Theses and Dissertations

1-1-2012

Exploiting Spatial and Spectral Information in Hyperdimensional Imagery

Matthew Allen Lee

Follow this and additional works at: https://scholarsjunction.msstate.edu/td

Recommended Citation

Exploiting spatial and spectral information in hyperdimensional imagery

By

Matthew Allen Lee

A Dissertation
Submitted to the Faculty of
Mississippi State University
in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy
in Computer Engineering
in the Department of Electrical and Computer Engineering

Mississippi State, Mississippi

August 2012

Exploiting spatial and spectral information in hyperdimensional imagery

By

Matthew Allen Lee

Approved:

_____
Lori M. Bruce
Associate Dean and Professor of
Electrical and Computer Engineering
(Director of Dissertation)

_____
Saurabh Prasad
Assistant Professor of Electrical and
Computer Engineering
University of Houston
(Committee Member)

_____
Susan Bridges
Research Professor of Computer Science
and Engineering
(Committee Member)

_____
James Aanstoos
Associate Research Professor of
Geosystems Research Institute
(Committee Member)

_____
James E. Fowler
Professor of Electrical and Computer
Engineering
(Graduate Coordinator)

_____
Sarah A. Rajala
Dean
Bagley College of Engineering

Name: Matthew Allen Lee

Date of Degree: August 11, 2012

Institution: Mississippi State University

Major Field: Computer Engineering

Director of Dissertation: Dr. Lori Mann Bruce

Title of Study: Exploiting spatial and spectral information in hyperdimensional
                   imagery

Pages in Study: 92

Candidate for Degree of Doctor of Philosophy

In this dissertation, new digital image processing methods for hyperdimensional imagery are developed and experimentally tested on remotely sensed Earth observations and medical imagery. The high dimensionality of the imagery is either inherent due to the type of measurements forming the image, as with imagery obtained with hyperspectral sensors, or the result of preprocessing and feature extraction, as with synthetic aperture radar imagery and digital mammography.

In the first study, two omni-directional adaptations of gray level co-occurrence matrix analysis are developed and experimentally evaluated. The adaptations are based on a previously developed rubber band straightening transform that has been used for analysis of segmented masses in digital mammograms. The new methods are beneficial because they can be applied to imagery where the region of interest is either poorly segmented or not segmented. The methods are based on the concept of extracting circular windows s around each pixel in the image which are radially resampled to derive rectangular images. The images derived from the resampling are then suitable for standard GLCM techniques. The methods were applied to both remotely sensed synthetic

aperture radar imagery, for the purpose of automated detection of landslides on earthen levees, and to digital mammograms, for the purpose of automated classification of masses as either benign or malignant. Experimental results show the newly developed methods to be valuable for texture feature extraction and classification of un-segmented objects.

In the second study, a new technique of using spatial information in spectral band grouping for remotely sensed hyperspectral imagery is developed and experimentally tested. The technique involves clustering the spectral bands based on similarity of spatial features extracted from each band. The newly developed technique is evaluated in automated classification systems that utilize a single classifier and systems that utilize multiple classifiers combined with decision fusion. The systems are experimentally tested on remotely sensed imagery for agricultural applications. The spatial-spectral band grouping approach is compared to uniform band windowing and spectral only band grouping. The results show that the spatial-spectral band grouping method significantly outperforms both of the comparison methods, particularly when using multiple classifiers with decision fusion.

DEDICATION

I dedicate this dissertation to my brother Mark Edward Lee, who died in his sleep

shortly before I finished my master's degree in 2006 at age 21. "I shall go to him, but he

shall not return to me" – 2 Samuel 12:23.

ACKNOWLEDGEMENTS

I also thank Mississippi State University and all the teachers who instructed me during my many years there.

Finally, thanks be to God for giving me the strength to accomplish this task.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 Background

A hyperdimensional image (HDI) is an *n*-dimensional dataset where two dimensions, usually referred to as [*x, y*], represent a physical space. Each [*x, y*] element represents a spatial location, while the remaining *n*-2 dimensions represent phenomena occurring per spatial location. For example, an HDI can be produced in the ways listed below.

- An HDI can be produced by a sensor that simultaneously records a large number of measurements per spatial area, producing a large number of sub-images of the same area. For example: a hyperspectral camera measures optical radiance, resulting in a three dimensional image cube where [*x,y*] varies across physical space and [*z*] varies across the electromagnetic spectrum.

- An HDI can be composed of a large number images of the same scene acquired by the same sensor at different times.

- An HDI can be created by extracting a large number of spatially varying features from a smaller set of images of the same scene.

- An HDI can also result from multisource data, for example multiple sensors recording measurements of the same scene.

Per-pixel processing is quite common for HDI. For example with hyperspectral imagery, many image processing techniques have been developed that exploit the spectral dimension on a per-pixel basis. The focus of this dissertation is on algorithms that use spatial information to aid in the processing of HDIs. With synthetic aperture radar imagery, this dissertation introduces image analysis techniques that produce HDIs. With hyperspectral imagery for example, this dissertation introduces image analysis techniques that exploit both spectral and spatial information simultaneously in the DH.

1.2   Hyperspectral Imagery

One common source of an HDI is hyperspectral imagers. Thus it is useful to study hyperspectral techniques since many of the problems experienced are mirrored in other HDIs. Hyperspectral data first became available to the public for scientific use in the 1990's via the National Aeronautics and Space Administration (NASA) with their Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor operated at the Jet Propulsion Laboratory [1] and Compact Airborne Spectrographic Imager (CASI), manufactured by Itres Research Ltd. [2]. Hyperspectral imagery is becoming more commonly used in remote sensing applications. Hyperspectral sensors are typically passive instruments used to obtain measurements of electromagnetic radiation across a range of the spectrum for a given on-ground scene. The hyperspectral sensor records 100's to 1000's of finely spaced, narrow frequency bands. This information can then be used to infer the chemical composition of the visible surfaces in the scene similar to the way lab spectrometers are used to measure the chemical composition of a sample. This

sort of information is very useful for target classification in many applications. A few applications in which hyperspectral imagery is useful include:

- ecology, as in detection of invasive plant species or measuring and characterizing biomes;

- agriculture, as in detection of plant stress caused by moisture, nitrogen deficiency, plant disease, or crop pests;

- archeology, as in differentiating manmade and natural features, which aids in identification of potential archeological sites; and

- land usage, as in determining the extent of agriculture, urbanization, and natural land cover.

Hyperspectral sensors, however, have some drawbacks, such as over-dimensionality and noise. Because of the high dimensionality in hyperspectral data, enormous bandwidth is required for transmission, large space is required for storage, and a large number of ground-truth samples are required for supervised target recognition and classification. Noise degrades the ability to measure the electromagnetic spectrum radiating from objects in the scene, and the noise primarily is caused by sensor electronics and the Earth's atmospheric effects. The hyperspectral signals can contain additional noise due to the sensor's low spectral resolution, low spatial resolution, and/or limited dwell time. Typically, noise is modeled using probability density functions (PDFs) [3]. Atmospheric noise typically results from the light being absorbed and diffused as it passes through the atmosphere. These effects are typically frequency dependent and reduce the signal strength as a function of frequency. Atmospheric noise is made more difficult to quantify and correct because the atmosphere is not homogeneous,

3

varying with altitude, location on the Earth, and time. Atmospheric correction techniques are available for the purpose of removing atmospheric noise. These techniques require certain atmospheric measurements that are input in to the process as parameters. However, when these measurements are not available or incorrect, the atmospheric correction may not eliminate the noise. Because of the noise in hyperspectral images, it can be extremely difficult to differentiate two targets that have similar spectrums using only the spectral information per pixel.

There are two common approaches to object classification in a hyperspectral image. The first approach is to use the spectral information to classify each pixel independently, i.e. per-pixel processing. The advantage of this approach is that is relatively simple and does not require objects to be segmented. However, when there are objects with similar spectral signatures, this method can fail and produce classification maps with significant salt and pepper noise. This is because of the noise in the hyperspectral image and overlap of spectral signature distributions in objects with similar chemical compositions. The second common technique is to segment objects in the hyperspectral scene and then classify the objects based on their spectral information. The advantage of this technique is that it can use the spectrum from all the pixels within an object plus any additional spatial features obtained from the object such as size, shape, texture, etc. Typically this approach eliminates the salt and pepper noise in the classification maps, but it has the possibility of misclassifying an entire object if the segmentation performance is poor.

Edge detection is one common and effective step in segmenting objects in images. Most edge detection algorithms key on discontinuities in the image to locate object edges.

Research [4] has shown that discontinuities in image brightness are likely to correlate with

- discontinuities in depth,

- discontinuities in surface orientation,

- changes in material properties, and

- variations in scene illumination.

Such discontinuities generally correspond to the boundaries of objects. Furthermore, it has been shown that in color images, image intensity (or brightness) typically accounts for 90% of the edges in a scene, which means 10% of edges can typically only be detected using the color information in the image[5]. In hyperspectral images, the percentage of edges that are missed by image intensity is likely higher because there are many more spectral bands, and therefore, a larger space in which the signal energy can be distributed. Edge information also enables many other algorithms to be utilized in image processing. According to [6], a few algorithms that use edge information include

- curve-based stereo vision,

- contour-based image compression,

- edge-based target recognition, and

- edge-based face detection.

While not all of these algorithms are necessarily applicable to the field of remote sensing, the techniques used in these algorithms could possibly be applied to problems in remote sensing. Within remote sensing, edge information has been used for

- morphological feature extraction,

- edge-based object segmentation,

5

- object-based classification, and

- detection of linear objects.

The problem with applying edge detection to hyperspectral images is that the definition of an edge in hyperspectral space is not intuitive. This is because discontinuities can have many different forms in a hyperspectral image, and noise can cause false edges to appear. Because of these difficulties, there have been few breakthroughs in hyperspectral edge detection.

1.3   Contributions of this Dissertation

This dissertation introduces two new image processing techniques that exploit spatial information in HDI analysis. Some of the experiments focus solely on hyperspectral image processing, but many of their conclusions can be applied to other types of HDIs. The primary contributions of this dissertation are as follows.

1) Develop an omni-directional texture analysis technique that can be applied to objects that are poorly segmented or not segmented and apply it to HDI.

    a) Develop a technique that extends the concept of the rubber band straightening transform and gray level co-occurrence matrix (GLCM) texture features, such that the new technique does not require object edge information.

    b) Implement the newly developed technique in software.

    c) Experimentally evaluate its efficacy on real-world HDI.

    d) Compare the newly developed method to standard texture analysis techniques.

6

2) Develop a hyperspectral band grouping technique that concurrently uses spatial and spectral information and apply it to HDI.

  a) Develop a technique that utilizes spatial information, such as edges or textures, to guide a spectral band grouper and then uses the spatial-spectral band groups to extract features for classification purposes.

  b) Implement the newly developed methods and incorporate them in to automated classification systems, both single classifier and multiple classifier systems, in software.

  c) Experimentally evaluate their efficacy on real-world HDI.

  d) Conduct sensitivity studies to determine the robustness of the algorithm to design parameters, such as selected spatial filters, number of band groups, etc.

  e) Compare the newly developed methods to standard spectral band grouping techniques that do not utilize spatial information.

1.4   References

[1]   *AVIRIS system specifications*. Available: http://aviris.jpl.nasa.gov/

[2]   *CASI 550 - VNIR Spectrographic Imaging System*. Available: http://www.itres.com/products/imagers/casi550/

[3]   R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2 ed. Upper Saddle River, New Jersey: Prentice Hall, 2002.

[4]   H. G. Barrow and J. M. Tenenbaum, "Interpreting line drawings as three-dimensional surfaces," *Artificial Intelligence,* vol. 17, pp. 75-116, 1981.

[5]   N. Lianqiang and L. Wenju, "Color Edge Detection Based on Direction Information Measure," *Proceedings of The Sixth World Congress on Intelligent Control and Automation (WCICA),* pp. 9533-9536, 2006.

[6]   B. Kurt and M. Gokmen, "Goal oriented edge detection," *Proceedings of 23rd International Symposium on Computer and Information Sciences (ISCIS '08)*, pp. 1-5, 2008.

CHAPTER II

CURRENT STATE OF KNOWLEDGE

2.1 Edge Detection

A reoccurring problem in hyperspectral edge detection is the difficulty of determining a mathematically precise definition of an edge. Instinctively, it seems to be globally understood that an edge is a boundary between two regions in the image and is usually characterized by a discontinuity. However, a widely accepted measurement of a discontinuity in a hyperspectral space does not exist. This particular problem is not present in gray level edge detection because it is defined as locations where the first derivative of the image forms coherent ridges [5]. This solution is not easily applied to a hyperspectral image because only partial derivatives can be computed in the hyperspectral space. However, one can use this definition and common gray level edge detection algorithms to serve as a foundation for algorithms that operate in higher dimensionality space.

2.1.1 Conventional Gray Level Edge Detection

Gray level edge detection is generally well understood. It typically involves four steps, which are: preprocessing, gradient detection, thresholding, and post-processing.

In preprocessing of gray scale images, the goal is to denoise the image. Such noise usually comes from the sensor that collected the image. Types of noise that

9

typically occurs in images are white (Gaussian) noise, salt and pepper noise, and blurring. White noise is generally dealt with by using a low pass filter. The filter could be a neighborhood averaging filter, median filter, or a mode filter [1]. Although unnecessary in gray scale, in images of higher dimensionality, it might be useful to reduce the dimensionality of the image in the preprocessing step.

In gradient detection, the objective is to measure how much the image is changing in a region around each pixel. Since the region around pixels located on edges changes more, the measured value will be higher for these pixels. In some edge detection systems, it is also necessary to know the direction in which the change is increasing. The measurement of change and the direction of change are referred to as the gradient magnitude and gradient angle, respectively. Generally these measurements are obtained by convolving a highpass filter with the image. There are two main types of filters used for edge detection: directional and non-directional. Directional filters are more common because a pair of such filters that measure the gradient in perpendicular directions can be used to estimate the gradient angle by computing the inverse tangent. In addition, the gradient magnitude can be measured by summing the result of both filters. Examples of such filters are Roberts [6], Prewitt [7], and Sobel [8]. Non-directional edge detectors are faster since they require one convolution pass because they can detect edges in any direction equally well. However, such filters may be too sensitive to noise, and, as indicated, provide no directional information. Common types of non-directional filters are Laplacian filters [1].

The next step in gray level edge detection is thresholding the gradient magnitude information. While thresholding, which converts a continuous valued image into a binary

10

image by assigning values less than a threshold to 0 and values greater than the threshold to 1, is a simple process, the difficulty is in choosing the threshold value. The simplest way to choose a threshold is to manually select a value. This method is simple and can be very effective when objects in the image are smooth and distinct from each other. However, if these conditions are not met, the value must be tuned, and it may not be possible to choose a value that works for images taken under different conditions. Often the process of selecting a threshold is automated. Barrow and Tenenbaum [1] describe a technique where a gray level histogram is used. In this method, the value of the threshold is determined by the valley in the histogram between the non-edge pixels and edge pixels. In some cases, a single global threshold might not work because the brightness and contrast may not be constant throughout the image. In this case, adaptive thresholding may be used. One approach to adaptive thresholding is breaking the image into tiles or windowing the image, and then using an automated threshold selection technique [1]. In some cases the noise in the gradient information may be too great to get coherent edges even with adaptive thresholding. In such situations, local thresholding can be used to complete the edges. One such techniques is to use a lower threshold for pixels that are near edge pixels and have similar gradient angles to the edge pixels [1]. This is similar to region growing segmentation techniques, but in this case linear regions, which likely correspond to edges, are grown. Then there are numerous ways to choose this second threshold. Most of these are variations on the methods of choosing the main threshold.

In the post-processing step, the binary image from the thresholding step is refined. Generally small gaps are filled in and unconnected edges, which were likely generated by

noise in the original image, are removed. The most common techniques for doing this are the Hough transform and morphological processing [1].

There are many papers written on edge detection using one channel, and it would be impossible to reference them all in a single document. However, the following is a list of a few relevant papers, and a brief description of what is significant in each paper.

- In [9], filtering was done using wavelets by combining the edge detection results from several reconstruction scales.

- In [10], a morphological technique that uses multiple directional structural elements and fuses the results is described.

- In [11], a neural network was used to identify edges instead of the differential operator and thresholding.

- In [4], a support vector machine was used to classify the pixels as edge or non-edge.

- In [12], the results were inferior to several other techniques, but the authors used empirical mode decomposition instead of gradient operators, and then thresholded the first intrinsic mode function.

- In [13], the decisions of multiple edge detectors were fused to get a final edge map.

- In [14], adaptive thresholding was based on an edge reliability metric.

- In [15], the authors use cellular neural networks to filter noise, quantize the image, and detect edges in remote sensed images. There is also a dilation and erosion step that does not use a cellular neural network.

12

- In [16], an adaptive threshold technique based on watersheds is introduced.

2.1.2   Edge Detection with More than One Channel

When there is more than one channel, edge detection becomes more difficult. The additional bands make a mathematical definition of an edge more ambiguous, especially as more and more channels are added to the image cube. One common way of dealing with this problem is to use dimensionality reduction to get a set of fewer features than the number of channels. There are many ways to do this. The drawback of dimensionality reduction is information loss as the number of available features is decreased. Thus, there is a tradeoff between having a low number of features and loss of information. Additionally, unless the number of features is reduced to one, standard gray level techniques still cannot be used. Thus, we still have multiple channels to deal with. One obvious way to handle multiple channels is to use gray level edge detection on each channel, and then combine the results. There are many possible ways to do this. However, the major drawback is that they often have lower signal to noise ratios than methods that consider all the channels concurrently. Perhaps the simplest way to combine the gradient information is to average the gradients for all the bands to get one gradient image. This can be done by averaging the horizontal gradients for all the channels, and then averaging the vertical gradients for all the channels. Then the averaged vertical and horizontal gradients can be used to compute a gradient intensity and angle similar to the way it is done in gray level edge detection. A second technique is to compute the gradient intensity and angle for each channel, and then averaging the result for all the channels. However, this method assumes that the gradients for all channels represent edge

13

information for the same objects. This may break down when an edge is invisible in some bands and visible in other bands. There are other ways to combine the gradients, such as adding, multiplying, or even more complicated techniques, but these are generally applied similarly to one of the two averaging techniques. Barrow and Tenenbaum [1] describe a technique for color edge detection, where the gradient magnitude is computed by the equation:

$$g_{xy} = \frac{\partial R}{\partial x}\frac{\partial R}{\partial y} + \frac{\partial G}{\partial x}\frac{\partial G}{\partial y} + \frac{\partial B}{\partial x}\frac{\partial B}{\partial y}. \qquad 2.1$$

The gradient angle is computed by:

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{2g_{xy}}{g_{xx}-g_{yy}}\right), \qquad 2.2$$

where

$$g_{xx} = \left|\frac{\partial R}{\partial x}\right|^2 + \left|\frac{\partial G}{\partial x}\right|^2 + \left|\frac{\partial B}{\partial x}\right|^2, \qquad 2.3$$

and

$$g_{yy} = \left|\frac{\partial R}{\partial y}\right|^2 + \left|\frac{\partial G}{\partial y}\right|^2 + \left|\frac{\partial B}{\partial y}\right|^2. \qquad 2.4$$

Though this technique is intended for color images, it is possible to extend this technique to any number of channels. Another strategy is to use all the channels concurrently. One way of doing this is to treat each pixel as a vector and compute the Euclidean distance between symmetric pairs of pixels around the pixel that the gradient information is being computed for. This will indicate the magnitude for g_{xx} and g_{yy}, but it won't indicate the sign of the gradient. However, the sign can be assigned to the horizontal and vertical gradients using some technique. One way to do this is to compare

14

the dot products between each side and the center pixel, then choose a sign based on which side has the largest values. Once the gradient angles and intensities have been determined, typically gray level techniques of thresholding and post processing are used.

Though there are not as many papers on edge detection using multiple channels, here are a few examples.

- In [17], manifold techniques were used to do dimensionality reduction on a hyperspectral image.

- In [3], the edges were detected in HSV space by averaging gradient operators of two scales with varying direction.

- In [18], fuzzy color membership of the pixels was computed in HLS space, and a measure of color difference called spread index was computed then thresholded.

2.2   Hyperspectral Segmentation Techniques

Segmentation does not require edge detection. In fact most hyperspectral segmentation algorithms do not use any edge detection. Segmentation algorithms that do not use edge detection typically fall into one of three categories. These categories are region growing, thresholding, and pixel-by-pixel labeling (possibly with post-processing).

In region growing, seed points are determined using some method, then a region is grown from the seed points by repeatedly adding the pixels along a border of the region as long as they conform to a growth criteria. The seed points could be determined randomly, or by some other method, such as selecting pixels that closely match a prototypical end member signature. Selecting seed points randomly implies the

15

segmentation is unsupervised, but the other method described here could be supervised or unsupervised depending on how the end members are chosen. The growth criteria can be very complicated, but simple ones could consider how similar the pixels on the border are to the seed point, or mean signature of the region.

In the thresholding technique, regions are segmented out by thresholding some scalar (or possibly vector) quantity that is computed or extracted from the spectral signatures in the image. One common computed quantity used is Normalized Difference Vegetation Index (NDVI) [19], which could be used to segment vegetation. Sometimes, however, a subset of spectral bands could be thresholded also. Another algorithm that could be used in thresholding is the spectral angle mapper, which computes the angle between the pixels and an end member signature. The advantage of using the spectral angle mapper technique is that it is not affected by changes in light intensity.

The third technique is to use the spectral information to label each pixel independently. This technique can produce excellent results under the right conditions, but if the distributions of the spectral signatures of multiple classes overlap, the resulting segmentations can have significant salt-and-pepper noise. In some cases, the researchers will use a post processing technique, such as mode filtering, to remove the salt-and-pepper noise.

A few relevant hyperspectral segmentation papers are listed below.

- In [20], the authors used a spectral clustering algorithm to classify each pixel. The classification results are then considered the segmentation result.

16

- In [21], the paper describes a technique where spectral classification and spatial segmentation were recursively used to augment each other.

- The authors of [22] use post processing of a classification map generated by a semi-supervised classification technique.

- In [23], a region growing method where seed points were chosen using a spectral angle mapper with end members chosen manually. They also use a best band analysis approach that selects the best bands based on class variation.

2.3   Feature Selection

Feature selection techniques always fall into two categories: optimal and sub-optimal heuristic. In the optimal category, there are exhaustive search and the branch-and-bound algorithm [24]. In exhaustive search, the algorithm tests every possible combination of features and then selects the best combination based on some metric. The only variable of consequence in this technique is the metric. The branch-and-bound algorithm does not have to test every possible solution, but it requires that the accuracy must monotonically increase as the number of features increases. This is a situation that most often does not occur in large feature spaces because of the "curse of dimensionality." Eventually, adding more features will degrade the accuracy. There are a great number of metrics that can be used, but that is not the focus of this dissertation, and thus it will be assumed that there is some metric that can be optimized by selecting a set of features, and this metric correlates positively to classification accuracy in the final classification system (the metric could be classification accuracy itself). The advantage of this technique is that it always finds the best combination of features that optimizes the

17

metric for any problem. The reason exhaustive search is not always used is that it cannot be run in polynomial time, and thus is a non-polynomial (NP) time algorithm. In fact, the algorithm is *O(n!)*, which means that as *n* (the number of features) increases, the time it takes to run the algorithm quickly increases to the point that it is not feasible to use.

In hyperspectral images and other hyperdimensional feature spaces, the number of features is too large for an exhaustive search to be used. Commonly used non-exhaustive techniques do run in polynomial time, and are therefore more feasible. However, they do not guarantee that the best set of features will be selected. Usually, a good set of features will be chosen. The most common feature selection techniques are the following [25], [26] and [1].

- Greedy hill climbing

- Best first search

- Simulated annealing

- Genetic algorithms

- Forward selection with backwards rejection

There are several ways to improve feature selection algorithms, which has motivated many researchers to study many different algorithms. As a result, many less common algorithms have been tried. Most of these never get much attention beyond an introduction paper. A list of such algorithms is below.

- In[27], a graph technique is used for feature selection.

- Clustering is an approach that has seen sporadic use. In[28], the authors cluster features based on their values, and select a representative feature from each cluster.

18

- In[29], a biologically inspired algorithm based on the immune system was used for feature selection.

- In [30], an algorithm based on ant colony optimization was used to select features.

- In [31], an algorithm that chooses features based on rough set approximations was used to select features.

- In [32], the authors select wavelet features using a forward selection technique that repeatedly chooses the band with the least mutual information with the selected subset.

- In [33], another forward selection algorithm was used to select features. This time the information-theoretic optimality criterion was used to choose which feature to add to the selected subset.

- In [34], the authors employ a simple manual feature selection technique by using a square window to select features from a 2D FFT feature space.

2.4   Cellular Automata

The human brain is exceptionally complex, but on a small scale, it acts like a cellular automaton (CA) in that each neuron has a set of neighboring neurons, and the state of a neuron is influenced by the states of its neighboring neurons. Furthermore, string theory, one of the leading candidates for the as yet unformulated "Theory of Everything," proposes that the whole universe is in reality a discrete 10 to 11 dimensional space and what we experience is based on the state of vibrating strands of energy at every point [35]. If this is the case, then the whole universe is one massive CA. As far as a

19

computing model, CAs have fallen in and out of popularity since their inception by Von Neumann in the 1940s [36] because they can be a very powerful computing model. CAs derive their power from the fact that complex emergent behavior can emerge from the few simple rules each cell executes. This characteristic has ensured that CAs have received some attention in many different fields, and one day CAs may become the dominant computing model. However, they are still at the fringe of computing because they do not run efficiently on current general purpose computer, and it is prohibitively expensive to design new hardware for CAs that can do everything that a general purpose computer can do.

2.5   Clustering

Clustering is a technique that is used in of different fields as a tool for a vast number of problems. The two most common techniques are nearest neighbor (or agglomerative) clustering and K-means clustering [37]. However there are a number of variations of the two. Initially in nearest neighbor clustering, a threshold distance that represents the minimum distance between clusters is input, but the user does not know how many clusters will result. In K-means, the number of clusters is input, but the user does not know what the minimum distance between clusters will be. In general, K-means is a faster algorithm, but it does not guarantee the same results every time when the same input is given as nearest neighbor does. Also, in K-means, it is difficult to determine the best choice for the number of clusters, while in nearest neighbor, it is difficult to pick the minimum threshold. There are several techniques for making these choices, which have resulted in variations of these base algorithms. One variation of K-means is fuzzy c-means clustering.

20

2.6 Texture Analysis

Texture analysis is a very common technique in image processing. It is used in just about every application where image processing is used. The main goal of texture analysis is to measure some feature of the texture in the image or part of the image. There are a large number of ways to do this. One of the most common techniques is to use gray level co-occurrence matrices (GLCM). This technique is favored so much because it has the ability distinguish many types of textures. The basic strategy in GLCM is to create a matrix containing all the gray level combinations within the image or ROI with all pairs of pixels that have the same relative position. However, once the co-occurrence matrix is computed, there are a number of features that can be computed from the co-occurrence matrix. In "Texture Features for Image Classification," the paper that introduced the GLCM technique, the authors presented equations for calculating 14 different features [38]. These features are:

1. Angular Second Moment

2. Contrast

3. Correlation

4. Sum of Squares: Variance

5. Inverse Difference Moment

6. Sum Average

7. Sum Variance

8. Sum Entropy

9. Entropy

10. Difference Variance

11. Difference Entropy

12/13. Information Measures of Correlation (2 formulas were presented)

14. Maximal Correlation Coefficient

Since this introductory article, new features have been added. In Digital Image Processing [5], two new features are defined, called maximum probability and element difference moment of order k. Also, angular second moment is renamed uniformity, and inverse difference moment is generalized. The GLCM technique has been used in a great number of applications. A few examples of papers that utilize gray level co-occurrence matrices are listed below.

- "A Max-Min Measure for Image Texture Analysis" [39] uses the relative frequency of local extremes as the principal measure.

- "Texture Segmentation Using Multilayered Back Propagation" [40]uses co-occurrence matrices as inputs into a back propagation artificial neural network.

- "Unsupervised Image Segmentation Based on a Self-Organizing Feature Map and a Texture Measure" [41] uses co-occurrence matrices as inputs into a self-organizing map.

- "Classification of Microcalcifications Using a Multichannel Filtering Approach" [42] uses the output of high pass and low pass filters as input to GLCM.

22

- "Textural Features Corresponding to Textural Properties" [43] introduces a technique called Neighborhood Gray Tone Difference Matrix (NGTDM). From this NGTDM, they present functions for computing coarseness, contrast, busyness, complexity, and texture strength.

- In "A Comparative Study of Texture Measures for Terrain Classification," [44] the authors compare several different types of features. These types of feature are Fourier power spectrum, second-order gray level statistics, gray level difference statistics, and gray level run length statistics.

- "Texture Features for Classification of Ultrasonic Liver Images" [45] uses a few common texture features, but also introduces features based on multi-resolution analysis and a Brownian motion model, which assumes the texture is the result of a random walk.

- "Generalized gray level dependence method for prostate cancer classification" [46] extends the gray level dependence (AKA. GLCM) method to work with multispectral data.

- "Texture Image Segmentation Based on Gaussian Mixture Models and Gray Level Co-occurrence Matrix" [47] introduces a texture segmentation technique that uses Gaussian mixture models and GLCM.

- "Prediction of Cirrhosis Based on Singular Value Decomposition of Gray Level Co-occurrence Matrix and a Neural Network Classifier" [48] uses two approaches to predict Cirrhosis based on ultrasounds. The first uses standard GLCM features, while the second gets the features from the co-

23

occurrence matrix by singular value decomposition of the co-occurrence matrix itself. Both methods use artificial neural networks for classification.

- "Chinese sign language recognition based on gray-level co-occurrence matrix and other multi-features fusion" [49] uses several common GLCM features extracted for 0, 45, 90, and 135 degree angles.

- "A New Method for Iris Recognition using Gray-Level Coccurence Matrix" [50] segments a donut shaped region from the iris and straightens it out using a technique identical to Rubber Band Straightening Transform. Then standard GLCM features are used to identify the iris.

- "Segmentation of Blood Vessels in Retinal Images Using 2-D Entropies of Gray Level-Gradient Co-occurrence Matrix" [51] computes co-occurrence matrices that combine the gray level of the image and the gradient level of the image by using gray level as the row index and gradient level as the column index into the co-occurrence matrix.

- "Texture retrieval using grey-level co-occurrence matrix for Ikonos panchromatic images of earthquake in Java 2006" [52] extracts texture features from Ikonos panchromatic images of the earthquake that occurred in Java in 2006. The features they extract are contrast, entropy, homogeneity, and energy.

- "Gray-Level Co-occurrence Matrices as Features in Edge Enhanced Images" [53] solves a target recognition problem by first using a Sobel

24

edge operator to convert the image into a binary edge map with two gray levels, then GLCM features of the binary edge map are used to identify the targets.

- "Directional Analysis of Texture Images Using Gray Level Co-occurrence Matrix" [54] concedes that standard GLCM has a problem with its directional aspect. They attempt to find the best direction to analyze the texture using correlation.

- "A New Method of SAR Image Segmentation Based on the Gray Level Co-occurrence Martix and Fuzzy Neural Network" [55] uses GLCM and wavelet features as inputs into a fuzzy neural network to segment objects in synthetic aperture radar images.

Since the 1990s, wavelets have become a very popular tool in texture analysis, and there many papers on the subject. *Digital Image Processing* [5] goes into great detail in explaining them. Also, Fourier features are very common in texture analysis and have been around for a very long time. A description how to compute Fourier transforms is also in *Digital Image Processing*. When using Fourier transforms to extract features, a window has to be used so the features are extracted from a small ROI. "Texture feature based on local Fourier transform" [56] uses this technique to extract features from each pixel and its neighbors to analyze texture.

25

2.7 References

[1]   I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research,* vol. 3, pp. 1157-1182, 2003.

[2]   H. G. Barrow and J. M. Tenenbaum, "Interpreting line drawings as three-dimensional surfaces," *Artificial Intelligence,* vol. 17, pp. 75-116, 1981.

[3]   N. Lianqiang and L. Wenju, "Color Edge Detection Based on Direction Information Measure," *Proceedings of The Sixth World Congress on Intelligent Control and Automation (WCICA)*, pp. 9533-9536, 2006.

[4]   B. Kurt and M. Gokmen, "Goal oriented edge detection," *Proceedings of the 23rd International Symposium on Computer and Information Sciences (ISCIS '08)*, pp. 1-5, 2008.

[5]   R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, *2 ed*. Upper Saddle River, New Jersey: Prentice Hall, 2002.

[6]   L. G. Roberts, "Machine Perception of Three-Dimensional Solids," *Optical and Electro-Optimal Information Processing,* p. 35, 1965.

[7]   J. Prewitt, "Object Enhancement and Extraction," *Picture Process Psychopict,* pp. 75-149, 1970.

[8]   I. E. Sobel, "Camera models and machine perception," Stanford University, 1970.

[9]   J. I. Siddique and K. E. Barner, "Wavelet-based multiresolution edge detection utilizing gray level edge maps," *Proceedings of the International Conference on Image Processing, (ICIP 98),* vol.2., pp. 550-554, 1998.

[10]   Z. Yuqian, G. Weihua, and C. Zhencheng, "Edge Detection Based on Multi-Structure Elements Morphology," *Proceedings of The Sixth World Congress on Intelligent Control and Automation (WCICA),* pp. 9795-9798, 2006.

[11]   L. Weiqing, W. Chengbiao, W. Qun, and C. Guangshe, "An Edge Detection Method Based on Optimized BP Neural Network," *Proceedings of International Symposium on Information Science and Engineering,* pp. 40-44, 2008.

[12]   L. LingFei and P. ZiLiang, "An Edge Detection Algorithm of Image Based on Empirical Mode Decomposition," *Proceedings of Second International Symposium on Intelligent Information Technology Application,* pp. 128-132, 2008.

[13]   L. Jia and J. Xiaojun, "Edge detection based on decision-level information fusion and its application in hybrid image filtering*," Proceedings of International Conference on Image Processing,* vol. 1, pp. 251-254, 2004.

[14]   T. Sugiyama and K. Abe, "Edge detection method based on edge reliability with fixed thresholds: consideration of uniformity and gradation," *Proceedings of 15th International Conference on Pattern Recognition,* vol. 3, pp. 656-659, 2000.

[15]   X. Guo-bao, Z. Gui-yan, y. Lu, Y. Yi-xin, and S. Yu-li, "A CNN-based edge detection algorithm for remote sensing image*," Proceedngs of Control and Decision Conference,* pp. 2558-2561, 2008.

[16]   S. Beucher and C. Lantuejoul, "Use of watersheds in contour detection," presented at the *Proceeding sof International Workshop on Image Processing: Real-Time Edge and Motion Detection/Estimation*, 1979.

[17]   Z. Yuan, W. Bo, L. Deren, and L. Rongxing, "Edge detection on hyperspectral imagery via Manifold techniques," *Proceedings of Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing,* 2009.

[18]   E. Perumal, R. S. Rajesh, and P. Shanmugam, "Fuzzy-PL Transformation based Color Edge Detection," *Proceedings of 16th International Conference on Advanced Computing and Communications,* pp. 9-12, 2008.

[19]   J. R. G. Townshend, T. E. Goff, and C. J. Tucker, "Multitemporal Dimensionality of Images of Normalized Difference Vegetation Index at Continental Scales," *IEEE Transactions on Geoscience and Remote Sensing,* vol. GE-23, pp. 888-895, 1985.

[20]   N. Acito, G. Corsini, and M. Diani, "An unsupervised algorithm for hyperspectral image segmentation based on the Gaussian mixture model," *Proceedings of 2003 IEEE International Geoscience and Remote Sensing Symposium,* vol. 6, pp. 3745-3747, 2003.

[21]   N. Gorretta, J. M. Roger, G. Rabatel, V. Bellon-Maurel, C. Fiorio, and C. Lelong, "Hyperspectral image segmentation: The butterfly approach," *Proceedings of Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing,* 2009.

[22]   L. Jun, J. M. Bioucas-Dias, and A. Plaza, "Semi-supervised hyperspectral image segmentation," *Proceedings of Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing,* 2009.

[23] J. Ball and L. Bruce, "Level Set Hyperspectral Segmentation: Near-Optimal Speed Functions using Best Band Analysis and Scaled Spectral Angle Mapper," *Proceedings of IEEE International Geoscience and Remote Sensing Symposium*, pp. 2596-2600, 2006.

[24] P. M. Narendra and K. Fukunaga, "A Branch and Bound Algorithm for Feature Subset Selection," *IEEE Trans. Comput.,* vol. 26, pp. 917-922, 1977.

[25] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, Massachusetts: Addison-Wesley Publishing, 1989.

[26] P. N. Stuart J. Russell, *Artificial Intelligence A Modern Approach*. Upper Saddle River, New Jersey: Prentice Hall, 2003.

[27] C. Xi, F. Tao, H. Hong, and L. Deren, "Graph-Based Feature Selection for Object-Oriented Classification in VHR Airborne Imagery," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 49, pp. 353-365, 2011.

[28] J. Sen, Q. Yuntao, L. Jiming, L. Weixiang, and J. Zhen, "Feature extraction and selection hybrid algorithm for hyperspectral imagery classification," *Proceedings of IEEE International Geoscience and Remote Sensing Symposium,* pp. 72-75., 2010.

[29] C. Qiong, Z. Hong, L. Xingshan, and L. Zhaorui, "Image Feature Selection Method Based on Immune Encoding Mechanism," *Proceedings of Fourth International Conference on Natural Computation,* pp. 653-657, 2008.

[30] X. Lu and J. Li, "A remote sensing images feature selection approach based on Ant Colony Algorithm," *Proceedings of 2nd International Conference on Industrial Mechatronics and Automation (ICIMA),* pp. 354-357, 2010.

[31] P. Xin and Z. SuLi, "Remote sensing image feature selection based on α-torrent rough set theory," *Proceedings of Seventh International Conference on Fuzzy Systems and Knowledge Discovery (FSKD),* pp. 1034-1038, 2010.

[32] K. Huang and S. Aviyente, "Wavelet Feature Selection for Image Classification," *IEEE Transactions on Image Processing,* vol. 17, pp. 1709-1720, 2008.

[33] M. Vasconcelos and N. Vasconcelos, "Natural Image Statistics and Low-Complexity Feature Selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 31, pp. 228-244, 2009.

[34] G. Xingbin and L. Yongtan, "Feature extraction and feature selection of microwave scattering images," *Proceedings of the IEEE 1994 National Aerospace and Electronics Conference,* vol. 1, pp. 57-61, 1994.

[35]   B. Greene, *The Elegant Universe: Superstrings, Hidden Dimensions, and the Quest for the Ultimate Theory*, New York City, New York: W. W. Norton and Company, 1999.

[36]   N. Forbes, *Imitation of Life: How Biology is Inspiring Computing*. Cambridge, Massachusetts: The MIT Press, 2004.

[37]   R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2 ed. New York City, New York: John Wiley and Sons, Inc., 2001.

[38]   R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural Features for Image Classification," *IEEE Transactions on Systems, Man and Cybernetics,* vol. SMC-3, pp. 610-621, 1973.

[39]   O. R. Mitchell, C. R. Myers, and W. Boyne, "A Max-Min Measure for Image Texture Analysis," *IEEE Transactions on Computers,* vol. C-26, pp. 408-414, 1977.

[40]   W. J. Ho and C. F. Osborne, "Texture segmentation using multi-layered backpropagation," *Proceedings of 1991 IEEE International Joint Conference on Neural Networks,* vol. 2, pp. 981-986, 1991.

[41]   A. Visa, "Unsupervised image segmentation based on a self-organizing feature map and a texture measure," *Proceedings of Pattern Recognition, 1992. Vol.III. Conference C: Image, Speech and Signal Analysis,* pp. 101-104, 1992.

[42]   T. O. Gulsrud and S. O. Gabrielsen, "Classification of microcalcifications using a multichannel filtering approach," *Proceedings of IEEE 17th Annual Conference Engineering in Medicine and Biology Society, vol. 2,* pp. 889-890, 1995.

[43]   M. Amadasun and R. King, "Textural features corresponding to textural properties," *IEEE Transactions on Systems, Man and Cybernetics,* vol. 19, pp. 1264-1274, 1989.

[44]   J. S. Weszka, C. R. Dyer, and A. Rosenfeld, "A Comparative Study of Texture Measures for Terrain Classification," *IEEE Transactions on Systems, Man and Cybernetics, ,* vol. SMC-6, pp. 269-285, 1976.

[45]   W. Chung-Ming, C. Yung-Chang, and H. Kai-Sheng, "Texture features for classification of ultrasonic liver images," *IEEE Transactions on Medical Imaging,* vol. 11, pp. 141-152, 1992.

[46]   R. Khelifi, M. Adel, S. Bourennane, and A. Moussaoui, "Generalized gray level dependence method for prostate cancer classification," *Proceedings of 7th International Workshop on Systems, Signal Processing and their Applications (WOSSPA),* pp. 295-298, 2011.

[47]   J. Yu, "Texture Image Segmentation Based on Gaussian Mixture Models and Gray Level Co-occurrence Matrix," *Proceedings of 2010 International Symposium on Information Science and Engineering (ISISE),* pp. 149-152, 2010.

[48]   J. Virmani, V. Kumar, N. Kalra, and N. Khandelwal, "Prediction of Cirrhosis Based on Singular Value Decomposition of Gray Level Co-occurence Marix and a Nneural Network Classifier*," Proceedings of Developments in E-systems Engineering (DeSE),* pp. 146-151, 2011.

[49]   Q. Yang, J. Peng, and Y. Li, "Chinese sign language recognition based on gray-level co-occurrence matrix and other multi-features fusion," *Proceedings of 4th IEEE Conference on Industrial Electronics and Applications,* pp. 1569-1572, 2009.

[50]   A. Zaim, A. Sawalha, M. Quweider, J. Iglesias, and R. Tang, "A New Method for Iris Recognition using Gray-Level Coccurence Matrix," *Proceedings of 2006 IEEE International Conference on Electro/information Technology,* pp. 350-353, 2006.

[51]   H. Zhu, "Segmentation of blood vessels in retinal images using 2D entropies of gray level-gradient cooccurrence matrix," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP '04)*, vol.3, pp. iii-509-12, 2004.

[52]   L. Bingbing and L. Soo Chin, "Texture retrieval using grey-level co-occurrence matrix for Ikonos panchromatic images of earthquake in java 2006," *Proceedings of IEEE International Geoscience and Remote Sensing Symposium,* pp. 286-289, 2007.

[53]   P. J. Costianes and J. B. Plock, "Gray-level co-occurrence matrices as features in edge enhanced images," *Proceedings of 39th Applied Imagery Pattern Recognition Workshop (AIPR),* pp. 1-6, 2010.

[54]   H. Yong, Z. Chun-xia, and W. Hong-nan, "Directional Analysis of Texture Images Using Gray Level Co-Occurrence Matrix," *Proceedings of Pacific-Asia Workshop on Computational Intelligence and Industrial Application,* pp. 277-281, 2008.

[55]   X. Xiaorong, W. Xijie, X. Fang, and W. Hongfu, "A New Method of SAR Image Segmentation Based on the Gray Level Co-Occurrence Matrix and Fuzzy Neural Network," *Proceedings of 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM),* pp. 1-4, 2010.

[56]   Z. Feng, F. Ju Fu, and S. Qing Yun, "Texture feature based on local Fourier transform," *Proceedings of 2001 International Conference on Image Processing, 2001. Proceedings*, vol. 2, pp. 610-613, 2001.

CHAPTER III

OMNI-DIRECTIONAL TEXTURE ANALYSIS USING GRAY LEVEL CO-

OCCURRENCE MATRICES FOR OBJECTS THAT ARE POORLY

SEGMENTED OR NOT SEGMENTED

3.1   Introduction

Texture analysis is useful in digital image processing because most natural

surfaces are not uniform in pigmentation or height. There are almost an infinite number

of variations in textures of natural objects. Thus texture can be very useful in identifying

different objects because different objects tend to have different textures. As a result,

texture analysis is a very common technique for many digital image processing tasks. It

can be used in segmentation, classification, target recognition, or anomaly detection.

Texture analysis is also used in many applications. These applications run the complete

gamut of those for which image analysis is used. A few examples are medical imaging

[1], robot vision [2], and remote sensing [3].

In texture analysis, the goal is to obtain statistics about the height variations or

pigmentation distributions of a surface. Very rarely can this information be measured

directly, and most often a region of interest (ROI) needs to be statistically analyzed in

order to collect enough information about the surface. This means that a dilemma similar

to the "Heisenberg uncertainty principle" is encountered. As the ROI's size increases, the

sample size for computing the texture statistics increases resulting in improved texture

31

statistics and improved texture specificity, but the certainty about the location of the texture decreases. Conversely, as the ROI's size decreases, the certainty about the location of the texture increases, but the specificity of the texture decreases. This is because texture is rarely uniform across in an entire image. Non-uniformity is present because there can be multiple objects in the image, or variations in light intensity or angle of incidence.

As mentioned above, there is an almost infinite number of variations in texture. Some textures may have a constant directional pattern throughout the surface, such as in the rows of crops in a remotely sensed image of an agricultural field; some textures may have non-constant directional pattern, such as in ocean waves; then there are some textures that have no directional pattern, such as static on old television screens. One of the most common techniques for texture analysis is to use what is called gray level co-occurrence matrices (GLCM) [4]. In this technique, the image is first quantized to reduce the number of gray levels in the image. next a rectangular ROI is analyzed by creating a 2D histogram of the gray levels of all pairs of pixels in the ROI that have the same relative position. The 2D histogram is called a co-occurrence matrix. GLCM is very good at isolating textures from each other when they have different directional patterns or intensities in particular directions, and the direction of the patterns can be estimated. However, if the direction of the pattern is unknown or not constant, GLCM is not as effective. Typically, for omni-directional GLCM, one iteratively applies GLCM with varying position operators where the position operator's directional orientation varies. This approach is extremely computationally expensive, as the computation of the co-occurrence matrix is very costly. Furthermore, the iterative technique produces an

32

overabundance of features, which then need to be processed in the feature selection and classification steps. Since there are a large number of textures that do not have predictable directions in their textures, there needs to be a more computationally efficient omni-directional GLCM technique.

3.2   Methodology

The omni-directional texture analysis technique proposed here is based on co-occurrence matrices. Thus any feature that is calculated using co-occurrence matrices can be turned into an omni-directional feature. This is accomplished by altering the sampling pattern in the ROI. Before the co-occurrence matrix is computed, the image is typically quantized to reduce the number of gray levels. Typically, when a co-occurrence matrix is computed, a rectangular region is first sampled by using a raster scan pattern. Throughout the scan, two pixels are sampled using a pattern where the pixels have a constant relative position to each other described by the position operator. The values of the pixels are used as indexes into a two-dimensional matrix where a sum of all instances of each possible combination of pixel levels is kept. The size of the matrix needed to keep track of all possible combinations is m x m, where m is the number of gray levels in the image. After, the co-occurrence matrix is computed, the matrix is input into one or more of several possible functions that each compute a single feature for the ROI.

The rubber band straightening transform (RBST) is a special method developed specifically for digital mammography by Sahiner *et al.* [5]. In the RBST, pixels around a segmentation border are remapped into a rectangular two dimensional array in order to allow GLCM features to be extracted from the RBST image. The remapping is accomplished by following the perimeter of the segmentation border and sampling along

33

lines perpendicular to the border as describe in detail below. Thus, the method is very dependent on the segmentation of the tumor. Unfortunately a precise boundary cannot usually be determined for objects such as mammographic lesions because the edges are very soft thus only gradually deviate from the background as they are approached. Most radiologist have a concept of what a lesion looks like but it is very difficult for them to express reproducibly precise boundaries around lesions. Thus, many computer aided diagnosis (CAD) systems require the radiologists to only provide loose approximate boundaries around the lesion from which a general ROI can be determined. However, there are many automated segmentation algorithms in the literature, which could theoretically produce reproducible boundaries around lesions as long as the parameters and inputs remain the same. Even so, arriving at precise boundaries remains elusive because conceptually, the boundaries around lesions and similar objects are ambiguous.

The RBST algorithm is a very complicated technique, which first determines all of the pixels on the image border, and next numbers those pixels by progressing around the border in a clockwise fashion, and assigns an incrementing integer pixel label to each new pixel. An offset parameter, K, controls estimation of the border normal at each point by determining which previous and following border pixels to use to estimate the normal. Let (i_k, j_k) be the coordinates of the k-th border pixel and let $p(k)$ and $n(k)$ be the index of the pixel that is K pixels before (the previous pixel), and K pixels after (the following pixel) the current pixel, respectively. The coordinates of the previous and following pixels are given by $(i_{p(k)}, j_{p(k)})$ and $(i_{n(k)}, j_{n(k)})$, respectively. The line that joins these two pixels is used to determine the normal to the current pixel by finding the perpendicular extension to that line which is in the direction leaving the object of interest. Two other

34

parameters are used to control the radial distance used in evaluating the RBST. The user selects two radii, R_{OUT} and R_{IN}, which correspond to how far radially outward and inward, respectively, from each border pixel that the RBST image will be computed. The RBST is then generated by starting at R_{IN} pixels inside the border, and traveling normally to the border and outwards to R_{OUT} pixels away, and placing the pixel graylevel encountered at these locations in the RBST image. Therefore, if there are N_b boundary pixels, the RBST image will be a two-dimensional matrix sized N_b rows by $(R_{OUT} + R_{IN} + 1)$ columns. Figure 3.1 below shows an example case with a segmentation of a mammographic mass, or lesion, shown as a black line. The figure also shows the area where the RBST is extracted from the image and the resulting RBST image. In this example, the inward and outward radii are $R_{OUT} = 40$ and $R_{IN} = 40$, respectively, and the normal parameter $K = 20$.

Once the RBST image is obtained, traditional co-occurrence matrix texture features can be extracted. Clearly, the RBST is extremely sensitive to having an accurately segmented object edge. It is also clear that the RBST is computationally intensive. Thus, we have developed two simplified straightening transform methods.

The first method will be referred to as a simplified rubberband straightening transform (SRBST) and assumes the existence of a circular ROI. The method samples pixels on radial spokes from the center to the perimeter of the ROI (see figure 3.2). In this case, $(i_{p(k)}, j_{p(k)})$ and $(i_{n(k)}, j_{n(k)})$ are much simpler to compute since the perimeter is a circle. Also, the R_{IN} and R_{OUT} parameters can be varied in a similar manner to the RBST, transforming a circular ring around the ROI to a rectangular image. In addition,

35

R_{IN} can be set to a value equal to the radius of the circular ROI. Thus the entire circular ROI is transformed to a rectangular image. Since the sampled pixels are radial spokes of a circular ROI, the position operator is rotated as it moves across the ROI. Thus, if traditional co-occurrence matrix texture features are extracted from the SRBST image, the result is an omni-directional texture analysis method.

Our second approach to the resampling is called the spiral straightening transform (SST) [1]. This technique, like the first, starts with a circular ROI. Instead of sampling the ROI in a radial spoke pattern, it is sampled in a semi-spiral pattern with evenly spaced samples progressing from the ROI's center to its outer edge. (see figure 3.3). For computational ease, rather than actually computing a spiral path from the center of the ROI to its outer edge, we compute a series of R concentric rings, progressing from a small ring at the center to a large ring at the outer edge of the ROI. For each concentric ring, a SRBST is computed, and then the R SRBST images are concatenated to form a final SST image. Assuming R_{IN} and R_{OUT} are constant, then the rth concentric ring results in a two-dimensional matrix sized N_{b_r} rows by $\left(R_{OUT} + R_{IN} + 1 \right)$ columns, and the final SST image is sized

$$\left(\sum_{r=1}^{R} N_{b_r} \right) \times (R_{OUT} + R_{IN} + 1)$$

(3.1)
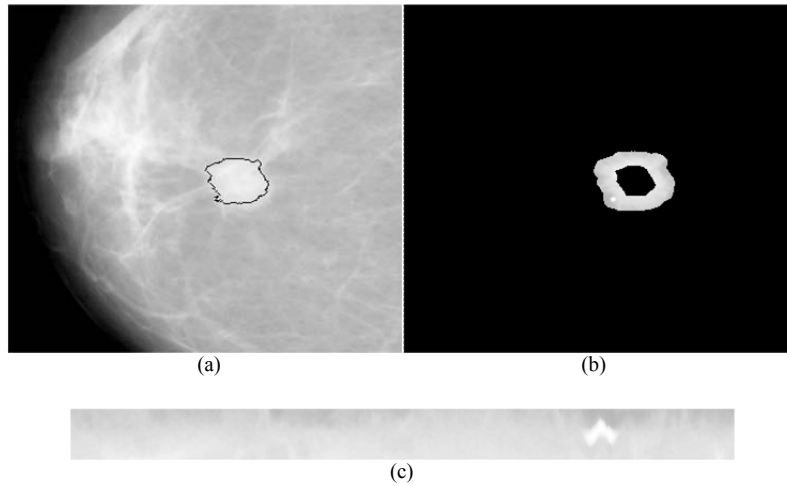
Figure 3.1   Example case of rubber band straightening transform.

(a) Mammogram showing lesion boundary in black. (b) RBST preimage in the original mammogram. (c) RBST image. [11].
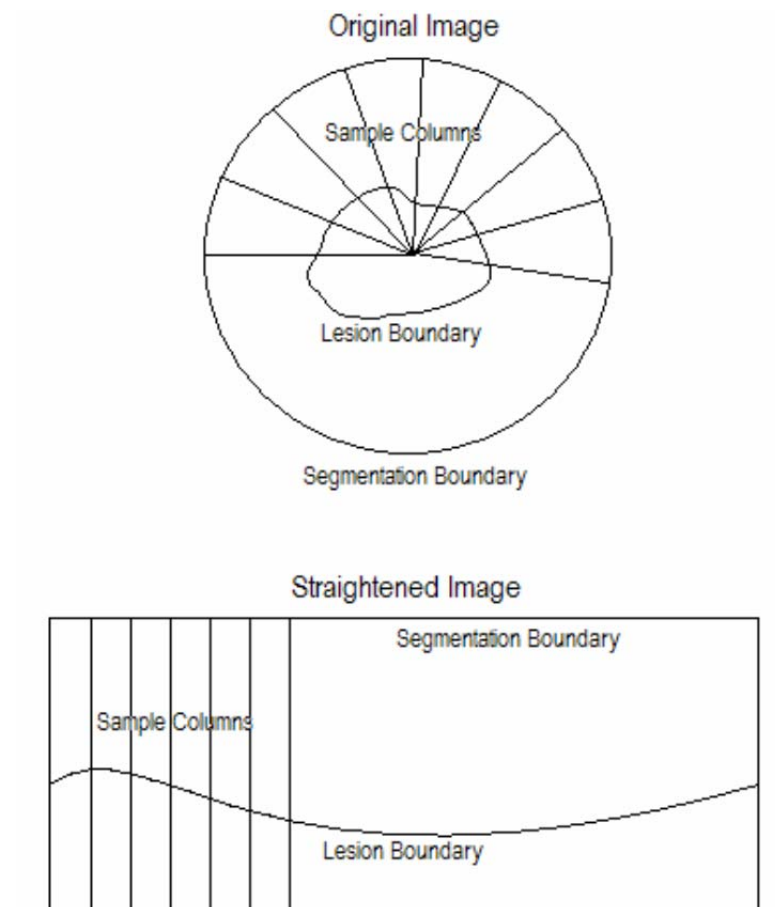
Figure 3.2   Illustration of simplified rubber band straightening transform [1].

37
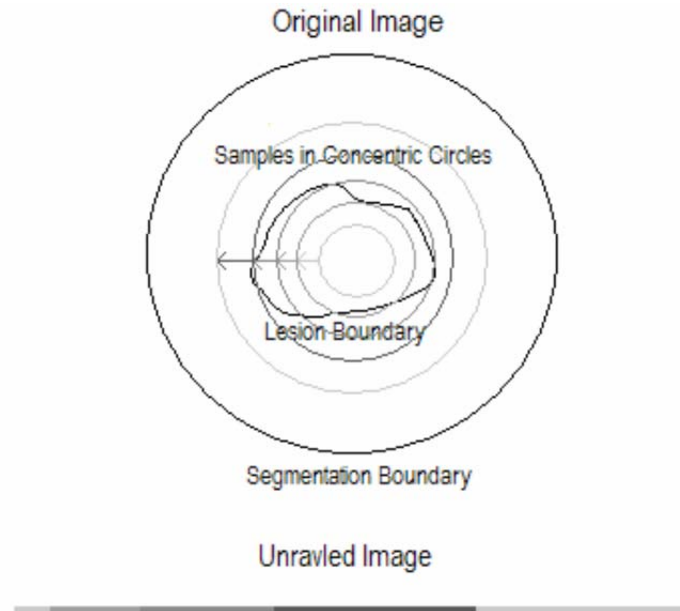
Figure 3.3   Illustration of the spiral straightening transform (SST) [1].

While the SST takes longer to compute because the number of samples is proportional to the area of the ROI instead of the perimeter (as in the SRBST), it has a few advantages over the SRBST. Very often the target class in a classification problem has particular spatial frequencies in its texture. Since the spatial frequency of the samples is constant with the SST, it may be more suitable for problems where optimal spatial frequency of the sampling can be determined. However, if the optimal spatial sampling frequency cannot be determined, the SRBST might be better since the spatial sampling frequency varies. Also, the variable spatial frequency of the SRBST may make it more prone to classification errors for some problems. Another advantage to having a constant sampling frequency is that there will be more samples with the correct angle. This allows for a much larger sample for computing statistically-based texture features, and thus there is a better chance of detecting the target.

38

3.3   Applications

There are many applications outside of image processing where directional transducers are preferred over omni-directional transducers because directional transducers can penetrate background noise better. Two common applications where this is the case are sonar and radar. In target recognition problems, omni-directional texture analysis will typically display this same behavior by producing a higher number of false positives than a directional texture analysis technique where the directional components of the texture are known and exploited. In cases where such knowledge is unavailable, omni-directional texture analysis may be the best option just as an omni-directional antenna is likely better than a highly directional antenna in cases where the direction to the source of a transmission is unknown.

In some problems the target may have radial symmetry. In such problems, the omni-directional texture analysis technique used in this paper may be better because it samples a circular ROI. In such cases, using a directional texture analysis technique may produce more false positives than the omni-directional technique because the omni-directional technique is better synchronized with the target texture. The synchronization should even be tolerant of small errors between the center of the ROI and the center of the target because of the high correlation between the normal vectors toward circles with centers that are close together. This same phenomenon makes omni-directional texture analysis useful for targets that elliptical or other pseudo-circular shapes.

3.4   Experiments

In order to demonstrate the utility of the proposed straightening transform methods and resulting omni-directional texture analysis, two experimental analyses are

presented here. The first is a situation in which the target has radial symmetry, and the second is a situation in which the directional components to the texture are unknown.

3.4.1   Breast Lesion Classification via Digital Mammography

In the first experimental study, mammograms were analyzed to classify lesions as either malignant or benign. Current mammography technology typically produces high resolution 2D grayscale images of the internal structure of the breasts. Extremely high resolution is needed in order to have the ability to detect microcalcifications, which are tiny abnormal deposits of calcium that are associated with breast cancer. However, even if microcalcifications cannot be detected in the mammogram, other clues may be present in the shape and texture of lesions to indicate whether the lesions are malignant or benign. Spiculations are tentacle-like structures that radiate from the lesion and are often correlated with malignancy. These spiculations make omni-directional texture analysis a good fit for this problem because they represent radial features that are typically only present in malignant lesions.

As discussed previously, it is a difficult task to get precise boundaries around mammographic lesions because the edges appear very soft. In the mammography community, the ambiguity has been resolved into three different classes of segmentation based on what is included. A segmentation that only includes the densest part of the lesion is called a core segmentation. The core in a lesion is distinguished by very little texture present because the density remains relatively uniform, and the density is typically at the extreme end of the gray scale. A segmentation that follows the boundary where the density begins to deviate from the normal background is called a periphery segmentation. The third segmentation type is the spiculation segmentation, which

40

includes the periphery as well as spiculations emanating from the lesion (if present). (Figure 3.4 shows a few illustrations of the different segmentations.) While the different types of segmentations appear to have solid conceptual definitions, it is important to note that normal breast tissue has many variations and structures that can mimic the appearance of lesions, and the breast is a 3-D object that is projected into a 2-D image, which means these structures very often overlap each other in the mammogram. Figure 3.5 illustrates the difficulty of obtaining precise segmentations in real mammograms. It shows several different lesions along with the ROI determined by the radiologist and the results of two different periphery segmentation algorithms. Very often the segmentation results vary greatly when different methods are used or the parameters of a method are varied slightly. The difficulty in obtaining segmentations of course makes it very difficult to use RBST in a CAD system.
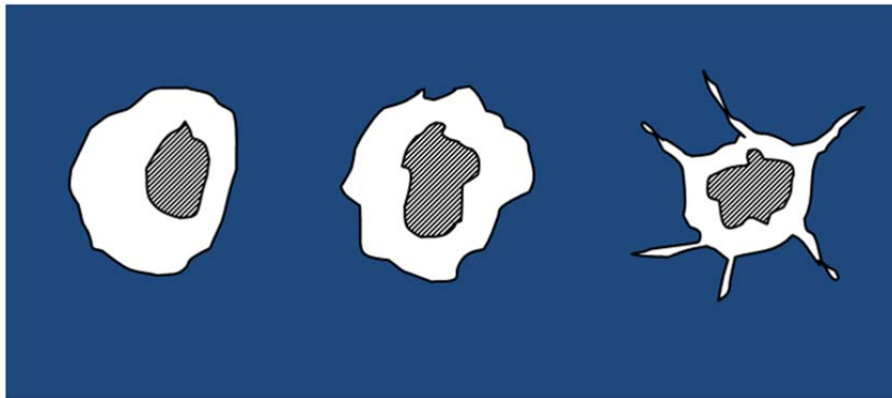
Figure 3.4   Examples of segmentations.

The first two on the left are examples of a periphery with a core segmentation. The last one on the right is a spiculation with a core segmetation.

41
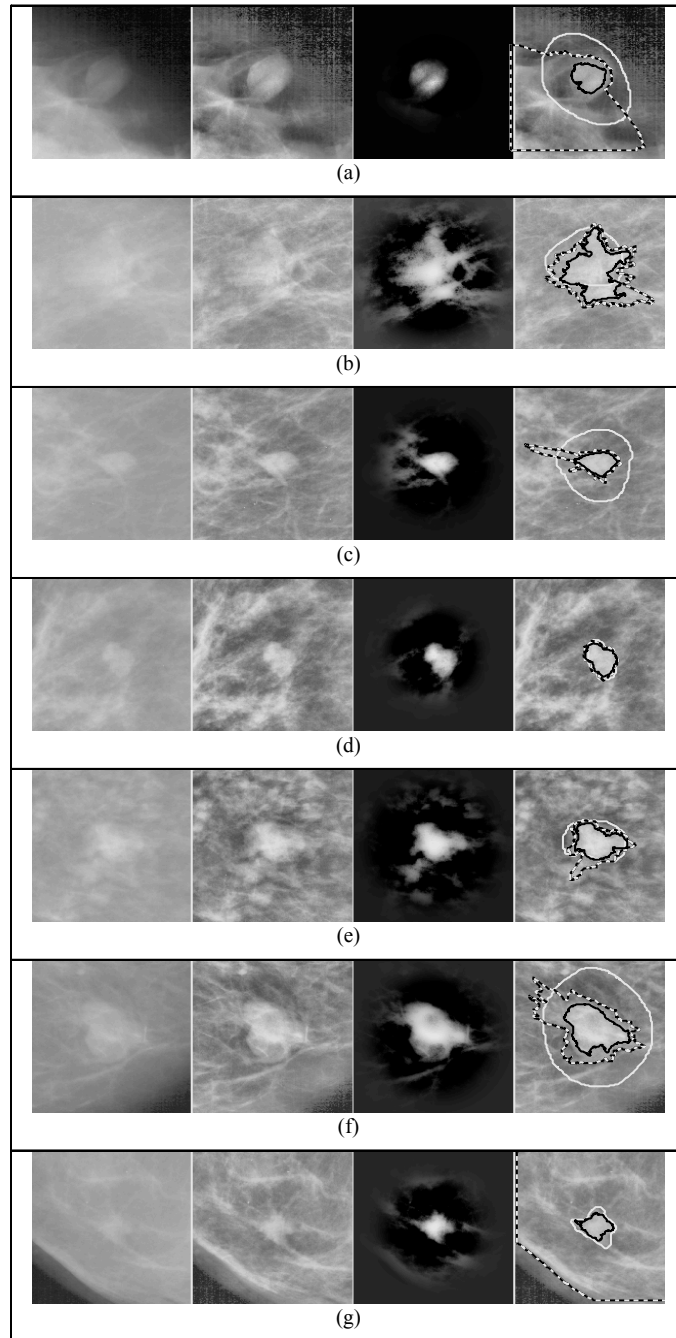
Figure 3.5   Several DDSM cases (a-g) illustrating the difficulty of accurate
                segmentation.

Left-to-right columns: Original mammogram image, contrast enhanced image, Level set
segmentation enhanced image, contrast enhanced image with overlays of the DDSM hand
drawn ROI (white line), level-set segmentation developed by Ball et al. (black line) and
Catarious segmentation (black and white dashed line). [11]

www.manaraa.com

The data used in this experiment was obtained from the University of South Florida (USF) Digital Database for Screening Mammography (DDSM) [10] and did not have the resolution necessary to detect microcalcifications in the lesions. The data did contain segmentations provided by experts in mammography. The segmentations were loose hand drawn approximately circular regions and also indicated whether the lesion was malignant or benign. Figure 3.6 below shows all of the cases from the DDSM used in this experiment. Notice that there were a total of 60 cases with 30 malignant and 30 benign.
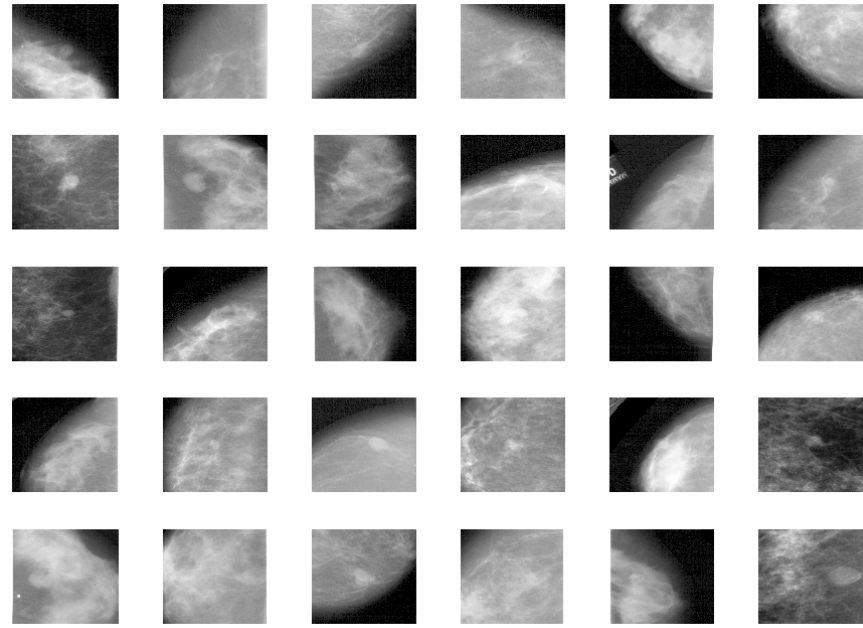
Benign
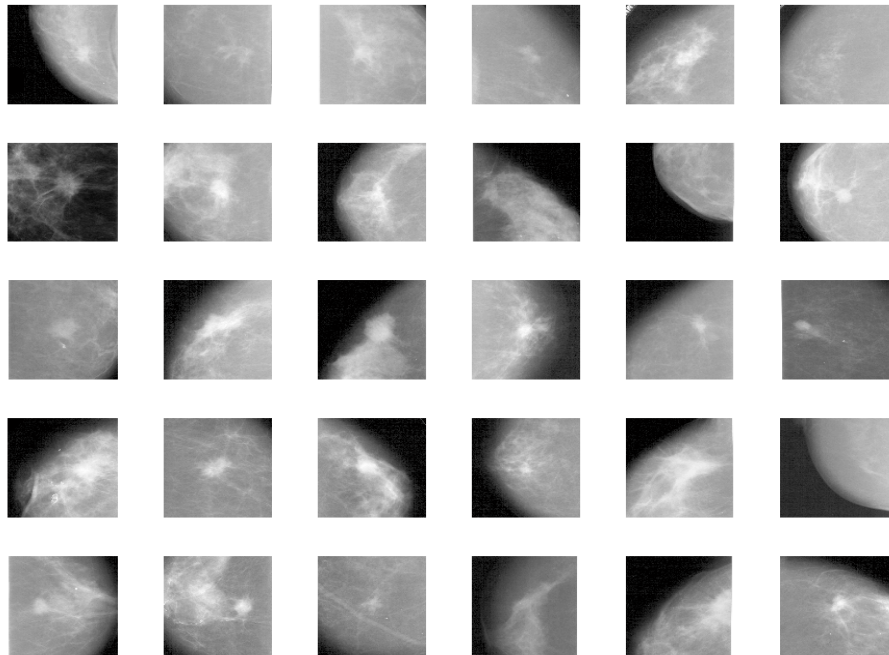
Malignant

Figure 3.6   Cropped DDSM case used in the study

The first step in the analysis was to use the expert ROI segmentations to extract a circular region with the lesion at the center. As indicated, the experts provided roughly circular regions. The circular ROIs were obtained by centering a circle at the centroid of the expert human segmentation with a diameter equal to the major axis of the expert segmentation. This virtually guaranteed that the ROI included all the regions segmented by the experts, and was likely centered close to the center of the lesion. Of course there may be possible segmentation shapes this approximation fails to produce desirable ROIs for, but it is likely easy to train a radiologist to make appropriate segmentations for the algorithm and a system could display the ROI so that corrections could be made if it is inaccurate. The second step is to use either the SRBST or SST to extract a co-occurrence matrix. This experiment diverged from the standard technique in that it used the raw values in the co-occurrence matrix as features instead of computing features from the co-occurrence matrix. Thus there were typically a large number of features since the co-occurrence matrix was size m x m, where m is the number of gray levels in the image. Thus if there were, for example, 256 gray levels in the image, there would be 65536 features in the co-occurrence matrix. The large number of features necessitated the use of a feature selection technique, which in this case was stepwise linear discriminate analysis (SLDA). In the SLDA step, features were chosen to maximize either receiver operating characteristic (ROC Az) [9] or a metric described in the thesis called class overlap rating (COR). After the SLDA step reduced the number of features to a more manageable number, a classifier used the selected features to classify the lesion as malignant or benign. There were three classifiers tested for this step. They were nearest neighbor,

nearest mean, and maximum likelihood. Leave one out protocol was used to test the method.

Previous studies using the DDSM database of digital mammograms have studied the use of the RBST. In [11], Ball conducted a study using the same set of DDSM mammograms that were used in this dissertation. The goal of Ball's study was to advance techniques for segmentation of mammographic masses. He compared his newly developed segmentation technique (ALSSM) to a more standard approach developed by Catarious (CSM) [14]. Both segmentation methods were followed with a RBST, feature extraction and optimization, and finally a classification of the mammographic mass or lesion as either benign or malignant. The feature sets used in his study were extensive, including not only GLCM texture features, but also morphological features based on the normalized radial distance (NRD) and patient age. The result was a feature vector of size 1031 (see table 3.1). Arguably however, patient age is the single most contributing feature to this type of computer aided diagnosis (CAD) system [16]. The feature reduction and classification methods employed were SLDA , ML, and K-nn, respectively. The results of his CAD system were overall accuracies in the range of 80-90% (see table 3.2), when the design parameters were optimized. When using patient age and morphological features alone, the results were in the range of 77-82%. Thus, the RBST-based texture features increased the overall classification accuracies by about 5%. In another study conducted by Gulsrud and Gabrielsen on a different set of data, where only GLCM features were used for classification (no age feature), an accuracy of up to 65% was obtained [17].

46

In comparison, this study utilizes only the SRBST or spiral transform-based texture features (and not including patient age), which resulted in overall accuracies in the range of 70-72%, when the design parameters were optimized. Thus, the SRBST and spiral transform approaches produce comparable results in terms of efficacy when compared to the RBST approach, yet the SRBST and spiral transforms do not require an accurate segmentation and are significantly less computationally expensive.

Table 3.1   Summary of features used in John Ball's study.

Feature Type and source	Features	Num. Features
Patient age (DDSM)	Patient Age	1
Morphological (SB)	Area, Axis ratio, Box ratio, Circularity, Convex hull area, Eccentricity, Equivalent diameter, Extent, Extent ratio, Major axis length, Minor axis length, Perimeter length, Solidity, Width to height ratio	14
Statistical (SB)	Gray level mean, Gray level std. dev, Gray level std. dev. ratio^3	3
NRL (SB)	Entropy, Length, Mean, Roughness, Std. dev., Zero crossing count	6
GLCM (SB)	Energy, Variance, Correlation, Inertia, Inverse Difference Moment, Entropy	144
GLCM (RBST)	Energy, Variance, Correlation, Inertia, Inverse Difference Moment, Entropy	864

Table 3.2   Classification accuracy using John Ball's CAD system.

Feature Set	Overall Accuracy (%)						Number of False Negatives					
	C	P	S	C+P	C+S	C+P+S	C	P	S	C+P	C+S	C+P+S
B	83	87	88	82	90	90	4	4	5	7	4	4

47

3.4.2 Levee Landslide Detection via Synthetic Aperture Radar

Earthen Levees protect large areas of populated and cultivated land in the United States from flooding. In the United States there are more than 150,000 kilometers of levee structures of varying designs and conditions. The potential loss of life and property associated with the catastrophic failure of levees can be extremely large [12]. Currently, there are limited processes in place to prioritize the monitoring of large numbers of dam and levee structures. There is a need to prioritize the monitoring of the network of dam and levee structures. Levee managers and federal agencies will benefit from any tools allowing them to assess levee health rapidly with robust techniques that identify, classify and prioritize levee vulnerabilities with lower costs than traditional programs not based on the use of remote sensing.

This project required omni-directional GLCM because the directional components of the textures in the levee are unknown. This is because the orientation of the levee follows the path of the river, which varies greatly. This can be seen in figure 3.7, which shows one of the radar images used in this experiment. In the image you can see the river surrounded by bright wooded terrain in the middle of the dark agricultural terrain. Although, the levee is not visible in the larger image, it is a dark line that separates the wooded terrain from the agricultural terrain. This is because the soil in this area of the country is very fertile, so every bit that is protected by the levee is utilized in agriculture.

48
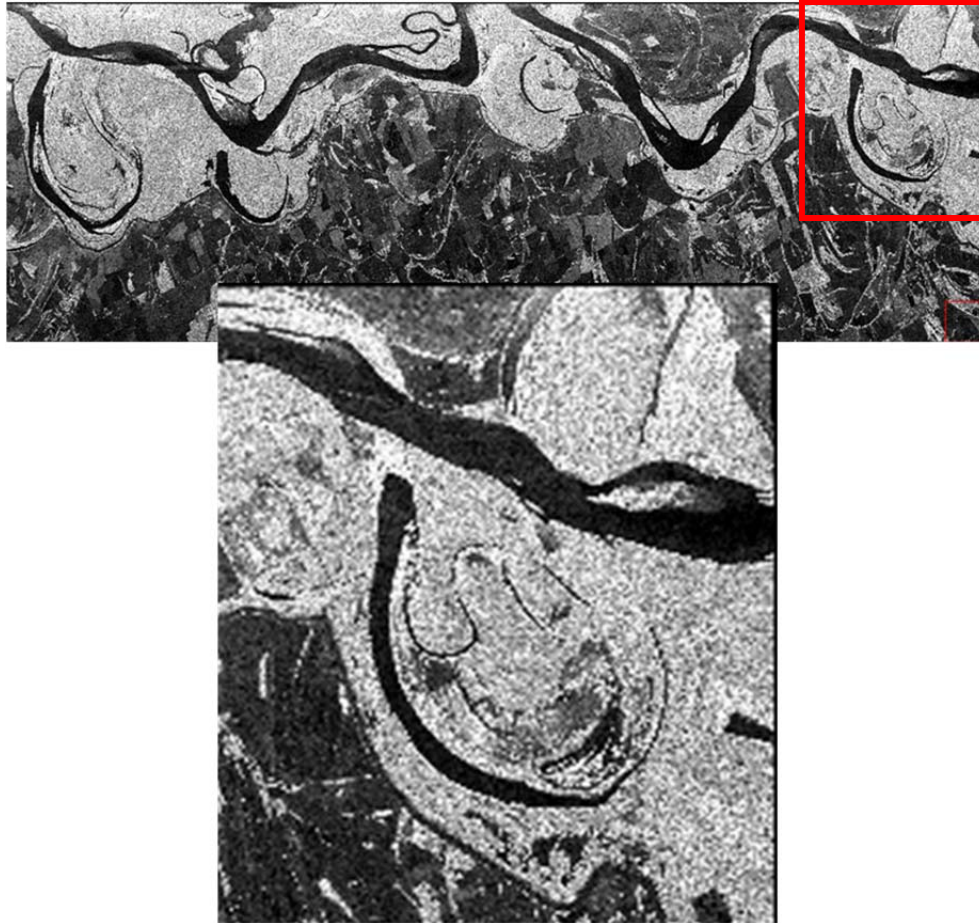
Figure 3.7   SAR image of the Mississippi River.

 The images have been rotated so north is to the right.

This experiment used L-band SAR data collected from JPL's UAVSAR [13] of the Mississippi River levee system between Vicksburg, MS and Clarksdale, MS. Each pixel has a spatial resolution of 6x6 m^2. The instrument created separate backscatter intensity images for three different polarizations. These polarizations are HH, HV, and VV. The aircraft made two passes over the collection area. In the first pass, the plane traveled south to north with the sensor looking westward, and in the second pass, the plane traveled north to south with the sensor looking eastward.

49

The data was collected on June 16, 2009, which according to the National Ocean and Atmospheric Administration (NOAA) data was hotter than usual and was the 6th driest June since 1895 for that region [15]. Ground truth for landslides on the levees was obtained using optical imagery from the National Agriculture Imagery Program (NAIP), the Army Core of Engineers records, and this study's researchers manually observing the levees. The landslides studied in this experiment were confirmed by at least two of these sources (see figure 3.8 below).

For detecting landslides on the levees, we developed a supervised classification system that fuses texture information from the HH, HV, and VV polarizations for both passes. There are four main steps to the classification system. They are (1) compute a uniformly illuminated composite image from the six input images (HV, HV, and VV for both passes); (2) extract omni-directional texture features from the six input images plus the composite image; (3) select features and train a statistical classifier; and (4) classify the levee section using maximum likelihood classification. The following subsections detail each of these steps.

One of the disadvantages of using SAR imagery for levee applications is that the image tends to be dim or in shadow on the reverse slopes of hills. Since an earthen levee is essentially a hill extended in a direction that parallels the river's course, it is difficult to collect imagery of both sides of a levee using a single pass with an airborne SAR since one side will be darker than the other. This was overcome by using data from two passes with the sensor looking in opposite directions at the target area to create a composite image. The first step in computing the composite image is to reduce the images from all the polarizations into a single grayscale image for each pass. This is done by computing

50

the Euclidean distance of the vector $\langle HH_{x,y}, HV_{x,y}, VV_{x,y} \rangle$ for each pixel (x, y). Next, both of the grayscale images are normalized by subtracting the mean for each image and then dividing by the standard deviation. This step ensures that both images have the same intensity. Finally, the grayscale images are combined by selecting the maximum value for each pixel. Figure 5 shows the HH images for both passes and the composite for a subset of the levee section.

Figure 3.8   Subset of levee. Left Image: Actual levee segment.
Right Image: Mask segment. Green: landslide; Red: healthy levee.

For the composite images, both directional and omni-directional texture features were extracted. For the directional texture analysis, a standard GLCM was computed and texture features extracted. For the omni-directional texture analysis, a SST was applied and then standard GLCM method was utilized with the SST image. For each of the analysis techniques, a number of spatial operators were utilized, as well as a large number of features extracted from the resulting GLCMs. As a result, a very large set of texture features results. In order to determine which types of textural features were best, three different tests were performed. The first test used all the features available, the second

51

used only directional features, and the third used only omni-directional features. However, in each test, there were still a large number of features, so a dimensionality reduction technique had to be used. The authors of the study chose to use stepwise linear discriminate analysis (SLDA) [9] for the dimensionality reduction. After dimensionality reduction, a maximum likelihood classifier was used to classify each pixel in the subsets as either a landslide or a normal levee pixel.

The accuracy was measured using two different methods. The first was to compute a standard pixel-by-pixel confusion matrix using the ground truth and the classification map output from the classifier. While being a standard approach, a confusion matrix does not account for spatial distribution of the classes in the classification map, and since landslides typically are larger than a single pixel, it is only necessary to detect a subset of the landslide pixels. Thus, for the second metric, contiguous regions classified as landslides are segmented and determined to be true positives if the region corresponds to an actual landslide. A region corresponds to a landslide if the center of the region is within a specified distance tolerance of the landslide. The tolerance is determined by the geo-registration accuracy of the SAR image, which is not constant because of foreshortening.

Typically, pixels in SAR images possess a high degree of spatial correlation. Thus, randomly selecting pixels for training and testing from the same levee section can lead to deceiving conclusions about the accuracy of a technique since highly correlated, adjacent pixels may be selected for both training and testing. In order to avoid this, pixels for training and testing were always selected from different sections of the levee in a leave-one-out fashion with one section used for testing and the other levee sections used

52

for training the classifier. The three sections of levee used in this study were each separated by several kilometers with one on a different side of the river. Thus, spatial correlation between the training and testing pixels was negligible.

In this study, the classification accuracy was determined using only standard GLCM, using only omni-directional GLCM, and using both types of features together. The confusion matrices for these three tests are shown in the tables below (3.3, 3.4, and 3.5). Judging by the confusion matrices and pixel-by-pixel classification accuracy, there is little difference between the three approaches. However, when the spatial distribution of the classification map is considered, it becomes clear that the technique that uses both types of GLCM features detects all four landslides and has fewer false positive regions (see Table 3.6).

From the results, a general trend can be seen where standard GLCM produces a classifier that is more likely to reject (thus producing fewer true positives and fewer false positives), and omni-directional GLCM is more likely to accept (thus producing more true positives and false positives). When using both types of features, the classifier results are balanced between using one type of feature or the other. This allows the classifier to detect all of the landslides while producing fewer false positives than using omni-directional GLCM only. One would expect that standard GLCM would be more selective because of its more directionally selective sampling and position operator, and thus have fewer false positives and true positives. The omni-directional GLCM is less selective, which helps it detect the landslides better but makes it more likely confused by similar textures that have a different directional element. In the levee experiment, the direction of the levees only varied by about 45 degrees, so the directional components to the texture

53

likely varied by about 45 degrees also. One would expect omni-directional GLCM to be most advantageous in cases where the directional components of the target texture vary by a full 360 degree range and when detecting all targets is more important than producing false positives. There are a great number of applications that fall in this category in geosciences and remote sensing.

Table 3.3   Confusion matrix using standard GLCM only. [6]

	slide	normal	
slide	38	83	0.31
normal	73	3375	0.98
	0.34	0.98	0.96

Table 3.4   Confusion matrix using omni-directional GLCM only. [6]

	slide	normal	
Slide	26	95	0.21
Normal	111	3337	0.97
	0.19	0.97	0.94

Table 3.5   Confusion matrix using both types of GLCM [6]

	slide	normal	
slide	51	70	0.42
normal	146	3302	0.96
	0.26	0.98	0.94

Table 3.6   True positives and false positives when contiguous regions classified as landslides are segmented.

	true positive	false positive
Standard GLCM	2	13
Omni-Directional GLCM	4	36
Both	4	22

Note that there are 4 landslides, so a maximum of 4 true positives. [6]

54

3.5 Conclusion

Two omni-directional adaptations of gray level co-occurrence matrix analysis were developed and experimentally evaluated. The adaptations are based on a previously developed RBST that has been traditionally used for analysis of segmented masses in digital mammograms. The new methods are beneficial in that they can be applied to imagery where the region of interest is either poorly segmented or not segmented. In brief, the methods are based on the concept of extracting circular windows around each pixel in the image (to compute local texture values) and the ROIs are radially resampled to derive rectangular images. The images derived from the resampling are then suitable for standard GLCM techniques. The methods were applied to both remotely sensed synthetic aperture radar imagery, for the purpose of automated detection of landslides on earthen levees, and to digital mammograms, for the purpose of automated classification of masses as either benign or malignant. Experimental results show the newly developed methods to be valuable for texture feature extraction and classification of un-segmented objects.

The omni-directional texture analysis technique described in this paper was shown to be useful in both remote sensing and medical imaging applications. As the levee example shows, omni-directional texture analysis may tend to get more false positives in some problems, but it will typically get fewer false negatives. This implies that omni-directional texture analysis will be better suited for problems where higher numbers of false positives are more tolerable than false negatives. Both examples presented in this paper are such cases, where false positives are more tolerable than false negatives.

55

These examples also demonstrate cases where the sampling pattern in omni-directional analysis is a better fit with the textures present. In the levee example, the levees used in the experiment varied in orientation by about 45 degrees, so the directional components in the texture may not have varied enough to make standard GLCM useless. In such cases, using features from omni-directional and standard GLCM can provide good results. In the mammography example, the texture direction varied around the center of the lesions. Thus, omni-directional texture analysis was more useful than the standard texture analysis techniques. The omni-directional GLCM techniques used in this study are also more computational efficient than the standard approach since it only a co-occurrence matrix to be computed once.

3.6   Acknowledgements

3.7 References

[1] Matthew A. Lee, "Analysis of Breast Lesions using a Simplified Rubber Band Straightening Transform and the Onion Transform," Master's Thesis, Mississippi State University, 2007.

[2] Hongyan Quan, "A New Method of Dynamic Texture Segmentation Based on Optical Flow and Level Set Combination," *2009 1st International Conference on Information Science and Engineering (ICISE)*, pp.1063-1066, 26-28 Dec. 2009.

[3] S. Newsam, L. Wang, S. Bhagavathy, B.S. Manjunath, "Using Texture to Annotate Remote Sensed Datasets," *Proceedings of the 3rd International Symposium on Image and signal Processing and Analysis, vol. 1, pp. 72-77, 18-20 Sept. 2003.*

[4] Robert M. Haralick, K. Shanmugam, Its'Hak Dinstein, "Textural Features for Image Classification," *IEEE Transactions on Systems, Man and Cybernetics*, vol.SMC-3, no.6, pp.610-621, Nov. 1973.

[5] B. Sahiner, H.-P, Chan, N. Petrick, M. Helvie, and M. Goodsitt, "Computerized Characterization of Masses on Mammograms; The Rubber Band Straightening Transform and Texture Analysis," *Medical Physics*, vol. 25, pp. 516 – 526, April 1998.

[6] Matthew A. Lee, James V. Aanstoos, Lori Mann Bruce, and Saurabh Prasad, "Application of Omni-Directional Texture Analysis to SAR Images for Levee Landslide Detection," *Proceedings of 2012 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, July 2012.

[7] Matthew R. Lopez, "Synthetic Aperture Radar," Sandia National Laboratories, November 11, 2010. Available: http://www.sandia.gov/radar/

[8] "UAVSAR Uninhabited Aerial Vehicle Synthetic Aperture Radar," Jet Propulsion Laboratory, August 31, 2010. Available: http://uavsar.jpl.nasa.gov/

[9] Richard O. Duda, Peter E. Hart, David G. Stork, *Pattern Classification*, 2 ed., New York: John Wiley & Sons, Inc.

[10] "University of South Florida Digital Mammography Home Page," University of South Florida. Available: http://marathon.csee.usf.edu/Mammography/Database.html

[11] John E. Ball, "Three Stage Level Set Segmentation of Mass Core, Periphery, and Spiculations for Automated Image Analysis of Digital Mammograms," PhD. Dissertation, Mississippi State University, 2007.

[12]   J. V. Aanstoos, K. Hasan, C. G. O'Hara, S. Prasad, L. Dabbiru, M. Mahrooghy, B. Gokaraju, M. Lee, R. Nobrega, "Earthen Levee Monitoring with Synthetic Aperture Radar", *Proceedings of 2011 IEEE Applied Imagery Pattern Recognition Workshop*, 2011.

[13]   Uninhabited Aerial Vehicle Synthetic Aperture Radar [Online]. http://www.jpl.nasa.gov/missions/index.cfm?mission=UAVSAR

[14]   D.M. Catarious, A.H. Baydush, and C.E. Floyd Jr., "Incorporation of an iterative, linear segmentation routine into a mammographic mass CAD system," *Medical Physics,* vol. 31, no. 6, pp. 1512-1520, Jun. 2004.

[15]   Climate of 2009 June in Historical Perspective [Online]. http://www.ncdc.noaa.gov/oa/climate/research/2009/jun/jun09.html

[16]   M.H. Dilhuydy, A. Le Treut, and J. Weber, "Architecture of the Breast," in *Mammography: A guide to interpretation*, Translated ed, A. Le Treut and M. H. Dilhuydy, Eds. Boston, MA: Mosby Year Book, pp. 38-65, 1991.

[17]   T. O. Gulsrud and S. O. Gabrielsen, "Classification of microcalcifications using a multichannel filtering approach," *IEEE 17th Annual Conference in Engineering in Medicine and Biology*, 1995.

CHAPTER IV

HYPERSPECTRAL DIMENSIONALITY REDUCTION USING CONCURRENT

SPATIAL-SPECTRAL GROUPING

4.1   Introduction

Hyperspectral imagery (HSI) contains enormous amounts of information about
the scene it depicts because of the large number of spectral bands. The huge amount of
information can be very useful in extracting further information from the scene, such as
classification, pixel unmixing, anomaly detection, etc. However, the amount of
information can cause problems because it requires large data storage, significant
processing resources, and high bandwidth to transmit. It has previously been suggested
that clustering or grouping similar bands may be a way to deal with some of the
requirements of HSI because bands with correlated spectral frequency tend to have
correlated information [1]. HSI contains two types of features that can be used for band
grouping (spectral features and spatial features). Commonly, band grouping has been
done using only spectral information about the scene because of the poor spatial
resolution common in many HSIs. Current spectral band grouping techniques can be
broken into two different categories. These categories are supervised and unsupervised.
Some examples of unsupervised spectral band grouping are uniform partitioning,
correlation and mutual information. The supervised techniques often use two metrics: one
to measure how similar the bands are, and the other to measure how well they distinguish

the classes. Two examples of supervised spectral band grouping techniques are Bhattacharyya Distance X Correlation and Jeffries Matusita X AMI [2, 11, 12]. In the past, hyperspectral images typically had spatial resolutions on the order of 30x30 m^2, and thus, it was only feasible to use spectral information in band grouping. However, with the proliferation of HSI, images with better spatial resolution have become more common, and thus it is more feasible to use spatial features in analysis of HSI.

4.2   Band Grouping

As mentioned in the introduction, band grouping is way to deal with the high dimensionality of HSI. Band grouping works by placing bands into small groups so that the groups can be processed individually at some step in the analysis. It is useful because many of the bands in a typical HSI contain redundant information. There are a large number of ways to group bands together. Some of them are supervised and others are unsupervised. Once a band group has been determined, there are a limited number of strategies to use the band grouping. In classification problems, the strategy used is mostly determined by the classification technique, which can either be a single classifier or multiple classifier with decision fusion (MCDF) approach [3, 13-15].

In the case of a single classifier, usually a set of features fewer in number than the bands in the group are extracted from each group. This reduces the dimensionality of the problem and alleviates some of the problems associated with high dimensionality. However, care must be taken because at the same time the dimensionality is being reduced, useful information may be discarded. Examples of ways to extract the features from the groups are averaging (weighted or unweighted) the bands in the group, selecting one band to represent the group, Principal Component Analysis (PCA), and Linear

Discriminate Analysis (LDA). Once the features are extracted from all the groups, they can be input into a classifier. There are many classifiers that can then be used.

In the multi-classifier approach there are more options for using the band groups. The most obvious option is to use individual classifiers for each group, and then fuse the results using a decision fusion technique such as majority vote or linear opinion pool (there are several other techniques) [4]. In this option, each classifier has highly correlated information that may allow it to sift through noise. The other simple option is to select different a different set of bands from all the groups for each classifier. Thus each classifier will have more diverse information. A third option, is to perform some sort of dimensionality reduction on the bands in each group and then feed the reduced features into the classifiers for each group. This is much like the single classifier approach except you use a classifier for each group.

Grouping bands together has several advantages. The most obvious is that when each group is processed separately, there are fewer hyperspectral bands, and thus a lower dimensionality. This is especially apparent in the multi-classifier approach, but is also present in the single classifier approach because there are fewer bands to extract features from and the total number of features that need to be processed by the classifier is reduced. Band grouping may also identify particular bands that are less useful (or even completely useless) to the problem being solved. For example if a band is so different from the other bands that it is in a group by itself, it could mean that the band is not working properly or contains a high degree of random noise since a high degree of redundancy in adjacent bands of a HSI is expected. However, this may not always be the case, so care must be taken to ensure that such bands are truly useless before they are

61

discarded. Finally, organizing bands into highly correlated groups may make it easier to visualize the HSI because analysts may be able to concentrate on the features extracted from the groups instead of all the bands within them.

4.3   Using Spatial Information in Spectral Band Grouping

Spatial features have proven to be very useful in many classification problems that use other sensors beside HSI. One example is a project where the objective was to detect kudzu (*Pueraria montana* var. *lobata)*, an invasive plant species in the southern United States that is originally from East Asia. Though only multispectral imagery was available, the investigators found that they could effectively distinguish kudzu from native vegetation because kudzu has an extremely smooth texture compared to the native vegetation. Figure 4.1 shows a false color image from the project with kudzu and natural vegetation. While HSI has a much greater spectral resolution than multispectral, noise often prevents it from being exploited to its full potential and often makes tasks that should be feasible in theory infeasible. Thus it may not be possible to distinguish one plant species (perhaps kudzu) from another (the native plants) because noise from the environment or sensor makes it impossible to detect the small differences in the spectrum. In many applications, it might help to combine spatial feature and with spectral features.

Figure 4.1   False color image showing kudzu (*Pueraria Lobata*).

Kudzu is in the regions of bright red, smooth textures surrounded by the regions of rough textured forest [16].

There are two categories of problems that will benefit from spatial-spectral band grouping. These are problems where shape is important, and problems where texture is important. In problems where shape is important, there may be good information in the spectrum, but shapes of regions should a defining (or extremely useful) feature to the problem. A few examples of hypothetical problems in which shape is important in classification are differentiating

- roads and building/parking lots,
- lakes, rivers, and canals,

63

- crops and similar natural vegetation.

In problems where texture is important, there again may be useful information in the spectrum, there should also be critical information in the texture of the objects. Hypothetical examples where texture is important are

- identifying invasive species, such as kudzu,

- detecting landslides, or perhaps even predicting them,

- identifying places where erosion via water or wind has occurred,

- detecting empty or fallow agricultural fields.

Spatial and spectral features can be combined in several ways. The most obvious is to concatenate the spatial and spectral features together to produce one large (perhaps massive) feature vector. This feature vector can then be input into a classifier, which then uses training data to make a decision. While this solution is straightforward and simple to implement, it exacerbates the high dimensionality problem, which is already a potentially serious concern with HSI. When this exacerbated problem is then tackled using some feature set selection algorithm such as Stepwise Linear Discriminate Analysis (SLDA) [5], the system tends to disregard a significant amount of potentially useful information, plus choosing the best set of features is an NP problem. Another way spatial and spectral features can be combined is to use some mathematical formula to combine the spatial and spectral features to create composite features. This strategy is very similar to a technique called panchromatic sharpening, or pan-sharpening [10] and may be very useful in many situations. However, it can be very difficult and require a great amount of labor to determine a problem specific formula to use in compositing the features, and in the end, it could prove useless. A third way to combine spatial and spectral features is to use a tree

64

classifier [6]. Tree classifiers may be very useful in problems where spatial and spectral features are combined if some sort of dimensionality reduction can be done. However, if dimensionality reduction cannot be done, the decision tree may grow very large, and there may not be enough data to train every decision node. A final possibility is to use the spatial and spectral features in a band grouping algorithm [7]. As indicated above, band grouping is a very good way to handle high dimensionality because it breaks the features into smaller groups that can be handled independently. Furthermore, when band grouping is combined with spatial features, it has additional advantages. First, there are several problems where spatial features may be better at distinguishing bands from each other than spectral features. Second, even in cases where spectral information is good at differentiating bands, spatial information also usually tends to be very good in real images. Thus even in the problems where spectral features are best, there are usually spatial features that are not much worse. This is best illustrated in an experiment done in developing this paper where pecan trees were distinguished from background trees using band grouping with a multi-classifier approach. In the experiment, spectral features had a correlation matrix mean of .987, while spatial features had a correlation matrix mean of .954. The difference was due to higher values farther away from the correlation matrix diagonal, which means that there is the potential for larger groups. Both methods did extremely well with the spectral only band grouping technique getting an accuracy of 99% and the spatial only band grouping technique getting an accuracy of 98%. It is important to note that the image had a 1 meter spatial resolution, so there was potential for good spatial features. Third, there are a large number of spatial features, so there is greater variety with using spatial features for band grouping. Thus, there is the potential

65

that suitable types of features can be found for the problem. Fourth, many spatial features contain some positional information about the scene that is not used in spectral only processing. Positional information such as what parts of the image have similar texture or spectral values is a very powerful way to distinguish bands from each other.

4.4   Spatial-Spectral Band Grouping with a Single Classifier

The 2011 IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS) conference paper that introduced the idea of spatial-spectral band grouping goes into detail about using a single classifier [7]. However, for convenience the information is summarized and expanded on here.

In the case, where only one classifier is used, the purpose of the band grouping is to perform a dimensionality reduction of the feature space. Typically this is a lossy dimensionality reduction. Thus, grouping bands that are highly correlated is important to prevent excessive information loss in the dimensionality reduction. If the bands are highly correlated, then the information lost is more likely to be associated with noise in the image. Because of the problem with information loss, this technique is not useful in all situations. The advantages of this technique are that it is simpler than the multi-classifier approach, it makes it easier to visualize the image since the image is summarized in a few features extracted from highly correlate bands, and the dimensionality reduction is unsupervised, which means that no training data is needed. The four steps in this method are listed below.

1. Extract a spatial feature (or features) from each band in the HSI. The spatial feature can be a single value or multiple values arranged in a vector, 2D matrix, or even a higher dimensional feature. The possible dimensionality increase is not a

concern at this point since these features will possibly only be used to group the bands. There are a great number of spatial features that can be used in this step, and each feature is likely to yield slightly different band groups. The great number of potential features gives the technique great flexibility. A few examples of feature that can be extracted from a band are edges via a Prewitt or Sobel high pass filter [5], entropy via gray level co-occurrence matrix [5], and supervised threshold of the bands. One more feature that is not usually considered to be a spatial feature but is when positional information is used in the clustering step is the spectral values themselves.

2. Cluster the bands based on the spatial features. This is the step where the bands are organized into band groups. Depending on the spatial feature extracted in the first step, it may be possible to utilize positional information in the clustering. If the original HSI has M spectral bands, positional information can be used by organizing the data in an M x N (N is the number of pixels in the image), and clustering the row vectors. There are several different clustering techniques that can be used in this step. Some such as k-means do not force contiguous band groups, while others do force contiguous groups. More research should be done to determine if one type of clustering is preferable than the other, but this is likely problem specific. Once band groups are determined by the clustering, the next question is what to put in the band groups to represent the bands. The original idea put forth in the WHISPERS paper was to place the spectral values for each band in the groups. This was the "spectral" part of the name "spatial-spectral band grouping." However, there are more options than just the spectral values. If a

67

spatial feature (or set of features) is extracted for each pixel for each spectral band, the spatial features may be placed in the band groups. It is important to note that increasing dimensionality here is probably not a good idea because in the next step the dimensionality is reduced, and adding more features may make it more difficult to reduce the dimensionality especially if a supervised dimensionality reduction technique is use. If it is desirable to use both spatial and spectral features at the same time in classification, the number of groups can be doubled, and the spatial features used in half the groups and the spectral features used in the other half of the groups. In the multi-classifier technique, there is the option to place the spatial and spectral features derived from the same bands in the same groups together. Thus, the number of features per band group would be increased. However, there are problems with this idea in a single classifier situation. First, the spatial and spectral features are likely not to have the same scale, which could produce a large discontinuity and make it harder to reduce the dimensionality. Second, increasing the dimensionality within groups means that more training data must be considered if a supervised dimensionality reduction technique is used.

3. Use the band groups to reduce dimensionality. In the single classifier approach, this is a very critical step in the algorithm. As indicated above, the dimensionality reduction is typically lossy, so it is important to choose a technique that preserves as much information as possible. This constraint may be alleviated if the bands in the groups are highly correlated, and if the bands are very highly correlated, the information lost may be associated with noise in the image. Two examples of

68

simple ways to reduce dimensionality of the bands are averaging the bands in each group and choosing a single band from each group to represent the group. These techniques are most likely to lose information than more complicated techniques, but they are very fast to implement and execute. More complex techniques can be done to avoid information loss such as Principal Component Analysis (PCA)[5], and Linear Discriminate Analysis (LDA) [5]. There are a great number of techniques that have been developed to reduce dimensionality of HSI that can be used in this step. It is important that the number of features representing each group be less than the total number of features in the groups, so that there will be fewer features used for classification in the next step.

4. The final step is to use the features representing each group in some sort of classifier to get a classification. There are a great number of classifiers that can be used here because almost any classifier will work. In the research this paper is based on, maximum likelihood was always used because it is a simple and effective classifier to use. A few other classifiers that can be used are Support Vector Machines [5], Artificial Neural Networks [6], Nearest Mean, and Nearest Neighbor [5].

4.4.1   Examples of Spatial-Spectral Band Grouping with One Classifier

As indicated above one of the challenges with spatial-spectral band grouping using a single classifier is that data can be lost when reducing the dimensionality of the features in the groups. This challenge is shown in the example presented here.

In the example, the Indian Pines dataset was used. Indian Pines is a dataset available at the University of Purdue web site. It was obtained using the AVIRIS sensor

and has 220 spectral bands between 400 nm and 2500 nm. In the experiment, several different spatial features were compared to uniform partitioning , and spectral correlation. The spatial features tested were entropy computed by the gray level cooccurance matrix method, edge gradient computed using a Prewitt filter, and a supervised thresholding. After the groups were created, the dimensionality was reduced by averaging the spectral values for the band within the groups. This means that information was likely lost in dimensionality reduction. The classifier was a maximum likelihood classifier. The techniques were compared by varying the number of band groups and computing the average overall accuracy for each group number using 50 ensembles. In each ensemble, 200 random samples were chosen from each class and 100 were assigned for training and 100 were assigned for testing. This allowed the standard deviation to be determined also, which did turn out to be usually in the range of 1 to 2 percentage points.

The results for this example are summarized in figures 4.2a and 4.2b. The results are somewhat surprising because uniform partitioning was as good as any other band grouping technique and responded to Higgs phenomenon a little less than the other techniques. However, the data does not reveal a best method to use since the better techniques are within 1 to 2 standard deviations of each other at their best number of band groups. The lack of separation between techniques is likely because information was lost in the dimensionality reduction step, so the advantage in using the more advanced band grouping technique was lost. This result combined with the results with the multi-classifier approach illustrates the need to use a better technique for taking advantage of the band groups than simply averaging the features. There are many ways that the dimensionality reduction can be done that will represent the groups better than

70

averaging. Some examples are LDA and PCA, which get a weighted averaging of several features. Perhaps if a technique such as this were used, there would be more separation between the classes.
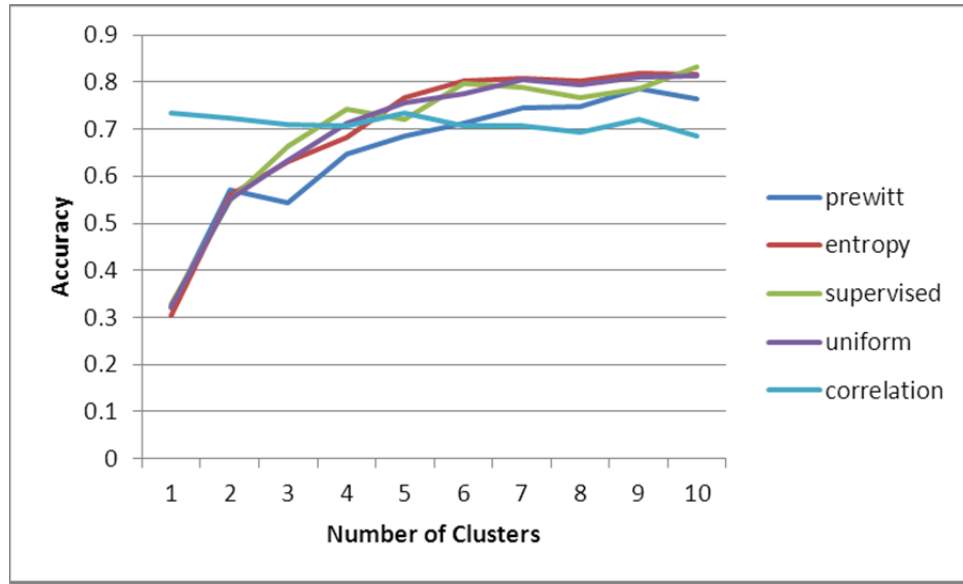
A.
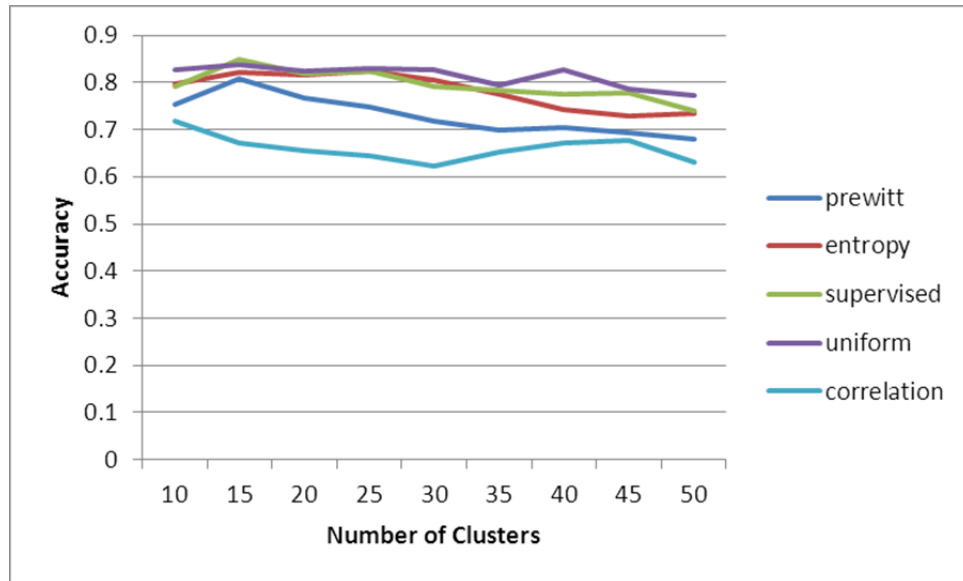
B.

Figure 4.2   Accuracy vs. Number of Clusters for single classifier.

It is important to note that the spectral correlation technique malfunctioned for clusters less than about 5.

71

4.5   Spatial-Spectral Band Grouping with Multi-Classifiers

The steps in spatial-spectral band grouping with multi-classifiers are very similar to the steps in the single classifier algorithm except that each band group is the input to a classifier. The advantage of the multi-classifier approach is that there is no lossy dimensionality reduction. The steps for the algorithm using multiple classifiers are listed below.

1. Cluster the bands based on a spatial feature extracted from the spectral bands. This step is identical to the first step in the single classifier algorithm. As in the single classifier approach, any gray level feature extraction technique can be used to extract the spatial features. The spatial features can be a single value for each band, a vector, a matrix, or even a higher dimensional feature. Different features will produce different clusterings. Thus, it may be useful to try different features for a problem because some features may work better than others.

2. The next step is to group the bands based on the clusters attained from the spatial features. This is again very similar to the single classifier approach. However, there is one more option available when using the multi-classifier approach. As in the single classifier approach, the groups can contain the original spectral values or the spatial feature values. Test may determine that one set of features is better than the other for the particular problem. It is also possible to use both the spatial and spectral features together. In the single classifier approach the best way to do this is to concatenate groups containing spectral and groups containing spatial features together. This doubles the number of groups, which may or may not be a good thing depending on the number of training samples and problem specific

72

factors in the single classifier approach. In the multi-classifier approach, the problem with increasing dimensionality is not as severe since there are fewer features that need to be considered by any classifier. A further advantage of using multi-classifiers is that the spatial features extracted from the bands in each group can be combined with the spectral features within each group. That is, the spatial and spectral features can be interleaved. In the single classifier approach, this caused problems because the dimensionality was increased in each band group, which was contrary to the final objective. However, in the multi-classifier approach, there is less of a problem if there is enough training data to support the additional features in the groups. Furthermore, it may be beneficial to process the spatial and spectral features together since they are definitely related.

3. The third step is where the difference between using a single classifier and multiple classifiers becomes most apparent. In this step, the bands in each group are processed by a classifier. Each band gets its own classifier. It is also possible to use multiple classifiers per group, but this has never been tested, so the utility of such an approach is unknown. The classifiers can be any type of classifier. Some common examples are maximum likelihood, nearest mean, nearest neighbor [5], support vector machine [5], and artificial neural network [6].

4. In the final step, the classification results of each of the classifiers are fused together. There are many different ways to do this decision fusion. A few techniques are majority vote, linear opinion pool, and log opinion pool [2]. There are many more different techniques available, so the researcher will have to determine which one is best for the problem.

73
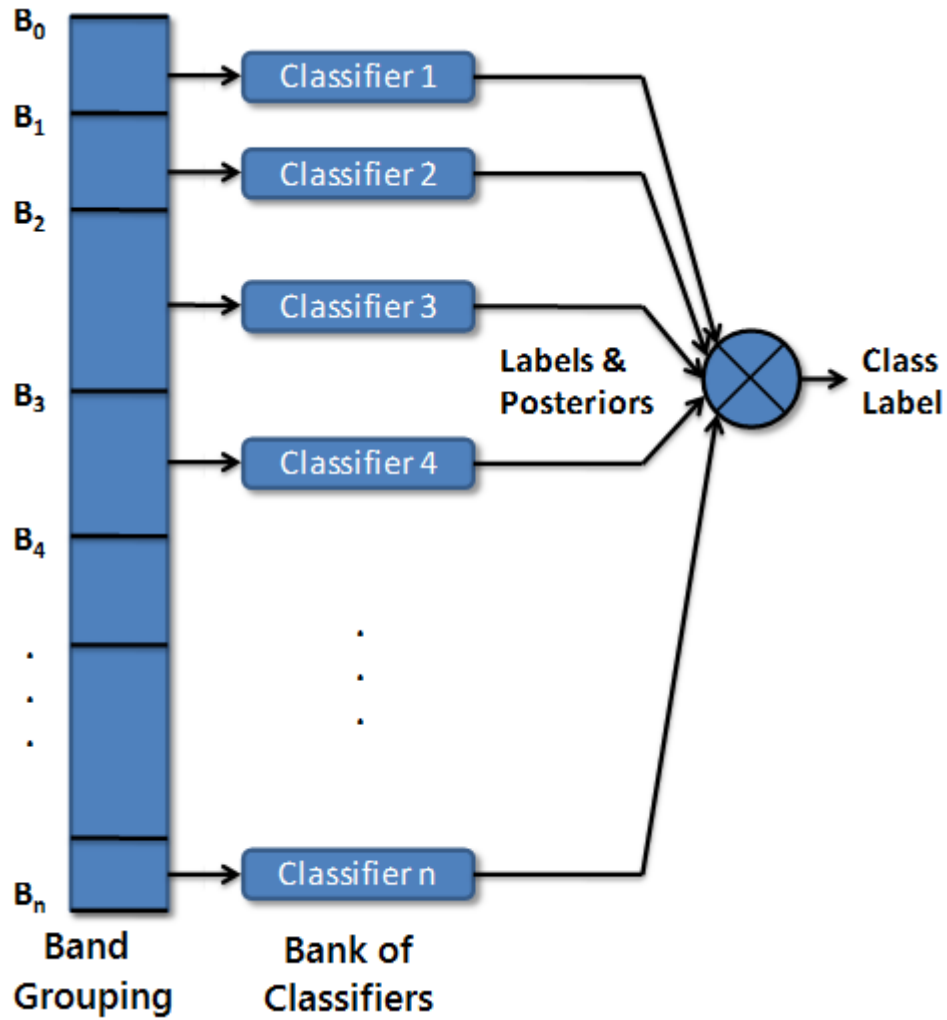
Figure 4.3   Flow Chart of MCDF Strategy.

The spectrum is broken into band groups which then are input into a classifier. The classifier optimizes the features using training data and makes a decision based on the input. The output from the classifiers are then fused into a single label.

4.5.1   Examples of Spatial-Spectral Band Grouping with Multiple Classifiers

Figure 4.4   Image of Indian Pines data set.

Figure 4.5   Image of Pecan 1 data set.

In this section, two data sets that benefit from the use of spatial-spectral band grouping with multiple classifiers will be discussed. One is the Indian Pines data set (see figure 4.4) [8], which was collected by researchers at Purdue University, and the other data set was collected by researchers at Mississippi State University on a test plot called Pecan 1 (figure 4.5). Indian Pines has a spatial resolution of about 4 meters, and was

captured by AVIRIS. There are 220 spectral bands ranging from 400 to 2500. There are several fields in Indian Pines that contain corn, soybeans, oats, hay and wheat. There are a few other classes that have been segmented. These include grass, buildings, roads, and woods. The researchers were studying different tilling practices for the corn and soybeans, so there are three different tills for each and the image was captured while both were just sprouting. Pecan 1 contains a single field with 1 meter spatial resolution. It was captured using a SpectIR sensor, which has 128 spectral bands ranging from 400 to 994 nm with a nominal spectral resolution of 5nm. The original intent of Pecan 1 was to study detection of herbicide in corn crops, so transects of the field were treated with one of 7 different concentrations of herbicide. Though both of the data sets were originally intended for agricultural research, they have significant differences. In Indian Pines, the spatial resolution was about 4 meters, but in Pecan 1, the spatial resolution was about 1 meter. Thus Pecan 1 should have much better texture information. Indian Pines however covers more area than Pecan 1, which covers just one field, so it has more diverse spatial features. Indian Pines was collected very shortly after the crops were planted, and thus the crops are just sprouting, but Pecan 1 contains mature crops.

The multi-classifier approach described above was used for both data sets. The spatial feature used in the clustering part was entropy, which was calculated using the gray level co-occurance matrix technique [9]. The radius of the region of interest (ROI) around the pixels was varied to determine the best size. Using a small ROI results in more localized measurements of the texture that is characterized by more noise, while useing a larger ROI results in less noisy texture measurements, but the information is less localized. Since both traits are highly desirable, there is an optimal radius that is

76

dependent on the quality of the texture information and spatial resolution of the HSI. In addition to varying the radius of the spatial feature, the features used by the classifiers was varied. As indicated in the second step of the method, the features used by the classifiers can be the spectral features or the spatial features. Furthermore, both types of features can be used simultaneously either by forming additional groups containing the spatial features, or by integrating the spatial features into the corresponding spectral feature groups. In the case of both datasets, maximum likelihood was used for classification because it is commonly used in the remote sensing community, is simple to implement, and provides a good baseline comparison for other techniques. The classifier outputs for all the groups were fused using a simple majority vote. The final decision accuracies were compared to the results using a uniform partitioning and a common unsupervised band grouping technique called spectral correlation band grouping. In all cases, the number of band groups was forced to be 5. Since there were four different radii and four ways the spectral and spatial features were used in this study, 16 sets of results were obtained. A summary of these results is provided below.

4.5.2   Indian Pines Data Set Results

The Indian Pines results reveal two major trends. The first trend is that spatial-spectral band grouping, as compared to uniform windowing and spectral band grouping, consistently produced the best overall accuracy in every configuration. The best overall accuracy (77.40%) for the Indian Pines data set was obtained using a 9x9 ROI using both the spatial and spectral features independently in their own groups (see table 4.1b). The overall accuracy using uniform partitioning was significantly lower than spatial-spectral band grouping techniques at 70.49% but was higher than the spectral correlation

77

technique which obtained 67.06%. The second trend reveals that the multi-classifier system is slightly sensitive to whether spectral and/or spatial features were used. On average, using spatial and spectral features independently in separate groups produces slightly better overall accuracy (on average it is 76.26%). The average overall accuracy of using only the spectral features is second at 75.62%, which is followed by using the spatial and spectral features in the same groups at 75.33%. Using spatial features had the worst average overall accuracy at 73.55%.

Table 4.1   Average confusion matrices using maximum likelihood classification, majority vote decision fusion, and 100 training samples per class.

	corn-no till	corn-min till	soybeans-no till	soybeans-min till	soybeans-clean till	grass/ trees	woods	
corn-no till	69.4	9.6	9.2	6.5	4.8	0.5	0	69.4%
corn-min till	8.8	74.6	2.2	7.3	7.1	0	0	74.6%
soybeans-no till	10.5	2.2	71.1	11.5	3.5	1.2	0	71.1%
soybeans-min till	15.4	17.7	16.1	41.2	8.6	1	0	41.2%
soybeans-clean till	5.4	7.6	0.5	1.3	84.6	0.6	0	84.6%
grass/ trees	0.1	0	0.8	0	0.1	98	1	98.0%
woods	0	0	0	0	0	2	98	98.0%
	63.5%	67.0%	71.3%	60.9%	77.9%	94.9%	99.0%	76.7%

A. 9x9 entropy spatial feature for clustering with only spectral features used for classification.

78

Table 4.1 (continued)

	corn-no till	corn-min till	soybeans-no till	soybeans-min till	soybeans-clean till	grass/trees	woods	
corn-no till	71.3	5.9	10.5	6.2	5.3	0.8	0	71.3%
corn-min till	7	71.7	1.8	11.1	8.3	0.1	0	71.7%
soybeans-no till	11.4	1.5	73.1	9.8	2.9	1.3	0	73.1%
soybeans-min till	13.6	10.7	18.5	44.3	12	0.9	0	44.3%
soybeans-clean till	4.8	6	1.3	1.7	85.2	1	0	85.2%
grass/trees	0	0	0.6	0	0.1	97.9	1.4	97.9%
woods	0	0	0	0	0	1.7	98.3	98.3%
	66.1%	74.9%	69.7%	60.77%	74.9%	94.5%	98.6%	77.4%

B. 9x9 entropy spatial feature for clustering with spatial and spectral features used in independent groups for classification.

	corn-no till	corn-min till	soybeans-no till	soybeans-min till	soybeans-clean till	grass/trees	woods	
corn-no till	63.8	10.4	12.2	9.2	3.9	0.5	0	63.8%
corn-min till	14.4	62	4	6.7	12.9	0	0	62.0%
soybeans-no till	14	4	63.9	13.2	4	0.9	0	63.9%
soybeans-min till	23.4	12.1	21.5	27.9	13.5	1.6	0	27.9%
soybeans-clean till	3.8	14.6	0.9	0.9	79.4	0.4	0	79.4%
grass/trees	0	0	0.3	0	0.1	98.2	1.4	98.2%
woods	0	0	0	0	0	1.8	98.2	98.2%
	53.5%	60.9%	62.4%	48.1%	70.0%	95.1%	98.6%	70.5%

C. Uniform Partitioning with only spectral features used for classification.

79

Table 4.1 (continued)

	corn-no till	corn-min till	soybeans-no till	soybeans-min till	soybeans-clean till	grass/trees	woods	
corn-no till	53.8	17.7	10.3	12.1	4.7	0.8	0.6	53.8%
corn-min till	11.2	69.5	3.8	9.1	6.2	0.2	0	69.5%
soybeans-no till	14.3	9.4	53.3	16.6	4.8	1.5	0.1	53.3%
soybeans-min till	19	22	16.8	30.7	9.9	1.4	0.2	30.7%
soybeans-clean till	8.4	20.4	2.3	3.3	64.9	0.6	0.1	64.9%
grass/trees	0.2	0	0.2	0	0	98.7	0.9	98.7%
woods	0	0	0	0	0	1.5	98.5	98.5%
	50.7%	50.8%	62.2%	42.5%	72.3%	94.3%	98.1%	67.0%

D. Spectral correlation with only spectral features used for classification.

A close examination of the confusion matrices reveals that the most difficult class is the soybeans with minimum till. Even using the best technique, the producer accuracy of this class was 49%, which the only of seven classes that had a producer accuracy of less than 70%. This class is significantly confused with the corn and soybean classes. Most likely this is because the image was taken very early in the growing season, so the crops are just emerging from the soil. Thus, in the corn and soybeans, the spectral signatures contain a great deal of soil. The soybean min-till is confused the most with the soybean no-till class followed by the corn no-till class. It is expected that soybean min-till should be confused the most with the other soybean classes because they both contain the

80

same basic signatures: soybeans and soil (tilling the soil does change its spectral signature, so this is not exactly the same). However, the soybean clean-till class is confused less with soybean min-till than either corn class. There are likely two reasons for this trend. First, the soybean clean-till is the least confused of all the corn and soybean classes because it is likely that the clean till in the field has the most significant impact on the spectral reflectance of the field no matter what the crop is at this early point in the growing season. Second, since there is likely some spatial correlation in the soil, proximity might influence confusion. The distances between the fields of soybean min-till and soybean clean-till are greater than the distances between fields containing soybean min-till and the corn classes. There is even a plot of corn no-till that boarders a soybean min-till field. This might also explain why corn no-till is more confused with soybean min-till than corn min-till. The accuracy of soybean min-till was the main thing that separated the good classifiers. Using the uniform partitioning, an accuracy of 27.9% was achieved, and using the spectral correlation band grouping, an accuracy of 30.7% was achieved. Both of these accuracies were well below the worst accuracy for a spatial-spectral technique, which was 36.4%, and even this accuracy is atypical for the spatial-spectral techniques because the mean accuracy for the soybean min-till class was 43.3%.

4.5.3   Pecan 1 Data Set Results

The Pecan 1 data set was captured in the summer of 2008 at a Mississippi State University experiment station near Brooksville, Mississippi. It was part of a series of experiments that determined the detection capabilities of herbicide drift in corn and wheat. The field the data used in this experiment was a corn field next to a pecan tree grove, and thus is often referred to a "Pecan 1" even though the experiment was done on

corn. The chemical herbicide used in the experiment was called glyphosate, which is the active ingredient in the herbicide called "Roundup." In typical herbicide drift scenarios either wind (if applied on a windy day) or water (if applied on a rainy day) will carry the herbicide from the intended target toward unintended plants. In a typical herbicide drift situation, the concentration of the chemical falls off with distance from the intended spray area. Figure 4.6a shows the typical dispersal pattern for a windblown herbicide drift event. However, in the interest of accounting for differences in fertility of agricultural fields, the corn field was sprayed in the pattern shown in figure 4.6b.
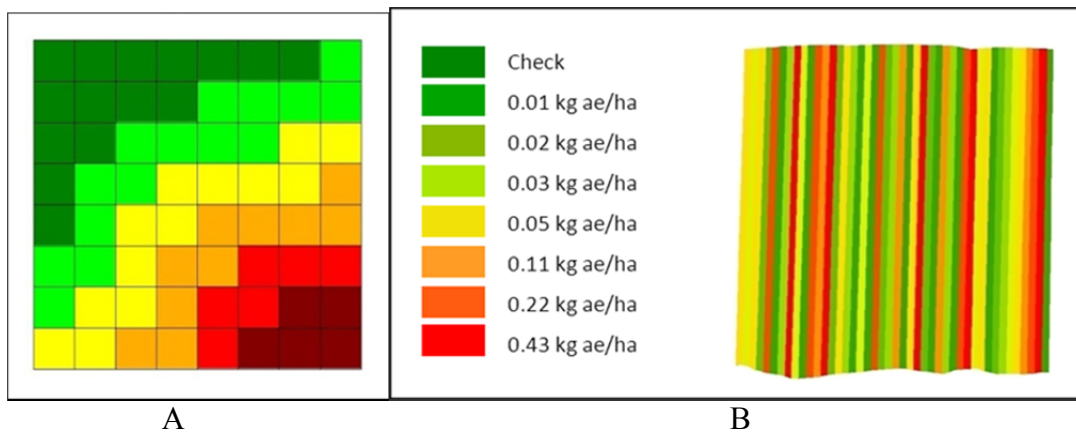
Figure 4.6   Simulated wind driven herbicide drift and spray pattern used on one third of Pecan 1

Hyperspectral data was collected from the field using two primary sources. The first source was the use of a handheld hyperspectral sensor that was deployed on a tractor driven through the corn field. The handheld sensor collected 2151 spectral samples in the range from 300 - 2500 nm, and was mounted on the tractor so that the sensor was aimed at the canopy of the corn where the shadow from the tractor would be minimized (see

82

figure 4.7). Though the handheld hyperspectral data was not used in this particular experiment, the tractor was also equipped with a differential GPS that was accurate down to a few centimeters, which was later used along with the notes taken by the researchers to identify the ground truth in the airborne imagery. The airborne imagery was collected by a ProSPecTIR-VNIR system with a spectral range of 400-994 nm spectral range. In this range, 128 spectral samples were collected with a spatial resolution of 1 m. Figure 4.5 shows a true color image of the corn field where the study was conducted.

Figure 4.7   Image of collection of handheld hyperspectral data mounted on a tractor. The sensor is mounted at the end of the white boom deployed to the left of the tractor to avoid the shadow of the tractor influencing the data.

The general trends for the Pecan 1 data set were very similar to the results obtained from the Indian Pines data set. As with the Indian Pines data set, the best overall classification accuracy was obtained using both the spatial and spectral features independently in their own groups. This technique was able to obtain a maximum overall accuracy of 78.77% when 7x7 ROI was used (see table 4.2). Using the uniform

83

partitioning, the overall accuracy was much lower at 63.91%, while using the spectral correlation technique produced an overall accuracy of 60.09%.

The results suggest that using a larger ROI may improve the results further, but this was not tested since the areas that contained the same spray rate were approximately 4 to 5 pixels wide and bordered areas with different spray rates. Thus, if the ROI is increased further, the adjacent areas would begin to influence the spatial feature used in band grouping. While this may not be a problem for the band grouping algorithm, it may mean algorithms that use the spatial features in the classification also may be influenced by spatially adjacent classes. The results also imply that unlike the Indian Pines data set, the spatial features are better than the spectral features for overall accuracy in the Pecan 1 data set. This is apparent because the mean overall accuracy when only the spectral features were used for classification was 66.57%, while the mean overall accuracy when only the spatial features were used for classification was 73.58%. However, the overall accuracy does not tell the whole story. There were classes that spectral features yielded better results, and features that spatial features yielded better results. The first 3 classes illustrate this well. In the first class, which was the control spray rate (no herbicide), the spectral only results were on average 72.9% accurate, but the spatial only results were on average 49.4% accurate. This means that the spectral features are clearly better for the control class. However, the classes with 1/32 and 1/16 concentrations of herbicide did much better with spatial features. On average, the spectral only algorithm resulted in a 53.17% producer classification accuracy for the 1/32 concentration class, and 41.87% producer classification accuracy for the 1/16 concentration class. The spatial only algorithm did much better at 79.00% for the 1/32 concentration and 100.00% (it never

84

made an error in a total of 30 different tests using 3 different ROI sizes) for the 1/16

concentration. These results help explain why the combined use of spectral feature

groups and spatial feature groups in a multi-classifier setup is advantageous.

Table 4.2   Confusion matrices for the Pecan 1 dataset.

Glufosinate Mixture	0	1/32	1/16	1/8	1/4	1/2	1	
0	55.6	33.8	4.2	4.3	1.6	0.3	0.2	55.6%
1/32	3.8	83	0	9.2	2.7	1	0.3	83.0%
1/16	0	0	100	0	0	0	0	100.0%
1/8	4.9	18.7	0	71.2	4	1.1	0.1	71.2%
¼	0.8	1.7	0	5.5	82.5	7.9	1.6	82.5%
½	0	0.1	0	0	9.2	70.4	20.3	70.4%
1	0	0	0	0	0.9	18.6	80.5	80.5%
	86.1%	60.5%	96.0%	79.0%	81.9%	71.2%	78.4%	77.6%

A. 7x7 entropy spatial feature for clustering with only spectral features used for classification.

Glufosinate Mixture	0	1/32	1/16	1/8	1/4	1/2	1	
0	63.6	27.6	2.5	3.4	2.2	0.1	0.6	63.6%
1/32	8	78.5	2.6	6.8	2.5	0.8	0.8	78.5%
1/16	0.4	0.4	99.2	0	0	0	0	99.2%
1/8	3.2	16.7	2.4	72	5	0.3	0.4	72.0%
1/4	0.3	0.7	0.1	4.6	85	6.7	2.6	85.0%
1/2	0.2	0	0	0	10.6	70.9	18.3	70.9%
1	0	0	0	0	0.1	17.7	82.2	82.2%
	84.1%	63.5%	92.9%	83.1%	80.7%	73.5%	78.7%	78.8%

B. 7x7 entropy spatial feature for clustering with spatial and spectral features used in independent groups for classification.

Table 4.2 (continued)

Glufosinate Mixture	0	1/32	1/16	1/8	1/4	1/2	1	
0	72.2	16.5	7.6	1.1	2	0.4	0.2	72.2%
1/32	19.6	48.3	21.5	6.1	3	1	0.5	48.3%
1/16	10.9	24.6	47.6	11.9	3.8	1.2	0	47.6%
1/8	7.3	11.5	16	58.1	6.3	0.7	0.1	58.1%
¼	1.4	0.9	1.8	7.7	77.7	7.6	2.9	77.7%
½	0.1	0	0	0	9.6	64.4	25.9	64.4%
1	0	0	0	0	1.5	19.4	79.1	79.1%
	65.2%	47.4%	50.5%	68.7%	74.9%	68.4%	72.8%	63.9%

C. Uniform Partitioning with only spectral features used for classification.

Glufosinate Mixture	0	1/32	1/16	1/8	1/4	1/2	1	
0	74.6	13.9	5	3.1	2	0.2	1.2	74.6%
1/32	21.1	53.2	13.4	4.8	3.8	0.5	3.2	53.2%
1/16	13.1	33.8	26.9	17.8	4.4	1.8	2.2	26.9%
1/8	13	11.9	7.4	61.7	4.2	0.9	0.9	61.7%
¼	0.4	2.2	5.4	10.2	69.4	6.4	6	69.4%
½	0	0.3	0.8	0	13.4	58.1	27.4	58.1%
1	0	0.1	0.4	0	3.9	18.9	76.7	76.7%
	61.2%	46.4%	46.1%	64.2%	69.1%	67.1%	65.6%	60.1%

D. Spectral correlation with only spectral features used for classification.

The matrices were computed using maximum likelihood classification, majority vote decision fusion, and 100 training samples per class.

A separate study done by the authors focused on determining how soon after the herbicide drift event the herbicide could be detected in the corn using the handheld hyperspectral data. In this study, the hyperspectral data was collected 1, 4, 8, and 14 days after the herbicide was sprayed on the corn. It happens that the imagery was also collected on day 8 (a fact that can be verified by the presence of the white trucks parked partially under a pecan tree in the full SpecTIR image). The results of the handheld study are presented in table 4.3 for comparison purposes. Upon comparison of the results from day 8, it becomes immediately apparent that the total accuracy shows a very significant

86

improvement over all other techniques when using Spatial-Spectral Band Grouping. One trend that has become apparent after many studies on the handheld data and imagery is that the imagery usually produces better results. This may be due to the time it takes to collect the handheld data, which is approximately 5 hours. Although white references were taken regularly every few minutes, there is likely more atmospheric variation uncompensated for in the handheld data since the imagery of the whole field was collected in just a few seconds, which leaves less time for atmospheric variation. Fortunately, three of the experiments were repeated for the imagery thus can be compared directly to the results from this study. The experiments that were repeated are Principal Component Analysis (PCA) with a single classifier (15% overall classification accuracy), SLDA with a single classifier (60% overall classification accuracy), and Supervised Discrete Wavelet Transform - Multi-Classifier Decision Fusion (DWT - MCDF) (65% overall classification accuracy). Thus, even when comparing Spatial-Spectral Band Grouping to other techniques using the hyperspectral imagery, a significant improvement in overall accuracy is observed.

Table 4.3   Total accuracy for several techniques using hyperspectral handheld data.

# Days After Treatment	PCA Single Classifier	Unsupervised Band Grouping Single Classifier	Unsupervised Band Grouping MCDF	SLDA Single Classifier	Supervised Band Grouping Single Classifier	Supervised MCDF	Supervised DWT-MCDF
1	11%	12%	21%	13%	19%	32%	36%
4	14%	14%	28%	14%	27%	39%	42%
8	13%	14%	32%	17%	33%	50%	52%
14	14%	17%	44%	20%	41%	61%	64%

Day 8 is highlighted because it corresponds to the date of the SpecTIR imagery.

87

4.6   Conclusion

This work describes a newly developed spatial-spectral band grouping technique, and presents two different strategies for its use. Spatial-spectral band grouping (as with other types of band grouping) can be utilized with a single or multiple classifiers. In the single classifier approach, dimensionality reduction is used to reduce the number of features in each group and then a single classifier makes a decision based on the combined set of reduced features. In the multi-classifier approach, the features of each spectral band group are processed with an individual classifier, and then the decisions of the classifiers are fused. The results of the single classifier approach show the necessity of using a more advanced method of reducing the dimensionality of the groups than simply averaging them. It may perhaps be more advantageous to use some sort of linear combination of the features such as LDA, PCA, or a kernel-based approach, or use a multi-classifier approach. The multi-classifier approach examples demonstrate that spatial-spectral band grouping is advantageous and significantly outperforms spectral only band grouping.

The methods were quantitatively assessed by comparing the overall classification accuracy to several different techniques on the same data. Spatial-Spectral Band Grouping shows a significant increase in overall classification accuracy over all the other techniques.

4.7 References

[1]   D. Manolakis, R. Lockwood, and T. Cooley, "On The Spectral Correlation Structure of Hyperspectral Imaging Data," *Proceedings of Internatinal Geoscience and Remote Sensing Symposium (IGARSS),* pp. II-581-II-584, 2008.

[2]   S. Prasad and L. M. Bruce, "Decision Fusion With Confidence-Based Weight Assignment for Hyperspectral Target Recognition," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, pp. 1448-1456, 2008.

[3]   S. Prasad, L. M. Bruce, and H. Kalluri, "A Robust Multi-Classifier Decision Fusion Framework for Hyperspectral, Multi-Temporal Classification," *Proceedings of Internatinal Geoscience and Remote Sensing Symposium (IGARSS),* pp. II-273-II-276, 2008.

[4]   J. A. Benediktsson and J. R. Sveinsson, "Multisource remote sensing data classification based on consensus and pruning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, pp. 932-936, 2003.

[5]   R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification, 2 ed.* New York City, New York: John Wiley and Sons, Inc., 2001.

[6]   P. N. Stuart J. Russell, *Artificial Intelligence A Modern Approach*. Upper Saddle River, New Jersey: Prentice Hall, 2003.

[7]   M. A. Lee, L. M. Bruce, and S. Prasad, "Concurrent spatial-spectral band grouping: Providing a spatial context for spectral dimensionality reduction," *Proceedings of Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pp. 1-4, 2011.

[8]   (1994), MultiSpec: A Freeware Multispectral Image Data Analysis System. Available: https://engineering.purdue.edu/~bieh/MultiSpec/hyperspectral.html

[9]   R. C. Gonzalez and R. E. Woods, *Digital Image Processing, 2 ed.* Upper Saddle River, New Jersey: Prentice Hall, 2002.

[10]   Robert A. Schowengerdt, *Remote Sensing - Models and Methods for Image Processing, 3rd ed.* Academic Press, 2006.

[11]   A. Cheriyadat, L. M. Bruce, "Decision Level Fusion with Best- Bases for Hyperspectral Classification", *Proc. IEEE GRSS Workshop on Advances in Techniques for Analysis of Remotely Sensed Data*, October 2003.

[12]   S. Venkataraman, L.M. Bruce, "Hyperspectral Dimensionality Reduction via Localized Discriminant Bases," *Proc. IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, Seoul, Korea, July 25-29, 2005.

[13]   T.R. West, S. Prasad, L.M. Bruce, D. Reynolds, "Utilization of Local and Global Hyperspectral Features via Wavelet Packets and Multiclassifiers for Robust Target Recognition," *Proceedings of IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, Cape Town, South Africa, July 2009.

[14]   T.R. West, S. Prasad, L.M. Bruce, D. Reynolds, T. Irby "Rapid Detection of Agricultural Food Crop Contamination via Hyperspectral Remote Sensing," *Proceedings of IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, Cape Town, South Africa, July 2009.

[15]   T.R. West, S. Prasad, "Wavelet Packet Tree Pruning Metrics for Hyperspectral Feature Extraction," *Proceedings of IEEE Geoscience and Remote Sensing Symposium (IGARSS)*, Boston, MA, July 2008.

[16]   Darrell Wesley Johnson, "Assessing Resolution Tradeoffs of Remote Sensing Data via Classification Accuracy Cubes for Sensor Selection and Design," Masters Thesis, Mississippi State University, 2006.

CHAPTER V

CONCLUSIONS

This dissertation developed two new digital image processing methods for
hyperdimensional imagery and tested them on natural remotely sensed Earth images and
medical images. The hyperdimensionality of the imagery was either due to the sensor
used in capturing the image or derived from an image via preprocessing and feature
extraction.

In chapter 3, the hyperdimensionality was derived from synthetic aperture radar
images of the Mississippi River levee system and digital mammography images. The
techniques used for the levee system were developed from the techniques used in the
mammography example. As in the mammography example, the new techniques improve
target recognition. Furthermore, the new techniques demonstrate the omni-directional
property as predicted. The new methods also have the advantage that they can be used
without a segmentation algorithm because they are based on the concept of extracting
circular regions of a predetermined radius. This is different from the standard gray level
co-occurrence matrix (GLCM) process, where rectangular regions are always extracted.
Once a circular region is extracted, it is resampled radially in order to create a rectangular
region, which the standard GLCM can be computed from. Since this new method
computes a GLCM, many different features can then be derived from it. Thus, the new
features are omni-directional versions of the standard types and will respond to textural

91

patterns regardless of their directional components. This was shown to be useful in cases where the directional components of the texture had an unknown or unpredictable direction, or the direction was not constant throughout the image.

The fourth chapter developed and tested a new method that uses spatial and spectral information in a hyperspectral image concurrently. This new technique uses spatial information in each spectral band to group similar bands together. The new technique was tested on agricultural hyperspectral images in automated classification systems that use a single classifier or multiple classifiers with decision fusion. The new spatial-spectral band grouping technique was compared to standard spectral only band grouping techniques such as uniform partitioning and spectral correlation. The experiments show that for a single classifier where bands within groups are averaged together, the new method does improve classification accuracy over spectral correlation. However, when multiple classifiers are used with decision fusion, the classification accuracy improves significantly over both standard techniques.

92