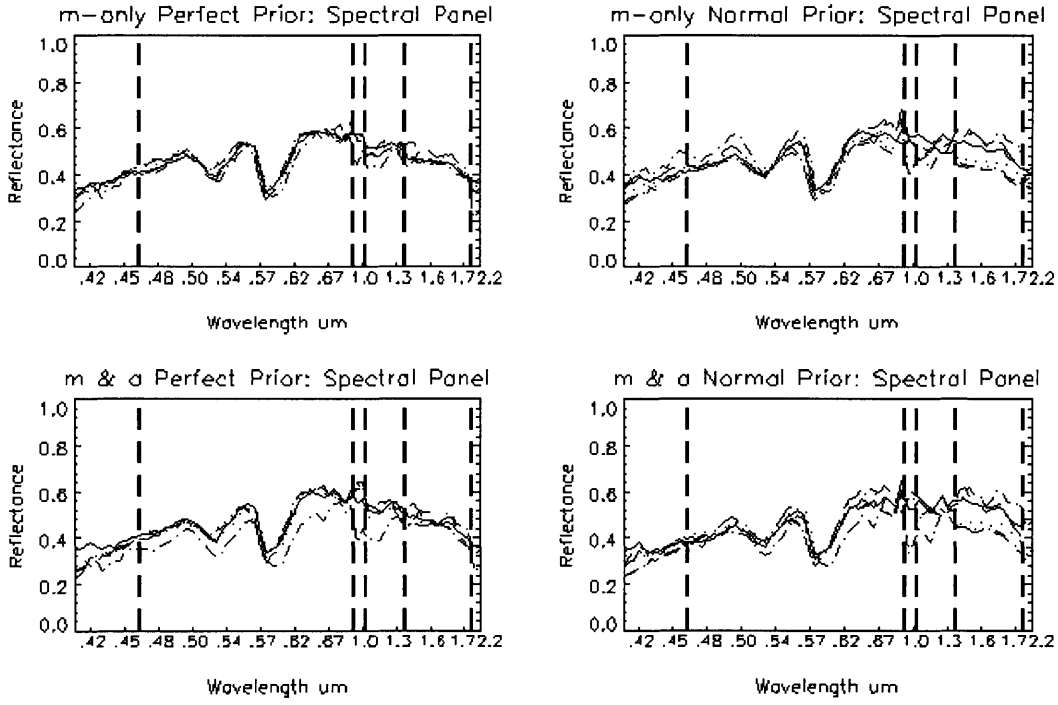
prior mean error. Since the priors used in the algorithm testing in this thesis are derived from an ELM estimates that we can consider ground truth, the prior noise means are all effectively zero and cause no error. To provide guidance for applications where there is non-zero prior mean error, Figure 6.13 plots the Combined SPIRE reflectance error for a single pixel with r=0.50 in the multiplicative noise only case (**a**=0), with a value of m=600 Watts/m^2/sr/μm was used at all pixels.

6.6.3 Perfect versus Normal Priors

To gain insight into the overall effects of these noise sources on the results of Figure 6.6, the Combined SPIRE processing of Section 6.4 was redone using a "perfect prior" consisting



(a)



(b)

Figure 6.14. *Combined SPIRE Case III (m-only) and Case IV (m & a) reflectance estimates for all six runs using a perfect prior and a normal (noisy) prior, for the 2 percent panel (a) and the spectral panel (b).*

of the ELM reflectance estimate for each test cube. In addition, the ELM estimated additive **a** vectors were subtracted to create "**m**-only" test cubes which were also processed with perfect priors and priors generated from Run 07. Figure 6.14(a) shows these four results for the 2 percent panel while 6.14(b) shows these results for the spectral panel; they can be compared directly to Figures 5.13 and 3.14. We see that the introduction of spatial filtering did not reintroduce the high variations at longer wavelengths for low reflectance materials like the 2 percent panel experienced under Spatial SPIRE. Prior noise did slightly affect the Combined SPIRE results for the spectral panel by increasing the variance to more than that of either the Spatial or Spectral results.

6.7 Summary and Conclusions

In this chapter we developed Combined SPIRE algorithms that estimate surface reflectance using a prior reflectance image, prior ensembles of **m** and **a**, and combined PC and spatial frequency filtering to remove the multiplicative and additive noise effects. We developed Combined SPIRE algorithms for the slowly-spatially-varying multiplicative-noise-only case and for the case where a spatially uniform additive noise is present as well. We compared the performance of the Combined SPIRE algorithms to ELM and ATREM on six HYDICE hyperspectral image cubes from the ARM Site data collect.

From these experiments we can draw the following conclusions:

- Performance of Combined SPIRE was very similar to that of ELM on all pixels.
- Performance of Combined SPIRE was consistently far better than ATREM on all pixels.
- Performance of Combined SPIRE was slightly worse than Spatial SPIRE on all

pixels, except low reflectivity materials, where Combined SPIRE performance does not suffer as much during low SNR conditions.

- Performance of Combined SPIRE was slightly worse than Spectral SPIRE on unmodified pixels.
- Combined SPIRE algorithms are computationally less costly than Spatial SPIRE, and only slightly more costly than Spectral.
- As image cube dimensions grow, Combined and Spectral SPIRE computational costs grow slower than Spatial SPIRE computational costs.
- Combined SPIRE algorithms are insensitive to calibration noise if the ensembles of **m** and **a** used are derived empirically from real sensor data.
- The effects of sensor and prior noise can be overcome with adequate SNR.

Based on these conclusions, we were successful in combining Spatial SPIRE's superior performance on modified pixels with Spectral SPIRE's superior performance on unmodified pixels while maintaining Spectral SPIRE's low computational cost. In addition, we were also able to reduce Spatial SPIRE's errors under low SNR conditions and maintain its insensitivity to misregistration noise.

We now move on to Chapter 7 in which we review the taxonomy of SPIRE algorithms developed in this thesis, and explore an example classification application using the reflectance estimates from SPIRE, ELM, and ATREM algorithms. We also develop a Selective SPIRE technique which chooses between different SPIRE reflectance estimates to maximize classification performance.

Chapter 7

Taxonomy of SPIRE Algorithms and their Use in Applications

7.1 Introduction

In this chapter we review the overall SPIRE algorithm taxonomy and the performance of the individual SPIRE reflectance estimation algorithms developed in this thesis. Based on these results we develop a "Selective" SPIRE technique that utilizes reflectance estimates from different SPIRE algorithms to maximize overall classification performance. We then investigate an example classification application, in which we compare classification performance on reflectance estimates from ELM, ATREM, and SPIRE. We also investigate the use of SPIRE algorithms when no prior reflectance information is available and demonstrate that such pseudo-reflectance SPIRE results can be used successfully in classification and change detection applications, eliminating the need for physics-based or ELM algorithms to generate the prior reflectance image. We then provide final guidance on which SPIRE algorithm to use for different applications, and what type of prior information is required by each.

7.2 Review of SPIRE Algorithm Taxonomy

Before we make suggestions on which SPIRE algorithm to use in different applications, we will review the different SPIRE algorithms available to us. Figure 7.1 is identical to Figure 2.9 and depicts the SPIRE algorithm taxonomy for the algorithms developed and tested in this thesis, organized by the spatial variability of the multiplicative noise m and additive noise a. Spatially varying m and a are assumed to be band-limited to low spatial frequencies as was be discussed in Chapter 3. This figure is repeated here for the reader's convenience.

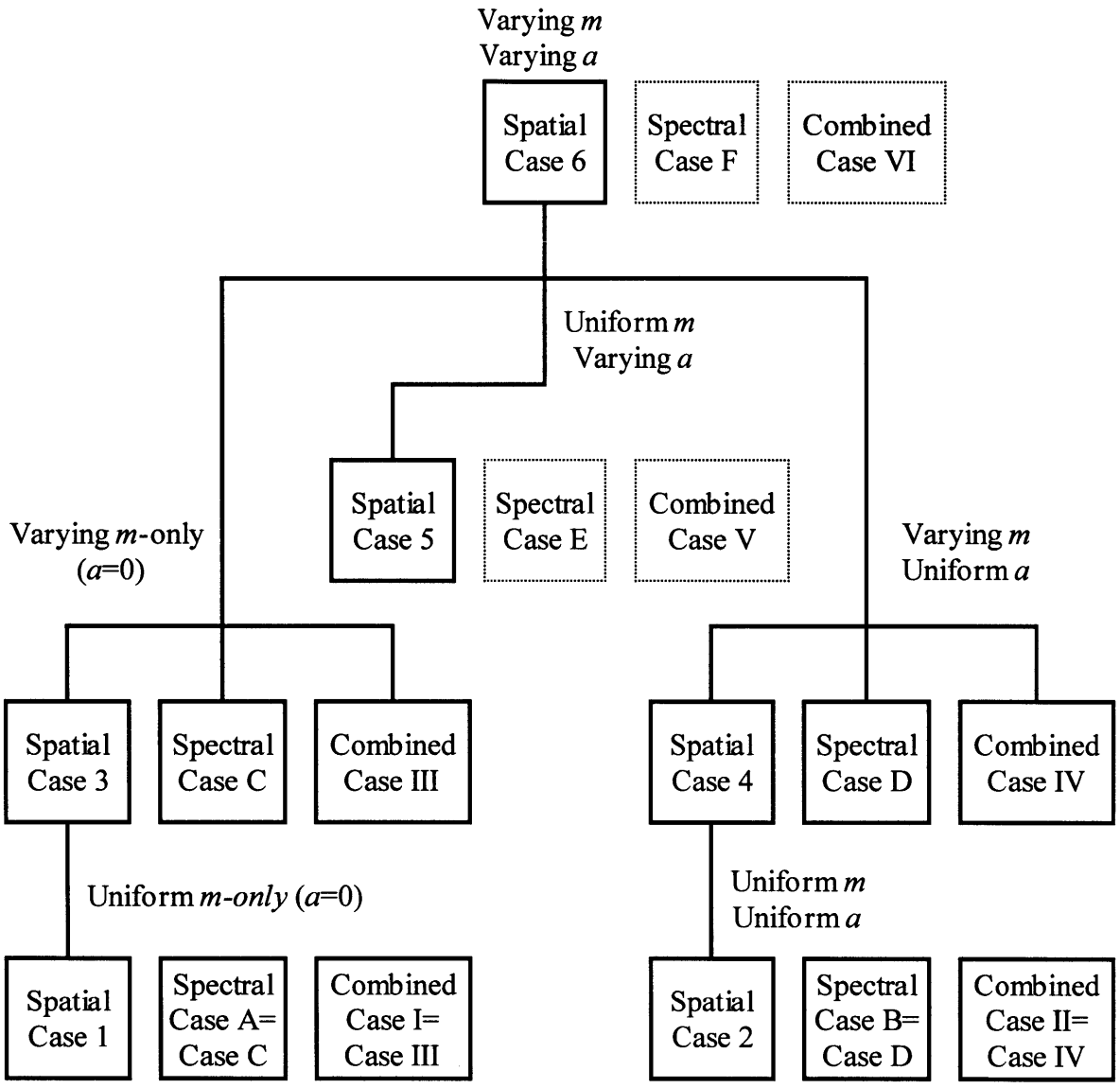


Figure 7.1. *SPIRE algorithm taxonomy based on the spatial variability of the multiplicative noise m and additive noise a. Spatially varying m and a are assumed to be band-limited to lower spatial frequencies. Algorithms outlined in dashed lines were not developed and are suggestions for further work. For Spectral and Combined, algorithms for cases where m is uniform are identical to those for when m is varying.*

At the top of the tree is the most general case considered with spatially varying m and spatially varying a, for which only the Spatial SPIRE Case 6 algorithm was developed. As discussed in Chapter 6, a Combined SPIRE Case VI algorithm should be feasible, though this is

left as a suggestion for further work. A comparable Spectral SPIRE Case F algorithm is certainly a possible area of further work, but as discussed in Chapter 5, the results of this thesis do not provide any guidance for how to proceed.

On the left branch are algorithms that work when the additive noise is zero. The trio of Spatial Case 3, Spectral Case C, and Combined Case III algorithms all handle this case. For the case where the multiplicative noise m is spatially uniform (and additive noise a is zero), a computationally efficient Spatial Case 1 algorithm was developed. On the center branch is the condition where the multiplicative noise m is spatially uniform but the additive noise a is spatially varying, solved by the computationally efficient Spatial SPIRE Case 5 algorithm. This case appears to be an unusual one in that no practical applications have yet to be identified for it, though one can envision certain future robotic applications. On the right branch are algorithms that work when the multiplicative noise m is spatially varying and the additive noise a is spatially uniform, which includes the Spatial Case 4, Spectral Case D, and Combined Case IV algorithms. For the case where the multiplicative noise is spatially uniform as well, the computationally efficient Spatial Case 2 algorithm can be used.

7.3 Review of Individual SPIRE Algorithm Performance

Having reviewed the SPIRE algorithms available to us, it is also appropriate to review and summarize their relative performance. We will concentrate on the Spatial Case 4, Spectral Case D, and Combined Case IV algorithms for comparison purposes, since they are the most useful for airborne remote sensing applications, and were analyzed in detail in this thesis.

Figure 7.2(a) shows a scatter plot of mean and standard deviation performance of Spatial (diamonds), Spectral (triangles), and Combined (squares) SPIRE algorithms for estimation of surface spectral reflectance for all of the 18 unmodified pixel types selected. The horizontal axis

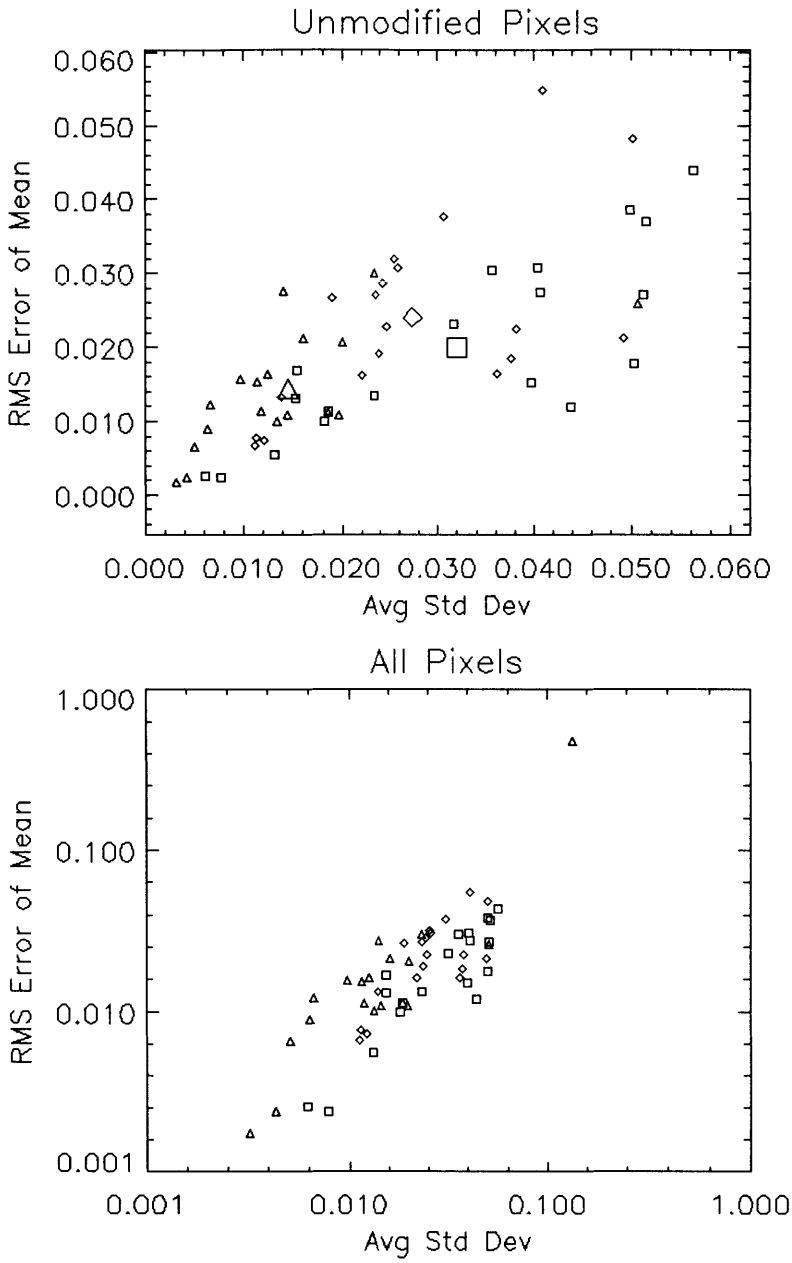


Figure 7.2. *Scatter plot of mean and standard deviation performance of Spatial (diamonds), Spectral (triangles), and Combined (squares) SPIRE algorithms for estimation of surface spectral reflectance for all of the 18 unmodified pixel types selected (a). The horizontal axis represents the average standard deviation over all the spectral channels, where the standard deviation in each channel was calculated over the reflectance estimates of all six Runs (06-31). The vertical axis is the RMS error over all the spectral channels for the mean reflectance estimate minus the mean reflectance estimate of ELM as ground truth. The larger symbols represent the mean of the points plotted with that symbol. We see that Spectral SPIRE has better standard deviation and RMS error performance than Spatial or Combined on unmodified pixels. In (b), the modified pixel is included and log axes are used, showing Spectral's poorer performance on it.*

represents the average standard deviation over all the spectral channels, where the standard deviation in each channel was calculated over the reflectance estimates of all six Runs (06-31). The vertical axis is the RMS error over all the spectral channels for the mean reflectance estimate minus the mean reflectance estimate of ELM as ground truth. We see that in aggregate, Spectral SPIRE has better standard deviation and RMS error performance than Spatial or Combined on unmodified pixels. In Figure 7.2(b), the modified pixel is included, showing Spectral's poorer performance on it.

To gain insight into the relative performance of Spatial, Spectral, and Combined SPIRE on individual pixel types, Figure 7.3 (a)-(s) plots the reflectance estimates for each of the 19 pixel types plotted in Chapters 3, 5, and 6. For each pixel type, the reflectance estimates for all runs are plotted for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM. In addition, a scatter plot of the average channel standard deviation across runs versus the RMS error of the mean of the estimates with respect to the mean for the ELM estimates is shown, and is the same information plotted for SPIRE in Figure 7.2, but for a single pixel. This scatterplot can be interpreted as a measure of the two errors related to the mean and variance of the reflectance estimates by each algorithm, where the best performance is at the (0,0) origin. In Appendix G, Table G.1 lists the average channel standard deviation across all six runs for each Band for all five estimators, and Table G.2 list the same information as a percentage of the channel mean across the six runs.

In Figure 7.3, we see that the scatterplot performance of Spectral SPIRE is often the best on all the unmodified pixels, with the exception of grass and the resolution panel where it is slightly poorer than Spatial SPIRE, and the emissivity panels where it is on par with the Spatial and Combined. In the case of grass and mowed grass, one can argue that these represent pixels

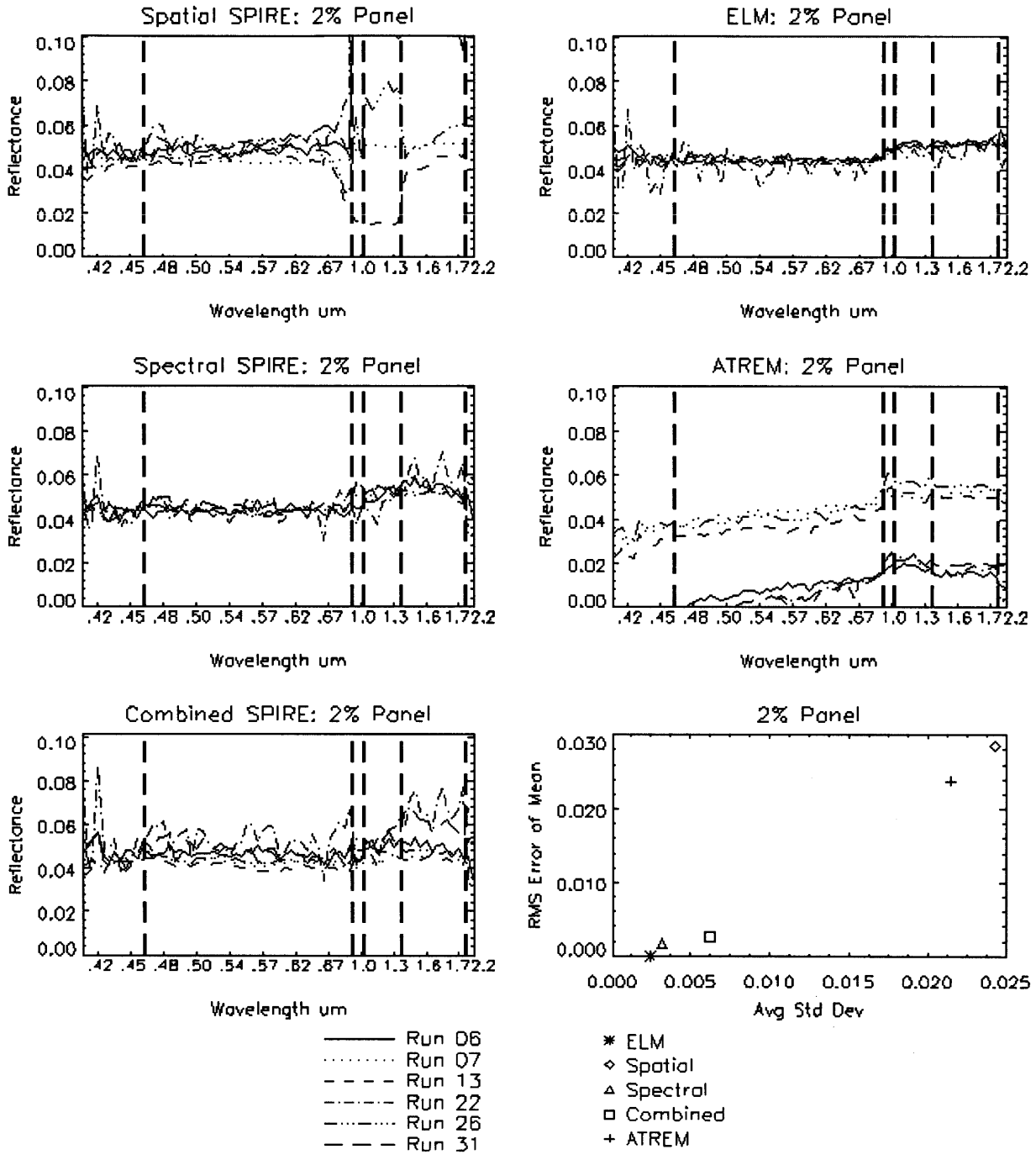


Figure 7.3(a). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the 2 percent panel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

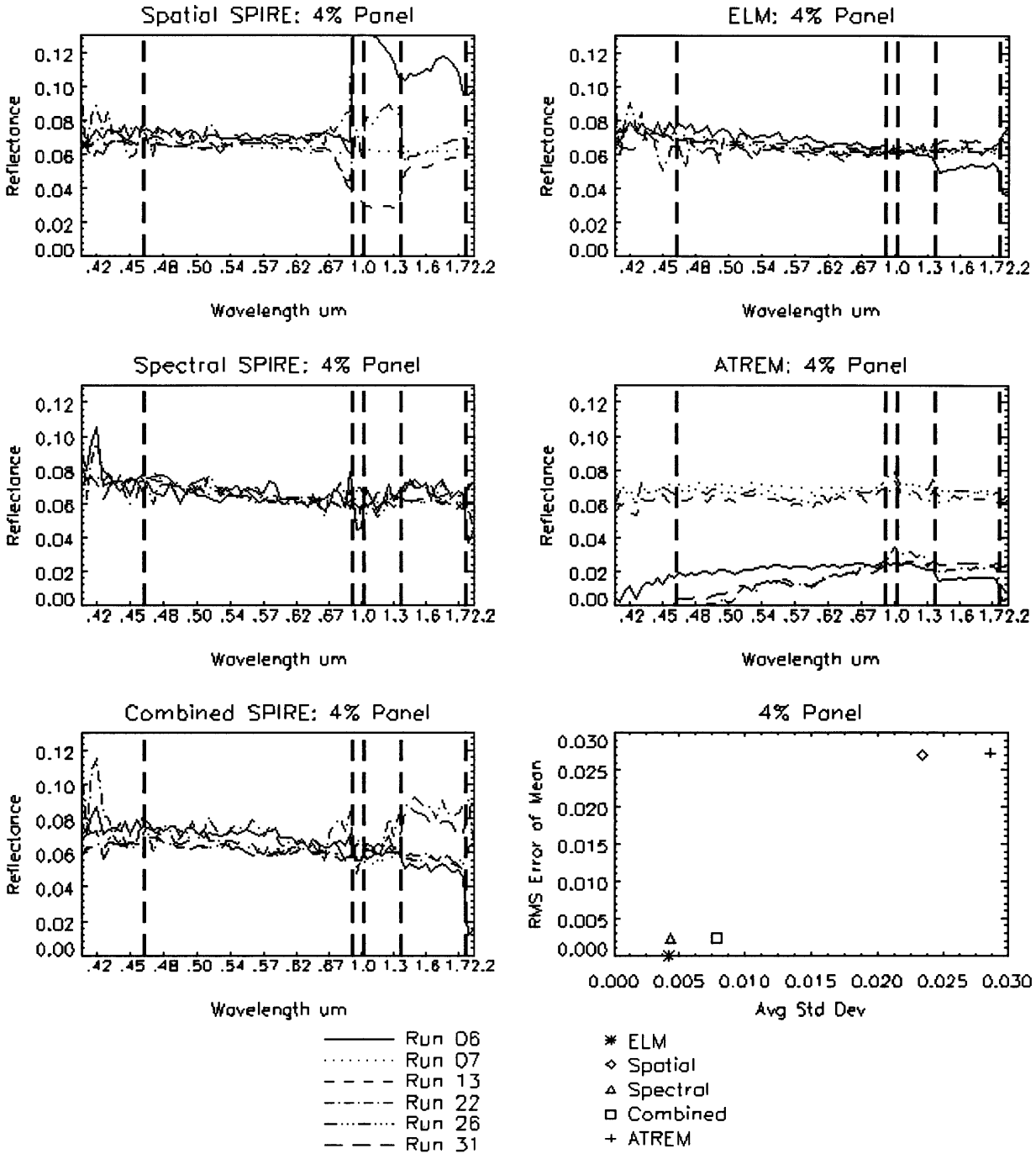


Figure 7.3(b). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the 4 percent panel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

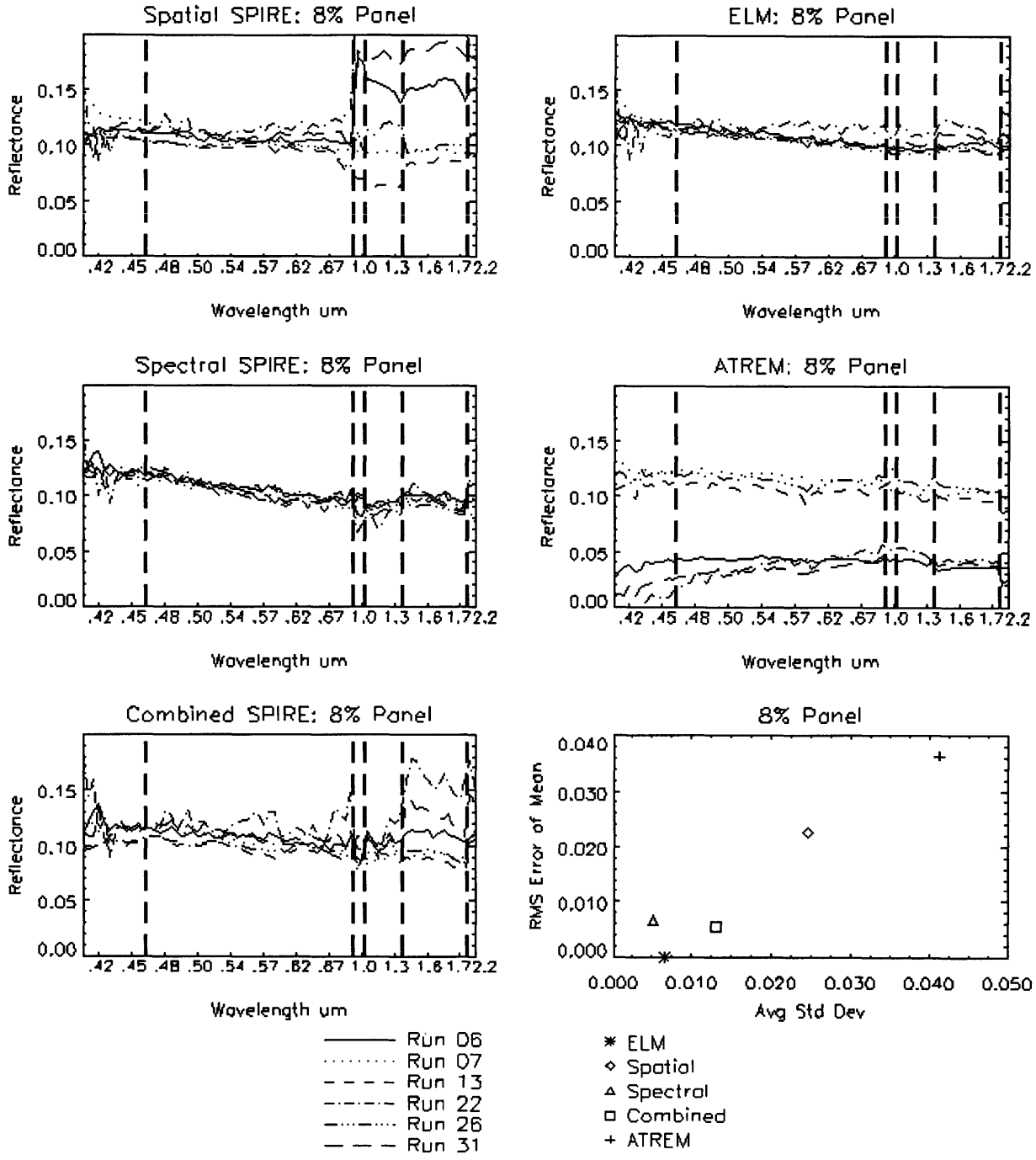


Figure 7.3(c). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the 8 percent panel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

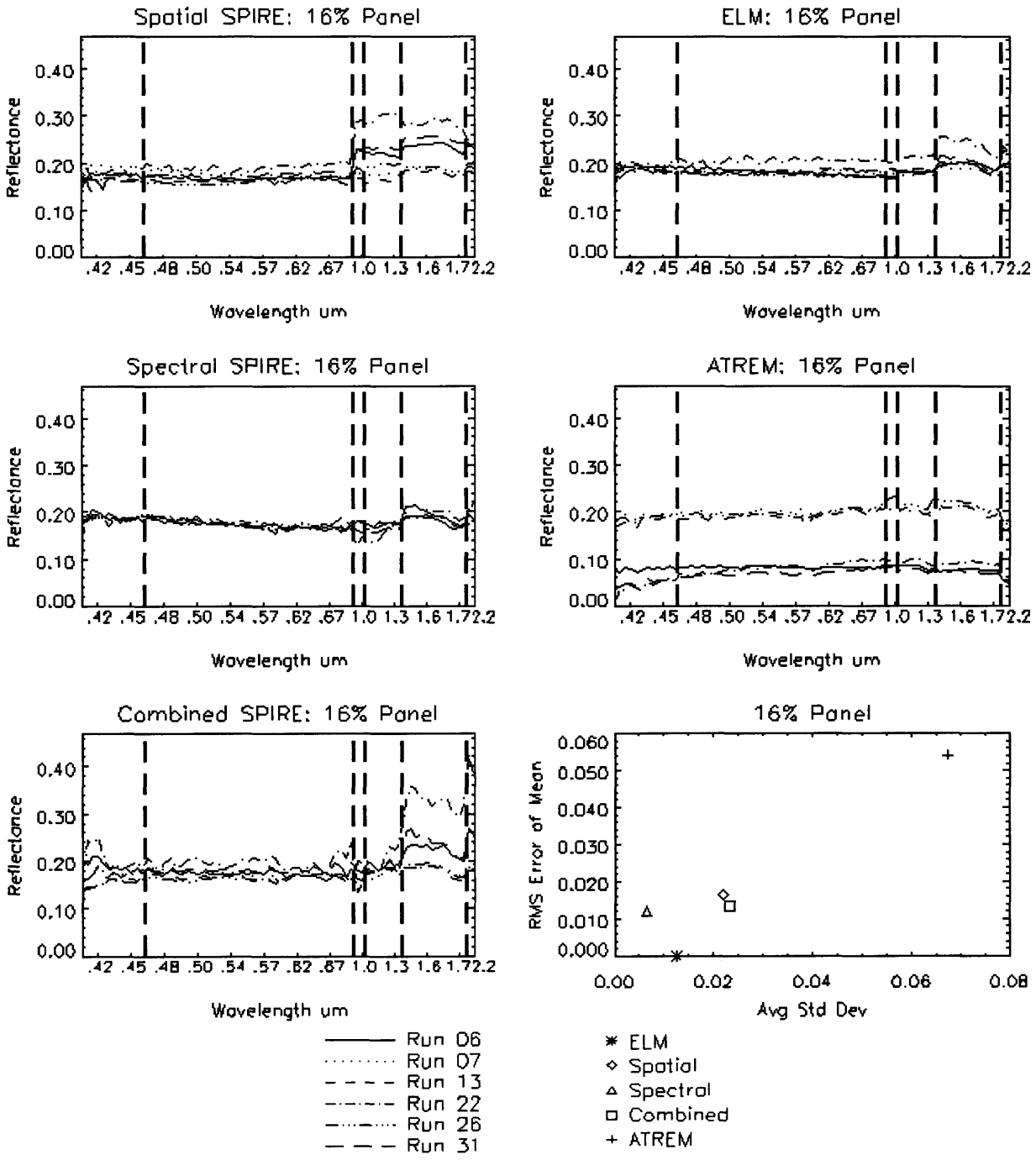


Figure 7.3(d). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the 16 percent panel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

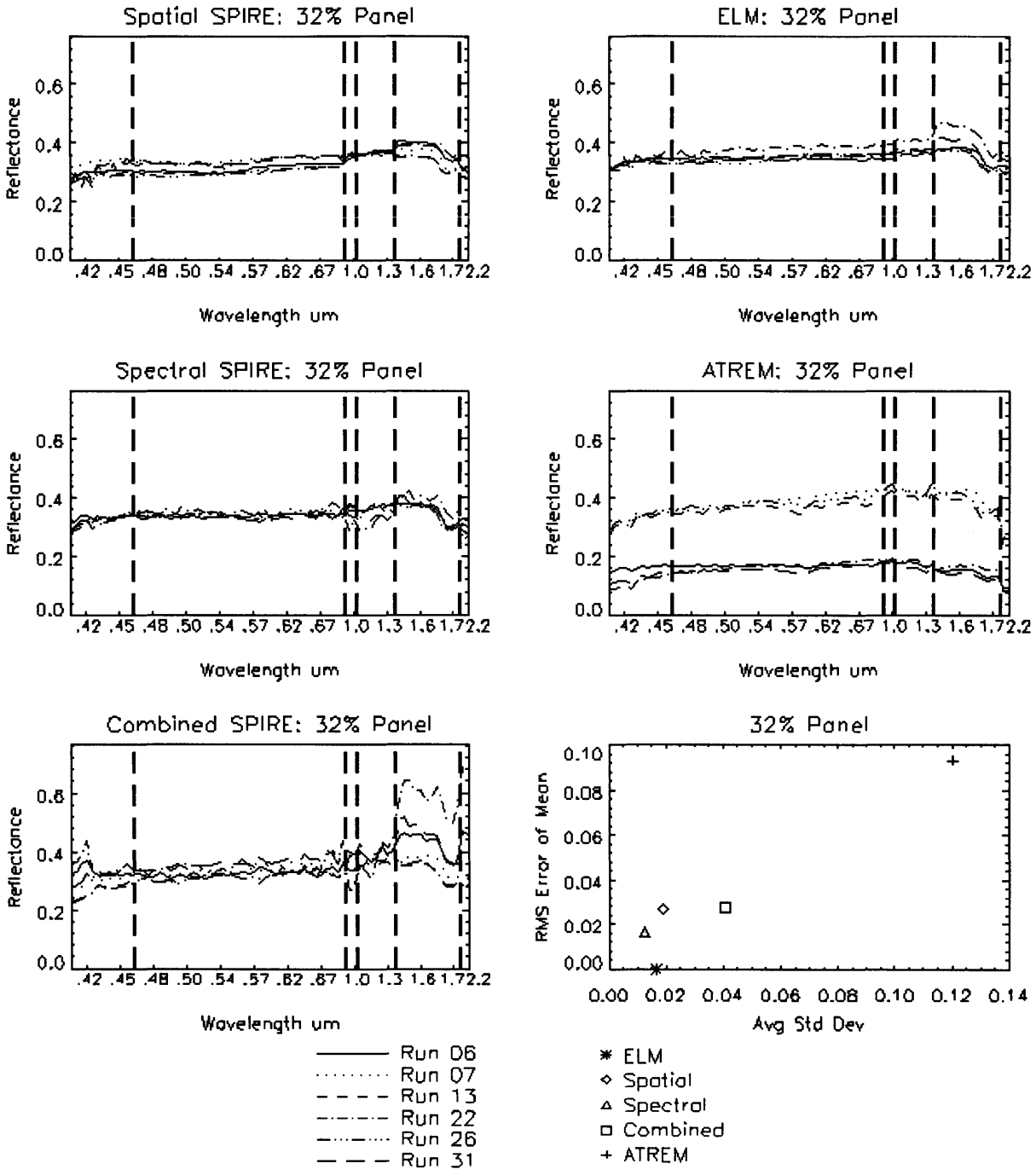


Figure 7.3(e). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the 32 percent panel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

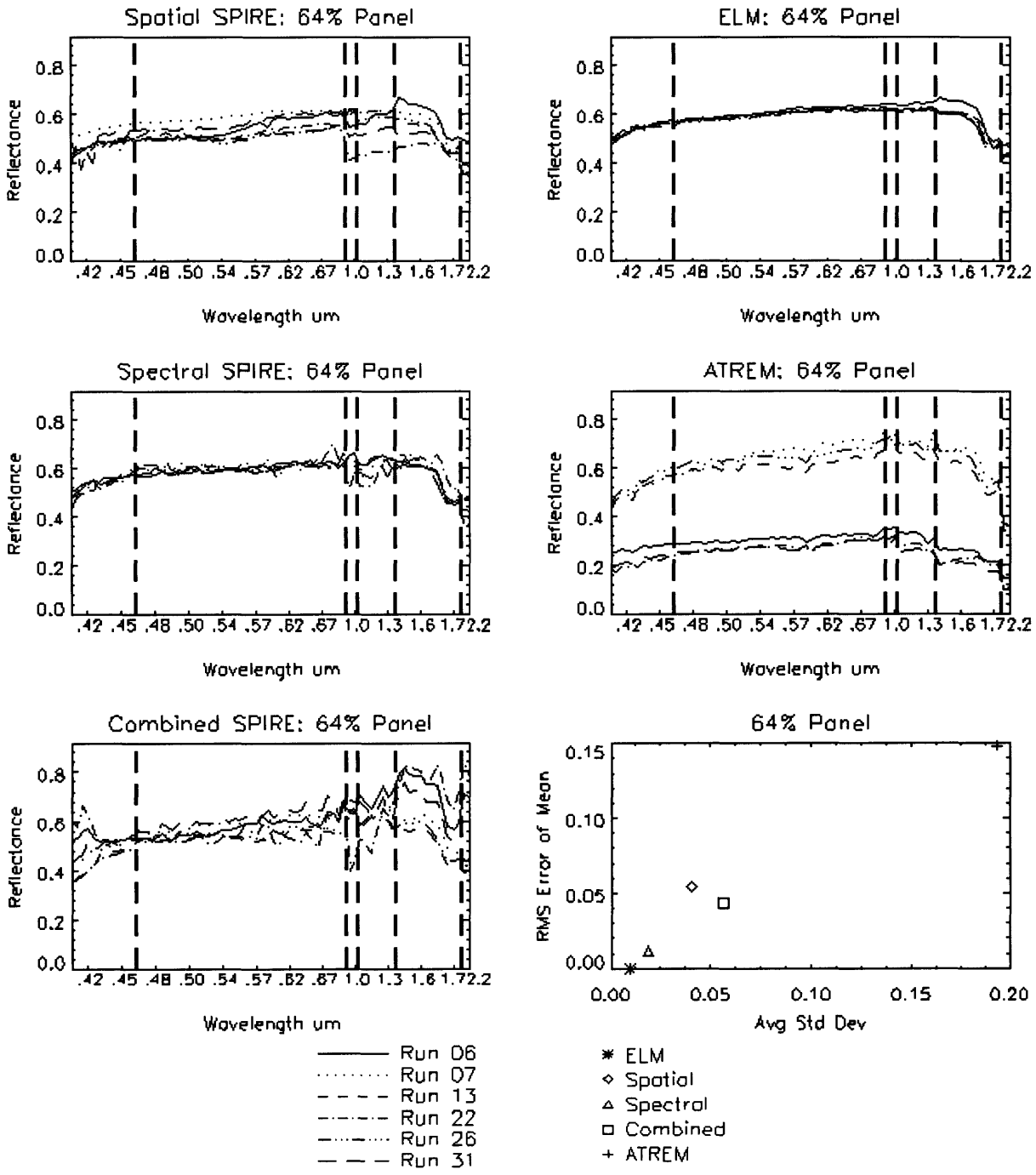


Figure 7.3(f). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the 64 percent panel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

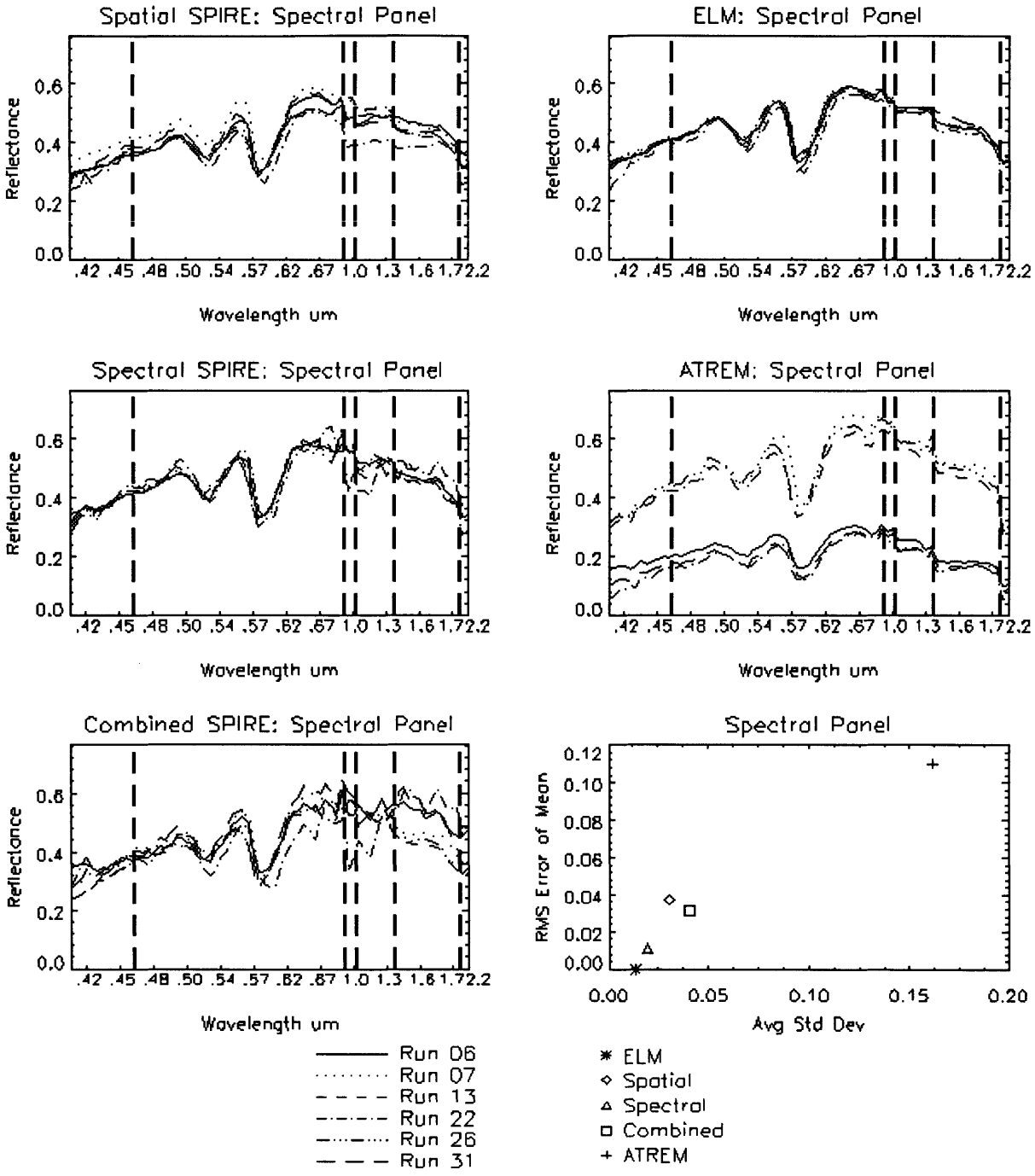


Figure 7.3(g). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the spectral panel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

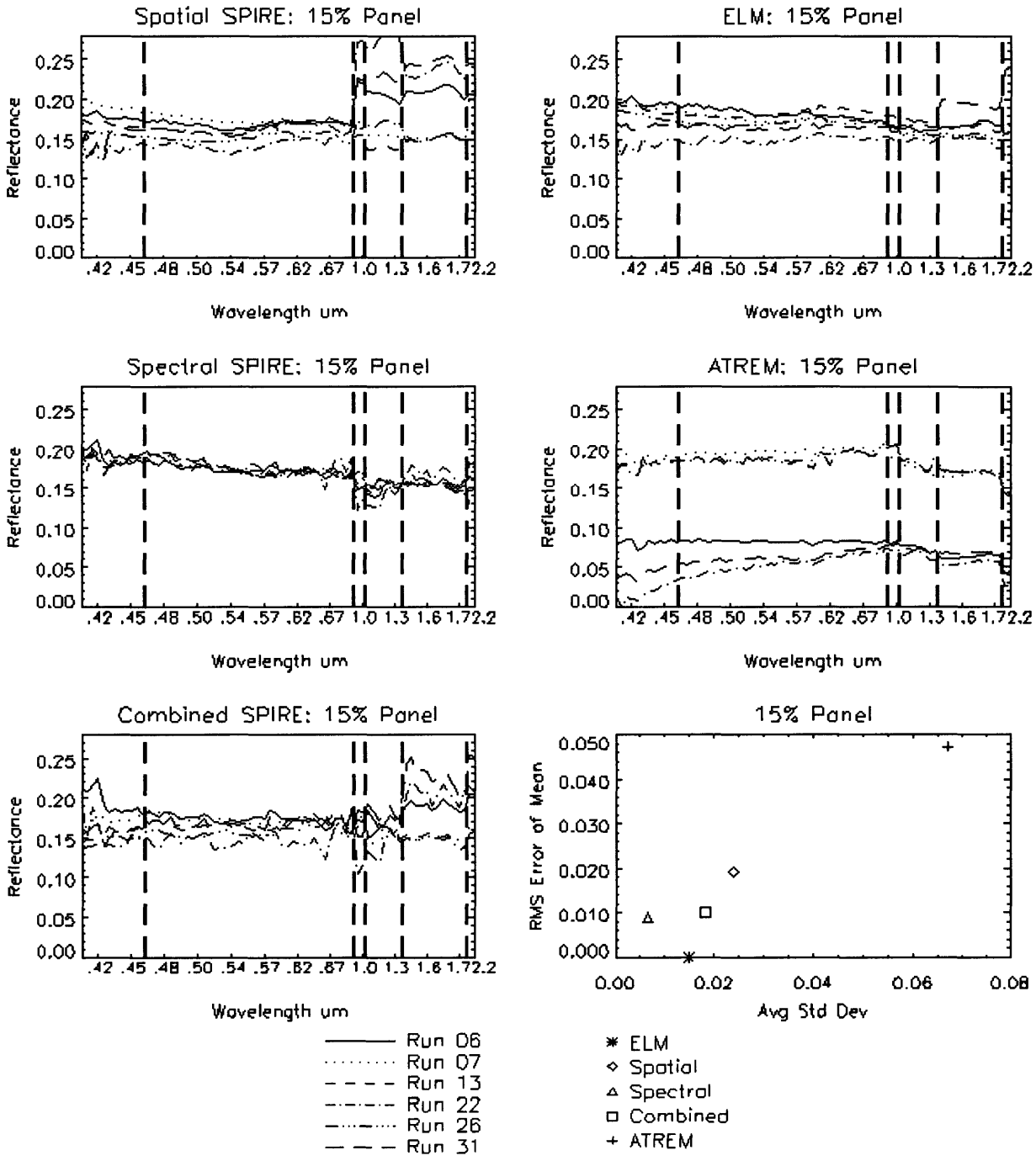


Figure 7.3(h). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the 15 percent panel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

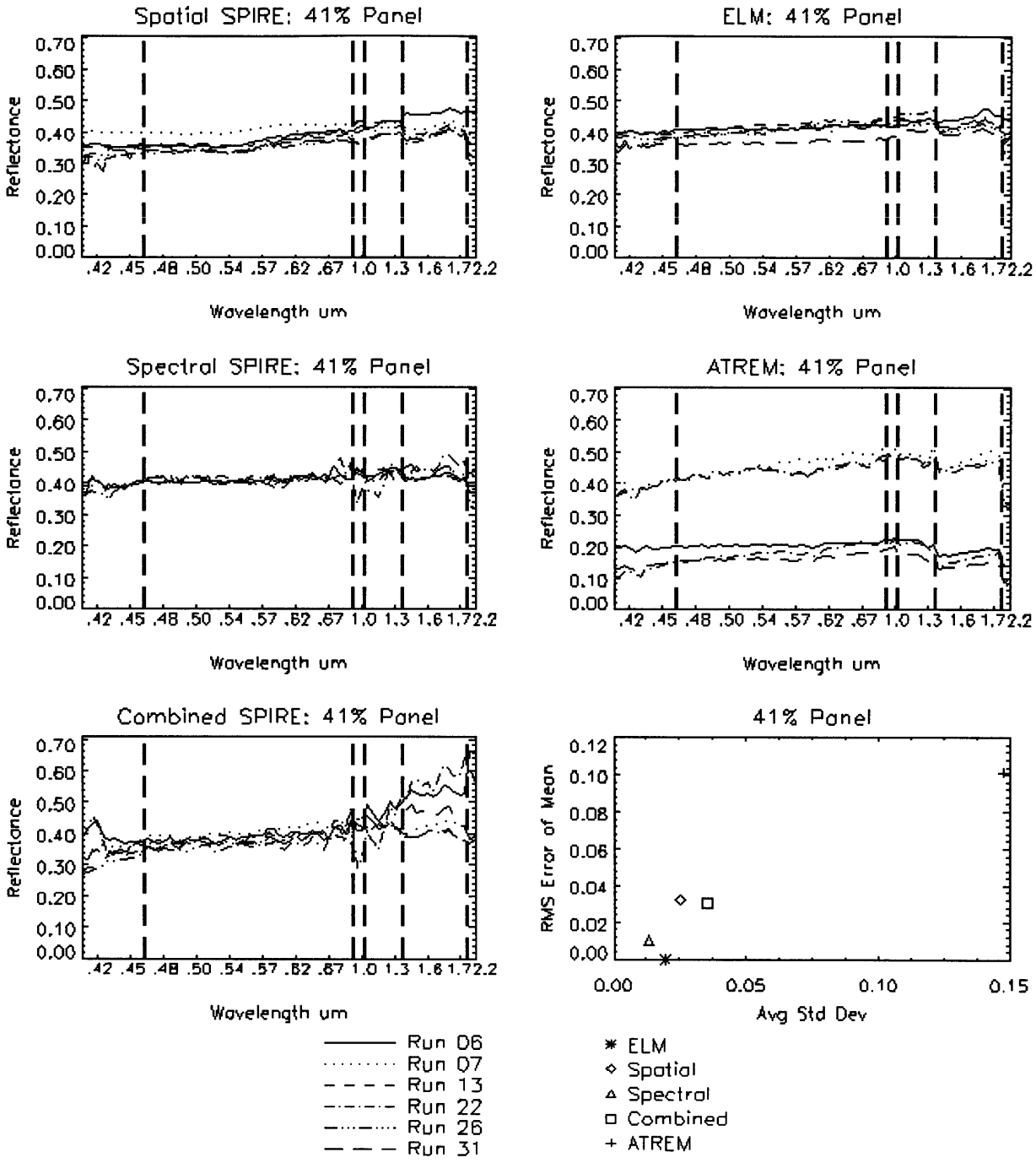


Figure 7.3(i). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the 41 percent panel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

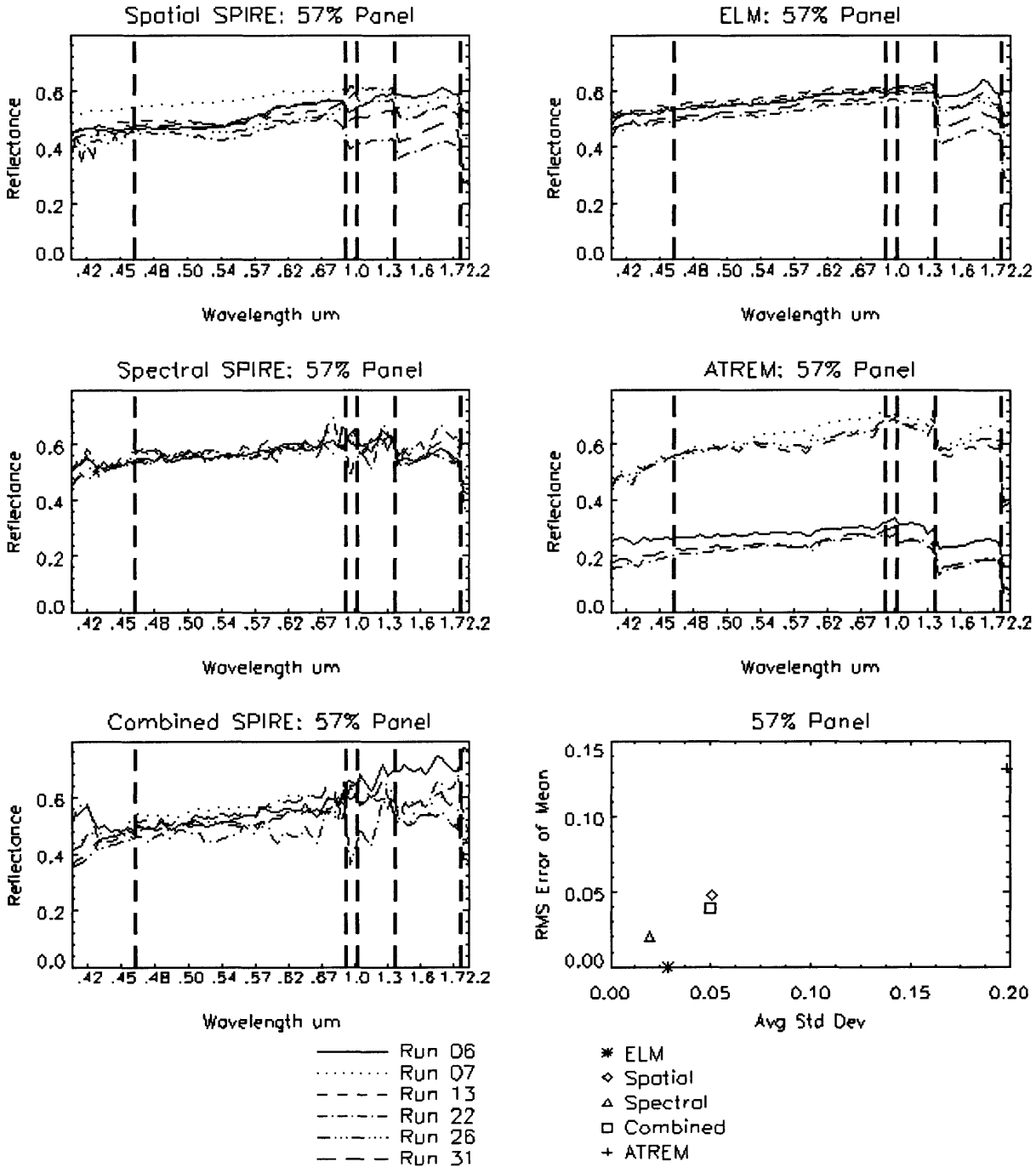


Figure 7.3(j). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the 57 percent panel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

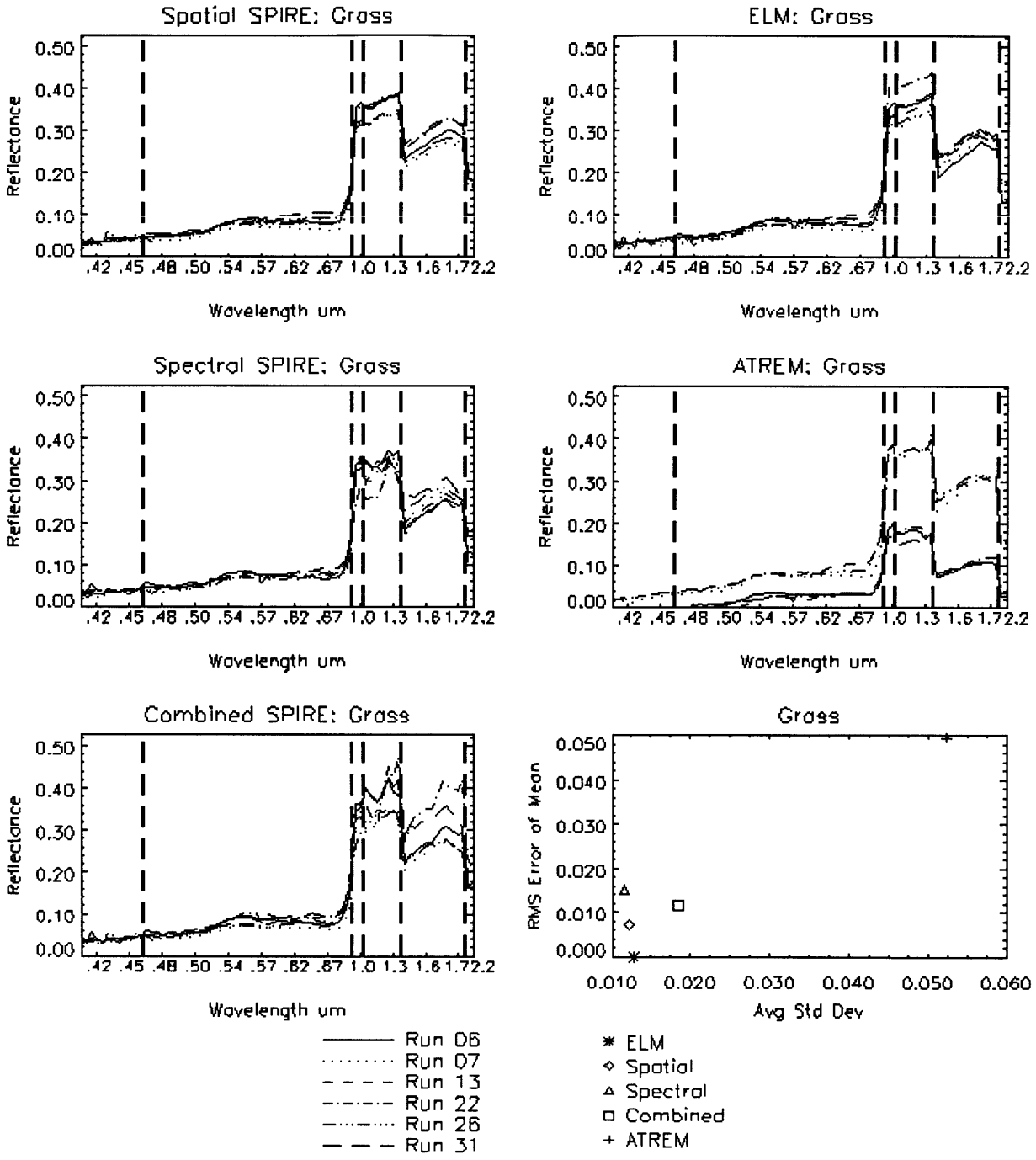


Figure 7.3(k). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the grass, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

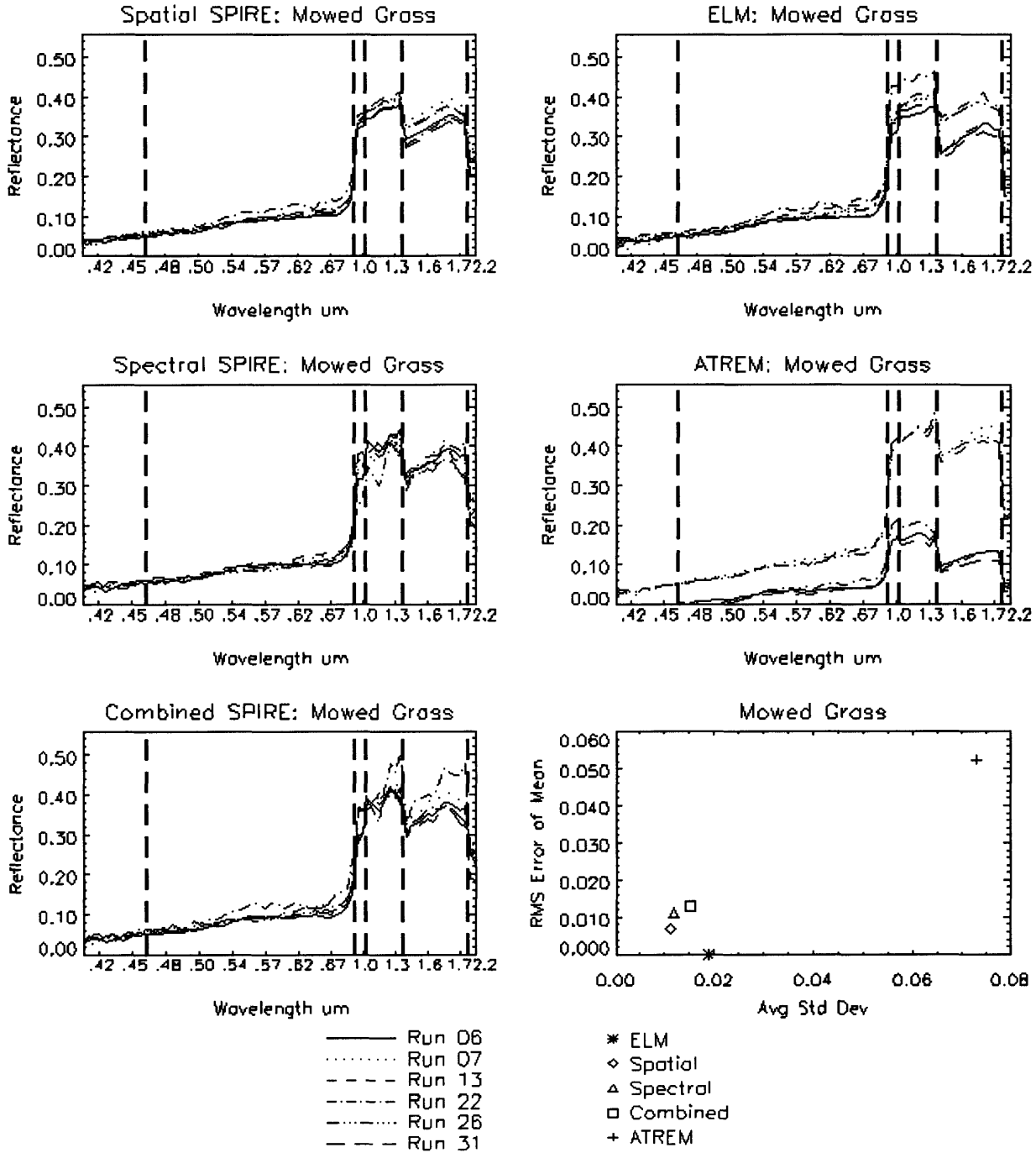


Figure 7.3(l). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for mowed grass, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

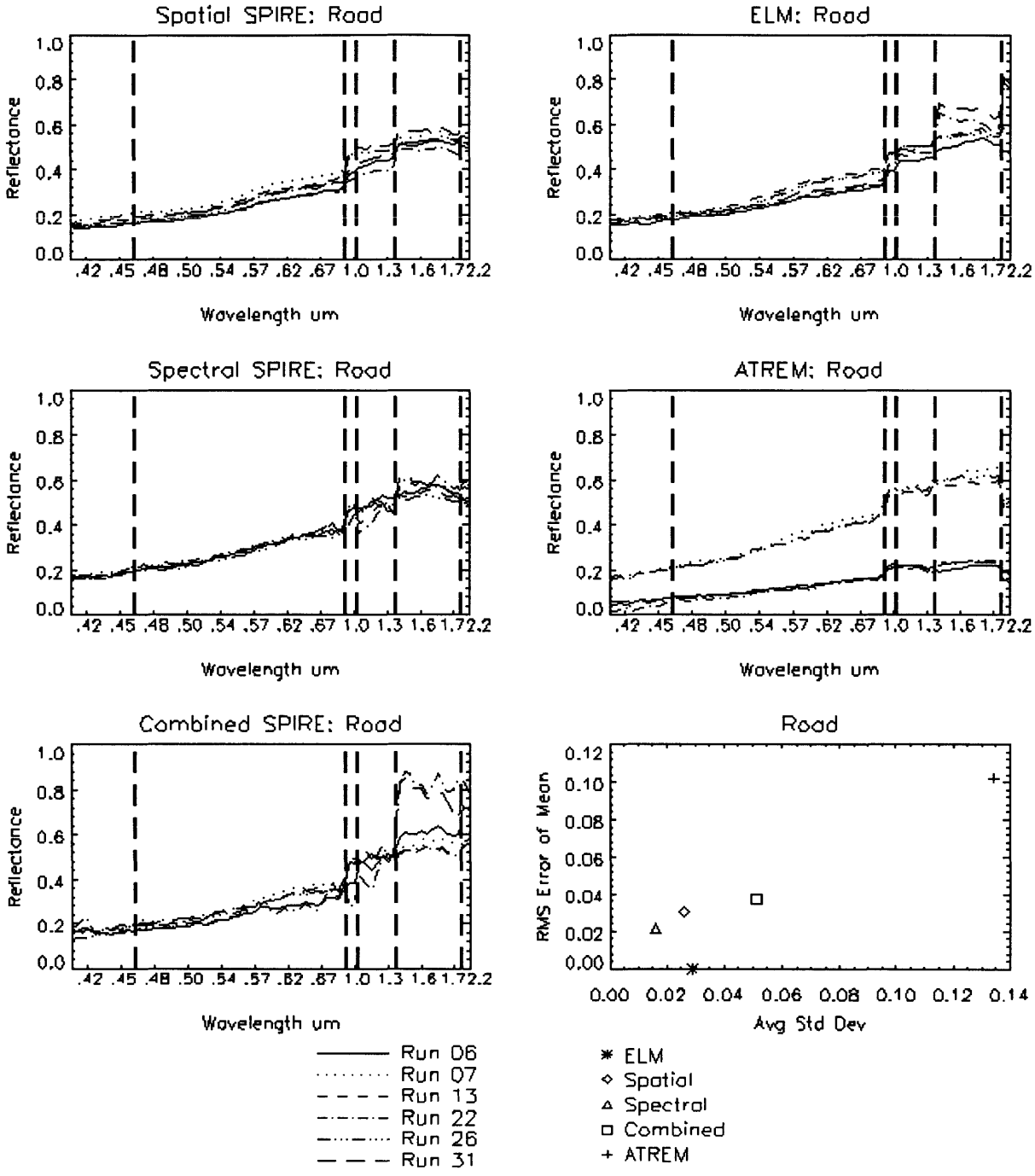


Figure 7.3(m). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the road, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

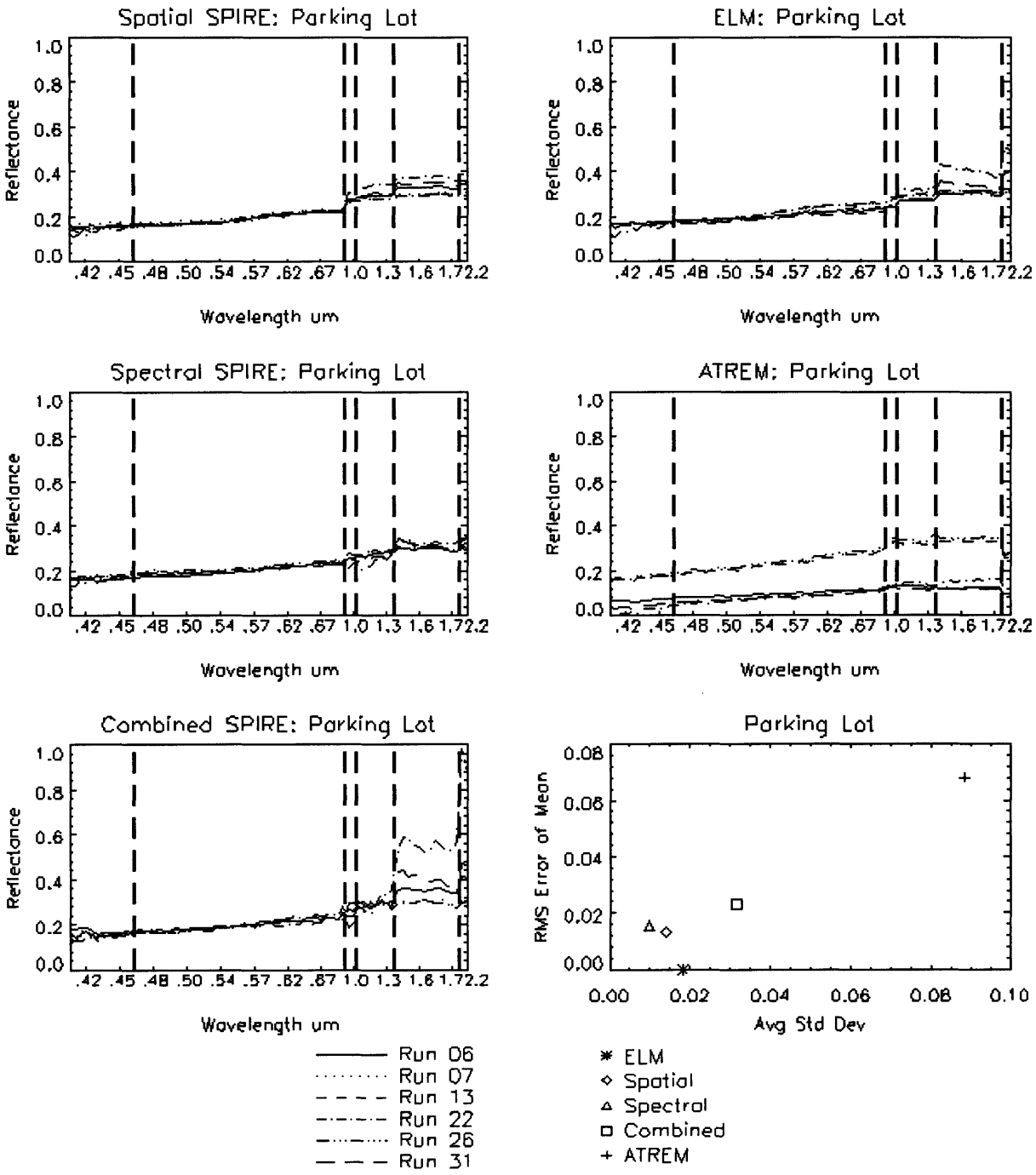


Figure 7.3(n). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the parking lot, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

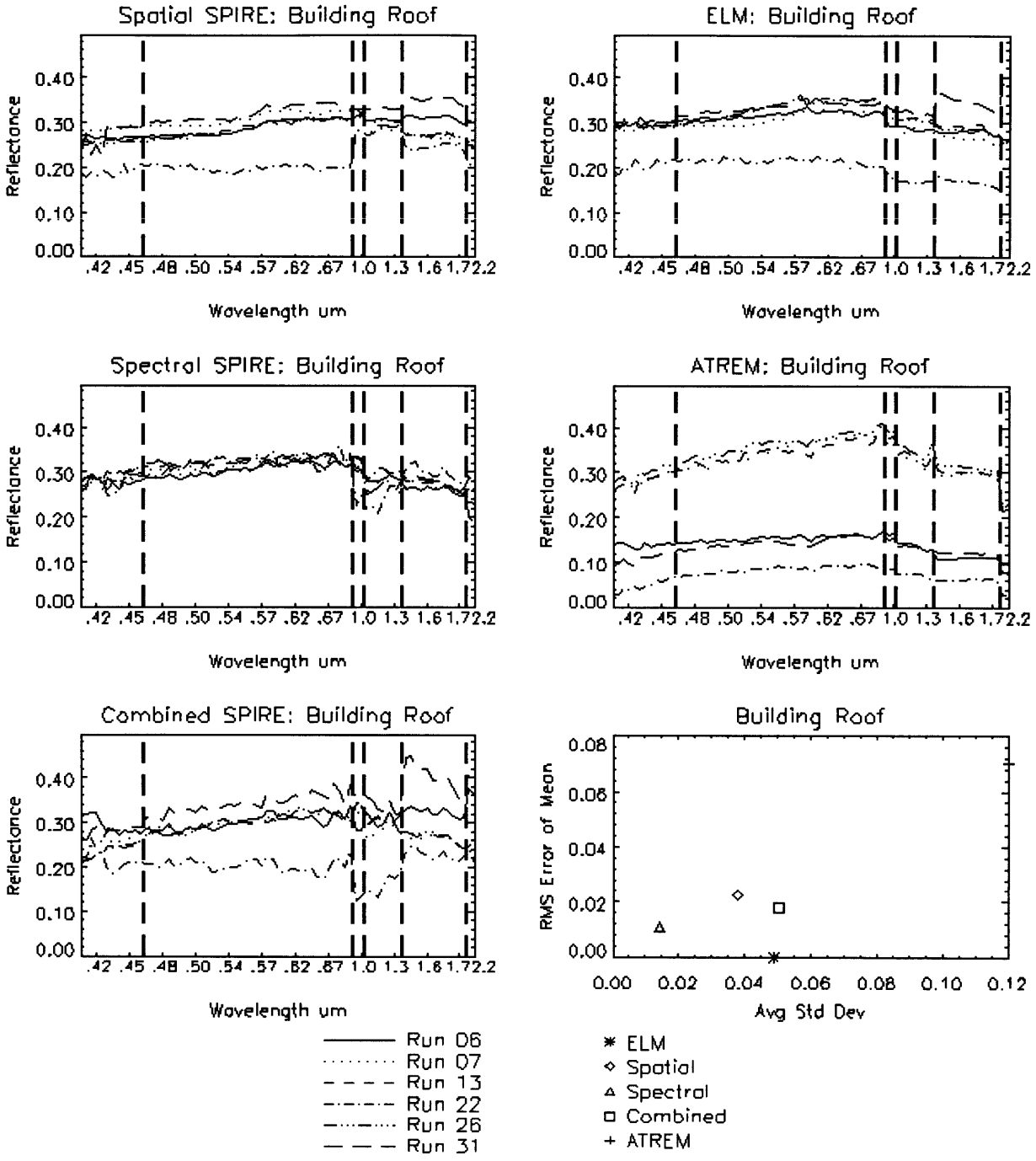


Figure 7.3(o). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the building roof, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

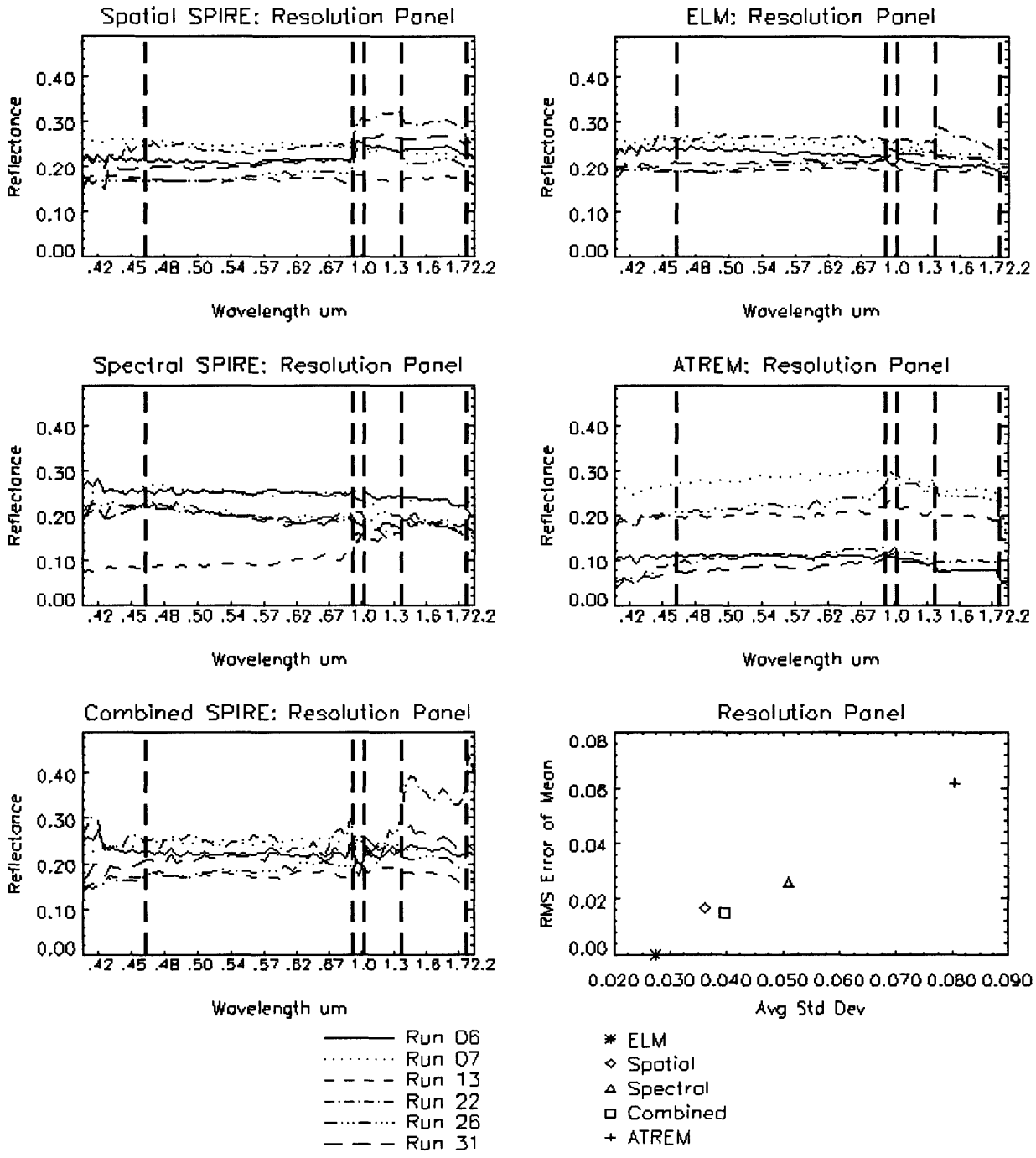


Figure 7.3(p). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the resolution panel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

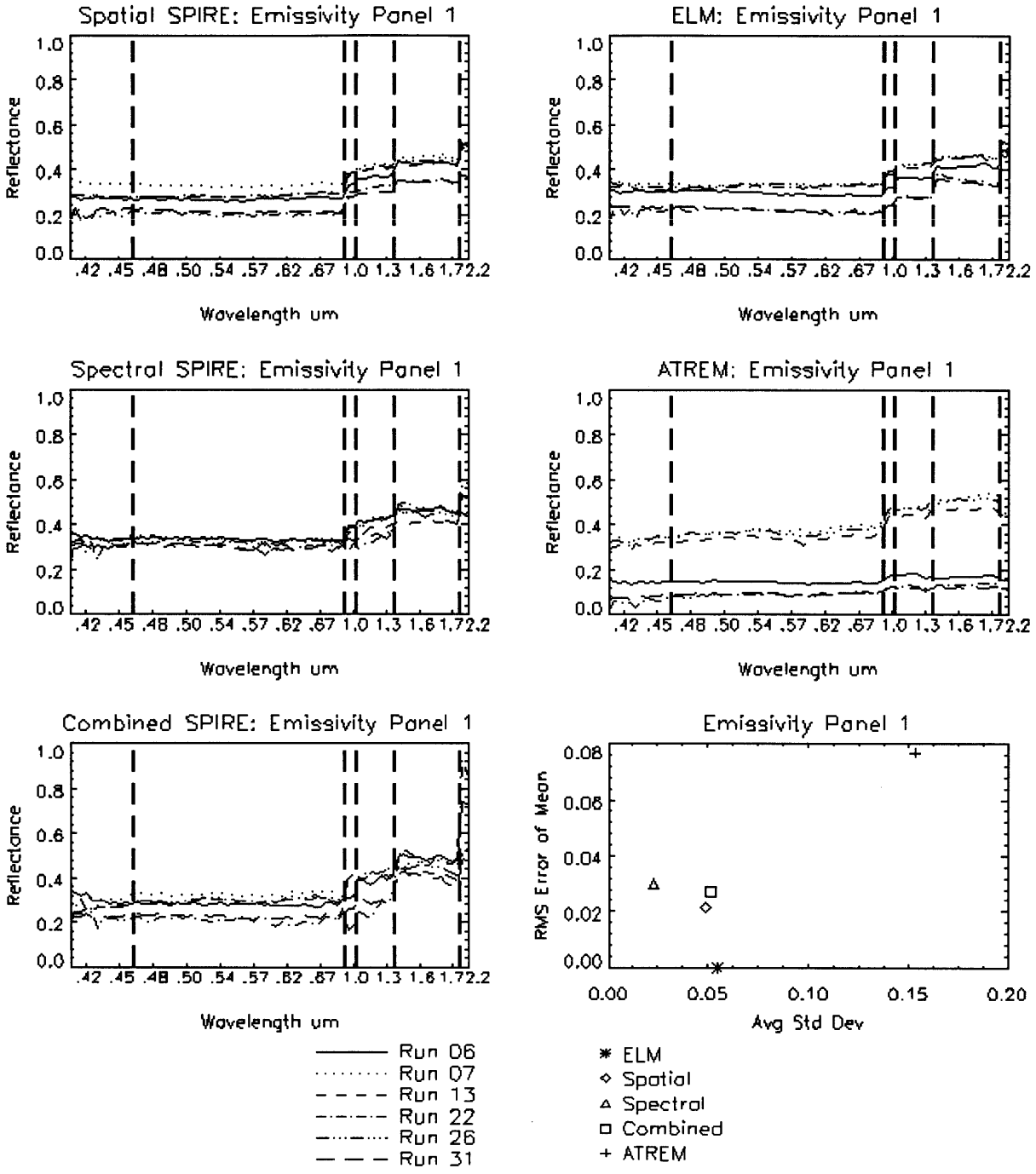


Figure 7.3(q). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the emissivity panel 1, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

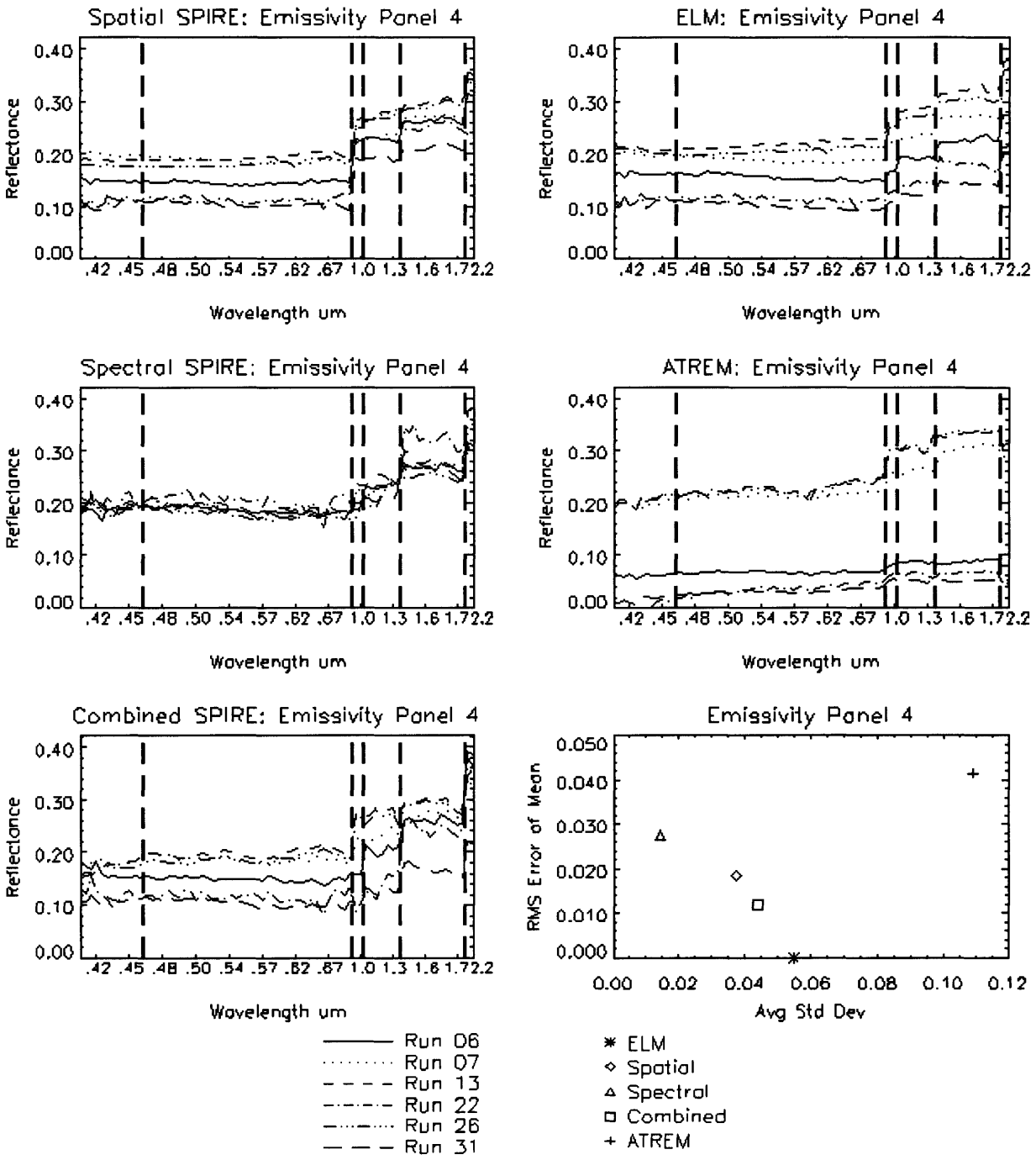


Figure 7.3(r). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the emissivity panel 4, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate.*

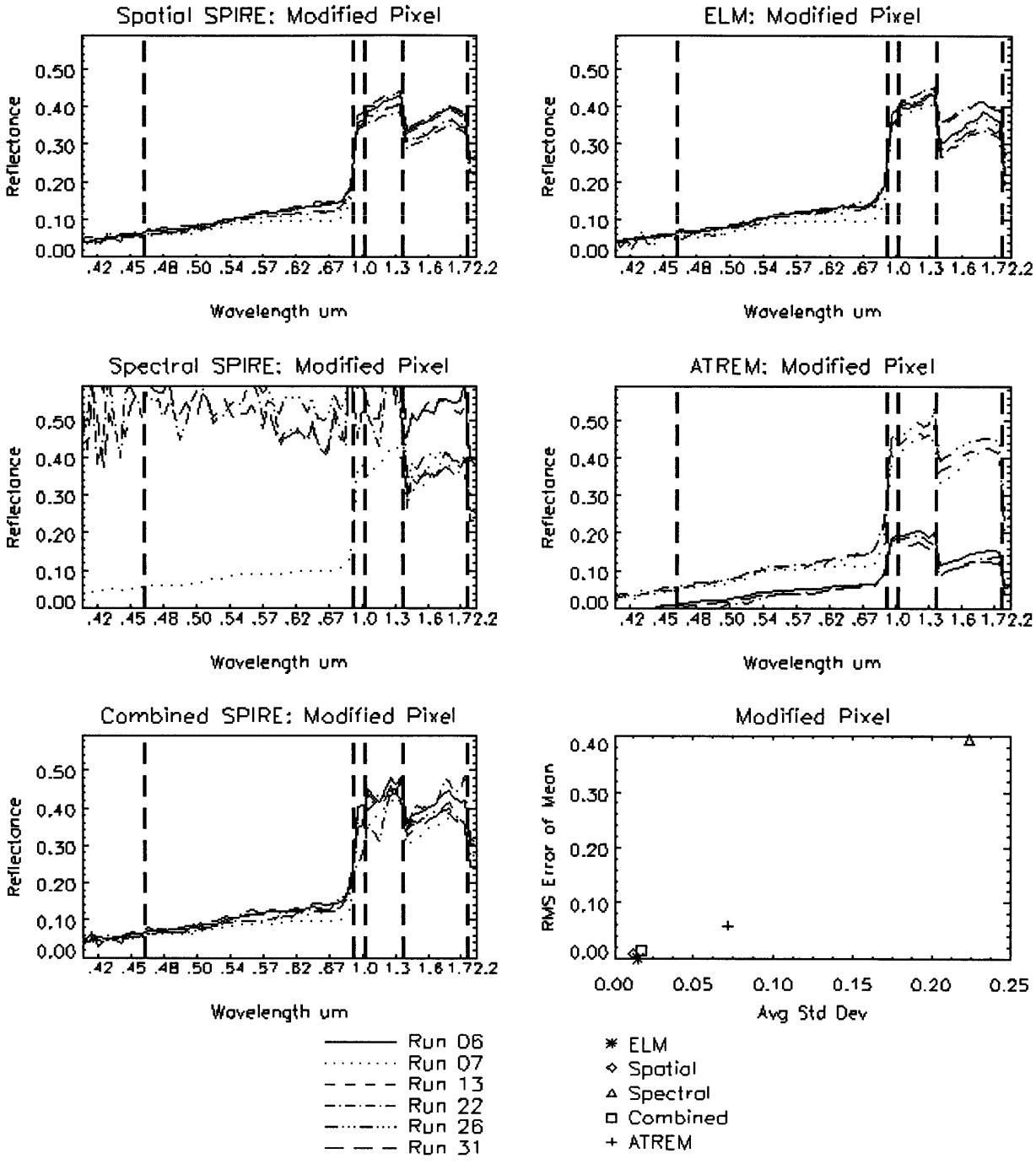


Figure 7.3(s). *Reflectance estimates for Spatial SPIRE, Spectral SPIRE, Combined SPIRE, ELM, and ATREM, for all runs for the modified pixel, with a scatter plot of average channel standard deviation from the mean for all six runs versus the RMS error of the mean estimate with respect to the mean ELM estimate. The modified pixel was a truck in Run 07, and changed to mowed grass in all subsequent Runs. For Run 07, a similar mowed grass pixel is plotted for comparison.*

that may have changed, since the grass is not uniform and has a mottled appearance. Therefore one would expect Spatial SPIRE to do better. The resolution panel also can be considered a changed, or modified pixel in the images collected at higher altitude, because the panel then was so small that the best pixel selected was likely not pure. The high variance of the emissivity panels for all algorithms may have been caused by the fact that they were not initially cleaned upon deployment, but were cleaned after the thunderstorm on 25 June 1997.

We now summarize the salient performance differences between the Spatial Case 4, Spectral Case D, and Combined Case IV SPIRE algorithms:

- Spectral SPIRE works the best on unmodified pixels, and poorly on modified pixels.
- Spatial SPIRE works the best on modified higher reflectivity pixels, and on modified low reflectivity pixels if there is adequate SNR
- Combined SPIRE works almost as well as Spatial on modified higher reflectivity pixels, and better than Spatial on modified low reflectivity pixels under low SNR conditions.

7.4 Use of SPIRE in an Example Classification Application

Many classification applications involve a human analyst who studies a hyperspectral image and identifies known or interesting pixels in the image. These pixels are then used as training samples to train a classifier algorithm that can be used to classify other pixels in the image. Such supervised classifiers (Schowengerdt, 1997) are often applied to a subset of Principal Components after a PCA has been performed, to limit the dimensionality of the problem. Figure 7.4 depicts the logical flow of such a supervised classification process done in a PC subset. This figure is similar to Figure 2.3, with the main difference being that the

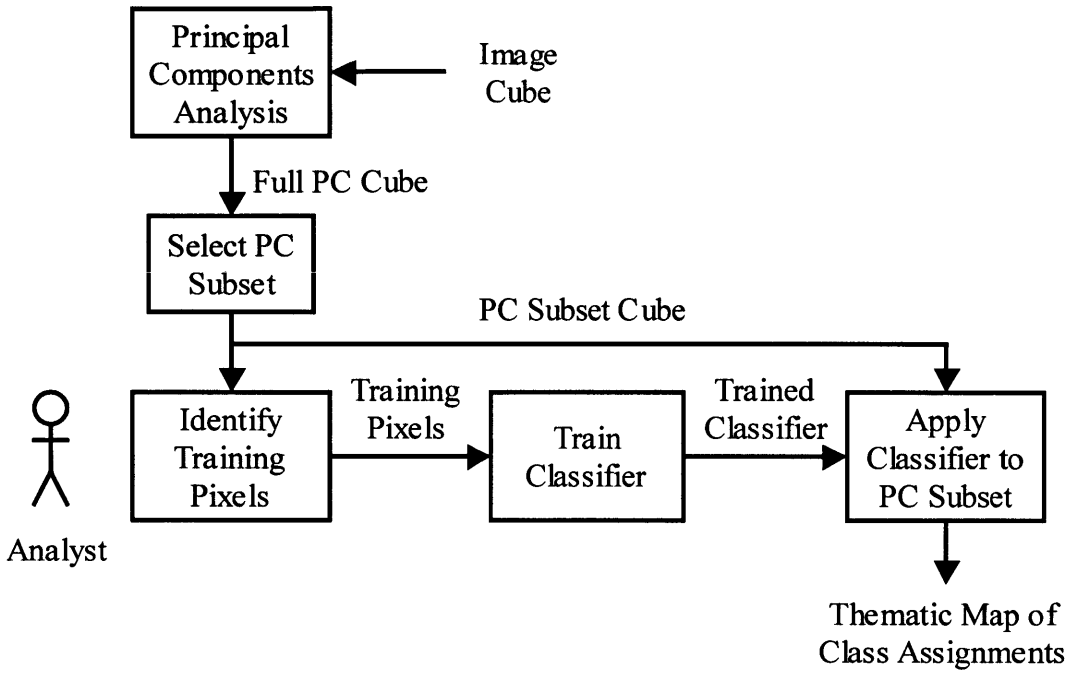
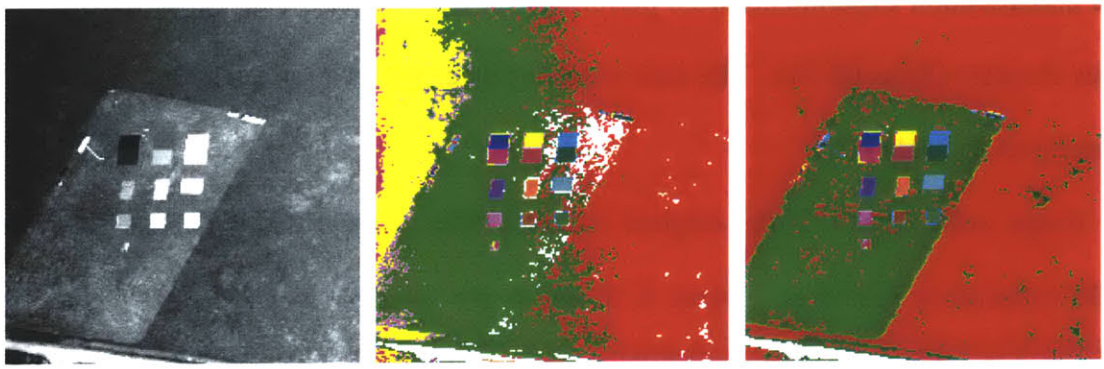


Figure 7.4. *Supervised classification processing in a PC subset. A principal components analysis is performed on the original image cube. A subset of the lowest PCs is selected which contain the significant degrees of freedom in the data. A human analyst then identifies training pixels for the material classes to be identified. Next, a classifier algorithm is trained using the training pixels from the PC subset. The classifier is then applied to the rest of the pixels in the PC subset cube and the thematic map generated.*

classification is done in just the low PCs (with highest variance) which contain the significant degrees of freedom present in the data.

As discussed in Chapter 2, classification can be done using the radiance spectra directly from the image *i* without estimating reflectance. Classifying within the same radiance image that the training pixels were drawn from suffers if the illumination noises vary within the image, and applying the classifier to other images suffers if illumination conditions change between images. Both of these problems can be solved by estimating reflectance and then training and classifying in reflectance space instead. By removing the in-scene and scene-to-scene illumination variations, these noises no longer impede the classifier.



(a) (b) (c)

Figure 7.5. *Breakdown of supervised classification within a radiance image cube. A simulated hyperspectral radiance cube (a) was generated with a multiplicative noise m that varied horizontally as a ramp across the image and a uniform additive noise* ***a****. A minimum distance classifier was trained on 15 pixel types in the radiance image and then applied, with the results in (b). We see in (b) that the varying multiplicative noise confused the classifier. In (c), the radiance image has been processed using the Spatial SPIRE Case 4 algorithm and the training and classification redone in the estimated reflectance cube. By removing the spatially varying illumination noise, classification performance was improved.*

Figure 7.5 illustrates how classification can break down in a single radiance image if the illumination noise m varies across the scene. A simulated hyperspectral reflectance image was multiplied by a horizontal ramp to simulate a spatially varying multiplicative noise, and then a spatially uniform additive noise vector was added to each pixel. A single channel image is shown in Figure 7.5(a) where the gradient in illumination is apparent from left to right. ENVI's minimum distance classifier (*ENVI User's Guide*, 1997) was then trained on 15 pixel classes and applied to the image, with the classification results shown Figure 7.5(b). We see that the classification performance is poor due to the spatially varying illumination. In Figure 7.5(c), we have processed the simulated radiance cube using the Spatial SPIRE Case 4 algorithm, retrained the classifier, and reapplied it. We see that by using SPIRE to estimate reflectance, the varying illumination noise has been eliminated and classifier performance has been greatly improved.

To demonstrate a similar improvement in classification across multiple images, a series

of nine classification experiments were performed on the six test image cubes processed in the previous chapters (Runs 06-31). The nine experiments consisted of classifying in the nine types of cubes listed in Table 7.1.

Image cube type 1 is the original radiance cube. Types 2 and 3 are the ELM and ATREM reflectance estimates. Types 4, 5, and 6 are the reflectance estimates from Spatial, Spectral, and Combined SPIRE algorithms. Type 7 is the prior reflectance generated from Run 07 ELM reflectance estimates that was used in the SPIRE processing for types 4-6. These prior cubes were included mainly as a reference and should all experience excellent classification results. Type 8 is the result of a "Selective" SPIRE technique that chooses between different SPIRE algorithm outputs to optimize classification performance which is described in Section 7.4.1. Type 9 is the result of Spatial SPIRE processing without a prior reflectance to generate "pseudo" reflectance from a pseudo prior reflectance which is described in Section 7.4.2.

Type	Description
1	Original *i* radiance cubes
2	ELM estimated reflectance cubes
3	ATREM estimated reflectance cubes
4	Spatial SPIRE estimated reflectance cubes
5	Spectral SPIRE estimated reflectance cubes
6	Combined SPIRE estimated reflectance cubes
7	Run 07 ELM derived prior reflectance cubes used in 3)-5)
8	Reflectance cubes created using the Selective SPIRE technique
9	Case 2 Spatial SPIRE estimated reflectance cubes generated using a pseudo prior reflectance

Table 7.1. *Image cube types tested in classification experiments.*

7.4.1 Selective SPIRE Technique For Classification Applications

Thematic maps output by a classifier are used by analysts to interpret the meaning and impact of changes in an image. Therefore, analysts need both the modified and unmodified

pixels classified properly to allow proper interpretation of the results. An image that shows only the modified pixels and their locations does not allow the analyst to properly relate the changes to the rest of the image. We will now develop a technique that selects between the different SPIRE algorithm's reflectance estimates in a way that optimizes the classification performance.

The first step is to chose a technique for detecting changes. Any change detection must be done by comparing one of the SPIRE reflectance estimates against the prior reflectance. While Spectral SPIRE will have very low differences between its estimates and the prior on unmodified pixels, it will also tend to have small differences at the modified pixels since much of the prior information ends up in the reflectance estimates. This is the cause of Spectral SPIRE's poor performance on modified pixels and makes it less suitable as a change detector than Spatial or Combined SPIRE. In the absence of low SNR conditions and restrictions on computational cost, Spatial SPIRE is the best choice for use in change detection since its good performance on both modified and unmodified pixels lets changes stand out. If low SNR conditions are present, then Combined SPIRE will have better performance overall than Spatial. If computational costs are an issue, then Combined SPIRE is also the correct choice, since it will have almost as good performance as Spatial but at far less cost.

Once we have a technique for detecting and estimating reflectance changes, then these results can be combined with either the prior reflectance or Spectral SPIRE reflectance estimates for the unmodified pixels, and then classification run on the resulting cube. One could argue that if a pixel is unmodified, then one should simply use the prior reflectance for such a pixel. There are, however, reasons to use the Spectral SPIRE reflectance estimate instead.

Assume, for example, that we use a vector distance threshold for detecting changes, that allows for some amount of noise to be present without generating a false alarm. With such a

threshold, there will be some pixels with small changes that will slip through and be labeled as unmodified. If we simply use the prior reflectance, then these changes will be completely lost. If we use the Spectral SPIRE result, only the part of the reflectance change that ends up in the log **m** noise PCs will be lost. As we saw in Figure 5.9, significant amounts of signal from a change in reflectance still remains in the higher, non-log **m** PCs. A classifier could be trained to identify pixels that experience such changes based on the information in the higher PCs.

In the absence of low SNR conditions, Spatial SPIRE has the best classification performance on modified, or changed, pixels while Spectral SPIRE performs the best on unmodified pixels. However, if computational costs are an issue, then Combined SPIRE has nearly the same performance as Spatial, with much lower computational costs. When low SNR conditions exist, as they do in the test data set used in this thesis, Combined SPIRE has the best classification performance on modified, low reflectivity pixels.

Therefore, our Selective SPIRE strategy is to use Spatial or Combined SPIRE to detect changes, then use Spectral SPIRE reflectance estimates on unmodified pixels and Spatial or Combined SPIRE reflectance estimates for modified pixels. The choice between Spatial and Combined will depend on the whether or not Spatial SPIRE is hampered by low SNR conditions and the importance of computational costs.

Figure 7.6 depicts the processing block diagram for the Selective SPIRE technique. Due to the presence of low SNR conditions in our data set, we have chosen to use Combined SPIRE for both change detection and reflectance estimation of modified pixels. Use of Combined SPIRE also keeps the Selective SPIRE technique computationally efficient. Combined SPIRE is first run on the image to generate a reflectance estimate which is used to identify modified

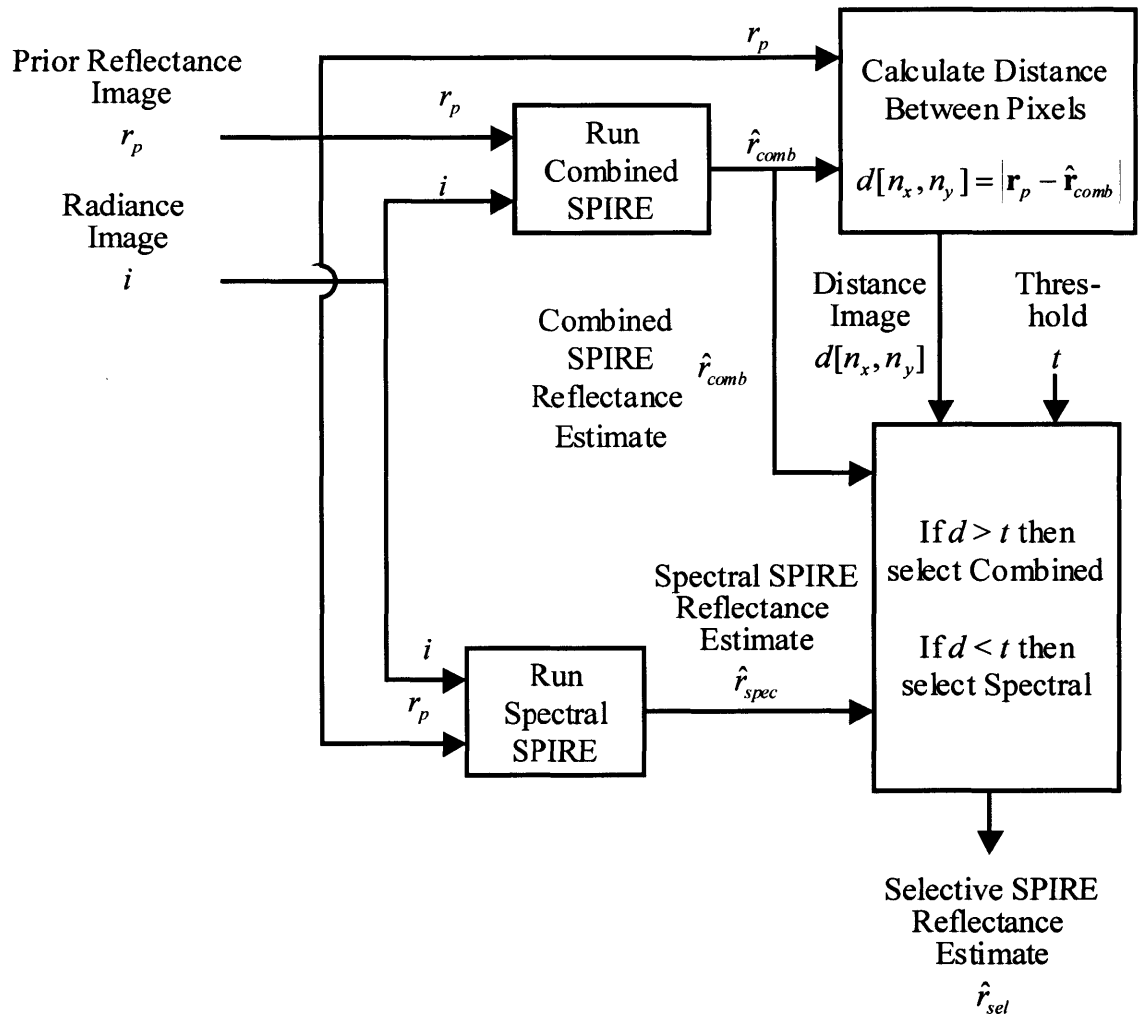


Figure 7.6. *Processing block diagram for Selective SPIRE technique. Combined SPIRE is first run on the image to generate a reflectance estimate which is used to identify modified pixels. The identification is done using a distance calculation which is the magnitude of the difference between the estimated pixel reflectance vector and the prior reflectance vector for each pixel. The distance at each pixel is then used to select between Combined and Spectral reflectance estimates, where modified pixels with distances above the threshold use the Combined SPIRE estimates, and unmodified pixels use the Spectral SPIRE estimate. For high SNR conditions and when computational costs are not an issue, Spatial SPIRE can be substituted for Combined.*

pixels. The identification is done using a distance calculation which is the magnitude of the difference between the estimated pixel reflectance vector and the prior reflectance vector for each pixel. For two reflectance images r_1 and r_2, the distance $d[n_{x_o}, n_{y_o}]$ between the pixel spectra $\mathbf{r}_1[n_{x_o}, n_{y_o}, n_\lambda]$ and $\mathbf{r}_2[n_{x_o}, n_{y_o}, n_\lambda]$ is defined as:

$$d[n_{x_o}, n_{y_o}] = \sqrt{\sum_{n_\lambda=1}^{N} \left(\mathbf{r}_1[n_{x_o}, n_{y_o}, n_\lambda] - \mathbf{r}_2[n_{x_o}, n_{y_o}, n_\lambda]\right)^2} \\ = \left|\mathbf{r}_1[n_\lambda] - \mathbf{r}_2[n_\lambda]\right| \tag{7.1}$$

The distance at each pixel is then used to select between Combined and Spectral reflectance estimates, where modified pixels with a distance above the threshold use the Combined SPIRE estimates, and unmodified pixels use the Spectral SPIRE estimate.

The selective SPIRE technique was run on all six test data cubes, using a threshold of 1.0, which was selected after analyzing the distances between the Combined SPIRE and ELM reflectance estimates for the cubes, and using knowledge of where changes had actually occurred in the images, such as the truck moving. The optimum technique for setting such a threshold was not explored and is left as an area for further work. The resulting Selective SPIRE reflectance estimates were used in the classification experiments.

7.4.2 Classification with SPIRE Pseudo Reflectance

Spatial SPIRE algorithms replace the low spatial frequency reflectance information in an estimated reflectance image with that from the prior reflectance. This causes no problems so long as our assumption that any changes since the prior have been at higher frequencies is not violated. However, if there are errors or differences in these prior low spatial frequency components, then these errors are introduced into the new estimated reflectance. If many images are processed with this same prior, then the same low frequency errors are introduced into all of the images.

One can take advantage of this behavior to effectively use Spatial SPIRE algorithms without prior reflectance information for certain applications. If we select a fairly arbitrary prior

reflectance, which we will call a pseudo prior, and process images with it, then the output of the SPIRE algorithm will be a reflectance estimate that has the same incorrect low spatial frequency content as the pseudo prior. We will refer to such reflectance estimates as pseudo reflectance.

One example of a pseudo prior would be to simply scale a prior radiance image using its maximum radiance so that all of the pixel values fall between zero and some number less than or equal to one. This pseudo prior can then be used as the prior reflectance when running SPIRE on subsequent images. The reflectance estimates for these later images will be incorrect in an absolute sense, but all images processed using the same pseudo prior will have the same systematic errors. The resulting pseudo-reflectance image cubes can then be used for performing change detection, since the errors introduced by using the pseudo prior would be the same in all of them. More importantly, any classifier trained on one of the image cubes can then be applied to all other image cubes that were processed using the same pseudo prior, allowing successful application of the classifier across images in which illumination conditions have changed, just as with SPIRE results using a real prior.

Figure 7.7 depicts the processing flow of such a cross-image classification using SPIRE with a pseudo prior reflectance. The first radiance image is scaled so that all of its pixel values are between zero and one. This pseudo prior is then used to process the first image with a Spatial SPIRE algorithm to generate a pseudo reflectance image. A supervised classifier is then trained using selected pixels from the pseudo reflectance generated from the first image. A second radiance image collected under different illumination conditions is then processed with Spatial SPIRE using the same pseudo prior. The resulting second pseudo reflectance image can then be successfully classified using the classifier trained from the first image. The first pseudo

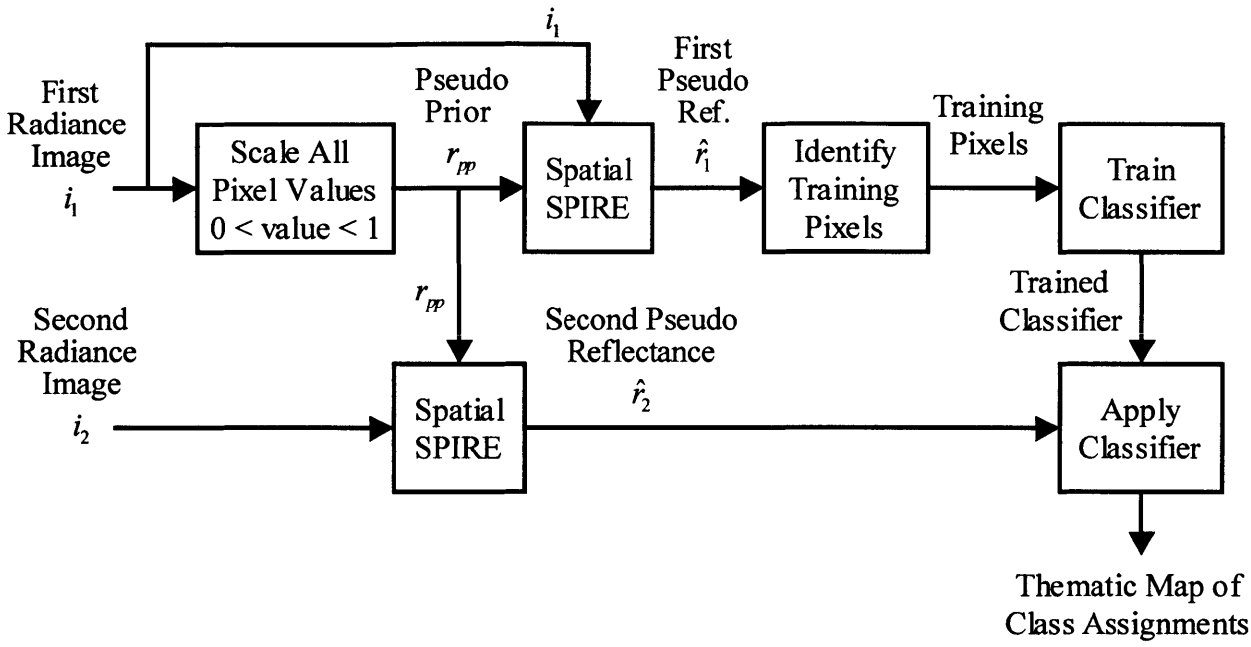


Figure 7.7. *Cross-image classification using SPIRE with a pseudo prior reflectance. The first radiance image is scaled so that all of its pixel values are between zero and one. This pseudo prior is then used to process the first image with a Spatial SPIRE algorithm to generate a pseudo reflectance image. A supervised classifier is then trained using selected pixels from the pseudo reflectance from the first image. A second radiance image collected under different illumination conditions is then processed with Spatial SPIRE using the same pseudo prior. The resulting second pseudo reflectance image can then be successfully classified using the classifier trained from the first image. The first pseudo reflectance image can also be successfully classified, even if the illumination noise varies across the image.*

reflectance image can also be successfully classified, even if the illumination noise varies across the image. This pseudo reflectance SPIRE processing was applied to all six test data cubes, using the Spatial Case 2 algorithm, to generate the pseudo reflectance cubes used in the classification experiments.

Currently, tests of using a pseudo prior have been limited to Spatial SPIRE algorithms. Since the PC rotations used by Spectral and Combined SPIRE algorithms are defined using ensembles of **m** and **a** vectors independently of prior reflectance, it is expected that the use of pseudo priors will work with Spectral and Combined SPIRE algorithms, and Selective SPIRE as well. Confirmation of this is left as a suggestion for further work.

7.4.3 Classification Experiment Results

For each type of image cube, a PCA was performed on the cube for Run 07 and the resulting PC rotation applied to the other five cubes. Next, new cubes were generated from each PC cube containing only PCs 1-20 from each cube, since PCs 20-73 contained mostly noise. Training pixels were then collected from the Run 07 cube that corresponded to the 19 pixel types studied in Chapters 3, 5, and 6. For each pixel type, or class, only a single training pixel was used. ENVI's minimum distance classifier (*ENVI User's Guide*, 1997) was then trained using the 19 pixel classes and applied to all six test image cubes of each type. Many other classifiers exist, including ones that use ensembles of training pixel to capture natural variations within each class, but a thorough exploration of how they all perform with the different reflectance estimates is beyond the scope of this thesis.

Figure 7.8 plots the PC spectra of the 19 pixels used to train the ELM classifier with the legend depicting the colors assigned to each pixel class. For black and white copies of this thesis, these colors will appear as shades of gray, but the resulting thematic maps will still show qualitative relative performance. For color copies, we must note that a limitation in the ENVI software only supported 17 colors, so that the white and red colors are reused. White is assigned to both the 64 percent and resolution panels, while red is assigned to both the 4 percent panel and the parking lot.

Figure 7.9 shows the classification results for all nine image variants of the Run 07 image cube. The results are depicted as thematic maps which display the color from Figure 7.8 of the class to which each pixel was assigned by the classifier. Since these cubes were themselves used to train the nine instances of the classifier, classification performance is very

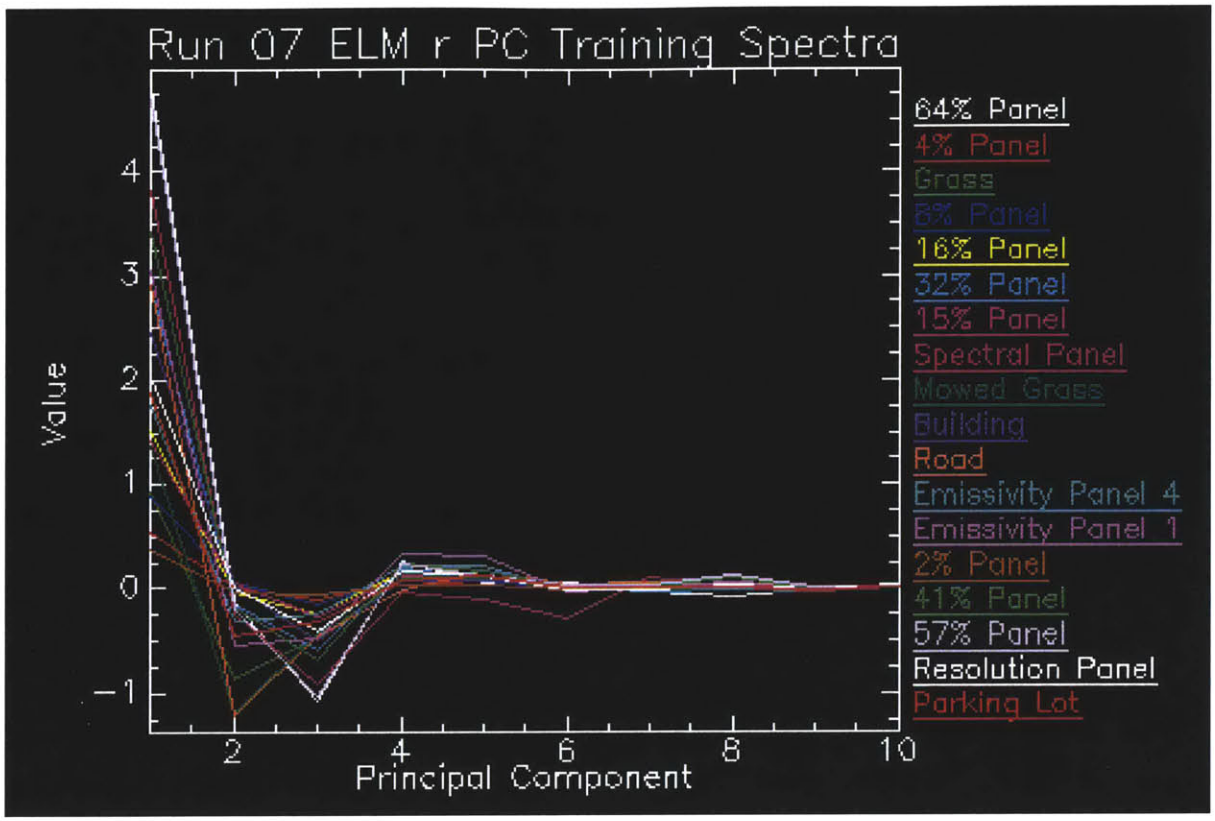


Figure 7.8. *Spectra used to train minimum distance classifier for Empirical Line Method (ELM) reflectance results. The material spectra are in Principal Components space as defined by Principal Components analysis of the ELM reflectance estimates for the Run 07 reflectance cube. Note that the colors of white (64 percent and resolution panels) and red (4 percent panel and parking lot) were each assigned to two material classes due to a finite set of colors available. These same material-to-color assignment were also used for the classification tests of ATREM and SPIRE reflectance estimates.*

good in all of them. The results for classifying directly in the radiance image cube shown in Figure 7.9(a) shows slightly poorer classification results than the rest, with some of the lower left mowed grass misclassified, and more of the road misclassified than the others. Note that in all of the different types of cubes, mixed pixels (such as those at the edges of panels) will tend to be misclassified, since the classifier was not trained on them.

Figures 7.10-7.14 depict classification results from applying the Run 07 trained classifiers to the other five Runs (Runs 13, 26, 06, 22, and 31). The results shown in Figures 7.10 (Run 13) and 7.11 (Run 26) are very similar. Both of these runs were collected on the same

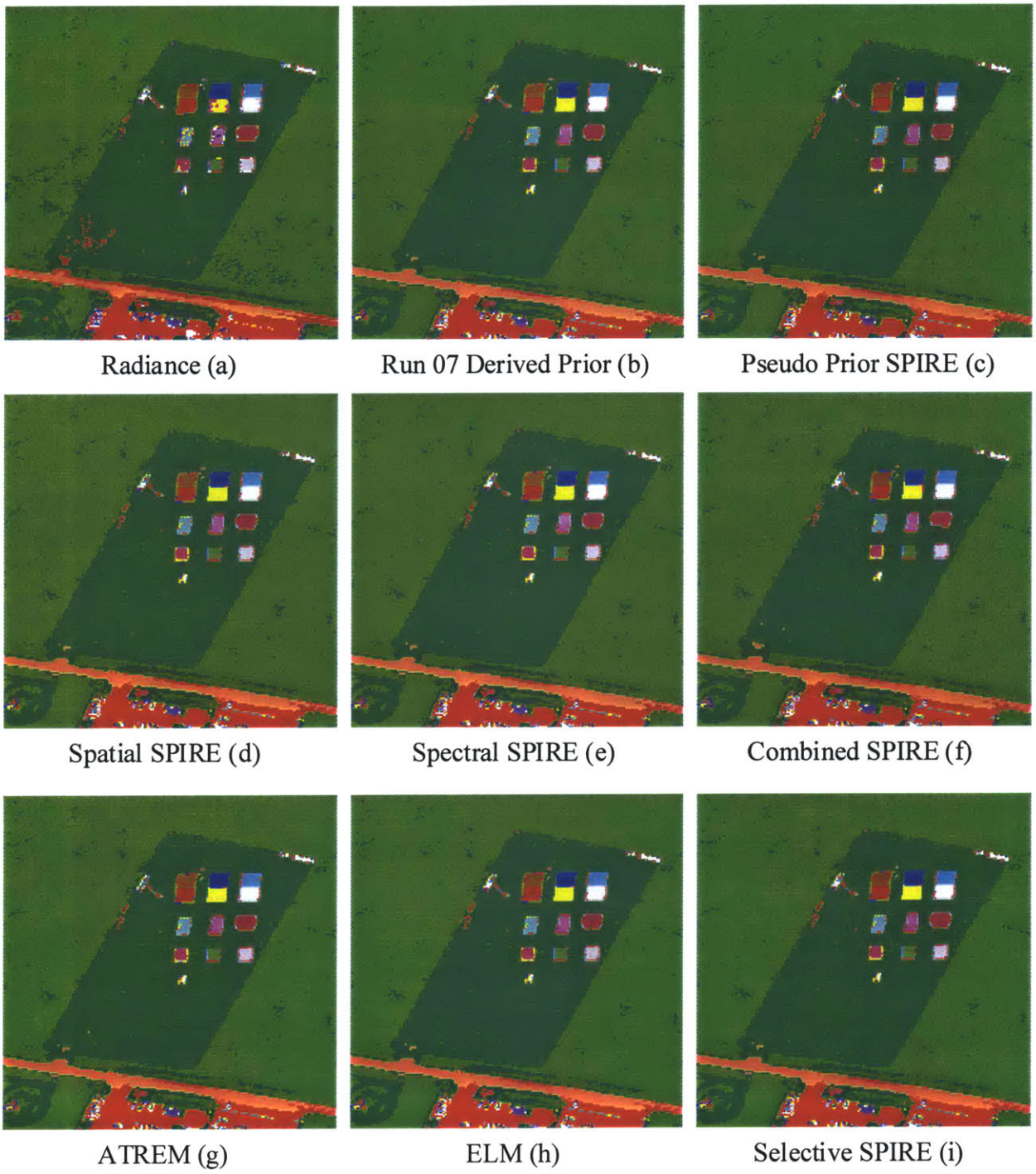


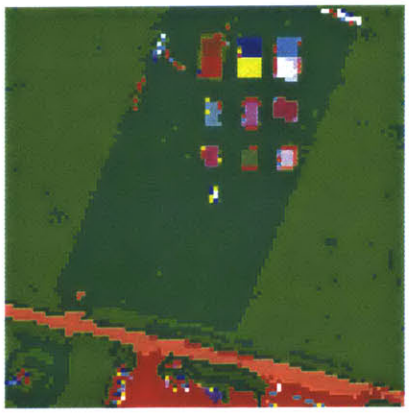
Figure 7.9. *Classification results for the Run 07 PC subset image cubes. In each of the nine types of image cubes tested, training pixels were selected from the Run 07 image cube for 19 material classes. A minimum distance classifier was then trained and applied to the rest of the Run 07 image cube. We see that classification performance was good for all nine types of cubes, with the Radiance cube (a) showing slightly poorer results than the rest.*

Radiance (a)

Run 07 Derived Prior (b)

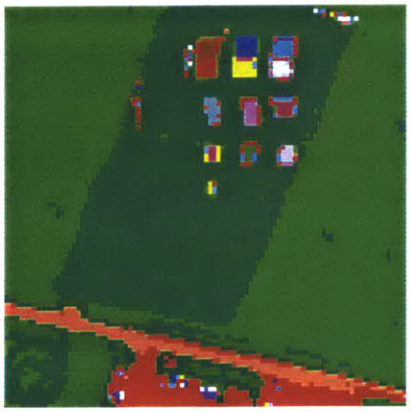
Pseudo Prior SPIRE (c)

Spatial SPIRE (d)



Spectral SPIRE (e)

Combined SPIRE (f)



ATREM (g)

ELM (h)

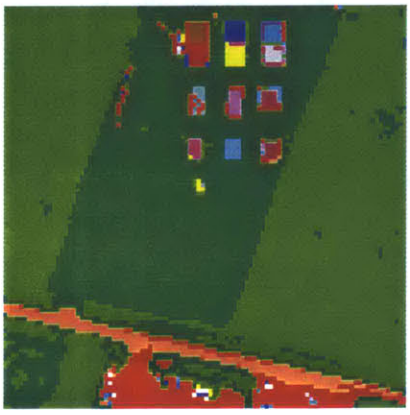
Selective SPIRE (i)

Figure 7.10. *Classification results for the Run 13 PC subset image cubes. In each of the nine types of image cubes tested, the minimum distance classifier trained on the Run 07 image cube was to the Run 13 PC subset image cube. Classification performance was good for all nine types of cubes, since the illumination conditions for Run 13 were very similar to those of Run 07. Note that the Run 07 prior image in (b) has the truck parked at the edge of the mowed grass to the left of the 2 percent panel. The truck has moved, leaving grass at this location in Run 13. Spectral SPIRE (e) does not correctly classify this area as mowed grass.*

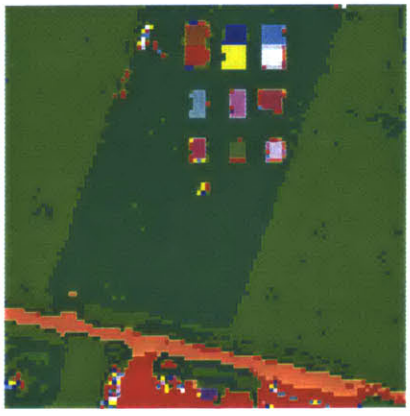
Radiance (a)

Run 07 Derived Prior (b)

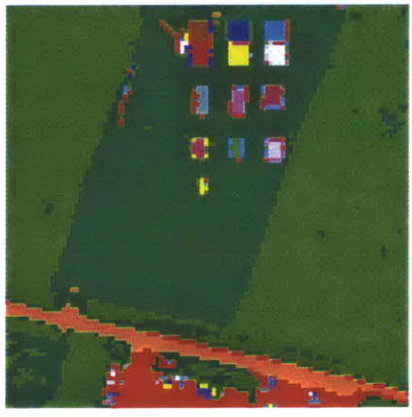
Pseudo Prior SPIRE (c)



Spatial SPIRE (d)



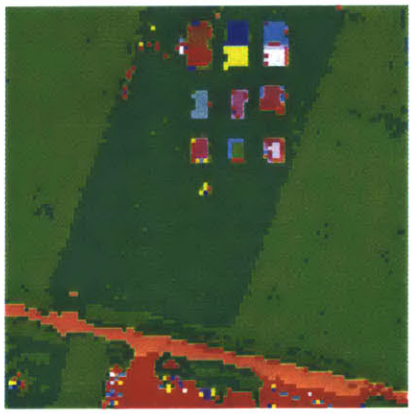
Spectral SPIRE (e)



Combined SPIRE (f)

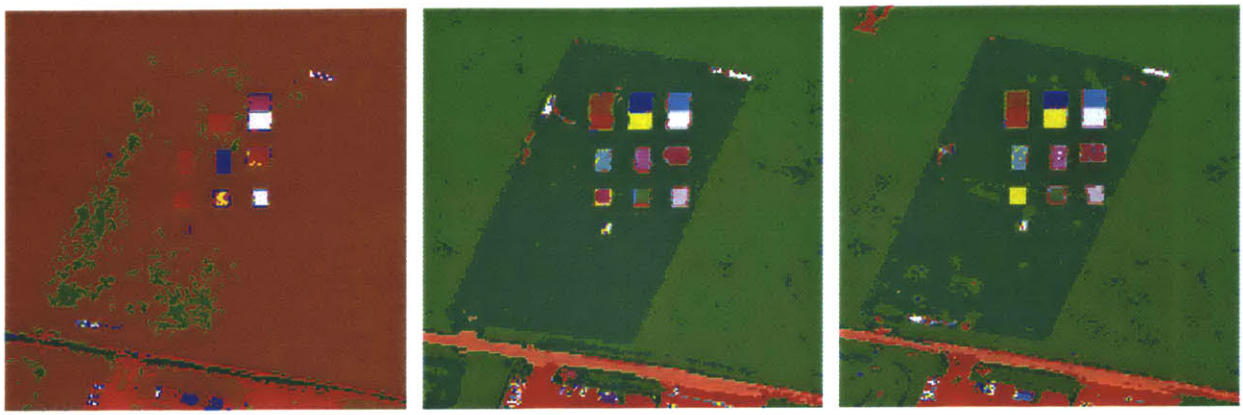
ATREM (g)

ELM (h)



Selective SPIRE (i)

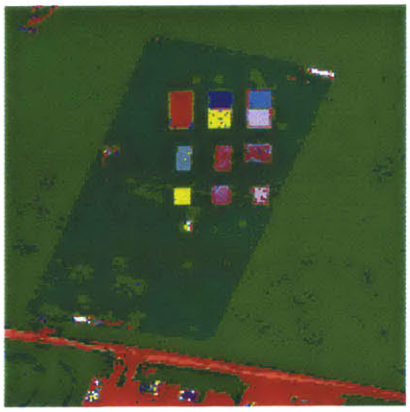
Figure 7.11. *Classification results for the Run 26 PC subset image cubes. In each of the nine types of image cubes tested, the minimum distance classifier trained on the Run 07 image cube was applied to the Run 26 PC subset image cube. Classification performance was good for all nine types of cubes and similar to those for Run 13, since the illumination conditions for Run 26 were very similar to those of Run 07 and Run 13.*



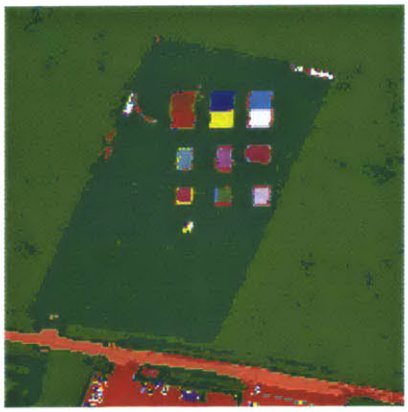
Radiance (a)

Run 07 Derived Prior (b)

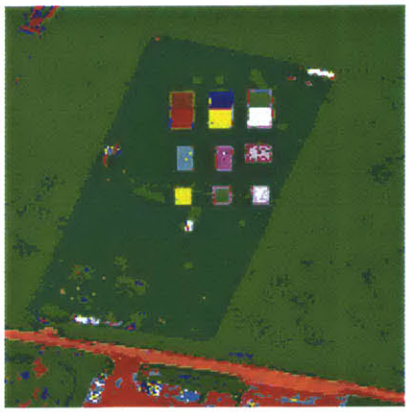
Pseudo Prior SPIRE (c)



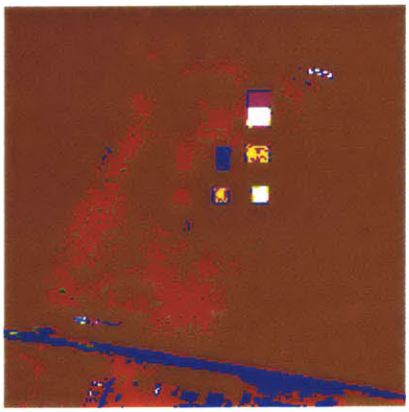
Spatial SPIRE (d)



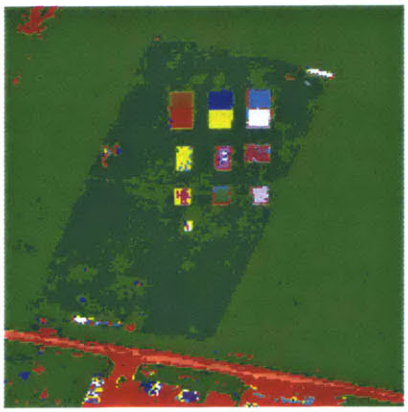
Spectral SPIRE (e)



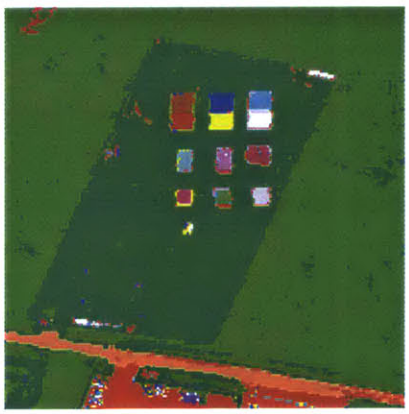
Combined SPIRE (f)



ATREM (g)



ELM (h)



Selective SPIRE (i)

Figure 7.12. *Classification results for the Run 06 PC subset image cubes. In each of the nine types of image cubes tested, the minimum distance classifier trained on the Run 07 image cube was applied to the Run 06 PC subset image cube. Run 06 was collected on a different day than Run 07. Classification performance was poor in both the Radiance (a) and ATREM reflectance (g) cubes. The Selective SPIRE (i) results are the best overall.*

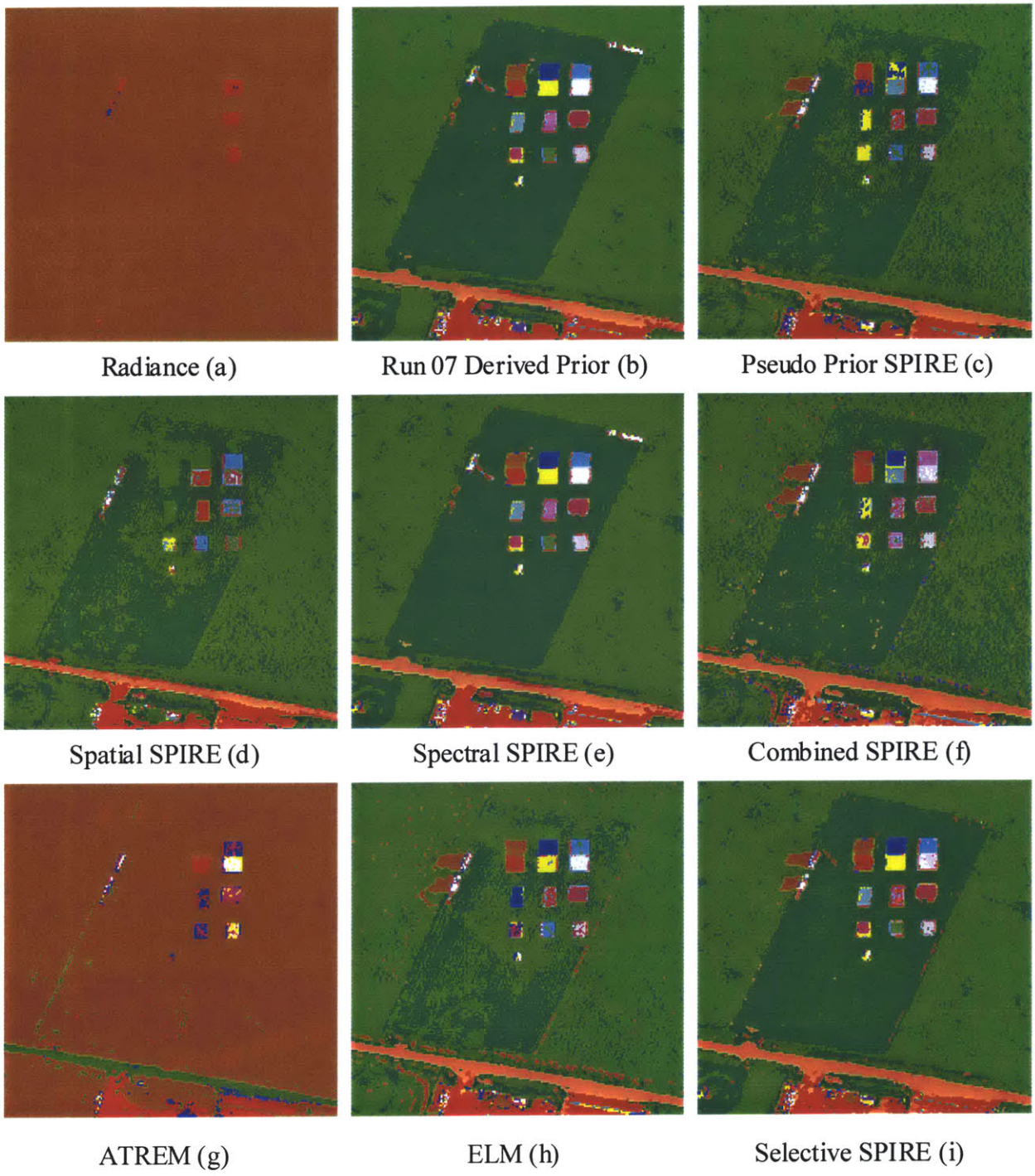


Figure 7.13. *Classification results for the Run 22 PC subset image cubes. In each of the nine types of image cubes tested, the minimum distance classifier trained on the Run 07 image cube was applied to the Run 22 PC subset image cube. Run 22 was collected on a different day than Run 07. Classification performance was poor in both the Radiance (a) and ATREM reflectance (g) cubes. The Selective SPIRE (i) results are the best overall.*

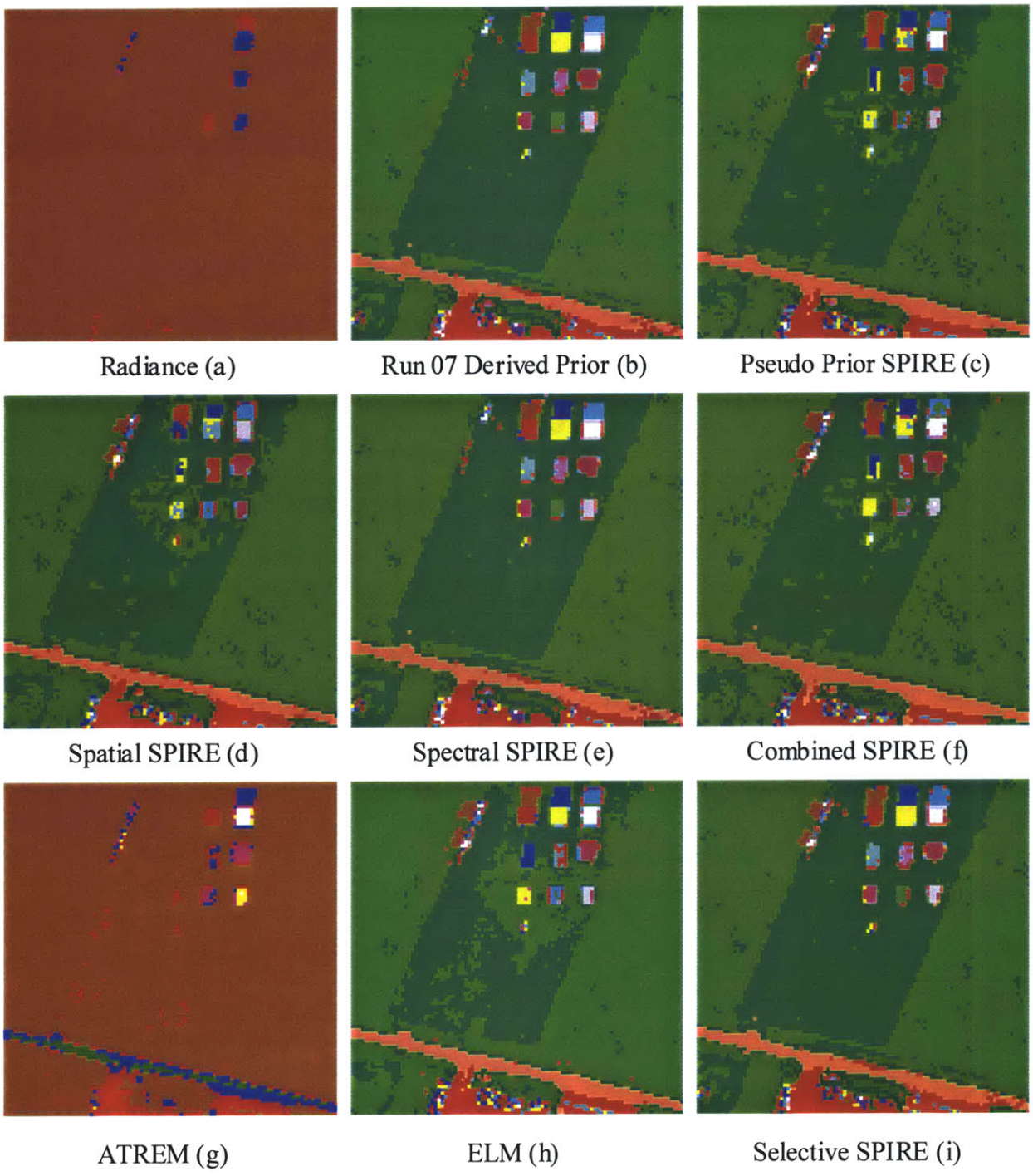


Figure 7.14. *Classification results for the Run 31 PC subset image cubes. In each of the nine types of image cubes tested, the minimum distance classifier trained on the Run 07 image cube was applied to the Run 31 PC subset image cube. Run 31 was collected on a different day than Run 07. Classification performance was poor in both the Radiance (a) and ATREM reflectance (g) cubes. The Selective SPIRE (i) results are the best overall.*

day as Run 07, under very similar illumination conditions. This allowed unmodified pixels to be classified well in all of the cubes, again with slightly poorer performance in the Radiance (a) cubes. This performance difference is due to the fact that the shape of the solar spectrum in the radiance cubes amplifies shorter wavelengths more than longer ones. Because of this, small differences between classes in the longer wavelengths contribute less to the distance calculation used by the classifier. In fact, the four classes on which the Radiance classifier has difficulty are grass, mowed grass, road, and parking lot which have their main inter-class differences at longer wavelengths. For the purposes of this discussion, we will refer to the failure of the Radiance classifier to correctly classify the road, as exemplified in Figure 7.11(a), as the "Road" failure.

In the Run 07 derived prior classification results of Figures 7.10(b) and 7.11(b), we can see that the truck was parked at the left edge of the mowed grass next to the 2 and 4 percent panels. In both Runs 13 and 26, the truck has been moved closer to the panels. The newly exposed mowed grass pixels that were under the truck in Run 07 are now modified pixels in subsequent runs. In Figures 7.10 and 7.11, we see that all of the other classifiers except for the Spectral SPIRE (e), correctly identified these pixels as grass. This is an excellent example of the impact of Spectral SPIRE's poor performance on modified pixels can have on downstream processing. In comparison, Pseudo Prior SPIRE (c), Spatial SPIRE (d), Combined SPIRE (f), ATREM (g), ELM (h), and Selective SPIRE (i) all performed well on these modified pixels. For the purposes of this discussion we will refer to the failure of the Spectral SPIRE classifier to correctly classify modified pixels as the "Change" failure.

In Figures 7.12-7.14 are shown the classification results for Runs 06, 22, and 31, all of which were collected on different days than Run 07, with different illumination conditions. Note the shadows cast by the truck (and a trailer) in Figures 7.13 and 7.14 at the left edge of the

mowed grass due to very low solar angles. What is striking about these three figures is the extremely poor classification performance on the Radiance (a) and ATREM (b) cubes. This is due to the cloud and haze induced illumination changes ignored when using the radiance images and not properly handled by ATREM. We will refer to this failure as the "Cloud" failure.

In Figures 7.12-7.14 we note that misclassification of grass and mowed grass occurred in Pseudo Prior SPIRE (c), Spatial SPIRE (d), Combined SPIRE (e) and ELM (h). These two classes are very close to each other, with the main differences in the longer wavelengths. Any Spatial SPIRE low SNR errors or other noise in these wavelengths (where SNR tends to be low) will tend to cause such misclassifications. Note that both Spectral SPIRE (e) and Selective SPIRE (i) do not suffer from this failure (nor does Run 07 Derived Prior (b), which should have excellent classification results since it is the same image merely rotated and shifted for each run). We will refer to this failure as the "Grass" failure.

Run 22 was the day of lowest SNR, causing problems for Spatial SPIRE and introducing larger noise errors in the longer wavelength channels. Consistent with our conclusions from Chapter 3, the Spatial SPIRE classification results in Figure 7.13(d) suffer the most under these conditions, misclassifying several panels and much of the mowed grass as grass. This is less of a problem in Figures 7.12(b) and 7.14(b), though the 2 percent panel is misclassified as a 4 percent panel. Similarly, the cloud shadows are classified as grass, though they are closer to the 2 percent panel in reflectivity (and should be misclassified as such). We will refer to such low-SNR induced failure as "SNR" failures.

Spectral SPIRE (e) and Selective SPIRE (i) consistently do the best on unmodified pixels in all the runs, while Spectral misclassifies modified pixels such as the grass that had been under the truck in Run 07. It is interesting to note that Spatial SPIRE (d) and Combined SPIRE (f)

perform better than ELM (h) in Figures 7.12 and 7.14, even though all three suffer from Grass failures. This is due to the fact that replacing the DC reflectance component with that from the prior eliminates the random variation between images of this component which is not removed by ELM. These variations in the DC component can have several sources, including sensor noise, errors in the ground truth spectra, and BRDF effects. Since the DC component has a large effect on distance measurements, minimum distance classification performance is affected. So for classification, SPIRE tends to not only do better than physics-based approaches such as ATREM, but surprisingly, it also does better than a ground truth based approach such as ELM. Whether or not SPIRE's superior performance over ELM will extend to more sophisticated classifiers is an area for further work.

The selective SPIRE technique has the best and most consistent classification results of all of the techniques tested. By combining Combined SPIRE's performance on modified pixels and Spectral SPIRE's performance on unmodified pixels, we achieved a higher performance than any of the techniques alone. We note in Figures 7.10(i)-7.14(i) that there are some red pixels that show up in every Selective SPIRE thematic map just to the left of the 4 percent pixel. These are mixed pixels that in the prior were mostly mowed grass with some of the truck's spectroradiometer boom in them. They are, therefore pixels with a small change in them, but the change was below the Selective SPIRE change detection threshold, as discussed in Section 7.4.1. While no longer mixed in the spatial sense of having two different materials within the same pixel, they are now mixed due to the Spectral SPIRE PC processing. Like all mixed pixels, they tend to be misclassified. An area of suggested further work would be to train a classifier to correctly identify them, or to develop a change detector to detect them as changed and use the Combined SPIRE results estimates in place of Spectral's.

We also note the similarity between Pseudo Prior SPIRE (c) and Combined SPIRE (f) results in Figures 7.9.7.14. The Pseudo Prior SPIRE results were generated with the Spatial SPIRE Case 2 algorithm, which is not quite as susceptible to the low SNR as the Spatial SPIRE Case 4 algorithm used to generate the Spatial SPIRE (d) results. Pseudo Prior SPIRE (c) performance did, however, suffer in Run 22 of Figure 7.13, where several panels also had poor classification performance. However, on the good SNR runs, classification performance was quite good. In fact, the classification performance was identical to running the Spatial Case 2 algorithm with the real priors of Figures 7.9(b)-7.14(b), showing that the use of a pseudo prior has no effect on change detection or classification results.

Table 7.2 is a failure elimination table for the eight of the nine classifiers discussed in this section. The classifier used for the Run 07 Derived Prior cubes is not included since it is not a useful classifier and was only included for comparison purposes. Solid circles indicate a full failure while open circles indicate a partial failure. The Radiance classifier exhibited the worst Road failure while ELM and Spatial SPIRE exhibited partial ones. Spectral SPIRE fails on modified, or changed pixels, while ELM, Spatial, Combined, and Pseudo Prior SPIRE all exhibit Grass failures. Radiance and ATREM suffer from fatal Cloud failures that ruin the classification results over the whole image. Spatial SPIRE also suffers from the low SNR induced SNR failures. Selective SPIRE is the only classifier that avoids all of these failures.

Failure	Radiance	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE	Selective SPIRE	Pseudo Prior SPIRE
Road	●	○		○				
Change					●			
Grass		●		●		●		●
Cloud	●		●					
SNR				●				

Table 7.2. *Failure elimination table for classifier performance.*

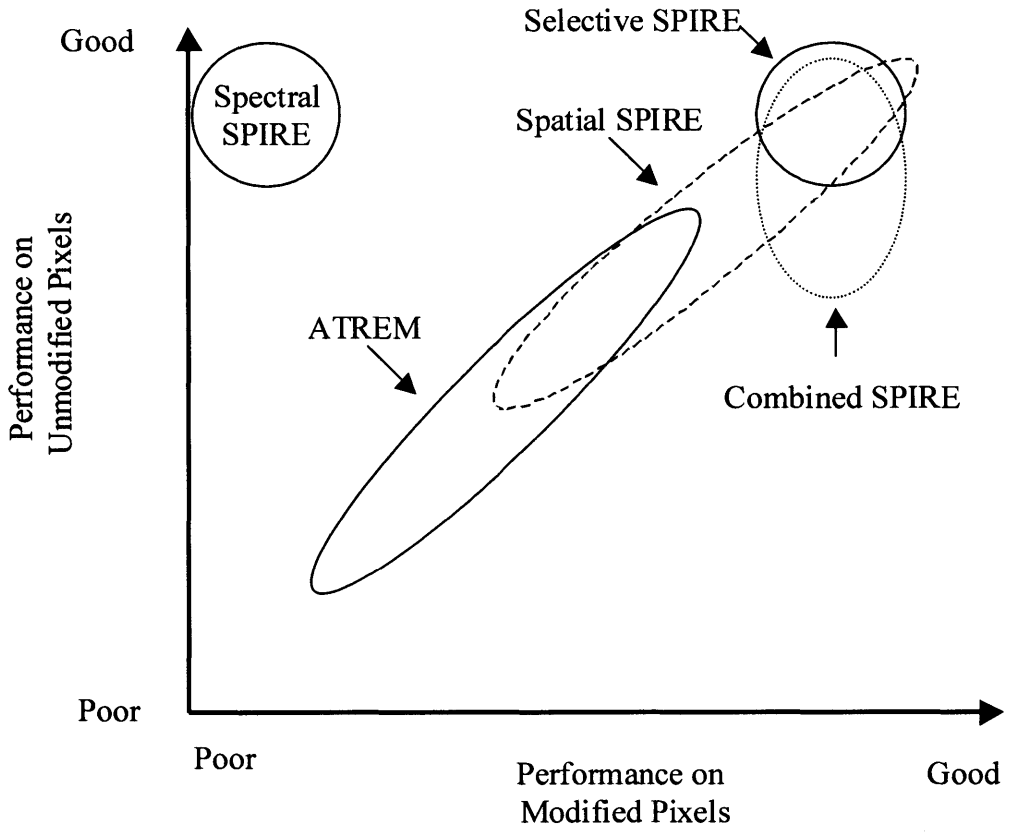


Figure 7.15. *Relative performance of reflectance estimators. This qualitative plot assumes operational conditions with sensor noise, prior noise, registration errors, and low SNR conditions present. Also, overhead clouds may be present, and no calibrated panels are in the scenes, eliminating ELM as an option. ATREM performs poorly when clouds are present, and Spatial SPIRE performs poorly on low reflectance materials due to low SNR.*

These results confirm the individual SPIRE performance conclusions reached in Chapters 3, 5, and 6 reviewed and summarized in Section 7.3. In addition, the consistent performance of the Selective SPIRE technique indicates that it is the best choice for classification applications. For applications where Pseudo Prior SPIRE is appropriate, classification and change detection can be done without relying on ELM or ATREM to generate a suitable prior reflectance.

Finally, Figure 7.15 qualitatively plots the performance of each of the reflectance estimation algorithms investigated in this thesis on both modified and unmodified pixels. This

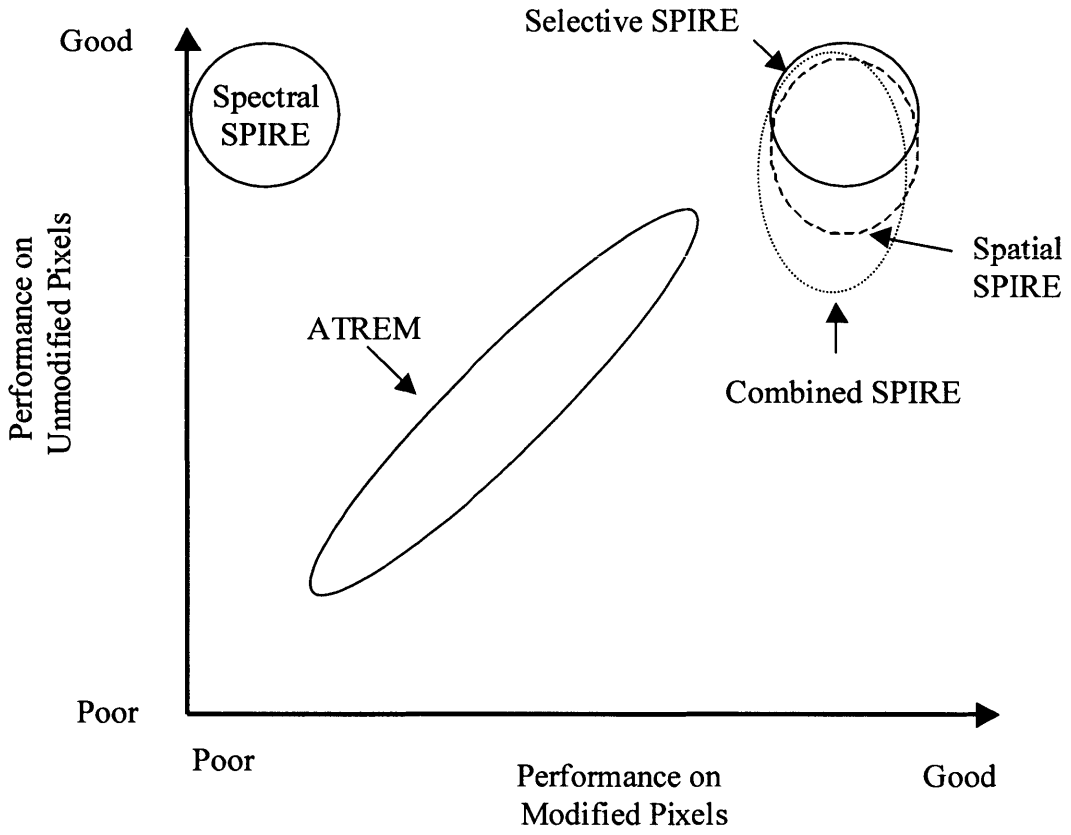


Figure 7.16. *Expected relative performance of reflectance estimators under moderately high SNR conditions. Assumptions for this plot are the same as for Figure 7.15 except that moderately high SNR is present in all channels. The higher SNR conditions lets Spatial SPIRE perform better than Combined SPIRE on modified pixels. Selective SPIRE would therefore use Spatial SPIRE estimates for modified pixels.*

plot assumes real world conditions where sensor noise, prior noise, registration errors, and low SNR conditions are present, based on our experience with the HYDICE test data set used in this thesis. We also assume that we are dealing with an operational airborne remotes sensing scenario, in which some scenes will have overcast skies and calibrated test panels are not available. This last condition means that ELM is not an option so it is not included in the plot.

In Figure 7.15 Selective SPIRE (using Combined SPIRE) is plotted as the best for both modified and unmodified pixels, with Combined SPIRE second overall. Spatial SPIRE works

well on higher reflectivity materials, but poorly on lower reflectivity ones because of SNR problems. Spectral SPIRE only works well on unmodified pixels and ATREM exhibits poor performance on all pixels if there is overhead clouds or haze, but can exhibit good performance on clear days. Under certain conditions the relative positions in Figure 7.15 can change. For example, the absence of low-SNR conditions would place Spatial SPIRE slightly better than Combined for modified pixels and just below Selective SPIRE for unmodified ones. Figure 7.16 shows a similar plot of expected relative performance if moderately high SNR is present in all channels. Under these conditions, Spatial SPIRE performs better than Combined on modified pixels so that Spatial SPIRE would be used by Selective SPIRE instead of Combined on modified pixels.

7.5 SPIRE Algorithm Recommendations for Different Applications and Conditions

In this section we provide guidance on which SPIRE algorithm to use for different applications and under different conditions. The Spectral SPIRE algorithms are not recommended due to their poor performance on modified pixels, except when used in the Selective SPIRE technique.

For a hyperspectral sensor, Table 7.2 supports the recommendation of using Selective SPIRE to generate the best possible reflectance estimates for all pixels. This is true for any application requiring estimates of reflectance, including classification and change detection. The Selective SPIRE technique can be used with either Spatial or Combined SPIRE, depending on the presence of low SNR conditions and the importance of computational cost. We recommend the use of the Selective SPIRE technique utilizing Combined SPIRE for change detection and estimation of modified pixel reflectance, with Spectral SPIRE estimates for unmodified pixels.

This algorithm is computationally efficient and robust under noise and low SNR. These recommendations are summarized in Table 7.3. For hyperspectral change detection or classification applications where no prior information is available, the use of a pseudo prior with SPIRE is recommended.

Computational Costs Importance	Low SNR Conditions	Change Detection	Modified Pixel Reflectance	Unmodified Pixel Reflectance
High	N/A	Combined	Combined	Spectral
N/A	Present	Combined	Combined	Spectral
Medium	Absent	Combined	Spatial	Spectral
Low	Absent	Spatial	Spatial	Spectral

Table 7.3. *Recommended SPIRE algorithms to use in Selective SPIRE depending on computational cost and SNR.*

For single channel and multispectral applications, only Spatial SPIRE algorithms are appropriate, since there is not enough spectral information to utilize the PC techniques of Spectral and Combined SPIRE. Computational cost is less of an issue in such applications, since the number of channel images to be processed is small. However, low SNR channels should be avoided in the design of such sensors if Spatial SPIRE algorithms are to be used with the data.

For classification applications under the stressing conditions of no available prior reflectance information, no calibrated panels in the scene, and either overhead clouds or multispectral data (so physics-based techniques cannot be used), only SPIRE run with a pseudo prior reflectance will allow successful cross-image classification.

When using only Spatial or Combined SPIRE techniques, there are several algorithms from which to choose based on the spatial variability of the multiplicative m and additive a noise terms. Table 7.4 lists various applications and conditions, and which Spatial or Combined algorithms are recommended based solely on the spatial variability of m and a. Under the

Application	Conditions	m	a	Recommended Algorithm	Comments
Remote Sensing	Small scene size relative to atmosphere	Uniform	Uniform	Spatial Case 2	Spatial Cases 4, 5, and 6 will work too
Remote Sensing	Large scene size relative to atmosphere	Varying	Uniform	Spatial Case 4 Combined Case IV	Spatial Case 6 will work too
Remote sensing	Significant spatial variability in upscatter	Varying	Uniform	Spatial Case 6	Combined Case VI could be developed
Manufacturing or inspection	Controlled lighting	Uniform	None	Spatial Case 1	Typically 1-3 spectral channels
Mobile robotics	Unknown lighting	Varying	None	Spatial Case 3 Combined Case III Spectral Case D with library matching	Shadows often present, could be handled by Spectral Case D

Table 7.4. *Recommended type of Spatial and Combined SPIRE algorithm based on spatial variability of m and a.*

Prior Information	Spatial SPIRE	Spectral SPIRE	Combined SPIRE	Source
Prior Reflectance	r_p	r_p	r_p	ELM, ATREM, Simulation, Pseudo Prior
Spatial variability of a	ϕ_a	-	ϕ_a	Scene Size, Application
Spatial variability of log m	$\phi_{\log m}$	-	$\phi_{\log m}$	Scene Size, Application
PC cutoff for a	-	$pc_{co\text{-}\mathbf{a}}$	$pc_{co\text{-}\mathbf{a}}$	Empirical ensemble {**a**}
PC cutoff for log m	-	$pc_{co\text{-}\log\mathbf{m}}$	$pc_{co\text{-}\log\mathbf{m}}$	Empirical ensemble {**m**}
PC rotation into a-PCs		$\mathbf{\Phi_a}$	$\mathbf{\Phi_a}$	Empirical ensemble {**a**}
PC rotation into log m-PCs		$\mathbf{\Phi}_{\log\mathbf{m}}$	$\mathbf{\Phi}_{\log\mathbf{m}}$	Empirical ensemble {**m**}

Table 7.5. *Prior Information required by SPIRE algorithms.*

Conditions heading, the size of the scene "relative to the atmosphere" is defined as whether or not the size of the scene is small or large relative to the atmosphere induced spatial variation of m and a. If the scene is small, then we can assume that m and a are spatially uniform. If the scene is large, then we should assume that m is spatially varying. In most airborne or spaceborne

remote sensing applications, a can be assumed to be spatially uniform, though some scenes may have varying upscatter due to smoke or transitions from rural to urban areas in the scene. Applications where there are cloud or other terrain shadows are not addressed. See Chapter 8 for suggestions of how SPIRE algorithms may be extended to handle such applications. Finally, Table 7.5 lists the types of prior information required by each type of SPIRE algorithm.

7.6 Summary and Conclusions

In this chapter we reviewed the overall taxonomy of SPIRE algorithms that were developed as part of this thesis work and reviewed the relative performance of Spatial, Spectral, and Combined SPIRE algorithms. We then developed the Selective SPIRE technique that combines the good performance of Spatial and Combined SPIRE on modified pixels with Spectral SPIRE's good performance on unmodified pixels, while maintaining computational efficiency. We also explored using a pseudo prior as input to Spatial SPIRE when no prior reflectance is available, to generate pseudo reflectances that can be used in change detection and classification applications.

The minimum distance classification experiments demonstrated that reflectance estimates from SPIRE algorithms allow cross-image classification in situations where ATREM reflectance estimates do not. This means that SPIRE algorithms allow classification to be done under operational conditions where both physics-based (overhead clouds) and ELM (no calibrated test panels) fail. Use of a pseudo-prior with SPIRE algorithms further extends the operational use of SPIRE into applications and conditions where no prior reflectance information is available. This is especially important for single channel and multispectral sensor systems where physics-based approaches are unable to generate prior reflectance data, and calibrated test panels are not typically present in the scene.

Chapter 8

Summary, Conclusions, and Suggestions for Further Work

In this chapter, we summarize the work presented in this thesis, discuss the conclusions reached, and make suggestions for possible further work.

8.1 Summary

In this thesis we addressed the problem of estimating changes in surface reflectance in hyperspectral image cubes, under unknown multiplicative and additive illumination noise. Rather than using the Empirical Line Method or the physics-based approach discussed in Chapter 2, we assumed the presence of a prior reflectance image cube and ensembles of typical multiplicative and additive illumination noise vectors, and developed algorithms which estimate reflectance using this prior information. These algorithms were developed under the additional assumptions that the illumination effects were band limited to lower spatial frequencies and that the differences in the surface reflectance from the prior were small in area relative to the scene, and have defined edges. These new algorithms were named Surface Prior Information Reflectance Estimation (SPIRE) algorithms.

In Chapter 3, we developed Spatial SPIRE algorithms for six cases defined by the presence or absence of the additive noise, and by whether or not the noise effects are spatially uniform or varying. For the multiplicative-noise-only cases, high-pass spatial filtering was used in log space to remove the illumination effects, followed by restoration of the lost low spatial frequency reflectance components using the prior reflectance. For cases with both additive and multiplicative noise, two-step iterative algorithms filter out the additive noise, move to log space, and then filter out the multiplicative noise, using a mean squared error (MSE) criterion to drive

the iteration to convergence. For the cases where the additive and multiplicative illumination noises are uniform across the scene, non-iterative solutions were developed that are very computationally efficient. With good SNR, Spatial SPIRE algorithms demonstrate very good performance in estimating the reflectance of both unmodified and modified pixels where local surface reflectance changes have occurred since the prior was collected. They operate on a single band basis, so are also applicable to single and multi-channel sensor data. The Spatial SPIRE Case 4 algorithm tested on the HYDICE test data set has problems estimating the reflectance of low reflectance materials under low SNR conditions.

In Chapter 4, we analyzed the degrees of freedom in the empirical HYDICE multiplicative and additive illumination noise ensembles, and compared them to the degrees of freedom in simulated noise ensembles generated using MODTRAN. We also performed a similar analysis on the reflectance spectra present in the HYDICE data used in our algorithm testing. We explored the need for zero-padding each illumination noise ensemble to force the DC component to be a degree of freedom so that it ends up in a top PC for effective removal. We also compared this Zero-padded PC technique to the Abutted-PC technique which does not require illumination ensembles to collect the illumination noise into top PCs. We discussed the effect of channel center wavelength drift at the edges of water vapor absorption bands on the principal components analysis and degrees of freedom in the ensembles. We also presented which spectral channels were dropped from the HYDICE data because of water vapor absorption, negative radiance values, and channel center wavelength drift.

In Chapter 5, we developed Spectral SPIRE algorithms that use low-pass Principal Component filters (PCF) to remove the illumination noises, followed by restoration of the lost high PC reflectance components using the prior reflectance. As in Chapter 3, this was done in

log space for the case where only multiplicative noise is present. When additive noise is also present, a two step iterative algorithm similar to that developed for Spatial SPIRE was developed. Such PCF based spectral SPIRE algorithms require significantly less computation than their Spatial SPIRE counterparts, since only those top PCs containing the bulk of the noise need to be processed. This, however, comes at the cost of poor performance in estimating the reflectance of modified pixels, while exhibiting superior performance in estimating the reflectance of unmodified pixels. We also briefly discussed an algorithm that uses the abutted-PC technique and matching against a spectral library to solve a multiplicative-noise-only problem with large area changes and quickly varying illumination noise.

In Chapter 6, we developed a combined spatial-spectral SPIRE algorithm that takes advantage of Spectral SPIRE's reduction in computational costs and Spatial SPIRE's ability to estimate the reflectance of modified pixels. We can view Combined SPIRE algorithms as being very similar to Spatial SPIRE algorithms, but with a reduction in problem complexity achieved by spectrally rotating the data so that only a few channel (PC) images need to be spatially filtered. These Combined SPIRE algorithms have nearly as good performance as Spatial SPIRE under good SNR conditions on all pixels, and do better than Spatial SPIRE on low reflectance materials during low SNR conditions. Combined SPIRE algorithms are also almost as computationally efficient as Spectral SPIRE, making them superior to Spatial SPIRE when low SNR conditions exist.

In Chapter 7, we reviewed the overall taxonomy of SPIRE algorithms that were developed as part of this thesis work and the relative performance of the individual Spatial, Spectral, and Combined algorithms. Based on these results, we developed a Selective SPIRE technique that combines the superior performance of Combined SPIRE on modified pixels, with

Spectral SPIRE's superior performance on unmodified pixels. We also explored using a pseudo prior as input to Spatial SPIRE when no prior reflectance is available, to generate pseudo reflectances that can be used in change detection and classification applications. We then performed minimum distance classification experiments on radiance cubes and reflectance estimated from ELM, ATREM, and the various SPIRE algorithms. All SPIRE algorithms were found to work better than ATREM with the Selective SPIRE algorithm having the best classification performance overall. Recommendations were made for which SPIRE algorithm to use for different applications and conditions.

8.2 Conclusions

8.2.1 Comparison of SPIRE to ELM and Physics Based Approaches

Treating ELM results as ground truth, we found that SPIRE algorithms perform better than the ATREM physics-based code, especially on days when clouds or haze were present above the sensor. SPIRE performance was similar to ELM's, without requiring calibrated test panels in the scene. However, SPIRE algorithms require a prior reflectance image cube in order to estimate reflectance in the current image. Like ELM, SPIRE algorithms are insensitive to calibration noise and errors, so that SPIRE algorithms work just as well on uncalibrated sensor data, while physics-based codes require accurate radiometric calibration for acceptable performance.

For hyperspectral applications we recommend the use of the Selective SPIRE technique utilizing Combined SPIRE for change detection and estimation of modified pixel reflectance, with Spectral SPIRE estimates for unmodified pixels. This algorithm is computationally efficient and robust under sensor noise, prior noise, misregistration, and low SNR.

In classification experiments, all SPIRE algorithms performed significantly better than ATREM. Classification broke down completely in ATREM estimated reflectance image cubes due to ATREM's poor performance when overhead clouds or haze was present. Combined and Selective SPIRE performed slightly better than ELM in classifying grass on lower SNR data, due to SPIRE's ability to reduce noise across images via use of the prior reflectance. For classification applications under the stressing conditions of no available prior reflectance information, no calibrated panels in the scene, and either overhead clouds or multispectral data (so that physics-based techniques are not an option), only SPIRE run with a pseudo prior reflectance will allow successful cross-image classification. Running the Spatial Case 2 algorithm with a pseudo prior was found to have identical classification performance as running the Spatial Case 2 algorithm with an ELM based prior.

8.2.2 Comparison of Spatial, Spectral, and Combined SPIRE Algorithms

For hyperspectral data, Spatial SPIRE is computationally expensive, but performs the best on both modified pixels of moderate to high reflectance materials, and performs better than Combined SPIRE on moderate to high reflectance pixels. However, Spatial SPIRE has problems estimating the reflectance of low reflectivity materials in images with low SNR in the longer wavelength channels.

Spectral SPIRE is computationally very efficient when compared to Spatial SPIRE, and performs better on unmodified pixels than Spatial SPIRE or Combined SPIRE, because the reflectance information replaced from the prior reduces the overall noise in the Spectral SPIRE reflectance estimates. However, for modified pixels, this replacement introduces large reflectance errors since changes in reflectance and the multiplicative noise are not well separated

in PC space.

Combined SPIRE performs nearly as well as Spatial SPIRE on both modified and unmodified pixels, and is far less sensitive to low SNR conditions. Combined SPIRE is also nearly as computationally efficient as Spectral SPIRE, and performs nearly as well on unmodified pixels as Spectral SPIRE. These attributes make Combined SPIRE the best compromise between computational efficiency and performance. However, if high SNR conditions are present, then Spatial SPIRE will out-perform Combined SPIRE on both modified and unmodified pixels, at the expense of higher computational costs.

The Selective SPIRE technique combines the superior performance of Combined (or Spatial) SPIRE on modified pixels with the superior performance of Spectral SPIRE on modified pixels. This results in the best possible estimates of reflectance.

8.2.3 General Applicability of SPIRE Algorithms

For single-band sensor data only Spatial SPIRE is applicable. For multispectral sensor data in which there is little or no correlation between channels, Spatial SPIRE is still the logical choice since the processing and noise reduction from a PCA rotation will be small.

In applications where both m and a are spatially uniform in all channels, the non-iterating Spatial Case 1 and Case 2 algorithms yield computationally efficient solutions with good accuracy. However, these could still benefit from being used in the Selective SPIRE technique if Spectral SPIRE estimates are available.

For practical hyperspectral applications involving change detection, Spectral SPIRE alone would never be used due to its poor performance on modified pixels. When computational costs are an issue, the Selective SPIRE algorithm using Combined and Spectral SPIRE algorithms is typically the best choice. If computational costs are not an issue, and low SNR

conditions are not present in the data, then a Selective SPIRE algorithm using Spatial instead of Combined SPIRE algorithms would be the best choice.

8.2.4 Degrees of Freedom of m, a, and r in HYDICE Data

The empirical ensembles of **m** and **a** used in this thesis were derived from several HYDICE data collection campaigns over varied conditions. It was found that there were 2-3 DOF in the additive noise **a** ensemble and 3-4 in the multiplicative noise **m** ensemble, both of which are consistent with MODTRAN simulation results and consideration of the physics involved. Zero-padded or Abutted Principal Components is required to force the DC of each ensemble into the lowest PCs so that they could be removed by replacing or filtering as few PCs as possible.

Channel center frequency drift in the HYDICE sensor causes large changes in response in those channels near the edges of water absorption bands. This showed up as an additional degree of freedom in the **m** and **a** ensembles data during PC analysis. By eliminating these channels from the **m** and **a** spectra, fewer PCs need to be processed to achieve good performance, since one PC was associated in each ensemble with the effects of this drift. Future sensor designers should endeavor to avoid such drift.

8.2.5 Noise Performance

In general, SPIRE algorithms degrade gracefully under increasing levels of sensor and prior noise. Any adverse effects from sensor noise can typically be overcome through adequate SNR. In addition, increasingly larger violations of the assumption that the high spatial frequency content of small area reflectance changes and the low frequency content of the slowly varying illumination noise do not overlap in frequency space, also induce graceful performance

degradations.

SPIRE algorithms are insensitive to radiometric calibration noise. This assumes that any calibration noise is of low spatial frequency within the band limits of the illumination noise, so that it is filtered out with the illumination noise. This means that once a prior is available, SPIRE algorithms can be applied directly to uncalibrated sensor data, greatly simplifying operational data processing. For this to be true for Spectral and Combined SPIRE algorithms, the **m** and **a** spectra must be empirically derived from data the same level (amount of calibration done) of data that is to be processed by SPIRE, so that the calibration noise is present and placed in the top PCs to be processed.

8.3 Suggestions for Further Work

8.3.1 Optimal Estimation and Filtering Techniques

We noted in Section 2.5.3 that there are many well established estimation techniques that rely on the statistical relationship between a measured signal and the desired parameter vector or random vector to be estimated. Such techniques include Wiener filters (Linear Least-Squares estimators), Maximum Likelihood estimators, and Bayesian Least Squares estimators. These techniques could be applied to the problem of estimating the reflectance r based on the radiance signal i. This requires the generation of a measurement model that statistically characterizes r and i. Such a measurement model is difficult to generate for the non-linear remote sensing problem, but could be done through an extensive measurement campaign, simulation, or a combination of the two. There may be successful strategies that combine the results of such estimators with SPIRE algorithms to improve performance overall. Other techniques applicable to non-linear problems such as Neural Networks and Expectation-Maximization (EM) may also

prove useful.

All three of the multiplicative-noise-only algorithms (Spatial Case 3, Spectral Case C, and Combined Case III) share the same general form in which the log m noise is first filtered out, and then the log r lost in this filtering is estimated from prior information and restored. In the algorithms developed and tested in this thesis, simple linear filters were used to both remove the noise and to estimate the lost signal from the prior, and a simple addition operation was used to merge the two. These simple techniques generated good results and demonstrated the utility of the SPIRE approach. SPIRE algorithms could therefore be improved by using optimal noise filtering, optimal estimation of lost r, and optimally weighted merging strategies.

Optimal spatial filters could be developed for filtering out illumination noise, including situations where the illumination noise and reflectance changes have some overlap in the spatial frequency domain. In PC space where there currently exists significant overlap, optimal PC filters that weight each PC differently are worth exploring. Statistical estimation techniques such as those described above for estimating r based on i, could also be used to estimate the lost log r (lost during the noise filtering) from the prior reflectance. Optimal weighting techniques for merging the filtered image with the estimated lost log r would be embodied in a combination of weighting strategies in the noise filtering, lost-log-r estimation, and merge operations.

8.3.2 Spectral Rotation Matrices for Separating Illumination Noise and Changes in Reflectance

While Zero-padded PC analysis effectively collects the illumination noise into a few low PCs for removal, it does not exclude changes in reflectance from these PCs. If a spectral transformation could be developed that more cleanly separates reflectance and illumination variations, then Spectral and Combined SPIRE would benefit. Since the ZPC analysis was done

only using the noise vector, strategies that also include likely or possible changes in surface reflectance may improve the separation. Modified Principal Components Analysis or Independent Components Analysis may be useful for defining such rotations.

8.3.3 Prior Information Generation and Collection

One of the main foundations of the work in this thesis is the assumption that prior estimates of reflectance and prior ensembles of noise vectors will be available for use in SPIRE processing. Given the encouraging results of this thesis, further work could be done into this area, working on the left side of Figure 2.8.

8.3.4 Atmospheric Spatial Filter Development

Since our HYDICE test image cubes did not contain illumination noises with significant spatial variability, very simple spatial filters proved effective. For applications and data with noises that have more spatial variability, work would need to be done to characterize the spatial frequency content of the illumination variations and design optimum filters to remove them.

8.3.5 Use of Low Spatial Resolution Prior Reflectance Data

Since Spatial and Combined SPIRE algorithms restore only the low spatial frequency components from the prior reflectance, it may be possible to use prior reflectance information that is of lower spatial resolution than the image being processed. This could have operational benefits in that the sensor system used to generate prior information would not need as high spatial resolution as the operational sensor on whose data we run SPIRE. For example, we could imagine using a satellite with a hyperspectral sensor with the high spectral resolution needed for physics-based reflectance estimation, but only the low spatial resolution needed for the SPIRE prior. The operational sensor could then be a low-altitude multispectral sensor with low spectral

resolution but high spatial resolution. Since there is often a trade between spatial and spectral resolution in sensor design, this could result in an optimized system.

8.3.6 Extension of SPIRE to Rapidly Varying Illumination Effects and Large Area Changes

In this thesis, we concentrated on applications where the spatial variability of the noise terms was band limited to lower frequencies and in which changes in the surface reflectance were small in relation to the area of the scene. It is important to note that both the Spectral Case C and Case D algorithms will work even if the multiplicative noise **m** is not band limited to lower frequencies, which is not reflected in the SPIRE taxonomy of Figure 7.1. In addition, their poor performance on modified pixels can potentially be improved by matching against a prior spectral library. However, as discussed in Chapter 5, such matching begins to enter the area of classification of reflectance values which, except for the simple classification example discussed in Chapter 7, is beyond the scope of this thesis. Further work exploring the large space of possible classification strategies should consider it as a valid possibility.

While the Spectral Case D algorithm requires that changes in reflectance since the prior are small with respect to the area of the scene, Case C does not. Therefore, in applications where there is no additive noise, Case C has the potential of being a general solution algorithm that can handle arbitrarily varying multiplicative noise such as shadows, and large area reflectance changes. This, however, requires matching against a spectral library to restore log reflectance lost in the PC filtering. This approach is applicable to problems in mobile robotics in which there is no appreciable additive noise, but it was not appropriate for the airborne remote sensing application explored in this thesis. In addition, this approach could be used in other applications if the additive noise is removed using some other technique.

In most practical applications where the illumination noises can be considered to be arbitrarily spatially varying (have high spatial frequency content), they tend to vary due to shadows which introduce sharp boundaries in the radiance image. In the regions between these boundaries the noises will tend to be slowly spatially varying or uniform. This suggests an approach in which the image is spatially segmented at sharp boundaries and then one of the Spatial SPIRE algorithms run within the segments. In other words, in an image with large shadowed and lit areas, one would run Spatial SPIRE independently within each area, and then combine the results. One may even be able to use the prior reflectance to discriminate between boundaries in the reflectance from shadow induced boundaries.

8.3.7 Matching Against a Prior Spectral Library

Spectral SPIRE demonstrates poor performance over modified pixels. However, matching the non-noise PCs against the same PCs of a prior spectral library could improve the estimates of the changes in reflectance, provided the library contains the new spectra. This is computationally expensive if the prior spectral library is large, which will be the case if fine resolution of many permutations of mixed material pixels are required to be in the library. Further work is required to understand if such an approach is feasible when realistic sized spectral libraries are employed.

8.3.8 Combined/Spectral SPIRE Testing on Spatially Slowly Varying Illumination Data

Since the HYDICE images used in this thesis for testing contained spatially uniform multiplicative and additive illumination effects, they did not provide an opportunity to test spatial filtering beyond mean filtering. Further testing and development of SPIRE algorithm on datasets containing spatially slowly varying illumination would be a logical next step.

8.3.9 Classification and SPIRE

Our Chapter 7 classification example leads to many suggestions for further work in using SPIRE reflectance estimates in classification:

- Classification could be done on just the higher non-log **m** PCs created by a high-pass PC filter. This would be equivalent to running Spectral SPIRE, then ignoring the low PC into which prior information was replaced.
- Develop optimum techniques for setting the threshold in Selective SPIRE based on the classifier used.
- Test the use of pseudo priors with Spectral, Combined, and Selective SPIRE.
- Research how to handle modified pixels that fall below the distance threshold in Selective SPIRE and are mislabeled as unmodified. This could be either a more sophisticated change detection algorithm or a classifier trained to identify them.

8.3.10 Combining SPIRE and Physics Based Algorithms

Since SPIRE algorithms require iteration to eliminate additive noise effects, a fruitful research area may be to use physics based approaches to only estimate the additive upscatter effects, and then use a multiplicative-noise-only SPIRE algorithm to remove the multiplicative noise. In many applications it may be possible to use a look-up table of precomputed additive noise vectors to choose from based on variables such as season, geography, altitude, and solar zenith angle.

8.3.11 Miscellaneous Suggestions

In this section we will discuss several varied suggestions for work that represent small extensions of the SPIRE algorithms developed.

- Develop Combined Case VI algorithm for spatially varying **m** and spatially varying **a**. This would require building DCT expansions of the multiplicative noise to remove in each processed PC, similar to the Spatial Case 6 algorithm. The Combined Case V algorithm would likely be a simplification of the Case VI.
- Develop other possible Selective SPIRE permutations such as choosing between Spatial and Combined reflectance estimates based on the material reflectivity of the Combined estimates. With such a strategy, higher reflectivity materials would use the Spatial estimates, while low reflectivity materials would use the Combined estimates. This would avoid the low-SNR problems of Spatial SPIRE on low reflectivity materials, while taking advantage of Spatial SPIRE's superior performance on higher reflectivity materials.

Appendix A

Multiplicative and Additive Atmospheric Effects In Visible-Near IR Ground Remote Sensing

In this appendix we give an example of how all the physical effects in a typical remote sensing scenario can be reduced to a single multiplicative and a single additive effect on the surface reflectance so that we can use the following image formation model:

$$i = rm + a \tag{A.1}$$

To do this we first derive the image formation equations for a spacebased Visible-Near IR ground remote sensing application as is done in (Schowendgert, 1997). In this application, the sensor is carried by a satellite above the atmosphere looking down at the surface of the Earth as depicted in Figure A.1(a). The sun is the only illumination source and there is an intervening atmosphere. In the visible-near IR bands, we can assume that all radiation that reaches the sensor is due to reflected and scattered solar illumination, and not due to surface thermal emissions.

There are three main components to the radiation reaching the sensor. The first is the unscattered, surface-reflected component described by:

$$L_{su}(x, y, \lambda) = \rho(x, y, \lambda) \frac{\tau_v(\lambda)\tau_s(\lambda)E^o(\lambda)}{\pi} \cos\left[\theta(x, y)\right] \tag{A.2}$$

$L_{su}(x, y, \lambda)$ is caused by the top of the atmosphere solar radiation $E^o(\lambda)$ passing through the atmosphere to the ground, and then being reflected into the field of view of the sensor, as depicted in Figure A.1(a). The downward solar path transmittance $\tau_s(\lambda)$ is a multiplicative

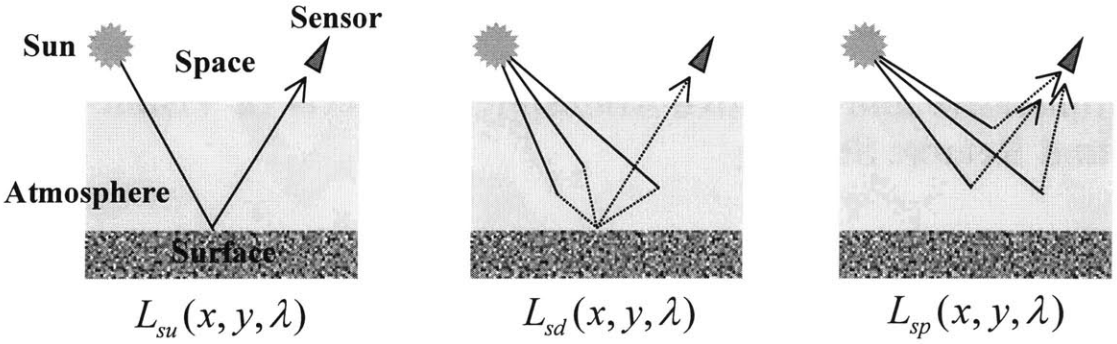


Figure A.1. *Three radiation components reaching sensor. The first component $L_{su}(x,y,\lambda)$ (a) is solar radiation that has been reflected by the surface into the sensor. The second component $L_{sd}(x,y,\lambda)$ (b) is the atmospheric downscattered radiation that has been reflected by the surface into the sensor. The third component $L_{sp}(x,y,\lambda)$ (c) is the atmospheric upscattered radiation that has not interacted with the surface but is scattered into the sensor. The first too effects are multiplicative noise on the surface reflectance signal while the third is additive noise.*

effect that describes the fraction of the solar radiation that reaches the surface, where any reductions are due to atmospheric absorption and scattering. Terrain effects embodied by $\cos[\theta(x,y)]$ alter the incoming radiation, which we assume is reflected by a Lambertian surface $(1/\pi)$ with diffuse spectral reflectance $\rho(x,y,\lambda)$. For a non-Lambertian surface, $\rho(x,y,\lambda)/\pi$ would be replaced by the BRDF function. The reflected radiation must once again pass through the atmosphere. The view-path transmittance $\tau_v(\lambda)$ describes the fraction of the reflected radiation reaching the sensor, with any reductions again due to atmospheric absorption and scattering.

The second radiation component reaching the sensor is the surface reflected-atmosphere scattered component:

$$L_{sd}(x,y,\lambda) = F(x,y)\rho(x,y,\lambda)\frac{\tau_v(\lambda)E^d(\lambda)}{\pi} \qquad \text{(A.3)}$$

$L_{sd}(x,y,\lambda)$ is caused by sunlight that is scattered downward (skylight) by the atmosphere and then reflected by the surface into the sensor field of view as depicted in Figure A.1(b). The skylight irradiance $E^d(\lambda)$ at the surface is also assumed to be reflected by a Lambertian surface with diffuse spectral reflectance $\rho(x,y,\lambda)$ and experiences reductions on the way to the sensor as described by the view-path transmittance $\tau_v(\lambda)$. The factor $F(x,y)$ accommodates the possibility that the sky may not be entirely visible by the surface pixel due to intervening topography.

The third radiation component reaching the sensor is the path-scattered component:

$$L_{sp}(x,y,\lambda) \tag{A.4}$$

$L_{sp}(x,y,\lambda)$ is the radiation at the sensor due to molecular Rayleigh scattering and aerosol and particulate Mie scattering. Both of these scattering effects scatter light into the sensor's field of view that has not been directly reflected by the surface, as depicted in Figure A.1(c).

Summing theses three components yields the total upwelling radiance at the sensor:

$$L_s(x,y,\lambda) = L_{su}(x,y,\lambda) + L_{sd}(x,y,\lambda) + L_{sp}(x,y,\lambda) \tag{A.5}$$

Which can be expanded using (A.2), (A.3), and (A.4) into:

$$L_s(x,y,\lambda) = \rho(x,y,\lambda)\left[\begin{array}{l}\dfrac{\tau_v(\lambda)\tau_s(\lambda)E^o(\lambda)}{\pi}\cos\left[\theta(x,y)\right] \\ +F(x,y)\dfrac{\tau_v(\lambda)E^d(\lambda)}{\pi}\end{array}\right] + L_{sp}(x,y,\lambda) \tag{A.6}$$

We now define effective multiplicative and additive effects $m(x,y,\lambda)$ and $a(x,y,\lambda)$ as:

$$m(x,y,\lambda) = \frac{\tau_v(\lambda)\tau_s(\lambda)E^o(\lambda)}{\pi}\cos\left[\theta(x,y)\right] + F(x,y)\frac{\tau_v(\lambda)E^d(\lambda)}{\pi} \tag{A.7}$$

$$a(x,y,\lambda) = L_{sp}(x,y,\lambda) \tag{A.8}$$

Substituting (A.7) and (A.8) into (A.6) gives us:

$$L_s(x,y,\lambda) = \rho(x,y,\lambda)m(x,y,\lambda) + a(x,y,\lambda) \tag{A.9}$$

Switching to our notation of i being the radiance at the sensor, r the surface reflectance, and dropping the three dimensional notation we obtain:

$$\begin{aligned} i(x,y,\lambda) &= r(x,y,\lambda)m(x,y,\lambda) + a(x,y,\lambda) \\ i &= rm + a \end{aligned} \tag{A.10}$$

Which has the same form as our desired model in (A.1).

Appendix B

Radiance and Prior Reflectance Single Channel Images

This appendix consists of Figures B.1 and B.2 which show single channel images from the radiance and prior reflectance images used to test SPIRE algorithms in this thesis. Figure B.1 shows the images from Runs 07, 13, and 26 while Figure B.2 shows them for Runs 06, 22, and 31. All of the images are from the same channel near 0.5 μm. All of the prior reflectance cubes were derived from the Run 07 ELM reflectance estimates. Images collected from higher altitudes have larger pixel sizes and appear less sharp.

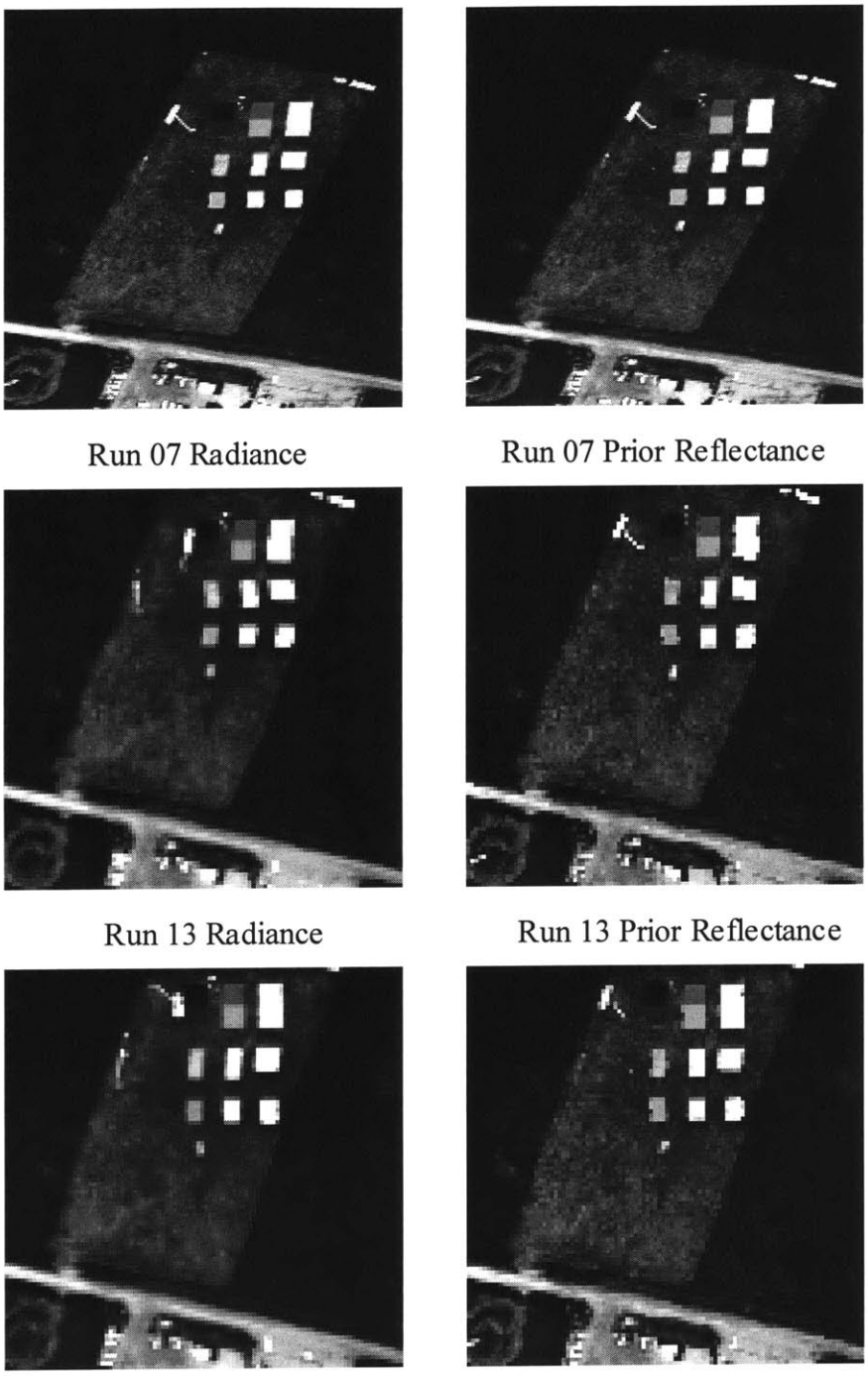
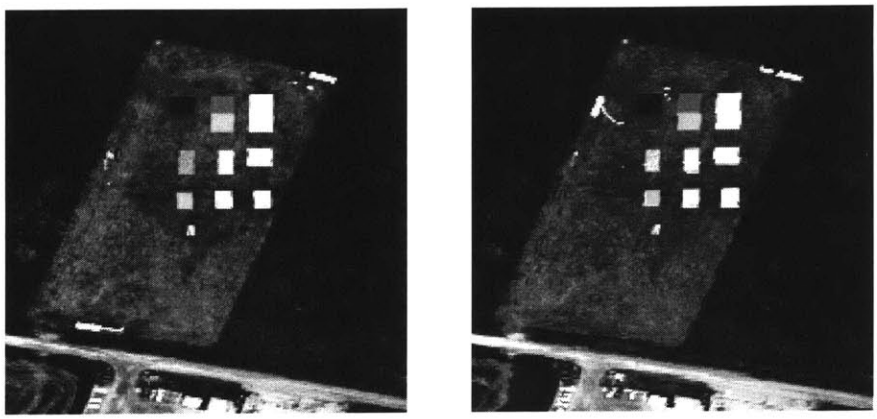
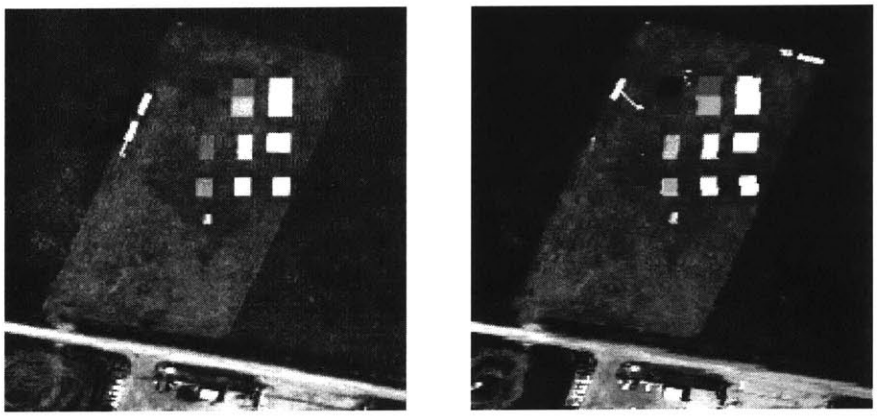


Figure B.1. *Radiance and prior reflectance single channel images for Runs 07, 13, and 26 from the HYDICE test data set used in this thesis. All images are from a channel near 0.5 μm. The prior reflectance images are all derived from the Run 07 ELM reflectance estimate.*



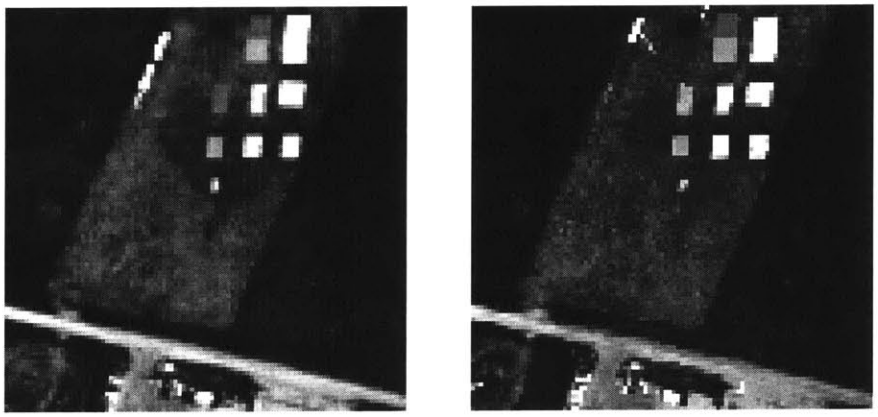
Run 06 Radiance

Run 06 Prior Reflectance



Run 22 Radiance

Run 22 Prior Reflectance



Run 31 Radiance

Run 31 Prior Reflectance

Figure B.2. *Radiance and prior reflectance single channel images for Runs 06, 22, and 31 from the HYDICE test data set used in this thesis. All images are from a channel near 0.5 μm. The prior reflectance images are all derived from the Run 07 ELM reflectance estimate.*

Appendix C

Additional Spatial SPIRE Results

This appendix consists of Figure C.1 and Tables C.1 and C.2. Figure C.1 is identical to Figure 3.8 that shows ELM, ATREM, and Spatial SPIRE reflectance estimates for the 19 example material pixels, except that the reflectance axes have a fixed range between 0 and 1. The reflectance axes in Figure 3.8 are set to the range appropriate for the data plotted to show the maximum amount of detail in each individual plot. The fixed range used in Figure C.1 is better suited for comparing the results of different pixels. Table C.1 lists the Spatial SPIRE average channel standard deviation over the six runs in each band for each algorithm and Table C.2 lists the same values but as a percentage of the mean estimated reflectance over the six runs.

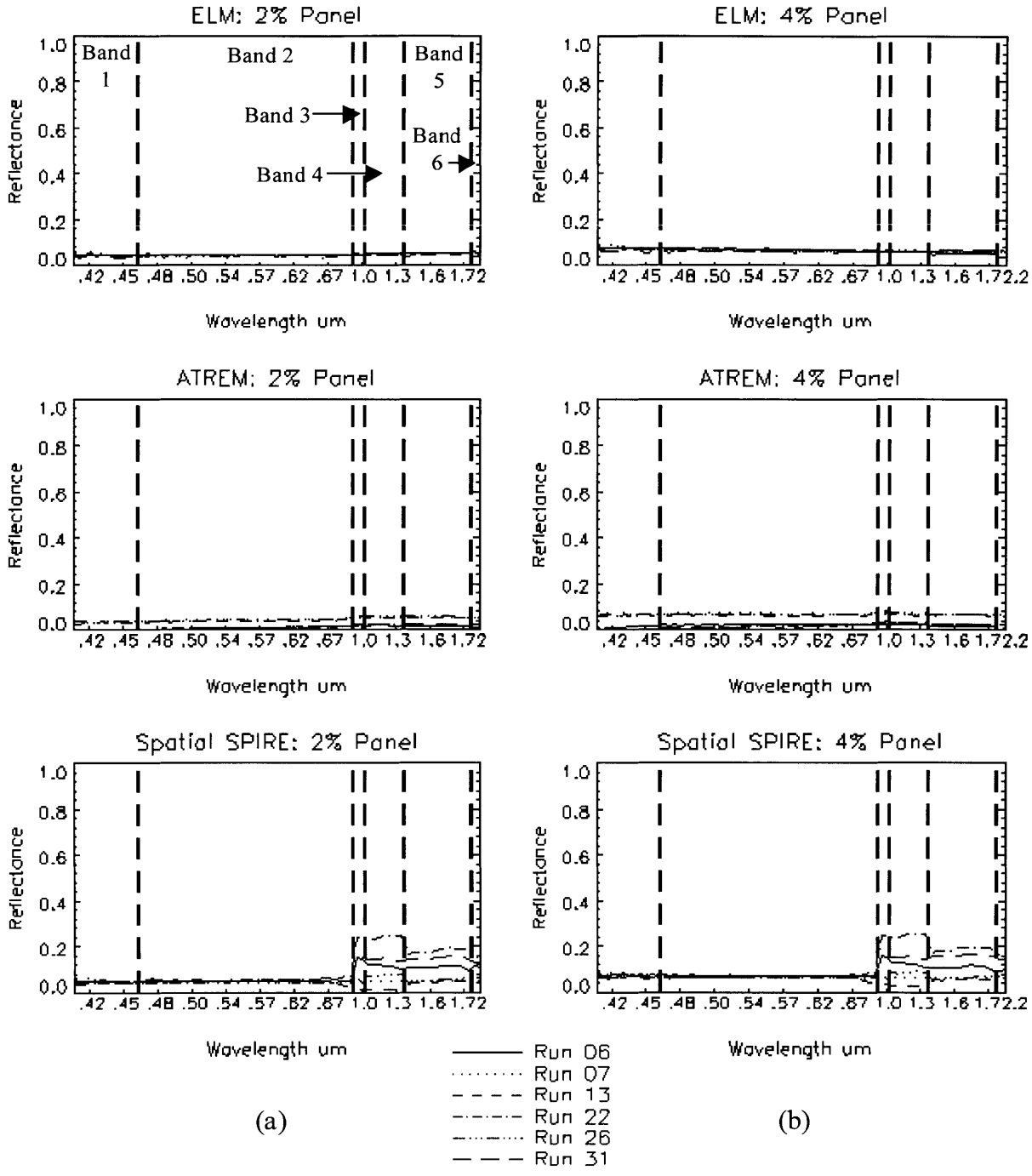


Figure C.1 (a) and (b). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the 2 percent panel (a) and the 4 percent panel (b). There are six bands of contiguous channels left after dropping problem channels, defined as Bands 1-6 as depicted in the upper left plot of ELM 2 percent panel reflectance. Spatial SPIRE's poor performance at longer wavelengths is due to errors in estimating the additive noise a under low SNR conditions.*

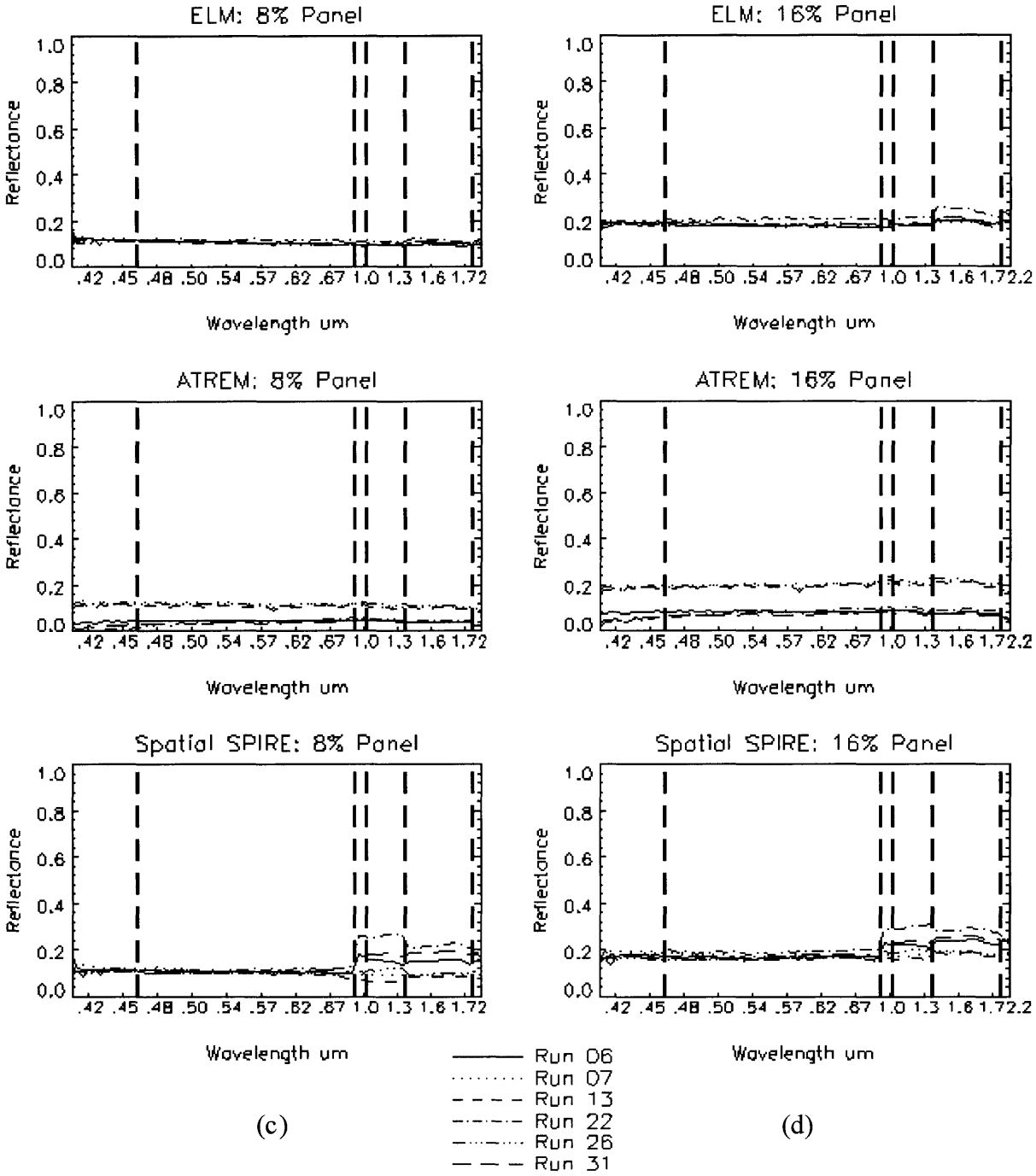


Figure C.1 (c) and (d). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the 8 percent panel (c) and the 16 percent panel (d).*

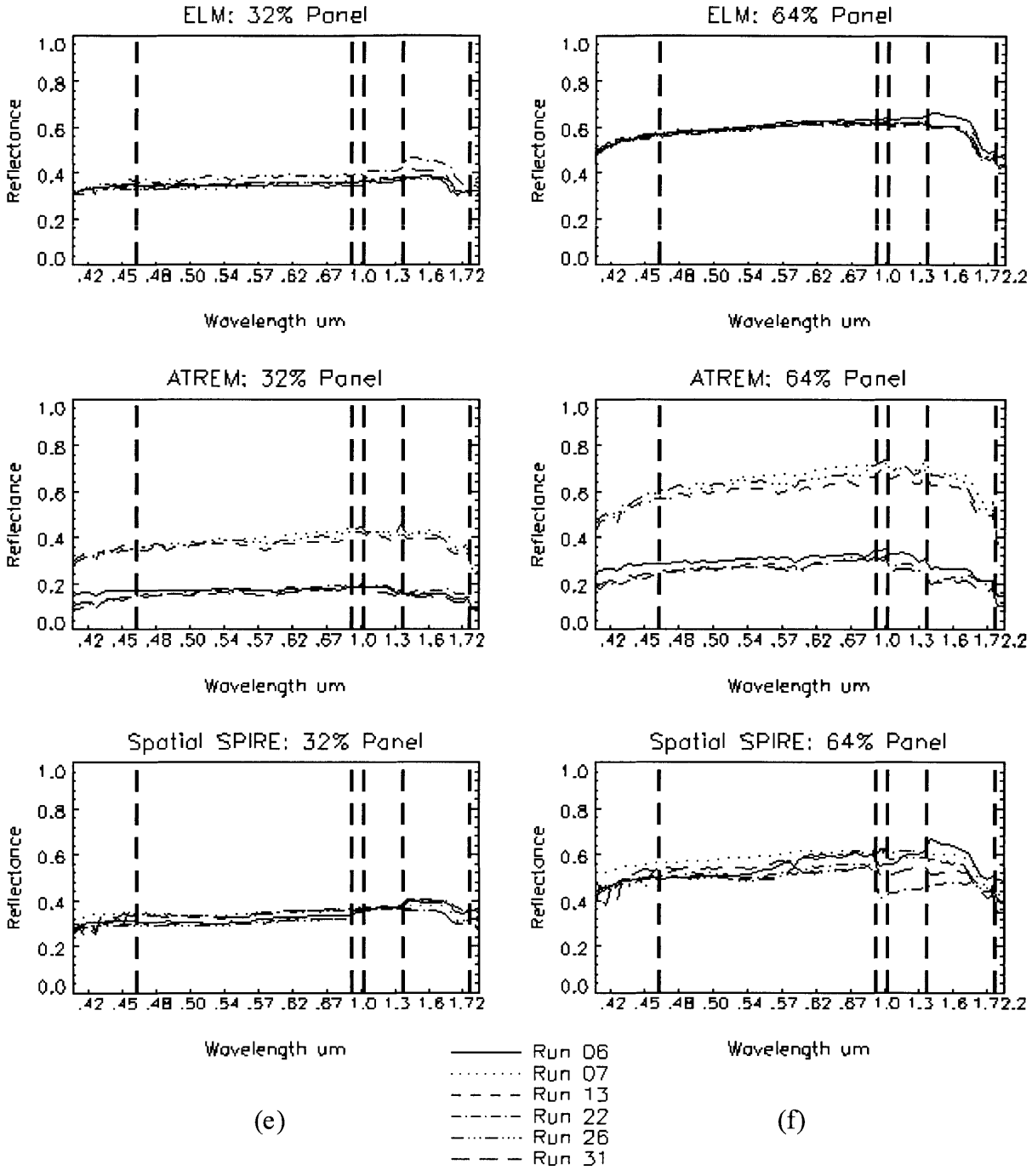


Figure C.1 (e) and (f). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the 32 percent panel (e) and the 64 percent panel (f).*

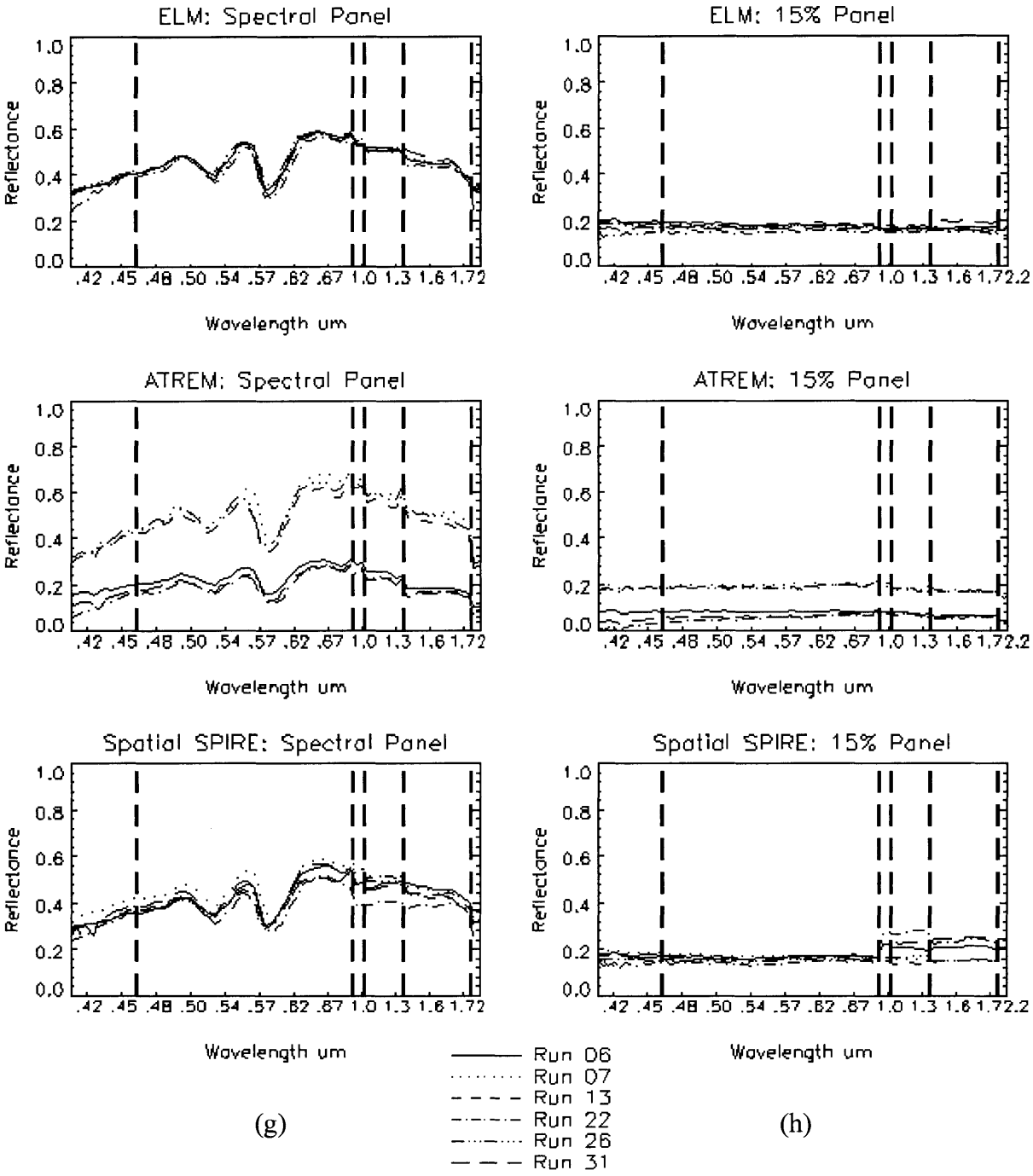


Figure C.1 (g) and (h). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the spectral panel (g) and the 15 percent panel (h).*

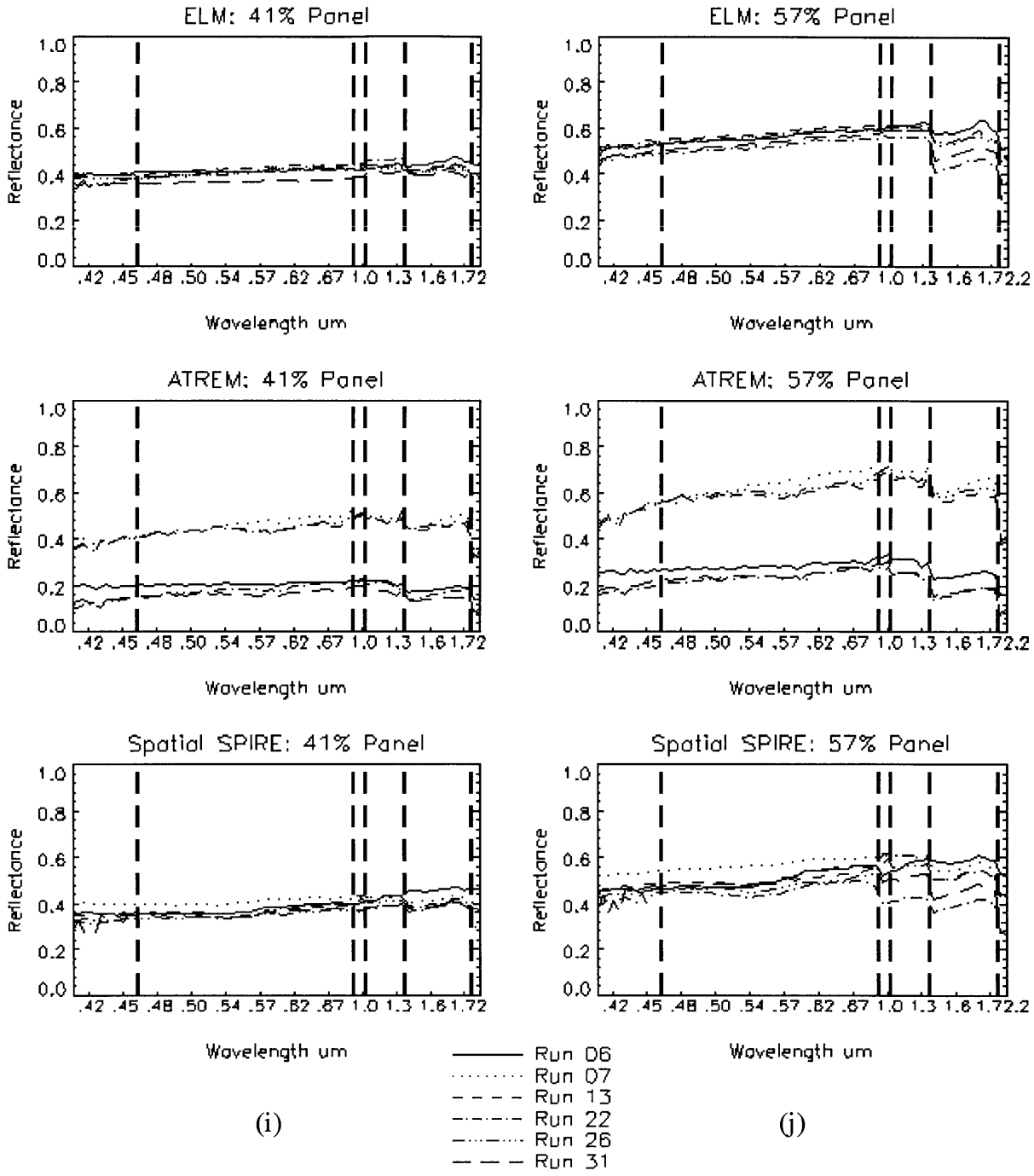


Figure C.1 (i) and (j). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the 41 percent panel (i) and the 57 percent panel (j).*

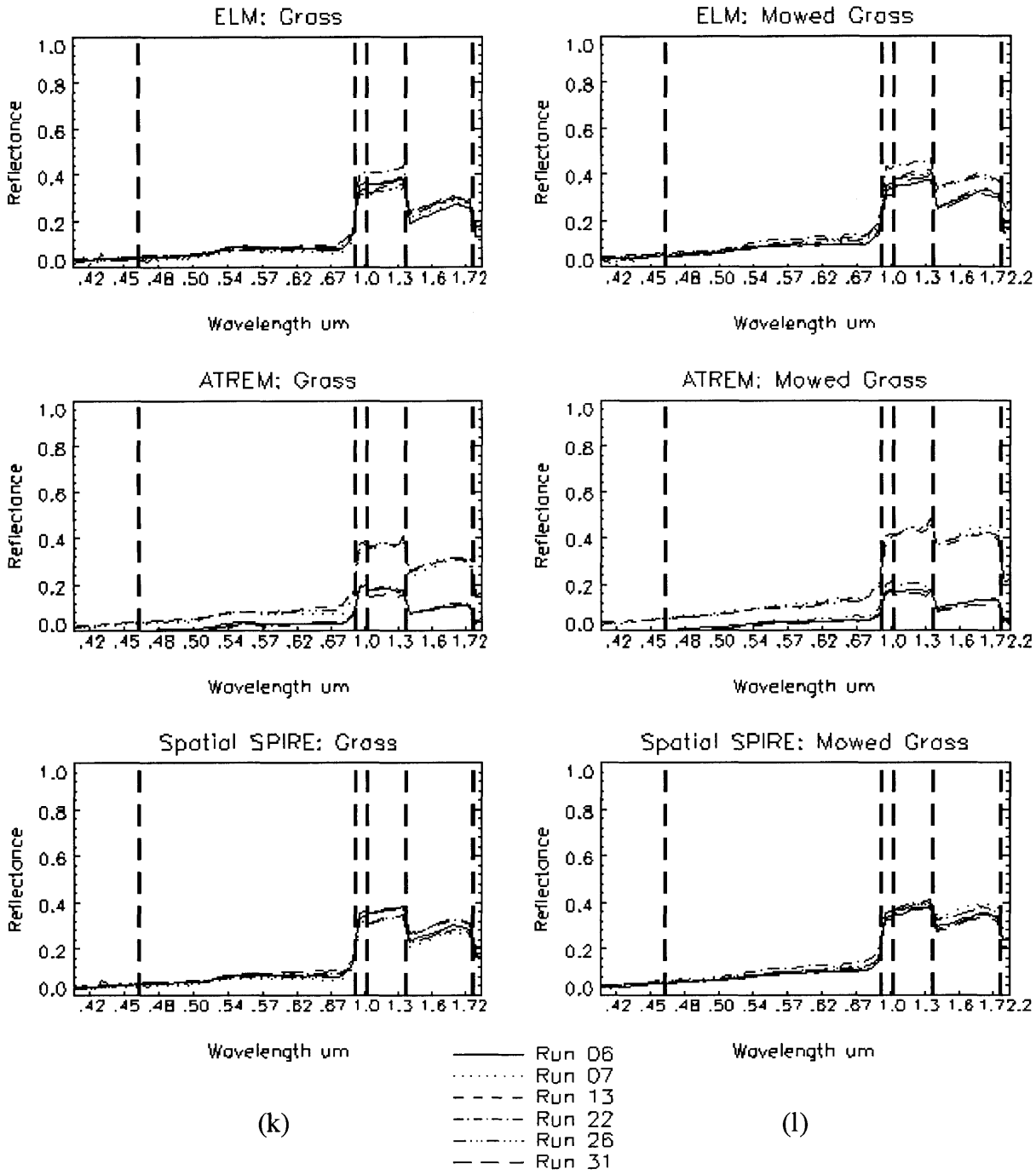


Figure C.1 (k) and (l). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on grass (k) and mowed grass (l).*

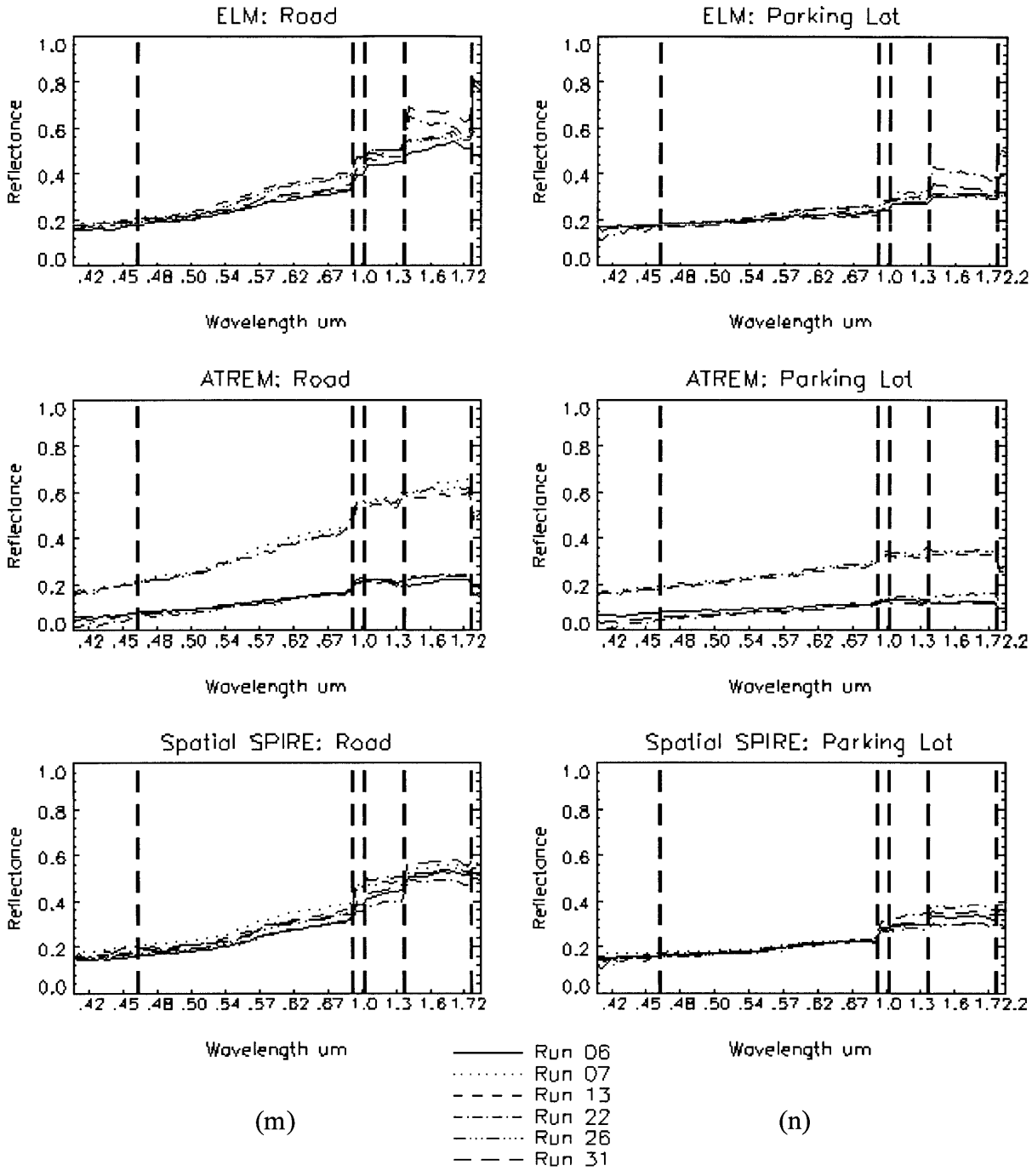


Figure C.1 (m) and (n). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the road (m) and the parking lot (n).*

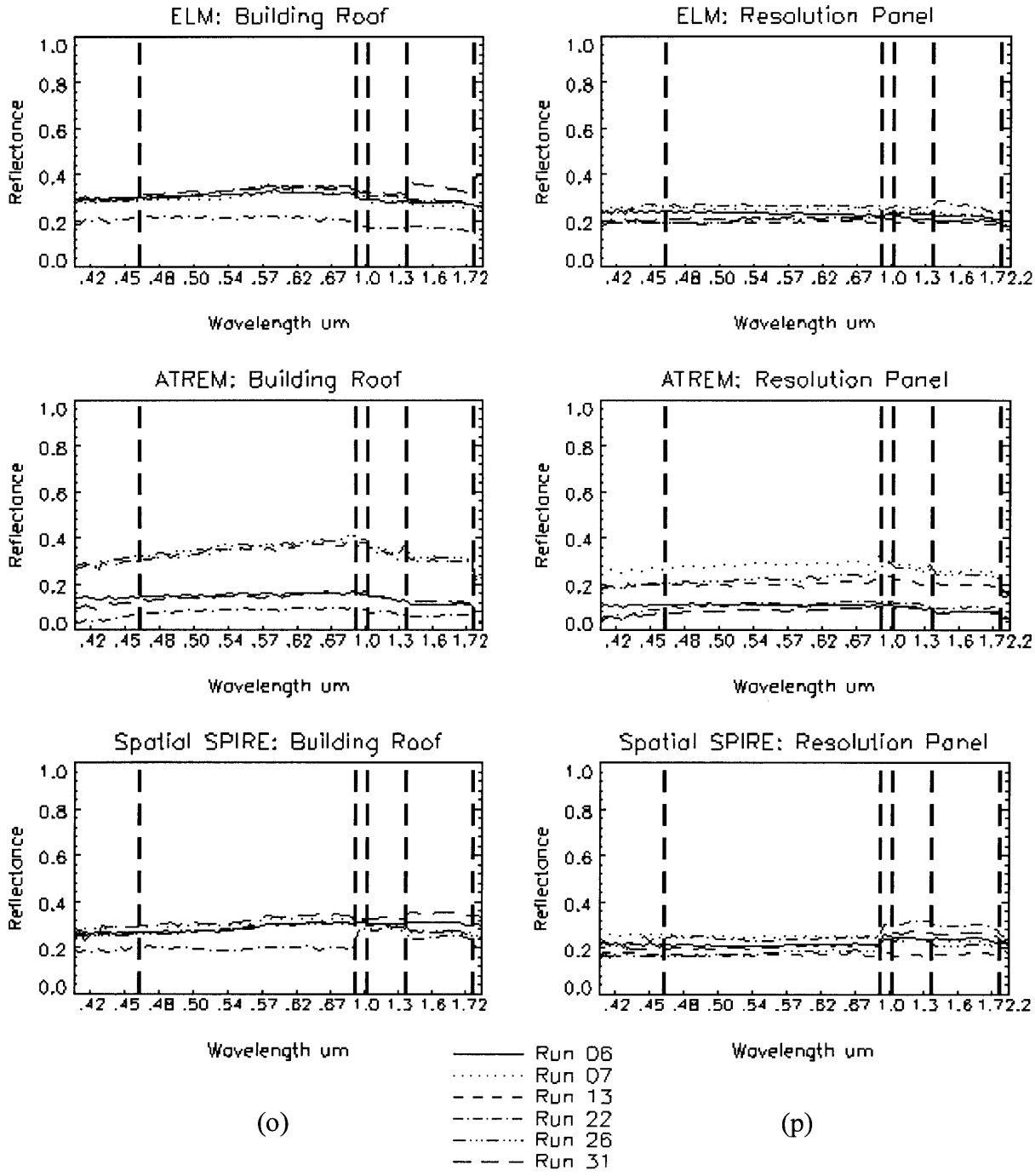


Figure C.1 (o) and (p). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on the building roof (o) and the resolution panel (p).*

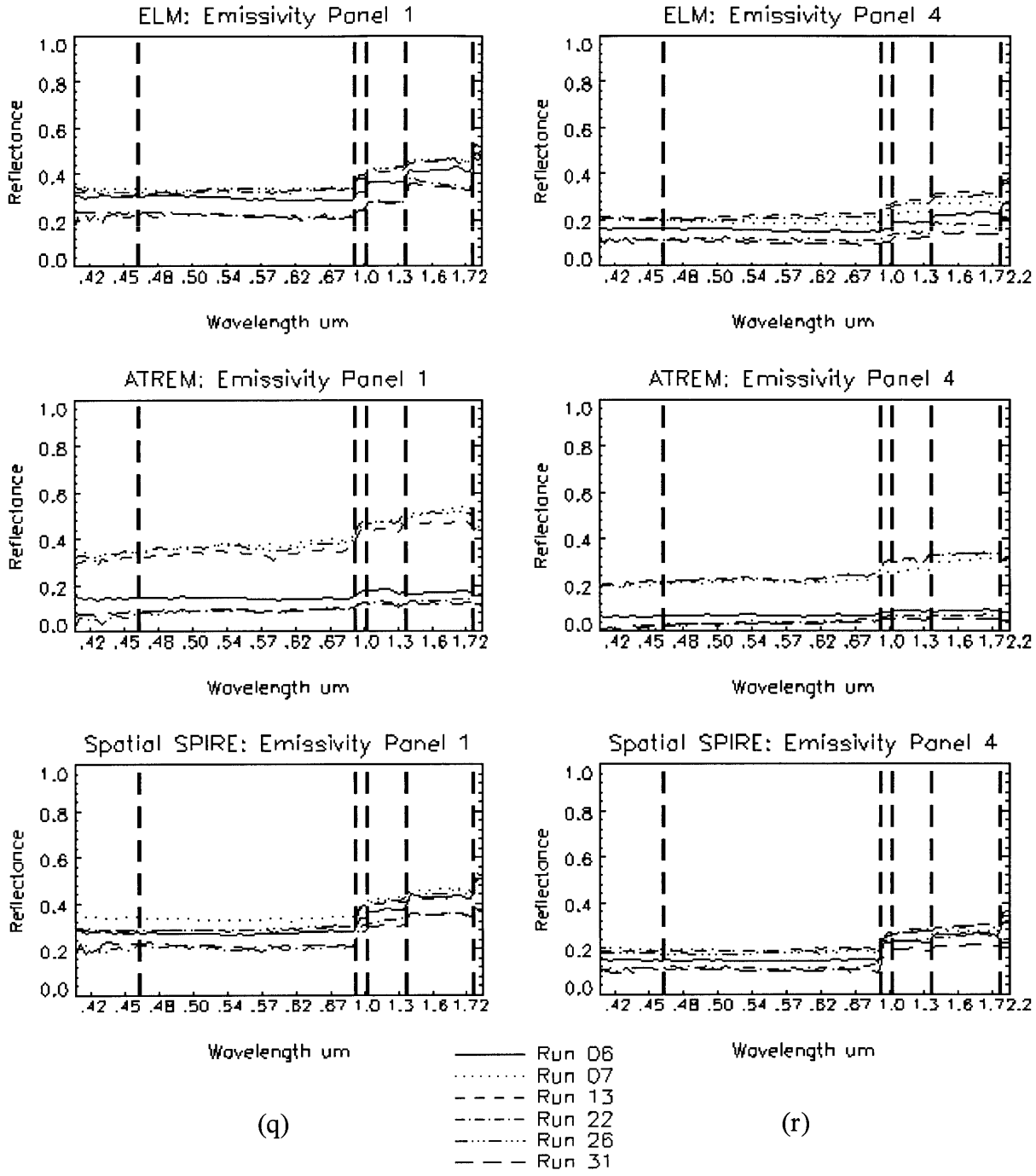
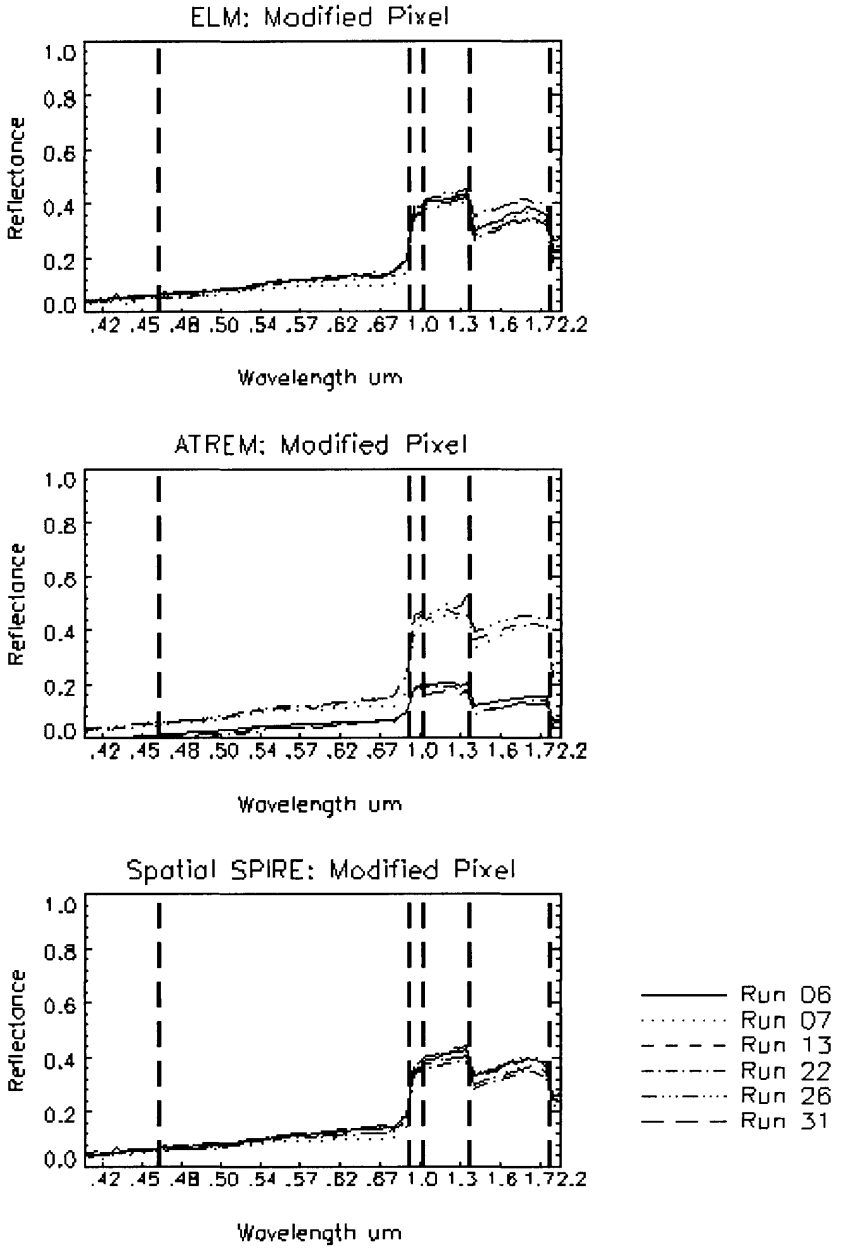


Figure C.1 (q) and (r). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for a single pixel on emissivity panel 1 (q) and emissivity panel 4 (r).*



(s)

Figure C.1(s). *ELM, ATREM, and Spatial SPIRE spectral reflectance estimates for all six runs for the modified pixel. The modified pixel was a truck in Run 07, and changed to mowed grass in all subsequent Runs. For Run 07, a similar mowed grass pixel is plotted for comparison.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
2% Panel	Band 1	0.005	0.029	0.005
	Band 2	0.002	0.020	0.005
	Band 3	0.001	0.019	0.085
	Band 4	0.002	0.018	0.080
	Band 5	0.002	0.020	0.058
	Band 6	0.004	0.023	0.044

Table C.1(a). *Spatial SPIRE Spatial SPIRE average channel standard deviation from the mean over all runs for the 2 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
4% Panel	Band 1	0.006	0.039	0.007
	Band 2	0.003	0.028	0.004
	Band 3	0.002	0.025	0.080
	Band 4	0.002	0.023	0.077
	Band 5	0.005	0.024	0.056
	Band 6	0.012	0.028	0.044

Table C.1(b). *Spatial SPIRE Spatial SPIRE average channel standard deviation from the mean over all runs for the 4 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
8% Panel	Band 1	0.008	0.054	0.010
	Band 2	0.005	0.041	0.008
	Band 3	0.006	0.037	0.071
	Band 4	0.007	0.033	0.071
	Band 5	0.008	0.035	0.055
	Band 6	0.012	0.038	0.039

Table C.1(c). *Spatial SPIRE Spatial SPIRE average channel standard deviation from the mean over all runs for the 8 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
16% Panel	Band 1	0.007	0.072	0.013
	Band 2	0.012	0.065	0.012
	Band 3	0.013	0.071	0.045
	Band 4	0.014	0.067	0.050
	Band 5	0.020	0.071	0.043
	Band 6	0.017	0.065	0.028

Table C.1(d). *Spatial SPIRE average channel standard deviation from the mean over all runs for the 16 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
32% Panel	Band 1	0.009	0.108	0.021
	Band 2	0.014	0.118	0.019
	Band 3	0.017	0.134	0.007
	Band 4	0.017	0.134	0.005
	Band 5	0.031	0.133	0.024
	Band 6	0.030	0.094	0.027

Table C.1(e). *Spatial SPIRE average channel standard deviation from the mean over all runs for the 32 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
64% Panel	Band 1	0.007	0.161	0.032
	Band 2	0.006	0.194	0.034
	Band 3	0.010	0.217	0.079
	Band 4	0.011	0.221	0.065
	Band 5	0.022	0.214	0.048
	Band 6	0.019	0.134	0.051

Table C.1(f). *Spatial SPIRE average channel standard deviation from the mean over all runs for the 64 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Spectral Panel	Band 1	0.016	0.125	0.028
	Band 2	0.014	0.165	0.029
	Band 3	0.009	0.201	0.064
	Band 4	0.007	0.190	0.042
	Band 5	0.013	0.173	0.025
	Band 6	0.032	0.115	0.034

Table C.1(g). *Spatial SPIRE average channel standard deviation from the mean over all runs for the spectral panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
15% Panel	Band 1	0.020	0.077	0.020
	Band 2	0.013	0.069	0.012
	Band 3	0.009	0.067	0.046
	Band 4	0.009	0.059	0.052
	Band 5	0.017	0.059	0.045
	Band 6	0.036	0.057	0.035

Table C.1(h). *Spatial SPIRE average channel standard deviation from the mean over all runs for the 15 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
41% Panel	Band 1	0.019	0.130	0.031
	Band 2	0.020	0.146	0.022
	Band 3	0.020	0.158	0.029
	Band 4	0.018	0.157	0.019
	Band 5	0.018	0.165	0.028
	Band 6	0.038	0.129	0.058

Table C.1(i). *Spatial SPIRE average channel standard deviation from the mean over all runs for the 41 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
57% Panel	Band 1	0.024	0.165	0.040
	Band 2	0.022	0.197	0.039
	Band 3	0.016	0.220	0.082
	Band 4	0.020	0.222	0.070
	Band 5	0.054	0.226	0.070
	Band 6	0.087	0.163	0.096

Table C.1(j). *Spatial SPIRE average channel standard deviation from the mean over all runs for 57 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Grass	Band 1	0.006	0.027	0.005
	Band 2	0.009	0.029	0.008
	Band 3	0.035	0.104	0.020
	Band 4	0.033	0.112	0.021
	Band 5	0.017	0.104	0.024
	Band 6	0.021	0.063	0.018

Table C.1(k). *Spatial SPIRE average channel standard deviation from the mean over all runs for grass.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Mowed Grass	Band 1	0.007	0.034	0.005
	Band 2	0.010	0.040	0.008
	Band 3	0.040	0.124	0.011
	Band 4	0.030	0.145	0.012
	Band 5	0.042	0.161	0.024
	Band 6	0.054	0.101	0.027

Table C.1(l). *Spatial SPIRE average channel standard deviation from the mean over all runs for mowed grass.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Road	Band 1	0.011	0.074	0.015
	Band 2	0.022	0.114	0.023
	Band 3	0.032	0.184	0.050
	Band 4	0.025	0.191	0.044
	Band 5	0.054	0.210	0.029
	Band 6	0.126	0.172	0.031

Table C.1(m). *Spatial SPIRE average channel standard deviation from the mean over all runs for the road.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Parking Lot	Band 1	0.012	0.071	0.013
	Band 2	0.010	0.082	0.005
	Band 3	0.018	0.109	0.016
	Band 4	0.018	0.107	0.023
	Band 5	0.040	0.113	0.033
	Band 6	0.077	0.090	0.035

Table C.1(n). *Spatial SPIRE average channel standard deviation from the mean over all runs for parking lot.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Building Roof	Band 1	0.037	0.108	0.035
	Band 2	0.047	0.126	0.043
	Band 3	0.058	0.137	0.017
	Band 4	0.056	0.124	0.018
	Band 5	0.058	0.114	0.038
	Band 6	0.081	0.088	0.048

Table C.1(o). *Spatial SPIRE average channel standard deviation from the mean over all runs for the building roof.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Resolution Panel	Band 1	0.030	0.079	0.036
	Band 2	0.029	0.079	0.032
	Band 3	0.023	0.090	0.040
	Band 4	0.022	0.085	0.047
	Band 5	0.025	0.083	0.042
	Band 6	0.027	0.067	0.032

Table C.1(p). *Spatial SPIRE average channel standard deviation from the mean over all runs for the resolution panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Emissivity Panel 1	Band 1	0.052	0.131	0.052
	Band 2	0.054	0.140	0.048
	Band 3	0.071	0.174	0.048
	Band 4	0.069	0.180	0.050
	Band 5	0.053	0.198	0.048
	Band 6	0.028	0.174	0.067

Table C.1(q). *Spatial SPIRE average channel standard deviation from the mean over all runs for emissivity panel 1.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Emissivity Panel 4	Band 1	0.046	0.098	0.041
	Band 2	0.049	0.098	0.039
	Band 3	0.066	0.123	0.027
	Band 4	0.067	0.125	0.032
	Band 5	0.070	0.141	0.031
	Band 6	0.075	0.134	0.055

Table C.1(r). *Spatial SPIRE average channel standard deviation from the mean over all runs for emissivity panel 4.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Modified Pixel	Band 1	0.006	0.032	0.005
	Band 2	0.011	0.038	0.010
	Band 3	0.017	0.139	0.015
	Band 4	0.016	0.154	0.020
	Band 5	0.033	0.156	0.021
	Band 6	0.034	0.094	0.016

Table C.1(s). *Spatial SPIRE average channel standard deviation from the mean over all runs for the modified pixel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
2% Panel	Band 1	11.2	677.0	11.1
	Band 2	4.5	98.4	11.2
	Band 3	2.0	48.5	77.2
	Band 4	3.4	49.4	75.2
	Band 5	4.0	56.5	59.0
	Band 6	8.4	74.1	47.4

Table C.2(a). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 2 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
4% Panel	Band 1	8.5	132.2	9.3
	Band 2	5.1	70.8	6.8
	Band 3	2.9	48.7	66.6
	Band 4	2.9	49.2	66.6
	Band 5	8.3	57.3	53.2
	Band 6	20.0	76.3	43.4

Table C.2(b). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 4 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
8% Panel	Band 1	6.4	80.1	9.1
	Band 2	4.8	54.3	7.5
	Band 3	5.7	45.8	48.0
	Band 4	6.9	45.1	49.3
	Band 5	8.2	49.5	39.5
	Band 6	10.7	62.7	28.4

Table C.2(c). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 8 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
16% Panel	Band 1	3.9	59.1	7.3
	Band 2	6.4	47.7	6.8
	Band 3	7.2	46.4	21.0
	Band 4	7.3	46.3	23.3
	Band 5	9.7	49.4	19.5
	Band 6	8.1	57.7	12.8

Table C.2(d). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 16 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
32% Panel	Band 1	2.6	46.9	7.1
	Band 2	4.0	43.6	6.0
	Band 3	4.5	43.7	2.1
	Band 4	4.6	45.9	1.4
	Band 5	8.1	49.5	6.8
	Band 6	9.2	53.8	8.8

Table C.2(e). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 32 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
64% Panel	Band 1	1.3	42.6	6.5
	Band 2	1.1	42.2	6.3
	Band 3	1.5	42.1	14.3
	Band 4	1.8	45.8	11.6
	Band 5	3.8	51.8	9.0
	Band 6	4.3	55.8	12.6

Table C.2(f). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 64 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Spectral Panel	Band 1	4.6	49.8	8.6
	Band 2	3.2	45.1	6.8
	Band 3	1.7	44.2	13.0
	Band 4	1.4	47.5	8.7
	Band 5	2.9	53.4	5.9
	Band 6	9.8	61.6	10.7

Table C.2(g). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the spectral panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
15% Panel	Band 1	11.2	66.9	12.1
	Band 2	7.5	53.6	7.2
	Band 3	5.8	48.3	23.0
	Band 4	5.7	47.6	26.5
	Band 5	10.2	50.7	23.6
	Band 6	21.1	60.0	18.5

Table C.2(h). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 15 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
41% Panel	Band 1	5.0	48.4	8.9
	Band 2	4.9	46.3	6.0
	Band 3	4.8	44.3	7.3
	Band 4	4.3	46.5	4.7
	Band 5	4.2	53.1	7.0
	Band 6	9.8	59.2	15.8

Table C.2(i). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 41 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
57% Panel	Band 1	4.7	46.0	8.8
	Band 2	3.9	45.5	7.8
	Band 3	2.7	44.0	15.3
	Band 4	3.3	47.5	12.8
	Band 5	10.2	56.6	13.9
	Band 6	20.4	65.1	23.3

Table C.2(j). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 57 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Grass	Band 1	16.0	238.9	12.5
	Band 2	11.6	78.7	11.3
	Band 3	10.2	37.3	6.0
	Band 4	9.2	41.1	6.1
	Band 5	6.5	54.4	8.6
	Band 6	12.8	67.4	9.5

Table C.2(k). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for grass.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Mowed Grass	Band 1	17.5	1843.8	10.3
	Band 2	10.5	70.2	8.7
	Band 3	11.5	42.7	3.2
	Band 4	7.7	47.2	3.2
	Band 5	12.6	62.5	7.0
	Band 6	24.7	74.6	11.3

Table C.2(l). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for mowed grass.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Road	Band 1	6.4	64.3	9.3
	Band 2	7.6	52.9	9.1
	Band 3	7.2	48.5	11.8
	Band 4	5.1	49.9	9.6
	Band 5	9.4	50.6	5.5
	Band 6	19.9	51.9	5.7

Table C.2(m). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for road.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Parking Lot	Band 1	7.2	67.3	8.5
	Band 2	4.9	51.4	2.9
	Band 3	6.6	47.8	5.6
	Band 4	6.4	47.6	7.6
	Band 5	12.2	48.5	10.0
	Band 6	21.3	48.3	10.7

Table C.2(n). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the parking lot.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Building Roof	Band 1	13.3	56.1	13.9
	Band 2	15.3	52.1	15.3
	Band 3	19.6	53.2	5.3
	Band 4	20.1	54.9	6.1
	Band 5	21.4	57.1	13.4
	Band 6	30.8	60.8	18.1

Table C.2(o). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for building roof.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Resolution Panel	Band 1	13.6	54.5	17.7
	Band 2	12.7	46.8	15.3
	Band 3	9.8	46.6	16.0
	Band 4	9.8	48.1	19.3
	Band 5	11.4	52.7	17.9
	Band 6	13.5	59.9	15.3

Table C.2(p). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the resolution panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Emissivity Panel 1	Band 1	17.9	63.2	19.5
	Band 2	18.9	59.9	18.2
	Band 3	21.6	59.3	13.9
	Band 4	18.9	60.1	13.4
	Band 5	12.7	61.8	11.7
	Band 6	5.6	61.3	14.6

Table C.2(q). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for emissivity panel 1.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Emissivity Panel 4	Band 1	27.7	86.1	26.3
	Band 2	29.9	73.3	25.5
	Band 3	34.6	69.7	11.5
	Band 4	32.2	69.8	13.3
	Band 5	29.1	71.8	11.9
	Band 6	25.4	73.8	18.3

Table C.2(r). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for emissivity panel 4.*

Pixel	Band	ELM	ATREM	Spatial SPIRE
Modified Pixel	Band 1	11.9	151.4	10.6
	Band 2	10.2	58.8	9.8
	Band 3	4.5	44.7	4.3
	Band 4	3.9	47.5	4.9
	Band 5	9.5	59.0	6.0
	Band 6	14.5	67.9	6.3

Table C.2(s). *Spatial SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the modified pixel.*

Appendix D

Spectral SPIRE Case C Algorithm with Spectral Library Matching

In this appendix we provide details on the experiment briefly discussed in Chapter 5 with results shown in Figure 5.5 using the algorithm depicted in Figure 5.4. This experiment involves using the Spectral SPIRE Case C algorithm with matching against a spectral library to restore the $\Delta \log \mathbf{r}$ lost in the high pass PC filtering. Please refer to Section 5.3.2.1 for a description of the algorithm in Figure 5.4.

D.1 Generation of Simulated Reflectance Cube

The HYDICE ARM site data from Run 07 of 24 June 1997 was selected for use in generating the test reflectance images which were used to simulate the hyperspectral reflectance cube. A 200x200 pixel sub-cube was cut from the large image cube of Run 07. The selected sub-cube contains a variety of image features, including test panels, grass, mowed grass, roads, buildings, and vehicles. A single band image from this sub-cube is shown in Figure D.1(a).

ELM calibration was then done to reduce the sub image cube to an estimate of reflectance. ELM processing requires known spectra for pixels in the image. Ground truth measurements of spectra over the test panels were available with the ARM site image data. Ground truth spectra for the 2 percent and 64 percent panels, collected on 24 June 1997, were used to perform the ELM calibration. The sub-cube was pre-processed to remove artifacts involving integer wrapping of some bright pixel into negative numbers. Using ENVI 3.1's Empirical Line Calibration (ELM) routine, the Run 07 sub-cube was processed to estimate reflectance.

However, the ELM calibration was not perfect. The ELM algorithm does not make any

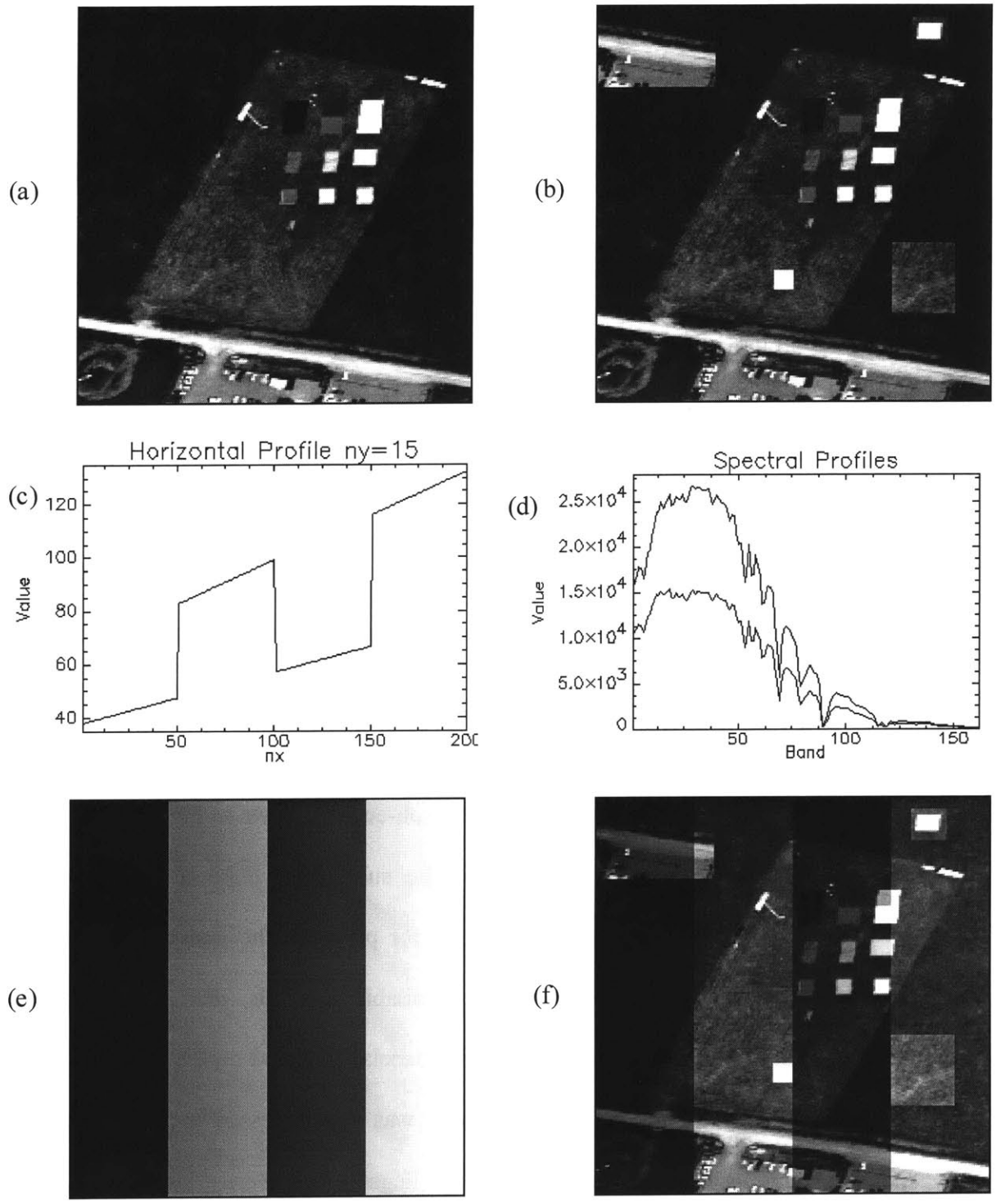


Figure D.1. *Original (a) and modified (b) reflectance band images. Horizontal (c) and spectral (d) profiles of multiplicative noise image (e), and resultant simulated radiance image (f) for testing the Case C algorithm using spectral library matching.*

physical assumptions about the reflectivity of the scene, but simply performs a linear regression to estimate the multiplicative and additive effects. This means that inaccuracies in the ground truth spectra and spatial variations in illumination effects can cause estimates of the reflectance to be negative or greater than unity. The ELM reflectance estimates for some of the Run 07 pixels were slightly negative or slightly greater than unity. To compensate for this effect, all channels containing negative reflectance values were shifted by adding the absolute value of the smallest pixel value plus a small offset of 0.001 so that logarithms could be applied. Next, each channel with a reflectance greater than one was scaled so that the maximum reflectance became 0.99. Finally, noisy spectral channel images within the water absorption bands were removed from the sub-cube, as were other channels containing striping artifacts, leaving 162 of the original 210 HYDICE spectral channel images in the sub-cube.

D.2 Simulated Reflectance Changes

The reflectance cube generated using ELM in the previous section served as both the prior spectral library as well as the prior reflectance image cube for SPIRE processing. A reflectance cube containing changes from the original was needed to generate the simulated radiance cube to simulate a subsequent data collection to be processed. This was generated by modifying the original reflectance cube with copies of spectra from one portion of the image to another. This corresponds to changes occurring in the image which are contained in spectral library. Figure D.1(b) depicts a single band image from the modified reflectance cube. The parking lot from the lower right has been copied to the upper left, a portion of the mowed (lighter shade) grass has been copied to the unmowed grass in the lower left, and the spectral panel test has been copied to the upper right corner of the image. In addition, a simulated test panel with spectrally uniform 50 percent reflectance was introduced below the other test panels. This

simulated 50 percent reflectance spectrum is not present in the spectral library.

D.3 Simulated Radiance Cube Generation

To generate a simulated radiance cube for this case, a multiplicative noise cube was generated and multiplied times the modified reflectance cube of Figure D.1(b). A single band image form the multiplicative noise cube used is shown in Figure D.1(e), with a horizontal profile across it plotted in Figure D.1(c), showing that spatially it contains a square wave with a superimposed horizontal ramp. In the spectral dimension, the high and low portions of the square wave had the high and low spectral shapes shown in Figure D.1(d), which are actual radiance spectra measured over the 2 percent and 64 percent spectral panels, so they contain typical solar spectrum shape and atmospheric absorption features. The net effect is a multiplicative noise effect that varies both spectrally and spatially across the image, with both high and low spatial frequency components. A single band image from the resulting simulated radiance cube is shown in Figure D.1(f).

D.4 Results of Processing Simulated Radiance Test Cube

The radiance cube from Figure D.1(f) was process using the Spectral SPIRE Case D algorithm depicted in Figure 5.4 that uses spectral library matching. Abutted Principal Components (APC) analysis was used to generate the PC rotation matrix. It was determined using the graphical technique described in Section D.5 that there were nine log **m** PCs that needed to be zeroed to remove the spatially varying illumination noise. Figure D.2(a) shows a single band image from the estimated reflectance cube output by the algorithm. Reflectance estimation was perfect except for over the simulated 50 percent test panel whose spectra is not in the spectral library. Figure D.2(b) shows the distance image resulting from the classification

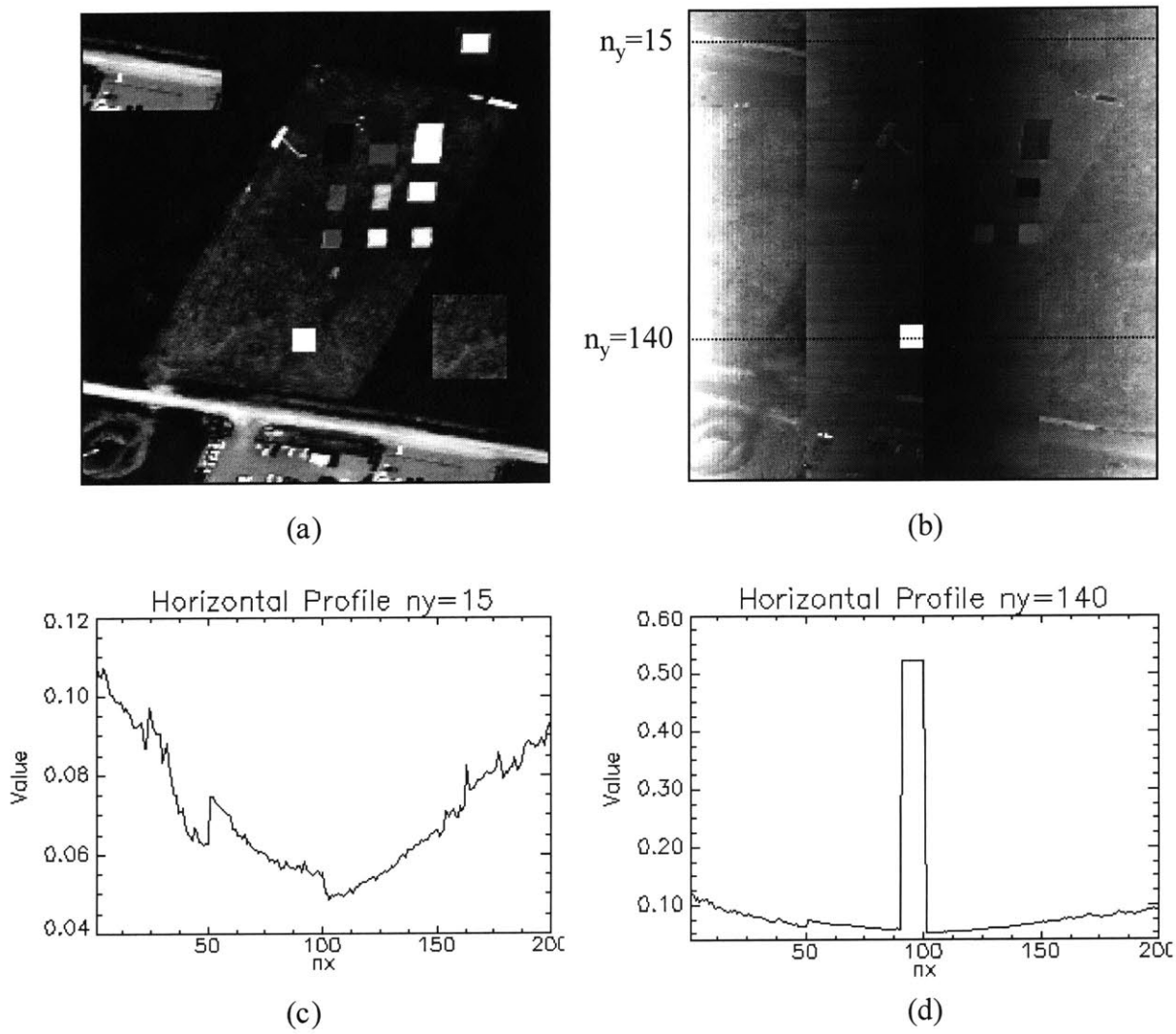
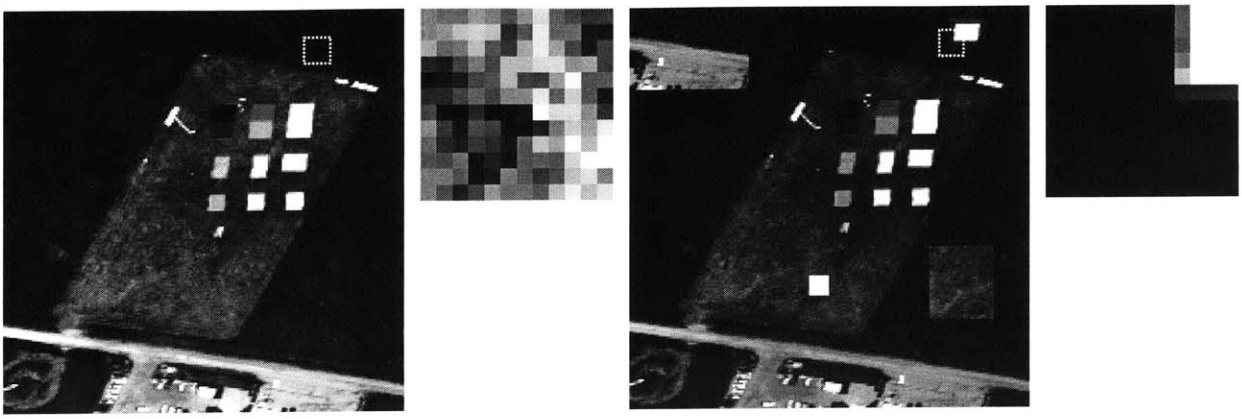
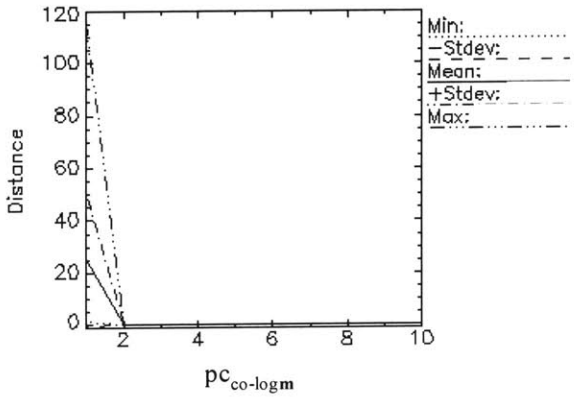


Figure D.2. *Estimated reflectance channel image (a), identical to modified reflectance image in Figure D.1(b), except for the unknown simulated 50 percent test panel. Distance image from library classification is shown in (b) with horizontal profiles through n_y=15 (c) and n_y=140 (d). Variation in (d) is due to small amounts of log* ***m*** *noise in higher PCs, but matching overcomes this noise. The unknown panel has highest distance and large reflectance error since its material is not present in the spectral library.*

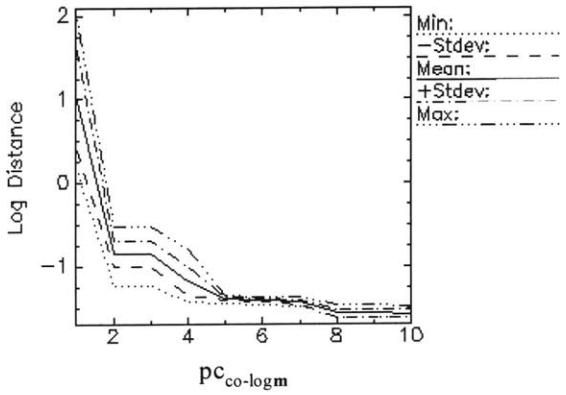
step. Figure D.2(c) shows a horizontal profile through the distance image at n_y=15 where all materials were in the spectral library. Figure D.2(d) shows a similar profile but through the 50 percent test panel at n_y=140. Since the 50 percent panel was not in the spectral library, its distance to the closest spectrum in the library is high. In Figures D.2(c) and (d) one can see the



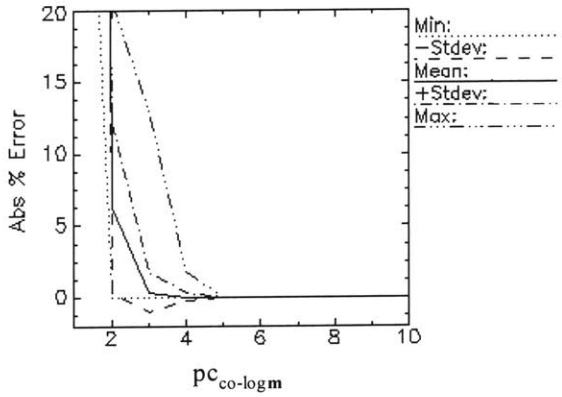
(a) (b)



(c)



(d)



(e)

Figure D.3. *Mini-cubes cropped from the original (a) and modified (b) reflectance cubes. Plot of average spectral distance versus* $pc_{co\text{-}logm}$*, both distance (c) and log distance (d). Plot of absolute percent error vs* $pc_{co\text{-}logm}$*. In (d), the correct value to use for* $pc_{co\text{-}logm}$ *is at the pinch point where the plots converge.*

residual multiplicative noise effects not completely eliminated by dropping the top PCs. However, these effects were small enough to still allow correct classification of all the known spectra.

D.5 Graphical Determination of the Number of PCs to Drop

To highlight the details of determining the number of top PCs to drop, two mini image cubes were extracted from the original reflectance cube of Figure D.1(a), and a simulated radiance cube generated by multiplying the modified reflectance cube of Figure D.1(b) by a horizontal ramp. The 10x10x162 mini-cubes were extracted from the upper right of the image covering a corner of the copied spectral reflectance panel as depicted in Figure D.3(a) and (b). The radiance mini-cube therefore contained both modified and unmodified pixels. The radiance mini-cube was processed several times through the Case C algorithm varying the number $pc_{co\text{-}logm}$ value for in each run. The $pc_{co\text{-}logm}$ value was varied from 1 to 10, and in each run, the mini-cube from Figure D.3(a) was used as the prior reflectance cube, while the full reflectance cube of Figure D.1(a) was used as the spectral library.

The classification distance of all the pixels in the mini-cube was calculated and plotted versus $pc_{co\text{-}logm}$ and is shown in Figure D.3(c), including the mean, min, max, and plus and minus standard deviation about the mean. By plotting the log of the distance in Figure D.3(d), the details of the transition near $pc_{co\text{-}logm} = 2$ is brought out. In Figure D.3(e), the absolute percent error between each pixel of the estimated reflectance cube and the actual modified reflectance cube is plotted versus $pc_{co\text{-}logm}$, again with mean, min, max, and plus and minus one standard deviation. We see from Figure D.3(e) that the mean, min, and max converge and the standard deviation drops to zero at $pc_{co\text{-}logm} = 5$. This corresponds to the pinch point in Figure D.3(d) where its five curves converge as well. This is the point where the graphical analysis determines

the correct value for $pc_{co\text{-}log\mathbf{m}}$ using a plot similar to the one in Figure D.3(d), generated from a set of sample spectra. The absolute percent error plot in Figure D.3(e) cannot be used for this since the actual reflectance is not known a priori.

Appendix E

Additional Spectral SPIRE Results

This appendix consists of Figure E.1 and Tables E.1 and E.2. Figure E.1 is identical to Figure 5.8 that shows ELM, ATREM, and Spectral SPIRE reflectance estimates for the 19 example material pixels, except that the reflectance axes have a fixed range between 0 and 1. The reflectance axes in Figure 5.8 are set to the range appropriate for the data plotted to show the maximum amount of detail in each individual plot. The fixed range used in Figure E.1 is better suited for comparing the results of different pixels. Table E.1 lists the Spectral SPIRE average channel standard deviation over the six runs in each band for each algorithm and Table E.2 lists the same values but as a percentage of the mean estimated reflectance over the six runs.

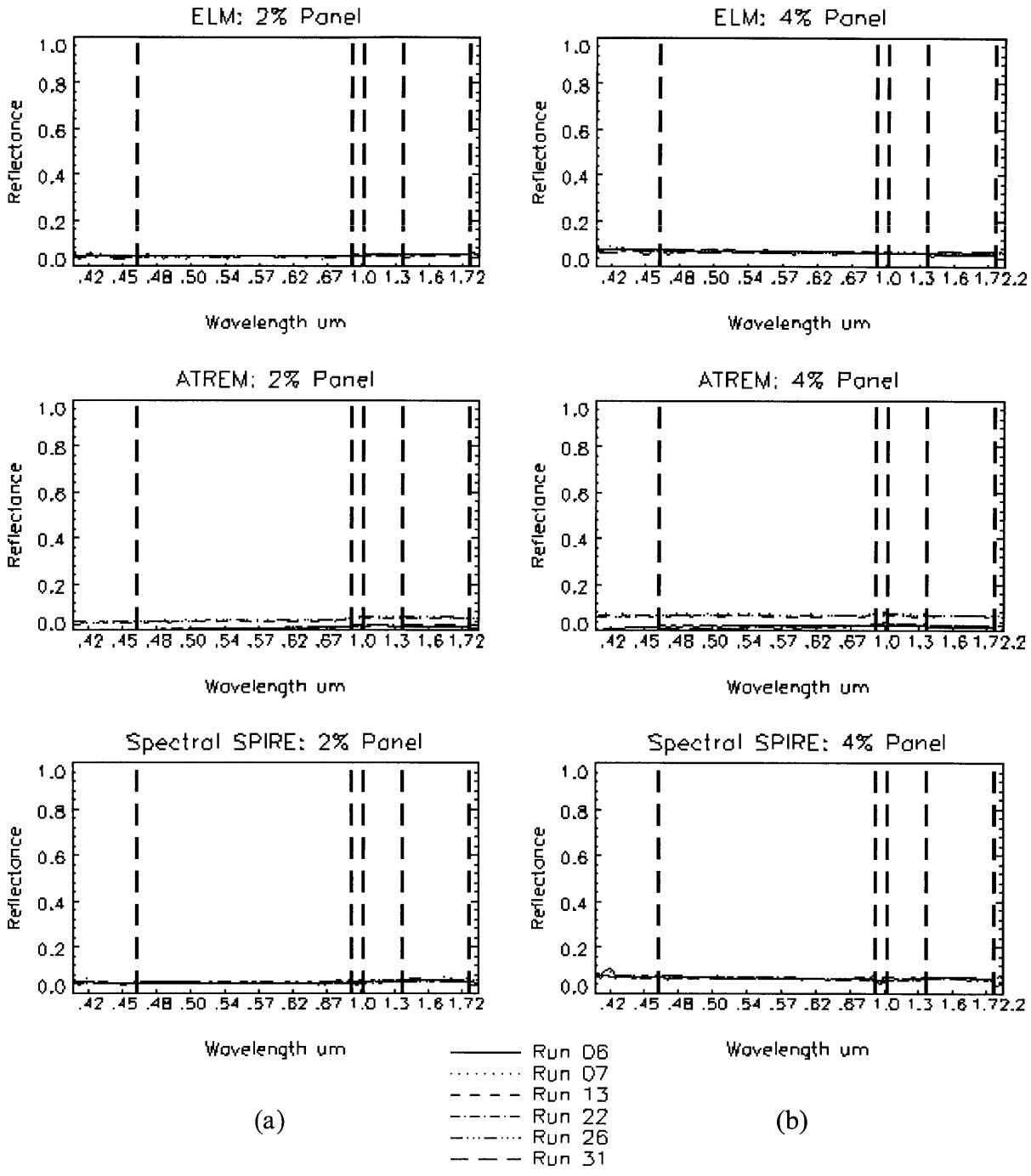


Figure E.1 (a) and (b). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the 2 percent panel (a) and the 4 percent panel (b).*

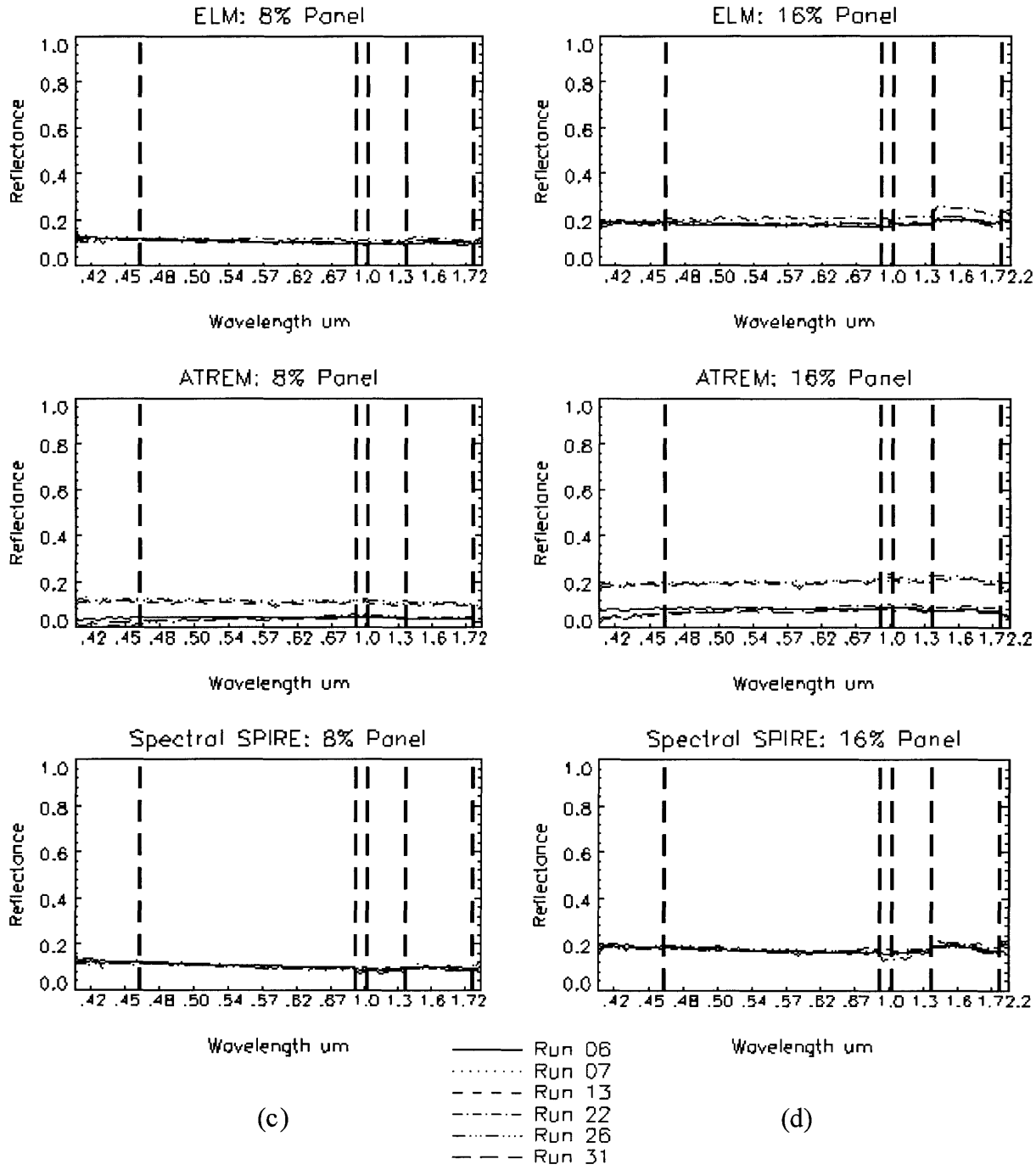


Figure E.1 (c) and (d). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the 8 percent panel (c) and the 16 percent panel (d).*

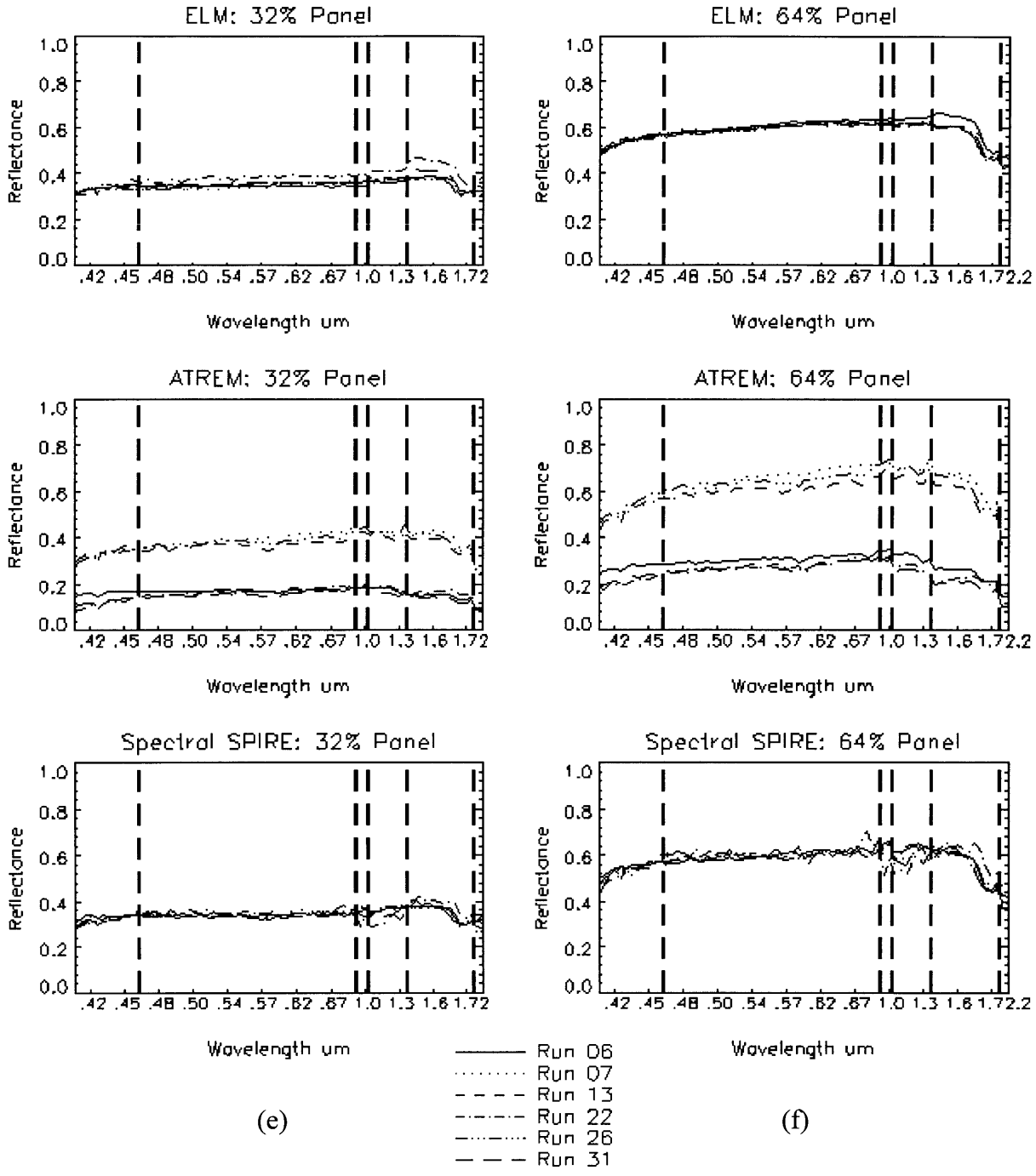


Figure E.1 (e) and (f). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the 32 percent panel (e) and the 64 percent panel (f).*

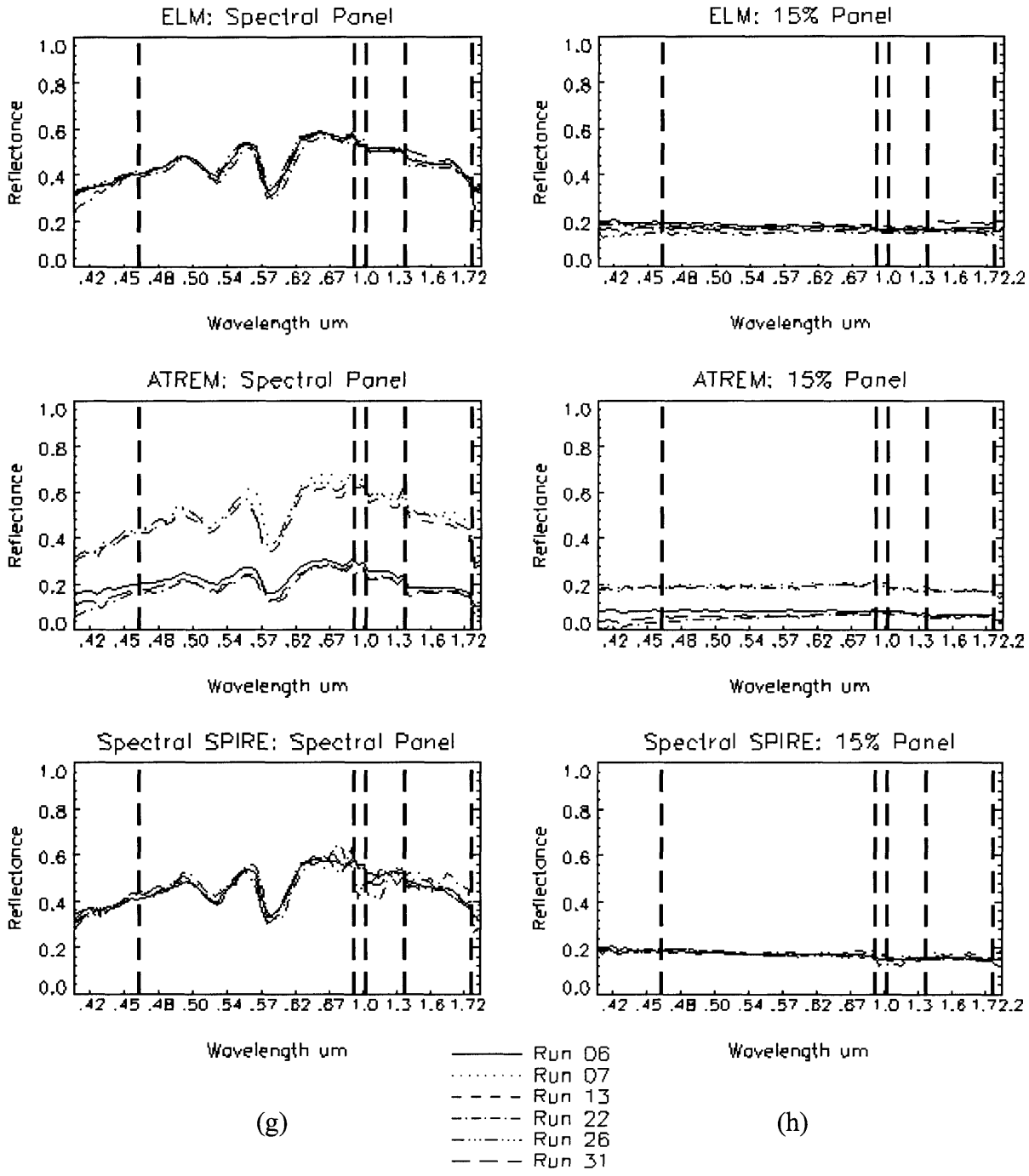


Figure E.1 (g) and (h). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the spectral panel (g) and the 15 percent panel (h).*

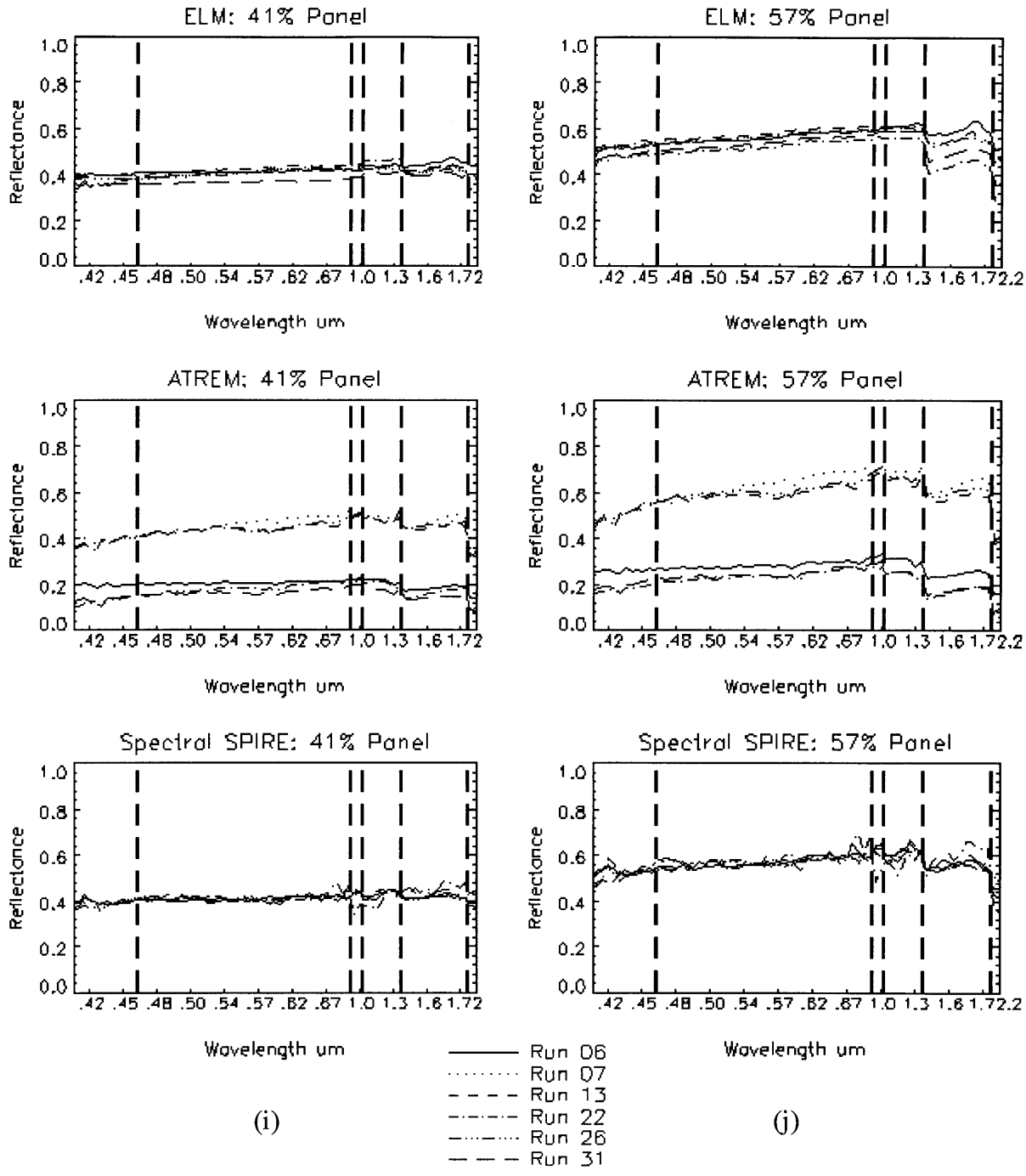


Figure E.1 (i) and (j). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the 41 percent panel (i) and the 57 percent panel (j).*

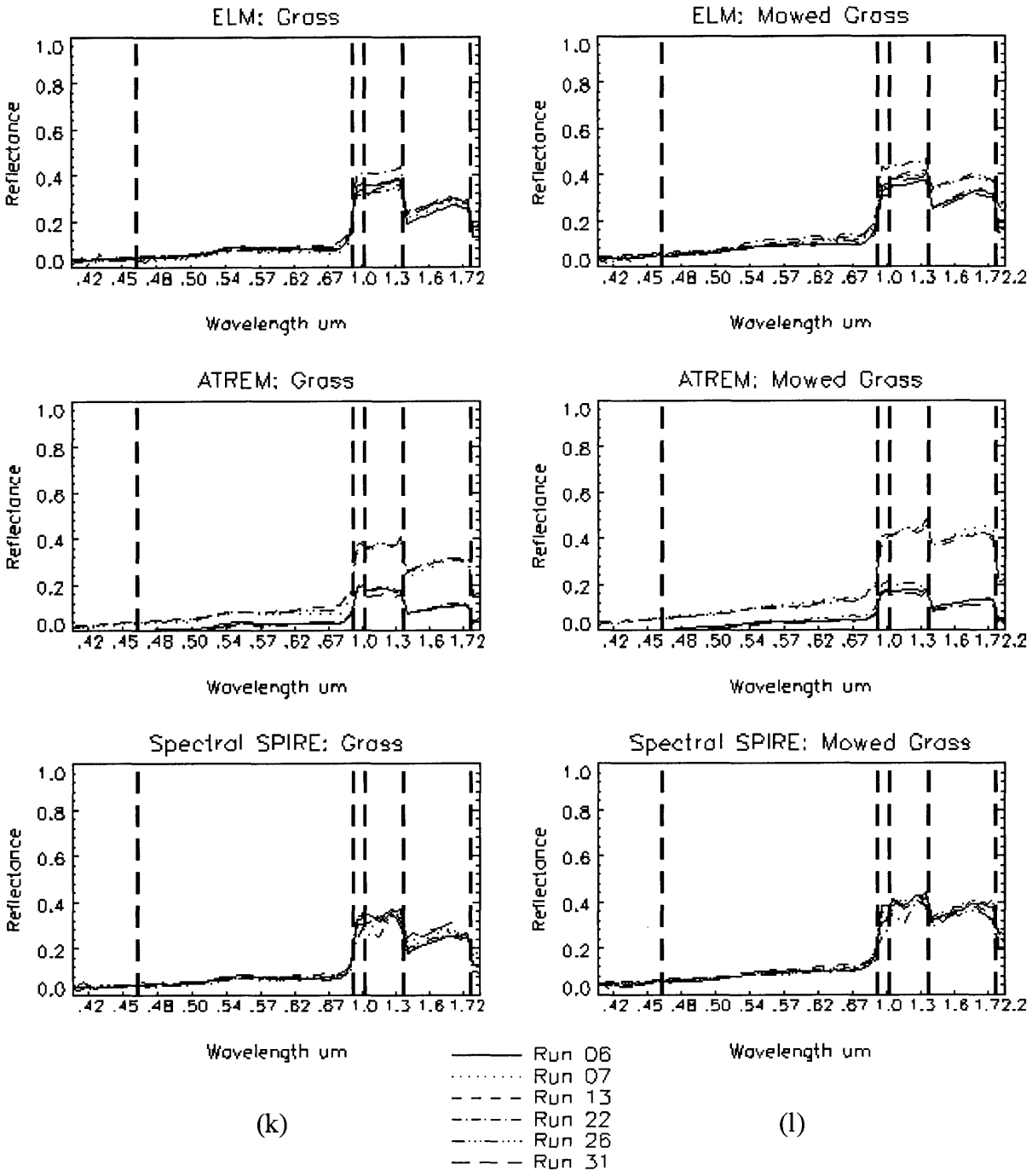


Figure E.1 (k) and (l). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on grass (k) and mowed grass (l).*

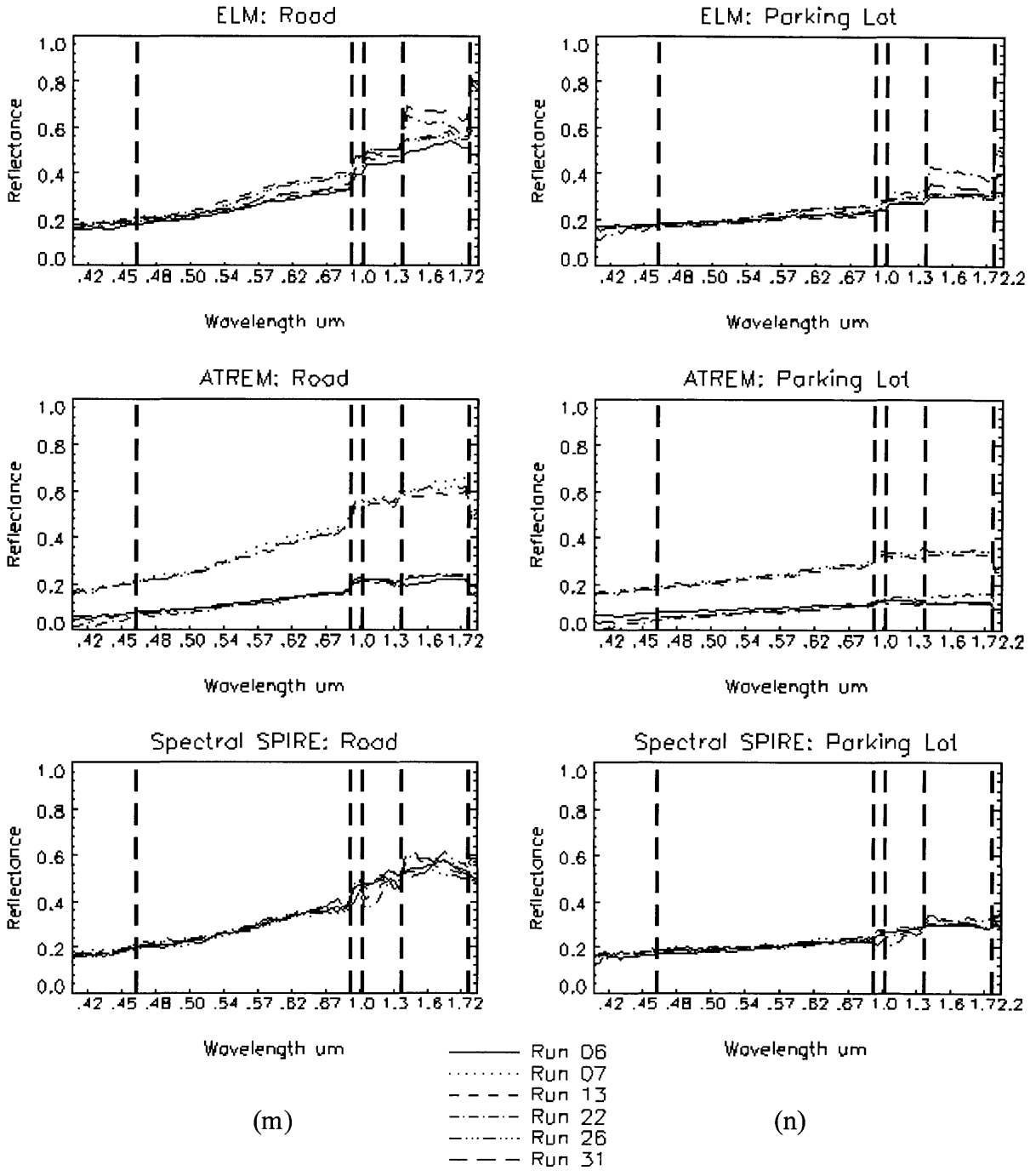


Figure E.1 (m) and (n). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the road (m) and the parking lot (n).*

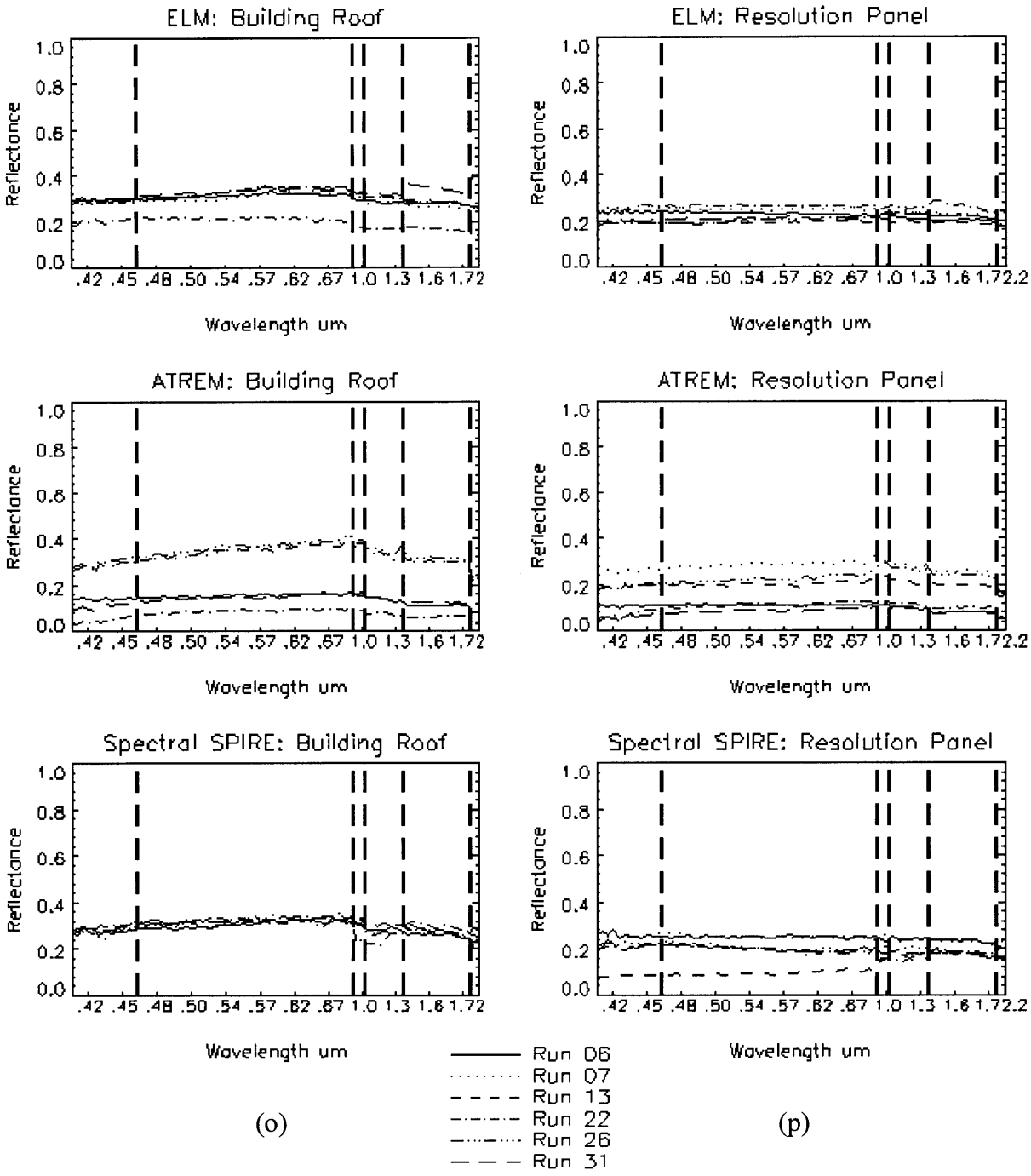


Figure E.1 (o) and (p). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on the building roof (o) and the resolution panel (p).*

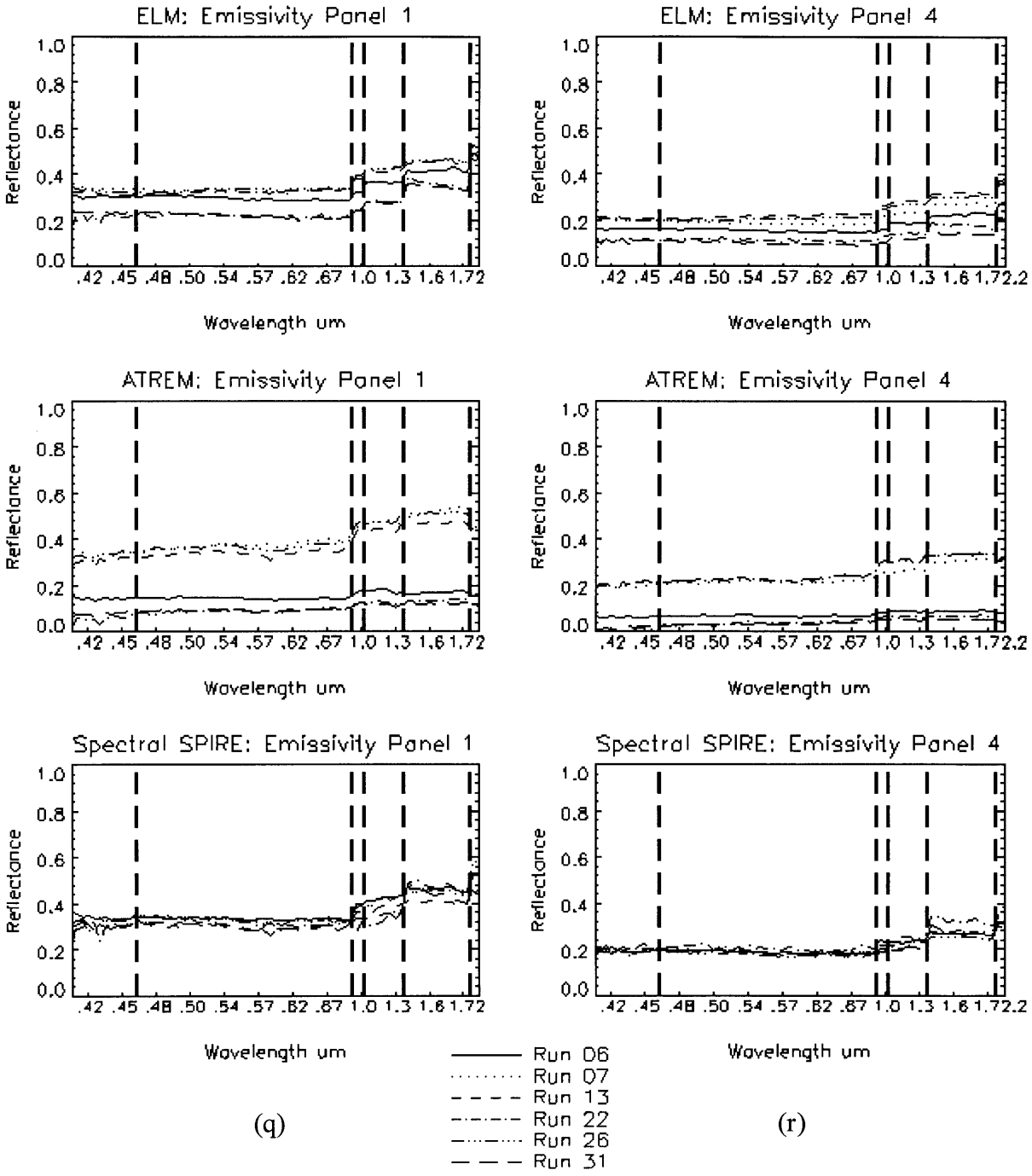
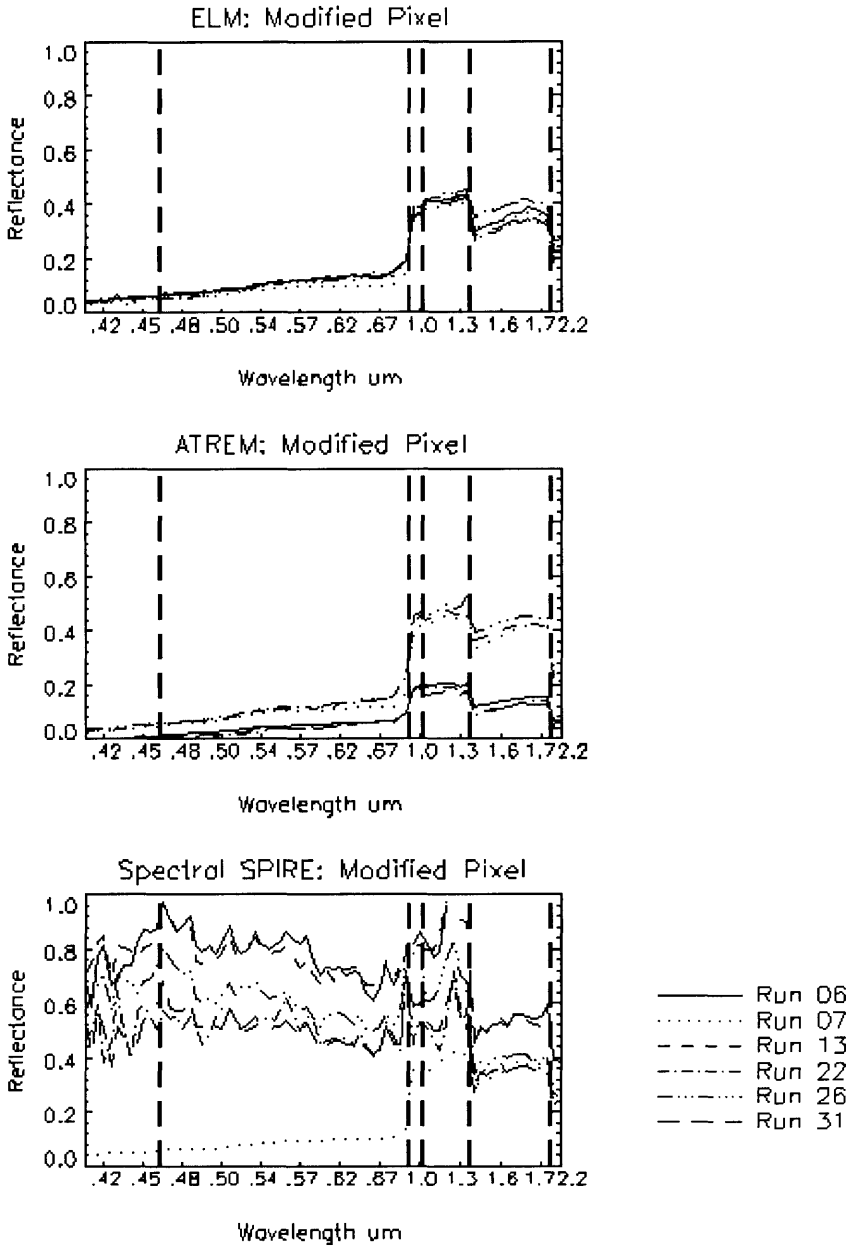


Figure E.1 (q) and (r). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six runs for a single pixel on emissivity panel 1 (q) and emissivity panel 4 (r).*



(s)

Figure E.1(s). *ELM, ATREM, and Spectral SPIRE spectral reflectance estimates for all six Runs for the modified pixel. The modified pixel was a truck in Run 07, and changed to mowed grass in all subsequent Runs. For Run 07, a similar mowed grass pixel is plotted for comparison.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
2% Panel	Band 1	0.005	0.029	0.004
	Band 2	0.002	0.020	0.002
	Band 3	0.001	0.019	0.006
	Band 4	0.002	0.018	0.003
	Band 5	0.002	0.020	0.004
	Band 6	0.004	0.023	0.011

Table E.1(a). *Spectral SPIRE Spectral SPIRE average channel standard deviation from the mean over all runs for the 2 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
4% Panel	Band 1	0.006	0.039	0.006
	Band 2	0.003	0.028	0.003
	Band 3	0.002	0.025	0.009
	Band 4	0.002	0.023	0.004
	Band 5	0.005	0.024	0.004
	Band 6	0.012	0.028	0.013

Table E.1(b). *Spectral SPIRE average channel standard deviation from the mean over all runs for the 4 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
8% Panel	Band 1	0.008	0.054	0.007
	Band 2	0.005	0.041	0.004
	Band 3	0.006	0.037	0.011
	Band 4	0.007	0.033	0.006
	Band 5	0.008	0.035	0.005
	Band 6	0.012	0.038	0.015

Table E.1(c). *Spectral SPIRE average channel standard deviation from the mean over all runs for the 8 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
16% Panel	Band 1	0.007	0.072	0.006
	Band 2	0.012	0.065	0.004
	Band 3	0.013	0.071	0.017
	Band 4	0.014	0.067	0.010
	Band 5	0.020	0.071	0.009
	Band 6	0.017	0.065	0.020

Table E.1(d). *Spectral SPIRE average channel standard deviation from the mean over all runs for the 16 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
32% Panel	Band 1	0.009	0.108	0.009
	Band 2	0.014	0.118	0.009
	Band 3	0.017	0.134	0.026
	Band 4	0.017	0.134	0.022
	Band 5	0.031	0.133	0.017
	Band 6	0.030	0.094	0.029

Table E.1(e). *Spectral SPIRE average channel standard deviation from the mean over all runs for the 32 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
64% Panel	Band 1	0.007	0.161	0.014
	Band 2	0.006	0.194	0.013
	Band 3	0.010	0.217	0.041
	Band 4	0.011	0.221	0.031
	Band 5	0.022	0.214	0.025
	Band 6	0.019	0.134	0.041

Table E.1(f). *Spectral SPIRE average channel standard deviation from the mean over all runs for the 64 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Spectral Panel	Band 1	0.016	0.125	0.012
	Band 2	0.014	0.165	0.017
	Band 3	0.009	0.201	0.048
	Band 4	0.007	0.190	0.027
	Band 5	0.013	0.173	0.023
	Band 6	0.032	0.115	0.039

Table E.1(g). *Spectral SPIRE average channel standard deviation from the mean over all runs for the spectral panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
15% Panel	Band 1	0.020	0.077	0.007
	Band 2	0.013	0.069	0.005
	Band 3	0.009	0.067	0.016
	Band 4	0.009	0.059	0.008
	Band 5	0.017	0.059	0.007
	Band 6	0.036	0.057	0.022

Table E.1(h). *Spectral SPIRE average channel standard deviation from the mean over all runs for the 15 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
41% Panel	Band 1	0.019	0.130	0.011
	Band 2	0.020	0.146	0.009
	Band 3	0.020	0.158	0.032
	Band 4	0.018	0.157	0.017
	Band 5	0.018	0.165	0.019
	Band 6	0.038	0.129	0.042

Table E.1(i). *Spectral SPIRE average channel standard deviation from the mean over all runs for the 41 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
57% Panel	Band 1	0.024	0.165	0.019
	Band 2	0.022	0.197	0.014
	Band 3	0.016	0.220	0.048
	Band 4	0.020	0.222	0.024
	Band 5	0.054	0.226	0.026
	Band 6	0.087	0.163	0.060

Table E.1(j). *Spectral SPIRE average channel standard deviation from the mean over all runs for 57 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Grass	Band 1	0.006	0.027	0.004
	Band 2	0.009	0.029	0.006
	Band 3	0.035	0.104	0.032
	Band 4	0.033	0.112	0.024
	Band 5	0.017	0.104	0.022
	Band 6	0.021	0.063	0.032

Table E.1(k). *Spectral SPIRE average channel standard deviation from the mean over all runs for grass.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Mowed Grass	Band 1	0.007	0.034	0.006
	Band 2	0.010	0.040	0.007
	Band 3	0.040	0.124	0.048
	Band 4	0.030	0.145	0.024
	Band 5	0.042	0.161	0.018
	Band 6	0.054	0.101	0.033

Table E.1(l). *Spectral SPIRE average channel standard deviation from the mean over all runs for mowed grass.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Road	Band 1	0.011	0.074	0.009
	Band 2	0.022	0.114	0.010
	Band 3	0.032	0.184	0.023
	Band 4	0.025	0.191	0.032
	Band 5	0.054	0.210	0.028
	Band 6	0.126	0.172	0.046

Table E.1(m). *Spectral SPIRE average channel standard deviation from the mean over all runs for the road.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Parking Lot	Band 1	0.012	0.071	0.010
	Band 2	0.010	0.082	0.006
	Band 3	0.018	0.109	0.021
	Band 4	0.018	0.107	0.018
	Band 5	0.040	0.113	0.012
	Band 6	0.077	0.090	0.022

Table E.1(n). *Spectral SPIRE average channel standard deviation from the mean over all runs for parking lot.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Building Roof	Band 1	0.037	0.108	0.012
	Band 2	0.047	0.126	0.012
	Band 3	0.058	0.137	0.034
	Band 4	0.056	0.124	0.023
	Band 5	0.058	0.114	0.014
	Band 6	0.081	0.088	0.037

Table E.1(o). *Spectral SPIRE average channel standard deviation from the mean over all runs for the building roof.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Resolution Panel	Band 1	0.030	0.079	0.065
	Band 2	0.029	0.079	0.058
	Band 3	0.023	0.090	0.043
	Band 4	0.022	0.085	0.037
	Band 5	0.025	0.083	0.027
	Band 6	0.027	0.067	0.030

Table E.1(p). *Spectral SPIRE average channel standard deviation from the mean over all runs for the resolution panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Emissivity Panel 1	Band 1	0.052	0.131	0.022
	Band 2	0.054	0.140	0.017
	Band 3	0.071	0.174	0.040
	Band 4	0.069	0.180	0.041
	Band 5	0.053	0.198	0.027
	Band 6	0.028	0.174	0.052

Table E.1(q). *Spectral SPIRE average channel standard deviation from the mean over all runs for emissivity panel 1.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Emissivity Panel 4	Band 1	0.046	0.098	0.009
	Band 2	0.049	0.098	0.010
	Band 3	0.066	0.123	0.021
	Band 4	0.067	0.125	0.015
	Band 5	0.070	0.141	0.026
	Band 6	0.075	0.134	0.034

Table E.1(r). *Spectral SPIRE average channel standard deviation from the mean over all runs for emissivity panel 4.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Modified Pixel	Band 1	0.006	0.032	0.276
	Band 2	0.011	0.038	0.260
	Band 3	0.017	0.139	0.246
	Band 4	0.016	0.154	0.208
	Band 5	0.033	0.156	0.087
	Band 6	0.034	0.094	0.074

Table E.1(s). *Spectral SPIRE average channel standard deviation from the mean over all runs for the modified pixel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
2% Panel	Band 1	11.2	677.0	8.4
	Band 2	4.5	98.4	5.0
	Band 3	2.0	48.5	13.2
	Band 4	3.4	49.4	6.4
	Band 5	4.0	56.5	7.8
	Band 6	8.4	74.1	21.1

Table E.2(a). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 2 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
4% Panel	Band 1	8.5	132.2	8.0
	Band 2	5.1	70.8	4.7
	Band 3	2.9	48.7	15.7
	Band 4	2.9	49.2	6.4
	Band 5	8.3	57.3	6.6
	Band 6	20.0	76.3	22.0

Table E.2(b). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 4 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
8% Panel	Band 1	6.4	80.1	5.7
	Band 2	4.8	54.3	3.6
	Band 3	5.7	45.8	12.8
	Band 4	6.9	45.1	6.4
	Band 5	8.2	49.5	5.1
	Band 6	10.7	62.7	14.3

Table E.2(c). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 8 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
16% Panel	Band 1	3.9	59.1	3.3
	Band 2	6.4	47.7	2.5
	Band 3	7.2	46.4	10.5
	Band 4	7.3	46.3	5.8
	Band 5	9.7	49.4	4.6
	Band 6	8.1	57.7	10.3

Table E.2(d). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 16 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
32% Panel	Band 1	2.6	46.9	2.9
	Band 2	4.0	43.6	2.6
	Band 3	4.5	43.7	7.3
	Band 4	4.6	45.9	6.3
	Band 5	8.1	49.5	4.8
	Band 6	9.2	53.8	9.4

Table E.2(e). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 32 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
64% Panel	Band 1	1.3	42.6	2.8
	Band 2	1.1	42.2	2.2
	Band 3	1.5	42.1	6.6
	Band 4	1.8	45.8	5.1
	Band 5	3.8	51.8	4.5
	Band 6	4.3	55.8	9.7

Table E.2(f). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 64 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Spectral Panel	Band 1	4.6	49.8	3.4
	Band 2	3.2	45.1	3.6
	Band 3	1.7	44.2	8.9
	Band 4	1.4	47.5	5.4
	Band 5	2.9	53.4	5.3
	Band 6	9.8	61.6	11.5

Table E.2(g). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the spectral panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
15% Panel	Band 1	11.2	66.9	3.5
	Band 2	7.5	53.6	2.7
	Band 3	5.8	48.3	10.1
	Band 4	5.7	47.6	5.2
	Band 5	10.2	50.7	4.5
	Band 6	21.1	60.0	13.3

Table E.2(h). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 15 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
41% Panel	Band 1	5.0	48.4	2.9
	Band 2	4.9	46.3	2.2
	Band 3	4.8	44.3	7.7
	Band 4	4.3	46.5	4.1
	Band 5	4.2	53.1	4.3
	Band 6	9.8	59.2	10.7

Table E.2(i). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 41 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
57% Panel	Band 1	4.7	46.0	3.6
	Band 2	3.9	45.5	2.5
	Band 3	2.7	44.0	8.0
	Band 4	3.3	47.5	4.0
	Band 5	10.2	56.6	4.5
	Band 6	20.4	65.1	13.1

Table E.2(j). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 57 percent panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Grass	Band 1	16.0	238.9	11.9
	Band 2	11.6	78.7	9.4
	Band 3	10.2	37.3	10.0
	Band 4	9.2	41.1	7.5
	Band 5	6.5	54.4	8.9
	Band 6	12.8	67.4	21.6

Table E.2(k). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for grass.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Mowed Grass	Band 1	17.5	1843.8	11.9
	Band 2	10.5	70.2	7.1
	Band 3	11.5	42.7	13.9
	Band 4	7.7	47.2	6.3
	Band 5	12.6	62.5	5.0
	Band 6	24.7	74.6	14.0

Table E.2(l). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for mowed grass.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Road	Band 1	6.4	64.3	4.9
	Band 2	7.6	52.9	3.5
	Band 3	7.2	48.5	5.0
	Band 4	5.1	49.9	6.9
	Band 5	9.4	50.6	5.1
	Band 6	19.9	51.9	8.6

Table E.2(m). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for road.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Parking Lot	Band 1	7.2	67.3	5.8
	Band 2	4.9	51.4	3.1
	Band 3	6.6	47.8	8.3
	Band 4	6.4	47.6	6.7
	Band 5	12.2	48.5	3.9
	Band 6	21.3	48.3	6.9

Table E.2(n). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the parking lot.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Building Roof	Band 1	13.3	56.1	4.2
	Band 2	15.3	52.1	3.6
	Band 3	19.6	53.2	11.2
	Band 4	20.1	54.9	8.4
	Band 5	21.4	57.1	5.1
	Band 6	30.8	60.8	14.7

Table E.2(o). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for building roof.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Resolution Panel	Band 1	13.6	54.5	31.7
	Band 2	12.7	46.8	28.9
	Band 3	9.8	46.6	21.5
	Band 4	9.8	48.1	19.3
	Band 5	11.4	52.7	14.0
	Band 6	13.5	59.9	16.2

Table E.2(p). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the resolution panel.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Emissivity Panel 1	Band 1	17.9	63.2	7.0
	Band 2	18.9	59.9	5.4
	Band 3	21.6	59.3	11.0
	Band 4	18.9	60.1	10.7
	Band 5	12.7	61.8	5.9
	Band 6	5.6	61.3	10.2

Table E.2(q). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for emissivity panel 1.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Emissivity Panel 4	Band 1	27.7	86.1	4.9
	Band 2	29.9	73.3	5.5
	Band 3	34.6	69.7	10.1
	Band 4	32.2	69.8	6.8
	Band 5	29.1	71.8	9.3
	Band 6	25.4	73.8	10.3

Table E.2(r). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for emissivity panel 4.*

Pixel	Band	ELM	ATREM	Spectral SPIRE
Modified Pixel	Band 1	11.9	151.4	50.5
	Band 2	10.2	58.8	47.1
	Band 3	4.5	44.7	35.7
	Band 4	3.9	47.5	31.2
	Band 5	9.5	59.0	21.0
	Band 6	14.5	67.9	24.5

Table E.2(s). *Spectral SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the modified pixel.*

Appendix F

Additional Combined SPIRE Results

This appendix consists of Figure F.1 and Tables F.1 and F.2. Figure F.1 is identical to Figure 6.6 that shows ELM, ATREM, and Combined SPIRE reflectance estimates for the 19 example material pixels, except that the reflectance axes have a fixed range between 0 and 1. The reflectance axes in Figure 6.6 are set to the range appropriate for the data plotted to show the maximum amount of detail in each individual plot. The fixed range used in Figure F.1 is better suited for comparing the results of different pixels. Table F.1 lists the Combined SPIRE average channel standard deviation over the six runs in each band for each algorithm and Table F.2 lists the same values but as a percentage of the mean estimated reflectance over the six runs.

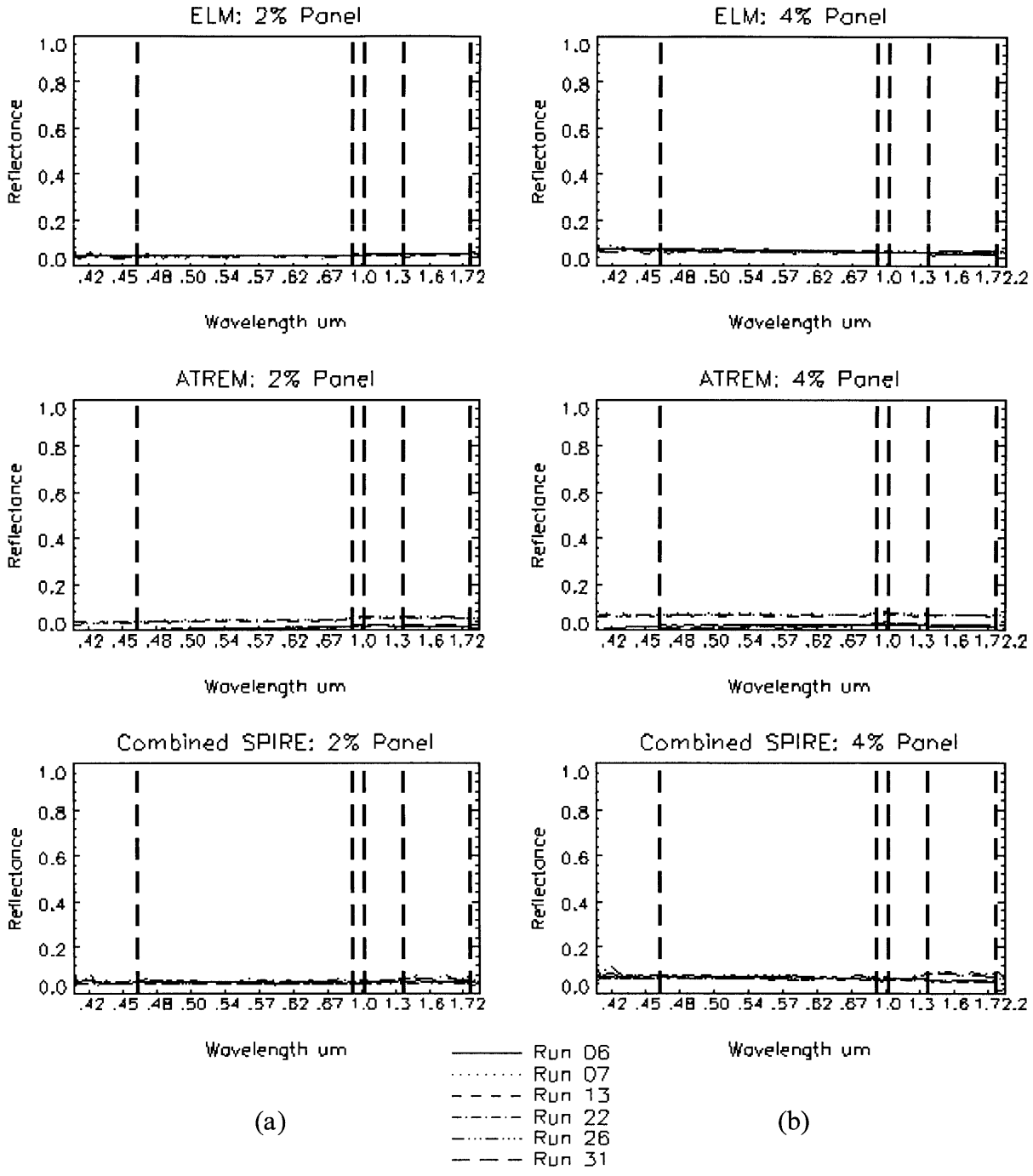


Figure F.1 (a) and (b). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the 2 percent panel (a) and the 4 percent panel (b).*

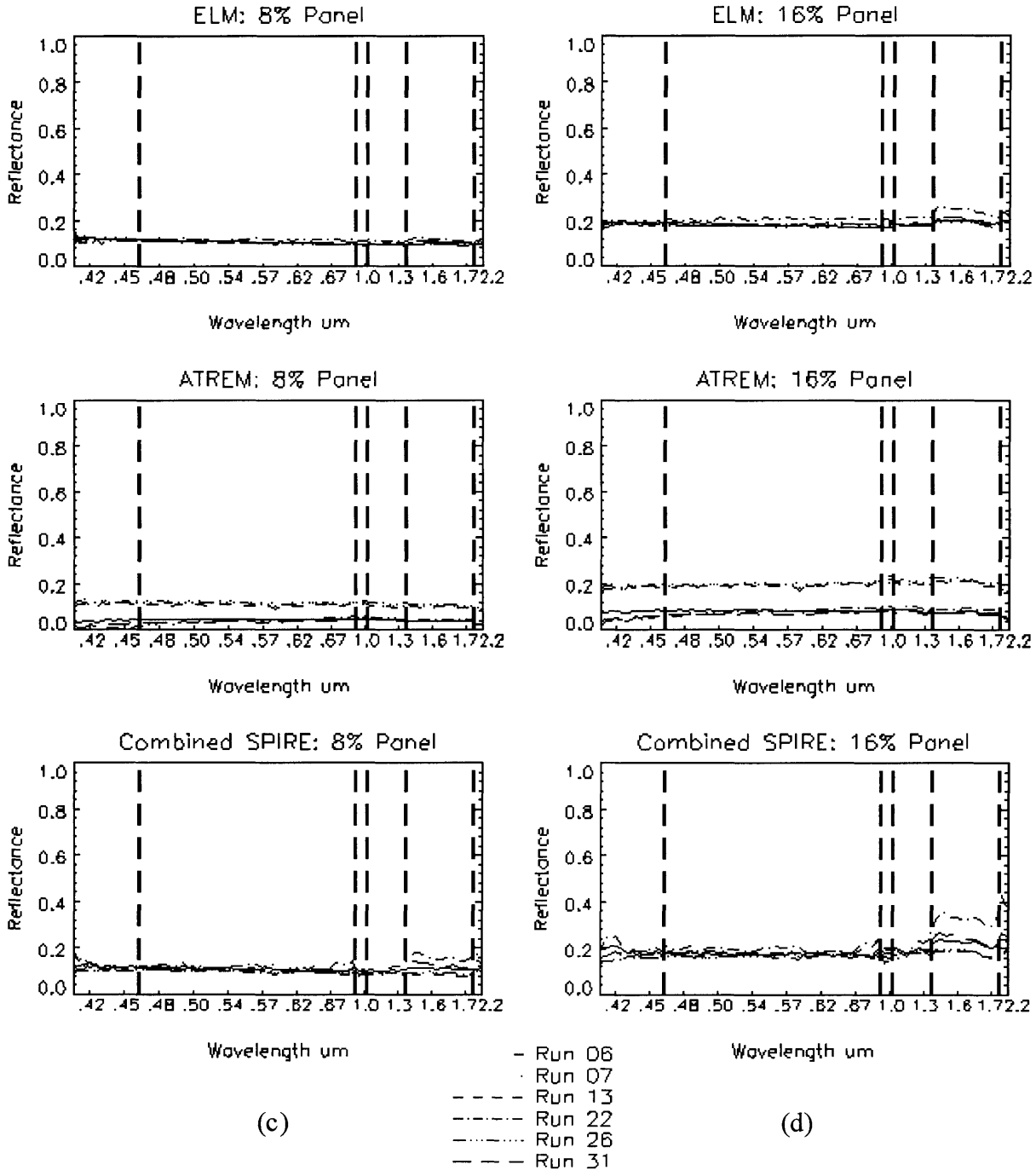


Figure F.1 (c) and (d). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the 8 percent panel (c) and the 16 percent panel (d).*

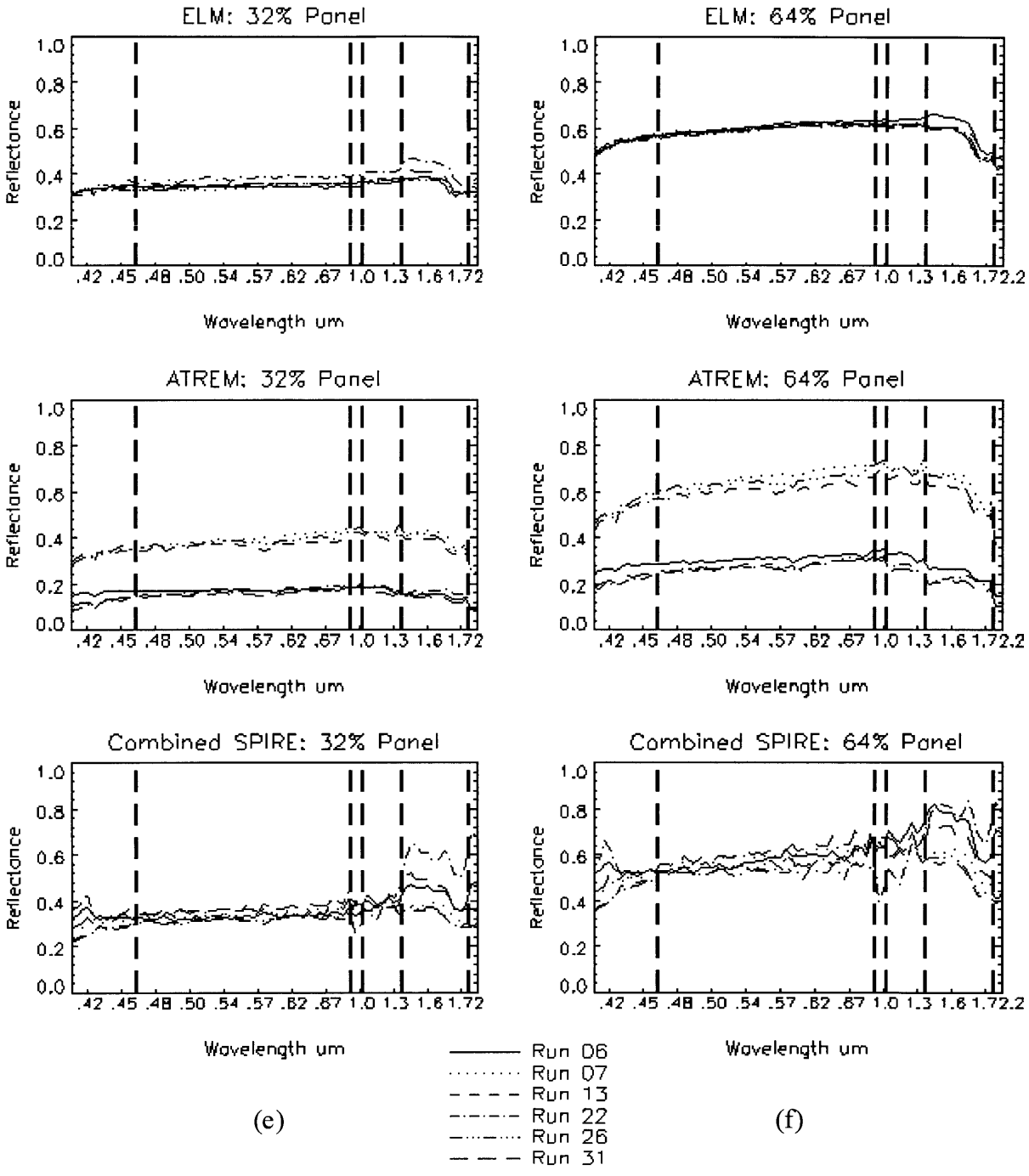


Figure F.1 (e) and (f). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the 32 percent panel (e) and the 64 percent panel (f).*

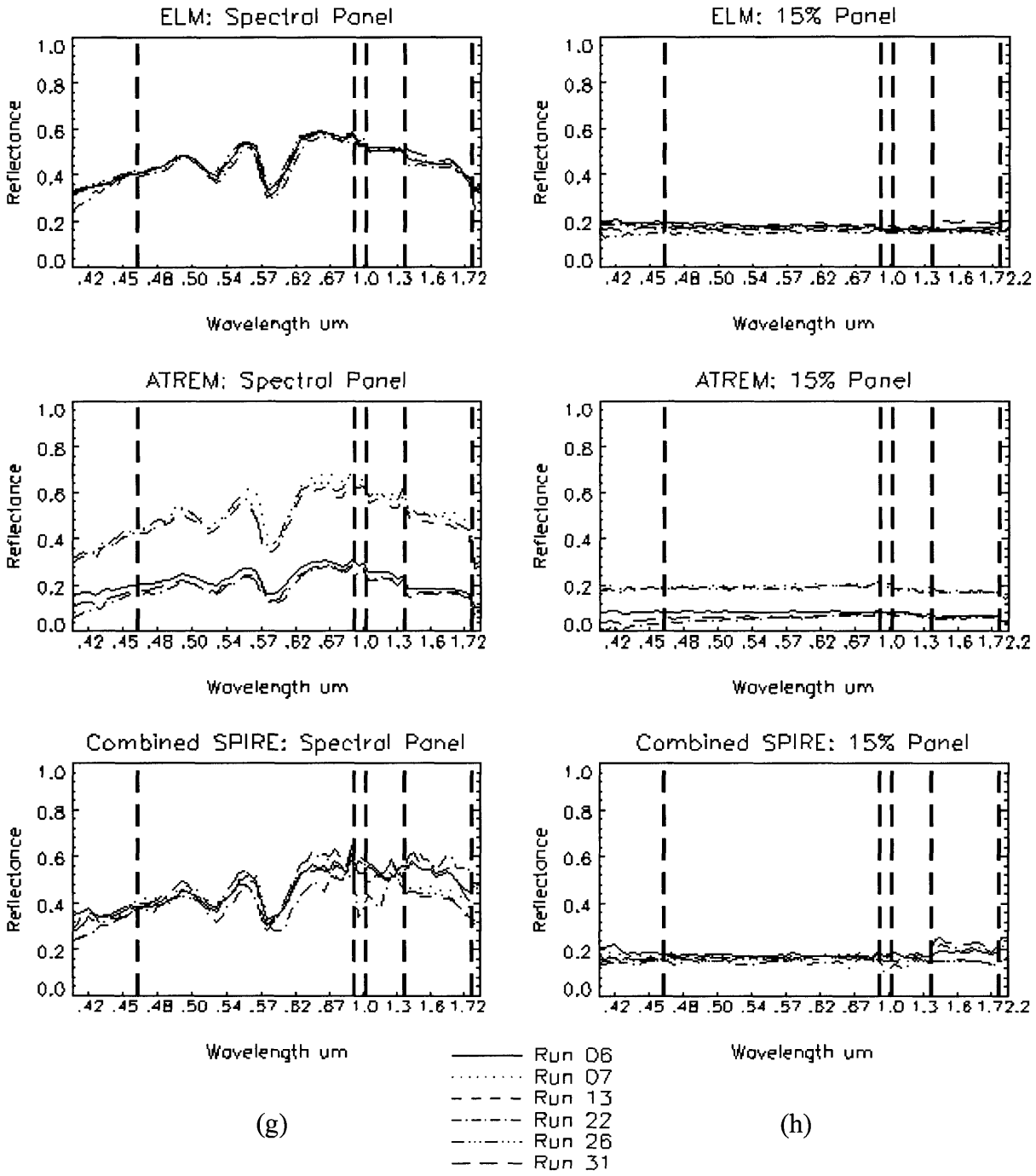


Figure F.1 (g) and (h). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the spectral panel (g) and the 15 percent panel (h).*

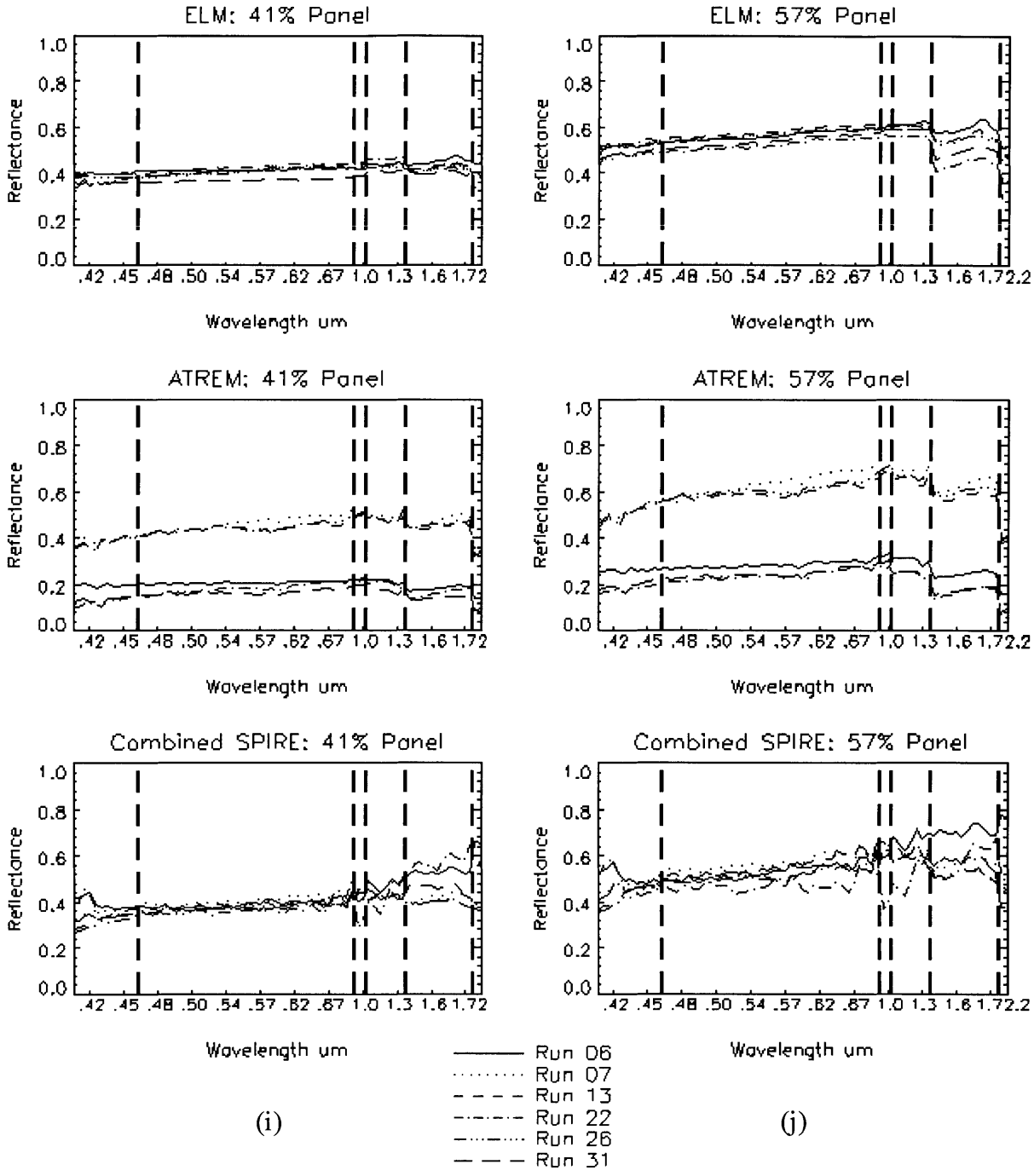


Figure F.1 (i) and (j). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the 41 percent panel (i) and the 57 percent panel (j).*

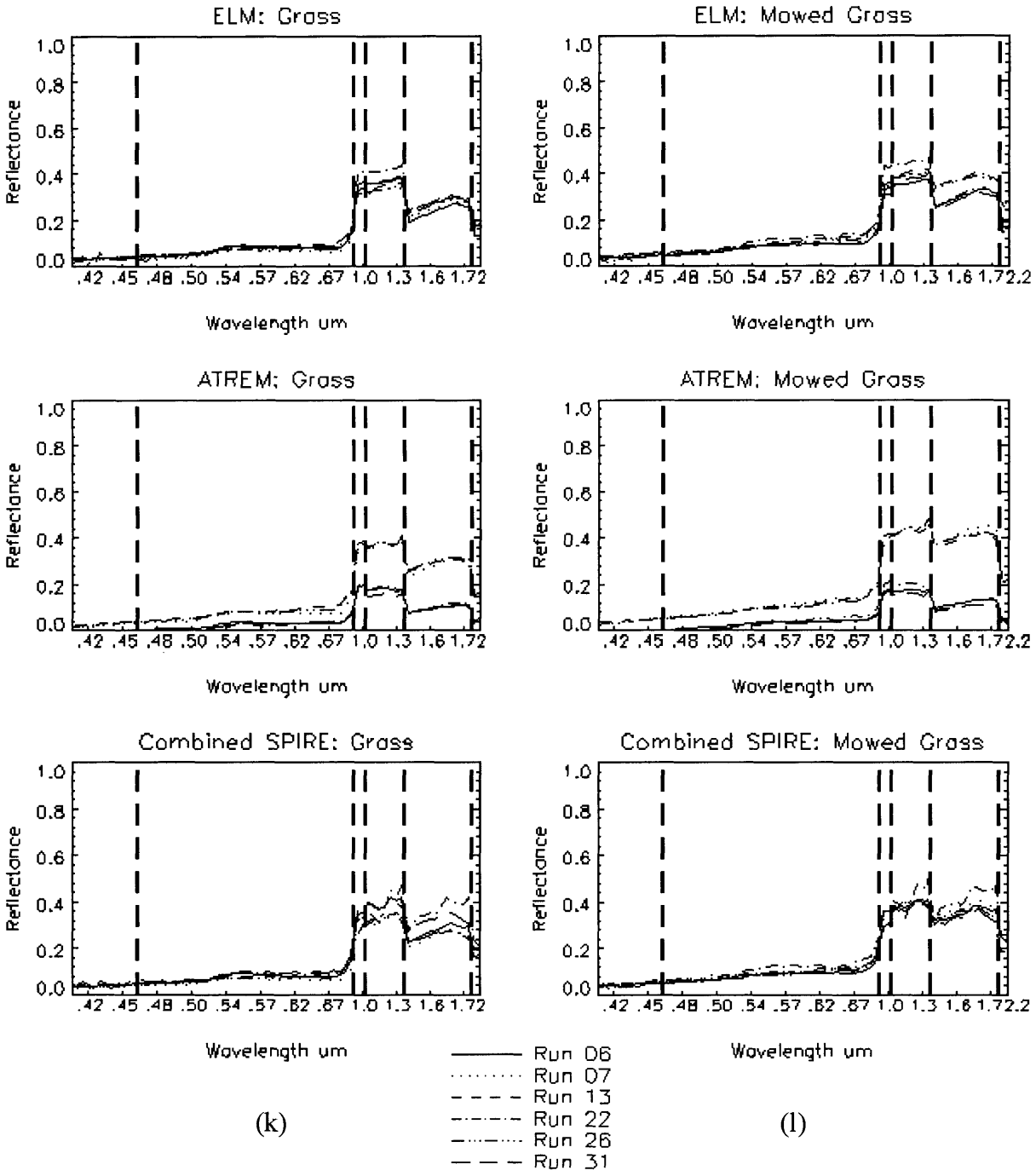


Figure F.1 (k) and (l). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on grass (k) and mowed grass (l).*

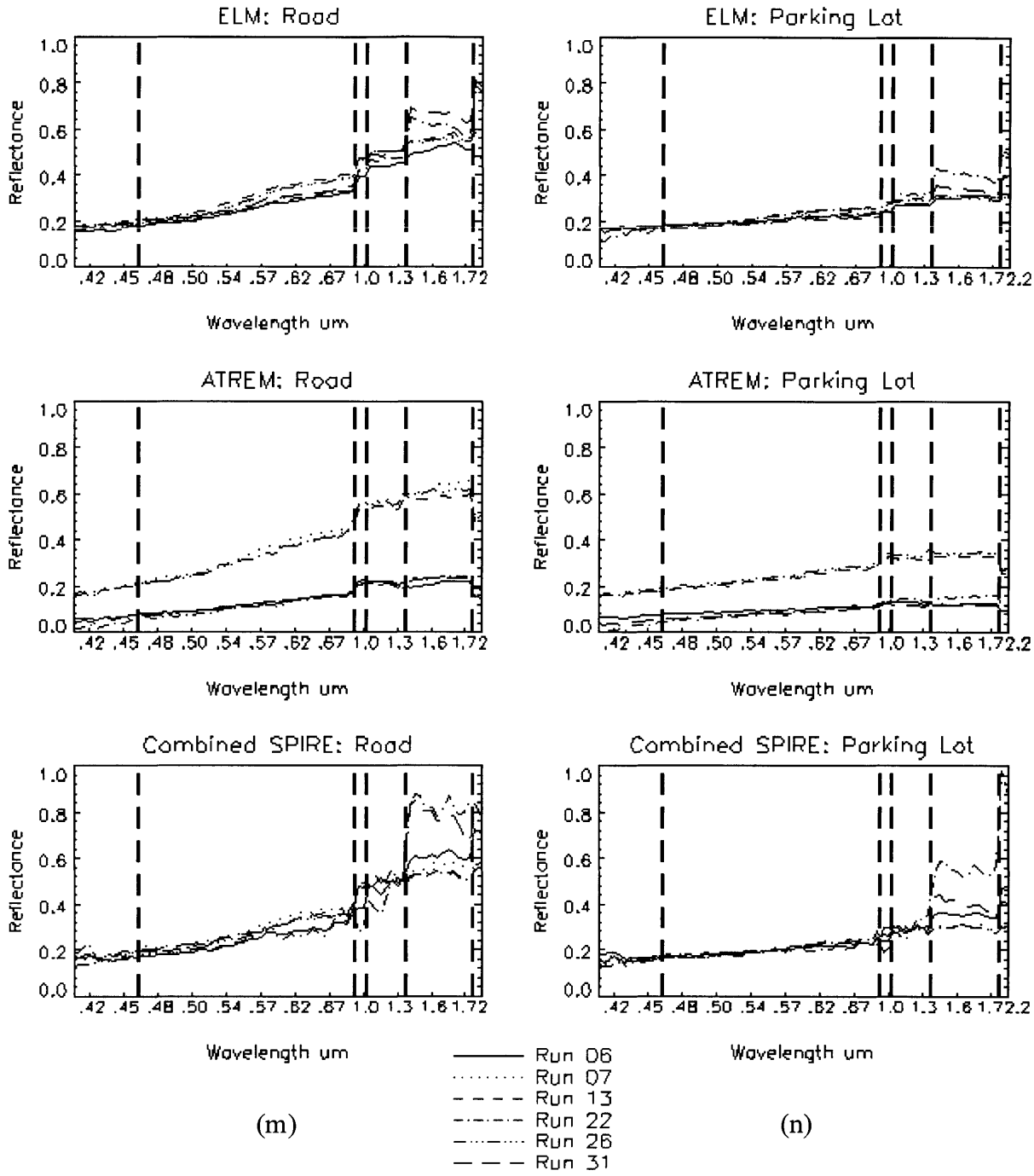


Figure F.1 (m) and (n). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the road (m) and the parking lot (n).*

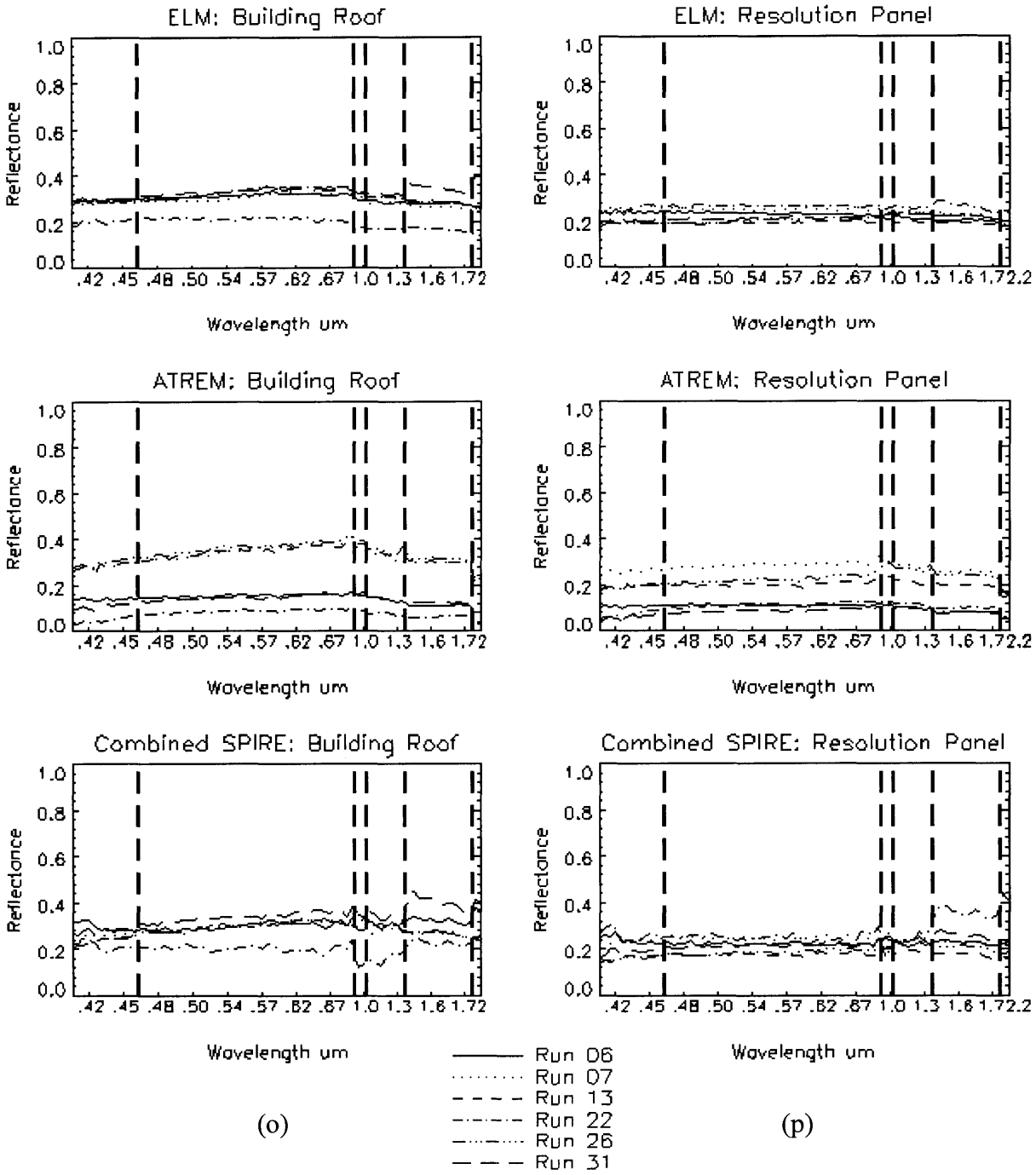


Figure F.1 (o) and (p). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the building roof (o) and the resolution panel (p).*

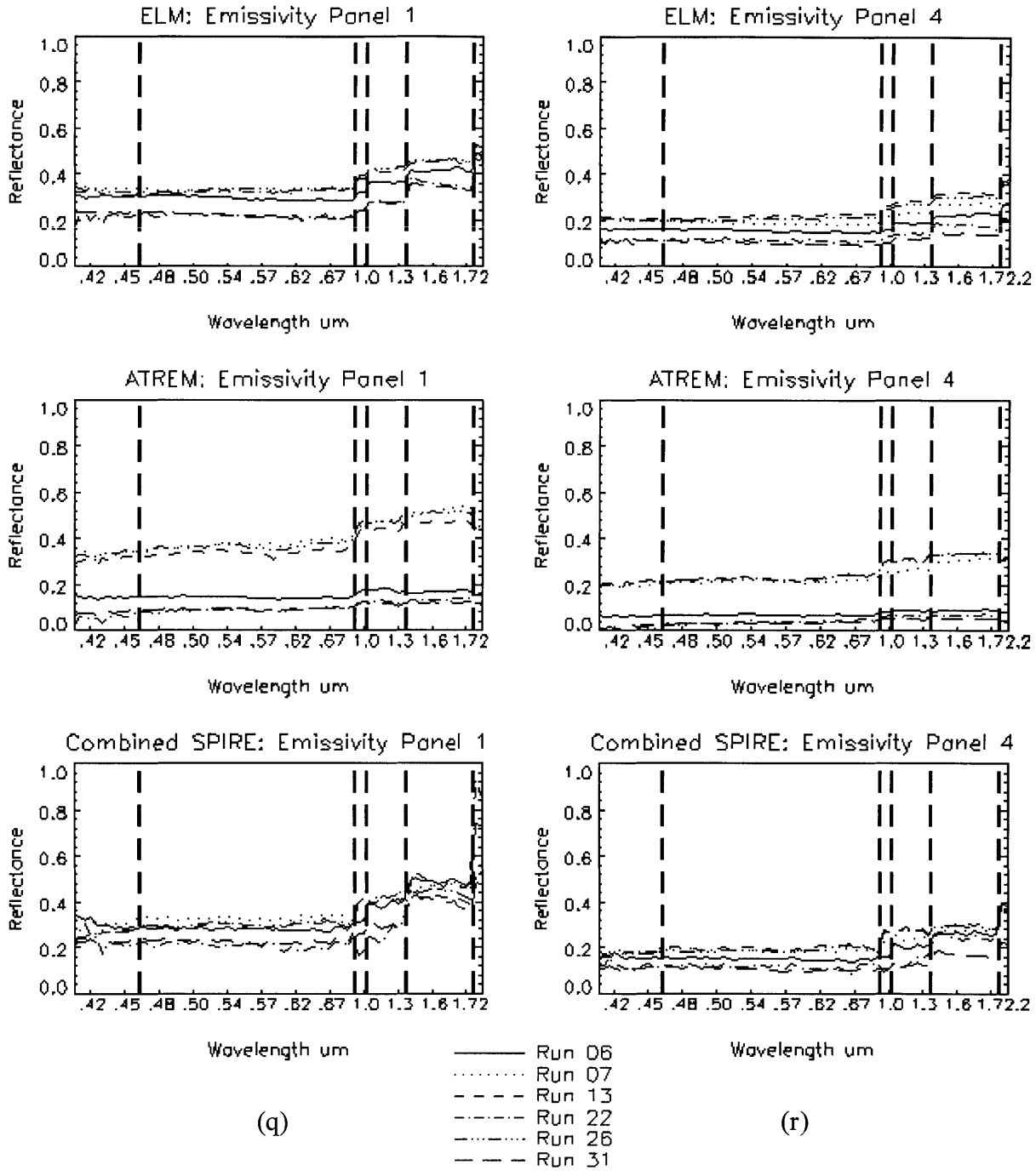
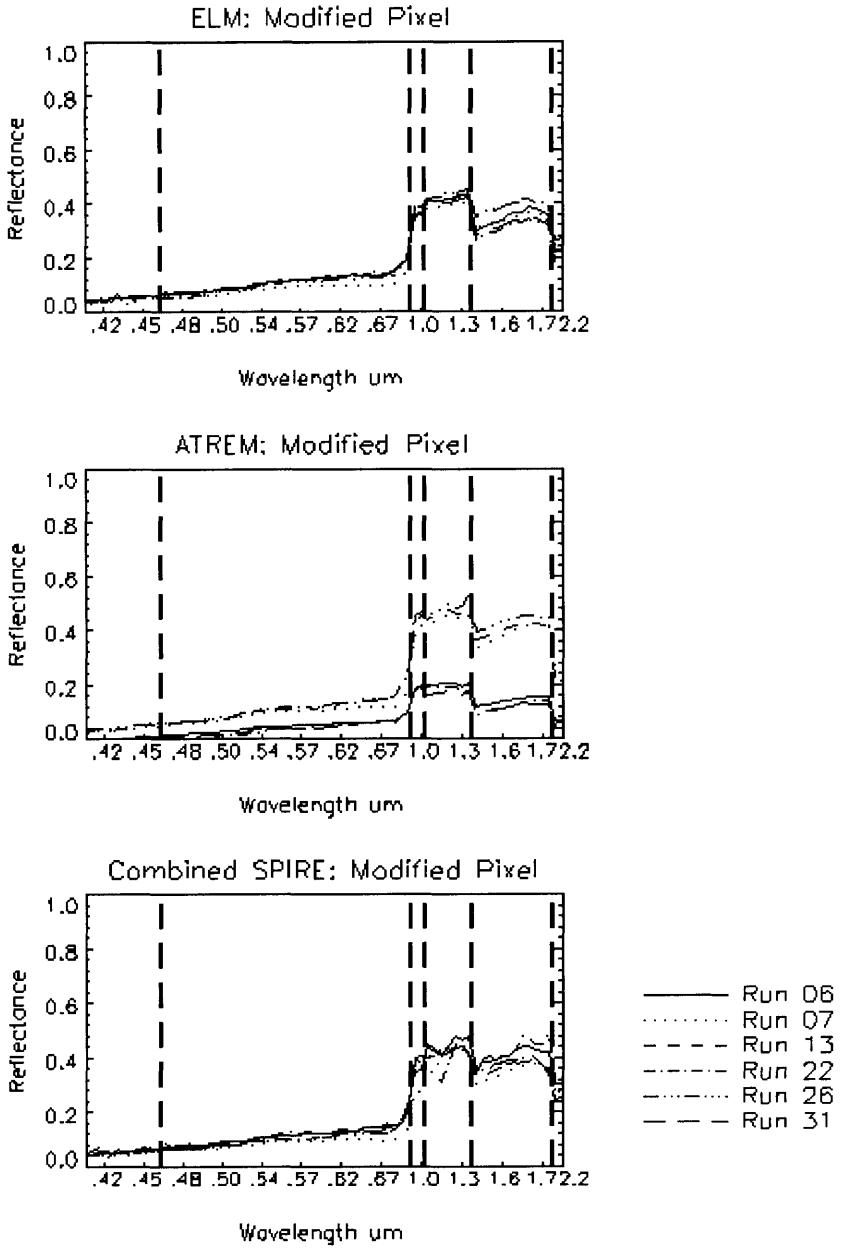


Figure F.1 (q) and (r). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on emissivity panel 1 (q) and emissivity panel 4 (r).*



(s)

Figure F.1(s). *ELM, ATREM, and Combined SPIRE spectral reflectance estimates for all six runs for a single pixel on the modified pixel. The modified pixel was a truck in Run 07, and changed to mowed grass in all subsequent Runs. For Run 07, a similar mowed grass pixel is plotted for comparison.*

Pixel	Band	ELM	ATREM	Combined SPIRE
2% Panel	Band 1	0.005	0.029	0.006
	Band 2	0.002	0.020	0.005
	Band 3	0.001	0.019	0.004
	Band 4	0.002	0.018	0.005
	Band 5	0.002	0.020	0.010
	Band 6	0.004	0.023	0.013

Table F.1(a). *Combined SPIRE average channel standard deviation from the mean over all runs for the 2 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
4% Panel	Band 1	0.006	0.039	0.009
	Band 2	0.003	0.028	0.005
	Band 3	0.002	0.025	0.006
	Band 4	0.002	0.023	0.005
	Band 5	0.005	0.024	0.015
	Band 6	0.012	0.028	0.026

Table F.1(b). *Combined SPIRE average channel standard deviation from the mean over all runs for the 4 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
8% Panel	Band 1	0.008	0.054	0.012
	Band 2	0.005	0.041	0.009
	Band 3	0.006	0.037	0.009
	Band 4	0.007	0.033	0.010
	Band 5	0.008	0.035	0.028
	Band 6	0.012	0.038	0.026

Table F.1(c). *Combined SPIRE average channel standard deviation from the mean over all runs for the 8 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
16% Panel	Band 1	0.007	0.072	0.020
	Band 2	0.012	0.065	0.012
	Band 3	0.013	0.071	0.021
	Band 4	0.014	0.067	0.015
	Band 5	0.020	0.071	0.057
	Band 6	0.017	0.065	0.080

Table F.1(d). *Combined SPIRE average channel standard deviation from the mean over all runs for the 16 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
32% Panel	Band 1	0.009	0.108	0.036
	Band 2	0.014	0.118	0.021
	Band 3	0.017	0.134	0.047
	Band 4	0.017	0.134	0.026
	Band 5	0.031	0.133	0.097
	Band 6	0.030	0.094	0.156

Table F.1(e). *Combined SPIRE average channel standard deviation from the mean over all runs for the 32 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
64% Panel	Band 1	0.007	0.161	0.050
	Band 2	0.006	0.194	0.035
	Band 3	0.010	0.217	0.093
	Band 4	0.011	0.221	0.055
	Band 5	0.022	0.214	0.105
	Band 6	0.019	0.134	0.176

Table F.1(f). *Combined SPIRE average channel standard deviation from the mean over all runs for the 64 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Spectral Panel	Band 1	0.016	0.125	0.026
	Band 2	0.014	0.165	0.031
	Band 3	0.009	0.201	0.089
	Band 4	0.007	0.190	0.040
	Band 5	0.013	0.173	0.071
	Band 6	0.032	0.115	0.072

Table F.1(g). *Combined SPIRE average channel standard deviation from the mean over all runs for the spectral panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
15% Panel	Band 1	0.020	0.077	0.019
	Band 2	0.013	0.069	0.012
	Band 3	0.009	0.067	0.027
	Band 4	0.009	0.059	0.016
	Band 5	0.017	0.059	0.034
	Band 6	0.036	0.057	0.038

Table F.1(h). *Combined SPIRE average channel standard deviation from the mean over all runs for the 15 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
41% Panel	Band 1	0.019	0.130	0.036
	Band 2	0.020	0.146	0.018
	Band 3	0.020	0.158	0.055
	Band 4	0.018	0.157	0.031
	Band 5	0.018	0.165	0.075
	Band 6	0.038	0.129	0.126

Table F.1(i). *Combined SPIRE average channel standard deviation from the mean over all runs for the 41 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
57% Panel	Band 1	0.024	0.165	0.044
	Band 2	0.022	0.197	0.036
	Band 3	0.016	0.220	0.101
	Band 4	0.020	0.222	0.058
	Band 5	0.054	0.226	0.071
	Band 6	0.087	0.163	0.138

Table F.1(j). *Combined SPIRE average channel standard deviation from the mean over all runs for 57 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Grass	Band 1	0.006	0.027	0.005
	Band 2	0.009	0.029	0.009
	Band 3	0.035	0.104	0.029
	Band 4	0.033	0.112	0.038
	Band 5	0.017	0.104	0.048
	Band 6	0.021	0.063	0.022

Table F.1(k). *Combined SPIRE average channel standard deviation from the mean over all runs for grass.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Mowed Grass	Band 1	0.007	0.034	0.006
	Band 2	0.010	0.040	0.010
	Band 3	0.040	0.124	0.034
	Band 4	0.030	0.145	0.023
	Band 5	0.042	0.161	0.031
	Band 6	0.054	0.101	0.031

Table F.1(l). *Combined SPIRE average channel standard deviation from the mean over all runs for mowed grass.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Road	Band 1	0.011	0.074	0.016
	Band 2	0.022	0.114	0.022
	Band 3	0.032	0.184	0.083
	Band 4	0.025	0.191	0.036
	Band 5	0.054	0.210	0.133
	Band 6	0.126	0.172	0.351

Table F.1(m). *Combined SPIRE average channel standard deviation from the mean over all runs for the road.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Parking Lot	Band 1	0.012	0.071	0.012
	Band 2	0.010	0.082	0.007
	Band 3	0.018	0.109	0.038
	Band 4	0.018	0.107	0.018
	Band 5	0.040	0.113	0.100
	Band 6	0.077	0.090	0.243

Table F.1(n). *Combined SPIRE average channel standard deviation from the mean over all runs for parking lot.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Building Roof	Band 1	0.037	0.108	0.033
	Band 2	0.047	0.126	0.047
	Band 3	0.058	0.137	0.085
	Band 4	0.056	0.124	0.062
	Band 5	0.058	0.114	0.062
	Band 6	0.081	0.088	0.076

Table F.1(o). *Combined SPIRE average channel standard deviation from the mean over all runs for the building roof.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Resolution Panel	Band 1	0.030	0.079	0.044
	Band 2	0.029	0.079	0.032
	Band 3	0.023	0.090	0.033
	Band 4	0.022	0.085	0.024
	Band 5	0.025	0.083	0.063
	Band 6	0.027	0.067	0.092

Table F.1(p). *Combined SPIRE average channel standard deviation from the mean over all runs for the resolution panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Emissivity Panel 1	Band 1	0.052	0.131	0.040
	Band 2	0.054	0.140	0.047
	Band 3	0.071	0.174	0.097
	Band 4	0.069	0.180	0.071
	Band 5	0.053	0.198	0.038
	Band 6	0.028	0.174	0.168

Table F.1(q). *Combined SPIRE average channel standard deviation from the mean over all runs for emissivity panel 1.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Emissivity Panel 4	Band 1	0.046	0.098	0.031
	Band 2	0.049	0.098	0.040
	Band 3	0.066	0.123	0.079
	Band 4	0.067	0.125	0.064
	Band 5	0.070	0.141	0.047
	Band 6	0.075	0.134	0.061

Table F.1(r). *Combined SPIRE average channel standard deviation from the mean over all runs for emissivity panel 4.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Modified Pixel	Band 1	0.006	0.032	0.006
	Band 2	0.011	0.038	0.012
	Band 3	0.017	0.139	0.052
	Band 4	0.016	0.154	0.031
	Band 5	0.033	0.156	0.031
	Band 6	0.034	0.094	0.032

Table F.1(s). *Combined SPIRE average channel standard deviation from the mean over all runs for the modified pixel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
2% Panel	Band 1	11.2	677.0	12.3
	Band 2	4.5	98.4	11.0
	Band 3	2.0	48.5	9.6
	Band 4	3.4	49.4	9.5
	Band 5	4.0	56.5	18.9
	Band 6	8.4	74.1	25.1

Table F.2(a). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 2 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
4% Panel	Band 1	8.5	132.2	12.0
	Band 2	5.1	70.8	7.4
	Band 3	2.9	48.7	10.2
	Band 4	2.9	49.2	8.5
	Band 5	8.3	57.3	23.0
	Band 6	20.0	76.3	42.0

Table F.2(b). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 4 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
8% Panel	Band 1	6.4	80.1	10.8
	Band 2	4.8	54.3	8.2
	Band 3	5.7	45.8	9.6
	Band 4	6.9	45.1	10.2
	Band 5	8.2	49.5	25.5
	Band 6	10.7	62.7	21.5

Table F.2(c). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 8 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
16% Panel	Band 1	3.9	59.1	11.6
	Band 2	6.4	47.7	7.0
	Band 3	7.2	46.4	11.8
	Band 4	7.3	46.3	7.9
	Band 5	9.7	49.4	25.5
	Band 6	8.1	57.7	32.4

Table F.2(d). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 16 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
32% Panel	Band 1	2.6	46.9	11.8
	Band 2	4.0	43.6	6.2
	Band 3	4.5	43.7	13.0
	Band 4	4.6	45.9	6.9
	Band 5	8.1	49.5	23.4
	Band 6	9.2	53.8	39.1

Table F.2(e). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 32 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
64% Panel	Band 1	1.3	42.6	10.5
	Band 2	1.1	42.2	6.1
	Band 3	1.5	42.1	15.2
	Band 4	1.8	45.8	8.8
	Band 5	3.8	51.8	17.0
	Band 6	4.3	55.8	33.2

Table F.2(f). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 64 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Spectral Panel	Band 1	4.6	49.8	8.4
	Band 2	3.2	45.1	6.8
	Band 3	1.7	44.2	16.6
	Band 4	1.4	47.5	7.9
	Band 5	2.9	53.4	14.6
	Band 6	9.8	61.6	18.7

Table F.2(g). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the spectral panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
15% Panel	Band 1	11.2	66.9	11.8
	Band 2	7.5	53.6	7.1
	Band 3	5.8	48.3	16.9
	Band 4	5.7	47.6	10.0
	Band 5	10.2	50.7	19.3
	Band 6	21.1	60.0	19.7

Table F.2(h). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 15 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
41% Panel	Band 1	5.0	48.4	10.3
	Band 2	4.9	46.3	4.7
	Band 3	4.8	44.3	13.2
	Band 4	4.3	46.5	7.2
	Band 5	4.2	53.1	16.1
	Band 6	9.8	59.2	27.2

Table F.2(i). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 41 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
57% Panel	Band 1	4.7	46.0	9.5
	Band 2	3.9	45.5	6.8
	Band 3	2.7	44.0	17.2
	Band 4	3.3	47.5	9.8
	Band 5	10.2	56.6	12.3
	Band 6	20.4	65.1	27.5

Table F.2(j). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for 57 percent panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Grass	Band 1	16.0	238.9	13.0
	Band 2	11.6	78.7	12.2
	Band 3	10.2	37.3	8.7
	Band 4	9.2	41.1	10.4
	Band 5	6.5	54.4	16.6
	Band 6	12.8	67.4	12.1

Table F.2(k). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for grass.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Mowed Grass	Band 1	17.5	1843.8	12.8
	Band 2	10.5	70.2	10.5
	Band 3	11.5	42.7	10.0
	Band 4	7.7	47.2	5.8
	Band 5	12.6	62.5	8.4
	Band 6	24.7	74.6	12.9

Table F.2(l). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for mowed grass.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Road	Band 1	6.4	64.3	9.7
	Band 2	7.6	52.9	8.2
	Band 3	7.2	48.5	18.9
	Band 4	5.1	49.9	7.5
	Band 5	9.4	50.6	20.9
	Band 6	19.9	51.9	45.0

Table F.2(m). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for road.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Parking Lot	Band 1	7.2	67.3	8.2
	Band 2	4.9	51.4	3.5
	Band 3	6.6	47.8	14.6
	Band 4	6.4	47.6	6.2
	Band 5	12.2	48.5	27.4
	Band 6	21.3	48.3	53.6

Table F.2(n). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the parking lot.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Building Roof	Band 1	13.3	56.1	13.2
	Band 2	15.3	52.1	15.9
	Band 3	19.6	53.2	28.7
	Band 4	20.1	54.9	22.1
	Band 5	21.4	57.1	21.1
	Band 6	30.8	60.8	26.0

Table F.2(o). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for building roof.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Resolution Panel	Band 1	13.6	54.5	21.1
	Band 2	12.7	46.8	14.6
	Band 3	9.8	46.6	14.6
	Band 4	9.8	48.1	10.6
	Band 5	11.4	52.7	25.9
	Band 6	13.5	59.9	38.0

Table F.2(p). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the resolution panel.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Emissivity Panel 1	Band 1	17.9	63.2	15.2
	Band 2	18.9	59.9	17.1
	Band 3	21.6	59.3	29.7
	Band 4	18.9	60.1	19.5
	Band 5	12.7	61.8	8.5
	Band 6	5.6	61.3	27.8

Table F.2(q). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for emissivity panel 1.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Emissivity Panel 4	Band 1	27.7	86.1	20.5
	Band 2	29.9	73.3	25.7
	Band 3	34.6	69.7	41.8
	Band 4	32.2	69.8	30.9
	Band 5	29.1	71.8	18.5
	Band 6	25.4	73.8	18.2

Table F.2(r). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for emissivity panel 4.*

Pixel	Band	ELM	ATREM	Combined SPIRE
Modified Pixel	Band 1	11.9	151.4	12.2
	Band 2	10.2	58.8	11.0
	Band 3	4.5	44.7	14.3
	Band 4	3.9	47.5	7.5
	Band 5	9.5	59.0	8.2
	Band 6	14.5	67.9	11.7

Table F.2(s). *Combined SPIRE average channel standard deviation from the mean, as a percentage of the mean, over all runs for the modified pixel.*

Appendix G

Additional Chapter 7 Results

This appendix consists of two tables that support Figure 7.3. Table G.1 lists the average channel standard deviation across all six runs for each Band for the ELM, ATREM, Spatial SPIRE, Spectral SPIRE, and Combined SPIRE estimators for the reflectance estimates on the 19 selected pixels shown in Figure 7.3. Table G.2 list the same information as a percentage of the channel mean across the six runs.

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
2% Panel	Band 1	0.005	0.029	0.005	0.004	0.006
	Band 2	0.002	0.020	0.005	0.002	0.005
	Band 3	0.001	0.019	0.085	0.006	0.004
	Band 4	0.002	0.018	0.080	0.003	0.005
	Band 5	0.002	0.020	0.058	0.004	0.010
	Band 6	0.004	0.023	0.044	0.011	0.013

Table G.1(a). *Average channel standard deviation from the mean over all runs for the 2 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
4% Panel	Band 1	0.006	0.039	0.007	0.006	0.009
	Band 2	0.003	0.028	0.004	0.003	0.005
	Band 3	0.002	0.025	0.080	0.009	0.006
	Band 4	0.002	0.023	0.077	0.004	0.005
	Band 5	0.005	0.024	0.056	0.004	0.015
	Band 6	0.012	0.028	0.044	0.013	0.026

Table G.1(b). *Average channel standard deviation from the mean over all runs for the 4 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
8% Panel	Band 1	0.008	0.054	0.010	0.007	0.012
	Band 2	0.005	0.041	0.008	0.004	0.009
	Band 3	0.006	0.037	0.071	0.011	0.009
	Band 4	0.007	0.033	0.071	0.006	0.010
	Band 5	0.008	0.035	0.055	0.005	0.028
	Band 6	0.012	0.038	0.039	0.015	0.026

Table G.1(c). *Average channel standard deviation from the mean over all runs for the 8 percent* panel.

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
16% Panel	Band 1	0.007	0.072	0.013	0.006	0.020
	Band 2	0.012	0.065	0.012	0.004	0.012
	Band 3	0.013	0.071	0.045	0.017	0.021
	Band 4	0.014	0.067	0.050	0.010	0.015
	Band 5	0.020	0.071	0.043	0.009	0.057
	Band 6	0.017	0.065	0.028	0.020	0.080

Table G.1(d). *Average channel standard deviation from the mean over all runs for the 16 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
32% Panel	Band 1	0.009	0.108	0.021	0.009	0.036
	Band 2	0.014	0.118	0.019	0.009	0.021
	Band 3	0.017	0.134	0.007	0.026	0.047
	Band 4	0.017	0.134	0.005	0.022	0.026
	Band 5	0.031	0.133	0.024	0.017	0.097
	Band 6	0.030	0.094	0.027	0.029	0.156

Table G.1(e). *Average channel standard deviation from the mean over all runs for the 32 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
64% Panel	Band 1	0.007	0.161	0.032	0.014	0.050
	Band 2	0.006	0.194	0.034	0.013	0.035
	Band 3	0.010	0.217	0.079	0.041	0.093
	Band 4	0.011	0.221	0.065	0.031	0.055
	Band 5	0.022	0.214	0.048	0.025	0.105
	Band 6	0.019	0.134	0.051	0.041	0.176

Table G.1(f). *Average channel standard deviation from the mean over all runs for the 64 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Spectral Panel	Band 1	0.016	0.125	0.028	0.012	0.026
	Band 2	0.014	0.165	0.029	0.017	0.031
	Band 3	0.009	0.201	0.064	0.048	0.089
	Band 4	0.007	0.190	0.042	0.027	0.040
	Band 5	0.013	0.173	0.025	0.023	0.071
	Band 6	0.032	0.115	0.034	0.039	0.072

Table G.1(g). *Average channel standard deviation from the mean over all runs for the spectral panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
15% Panel	Band 1	0.020	0.077	0.020	0.007	0.019
	Band 2	0.013	0.069	0.012	0.005	0.012
	Band 3	0.009	0.067	0.046	0.016	0.027
	Band 4	0.009	0.059	0.052	0.008	0.016
	Band 5	0.017	0.059	0.045	0.007	0.034
	Band 6	0.036	0.057	0.035	0.022	0.038

Table G.1(h). *Average channel standard deviation from the mean over all runs for the 15 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
41% Panel	Band 1	0.019	0.130	0.031	0.011	0.036
	Band 2	0.020	0.146	0.022	0.009	0.018
	Band 3	0.020	0.158	0.029	0.032	0.055
	Band 4	0.018	0.157	0.019	0.017	0.031
	Band 5	0.018	0.165	0.028	0.019	0.075
	Band 6	0.038	0.129	0.058	0.042	0.126

Table G.1(i). *Average channel standard deviation from the mean over all runs for the 41 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
57% Panel	Band 1	0.024	0.165	0.040	0.019	0.044
	Band 2	0.022	0.197	0.039	0.014	0.036
	Band 3	0.016	0.220	0.082	0.048	0.101
	Band 4	0.020	0.222	0.070	0.024	0.058
	Band 5	0.054	0.226	0.070	0.026	0.071
	Band 6	0.087	0.163	0.096	0.060	0.138

Table G.1(j). *Average channel standard deviation from the mean over all runs for 57 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Grass	Band 1	0.006	0.027	0.005	0.004	0.005
	Band 2	0.009	0.029	0.008	0.006	0.009
	Band 3	0.035	0.104	0.020	0.032	0.029
	Band 4	0.033	0.112	0.021	0.024	0.038
	Band 5	0.017	0.104	0.024	0.022	0.048
	Band 6	0.021	0.063	0.018	0.032	0.022

Table G.1(k). *Average channel standard deviation from the mean over all runs for grass.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Mowed Grass	Band 1	0.007	0.034	0.005	0.006	0.006
	Band 2	0.010	0.040	0.008	0.007	0.010
	Band 3	0.040	0.124	0.011	0.048	0.034
	Band 4	0.030	0.145	0.012	0.024	0.023
	Band 5	0.042	0.161	0.024	0.018	0.031
	Band 6	0.054	0.101	0.027	0.033	0.031

Table G.1(l). *Average channel standard deviation from the mean over all runs for mowed grass.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Road	Band 1	0.011	0.074	0.015	0.009	0.016
	Band 2	0.022	0.114	0.023	0.010	0.022
	Band 3	0.032	0.184	0.050	0.023	0.083
	Band 4	0.025	0.191	0.044	0.032	0.036
	Band 5	0.054	0.210	0.029	0.028	0.133
	Band 6	0.126	0.172	0.031	0.046	0.351

Table G.1(m). *Average channel standard deviation from the mean over all runs for the road.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Parking Lot	Band 1	0.012	0.071	0.013	0.010	0.012
	Band 2	0.010	0.082	0.005	0.006	0.007
	Band 3	0.018	0.109	0.016	0.021	0.038
	Band 4	0.018	0.107	0.023	0.018	0.018
	Band 5	0.040	0.113	0.033	0.012	0.100
	Band 6	0.077	0.090	0.035	0.022	0.243

Table G.1(n). *Average channel standard deviation from the mean over all runs for parking lot.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Building Roof	Band 1	0.037	0.108	0.035	0.012	0.033
	Band 2	0.047	0.126	0.043	0.012	0.047
	Band 3	0.058	0.137	0.017	0.034	0.085
	Band 4	0.056	0.124	0.018	0.023	0.062
	Band 5	0.058	0.114	0.038	0.014	0.062
	Band 6	0.081	0.088	0.048	0.037	0.076

Table G.1(o). *Average channel standard deviation from the mean over all runs for the building roof.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Resolution Panel	Band 1	0.030	0.079	0.036	0.065	0.044
	Band 2	0.029	0.079	0.032	0.058	0.032
	Band 3	0.023	0.090	0.040	0.043	0.033
	Band 4	0.022	0.085	0.047	0.037	0.024
	Band 5	0.025	0.083	0.042	0.027	0.063
	Band 6	0.027	0.067	0.032	0.030	0.092

Table G.1(p). *Average channel standard deviation from the mean over all runs for the resolution panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Emissivity Panel 1	Band 1	0.052	0.131	0.052	0.022	0.040
	Band 2	0.054	0.140	0.048	0.017	0.047
	Band 3	0.071	0.174	0.048	0.040	0.097
	Band 4	0.069	0.180	0.050	0.041	0.071
	Band 5	0.053	0.198	0.048	0.027	0.038
	Band 6	0.028	0.174	0.067	0.052	0.168

Table G.1(q). *Average channel standard deviation from the mean over all runs for emissivity panel 1.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Emissivity Panel 4	Band 1	0.046	0.098	0.041	0.009	0.031
	Band 2	0.049	0.098	0.039	0.010	0.040
	Band 3	0.066	0.123	0.027	0.021	0.079
	Band 4	0.067	0.125	0.032	0.015	0.064
	Band 5	0.070	0.141	0.031	0.026	0.047
	Band 6	0.075	0.134	0.055	0.034	0.061

Table G.1(r). *Average channel standard deviation from the mean over all runs for emissivity panel 4.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Modified Pixel	Band 1	0.006	0.032	0.005	0.276	0.006
	Band 2	0.011	0.038	0.010	0.260	0.012
	Band 3	0.017	0.139	0.015	0.246	0.052
	Band 4	0.016	0.154	0.020	0.208	0.031
	Band 5	0.033	0.156	0.021	0.087	0.031
	Band 6	0.034	0.094	0.016	0.074	0.032

Table G.1(s). *Average channel standard deviation from the mean over all runs for the modified pixel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
2% Panel	Band 1	11.2	677.0	11.1	8.4	12.3
	Band 2	4.5	98.4	11.2	5.0	11.0
	Band 3	2.0	48.5	77.2	13.2	9.6
	Band 4	3.4	49.4	75.2	6.4	9.5
	Band 5	4.0	56.5	59.0	7.8	18.9
	Band 6	8.4	74.1	47.4	21.1	25.1

Table G.2(a). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 2 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
4% Panel	Band 1	8.5	132.2	9.3	8.0	12.0
	Band 2	5.1	70.8	6.8	4.7	7.4
	Band 3	2.9	48.7	66.6	15.7	10.2
	Band 4	2.9	49.2	66.6	6.4	8.5
	Band 5	8.3	57.3	53.2	6.6	23.0
	Band 6	20.0	76.3	43.4	22.0	42.0

Table G.2(b). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 4 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
8% Panel	Band 1	6.4	80.1	9.1	5.7	10.8
	Band 2	4.8	54.3	7.5	3.6	8.2
	Band 3	5.7	45.8	48.0	12.8	9.6
	Band 4	6.9	45.1	49.3	6.4	10.2
	Band 5	8.2	49.5	39.5	5.1	25.5
	Band 6	10.7	62.7	28.4	14.3	21.5

Table G.2(c). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 8 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
16% Panel	Band 1	3.9	59.1	7.3	3.3	11.6
	Band 2	6.4	47.7	6.8	2.5	7.0
	Band 3	7.2	46.4	21.0	10.5	11.8
	Band 4	7.3	46.3	23.3	5.8	7.9
	Band 5	9.7	49.4	19.5	4.6	25.5
	Band 6	8.1	57.7	12.8	10.3	32.4

Table G.2(d). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for 16 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
32% Panel	Band 1	2.6	46.9	7.1	2.9	11.8
	Band 2	4.0	43.6	6.0	2.6	6.2
	Band 3	4.5	43.7	2.1	7.3	13.0
	Band 4	4.6	45.9	1.4	6.3	6.9
	Band 5	8.1	49.5	6.8	4.8	23.4
	Band 6	9.2	53.8	8.8	9.4	39.1

Table G.2(e). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 32 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
64% Panel	Band 1	1.3	42.6	6.5	2.8	10.5
	Band 2	1.1	42.2	6.3	2.2	6.1
	Band 3	1.5	42.1	14.3	6.6	15.2
	Band 4	1.8	45.8	11.6	5.1	8.8
	Band 5	3.8	51.8	9.0	4.5	17.0
	Band 6	4.3	55.8	12.6	9.7	33.2

Table G.2(f). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for 64 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Spectral Panel	Band 1	4.6	49.8	8.6	3.4	8.4
	Band 2	3.2	45.1	6.8	3.6	6.8
	Band 3	1.7	44.2	13.0	8.9	16.6
	Band 4	1.4	47.5	8.7	5.4	7.9
	Band 5	2.9	53.4	5.9	5.3	14.6
	Band 6	9.8	61.6	10.7	11.5	18.7

Table G.2(g). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for the spectral panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
15% Panel	Band 1	11.2	66.9	12.1	3.5	11.8
	Band 2	7.5	53.6	7.2	2.7	7.1
	Band 3	5.8	48.3	23.0	10.1	16.9
	Band 4	5.7	47.6	26.5	5.2	10.0
	Band 5	10.2	50.7	23.6	4.5	19.3
	Band 6	21.1	60.0	18.5	13.3	19.7

Table G.2(h). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for 15 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
41% Panel	Band 1	5.0	48.4	8.9	2.9	10.3
	Band 2	4.9	46.3	6.0	2.2	4.7
	Band 3	4.8	44.3	7.3	7.7	13.2
	Band 4	4.3	46.5	4.7	4.1	7.2
	Band 5	4.2	53.1	7.0	4.3	16.1
	Band 6	9.8	59.2	15.8	10.7	27.2

Table G.2(i). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for the 41 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
57% Panel	Band 1	4.7	46.0	8.8	3.6	9.5
	Band 2	3.9	45.5	7.8	2.5	6.8
	Band 3	2.7	44.0	15.3	8.0	17.2
	Band 4	3.3	47.5	12.8	4.0	9.8
	Band 5	10.2	56.6	13.9	4.5	12.3
	Band 6	20.4	65.1	23.3	13.1	27.5

Table G.2(j). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for 57 percent panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Grass	Band 1	16.0	238.9	12.5	11.9	13.0
	Band 2	11.6	78.7	11.3	9.4	12.2
	Band 3	10.2	37.3	6.0	10.0	8.7
	Band 4	9.2	41.1	6.1	7.5	10.4
	Band 5	6.5	54.4	8.6	8.9	16.6
	Band 6	12.8	67.4	9.5	21.6	12.1

Table G.2(k). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for grass.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Mowed Grass	Band 1	17.5	1843.8	10.3	11.9	12.8
	Band 2	10.5	70.2	8.7	7.1	10.5
	Band 3	11.5	42.7	3.2	13.9	10.0
	Band 4	7.7	47.2	3.2	6.3	5.8
	Band 5	12.6	62.5	7.0	5.0	8.4
	Band 6	24.7	74.6	11.3	14.0	12.9

Table G.2(l). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for mowed grass.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Road	Band 1	6.4	64.3	9.3	4.9	9.7
	Band 2	7.6	52.9	9.1	3.5	8.2
	Band 3	7.2	48.5	11.8	5.0	18.9
	Band 4	5.1	49.9	9.6	6.9	7.5
	Band 5	9.4	50.6	5.5	5.1	20.9
	Band 6	19.9	51.9	5.7	8.6	45.0

Table G.2(m). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for road.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Parking Lot	Band 1	7.2	67.3	8.5	5.8	8.2
	Band 2	4.9	51.4	2.9	3.1	3.5
	Band 3	6.6	47.8	5.6	8.3	14.6
	Band 4	6.4	47.6	7.6	6.7	6.2
	Band 5	12.2	48.5	10.0	3.9	27.4
	Band 6	21.3	48.3	10.7	6.9	53.6

Table G.2(n). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for the parking lot.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Building Roof	Band 1	13.3	56.1	13.9	4.2	13.2
	Band 2	15.3	52.1	15.3	3.6	15.9
	Band 3	19.6	53.2	5.3	11.2	28.7
	Band 4	20.1	54.9	6.1	8.4	22.1
	Band 5	21.4	57.1	13.4	5.1	21.1
	Band 6	30.8	60.8	18.1	14.7	26.0

Table G.2(o). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for building roof.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Resolution Panel	Band 1	13.6	54.5	17.7	31.7	21.1
	Band 2	12.7	46.8	15.3	28.9	14.6
	Band 3	9.8	46.6	16.0	21.5	14.6
	Band 4	9.8	48.1	19.3	19.3	10.6
	Band 5	11.4	52.7	17.9	14.0	25.9
	Band 6	13.5	59.9	15.3	16.2	38.0

Table G.2(p). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for the resolution panel.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Emissivity Panel 1	Band 1	17.9	63.2	19.5	7.0	15.2
	Band 2	18.9	59.9	18.2	5.4	17.1
	Band 3	21.6	59.3	13.9	11.0	29.7
	Band 4	18.9	60.1	13.4	10.7	19.5
	Band 5	12.7	61.8	11.7	5.9	8.5
	Band 6	5.6	61.3	14.6	10.2	27.8

Table G.2(q). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for emissivity panel 1.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Emissivity Panel 4	Band 1	27.7	86.1	26.3	4.9	20.5
	Band 2	29.9	73.3	25.5	5.5	25.7
	Band 3	34.6	69.7	11.5	10.1	41.8
	Band 4	32.2	69.8	13.3	6.8	30.9
	Band 5	29.1	71.8	11.9	9.3	18.5
	Band 6	25.4	73.8	18.3	10.3	18.2

Table G.2(r). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for emissivity panel 4.*

Pixel	Band	ELM	ATREM	Spatial SPIRE	Spectral SPIRE	Combined SPIRE
Modified Pixel	Band 1	11.9	151.4	10.6	50.5	12.2
	Band 2	10.2	58.8	9.8	47.1	11.0
	Band 3	4.5	44.7	4.3	35.7	14.3
	Band 4	3.9	47.5	4.9	31.2	7.5
	Band 5	9.5	59.0	6.0	21.0	8.2
	Band 6	14.5	67.9	6.3	24.5	11.7

Table G.2(s). *Average channel standard deviation from the mean, as a percentage of the mean, over all runs for the modified pixel.*

References

Adler-Golden, S., A. Berk, L.S. Bernstein, S. Richsmeier, P.K. Acharya, M.W. Matthew, G.P. Anderson, C. Allred, L. Jeong, and J. Chetwynd, 1998a: "FLAASH, A MODTRAN4 Atmospheric Correction Package for Hypersoectral Data Retrievals and Simulations", *Proc. of the 7th Ann. JPL Airborne Earth Science Workshop,* JPL Publication 97-21, **1**, 9-14.

1997, "Atmospheric Removal Program **(ATREM) User's Guide**, Version 3.0," Center for the Study of Earth From Space (CSES) Cooperative Institute for Research in Environmental Sciences (CIRES), University of Colorado, Boulder, Colorado, 27 pages.

Barnes, P.L., *In-Scene Atmospheric Correction for Multispectral Imagery*, M.S. Thesis, College of Science, Rochester Institute of Technology, Rochester, NY, ©1997

Basedow, R.W., D.C. Armer, and M.E. Anderson, 1995, "HYDICE System: Implementation and Performance," in *Imaging Spectrometry*, M.R. Descour, J.M. Mooney, D.L. Perry, and L.R. Illing, eds. Proc. SPIE, Vol. 2480, 258-267.

Berk, A., L.S. Bernstein, G.P. Anderson, P.J. Acharya, D.C. Robertson, J.H. Chetwynd, and S.M. Adler-Golden, 1998, "MODTRAN Cloud and Multiple Scattering Upgrades with Application to AVIRIS, " Remote Sens. Environ., Vol. 65, 367-375.

Dowling, J.E., *The Retina: An Approachable Part of the Brain*, The Belknap Press, Cambridge, MA, ©1987

ENVI User's Guide, Version 3.1, Better Solutions Consulting Limited Liability Company, Lafayette, Colorado, USA, 1997, pp. 298-301.

Fukunaga, K, *Introduction to Statistical Pattern Recognition*, Second Edition, Academic Press, Inc., San Diego, CA, © 1990

Griffin, M.K., H.K. Burke, J. Vail, and S. Adler-Golden, "Evaluation of Atmospheric Compensation Algorithms for Hyperspectral Data Analysis", MIT Lincoln Laboratory Project Report HTAP-1, 9 June 1999.

Helstrom, C.W., *Probability and Stochastic Processes for Engineers*, Second Edition, Macmillan Publishin Company, New York, NY, © 1991

Horn, B.K.P., *Robot Vision*, The MIT Press, Cambridge, MA, pp. 188-196, ©1986

Hulbert, A.C., "Formal Connections Between Lightness Algorithms", *Journal of the Optical Society of America A*, vol. 3 pp. 1684-1693, 1986.

Hulbert, A.C., *The Computation of Color*, Ph.D. Thesis, Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, September, 1989.

Jain, A.K., **Dubes**, R.C., *Algorithms for Clustering Data*, Prentice Hall, Englewood Cliffs, NJ, © 1998

Kaiser, P.K., **Boynton**, R.M., *Human Color Vision*, Optical Society of America, USA, © 1996

Kaufman, Y.J, and C. Sendra, 1988, "Algorithm for Atmospheric Corrections", *Int. J. Rem. Sens.*, **9**, 1357-1381.

Kaufman, Y.J, and B-C. Gao, 1992, "Remote Sensing of Water Vapor in the Near IR from EOS/MODIS", *IEEE Trans. On Goes. & Rem. Sens.*, **30**, 871-884.

Kruse, F.A. and K.S. Kierein-Young, J.W. Boardman, "Mineral Mapping at Cuprite, Nevada with a 63-Channel Imaging Spectrometer.", Photogrammetric Engineering and Remote Sensing 56(1990):83-92.

Land, E., "The Retinex", American Scientist, vol. 52, pp. 247-264, 1964.

Land, E., "Recent Advances in Retinex Theory", *Vision Research*, vol. 26, no.1, pp. 7-21, 1986

Land, E., "An Alternative Techniques for the Computation of the Designator in the Retinex Theory of Color Vision", *Proc. Nat. Acad. Sci.*, vol. 83, pp. 3078-3080, 1986.

Lee, J., *Blind Noise Estimation and Compensation for Improved Characterization of Multivariate Processes*, Ph.D. Thesis, Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science, Cambridge, MA, © 2000

Lee, J. and **Staelin**, D.H., "Iterative signal-order and noise estimation for multivariate data", *Electronics Letters*, vol. 37, No. 2, pp. 134-135, January 18, 2001

Lillesand, T.M., and **Kiefer**, R.W., "Remote Sensing and Image Interpretation", Third Edition, John Wiley & Sons, Inc., New York, NY, ©1994.

Lim J.S., *Two-Dimensional Signal and Image Processing*, Prentice Hall, Inc. Englewood Cliffs, New Jersey, pp. 463-468, ©1990

Atmospheric Compensation Investigation ARM Site Ground Truth Data Collection Report, MTR-97-011, **Lockheed-Martin** Missiles & Space, Co., Inc., Palo Alto, CA.

Malvar, H.S., *Signal Processing with Lapped Transforms*, Artech House, Norwood, Massachusetts, © 1992

Nischan, M.L, Kerekes, J.P., Baum, J.E., Basedow, R.W., *Analysis of HYDICE Noise Characteristics and Their Impact on Subpixel Object Detection*, Imaging Spectrometry V; Proceedings of the Meeting, Denver, CO, July 19-21, 1999 (A00-14901 02-35), Bellingham, WA, Society of Photo-Optical Instrumentation Engineers (SPIE Proceedings. Vol. 3753), 1999, p. 112-123

Oppenheim, A.V., Schafer, R.W., *Digital Signal Processing*, Prentice Hall, Inc. Englewood Cliffs, New Jersey, 1975

Oppenheim, A.V., Schafer, R.W., Buck, J.R., *Digital Signal Processing,* Second Edition, Prentice Hall, Inc. Englewood Cliffs, New Jersey, 1999

Oppenheim, A.V., Willsky, A. S., Nawab, S.H., *Signals and Systems,* Second Edition, Prentice Hall, Inc. Englewood Cliffs, New Jersey, 1997

Otten, L.J., Meigs, A.D., Jones, B.A., Prinzing, P., Fronterhouse, D.S., "Payload Qualification and Optical Performance Test Results for the MightySat II.1 Hyperspectral Imager", *Sensors, Systems, and Next-Generation Satellites II*; Proceedings of the Meeting, Barcelona, Spain, Sept. 21-24, 1998 (A99-24101 05-19), Bellingham, WA, Society of Photo-Optical Instrumentation Engineers (SPIE Proceedings. Vol. 3498), 1998, p. 231-238

Papoulis, A., *Probability, Random Variables, and Stochastic Processes*, Third Edition, McGraw-Hill, Inc., New York, NY, © 1991

Pearlman, J., Segal, C., Liao, L., Carman, S, Folkman, M, Ungar, S. Browne, B., Ong, L., "Development and Operations of the EO-1 Hyperion Imaging Spectrometer", *Earth observing systems V*; Proceedings of the Conference, San Diego, CA, Aug. 2-4, 2000 (A01-23462 05-19), Bellingham, WA, Society of Photo-Optical Instrumentation Engineers (SPIE Proceedings. Vol. 4135), 2000, p. 243-253

Richards, J.A., "Remote Sensing Digital Image Analysis, An Introduction", Second Edition, Springer-Verlag, New York, NY, ©1993.

Rickard, L.J., R. Basedow, E. Zalewski, P. Silvergate, and L.M. Lander, 1993: HYDICE: An airborne system for hyperspectral imaging. *SPIE Proceedings,* **1937**, 173-179

Sanders, L.C., Kociolowicz, A.L., Brown, S.D., Raqueno, R.V., Schott, J.R., *Evaluation of Atmospheric Correction Techniques on Airborne Hyperspectral Remote Sensing Images*, RIT/DIRS Report #97/98-61, June 1998, Rochester Institute of Technology, Rochester NY

Schott, J.R., *Remote Sensing, The Image Chain Approach*, Oxford University Press, Inc., New York, NY, ©1997.

Schowengerdt, R.A., "Remote Sensing, Models and Methods for Image Processing", Second Edition, Academic Press, San Diego, CA, © 1997.

Spillman, L., **Werner**, J.S., *Visual Perception: The Neurophysiological Foundations*, Academic Press, San Diego, CA, © 1990

Strang, G., *Introduction to Linear Algebra*, Second Edition, Wellesley-Cambridge Press, Wellesley, MA, ©1998

Van Trees, H.L., *Detection, Estimation, and Modulation Theory*, Part I, John Wiley & Sons, New York, NY, ©1968

Vane, G., 1987: Airborne Visible/Infrared Imaging Spectrometer (AVIRIS). *JPL Publication 87-38*, 97 p.

Vane, G M. Chrisp, H. Enmark, S. Macenka and J. Solomon, 1984: Airborne Visible/Infrared Imaging Spectrometer: An advanced tool for earth remote sensing. *Proc. 1984 IEEE Int'l Geosc. & Rem. Sens. Symp.*, SP215, 751-757.

Vane, G., R.O. Greene, T.G. Chrien, H.T. Enmark, E.G. Hansen, and W.M Porter, 1993, "The Airborne Visible/Infrared Imaging Spectrometer (AVIRIS)," Remote Sensing Environ., Vol. 44, 127-143.

Vermote, E., D. Tanre, J.L. Deuze, M. Herman, and J.J. Morcetee, 1997, "Second Simulation of the Satellite Signal in the Solar Spectrum (6S)., *6S User's Guide Version 2*, NASA-GSFC, Greenbelt, MD, 218 p.