HYPERSPECTRAL IMAGE CLASSIFICATION USING SPECTRAL HISTOGRAMS AND SEMI-SUPERVISED LEARNING

By

Sol Marie Cruz Rivera

A thesis submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

ELECTRICAL ENGINEERING

UNIVERSITY OF PUERTO RICO
MAYAGÜEZ CAMPUS

July, 2008

Approved by:

Miguel Vélez Reyes, Ph.D Member, Graduate Committee	Date
Shawn Hunt, Ph.D Member, Graduate Committee	Date
Vidya Manian, Ph.D President, Graduate Committee	Date
Héctor Carlo, Ph.D Representative of Graduate Studies	Date
Isidoro Couvertier Reyes, Ph.D Chairperson of the Department	Date

Abstract of Dissertation Presented to the Graduate School
of the University of Puerto Rico in Partial Fulfillment of the
Requirements for the Degree of Master of Science

HYPERSPECTRAL IMAGE CLASSIFICATION USING SPECTRAL HISTOGRAMS AND SEMI-SUPERVISED LEARNING

By

Sol Marie Cruz Rivera

July 2008

Chair: Vidya Manian
Major Department: Electrical and Computer Engineering

Different classification methods have been applied to hyperspectral images during the last decade. Many of these methods have so far used pixel spectral signatures. Methods that include spatial information in the analysis achieve a better classification accuracy than those that only account for spectral signature of pixels. In this research, an algorithm that extracts regional texture information by computing spectral difference histograms over window extents in hyperspectral images was developed. The spectral angle distance was used as the spectral metric and different window sizes were explored for compute the histogram. The histograms were used in a semi-supervised learning framework that uses both labeled and unlabeled samples for training the Support Vector Machine classifier. Algorithm validation and comparisons are done with real and synthetic hyperspectral images. The method performs well with high spatial resolution images. The algorithm performs well under different Gaussian noise levels.

Resumen de Disertación Presentado a Escuela Graduada
de la Universidad de Puerto Rico como requisito parcial de los
Requerimientos para el grado de Maestría en Ciencias

CLASIFICACIÓN DE IMÁGENES HIPERESPECTRALES UTILIZANDO HISTOGRAMAS ESPECTRALES Y UN ENTRENAMIENTO SEMI-SUPERVISADO

Por

Sol Marie Cruz Rivera

Julio 2008

Consejero: Vidya Manian
Departamento: Ingeniería Eléctrica y Computadoras

Diferentes métodos de clasificación han sido aplicados a imágenes hiperespectrales durante la década pasada. Muchos de estos métodos toman en cuenta la firma espectral de los píxeles. Métodos que incluyen información espacial en su análisis obtienen un mejor rendimiento del clasificador en comparación con aquellos métodos que solo toman en cuenta la firma espectral de los píxeles. En ésta investigación, se desarrolló un algorítmo para imágenes hiperespectrales que extrae información de textura por medio de ventanas mediante el cómputo de histogramas de diferencia espectral. Como métrica espectral se utilizó la distancia de ángulo espectral y se exploraron diferentes tamaños de ventana para el cómputo del histograma. Los histogramas fueron utilizados en un entrenamiento semi-supervisado que utiliza muestras etiquetadas y no etiquetadas para entrenar las máquinas de vectores de soporte ("Support Vector Machines", SVMs por sus siglas en inglés). Validaciones y comparaciones del algorítmo son realizadas con imágenes hiperespectrales reales y sintéticas. El algorítmo produce buenos resultados cuando imágenes hiperespectrales

de alta resolución espacial son utilizadas. El algorítmo fue a su vez validado con data hiperespectral bajo diferentes niveles de ruido Gausiano produciendo también buenos resultados.

To GOD ...

To Héctor, for believing in me ...

To people with physical impediments specially to those with visual problems, there is no barrier that we can not surpass...

ACKNOWLEDGMENTS

Thanks to GOD for giving me the life and the opportunity to meet wonderful people during all these years of study in UPRM. Thanks for giving me the patience and faith that helped me finish this work.

I want to thank my advisor, Dr. Vidya Manian for giving me the opportunity to work with her in this research. Thanks to the members of my committee Dr. Miguel Vélez and Dr. Shawn Hunt. Thanks to the graduate studies representative, Dr. Héctor Carlo for his review and comments on the thesis. Also, thanks to Dr. Ramón Vázquez to went as substitute of Dr. Shawn Hunt. Thanks to all my professors at my graduate studies for sharing with me their knowledge: Dr. Miguel Vélez, Dr. Raúl Torres, Dr. Badrinath Roysam, Dr. Hamed Parsiani, Dr. Shawn Hunt, Dr. Damaris Santana, Dr. Fernando Gilbes and Dr. Domingo Rodríguez.

I want to thank Sandy Montalvo for her friendship, patience and support. Thanks to the staff of LARSIP and CenSSIS: Marie, Samuel, Ricardo, Maribel, Vanessa and Claribel. Special thanks to Samuel for his knowledge, patience and support in those days of work. Thanks to all my partners at LARSIP laboratory, special thanks to Checho and Wilma for their unconditional friendship, and to Santiago for helping me to optimize the semi-supervised SVM code.

Thanks to my family: my father Rafael, my mother Migna, and my sisters Vanesa and Migna Liz, for their unconditional support and for believing in me.

Thanks to my love Héctor for his unconditional love and patience and for encouraging me each day during this time that we were separated.

Thanks to the Center for Subsurface Sensing and Imaging Systems (CenSSIS) Engineering Research Centers Program of the US National Science Foundation that was the primary sponsor of this work under grant EEC-9986821 and partially by DoD under grant W911NF0610008. The research performed here used facilities of CenSSIS at the University of Puerto Rico-Mayagez.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

HSI	Hyperspectral Imagery
HYDICE	Hyperspectral Digital Imagery Collection Experiment
SDH	Spectral Difference Histogram
IFOV	Instantaneous Field Of View
PCA	Principal Component Analysis
SVDSS	Singular Value Decomposition band Subset Selection
SVMs	Support Vector Machines
ORM	Overall Risk Minimization
ICAMM	Independent Component Analysis Mixture Model
ML	Maximum Likelihood
NN	Neural Networks
Reg-RBFNN	Regularized Radial Basis Function Neural Networks
KFD	Kernel Fisher Discriminant
Reg-AB	Regularized AdaBoost
CH	Co-occurrence histogram
OBDFB	Octave-Band Directional Filter Bank
MRF	Markov Random Field
LARSIP	Laboratory for Applied Remote Sensing and Image Processing
HIAT	Hyperspectral Image Analysis Toolbox
SAD	Spectral Angle Distance
FE	Feature Extraction

LIST OF SYMBOLS

m	Meters
nm	nanometers

CHAPTER 1
INTRODUCTION

Hyperspectral imagery (HSI) plays an important role in many remote sensing applications. They are characterized by large amounts of data taken at narrow and contiguous spectral bands [1],[2] providing us important information about the spectral characteristics of the materials that are present in the scene. HSI sensors have high spectral resolution and some of them also have high spatial resolution. Both, the spectral and spatial information are important in image analysis; while the spectral resolution helps us to discriminate between different materials, the spatial resolution measure the spatial detail in an image. A high quality, portable and easy-to-use spectral imaging system is the SOC-700 hyperspectral Imager. It has a spectral resolution of 4nm with 120 spectral bands in a range of 400 to 900 nm. Another example of a sensor with high spectral and spatial resolution is the Hyperspectral Data Imagery Collection Experiment (HYDICE), which captures the information in 210 contiguous bandwidths from the visible to shortwave infrared (400-2500 nm) with a spatial resolution that varies between 1 to 4 meters depending on the aircraft's altitude above ground level [2], [3]. Table 1–1 summarizes the capabilities of the sensors mentioned above.

Recently, interest in HSI has increased. This type of data is frequently used in land cover classification, detection and target recognition, search and rescue operations, and also in biomedical applications such as cancer diagnosis. Therefore, it is

Table 1–1: HSI Sensor Characteristics

Sensor	Spectral Resolution	Spatial Resolution	No. of Bands	Spectral Range
SOC-700	4nm	high[a]	120	400-900nm
HYDICE	10nm	1-4m	210	400-2500nm

[a] Depends on the IFOV of the camera and on the distance from the ground to the sensor (H). The IFOV of the SOC-700 camera is 0.0078125. $spatialRes = IFOV \times H$

very important that the accuracy of the classification methods used to classify and analyze this type of data be as high as possible.

1.1 Problem Statement

Different classification methods have been applied to hyperspectral images during the last decade. Many of these methods have so far used pixel spectral signatures (see Figure 1–1). With the increasing spatial resolution of sensors and cameras, pixel based methods perform poorly. Methods that include spatial information in the analysis achieve a better classification accuracy than those that only accounts the spectral signature of pixels [1], [4], [5], [10] . Hence, spatial information has to be exploited to improve the performance of the classifier.

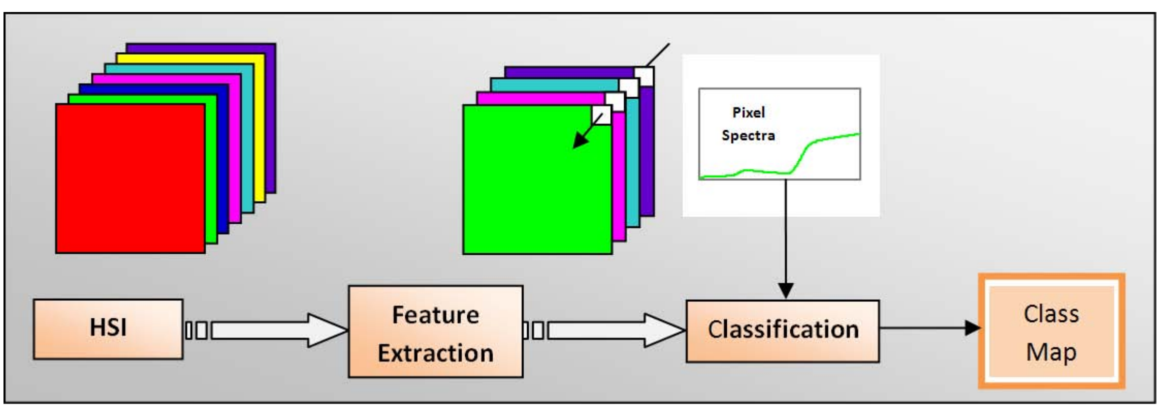


Figure 1–1: Block diagram of a general pixel-based classification.

The integration of spatial and spectral information in hyperspectral image analysis has been identified as a highly desirable objective by the remote sensing community. In this thesis, joint distribution of pixels rather than single pixel distribution is used to develop an algorithm for spatial-spectral feature extraction. This algorithm will extract regional texture information by computing spectral difference histograms over window extents. A general block diagram of the proposed Spectral Difference Histogram (SDH) based classification is shown in Figure 1–2.

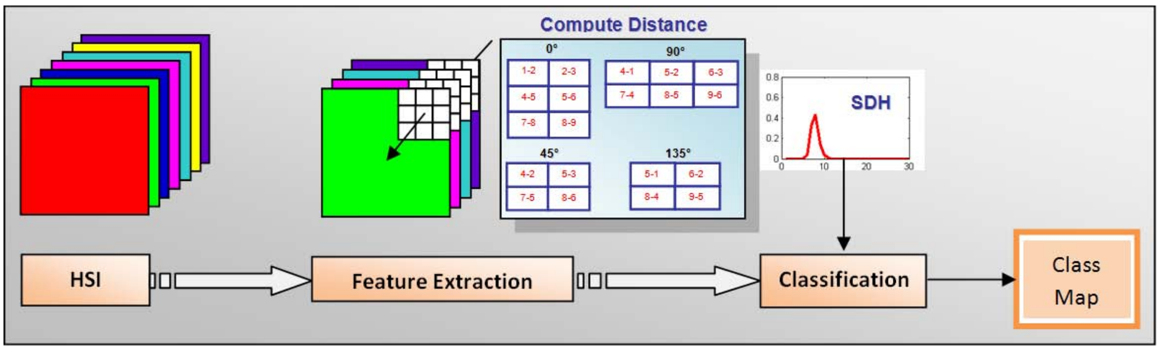


Figure 1–2: Block diagram of the SDH-based classification.

1.2 Objectives

The main objective of this work is to develop an algorithm for spatial-spectral feature extraction for hyperspectral image classification. The specific objectives of this research are as follows:

- Develop a joint pixel histogram method integrating spatial and spectral information for hyperspectral image classification based on spectral difference histograms computed over window extents.
- Apply the method using a semi-supervised learning algorithm for classification.

- Experiment with different parameters such as spatial extents (window sizes) for improving classification.
- Validate the algorithm with real and synthetic hyperspectral datasets with different noise levels.

1.3 Contributions of this work

The most important contribution of this algorithm will be better integration of spatial and spectral information for feature extraction and classification of hyperspectral images.

1.4 Thesis Outline

The thesis is organized as follows: Chapter 2 provides an overview of remote sensing and different classification methods. Chapter 3 describes the spectral difference histogram algorithm. Chapter 4 presents different experiments and the obtained results . Finally, the conclusions and recommendations for future work are presented in Chapter 5.

CHAPTER 2
THEORETICAL BACKGROUND AND LITERATURE REVIEW

This chapter presents an overview of remote sensing and hyperspectral images. Different image classification methods are discussed. A literature review of different feature extraction and classification methods that currently exists is presented.

2.1 Remote Sensing

The term "remote sensing" is commonly used to describe the science and art of identifying, observing and measuring an object without coming into direct contact with it [6]. It was first used in the early 1960s in the United States by Ms. Evelyn Pruitt of the U.S. Office of Naval Research. It is dated from 1858 when Gaspard Felix Tournachon took the first-known aerial photograph from a balloon near Paris, France [6]. In the following years, other platforms such as kites, rockets and pigeons were experimented, but the great step forward was the invention of the airplane, a much stable and reliable platform. Finally, remote sensing moved to outer space with the invention of satellites.

2.1.1 Data Collection

Remote sensing instruments measured reflected and/or emitted electromagnetic radiation using aerial and satellite platforms. Figure 2–1 shows the electromagnetic spectrum. Remotely sensed data are collected using passive or active remote sensing systems [6]. Passive sensors record electromagnetic radiation that is reflected or emitted from the surface of Earth. On the other hand, active sensors are independent of the Sun's electromagnetic energy or the thermal properties of the Earth. They measure the backscatter energy that is produced by the interaction of the electromagnetic energy and the terrain.

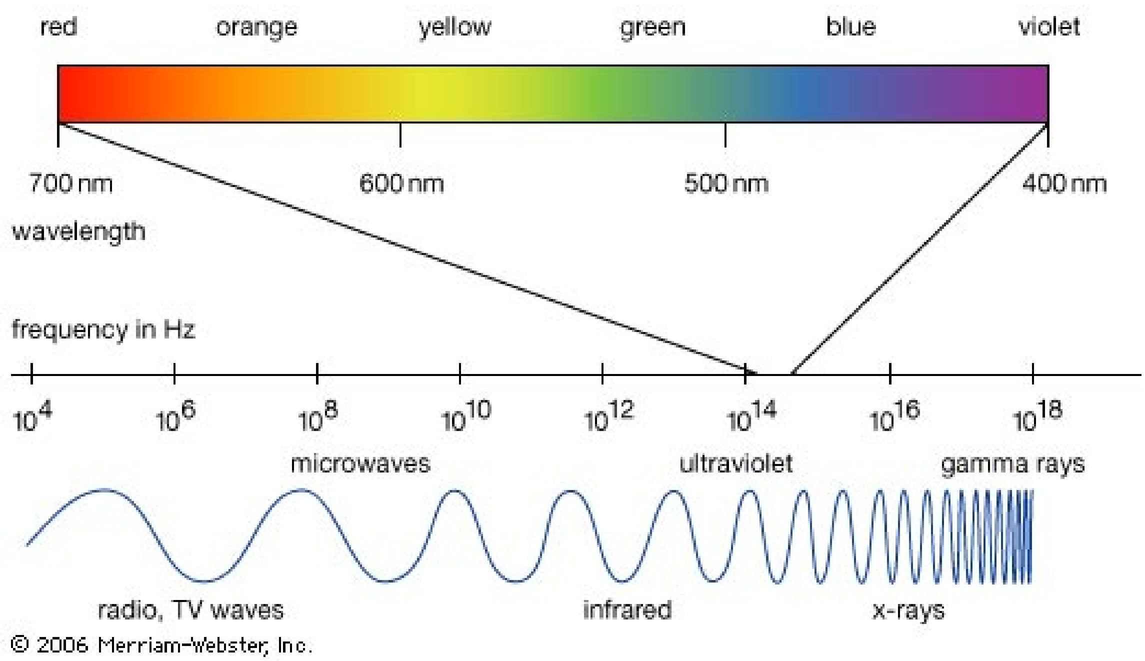


Figure 2–1: The electromagnetic spectrum.

The remote sensing process involves an interaction between incident radiation and the target of interest. It involves the following seven elements (see Figure 2–2):

- (A) Energy source or illumination
- (B) Radiation and the atmosphere

- (C) Interaction with the target
- (D) Recording of energy by the sensor
- (E) Transmission, reception and processing
- (F) Interpretation and analysis
- (G) Application

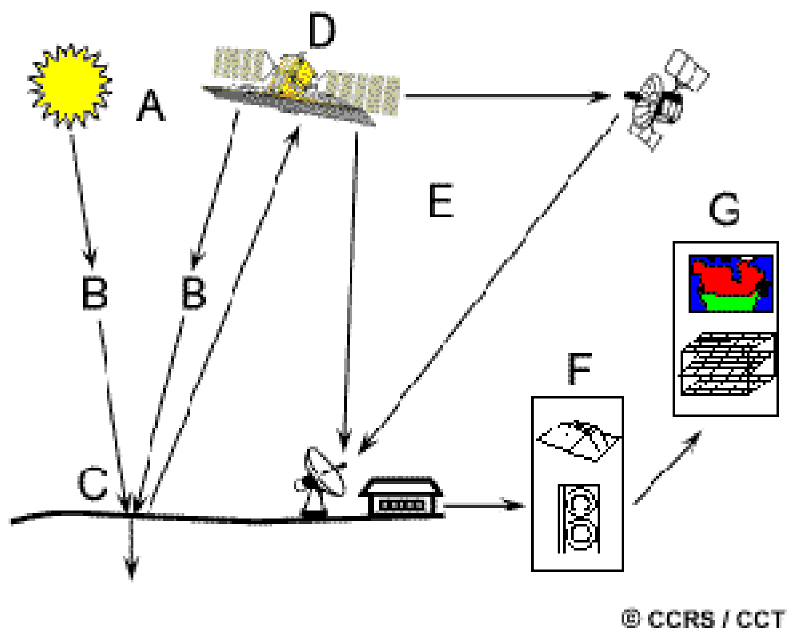


Figure 2–2: The Remote Sensing process [8].

2.1.2 Spectral Resolution

Spectral resolution is defined as the number and dimension (size) of specific wavelength intervals, referred to as bands or channels, in the electromagnetic spectrum to which a remote sensing instrument is sensitive [6]. Data generated by sensors may consist of one spectral band (panchromatic image), few spectral bands (multispectral images) or many narrow and contiguous spectral bands (hyperspectral images) [7]. Spectral resolution provides us important information about the spectral characteristics of the materials that are present in the scene. Different materials can be discriminated by means of its spectral signature (see Figure 2–5). An example of a hyperspectral data cube is shown in Figure 2–3 to understand the concept of spectral resolution.

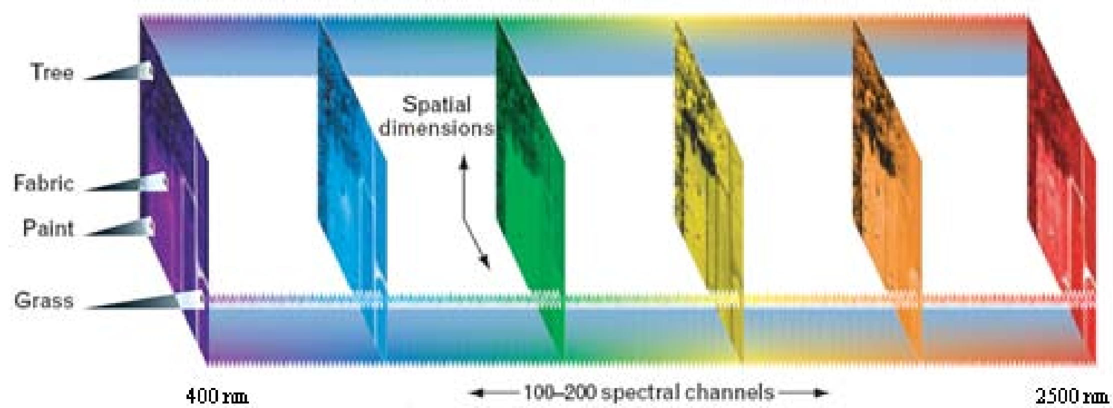


Figure 2–3: A three-dimensional data cube. It can be treated as a stack of two-dimensional spatial images, each corresponding to a particular narrow spectral band. Image Courtesy of [9].

2.1.3 Spatial Resolution

Spatial resolution is a measure of the spatial detail in an image, which is a function of the design of the sensor and its operating altitude above the surface [7], [8]. In other words, how much of the earth's surface a single pixel covers (see Figure 2–4(a) and (b)). High spatial resolution provides more information about the area under study as shown in Figure 2–4(b). Spatial resolution of passive sensors depends primarily on their instantaneous field of view (IFOV) as shown in Figure 2–4(c) [8]. The IFOV is the angular cone of visibility of the sensor (B) and determines the area on the Earth's surface which is seen from a given altitude at one particular moment in time. The size of the area viewed is determined by multiplying the IFOV by the distance from the ground to the sensor (H) [6], [8].

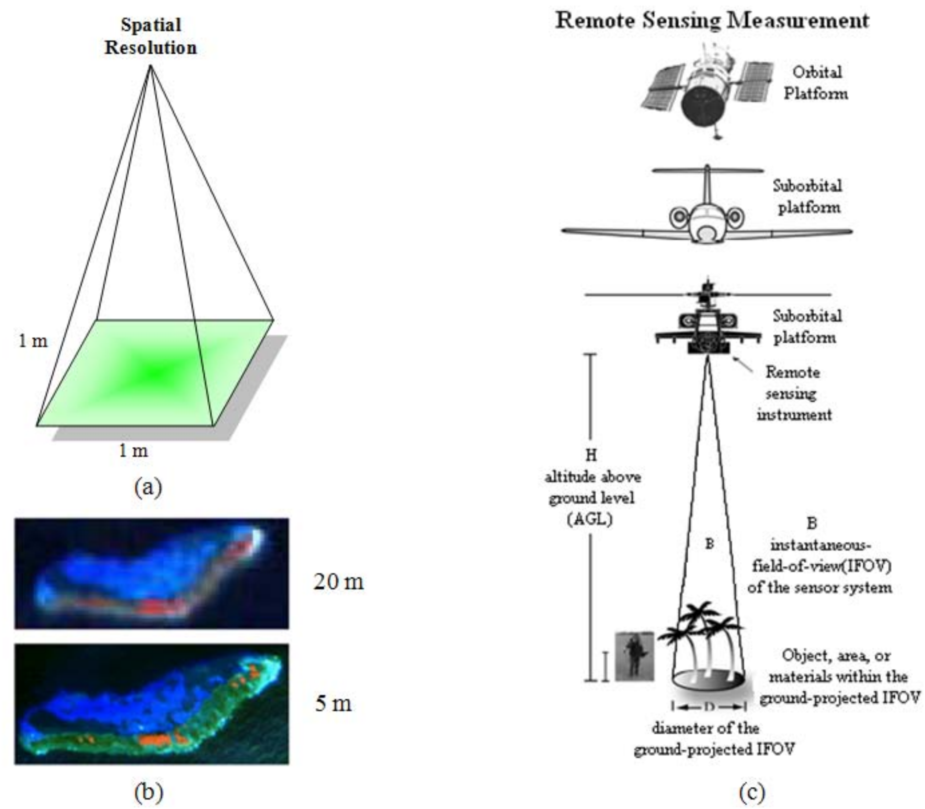


Figure 2–4: (a) Spatial resolution - the size of the field of view, e.g. $1m \times 1m$. (b) An example of different spatial resolution images taken by SPOT (upper) and CASI (lower) sensors of Enrique Reef at Lajas, PR. (c) A remote sensing measurement. The remote sensing instrument collects information of an object within the IFOV of the sensor system without coming into direct contact with it (Figure reprinted from [6]).

2.2 Hyperspectral Imagery (HSI)

Hyperspectral images are characterized by large amounts of data taken at narrow and contiguous spectral bands [1], [2]. This type of data is frequently used in land cover classification, detection and target recognition, search and rescue operations, and also in biomedical applications such as cancer diagnosis [1], [2]. Recently, interest in hyperspectral images has increased. The reason is principally the detailed information provided by the high spectral resolution which helps to discriminate better between different objects in an image. To see this contribution, Figure 2–5 illustrates the concept of HSI. It illustrates the spectral variation of three different components: Soil, Water and Vegetation. With the spectra of each one we can discriminate between them.

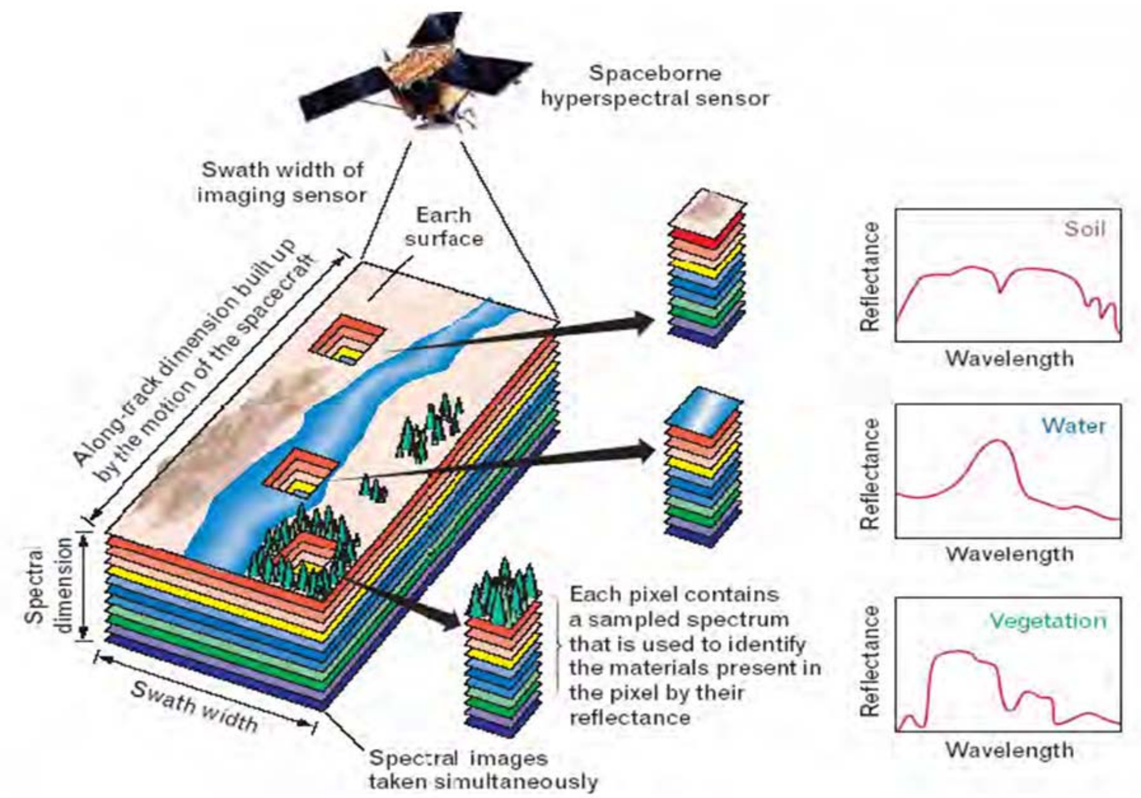


Figure 2–5: The hyperspectral imaging concept. It illustrates the spectral variation of three different components: Soil, Water and Vegetation. Image Courtesy of [9].

Due to the high dimensionality of the HSI, methods for reducing the image dimensionality are often applied. This reduction must be done in such a way that the data redundancy is minimized without losing relevant information about objects of interest.

2.2.1 Dimensionality Reduction

Dimensionality reduction is the general problem of projecting a data set into a lower dimensional space. One of the most widely used dimension reduction methods in remote sensing is the principal component analysis (PCA) [10]. Vélez et. al [11] proposed an unsupervised mechanism of selecting spectral bands based on Singular Value Decomposition (SVDSS) that approximates PCA dimensionality reduction [12]. Band subset selection, and discriminant and independent component analysis methods are also used to reduce the dimensionality of the data before the classification process [1], [10], [13]. PCA and SVDSS methods are discussed next.

2.2.1.1 PCA

PCA uses a linear transformation , the principal-component transform ($\mathbf{y} = \mathbf{A}\mathbf{x}$) [14], to rotate and translate multiband spectral data into a new set of coordinate system [15]. It is used to decorrelate data and maximize the information content in a reduced number of features [15]. The covariance matrix is first computed over the pixel spectra contained in the hyperspectral data cube of interest. Eigenvalues and eigenvectors are then obtained for the covariance matrix [15]:

$$\Sigma_x = E\left\{(\mathbf{x} - \mu_x)(\mathbf{x} - \mu_x)^T\right\} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{\mathbf{T}} \tag{2.1}$$

where $\mathbf{x}$ is the spectral vector data , μ_x is the mean spectral vector over the data cube, $\mathbf{V}$ is a matrix consisting of columns of eigenvectors and $\mathbf{\Lambda}$ is a diagonal matrix of eigenvalues.

We use the eigenvectors as a new coordinate system to transform the hyperspectral data cube into principal components [15]. If the transformed spectral data $\mathbf{x}$ is represented as $\mathbf{y}$ in the new coordinate system, then the principal-component transformation is a linear transformation $\mathbf{V}^T$ of the original coordinates such that

$$\mathbf{y} = \mathbf{V^T x} \tag{2.2}$$

In y space the covariance matrix is given as

$$\Sigma_y = E\left\{(\mathbf{y} - \mu_y)(\mathbf{y} - \mu_y)^T\right\} \tag{2.3}$$

where μ_y is the mean vector expressed in terms of the y coordinates [15]. A property of the transform is that

$$\mu_y = \mathbf{V}^T \mu_x \tag{2.4}$$

As a result, Σ_y is the diagonal matrix of eigenvalues of Σ_x such that

$$\begin{aligned} \Sigma_y &= E\left\{(\mathbf{V}^T\mathbf{x} - \mathbf{V^T}\mu_{\mathbf{x}})(\mathbf{V^T x} - \mathbf{V^T}\mu_{\mathbf{x}})^{\mathbf{T}}\right\} \\ &= \mathbf{V}^T E\left\{(\mathbf{x} - \mu_x)(\mathbf{x} - \mu_x)^T\right\}\mathbf{V} \\ &= \mathbf{\Lambda} \end{aligned} \tag{2.5}$$

Because Σ_y is a covariance matrix and is diagonal, its elements represents the variances of a set of orthogonal images in the transformed coordinate space [15]. The eigenvectors are arranged in descending order of the eigenvalues so that the data exhibit maximum variance in the first component, and so on, with the minimum variance in the last component. In other words, the new set of uncorrelated images are ordered in terms of decreasing information or equivalently decreasing variance [16].

In some circumstances the projections produced by this method are accepted due to the fact that some classes are distributed along the largest eigenvector which provides the direction that maximizes the data variance. In this case, PCA finds good class separability in its projections as shown in Figure 2–6(a). Unfortunately,

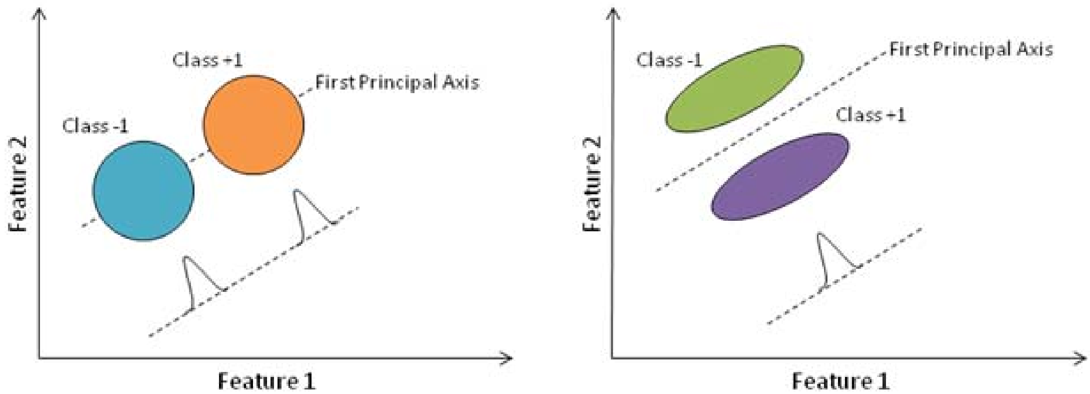


Figure 2–6: First Principal Axis of Class -1 and Class +1. (a) Optimal Separability Projection. (b) Poor Separability Projection.

information content in hyperspectral images does not always match these projections [10], [17]. PCA obtains a poor separability projection when the classes are not distributed along the largest eigenvector or the first principal axis as shown in Figure 2–6(b). Also it does not work properly in the detection and classification of small size objects relative to the scene. Methods that perform a dimensionality reduction of HSI using wavelets, yields a better or comparable classification accuracy compared to PCA, and also reduce the computational time requirements [17].

2.2.1.2 SVDSS

Vélez et.al [11] proposed an unsupervided mechanism for selecting spectral bands based on SVDSS that approximates PCA. The subset selection problem restricts the projection matrix **A** as follows:

$$\mathbf{A} = \mathbf{P}\begin{bmatrix} \mathbf{I_p} \\ \mathbf{0} \end{bmatrix} \tag{2.6}$$

where $\mathbf{P}$ is a permutation matrix[1] . The net effect of this constraint is that the dimension-reducing transformation $\mathbf{A}$ now selects a subset of the original variables $\mathbf{x}$ as follows [11], [12]:

$$\mathbf{y} = \mathbf{A^T x} = \left[\begin{array}{cc} \mathbf{I_p} & \mathbf{0} \end{array} \right] \mathbf{P^T x} = \begin{bmatrix} x_{i_1} \\ x_{i_2} \\ \vdots \\ x_{i_p} \end{bmatrix} \tag{2.7}$$

An important advantage of the selection of a subset of bands is the retention of the physical meaning of the data, that is, there are no data transformations [12]. The SVDSS algorithm is summarized in the following steps[2] :

1. Compute the covariance matrix Σ_{data} of the HSI.
2. Compute the QR factorization with pivoting of the matrix $\mathbf{V}_1^T$, where $\mathbf{V}_1$ is formed by the first p eigenvectors of Σ_{data}.
3. Compute $\bar{\mathbf{x}} = \mathbf{Px}$.
4. Take the first p elements of $\bar{\mathbf{x}}$ as the selected bands.

2.3 Image Classification

Image classification is the process of assigning all pixels in a digital image to particular classes according to their characteristics [8]. As a result we obtained a thematic map in which each pixel belongs to a particular class. Two main classification schemes are the Unsupervised and Supervised Classification. A halfway between the

[1] A matrix $\mathbf{P}$ is called a permutation matrix if exactly one entry in each row and column is equal to 1, and all other entries are 0. Multiplication by such matrices results in a permutation of the rows or columns of the object multiplied [11].

[2] Refer to [11].

unsupervised and supervised learning is known as Semi-Supervised Learning.

2.3.1 Unsupervised Classification

Unsupervised classification can be defined as the identification of natural groups or structures within the data. It clusters pixels in a data set based only on their statistics without using previous knowledge about the spectral classes present in the image. Some of the mostly used unsupervised classification methods are: Isodata and k-Means [18].

2.3.2 Supervised Classification

Supervised classification can be defined as the process of using samples of known identity (training data) to classify pixels of unknown identity. The training data are used to train the classifier which is tested with testing samples to evaluate the accuracy of the classifier. Some of the most commonly used supervised classification methods are: Maximum Likelihood, Minimum Distance, Mahalanobis Distance, and Neural Networks. Recently, support vector machines (SVMs) have been successfully used for hyperspectral data classification [13].

2.3.2.1 SVMs

SVM tries to find an optimal separating hyperplane that discriminate between two classes of interest. On each side of the hyperplane that separates the data, two parallel hyperplanes that passes through at least one vector of the two classes are constructed. These vectors are known as support vectors. The hyperplane that maximizes the distance between these two hyperplanes is called the optimal separating hyperplane [19]. A special property of SVMs is that they minimize the empirical classification error and maximize the geometric margin.

2.3.2.2 SVM - the linearly separable case

Consider first the case of a binary classification problem where we have a linearly separable training set Z. Assume that the training dataset consists of n training samples characterized by Z $= \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_n, y_n)\}$, where $\mathbf{x} \in R^N$ is a N-dimensional data vector, and $y_i \in \{-1, +1\}, 1 \leq i \leq n$, is a constant that denotes the class to which the sample $\mathbf{x}_i$ belongs. The objective of SVMs is to find a linear decision function defined by $f(\mathbf{x}) = sign(\mathbf{w^T} \cdot \mathbf{x} + b)$, where $\mathbf{w} \in R^N$ determines the orientation of the optimal hyperplane and $b \in R$ is a bias [20]. There can be infinitely many linear classifiers that separate the training set without errors and, consequently our task is to choose the best one [21].

If the the samples in the training set Z are linearly separable, then there exists a N-vector $\mathbf{w}$ and a scalar b such that:

$$\begin{array}{lll} \mathbf{w^T} \cdot \mathbf{x}_i + b \geq +1 & \text{if } y_i = +1, \text{ and} & \\ & & \text{for } 1 \leq i \leq n \\ \mathbf{w^T} \cdot \mathbf{x}_i + b \geq -1 & \text{if } y_i = -1, & \end{array} \tag{2.8}$$

These can be combined into one set of inequalities, so the hyperplanes for the two classes are represented by:

$$y_i[\mathbf{w^T} \cdot \mathbf{x_i} + b] \geq +1 \tag{2.9}$$

Given that the data is linearly separable, the two hyperplanes of the margin can be selected in a way that there are no samples between them. The optimal separating hyperplane, $\mathbf{w^T} \cdot \mathbf{x} + b = 0$, is located where the margin between the two classes is maximized. Margin can be defined as the sum of the smallest distance from the separating hyperplane to the closest sample of class +1 and the smallest distance from the separating hyperplane to the closest sample of class -1. The euclidean distance between these two hyperplanes is $\frac{2}{\|\mathbf{w}\|}$, where $\|\mathbf{w}\|$ is the Euclidean norm of $\mathbf{w}$. So maximizing the margin reduces now to minimizing the norm of the weigth vector $\mathbf{w}$ subject to the constraints in Equation (2.9). The optimal separating hyperplane can be obtained by solving the following constrained optimization problem [19], [21]:

$$\begin{gathered} min_{\mathbf{w},b}\left\{\frac{1}{2}\|\mathbf{w}\|^2\right\} \\ s.t.\ y_i[\mathbf{w^T}\cdot\mathbf{x_i}+b]\geq 1,\ \text{for } 1\leq i\leq n \end{gathered} \tag{2.10}$$

The vector $\mathbf{w}$ is perpendicular to the separating hyperplane. The perpendicular distance from the separating hyperplane to the origin can be expressed as $\frac{b}{\|\mathbf{w}\|}$. Figure 2–7(a) shows the optimal hyperplane for a two class separable case.

2.3.2.3 SVM - the linearly non-separable case

In general, the classes will not be separable. Figure 2–7(b) shows the optimal hyperplane for a two class non-separable case. In this case, we assume that one cannot separate the data without a misclassification error using the class of linear classifiers. Now, it is necessary to introduce slack variables, $\xi_i > 0$, to reduce the weighting of the misclassified vectors. The hyperplanes for the two classes become:

$$y_i[\mathbf{w^T}\cdot\mathbf{x_i}+b]+\xi_i\geq 1,\ for 1\leq i\leq n \tag{2.11}$$

Equation (2.10) now transforms to Equation (2.12), where $C > 0$ is a fixed penalty parameter [22].

$$\begin{gathered} min_{\mathbf{w},b,\xi}\left\{\frac{1}{2}\|\mathbf{w}\|^2+C\sum_{i=1}^{n}\xi_i\right\} \\ s.t.\ y_i[\mathbf{w^T}\cdot\mathbf{x_i}-b]\geq 1-\xi_i,\ \text{for } \xi_i>0,\ 1\leq i\leq n \end{gathered} \tag{2.12}$$

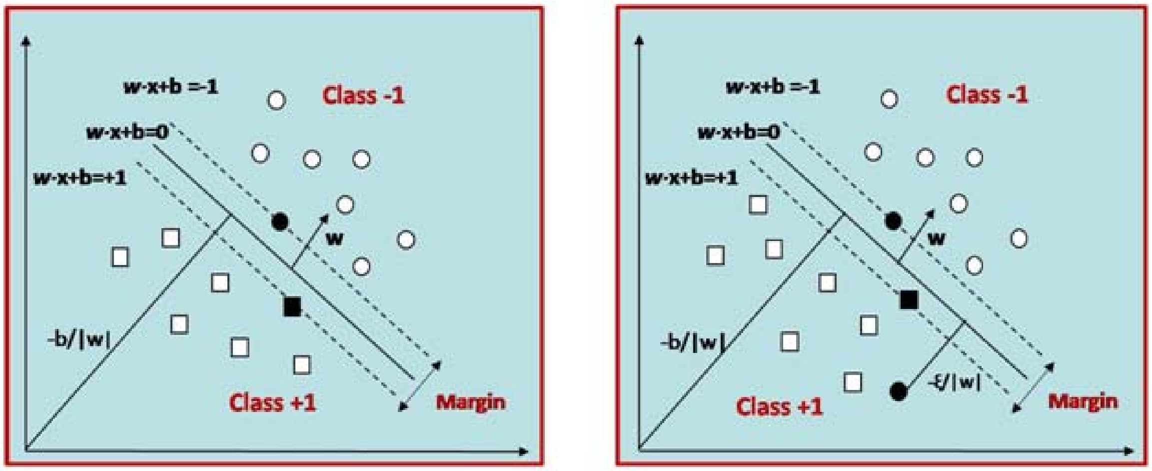


Figure 2–7: Linear optimal separating hyperplane for (a) the two class separable case, (b) the two class non-separable case. The support vectors are the colored samples.

2.3.2.4 The SVM for multiclass classification

In general, in real-life situations it is often necessary to separate more than two classes at the same time [21]. The simplest extension of the SVM to a k-class problem is to separate the observations from class c from the rest of $c = 1, \cdots, k$ classes. Here the "rest" means that all the observations from other classes than c are combined to form one class. The optimal separating hyperplane that discriminate the class c and the combined class is denoted by [21]:

$$\mathbf{x}^T \mathbf{w}^c + b^c, \; c = 1, \cdots, k$$

where the superscript c stands for the class which would be separated from the other observations. The decision rule f^c that assigns the vector $\mathbf{x}$ to the class c or to the combined class is [21]:

$$f^c(\mathbf{x}) = sgn\big(g^c(\mathbf{x})\big), \tag{2.13}$$

where $g^c(\mathbf{x}) = \mathbf{x}^T \mathbf{w}^c + b^c$. After all the k optimal separating hyperplanes have been found the final classifier $\mathbf{f_k}$ is [21]:

$$\mathbf{f}_k(\mathbf{x}) = argmax_c\big(f^c(\mathbf{x})\big). \tag{2.14}$$

2.3.3 Semi-Supervised Classification

Semi-supervised learning is halfway between supervised and unsupervised learning [23]. It makes use of both labeled and unlabeled data for training the classifier. Typically a small amount of labeled data and a large amount of unlabeled data are used. Semi-supervised algorithms that learn from both label and unlabeled samples have been used in the last few years. In [21], [24] this approach of considering also the test dataset is called overall risk minimization (ORM). The approach where the ORM principle is applied to the SVM methodology is called in [22] the semi-supervised SVM.

2.3.3.1 Semi-Supervised SVM

The semi-supervised SVM selects the optimal hyperplane that separates the input data based not only on the labeled samples but also in the unlabeled samples. As in Section 2.3.2.2, here we are considering the binary classification case. But now in addition to the labeled set Z, we need to include the unlabeled dataset. Suppose that we are given a labeled dataset Z (defined as in Section 2.3.2.2) and a unlabeled dataset X^* that consists of m unlabeled samples characterized by $X^* = \{\mathbf{x}_1^*, \mathbf{x}_2^*, \cdots, \mathbf{x}_m^*\}$ where $\mathbf{x}^* \in R^N$ is a N-dimensional data vector corresponding to the unlabeled samples.

In the semi-supervised SVM case the constrained optimization problem for the linearly separable case is:

$$\begin{gathered} min_{\mathbf{w},b}\left\{\frac{1}{2}\|\mathbf{w}\|^2\right\} \\ s.t.\ y_i[\mathbf{w^T}\cdot\mathbf{x_i}-b]\geq 1,\ \text{for } 1\leq i\leq n \\ y_j^*[\mathbf{w^T}\cdot\mathbf{x_j^*}-b]\geq 1,\ \text{for } 1\leq j\leq m \end{gathered} \tag{2.15}$$

where y_j^* is the unknown label of x_j^*.

In the linearly non-separable case an additional term is added to Equation (2.12) that drives the outputs of the unlabeled samples away from zero. Therefore, the optimization problem for a linearly non-separable data in the semi-supervised SVM is given by: [26],[27]

$$\begin{aligned} &min_{\mathbf{w},b,\xi,\xi^*}\left\{\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i + C^*\sum_{j=1}^{m}\xi_j{}^*\right\} \\ &s.t.\ y_i[\mathbf{w^T}\cdot\mathbf{x_i} - b] \geq 1-\xi_i,\ 1\leq i\leq n \\ &\quad y_j^*[\mathbf{w^T}\cdot\mathbf{x_j^*} - b] \geq 1-\xi_j^*,\ 1\leq j\leq m \end{aligned} \tag{2.16}$$

Equation (2.16) can be rewritten without the constraint as: [27]

$$min_{\mathbf{w},b,\xi,\xi^*}\left\{\frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{n}L\big(y_i[\mathbf{w^T}\cdot\mathbf{x_i} - b]\big) + C^*\sum_{j=1}^{m}L^*\big(y_j^*[\mathbf{w^T}\cdot\mathbf{x_j^*} - b]\big) \tag{2.17}$$

with $L(t) = max(0, 1-t)$. As it is explained in [26], the last term of equation 2.17 makes the problem non-convex and difficult to solve.

2.4 Literature Review

A literature review of different classification methods is presented in this section. Spectral and spatial based classification methods are considered.

2.4.1 Spectral based Classification

Several unsupervised and supervised algorithms have been developed for classification of multispectral images. Recently, support vector machines (SVMs) have been successfully used for hyperspectral data classification [13]. Independent Component Analysis mixture model (ICAMM) and SVMs algorithms have been developed explicitly in [20] for classifying HSI. Shah et. al. [20], showed that the SVM is a superior classification algorithm for classifying HSI than other two widely used supervised classifiers: Maximum Likelihood (ML) and back propagation Neural Networks (NN). Also the ICAMM classification algorithm produced significantly higher accuracy as compared to the K-means unsupervised algorithm.

Recently there has been an interest in kernel-based methods. A performance assessment between SVMs, regularized radial basis function neural networks (Reg-RBFNN), kernel Fisher discriminant (KFD) and regularized AdaBoost (Reg-AB) is presented in [13]. Camps-Valls et. al. [13], showed that SVMs yield better results than the other kernel-based methods. A semi-supervised graph-based method for the classification of hyperspectral images is presented in [28]. The proposed method was compared with the SVM method. It was shown that the semi-supervised graph-based method produces better classification maps compared to those obtained with SVM method. Another semi-supervised method was introduced in [22]. A comparison between semi-supervised SVM method and the traditional SVM approach was done. In every case, semi-supervised SVM either improved or showed no significant difference compared to standard approach.

The methods discussed above make use only of the spectral information of each pixel. Methods that use spatial information in the analysis are discussed below.

2.4.2 Spatial based Classification

Methods that include spatial information in the analysis achieve a better classification accuracy than those that accounts for only the spectral signature of pixels [1], [4], [5].

2.4.2.1 Spatial Histograms

Textures are sometimes described using co-occurrences matrices. They are usually derived from gray scale images but also from color images [29], [30]. The co-occurrence matrix is a square matrix with elements corresponding to the relative frequency of pairs of gray level pixels separated by a certain distance and for a given direction. Figure 2–8 shows an example of how the co-occurrence matrix is computed.

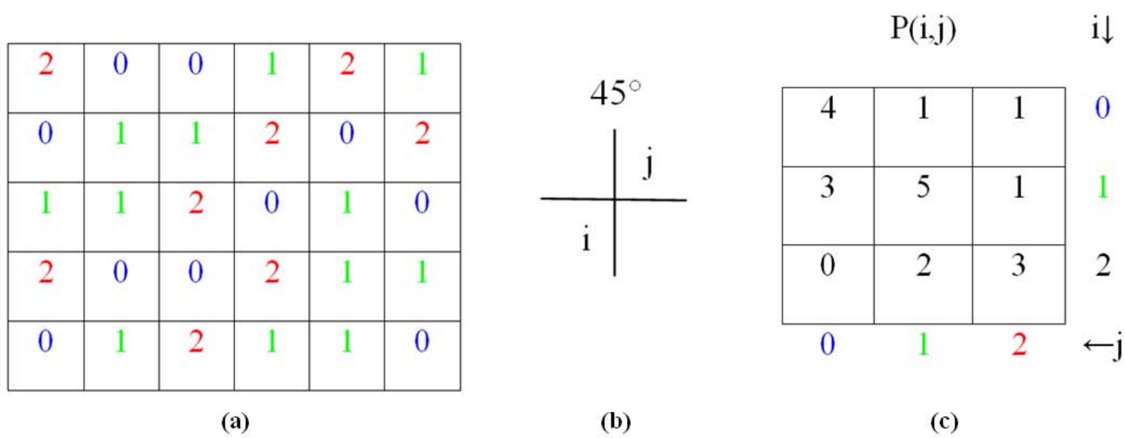


Figure 2–8: (a) Example of a digital image. (b) An example of a direction used to create the co-occurrence matrix. (c) Co-occurrence matrix using 45° direction.

A color co-occurrence histogram (CH) used for object recognition is presented in [29]. Color CH holds the number of occurrences of pairs of certain color pixels that occur in certain separation distances in image space. Color CHs are also used in [30] for object detection. A reduced multidimensional CH method in texture classification is presented in [31]. An efficient way of generating a co-occurrence matrix from irregularly shaped tessellation elements was presented in [32]. These approaches give good results for gray scale images, but the adaptation to hyperspectral images is computationally expensive.

2.4.2.2 Other Spatial Methods

Texture feature methods found in the literature are efficiently used for hyperspectral and multispectral image classification. A land cover and benthic habitat classification using texture features from hyperspectral and multispectral images are presented in [1]. Manian and Jimenez, demonstrated the efficacy of using spatial features for classification when compared to using only spectral features. For the classification they used the minimum distance classifier using texture features selected from a feature selection process, and the maximum likelihood classifier using the spectral features. For all the experiments performed, they obtain a better overall performance when using texture features. In conclusion, texture features provides a high discriminatory power and provide greater separability between classes.

A method for appending texture information to existing hyperspectral data to increase classification accuracy is presented in [5]. The octave-band directional filter bank (OBDFB) was used to extract texture features. Texture features are local energy estimates of the coefficients of the sub-bands of the decomposition feature vectors. They used ML classifier. It can be shown that the classification accuracy generally increased when texture feature information are taken into account. Hyperspectral texture characterization is studied in [33] by extending the wavelet transform to HSI.

A probabilistic vector texture model using a Gauss-Markov random field (MRF) which takes into account texture from a spatial and spectral point of view is proposed in [4]. The application of their work is in classification of urban areas. The covariance matrix parameter was their texture feature. The texture feature extraction algorithm used is based on the Bhattacharya distance between the distributions of the training samples. The classification criterion is the ML using an approximation of the probability distribution of the texture features. The proposed method yield better results than the other tested methods.

A morphological method that is based on making use of both spectral and spatial information for classification of HSI was proposed in [34]. They were investigating hyperspectral data with high spatial resolution from urban areas. They used a NN classifier with and without feature extraction, and the results were compared with those of the Gaussian ML classifier. The proposed method performed well in terms of accuracy and was comparable in accuracy to the ML classifier, especially when decision boundary feature extraction was applied on the extended morphological profile [34]. Plaza et al. presented in [35], an automated method based on mathematical morphology that performs unsupervised pixel purity determination and endmember extraction from multi/hyper-dimensional datasets. The idea of using endmembers derived from the data for classification and unmixing has been considered before, but few methods have exploited the spatial information existing between neighboring pixels. The method uses spectral and spatial information simultaneously. Results are comparable to those obtained with other methodologies.

The integration of spatial and spectral information in hyperspectral image analysis has been identified as a highly desirable objective by the remote sensing community. Methods that explored the relationship between neighbor pixel vectors will be developed in our work. This work focuses on using pixels relationship at different distances/angles.

2.5 Conclusions

In this chapter important information about HSI commencing from data collection, the importance of the spectral and spatial information, to application of different classification methods were summarized. The SVMs based on for the supervised and semi-supervised learning was explained. Spectral and spatial based classification methods were presented.

CHAPTER 3
SPATIAL-SPECTRAL FEATURE EXTRACTION

The integration of spatial and spectral information in hyperspectral image analysis has been identified as a highly desirable objective by the remote sensing community. Joint distribution of pixels rather than single pixel distribution is used to develop an algorithm for spatial-spectral feature extraction. The spectral difference histogram algorithm extracts regional texture information by computing spectral difference histograms over window extents. The methods that we used to develop the algorithm will be discussed in this chapter. Figure 3–1 shows a diagram that presents the classification methodology to be used.

3.1 Hyperspectral data

Real and synthetic hyperspectral datasets with different noise levels are used to test the algorithm. The datasets are obtained from (1) the SOC-700 hyperspectral camera, and (2) the Hyperspectral Digital Imagery Collection Experiment (HYDICE).

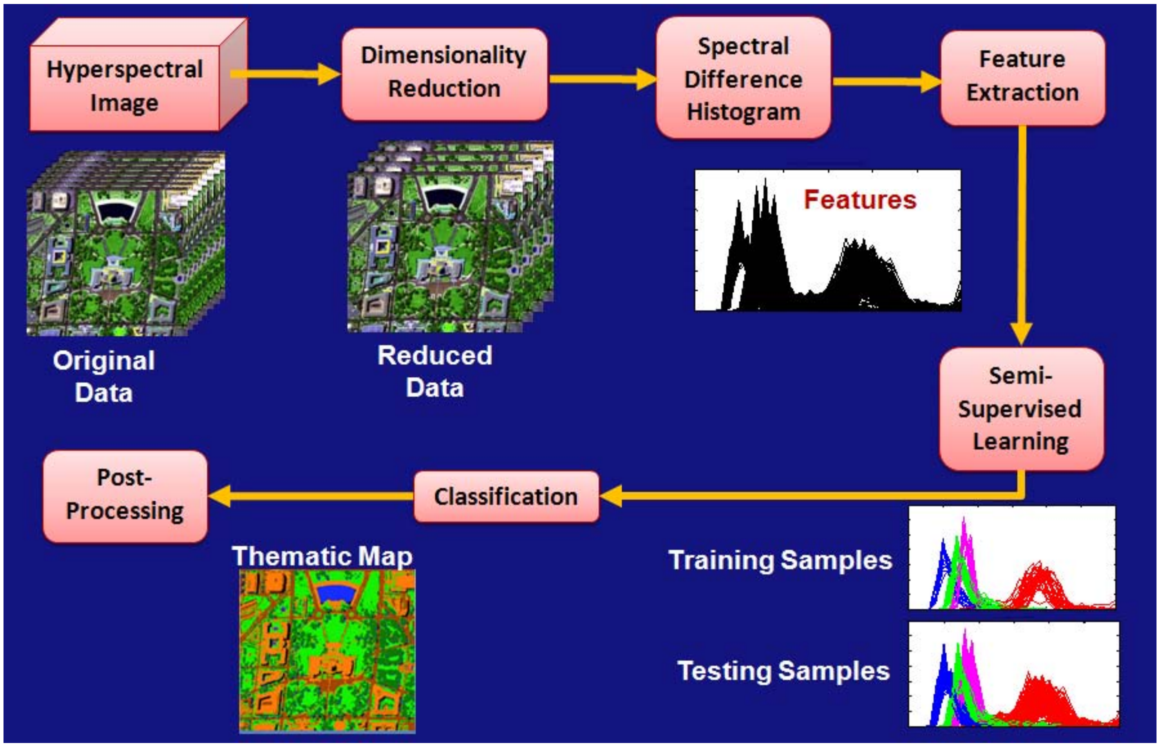


Figure 3–1: Block diagram for the classification methodology.

3.1.1 SOC-700 hyperspectral camera

Hyperspectral images of different textures are collected using the SOC-700 hyperspectral camera. This camera has a spectral resolution of 4 nm with 120 bands and a spectral range from 400 to 900 nm. It is available at the Laboratory for Applied Remote Sensing and Image Processing (LARSIP). Figure 3–2 shows the SOC-700 hyperspectral camera used to take the images. Mosaics of the different textures were made. The main reason why we used different hyperspectral textures to validate the algorithm is because they are used in literature [4],[36] to get an idea of the quality of the results. Different mosaics with different textures to validate the results of the algorithm have been used as they have precise ground truth of the data.

Figure 3–2: SOC-700 hyperspectral camera.

3.1.2 HYDICE sensor

The HYDICE sensor is a second generation, “state-of-the-art”, nadir-viewing, push broom, high resolution airborne imaging spectroradiometer [3]. This sensor system was developed by the Hughes Danbury Optical Systems in coordination with the Naval Research Laboratory and funded by the U.S. Government. It is mounted on a CV-580 aircraft. The sensor was intended for various purposes such as evaluations of vegetation, water quality, bathymetry, and minerals. The spatial resolution varies from 1 to 4 m depending on the aircraft's altitude above ground level, and the spectral resolution includes 210 contiguous bandwidths from the visible to shortwave infrared (400-2500 nm).

Figure 3–3: HYDICE sensor.

3.1.3 Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor

The Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) is a proven instrument in the realm of Earth Remote Sensing [37]. It is a unique optical sensor that delivers calibrated images of the upwelling spectral radiance in 224 contiguous spectral channels (bands) with wavelengths from 400 to 2500 nanometers [37]. AVIRIS has been flown on four aircraft platforms: NASA's ER-2 jet, Twin Otter International's turboprop, Scaled Composites' Proteus, and NASA's WB-57. The ER-2 flies at approximately 20 km above sea level, at about 730 km/hr [37]. The Twin Otter aircraft flies at 4km above ground level at 130km/hr [37].

AVIRIS uses a scanning mirror to sweep back and forth ("whisk broom" fashion) producing 614 pixels for the 224 detectors each scan. The pixel size and swath width of the AVIRIS data depend on the altitude from which the data is collected. When collected by the ER-2 20km above the ground each pixel produced by the instrument covers an area approximately 20 meters square on the ground with some overlap between pixels, thus yielding a ground swath about 11 kilometers wide. When collected by the Twin Otter (4km above the ground), each ground pixel is 4m square, and the swath is 2km wide.

Figure 3–4: AVIRIS sensor [38].

3.2 Algorithm Development

Joint distribution of pixels rather than single pixel distribution is used to develop an algorithm for spatial-spectral feature extraction. The methods used to obtain the final classification result are discussed in this section (see Figure 3–1).

3.2.1 Dimensionality Reduction

In this work, dimensionality reduction is only applied to the hyperspectral datasets (not to multispectral datasets). So, once we have the hyperspectral data that we will use, the next stage consists of reducing the dimension of the original data. There are different methods used for dimensionality reduction, here the SVDSS has been used (see Section 2.2.1.2). It is available in the Hyperspectral Image Analysis Toolbox (HIAT). With this method the data was reduced based on the largest eigenvalues of the covariance matrix. Each eigenvalue is equal to the variance of the corresponding band and the sum of all eigenvalues must equal to the sum of all the band variances of the original image, thus preserving the total variance of the data. The percentage of each eigenvalue relative to all eigenvalues is calculated until the sum of each percentage reach a predefined threshold. Percentages above 90% are appropriate to be chosen as the threshold since most of the data variance is in the selected bands.

3.2.2 Spectral Difference Histogram (SDH)

The SDH algorithm will extract regional texture information by computing SDHs over window extents. Figure 3–5 presents a block diagram for the SDH algorithm methodology. Before starting to compute the histograms some parameters are to be set. Some of these parameters are the window size and the spectral metric to be used. Different spectral metrics are used to compute distance between pixels. One way to make this is using the spectral angle distance (SAD).

3.2.2.1 SAD distance metric

Suppose we have two N-dimensional vectors, say a_i and b_j for example, defined as $a_i = (a_{i1}, a_{i2}, \cdots, a_{iN})^T$ and $b_j = (b_{j1}, b_{j2}, \cdots, b_{jN})^T$. The SAD between these two vectors can be calculated as:

$$SAD = \arccos\left(\frac{a_i \cdot b_j}{\|a_i\| \cdot \|b_j\|}\right) = \arccos\left(\frac{\sum_{l=1}^{N} a_{il} \cdot b_{jl}}{\left[\sum_{l=1}^{N} a_{il}^2\right]^{\frac{1}{2}} \cdot \left[\sum_{l=1}^{N} b_{jl}^2\right]^{\frac{1}{2}}}\right) \tag{3.1}$$

The interesting property of the SAD is that the cosine of the angle is equivalent to the correlation coefficient of the observations a_i and b_j. It means that the SAD is a statistical method for expressing similarity or dissimilarity between two pixel vectors a_i and b_j. The correlation coefficient ranges from -1 to $+1$. Therefore, the SAD values range between 0 and π. The closer the correlation coefficient is to -1 or $+1$, or in other words, the closer the SAD value is to 0 and π, the more the observations a_i and b_j are said to be correlated.

3.2.2.2 SDHs Computation

Suppose that we have an input image I of size $M \times N \times P$, and a square window W of size $m_w \times n_w \times p_w$ and center in $\left(\frac{m_w-1}{2}, \frac{n_w-1}{2}\right)$. The center of W is moved from pixel to pixel starting at the top left corner of I. For each W, Equation 3.2 is applied to compute distances between pixels in it at four different angles, $\alpha = 0, 45, 90, 135$:

$$S_\alpha(i,j) = SAD\Big(W(i,j,:), W(i+\Delta x, j+\Delta y, :)\Big) \tag{3.2}$$

for $i = x_i, x_i+1, \cdots, m_w$ and $j = y_i, y_i+1, \cdots, n_w - y_f$, where x_i and y_i are the start row and column of the window for computing the SDH, and y_f specify the final column of the window that accounts in the SAD calculation. Δx and Δy specify the row and column of the neigboring pixel, respectively. All these variables depends on α.

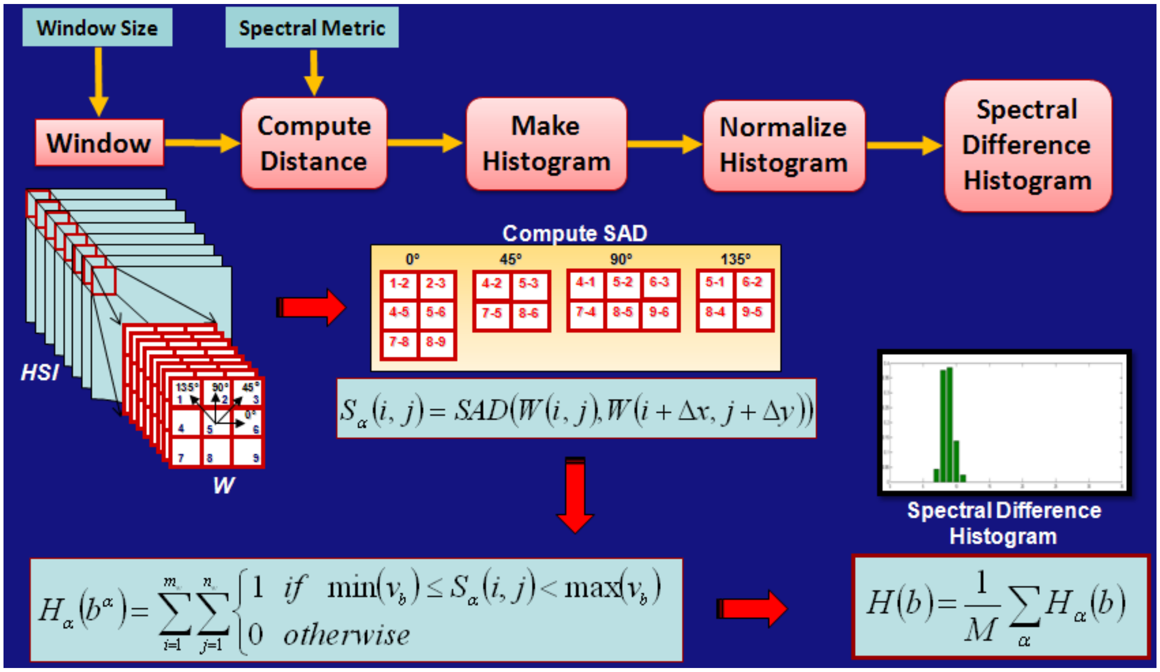


Figure 3–5: Spectral difference histogram algorithm methodology.

The SAD values obtained from angle α are assigned to a specific bin b to make a histogram H_α. For each bin, the number of SAD values that fall into are counted:

$$H_\alpha(b^\alpha) = \sum_{i=1}^{m_w}\sum_{j=1}^{n_w}\begin{cases}1 & \text{if } min(v_b) \le S_\alpha(i,j) < max(v_b)\\ 0 & \text{otherwise}\end{cases} \tag{3.3}$$

where $b = 1, 2, \cdots, n_b$, n_b is the number of bins of the histogram H_α, $min(v_b)$ and $max(v_b)$ are the minimum and maximum SAD values of the specified bin b, respectively. Equation 3.3 is applied for each angle α and then the resulting histograms:

H_{0°, H_{45°, H_{90° and H_{135° are combined to finally obtain the SDH represented by:

$$H(b) = \frac{1}{M} \sum_{\alpha} H_{\alpha}(b) \tag{3.4}$$

where $M = 2\big(2m_w n_w - 2m_w - n_w + 1\big)$ is the total number of SAD values for the window W. As the number of rows and columns of W is the same: $m_w = n_w = m$ we can express M in terms of m as: $M = 2\big(2m^2 - 3m + 1\big)$.

3.2.2.3 SDH - Number of bins and bin size

The histogram bins are the numerical ranges where you are going to group the data into [39]. They should have the same size and should encompass all of the data. When you make a histogram, you need to choose a bin size. If you choose too small bin size, the bar height at each bin suffers from significantly large statistical fluctuation due to the paucity of samples in each bin [40]. But, if you choose too large bin size, the histogram can not represent the shape of the underlying distribution because the resolution isn't good enough [40]. Generally you should never have fewer than 5 or 6 bins [39]. The more data you have the more bins you should have, and some people recommend using the square root of the number of data points as your bin size [39].

In the literature, we found a method (Matlab code) for selecting the bin size of a histogram [40]. It is computed as follows:

1. Divide the data range into N bins of width Δ. Count the number of events k_i that enter the i'th bin.
2. Calculate the mean and variance of the number of events as $k = \frac{1}{N}\sum_{i=1}^{N} k_i$ and $v = \frac{1}{N}\sum_{i=1}^{N} k_i - k^2$.
3. Compute a formula, $C(\Delta) = \frac{2k-v}{\Delta^2}$.

4. Repeat 1-3 while changing Δ. Find Δ^* that minimizes $C_n(\Delta)$. Δ^* will be the optimum bin size.

The number of bins, n_b can be calculated by : $n_b = \frac{max(SAD_{value})-min(SAD_{value})}{\Delta^*}$.

3.2.3 Feature Extraction

Joint distribution of pixels rather than single pixels distribution is used to develop the SDH algorithm for spatial-spectral feature extraction. The SDHs are selected as features to be used in the semi-supervised learning.

3.2.4 Semi-Supervised Learning and Classification Process

The classifier used in our work is the semi-supervised SVMs. The concept of how it works is explained in details in Section 2.3.3. The semi-supervised learning framework used both labeled and unlabeled samples for training the SVM classifier[1] . The classifier performance measured by the labeled and unlabeled samples were calculated and finally the HSI is classified.

3.2.5 Post-processing

After the HSI is classified, it is smoothed for edge correction using a majority filter of different sizes. Suppose we select an odd window (or filter) W_s of size $m_s \times n_s$ with center $\left(\frac{m_s-1}{2}, \frac{n_s-1}{2}\right)$. The center of the window is moved from pixel to pixel starting at the left corner of the thematic map obtained from the classification

[1] The semi-supervised SVM Matlab code used in our work was obtained from http://www.kyb.tuebingen.mpg.de/bs/people/chapelle/lds/.

process, say F. The majority filters (or windows) replaces the center pixel of W_s with the class of the majority of the neigboring pixels. Figure 3–6 shows an example that illustrates this concept. In Figure 3–6 (top left) the center pixel of W_s belongs to class 1. When the majority filter is applied it is replaced by class 3 (top right) since it is the class with majority of votes (21) in W_s (compared with class 1 and 2 that have 15 and 12 votes,respectively). The same occurs in the window of Figure 3–6 (down) where the center pixel is replaced by class 2 (20 votes).

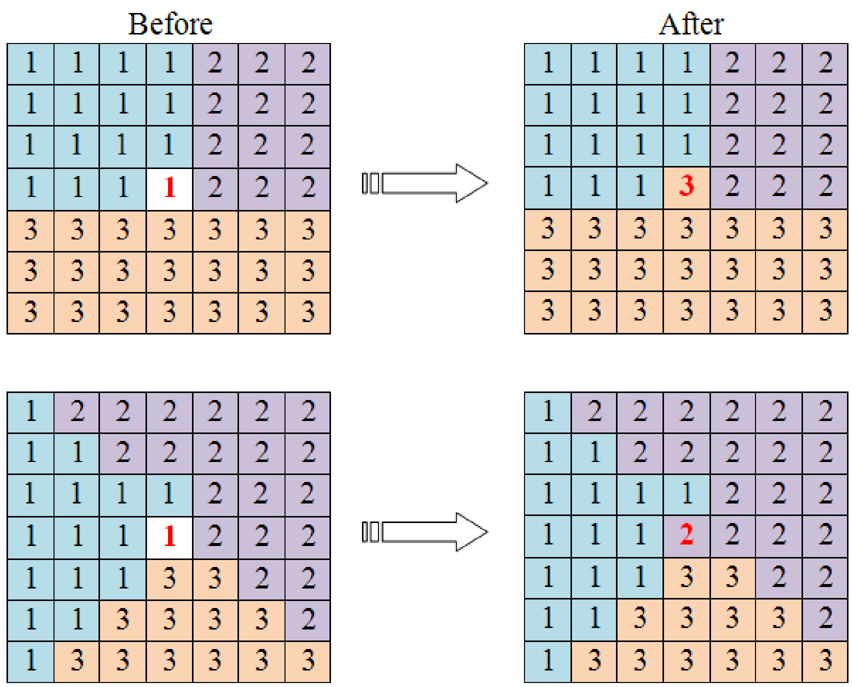


Figure 3–6: Example of a 7 × 7 majority filter.

3.3 Algorithm Extension to Multispectral Images

For multispectral images, the algorithm is similar to that of co-occurrence matrix computation for RGB images [29], [30], [31]. The histogram of occurrence of pixel pairs with the spectral values is constructed. This histogram is computed at different angles and pixel spacing as for the SDH. The pixel values may have to be quantized due to the large size and sparseness of the histogram.

3.4 Algorithm Validation and Performance Evaluation

Validation of the algorithm was performed using:

- Different texture types (natural and man made) such as iron, asphalt, concrete and coins.
- Synthetic images with different noise levels.
- Remote sensing hyperspectral images.

The main reason why we used different hyperspectral textures to validate the algorithm is because they are used in literature [4],[36] to get an idea of the quality of the results. Synthetic images were constructed from hyperspectral images using different classes of it and adding noise.

3.5 Conclusions

The proposed method for spatial-spectral feature extraction based on SDHs was explained in this chapter. The SVDSS method was used for reducing the dimensionality of the original data. The SAD was used as spectral metric to compute distance between pixels in the selected window. Majority filters are used to smooth the thematic map obtained from the classification process.

CHAPTER 4
DATA ANALYSIS AND VALIDATION

Real and synthetic hyperspectral datasets are used to test the algorithm. The datasets are obtained from (1) the SOC-700 camera, (2) HYDICE sensor, and (3) AVIRIS sensor. The algorithm was tested under noise.

4.1 Classification Results for SOC-700 camera data sets

In this section we present the classification results for the datasets that are collected using the SOC-700 hyperspectral camera.

4.1.1 Dataset 1 - Asphalt, Concrete, Iron and Coins Mosaic

Hyperspectral images of different textures are collected using the SOC-700 hyperspectral camera. Figure 4–1 shows the RGB color-composite of the original image of the first dataset that we used. It was created using bands 49, 32 and 18, and has dimensions of $128 \times 128 \times 120$. The HSI consists of four classes: iron (upper left), concrete (upper right), coins (lower left) and asphalt (lower right). Figure 4–5(a) shows the ground truth of the data in Figure 4–1.

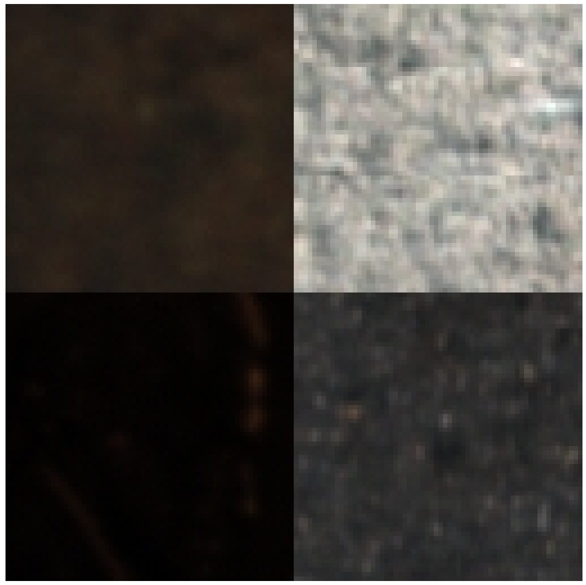


Figure 4–1: SOC-700 hyperspectral image with dimensions 128 × 128 × 120.

The dimensionality of the data was reduced based on the largest eigenvalues of the covariance matrix. For our experiments we selected the threshold to be greater than 92.0%. Most of the variability of the data is represented in the selected bands. Based on the obtained eigenvalues, we reduce the data to 60 bands since with these bands we have the 93.2% of the data variance. The selected bands were: 2, 8, 15, 18, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 69, 70, 72, 74, 77, 79, 81, 86 and 113.

4.1.1.1 Select bin size and the number of bins to compute the histogram

We start computing the histogram using the complete range of the SAD metric, that is from 0 to π, and using 32 bins of width 0.1. Since the data that we used do not have negative values, the range of the SAD metric is reduced from 0 to $\pi/2$. For all the experiments, we noticed that almost all the SAD values were assigned to the first bin, which is the one that corresponds to the values from (0, 0.1], and the

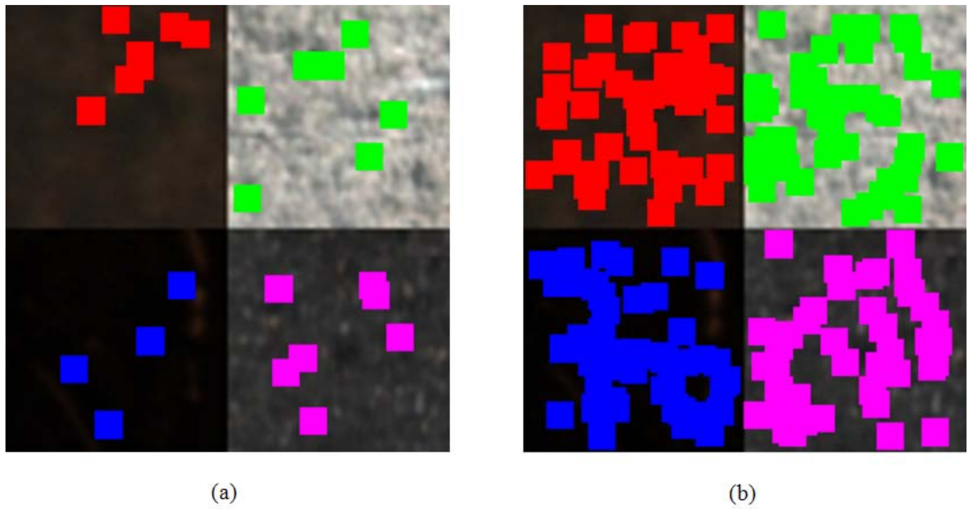


Figure 4–2: (a)Labeled (15 7×7 windows) and (b) unlabeled (175 7×7 windows) samples used to train and test the semi-supervised SVM classifier.

rest were assigned in the second bin (0.1 to 0.2]. Therefore, the histograms can not represent the shape of the underlying distribution because the resolution isn't good enough. Then, we experiment with different number of bins and bin sizes, and finally select 30 bins equally spaced by 0.005. It works well for all the experiments that we performed. When we use these values for the bin and bin size parameters (30 and 0.005 respectively), the histograms were representative of the classes to which they belongs, in other words we can discriminate between them. The optimal bin width calculated using the method found in [40] was 0.003 (for dataset 1). This value is near to the one we selected (0.005).

4.1.1.2 Select Window Size

The window size is an important factor for computing the SDH. The window can not be too large covering a large part of the data, considering that the spatial information is concentrated in the nearest neighbors. It should be neither too small as we lose important information for extracting the features. As explained in section 3.2.2, the SAD distance metric is computed between pixels in the window. Smaller windows does not give us too many numbers of SAD values to construct the histogram. It can be overcome if larger windows are used, but not too large. While the window size increase the number of SAD values also increase, therefore we can discriminate better between the classes. For example, for a 3×3 window we have only 20 SAD values to compute the histogram. On the other hand for windows of size 7×7 and 9×9 we have 156 and 272 SAD values, respectively. Some experiments were made to see the effect of the window size and to choose the most appropiate one. The SDH computed for the same training and testing samples shown in Figure 4–2 using windows of size 3×3, 7×7 and 9×9 are shown in Figure 4–3 and in Figure 4–4 respectively. The overall class performance for the three cases is summarized in Table 4–1. There was an improvement in the performance of the classifier when 7×7 windows are used compared with the 3×3 windows. If the window size is increase to 9×9 there was also an improvement in the performance but the obtained results for both cases were comparable. Since the computational time for 7×7 windows is lower, we select it to be the appropiate window size for our experiments.

Table 4–1: Overall Class Performance for different window sizes.

Window Size	No. errors Lab-Unlab	Performance(%) Lab-Unlab
3×3	2 - 15	92 - 91.4
7×7	0 - 1	100 - 99.4
9×9	0 - 0	100 - 100

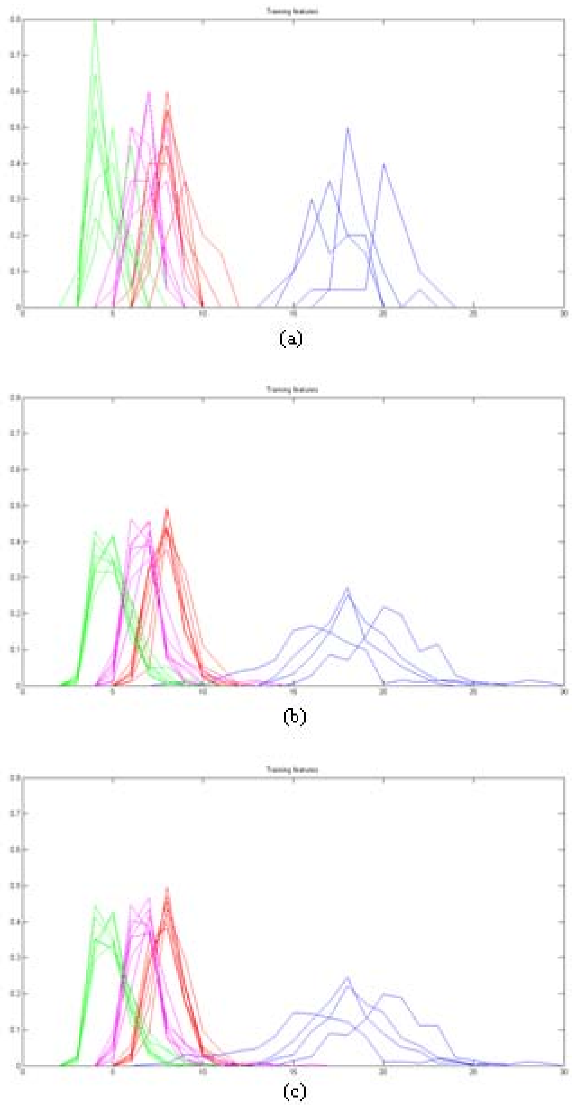


Figure 4–3: The histogram features computed for the labeled samples using windows of size (a) 3×3, (b) 7×7 and (c) 9×9.

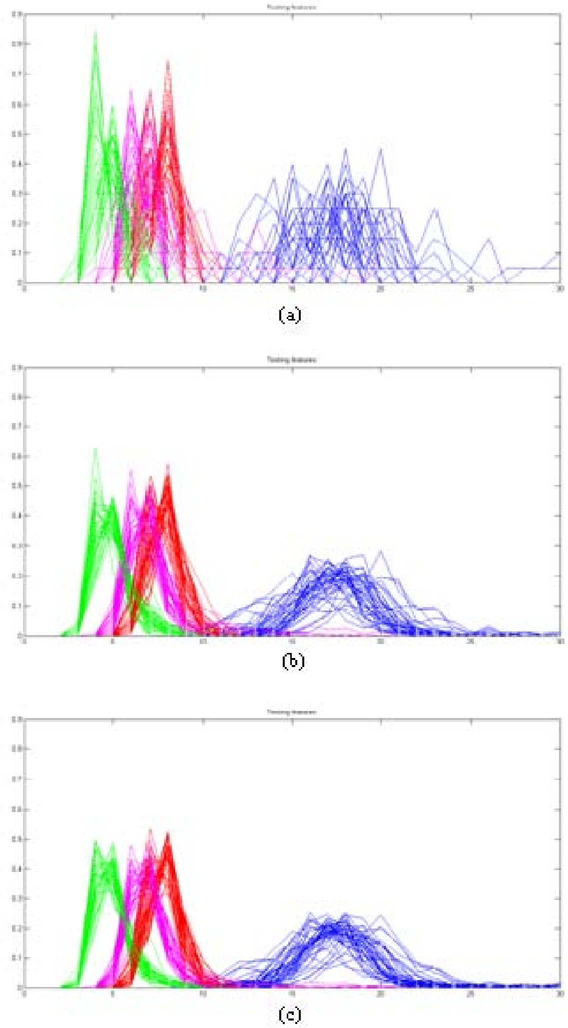


Figure 4–4: The histogram features computed for the unlabeled samples using windows of size (a) 3×3, (b) 7×7 and (c) 9×9.

4.1.1.3 Classification Process

To train the semi-supervised SVM, 25 windows of size 7×7 with known labels and 175 windows of size 7×7 with unknown labels are selected. The windows selected as labeled and unlabeled samples are shown in 4–2 (a) and 4–2 (b), respectively. The spectral difference histogram was computed and the obtained histogram features for labeled and unlabeled samples are shown in Figure 4–3 (b) and Figure 4–4 (b) respectively. The performance of both set of samples is calculated and shown in Table 4–2 and Table 4–3, respectively. It is seen from the Tables that all classes are fairly accurately classified with an overall class performance of 100% for labeled and 99% for unlabeled samples. Note that the unlabeled set contains 175 samples, seven times the labeled set. The image is classified and then smoothed for edge correction using windows of different sizes based on majority of votes. The thematic map obtained after the post-processing stage using 11×11 windows is shown in Figure 4–5(b). Figure 4–5(c) shows the wrongly classified pixels of Figure 4–5(a). Looking at the results we can see that the misclassified pixels are found in the boundaries between the classes.

Table 4–2: Classifier performance as measured by the labeled samples for the $128 \times 128 \times 60$ SOC-700 mosaic shown in Figure 4–1.

Class Name	No.of Samples	Number of Samples in the Class				Accur.+ (%)
		Iron	Concrete	Coins	Asphalt	
Iron	7	7	0	0	0	100
Concrete	7	0	7	0	0	100
Coins	4	0	0	4	0	100
Asphalt	7	0	0	0	7	100
TOTAL	25	4	4	3	4	
Reliability Accuracy (%)*		100	100	100	100	
OVERALL CLASS PERFORMANCE 100%						

+(100 - percent omission error); also called producer's accuracy

*(100 - percent commission error); also called user's accuracy

Table 4–3: Classifier performance as measured by the unlabeled samples for the $128 \times 128 \times 60$ SOC-700 mosaic shown in Figure 4–1.

Class Name	No.of Samples	Number of Samples in the Class				Accur.+ (%)
		Iron	Concrete	Coins	Asphalt	
Iron	43	43	0	0	0	100
Concrete	43	0	43	0	0	100
Coins	46	0	0	46	0	100
Asphalt	43	1	0	0	42	98
TOTAL	175	44	43	46	42	
Reliability Accuracy (%)*		98	100	100	100	
OVERALL CLASS PERFORMANCE 99.4%						

+(100 - percent omission error); also called producer's accuracy

*(100 - percent commission error); also called user's accuracy

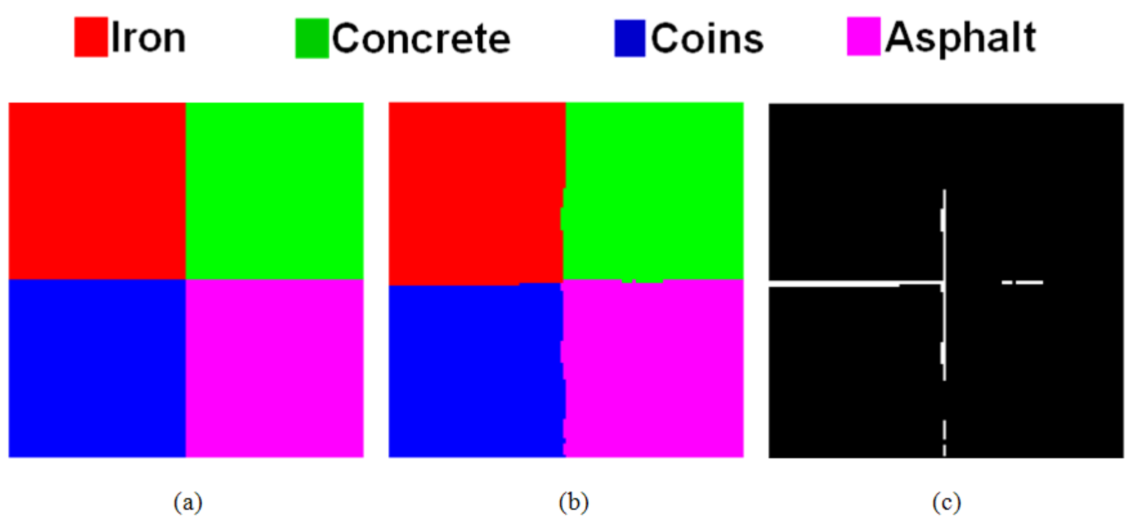


Figure 4–5: (a) Ground truth of the hyperspectral image in Figure 4–1. (b) Thematic map produced by the semi-supervised SVM classifier using spectral difference histograms, 25 labeled 7×7 windows and smoothing with 11×11 windows for the 60 spectral bands. (c) Misclassified pixels of (b), represented in white with respect to the ground truth.

4.1.2 Dataset 2 - Copper, Carton and Cell Phone HSI

Different textures are used to test the algorithm. In this case we used the SOC-700 hyperspectral camera to take an image of a scene that had three different textures: copper, carton and a cell phone. A true color composite of this HSI is shown in Figure 4–6. A subset with dimensions $400 \times 240 \times 120$ was made from Figure 4–6, and the true color composite of this subset is shown in Figure 4–7. Reducing the data to 100 bands gave us 97.8% of the eigenvalues. The selected bands using the SVDSS were: 1, 8, 19, 23, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119 and 120. Twelve (12) labeled and 580 unlabeled samples were used to train the semi-supervised SVM classiffier. Figure 4–8(a) and Figure 4–8(b) shows the selected windows for labeled and unlabeled samples, respectively. The computed SDH of these samples are shown in Figure 4–9 for the labeled samples and in Figure 4–10 for the unlabeled samples. The overall class performance measured by both set of samples was 100% and 94.8% and is shown in Table 4–6 and Table 4–7, respectively. It is seen from these tables that all the classes were reasonably accurately classified except there was confusion between copper and carton. It is noticed in Figure 4–10 where the SDH of both classes overlap.

A small subset of the HSI shown in Figure 4–7 with dimensions $72 \times 72 \times 100$ was made for classification purposes. Figure 4–13(a) shows an approximate ground truth of the data. The image was classified using the same labeled samples shown in

Figure 4–8(a) that are now viewed closely in Figure 4–12. The result of the classification is presented in Figure 4–13(b) after smoothing using 7×7 windows. Figure 4–13(c) shows the misclassified pixels of Figure 4–13(b) with respect to the ground truth. From the results we can see that the misclassified pixels are found in the boundary between the classes.

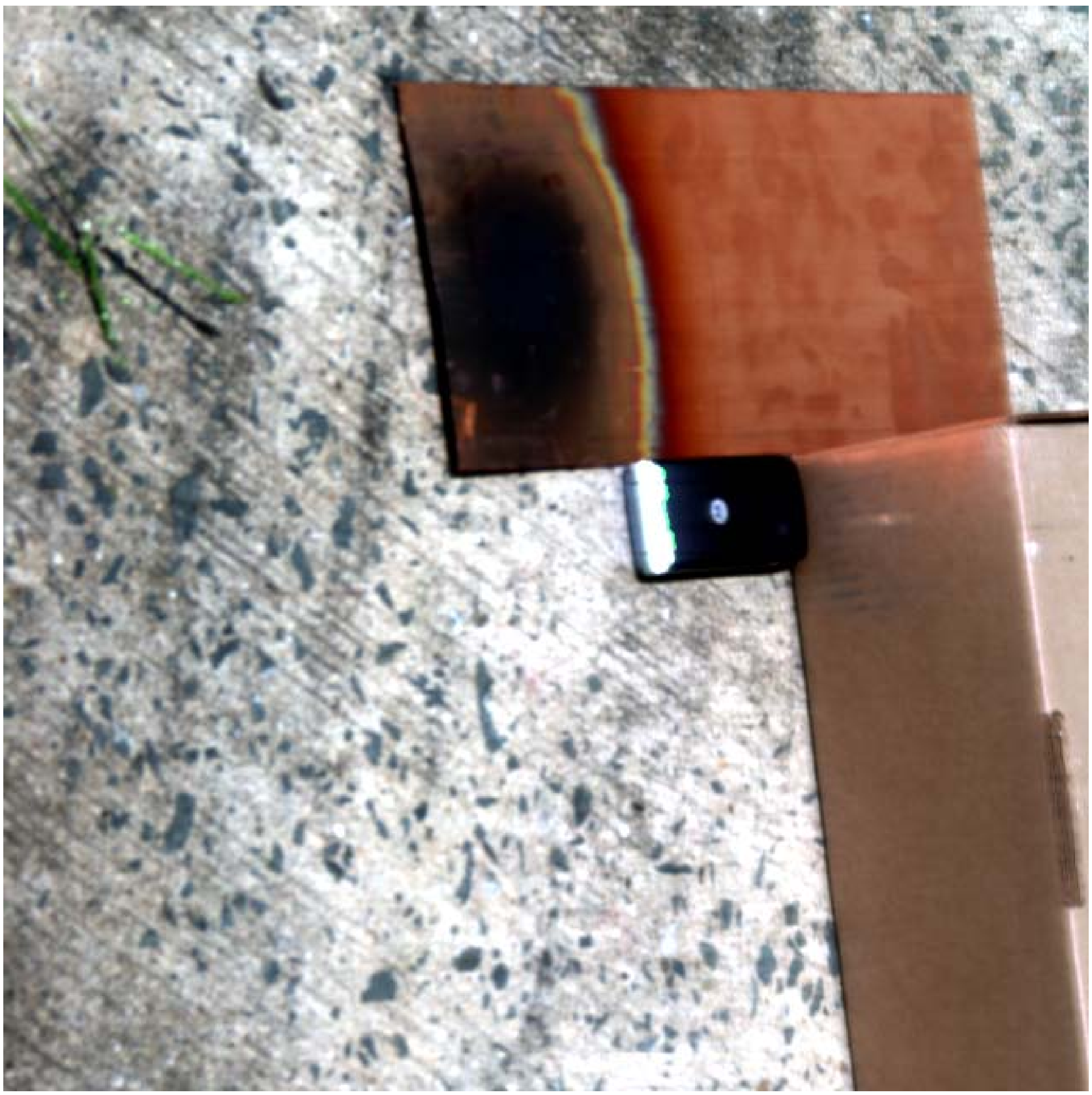


Figure 4–6: SOC-700 hyperspectral image with dimensions $640 \times 640 \times 120$.

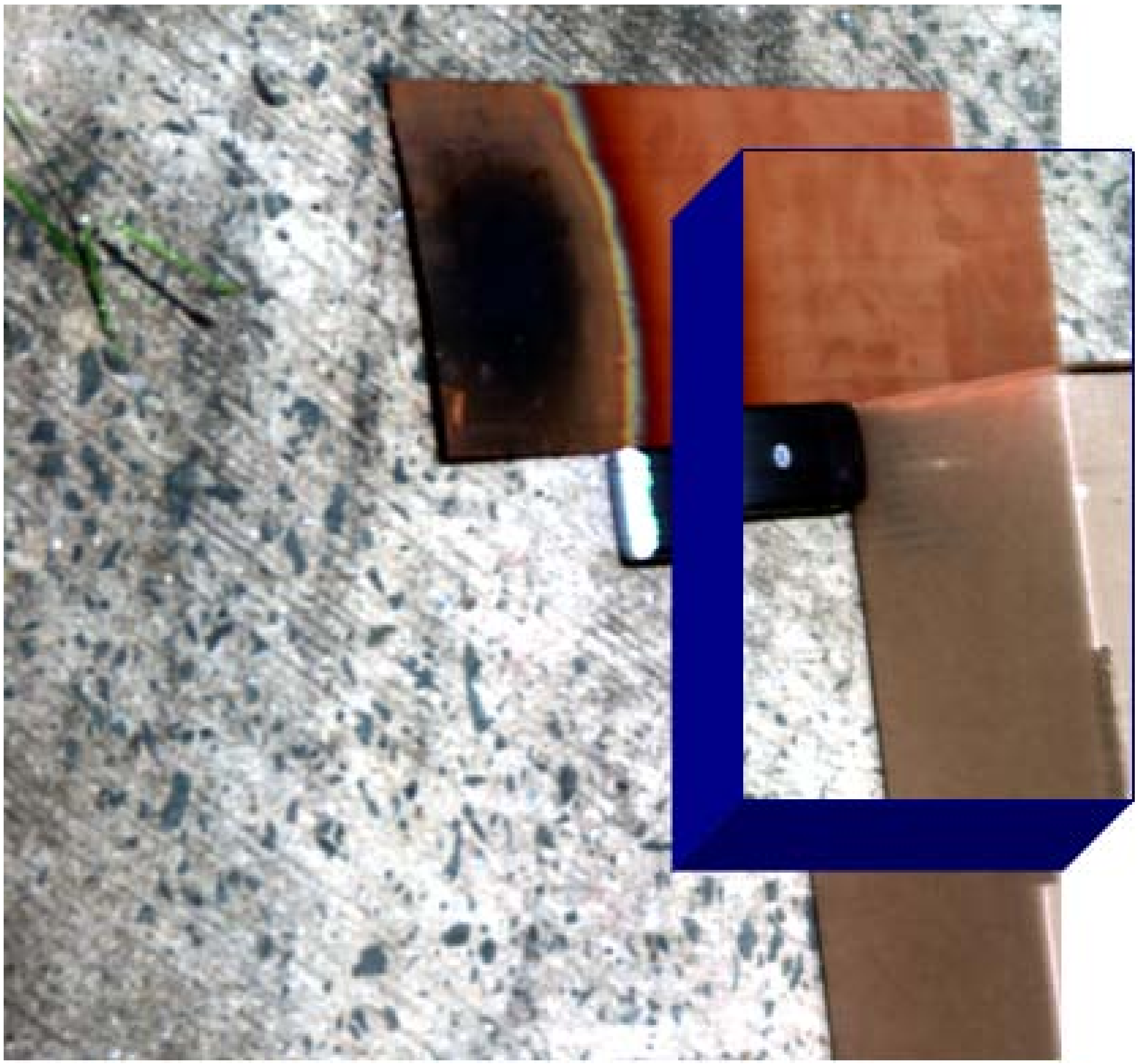


Figure 4–7: A subset of Figure 4–6 with dimensions $400 \times 240 \times 120$.

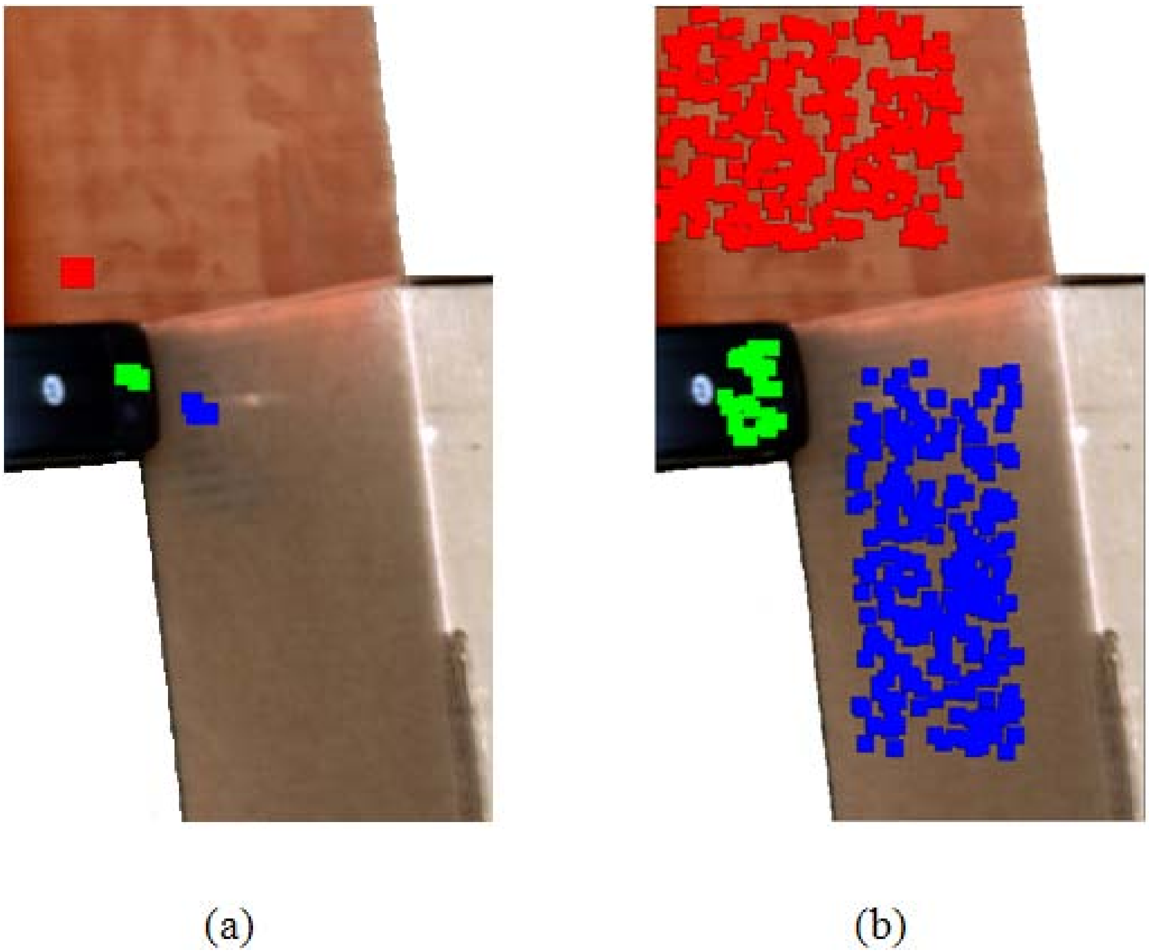


Figure 4–8: (a) Labeled and (b) unlabeled window samples of Figure 4–7.

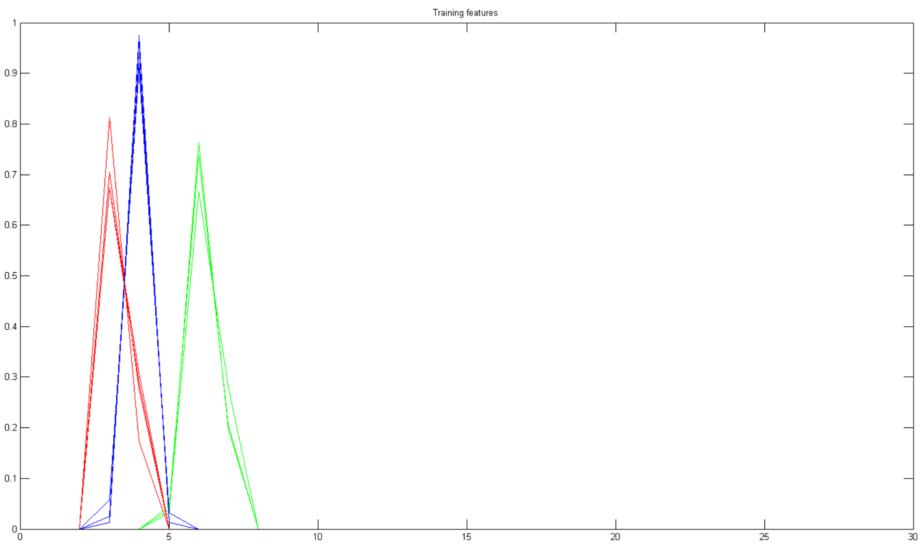


Figure 4–9: The SDH features for the labeled samples shown in Figure 4–8 (a).

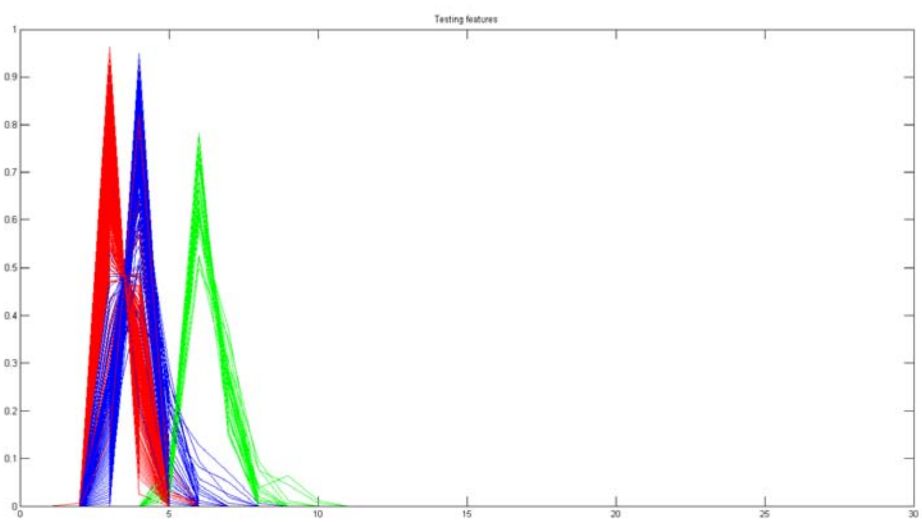


Figure 4–10: The SDH features for the unlabeled samples shown in Figure 4–8 (b).

Table 4–4: Classifier performance as measured by the labeled samples for the 400 × 240 × 120 SOC-700 mosaic shown in Figure 4–7.

Class Name	No.of Samples	Number of Samples in the Class Copper	Phone	Carton	Accur.$^{+}$ (%)
Copper	5	5	0	0	100
Phone	3	0	3	0	100
Carton	4	0	0	4	100
TOTAL	12	5	3	4	
Reliability Accuracy (%)*		100	100	100	
OVERALL CLASS PERFORMANCE 100%					
+(100 - percent omission error); also called producer's accuracy					
*(100 - percent commission error); also called user's accuracy					

Table 4–5: Classifier performance as measured by the unlabeled samples for the 400 × 240 × 120 SOC-700 mosaic shown in Figure 4–7.

Class Name	No.of Samples	Number of Samples in the Class Copper	Phone	Carton	Accur.$^{+}$ (%)
Copper	300	279	0	21	93
Phone	30	0	30	0	100
Carton	250	9	0	241	96
TOTAL	580	288	30	262	
Reliability Accuracy (%)*		97	100	92	
OVERALL CLASS PERFORMANCE 94.8%					
+(100 - percent omission error); also called producer's accuracy					
*(100 - percent commission error); also called user's accuracy					

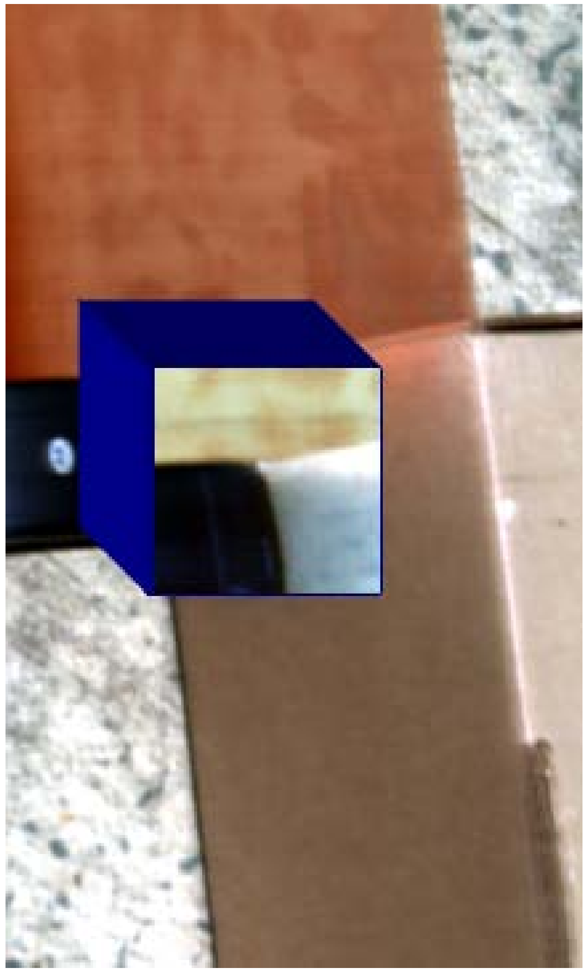


Figure 4–11: A subset of Figure 4–7 with dimensions $72 \times 72 \times 100$.

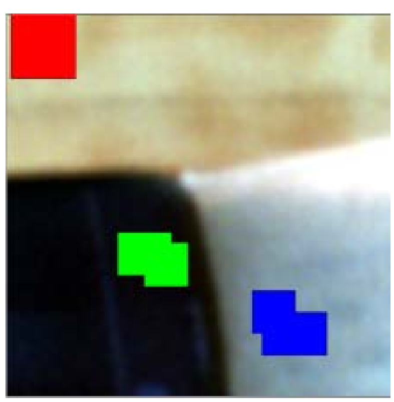


Figure 4–12: The labeled samples of the SOC-700 hyperspectral image with dimensions $72 \times 72 \times 100$.

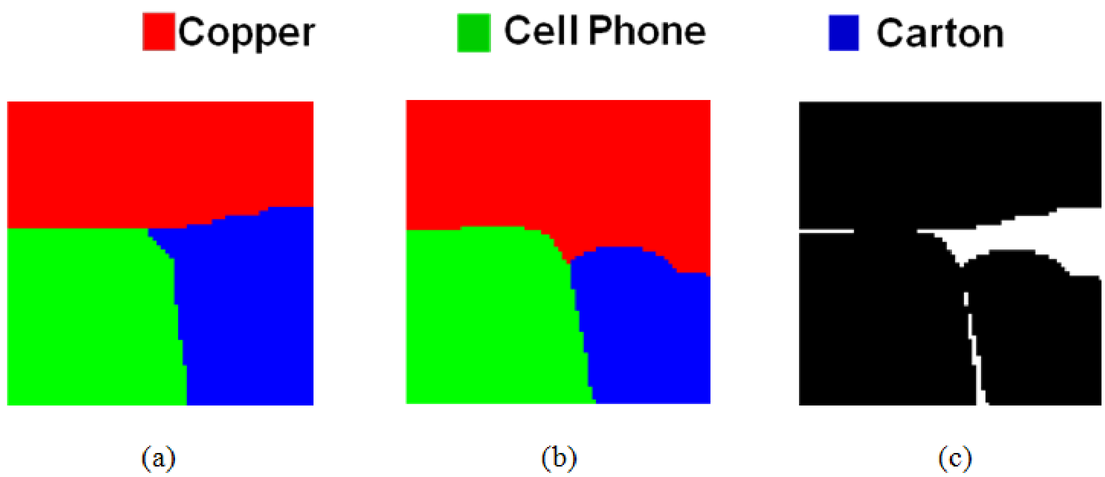


Figure 4–13: (a) Ground truth of the hyperspectral image subset of Figure 4–11. (b) Thematic map produced by the semi-supervised SVM classifier using spectral difference histograms, 12 labeled 7×7 windows and smoothing with 7×7 windows for the 100 spectral bands. (c) Misclassified pixels of (b).

4.1.3 Dataset 3 - Concrete, Leaf and Aluminum HSI

Another experiment consists of three different classes: concrete, leaf and aluminum. The true color-composite of the HSI using bands 49, 32 and 18 is shown in Figure 4–14. The dimensionality of it was reduced to 70 bands after apply the SVDSS. The selected bands were: 2, 8, 12, 14, 16, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 67, 68, 69, 70, 71, 72, 73, 75, 76, 78, 79, 81, 82, 85, 87, 90 and 110. These bands represent 95% of the eigenvalues. A subset of the original HSI was made and is shown in Figure 4–15. Fifteen (15) labeled and 350 unlabeled samples were selected from Figure 4–15 to calculate the performance of the semi-supervised SVM classifier. The selected samples are shown in Figure 4–16(a) and Figure 4–16(b) respectively. The SDH is computed over these samples and the results are shown in Figure 4–17 for the labeled and in Figure 4–18 for the unlabeled samples.

Figure 4–14: True color-composite of the original HSI of three different classes: concrete, leaf and aluminium. The dimensions of it are $640 \times 640 \times 120$.

Figure 4–15: True color-composite of a $250 \times 90 \times 120$ subset of Figure 4–14.

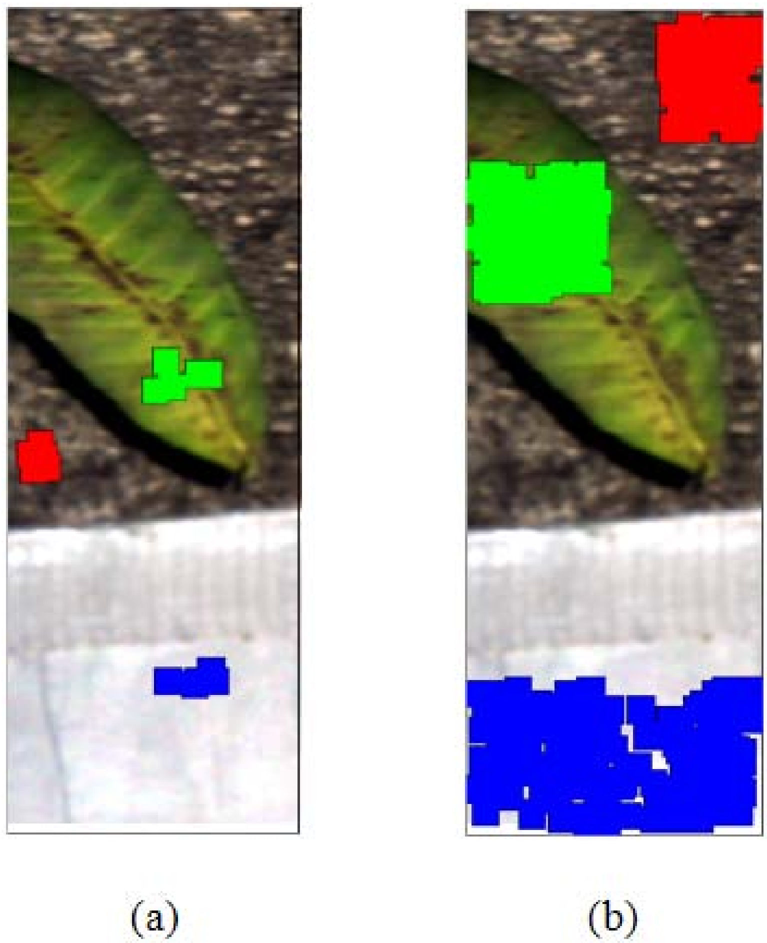


Figure 4–16: (a) The selected labeled (15) and (b) unlabeled samples (325) of Figure 4–15.

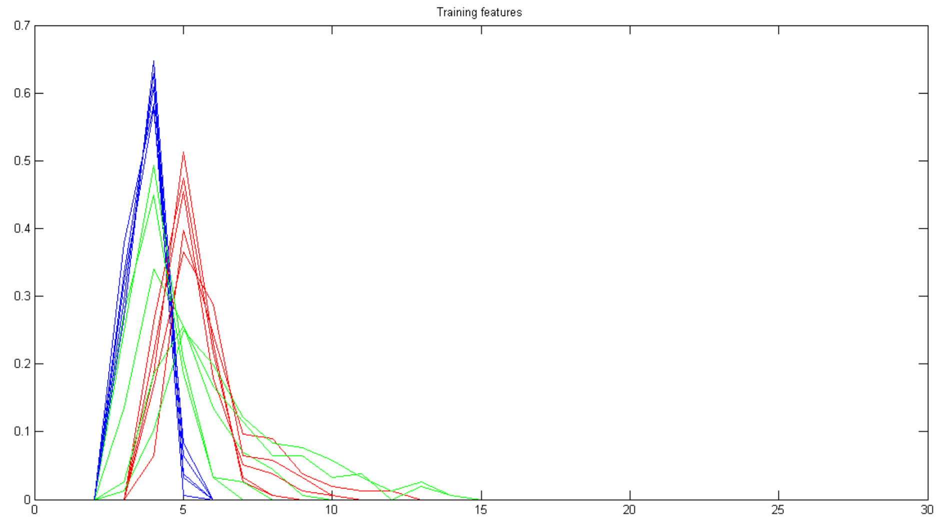


Figure 4–17: The SDH features for the labeled samples shown in Figure 4–16(a).

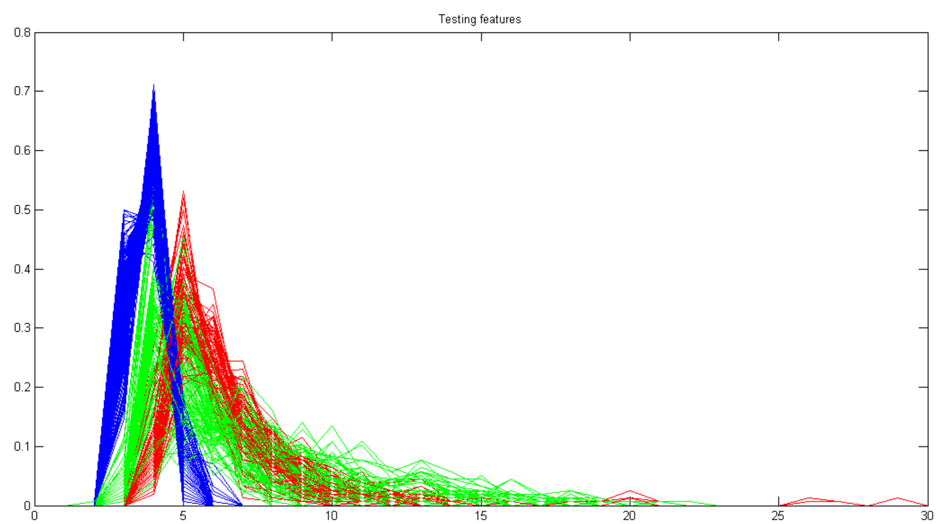


Figure 4–18: The SDH features for the unmlabeled samples shown in Figure 4–16(b).

Table 4–6: Classifier performance as measured by the labeled samples for the 250 × 90 × 100 SOC-700 subset shown in Figure 4–15.

Class Name	No.of Samples	Number of Samples in the Class Concrete	Leaf	Aluminum	Accur.+ (%)
Concrete	5	5	0	0	100
Leaf	5	0	5	0	100
Aluminum	5	0	0	5	100
TOTAL	15	5	5	5	
Reliability Accuracy (%)*		100	100	100	
OVERALL CLASS PERFORMANCE 100%					
+(100 - percent omission error); also called producer's accuracy					
*(100 - percent commission error); also called user's accuracy					

Table 4–7: Classifier performance as measured by the unlabeled samples for the 250 × 90 × 100 SOC-700 subset shown in Figure 4–15.

Class Name	No.of Samples	Number of Samples in the Class Concrete	Leaf	Aluminum	Accur.+ (%)
Concrete	75	73	2	0	97
Leaf	100	6	94	0	94
Aluminum	150	0	0	150	100
TOTAL	325	79	96	150	
Reliability Accuracy (%)*		92	98	100	
OVERALL CLASS PERFORMANCE 97.5%					
+(100 - percent omission error); also called producer's accuracy					
*(100 - percent commission error); also called user's accuracy					

4.2 Classification Results HYDICE sensor data sets

HYDICE airborne hyperspectral data of the Washington, DC Mall was also used in our experiments. Two hundred and ten bands were collected in the 400 to 2400 nm region of the visible and infrared spectrum. The water absorption bands were deleted, resulting in an image with 191 spectral bands. This dataset is available in the student CD-ROM of [41]. This hyperspectral image is used to construct a synthetic image with different class spectra and is shown in Figure 4–19 (left). Figure 4–19(right) shows the band 37 of the synthetic hyperspectral image mosaic constructed from the Washington DC Mall data. Its respective ground truth is shown in Figure 4–23(a). The selected classes are: Roofs, Grass, Trees, and Water [34]. The dimensionality of the original data was reduced to 40 bands since with those bands we obtained 96.7% of the data variance. The selected bands are: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 16, 17, 19, 23, 24, 35, 37, 40, 41, 42, 52, 55, 58, 61, 62, 63, 64, 65, 67, 68, 69, 70, 73, 77, 80, 84 and 112.

Ten (10) labeled samples of size 7x7 and 70 unlabeled samples are selected to train the semi-supervised SVM classifier. The selected windows are shown in 4–20(a) and 4–20(b) respectively. The SDH was computed for those windows to be the features used by the classifier. The SDHs for labeled and unlabeled samples are shown in Figure 4–21 and Figure 4–22. The information of the labeled and unlabeled samples are presented in Table 4–8 and Table 4–9 respectively. As we can see from these Tables all classes were accurately classified. The result of the classification is shown in Figure 4–23(b). The classification error is shown in Figure 4–23(c). We can see from the results that misclassified pixels are found in the class boundaries.

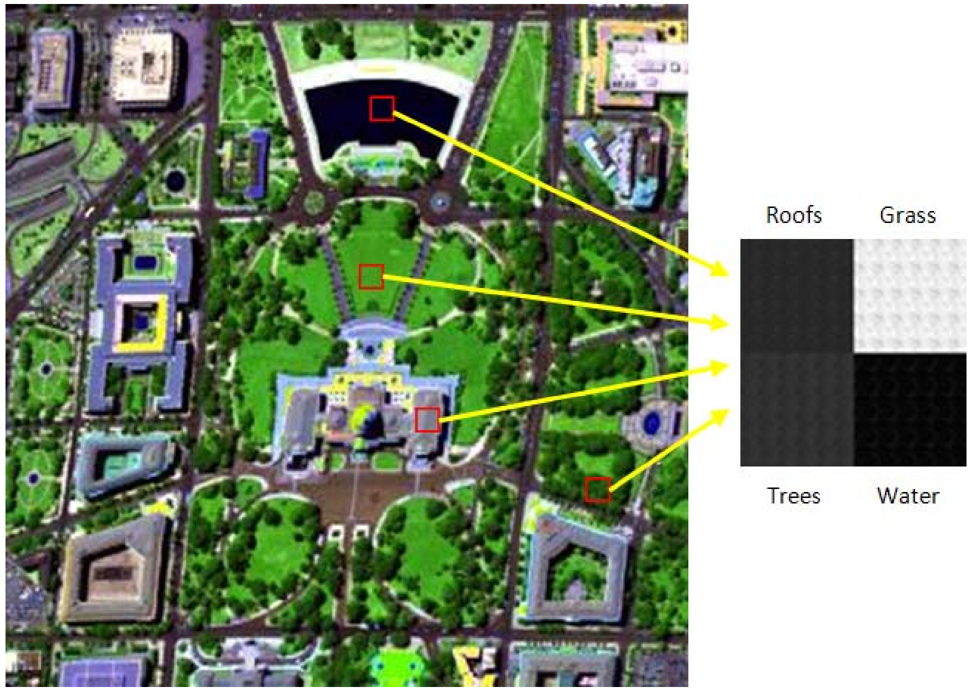


Figure 4–19: Color-composite of the HYDICE HSI of the Washington DC Mall (left). Band 37 of the original synthetic HSI of the Washington DC Mall (rigth).

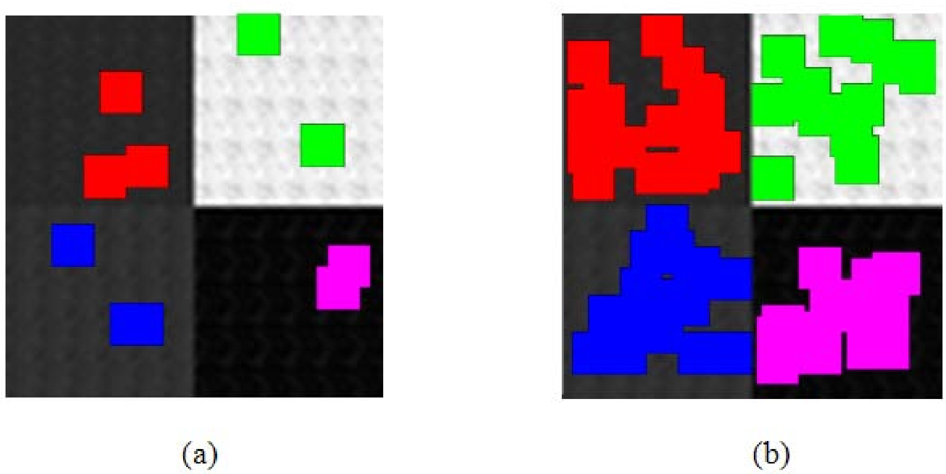


Figure 4–20: (a)Labeled (10 7×7 windows) and (b) unlabeled (70 7×7 windows)samples used to train the semi-supervised SVM classifier.

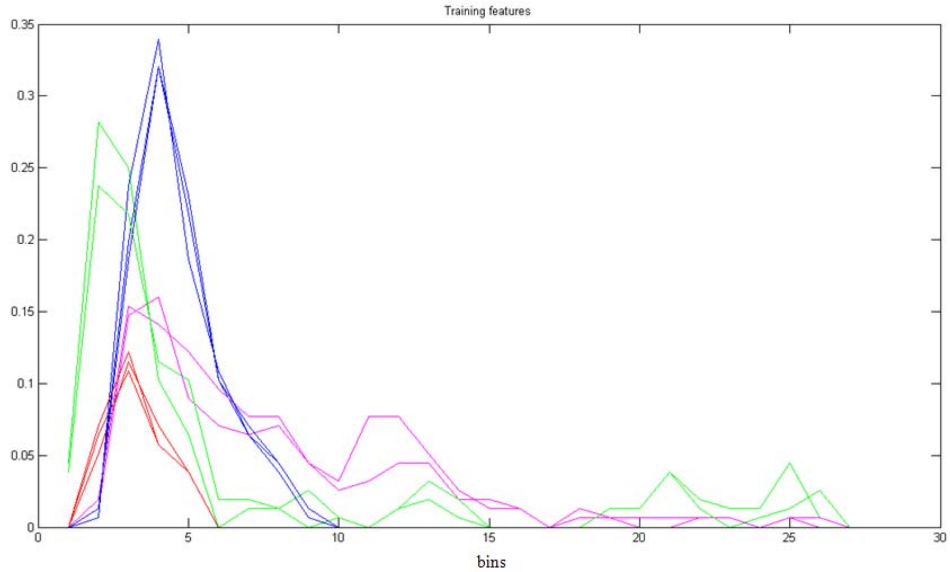


Figure 4–21: The histogram features computed for the labeled samples shown in Figure 4–20(a).

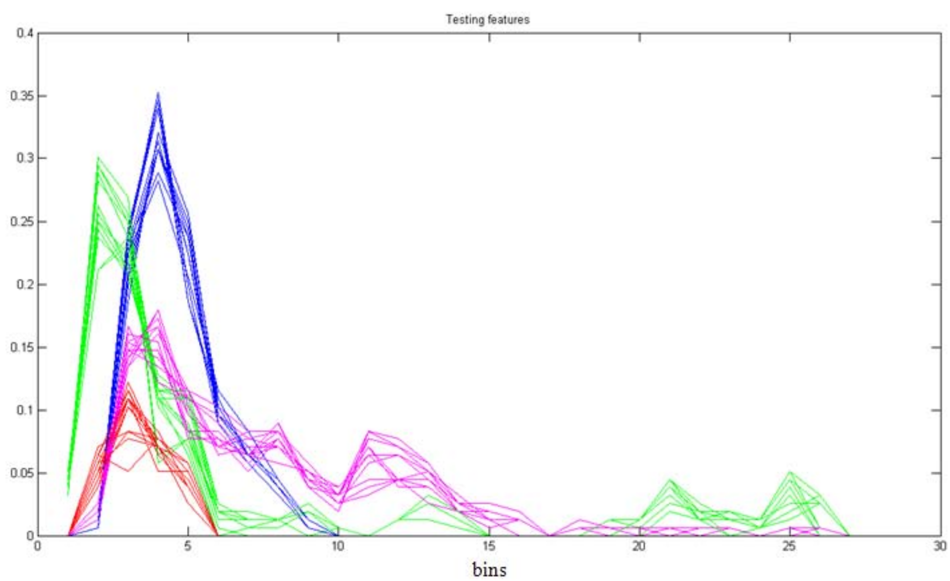


Figure 4–22: The histogram features computed for the unlabeled samples shown in Figure 4–20(b).

Table 4–8: Classifier performance as measured by the labeled samples for the $72 \times 72 \times 40$ HYDICE synthetic image shown in Figure 4–19.

Class Name	No.of Samples	Number of Samples in the Class Roofs	Trees	Grass	Water	Accur.+ (%)
Roofs	3	3	0	0	0	100
Trees	2	0	2	0	0	100
Grass	3	0	0	3	0	100
Water	2	0	0	0	2	100
TOTAL	10	3	2	3	2	
Reliability Accuracy (%)*		100	100	100	100	
OVERALL CLASS PERFORMANCE 100%						
+(100 - percent omission error); also called producer's accuracy						
*(100 - percent commission error); also called user's accuracy						

Table 4–9: Classifier performance as measured by the unlabeled samples for the $72 \times 72 \times 40$ HYDICE synthetic image shown in Figure 4–19.

Class Name	No.of Samples	Number of Samples in the Class Roofs	Trees	Grass	Water	Accur.+ (%)
Roofs	18	18	0	0	0	100
Trees	18	0	18	0	0	100
Grass	17	0	0	17	0	100
Water	17	0	0	0	17	100
TOTAL	70	18	18	17	17	
Reliability Accuracy (%)*		100	100	100	100	
OVERALL CLASS PERFORMANCE 100%						
+(100 - percent omission error); also called producer's accuracy						
*(100 - percent commission error); also called user's accuracy						

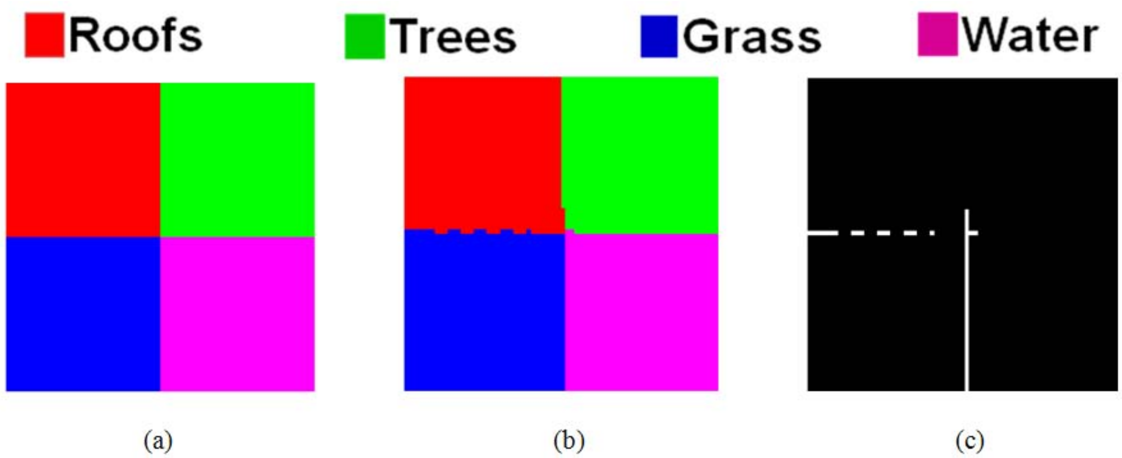


Figure 4–23: (a) Ground truth of the hyperspectral image of Figure 4–19. (b) Thematic map produced by the semi-supervised SVM classifier using spectral difference histograms, 10 labeled 7×7 windows and smoothing with 9×9 windows for the 40 spectral bands. (c) Misclassified pixels of (b), represented in white with respect to the ground truth.

4.3 Classification Results for 7-Classes Experiment

In this experiment we select 7 classes of the previous experiments and input them to the semi-supervised SVM classifier. The results are summarized in Table 4–10 for the labeled samples and in Table 4–11 for the case of the unlabeled samples.

Table 4–10: Classifier performance as measured by the labeled samples for the 7-class experiment.

Class Name	No. of Samples	Number of Samples in the Class							Accur.+ (%)
		Iron	Concrete	Coins	Asphalt	Copper	Phone	Carton	
Iron	7	7	0	0	0	0	0	0	100
Concrete	7	0	7	0	0	0	0	0	100
Coins	4	0	0	4	0	0	0	0	100
Asphalt	7	0	0	0	7	0	0	0	100
Copper	5	0	0	0	0	5	0	0	100
Phone	3	0	3	0	0	0	0	0	0
Carton	4	0	0	0	0	0	0	4	100
TOTAL	37	7	10	4	7	5	0	4	
Reliability Accuracy (%)*		100	70	100	100	100	0	100	
OVERALL CLASS PERFORMANCE		**91.9%**							

Table 4–11: Classifier performance as measured by the un6labeled samples for the 7-class experiment.

Class Name	No. of Samples	Number of Samples in the Class							Accur.+ (%)
		Iron	Concrete	Coins	Asphalt	Copper	Phone	Carton	
Iron	7	7	0	0	0	0	0	0	100
Concrete	7	0	7	0	0	0	0	0	100
Coins	4	0	0	4	0	0	0	0	100
Asphalt	7	0	0	0	7	0	0	0	100
Copper	5	0	0	0	0	5	0	0	100
Phone	3	0	3	0	0	0	0	0	0
Carton	4	0	0	0	0	0	0	4	100
TOTAL	37	7	10	4	7	5	0	4	
Reliability Accuracy (%)*		100	70	100	100	100	0	100	
OVERALL CLASS PERFORMANCE					91.9%				

4.4 Comparisons between SDH, PCA-FE and None-FE Methods

The performance obtained by the SDH algorithm was compared with the performance of the semi-supervised SVM when:

- PCA feature extraction method is applied.
- None feature extraction method is applied (pixel-based classification).

The labeled and unlabeled samples was selected as the center pixel of each window used by the SDH training process. The overall performance for the SDH, PCA-FE and none-FE algorithms is sumarized in Figure 4–24. As we can see from this graph, our method improves or is comparable to the other two methods. The thematic maps for the case of none-FE are shown from Figure 4–25 to Figure 4–27. For the case of dataset 3 of SOC-700 the results are summarized in Table 4–12.

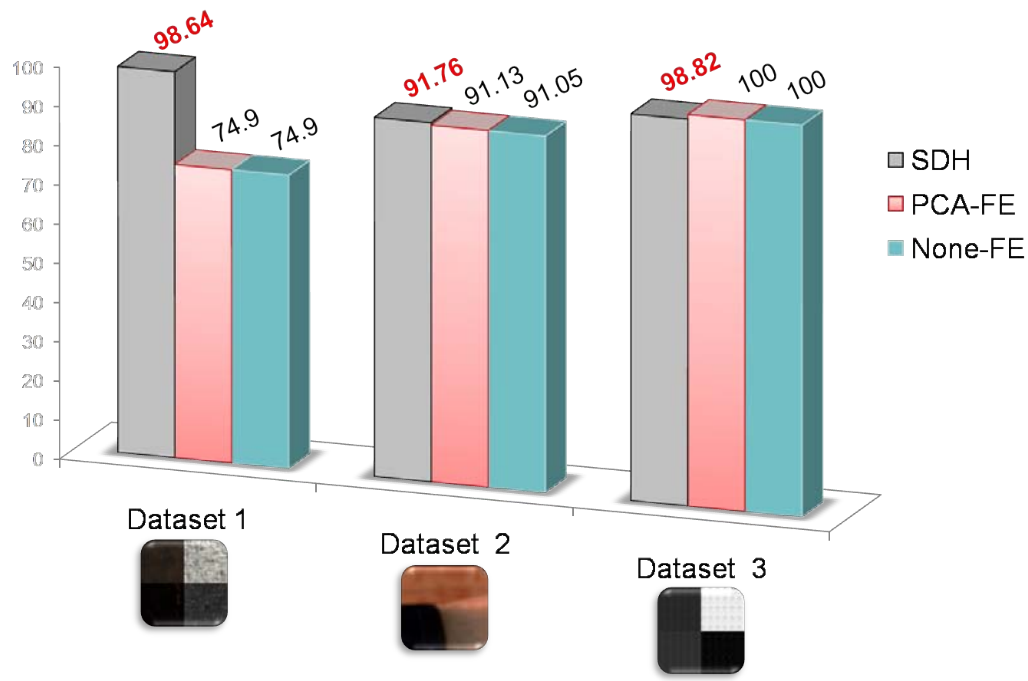
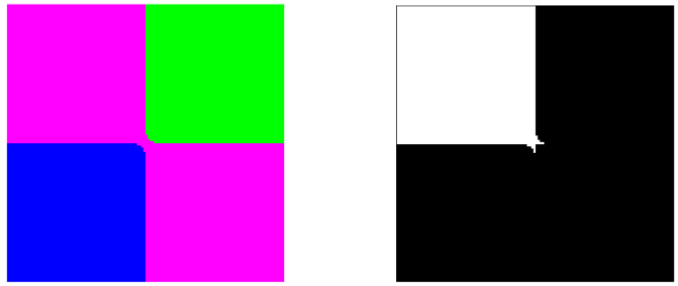


Figure 4–24: Comparison between SDH and pixel-based method for Semi-SVM classifier

Table 4–12: Comparison between the training and testing accuracy of the semi-supervised SVM using SDH, PCA-FE and none-FE method for dataset 3.

Dataset	Acc.(%) - SDH	Acc.(%) - PCA-FE	Acc.(%) - none-FE
3	100 - 97.5	100 - 99.69	100 - 98.77

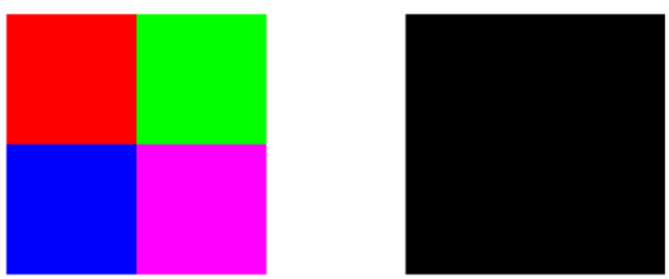


(a) (b)

Figure 4–25: (a)Themathic map produced by the semi-supervised SVM using the data of Figure 4–1 and the pixel spectra as feature. (b)Misclassified pixels of (a) represented in white with repect to the ground truth.

(a) (b)

Figure 4–26: (a)Themathic map produced by the semi-supervised SVM using the data of Figure 4–11 and the pixel spectra as feature. (b)Misclassified pixels of (a) represented in white with repect to the ground truth.



(a) (b)

Figure 4–27: (a)Themathic map produced by the semi-supervised SVM using the data of Figure 4–19(rigth) and the pixel spectra as feature. (b)Misclassified pixels of (a) represented in white with repect to the ground truth.

For the case of the data set 1, which is the case in where our method improves significantly the other two methods, we looked the class spectral signatures. As a result we found that the spectral signatures of iron and asphalt are very similar (see Figure 4–28). It could be the reason why the PCA-FE and None-FE methods failed to discriminate between these two classes.

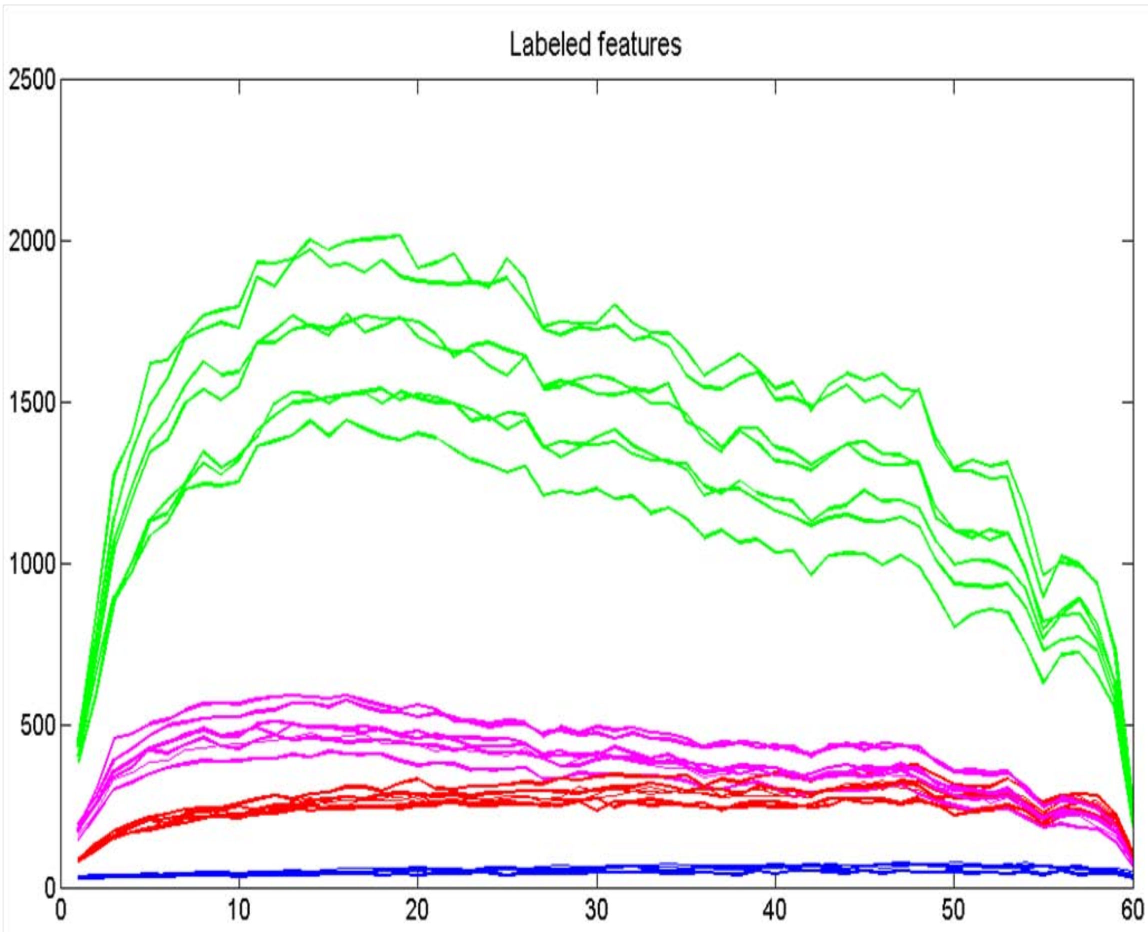


Figure 4–28: Spectral signatures of the classes of data set 1. Iron (red) and Asphalt (magenta) have a very similar spectral signature. PCA-FE and None-FE methods failed to discriminate between them.

4.5 Classification Results for AVIRIS Sensor data set

Real data taken with the AVIRIS sensor over the Enrique Reef in La Parguera Puerto Rico is also used in our experiments. The Enrique Reef is part of the "La Parguera" in the southwest coast of Puerto Rico in the Municipality of Lajas. Figure 4–29 shows the location of the Enrique reef.

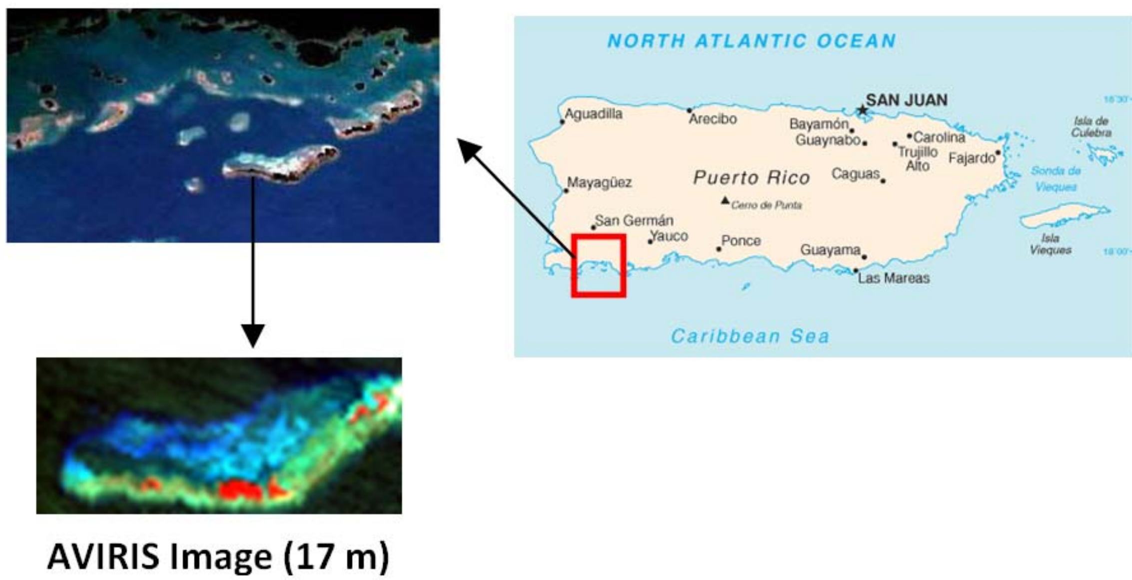


Figure 4–29: Location of the Enrique Reef.

The Enrique Reef data used in this experiment consist of 224 bands from 400 - 2500 nm with a spatial resolution of 17 m. Figure 4–30 shows a color-composite of the Enrique Reef data. For the case of our experiment we take only 3 classes: Water, Reef flat and Sand. Since we have not enough pixels of Mangrove class to compute the SDH, we did not take it in our analysis. To select the labeled samples we use the benthic map (map 158) provided by NOAA (see Figure 4–32). Due to the spatial resolution of the data we selected only one label sample for each class. Figure **??** shows the labeled samples that we used in this experiment. The results of the classification were shown in Figure 4–33 after post-processing using windows of size 3×3, 5×5 and 7×7.

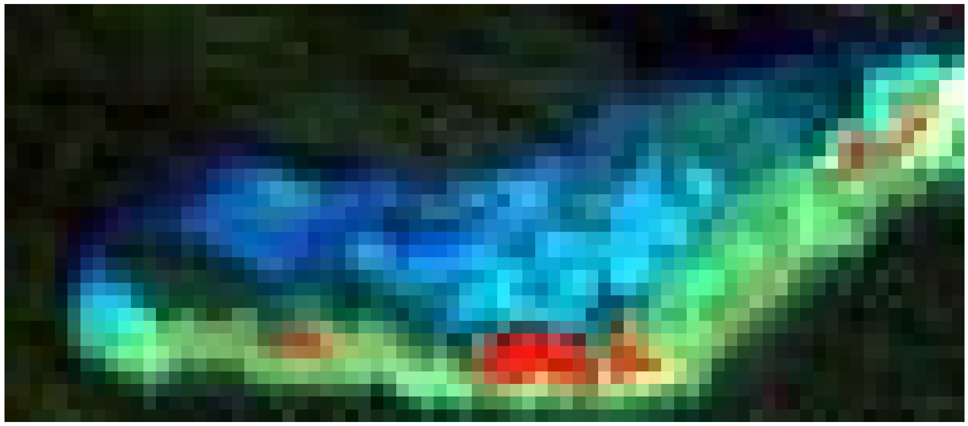


Figure 4–30: Color-composite of the AVIRIS data of the Enrique Reef.

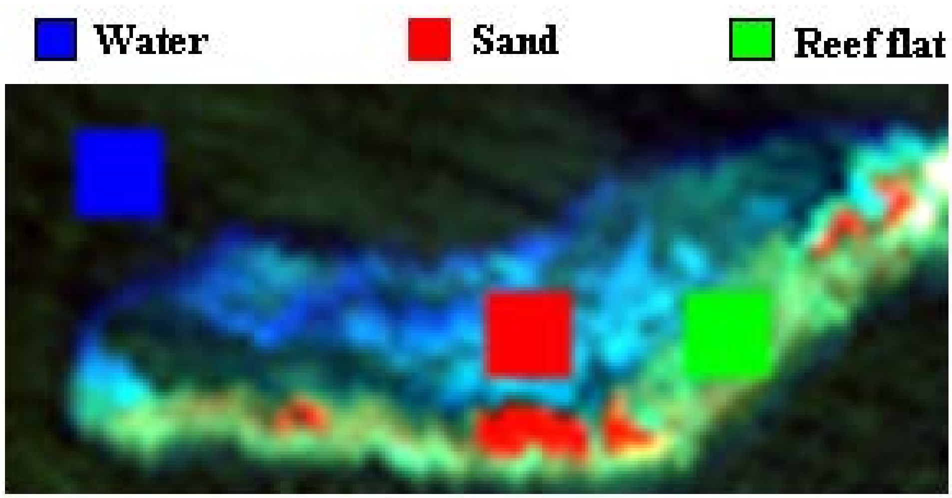


Figure 4–31: Labeled samples used in this experiment.

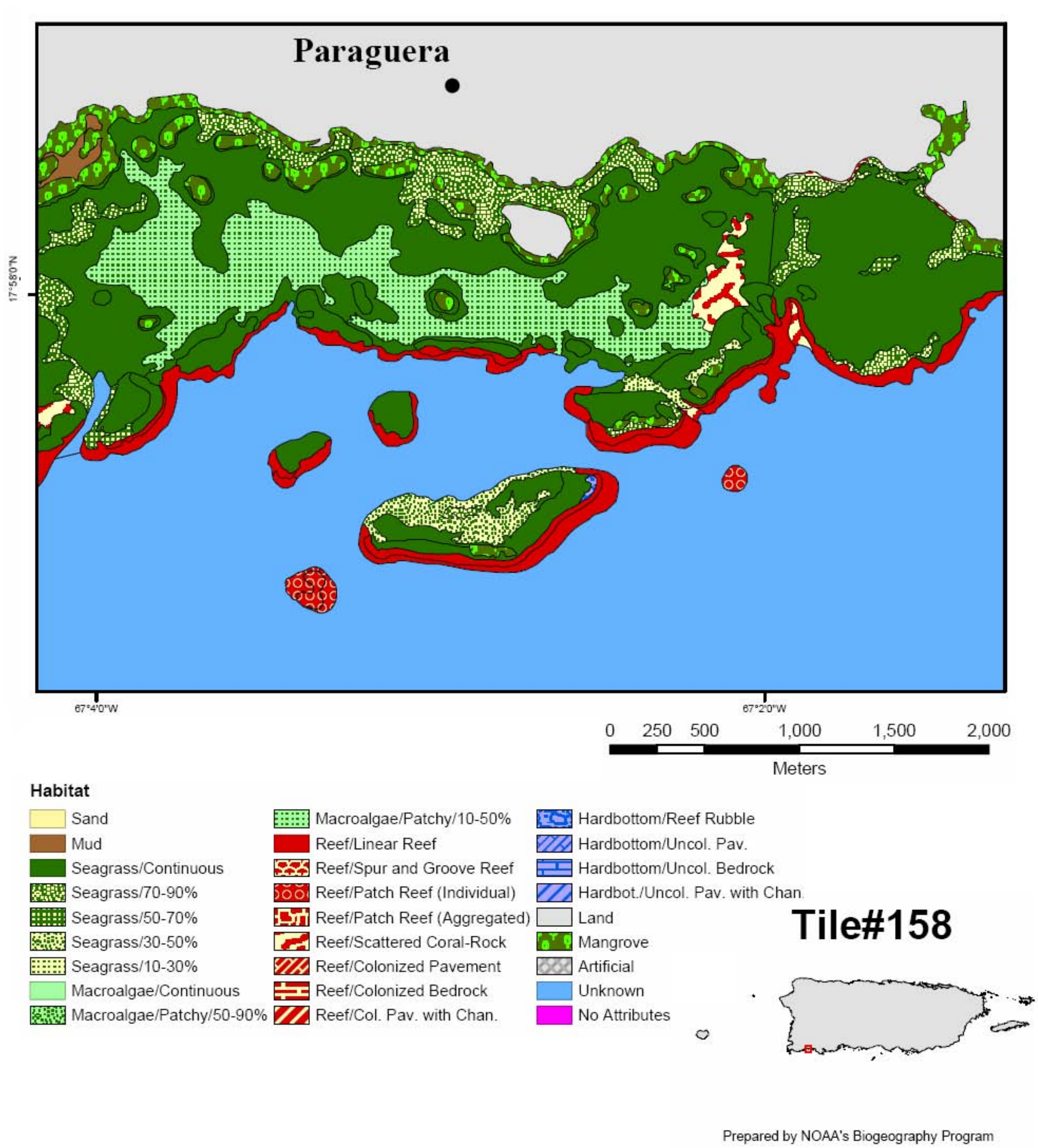


Figure 4–32: NOAA bentic Map of “La Parguera” at lajas Puerto Rico [42].

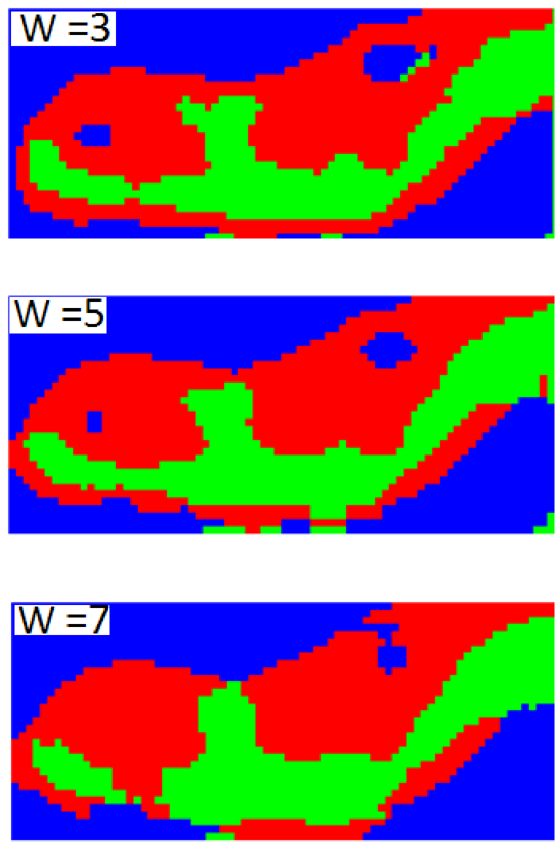


Figure 4–33: Themathic map obtained using the semi-SVM classifier after using majority filters of size 3×3 (top), 5×5 (middle) and 7×7 (bottom).

4.6 Classification Results when Gaussian Noise is Added

Another experiment to validate the performance of the algorithm was conducted. It consists of adding noise to the original image to see how sensitive the proposed algorithm is to the added noise. Mathematically it can be expressed as: $\mathbf{y} = \mathbf{x} + \mathbf{w}$, where $\mathbf{x}$ is the original image and $\mathbf{w}$ is the noise. The noise added is Gaussian noise with zero mean and variance σ^2, $N(0, \sigma^2)$, where σ^2 is one percent (1%) of the maximum value of bands [2]. We use the $128 \times 128 \times 120$ data of Figure 4–1 reduced to 60 bands by the SVD band subset selection method. Figures 4–34 to 4–37 shows the class spectra before and after adding Gaussian noise. The same labeled and unlabeled samples used in the previous experiment are selected, and the accuracy of both datasets was computed again but now under noise. The SDH for labeled and unlabeled is shown in Figure 4–38 and Figure 4–39.The overall class performance measured by the labeled and unlabeled samples was 100% and 99% respectively. It represents no change in the accuracy measured by the unlabeled samples indicating that the proposed algorithm is not sensitive to Gaussian noise for the HSI that we used. Figure 4–40(a) shows the results of the classification for the data in Figure 4–1 plus Gaussian noise. Figure 4–40(b) shows the misclassified pixels relative to the ground truth.

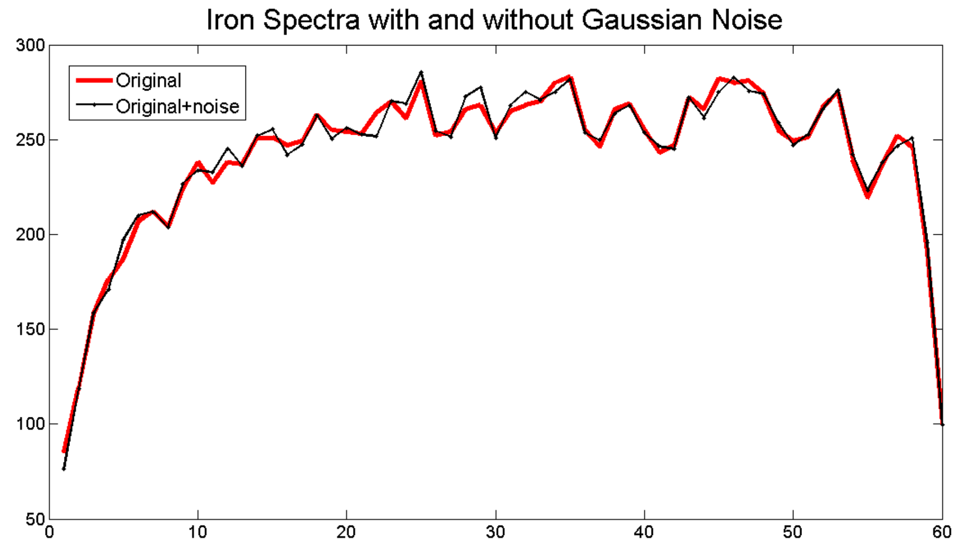


Figure 4–34: The spectra of iron with(black) and without(red) Gaussian noise.

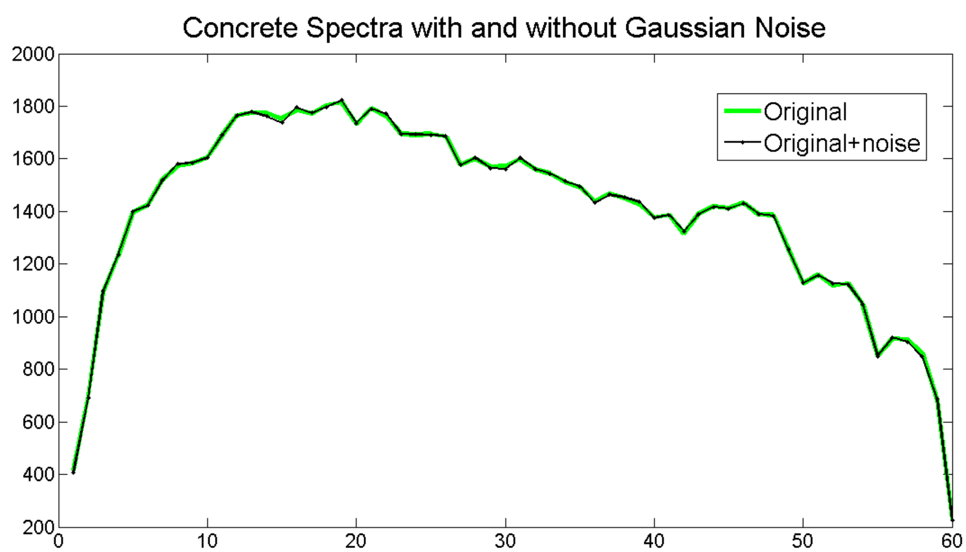


Figure 4–35: The spectra of concrete with(black) and without(green) Gaussian noise.

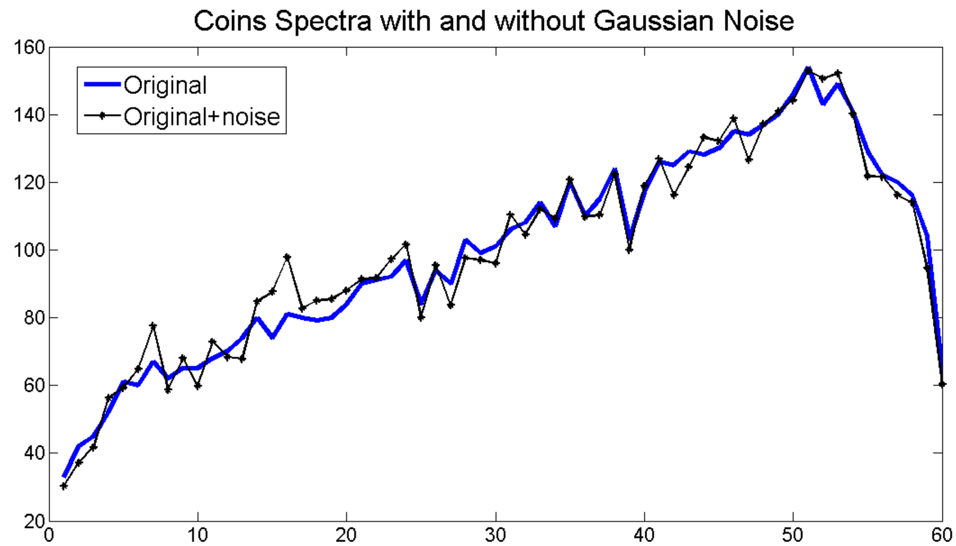


Figure 4–36: The spectra of coins with(black) and without(blue) Gaussian noise.

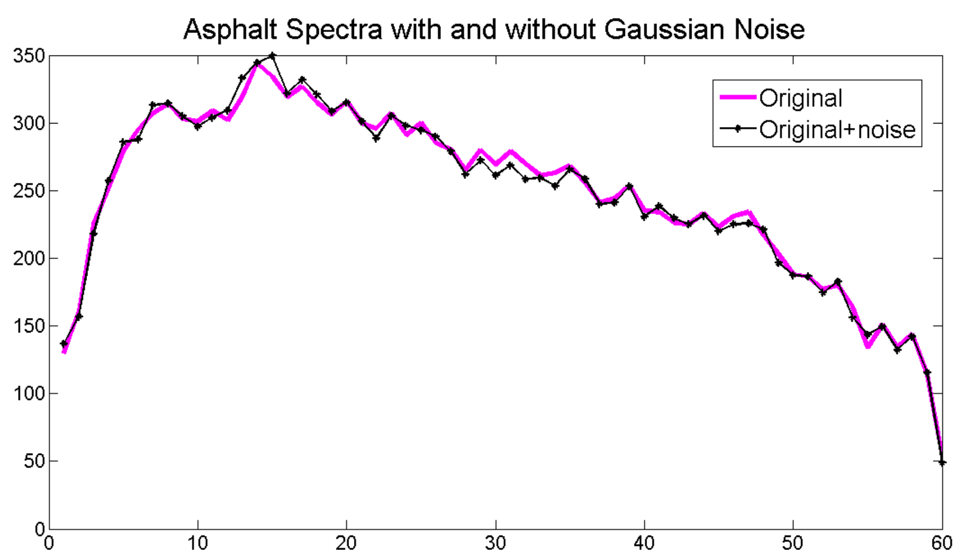


Figure 4–37: The spectra of asphalt with(black) and without(magenta) Gaussian noise.

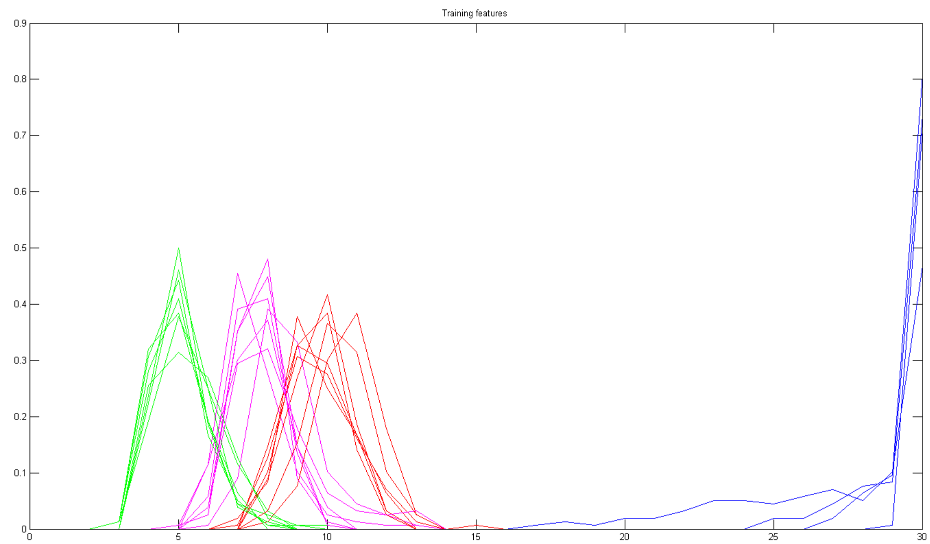


Figure 4–38: The histogram features computed for the labeled samples shown in Figure 4–2(a).

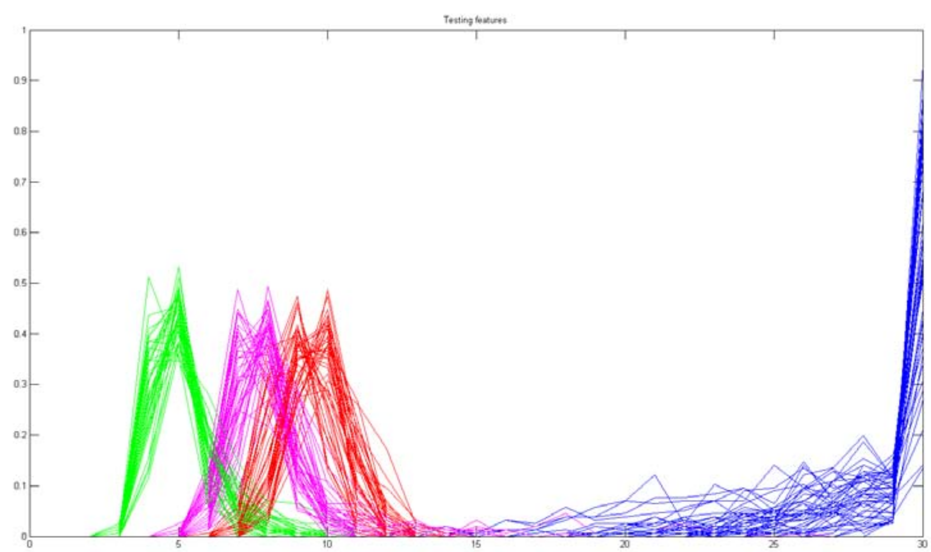
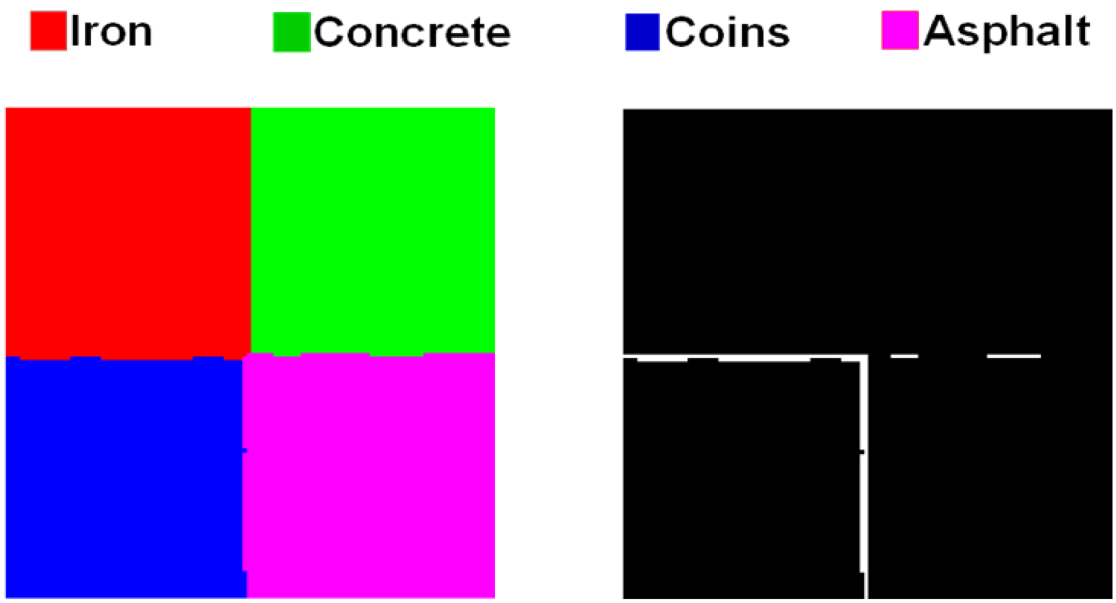


Figure 4–39: The histogram features computed for the unlabeled samples shown in Figure 4–2(b).



(a) (b)

Figure 4–40: (a) Thematic map produced by the semi-supervised SVM classifier using spectral difference histograms, 25 labeled 7×7 windows and smoothing with 11×11 windows for the 40 spectral bands when Gaussian noise is added. (b) Misclassified pixels of (a), represented in white with respect to the ground truth.

CHAPTER 5
CONCLUSIONS AND FUTURE WORK

This chapter summarizes the conclusions obtained from experimental results, the limitations of the SDH and semi-supervised SVM public implementation, and also presents recommendations for future work.

5.1 Conclusions

In this research we developed a new feature extraction method that accounts both the spatial and spectral information. It extracts regional texture information by computing spectral difference histograms over window extents. This algorithm represents an important advantage to the feature extraction field because based on our literature review there is not another method that extract features using the SDHs.

A semi-supervised learning algorithm was used for classification. The classifier that we used is the semi-supervised SVM. The algorithm was tested with HSI obtained from the SOC-700 hyperspectral camera and from HYDICE sensor, and it was able to classify those with accuracies above 91%. The algorithm performs well when the spatial resolution is higher.

The SVD band subset selection method available in the HIAT was used to reduce the dimensions of the input HSI data. The data was reduced to the number of bands that achieved more than 92% of the data variance.

Different window sizes were tested to construct the SDHs under the assumption that all pixels in the window were from the same class. Smaller window sizes were not sufficient to characterize the texture using the SDH. Therefore, larger window sizes were necessary to compute the SDH, but the window could not be too large considering that the spatial information is found in the nearest neighbors. The experimental results show that the misclassified pixels were found principally in the class boundaries. This occurs when the pixels in the window are not highly correlated, or in other words when there are pixels of more than one class in the window. Hence, there is a trade-off between the window size and the size of the objects that we want to study.

The proposed algorithm was tested under a noisy environment. The obtained results exhibited no change in the accuracy measured by the unlabeled samples, indicating that it is not sensitive to small levels of Gaussian noise for the HSI that we used, and that it works well when we have variability in the data.

Our method improved or is comparable with the PCA-FE and none-FE methods with overall accuracies grater than 91%. PCA-FE and None-FE methods failed to discriminate between Iron and Asphalt (for dataset 1) due to the similarity of their spectral signatures. Considering the spatial information in the analysis help us to discriminate between these classes. The errors produced by our method were founded principally between the class boundaries.

5.2 Limitations

The computation of the SDHs is limited to the following conditions:

- Data with high spatial resolution.
- Classes may contain more pixels than those contained inside the window.
- The assumption that all pixels in the window belongs to the same class.

The semi-supervised SVM public implementation used in our work has the following limitations:

- Not designed to work with more than a few thousand points (labeled+unlabeled).

5.3 Future Work

As future work we want to:

- Test the algorithm using data from other sensors with high spatial resolution.
- Experiment different pre-processing methods to improve the accuracy of the classifier.
- Improve the speed of the algorithm.
- Implement the algorithm in C++ software, parallel and distributed hardware.

REFERENCE LIST

[1] Manian V., Jiménez L.O.,"Land Cover and benthic habitat classification using texture features from hyperspectral and multispectral images", *SPIE Journal of Electronic Imaging*, vol. 16, no. 2,pp., June 2007.

[2] Rosario S., "Iterative Algorithms for Abundance Estimation on Unmixing of Hyperspectral Imaging". Thesis (M.S.) University of Puerto Rico, Mayagez Campus,pp.1-5, 2004.

[3] Gowda P.H., "Evaluation Of The Hydice Sensor For Mapping Floodplain Vegetation In Navigation Pool 13, Upper Mississippi River System", May 1997, http://www.umesc.usgs.gov/reports_publications/psrs/psr_1997_05.html

[4] Rellier G., Descombes X., Falzon F. and Zerubia J., "Texture Feature Analysis Using a Gauss-Markov Model in Hyperspectral Image Classification",*IEEE Transactions on Geoscience and Remote Sensing*, vol.42, no.7, pp. 1543-1551, July 2004.

[5] Hong P.S., Kaplan L.M. and Smith M.J.T., "Hyperspectral Image Segmentation using Filter Banks for Texture Augmentation",*IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data*, pp. 254-258, October 27-28, 2003.

[6] Jensen J.R.,"Remote Sensing of the Environment", Pearson Prentice Hall, Second Edition, 2007.

[7] MicroImages Inc., "Introduction to Remote Sensing of the Environment", 2006. http://www.microimages.com/getstart/pdf/introrse.pdf

[8] Natural Resources Cananda, "Tutorial: Fundamentals of Remote Sensing", http://www.ccrs.nrcan.gc.ca/resource/tutor/fundam/chapter1/01_e.php

[9] Shaw G.A., Burke H.K., "Spectral Imaging for Remote Sensing", *Lincoln Laboratory Journal*, vol.14, no.1, 2003.

[10] Plaza A., Martinez P., Plaza J. Perez R., "Dimensionality Reduction and Classification Hyperspectral Image Data Using Sequences Extended Morphological Transformations". *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, pp.466-479, March 2005.

[11] Vélez M. and Jiménez L.O., "Subset Selection Analysis for the Reduction of Hyperspectral Imagery",*Proceedings IEEE Intl. Geosciences and Remote Sensing Symposium IGARSS '98, vol. 3, pp. 1577-1581, July 1998.*

[12] Arzuaga-Cruz E., Jiménez-Rodriguez L.O., Vélez-Reyes M., "Unsupervised Feature Extraction and Band Subset Selection techniques based on Relative Entropy Criteriafor Hyperspectral data Analysis" *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery IX*, Proceedings of SPIE, vol.5093, pp.462-473, 2003.

[13] Camps-Valls G. and Bruzzone L., "Kernel-Based Methods for Hyperspectral Image Classification". *IEEE Transactions on Geoscience and Remote Sensing*, vol.43, no.6, pp. 1351-1362, June 2005.

[14] Schowengerdt R.A, "Remote Sensing - Models and Methods for Image Processing", Academic Press, Third Edition, pp. 193-198, 2007.

[15] Hsu S.M. and Burke H.K.,"Multisensor Fusion with Hyperspectral Imaging Data: Detection and Classification",*The Lincoln Laboratory Journal*, vol. 14, no. 1, pp.149, 2003.

[16] Harsanyi J.C. and Chang C., "Hyperspectral Image Classification and Dimensionality Reduction: An Orthogonal Subspace Projection Approach". *IEEE*

Transactions on Geoscience and Remote Sensing, vol.32, no.4, pp. 779-785, July 1994.

[17] Kaewpijit S., Le Moigne J. and El-Ghazawi T., "Automatic Reduction of Hyperspectral Imagery Using Wavelet SpectralAnalysis". *IEEE Transactions on Geoscience and Remote Sensing*, vol.41, no.4, pp. 863-871, April 2003.

[18] www.yale.edu/ceo/Projects/swap/landcover/Unsupervised_classification.htm

[19] Wikimedia Foundation Inc., "Support Vector Machines", 2008, http://en.wikipedia.org/wiki/Support_vector_machine

[20] Shah C.A.,Watanachaturaporn, P., Varshney, P.K., Arora, M.K., "Some Recent Results on Hyperspectral Image Classification", *IEEE Workshop on Advances in Techniques for Analysis of Remotely Sensed Data*, pp.346-353, 27-28 Oct. 2003.

[21] Eråstö P.,"Support Vector Machines - Background and Practice", 2001. http://ethesis.helsinki.fi/julkaisut/mat/rolfn/lt/erasto/supportv.pdf

[22] Bennett K. and Demiriz A., "Semi-Supervised Support Vector Machines", *Preceedings of Neural Information Processing Systems*, Denver, 1998.

[23] Chapelle O., Schlkopf B. and Zien A., "Semi Supervised Learning", MIT Press, Cambridge MA., 2006.

[24] Vapnik V.., "Estimation Dependences Based on Empirical Data", Springer-Verlag, 1982.

[25] Cohen I., Cozman F.G., Sebe N., Cirelo M.C. and Huang T.S., "Supervised Learning of Classifiers: Theory, Algorithms, and Their Application to Human-Computer Interaction", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 12, December 2004.

[26] Chapelle O. and Zien A., "Semi-Supervised Classification by Low Density Separation". *Proc. of the Tenth International Workshop on Artificial Intelligence*

and Statistics, pp. 57-64, 2005.

[27] Chapelle O., Chi M. and Zien A., "A Continuation Method for Semi-Supervised SVMs", *Proceedings of the 23rd International Conference on Machine Learning*, 2006.

[28] Camps-Valls G., Bandos T.V., and Zhou D. ,"Semi-Supervised Graph-Based Hyperspectral Image Classification", *IEEE Transactions on Geoscience and Remote Sensing*, vol.45, no.10, October 2007.

[29] Chang P. and Krumm J., "Object Recognition with Color Cooccurence Histograms", *IEEE Conference on Computer Vision and Pattern Recognition*, Fort Collins, CO, June 23-25, 1999.

[30] Crandall D. and Luo J., "Robust Color Object Detection using Spatial-Color Joint Probability Functions", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004.

[31] Valkealahti K. and Oja E., "Reduced Multidimensional Co-occurrence Histograms in Texture Classifications", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, January 1998.

[32] Bello F. and Kitney R., "Co-Occurrence-Based Texture Analysis Using Irregular Tessellations", *IEEE Proceedings of ICPR,* pp. 780-784, 1996.

[33] Mercier G. and Lennon M., "On the Characterization of Hyperspectral Texture",*IEEE Geoscience and Remote Sensing Symposium*, pp.2584-2586, 2002.

[34] Benediktsson J.A., Palmason J.A. and Svinsson J.R., "Classification of Hyperspectral Data from Urban Areas Bases on Extended Morphological Profiles", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, March 2005.

[35] Plaza A., Martnez P., Prez R. and Plaza J., "Spatial/Spectral Endmember Extraction by Multidimensional Morphological Operations", *IEEE Transactions*

on Geoscience and Remote Sensing, vol. 40, no. 9, September 2002.

[36] Liu X. and Wang D.,"Image and Texture Segmentation Using Local Spectral Histograms", *IEEE Transactions on Image Processing*, vol. 15, no. 10, October 2006.

[37] http://aviris.jpl.nasa.gov/

[38] Morillo S., "A Comparison of Resolution Enhancement Methods as Pre-processing for Classification of Hyperspectral Images ". Thesis (M.S.) University of Puerto Rico, Mayagez Campus, 2004.

[39] Sawin S.,"Histograms and the Shape of Distributions", http://cs.fairfield.edu/~sawin/Stats/Notes/histogram.pdf.

[40] Shimazaki H. and Shinomoto S.,"A Method for Selecting Bin Size of a Time Histogram", *Neural Computation***19**, **1503-1527**, 2007.

[41] D. A. Landgrebe, Signal Theory Methods in Multispectral Remote Sensing. Hoboken, NJ: Wiley, (2003)

[42] http://ccma.nos.noaa.gov/products/biogeography/benthic/htm/maps/pr_158.pdf