CLASSIFYING NATURAL BACKGROUND SCENES USING SINGULAR VALUE FEATURES

Thesis

Submitted to

Graduate Engineering & Research

School of Engineering

UNIVERSITY OF DAYTON

In Partial Fulfillment of the Requirements for

The Degree

Master of Science in Electro Optics

by

David Michael Cannon

UNIVERSITY OF DAYTON

Dayton, Ohio

December 1993

CLASSIFYING NATURAL BACKGROUND SCENES USING SINGULAR VALUE FEATURES

Approved by:

Steven C. Gustafson, Ph.D.
Advisory Committee
Committee Chairperson

David L. Flannery, Ph.D.
Advisory Committee
Committee Member

Leno M. Pedrotti, Ph.D.
Advisory Committee
Committee Member

Donald L. Moon, Ph.D.
Interim Associate Dean/Director
Graduate Engineering & Research
School of Engineering

Joseph Lestingi, D. Eng., P.E.
Dean
School of Engineering

ABSTRACT

CLASSIFYING NATURAL BACKGROUND SCENES USING SINGULAR VALUE FEATURES

Name: Cannon, David Michael
University of Dayton, 1993

Advisor: Dr. Steven C. Gustafson

This study developed texture extraction techniques for classifying natural background scenes using singular values features. Singular values (obtained using singular value decomposition) were used to produce a reduced one-dimensional feature space of texture attributes of natural scene regions. Scenes with tree, grass, and water regions were taken from FLIR imagery. Classification error was determined using a Bayes error estimate, and Bhattacharyya distance was used to quantify separation of features between regions. Although there were substantial variations within regional texture samples, good classification results were obtained using the singular value features. The highest classification accuracy (100 percent) was obtained when separating grass from water regions. The worst classification accuracy (77 percent) was obtained when separating grass from tree regions. Singular value feature results were also compared with Fourier power spectrum features. The singular value features provided slightly better overall classification results than the Fourier power spectrum features. These results may be refined and used to compare and grade synthetic and real background scenes to support automatic target recognition (ATR) modeling efforts.

ACKNOWLEDGMENTS

This thesis would not have been possible without the help and encouragement of many people. First, I am deeply indebted to Dr. Steven Gustafson for helping me focus and define the initial topic for this research project. He guided, encouraged, and provided me the freedom to learn and reshape the project as it developed. I greatly appreciated his patience and encouragement throughout the effort. Also, I would like to thank Dr. David Flannery and Dr. Leno Pedrotti for their technical advice and careful review of the manuscript.

I would like to extend a special thanks to Jim Leonard from the Air Force Wright Laboratories for sponsoring and providing the resources to carryout this project. Also, I would like to thank the guys in the model-based vision (MBV) laboratory, Vince Velton, Bruce York, and Rick Beam, for providing computer and technical support in extracting images and using the computer environment.

Finally, I wish to express my appreciation to my wife, Kathleen, and the kids, Michael, Bill, and Elizabeth, for tolerating and enduring this "passage of life" and encouraging me to "get-it-done!"

David M. Cannon

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

INTRODUCTION

1.1 Background

The problem of automatic target detection and recognition has been widely studied for over 30 years. Although numerous techniques have been developed in laboratories and other controlled environments, the application of effective techniques in the "real world" falls short. The recent failure of the "Desert Storm" campaign to find SCUD missiles is an important example. Ironically, as more sophisticated sensors and signal processing techniques are employed to find conventional targets, target designers develop more sophisticated ways to conceal targets. By reducing the effective signature of a target through camouflage or signature reduction techniques (visual, IR, radar, acoustic, etc.) or by increasing the clutter or noise background levels (jamming), the target designer effectively remains ahead in reducing the effectiveness of many target sensor systems.

Often it is possible to limit the search for targets to certain most-likely regions. For example, a tank is not likely to be found along a rugged mountainside or in a very dense wooded area but might be found hidden or moving along a treeline. Such *a priori* conditions, i.e., automatically identifying treelines or distinguishing tree from grass regions is an important focus of current scene analysis for automatic target recognition (ATR) research.

Many different approaches to developing synthetic background scenes (faceted models, fractal models, etc.) have been devised to support numerous ATR evaluation models. Techniques to accurately compare the quality of synthetic and real backgrounds are

needed. This study investigates texture feature extraction with singular value features to accurately classify and compare real or synthetic background regions.

Most images of natural scenes can be segmented into regions of different textures. Texture is one of the most important characteristics for identifying objects or regions of interest in an image, whether the image is a photomicrograph, an aerial photograph, or a satellite image. Haralick [1] and Weszka *et al.* [2] used texture features for classifying land-use and terrain categories from satellite imagery. Haralick obtained classification accuracies of 82 percent for aerial photographic images and 83 percent for satellite images using gray-tone spatial dependencies. Weszka *et al.* obtained classification accuracies of 90 percent on terrain samples representing three geological classes and compared texture features based on Fourier power spectrum, second-order gray level statistics, and first-order statistics of gray level differences.

Detecting and identifying targets or background scenes to obtain accurate classification results from a diverse natural world is not only difficult, it is also complicated by severe computational requirements for many real-time applications. Even with relatively fast digital computers, image processing requires trade-offs to process reasonable sized images and obtain timely results. Scalar or vector features are often used for computations to reduce dimensionality and retain representations of regions of interest. Optical image processing could potentially ease the computational load by performing Fourier transforms [3] and other operations optically [4-6].

Over the past three decades numerous techniques have been developed to extract features from signals and images. Nilsson [7] states that there is no general theory that allows the selection of features that are relevant for a particular problem. Thus, the design of feature extractors is empirical and uses many ad hoc strategies, although nature provides some guidance from biological prototypes, such as neural networks. One major problem is handling variations from image to image. Many techniques do not perform well for differ-

ent size, contrast, deformation, skewness, or other conditions. Over 20 years ago Duda [8] called for "rugged features" to represent the "real" world. These rugged features would not change due to normal variations in the image. Current research has branched into moment invariant features [9], fractal-based features [10], stochastic models [11, 12], and other techniques [13] that can function to some extent in noisy environments.

Ashjari [11] developed a stochastic texture measurement method based on the singular value decomposition (SVD) of a texture region. The singular values of a matrix contain information on the correlation content of the matrix elements and their interrelationships. Singular-value features reduce a two-dimensional matrix to a one-dimensional vector that has a unique structure for each class of texture. Ashjari showed that similar-looking texture scenes have similar ordered singular value distribution curves. His method demonstrated high classification accuracy for natural homogenous texture regions, such as close-up images of grass, sand, wool, and raffia. Additional efforts with SVD techniques have shown promise in other feature extraction applications [14-17]. Li Shu-Qiu *et al.* [17,18] reported that a faint electrocardiogram signal of a fetus in a background of strong noise could be extracted using SVD techniques.

This study extends SVD methods demonstrated by Ashjari to classifying natural background scenes from infra-red images. The scenes are taken from Forward Looking Infra-red (FLIR) images of tree, water, and grass backgrounds with targets along a tree-line. The goal is to accurately classify the tree regions from the grass regions. Different ranges are used to characterize singular values (SV) feature sensitivity to variations in the texture regions. Because the images of trees, grass, and water are relatively close range, the application of this study is limited to images used for tactical reconnaissance or tactical air-to-ground search missions. However, these techniques could be expanded to satellite land or terrain classification. They could also be used to compare and grade the quality of synthetic relative to real background regions.

1.2 Problem Statement

The problem addressed here is the classification of natural background regions from FLIR images. Successful and accurate classification would help automatic target detection and recognition systems to search regions where targets are most likely to be found, and could also provide a quantitative method for comparing and grading synthetic images relative to real images. Recently SVD methods have shown promise for extracting features from homogenous natural texture background images [11]. The SVD method is used here with background regions of trees, water, and grass characterized as texture features. Fourier power spectrum feature techniques are also developed and used to compare and evaluate the success of SVD methods.

1.3 Scope of Effort

Although a logical application of this study would be the classification of many kinds of natural texture background regions (such as water, marsh, densely and sparsely wooded regions, etc.), this effort was limited to classifying grass and water regions from tree regions in relatively close range (tactical) scenarios.

The evaluated scenes were limited to infra-red (FLIR) images, but the methods could also be extended to multispectral, visible, radar, laser radar, and other source imagery. Texture variations due to weather and different flight paths were not addressed. Although the numerous FLIR images provided by the Air Force Wright Laboratory Automatic Target Recognition Branch (WL/AARA) provided hundreds of frames of scenes, there were relatively few variations in the foliage background (most of the variations involved changes in the target conditions). Two different flight paths and four different ranges were used for the analysis.

No attempt was made to segment the regions in each image using automatic routines. Although segmentation that preserves target information is often difficult to achieve,

because of shadows (visible) or temperature (IR) differences, it is relatively easy to achieve for large natural background scenes. However, for this study, tree, water, and grass region samples were manually segmented. Care was taken to select homogenous regions of trees, water, or grass that did not include edges. The evaluation of the goodness of extracted features was performed by a Bayesian classifier, and the error criterion was the Bhattacharyya distance measure [11, 12]. The comparison to other feature extraction techniques was limited to Fourier power spectrum features.

This study could provide a tool for evaluating synthetic background from real background scenes, but synthetic imagery (or software to create synthetic imagery) of background scenes similar in texture to the FLIR imagery used is not yet available.

1.4 General Approach

The general approach involves generating and assessing singular value features for natural background scenes using SVD methods. At least 64 samples of *32x32* pixels were desirable from representative regions of tree, grass, and water for evaluation. Since the original images were 8-bit gray scale, they were first subjected to a statistical rescaling process (standardization) to produce normalized images. The Air Force Wright Laboratory Model-Based Vision (MBV) laboratory provided the use of their computer center to perform all analyses. The UNIX and Macintosh computer software versions of MATLAB [19] were used to implement all algorithms and calculations. The MATLAB ***svd*** function [19] generated from the LINPACK routine [20] was used to generate the diagonal row of 32 singular values for each sample. A family of singular value vector and scalar features was generated and assessed for each region. A statistically representative feature set was then calculated and evaluated using a Bayesian classifier. The Bhattacharyya distance was calculated to measure the separability classification accuracy between the two regions.

Fourier power spectrum features were also generated using a ring-wedge sector approach [21, 22] to create a set of 32 features for comparison.

1.5 Thesis Organization

The thesis is organized into six sections. The outline follows the same path used for the research effort.

Chapter 2 is a review of related research and literature. Many approaches and techniques have been implemented to address a broad range of pattern recognition, image analysis, and feature extraction problems. SVD method also provides solutions to many signal and image processing problems that warrant consideration.

Chapter 3 presents a development of the theories and approaches used to classify the natural background scenes. Texture feature extraction fundamentals are described. The motivation and theory of the singular values feature method is developed. Finally, classification accuracy measures are described.

Chapter 4 applies the SVD method to the problem of classifying tree, water, and grass regions from FLIR images. A set of singular value features is calculated and evaluated.

Chapter 5 compares the SVD results to a Fourier power spectrum method. A set of power spectrum features is generated, evaluated, and compared for robustness.

Chapter 6 presents conclusions and highlights the successes and limitations of the analysis. Finally, additional study areas and recommendations are outlined.

CHAPTER II

REVIEW OF RELATED RESEARCH AND LITERATURE

2.1 Introduction

Pattern recognition applications in texture analysis, singular value decomposition, and image processing in general have been intensely studied and modeled over many years. This chapter provides a summary of several significant efforts that have been presented in the literature.

There are many excellent texts on pattern recognition and image processing, such as those authored by Duda and Hart [23], Fukunaga [24], Hall [25], Bow [26], Jain [27], Tou and Gonzalez [28], and Banks [29]. In addition, the *Handbook of Pattern Recognition and Image Processing* [13] and *Digital Image Processing* [12] are helpful in defining terms and explaining the basic concepts and theory of this extensive subject. Although there are numerous approaches to solving pattern recognition problems, this effort is primarily concerned with statistical pattern recognition techniques. The first section discusses some statistical image analysis methods, and the second section highlights singular value decomposition methods in image processing.

2.2 Image Analysis Methods

The statistical approach to image analysis generates parameters that characterize the stochastic properties of the spatial distribution of gray levels in an image. Early work in image texture analysis sought to discover useful features that had some relationship to the fineness and coarseness, contrast, directionality, roughness, and regularity of image tex-

ture. Tamuro, Mori, and Yamawaki [30] discussed the relationship of such descriptive measures to human visual perception. They extracted features from known textured homogeneous images and then classified similar images based on the extracted features. For example, using microscopic imagery, discrimination between eosinophils (leukocyte readily stained by the red dye -- eosin) and large lymphocytes was accomplished using a texture feature for cytoplasm and a shape feature of the cell nucleus. For aerial imagery, discrimination of areas having natural vegetation and trees from areas having man-made objects, buildings, and roads was accomplished using textural features. These statistical textural feature approaches included use of the autocorrelation function, the spectral power density function, edgeness per unit area, spatial gray-tone co-occurrence probabilities, gray-tone run-length distributions, relative extreme spatial distributions, and mathematical morphology.

Bajcsy [31] and Weszka *et al.* [32] suggested the use of statistical features derived from a Fourier power spectrum of the image. Haralick *et al.* [1] defined textural features derived from a gray-level co-occurrence concept that included the spatial interdependence of sets of local features and not necessarily the gray-levels; their features were defined on the entries of another matrix called the generalized co-occurrence matrix. Zucker and Terzopoulos [33] presented a statistical approach for relating the structure in an image pattern to the co-occurrence matrix. An experimental study by Weszka *et al.* [32] revealed the superiority of the spatial gray-level dependence method relative to gray-level run length and power spectrum methods when the orientation of the test image is known.

Spatial frequency characteristics of two-dimensional images can be expressed by the autocorrelation function or by the power spectrum. Both may be calculated digitally and/or implemented in a real-time optical system. Lendaris and Stanley [34] used optical techniques to perform texture analysis on a database of low-altitude aerial photographs. They illuminated small circular sections of the images and used the Fraunhoffer diffraction

pattern to generate features for identifying photographic regions. The major discriminations of concern were man-made roads, intersections of roads, buildings, and orchards. Feature vectors extracted from these diffraction patterns consisted of 40 components. Twenty of the components were mean energy levels in concentric annular rings of the diffraction pattern, and the other 20 components were mean energy levels in 9° wedges of the diffraction pattern. Greater than 90% classification accuracy was reported using this technique.

Gramenopoulos [35] used a digital Fourier transform technique to analyze aerial images. He examined sub-images of *32x32* pixels and determined that for one LANDSAT image, spatial frequencies between 3.5 and 5.9 cycles/km contained most of the information required to discriminate among terrain types. An overall classification accuracy of 87% was achieved using image categories of clouds, water, desert, farms, mountains, urban, river bed, and cloud shadows. Horning and Smith [36] used a similar approach to interpret aerial multispectral scanner imagery.

Bajscy [37] and Bajscy and Lieberman [38] computed the two-dimensional power spectrum of a matrix of square image windows. They expressed the power spectrum in a polar coordinate system of radius versus angle and determined that directional textures tend to have peaks in the power spectrum along a line orthogonal to the direction of the texture and that blob-like textures tend to have peaks in the power spectrum at radii associated with the sizes of the blobs. This work also showed that texture gradients can be measured by determining trends of relative maxima of radii and angles as functions of the position of the image window whose power spectrum is being analyzed. For example, as the power peaks along the radial direction tend to shift toward larger values, the image surface becomes more finely textured.

In general, features based on Fourier power spectra have been shown to perform less well than features based on second-order gray-level co-occurrence statistics [1] or

those based on first-order statistics of spatial gray-level differences [2]. The presence of aperture effects has been hypothesized to account for part of the unfavorable performance by Fourier features compared to space-domain gray-level statistics, although experimental results indicate that this effect, if present, is minimal [39]. However, D'Astous and Jernigan [40] argue that the reason for the performance difference is that earlier studies using the Fourier transform features used summed spectral energies within band- or wedge-shaped regions in the power spectrum and that additional discriminating information could be obtained from the power spectrum for characteristics such as regularity, directionality, linearity, and coarseness. The degree of regularity can be measured by the relative strength of the highest non-dc peak in the power spectrum. Other peak features include the Laplacian at the peak, the number of adjacent neighbors of the peak containing at least 50% of the energy in the peak, the distance of the peak from the origin, and the polar angle of the peak. In the comparative experiment reported by D'Astous and Jernigan, the peak features yielded uniformly greater interclass difference than the co-occurrence features, and the co-occurrence features yielded uniformly greater interclass distances than the summed Fourier energy features.

Transforms other than the Fourier transform can be used for texture analysis. Kirvida [41] compared the Fourier, Hadamard, and Slant transforms for textural features of aerial images. Five classes were studied and a 74% correct classification rate was obtained using only spectral information, which increased to 98.5% when textural information was included in the analysis. No significant difference was reported in classification accuracy as a function of which transform was employed.

Stark and O'Toole [22] described hybrid optical and digital approaches to several statistical pattern recognition problems. They addressed earlier criticism of the Fourier power spectrum technique by significantly improving the resolution of the Fourier transform using high resolution optical techniques. They demonstrated that an optical-digital

computer could be used to recognize black lung disease (also called coal worker's pneumoconiosis) and to solve multiclass problems involving arbitrary texture. They used 16 wedges and 16 concentric rings to generate features of the Fourier power spectrum. The feature vectors were normalized to unity to negate noisy variations due to light fluctuations. They then used classical digital feature selection and classification algorithms, such as the Karhunen-Loeve transform, the Fukunaga-Koontz transform, the Foley-Sammon transform, and the Hotelling trace criterion to classify 64 samples. The Karhunen-Loeve transform and the Fukunaga-Koontz transform methods performed poorly, having correct classification percentages of less than 70%. On the other hand, the results obtained with the Foley-Sammon transform and the Hotelling trace criterion had correct classification rates near 90%. Their second assessment compared their optical digital computer to an all-digital computer approach to classify 50 samples of four different texture patterns. Again, the Foley-Sammon transform and the Hotelling trace criteria were used. Both approaches had high correct classification rates (approximately 100%). They suggested follow-on studies to look at textures that more closely resemble one another.

2.3 Singular Value Methods

The singular value decomposition (SVD) method has been available for over 100 years and was motivated by a need to algebraically decompose a real general matrix into a set of a diagonal and two different orthogonal matrices. The elements of the diagonal matrix were called principal values or singular values (SV). The algebraic method for generating the singular values for medium and large matrices is quite difficult and time consuming by hand. Numerous computer algorithms and techniques have been developed to simplify the computation [42, 43].

Ashjari [11] developed a texture measurement method based on the SVD of a texture sample. In this method an *nxn* texture sample is treated as an *nxn* matrix **X** and the

amplitude-ordered set of singular values s(i) for i = 1, 2, ..., n is computed. If the elements of **X** are spatially unrelated to one another, the singular values tend to be uniformly distributed in amplitude. However, if the elements of **X** are highly structured, the singular value distribution tends to be skewed such that the amplitudes of the first few singular values are much larger than any of the others.

To demonstrate his method for natural texture, Ashjari assessed four of Brodatz's [44] natural textures (raffia, grass, sand, and wool). Pratt and Faugeras [12, 45] conducted and compiled several studies of texture feature extraction techniques using the same Brodatz natural textures and error evaluation (Bhattacharyya distance). These studies included autocorrelation, decorrelation, histogram, dependency matrix, Law's microstructures, and SVD methods. The SVD method displayed some of the best results.

In addition to texture feature extraction, other investigators have used SVD for image processing and pattern recognition. Li Shu-Qiu *et al.* [17, 18] used SVD to extract a faint electrocardiogram signal of a fetus in a strong noise background. Hong [18] used the geometric attributes of an image derived from SV features to classify faces. Cheng [46] used SVD for identifying different aircraft for automatic target recognition.

Possible optical implementations of SVD methods are discussed by Kumar [47] and Karim *et al.* [4].

CHAPTER III

BACKGROUND CONCEPTS

3.1 Introduction

This chapter provides a background description of the general pattern recognition process, describes feature extraction concepts, and develops the theory used to classify natural background scenes. The methodology of pattern recognition and the topics of feature extraction, image analysis, and texture are presented first and some common terms and methods used throughout the study are defined. The mathematical description, motivation, and unique attributes of SVD in signal processing are discussed. Finally, descriptions of the Bayes classifier and Bhattacharyya distance methods used to evaluate the SV features are presented.

3.2 Pattern Recognition Process

The pattern recognition process [25,48] consist of three basic steps: segmentation, feature extraction, and classification (Figure 3.1). A sensor is assumed to have sampled the environment and generated an image consisting of many patterns and shapes representing the scene of interest.

The first step, segmentation, consists of dividing or separating the image into regions of similar attributes. The most basic attribute for segmentation is image amplitude (such as gray scale or color). Image edges and texture are also useful attributes for segmentation. Generally, no contextual information is used in segmentation. The segmentor does not attempt to recognize the individual segments or their relationship to one another.

This study does not use automatic methods to segment the image; segmentation is performed manually.

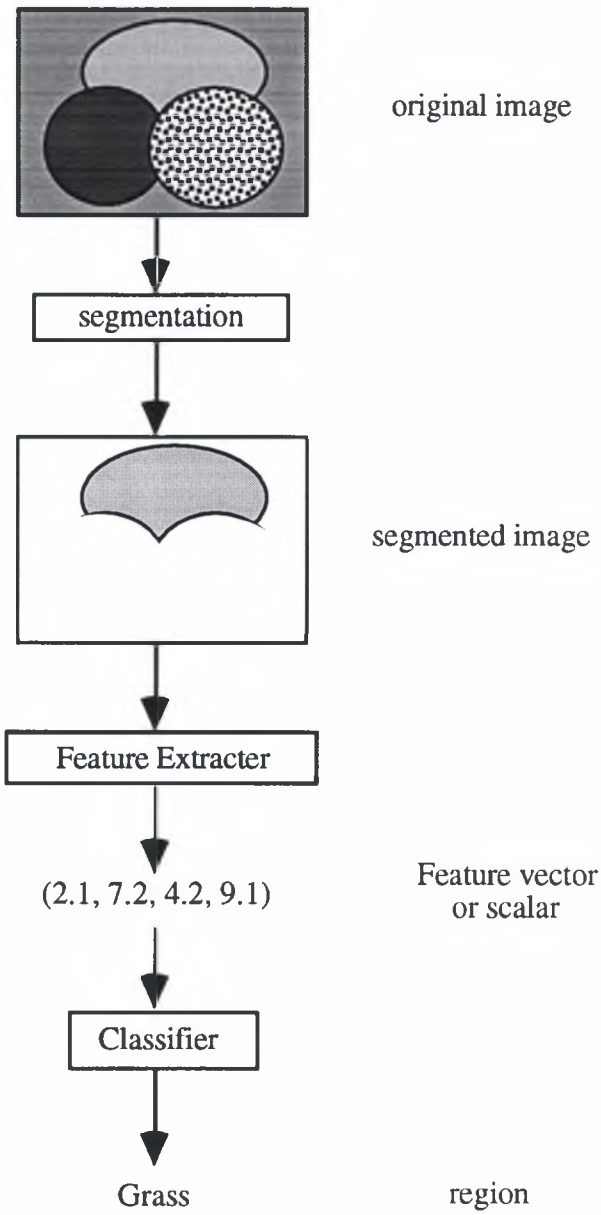

Figure 3.1. Pattern recognition process.

The next step, feature extraction, is the focus of this study. The following section defines some of the common terms and provides background for the methods used. Feature extraction in pattern recognition is primarily concerned with developing mathematical tools (digital image processing), optical techniques (optical computing), or a hybrid of each for reducing the dimensionality of a image or pattern. Pattern descriptors are often called features and represent the lower-dimensional properties of a pattern class or region. There are a number of reasons for reducing an image to a few fundamental numbers. One of the primary reasons is to minimize the hardware and software requirements when working with large image patterns or many classes of patterns. Reduced dimensionality also often helps improve classifier performance. Sometimes performance increases when additional features are added, but often at some point the size of the feature set reaches a point of diminishing return. Finally, the motivation for feature extraction (and the area this effort addresses) is that it provides a means for qualitatively discriminating between patterns of different classes. If the class overlap is too high, then perhaps a different technique or new source of information is needed to accurately discriminate between classes.

There are generally two quantitative approaches for evaluating image features: prototype performance and figure of merit. In the prototype performance approach, prototype image regions (independently categorized by some classification procedure) are compared using image features. The classification error is then measured for each feature set, and the best set of features is the one with the least classification error. The figure of merit approach measures the functional distance between sets of image features. A large distance implies a low classification error. Ashjari and Pratt [11,12] used the Bhattacharyya distance (described later in this chapter) to measure texture classification accuracy. This study also uses the Bhattacharyya distance figure of merit approach.

The final step, classification, involves assigning a label to the features in question. This portion of the process is not addressed in this study. Numerous classifier design techniques are discussed by Duda [23] and Fukunaga [24].

3.3 Common Terms and Definitions

Image Analysis

Image analysis is the extraction of measurements, data, or information from an image by automatic or semiautomatic methods and is also called image data extraction, scene analysis, image description, automatic photo interpretation, image understanding, and a variety of other names. The ultimate product of image analysis is usually numerical output rather than a picture. Image analysis also differs from classical pattern recognition in that analysis systems, by definition, are not limited to the classification of scene regions into a fixed number of categories, but rather are designed to describe complex scenes whose variety may be large and ill-defined in terms of *a priori* expectation.

Image Feature Extraction

An image feature is a distinguishing basic characteristic or attribute of an image. Some features are natural in the sense that are defined by the visual appearance of an image, while other so-called artificial features result from specific manipulations of an image. Natural features include the luminance of a region of pixels and gray scale textural regions. Image amplitude histograms and spatial frequency spectra are examples of artificial features. Image features are important in the isolation of regions of common property within an image (image segmentation) and the subsequent identification or labeling of such regions (classification).

Texture

Many portions of images of natural scenes are devoid of sharp edges over large areas. In these areas the scene can often be characterized as exhibiting a consistent structure analogous to the texture of cloth. Image texture measurements can be used to segment an image and classify its segments. Several authors have attempted qualitatively to define texture. The elements and rules of spacing or arrangement may be arbitrarily manipulated, provided a characteristic repetitiveness remains. Hawkins [49] describes texture as follows: "The notion of texture appears to depend upon three ingredients: (1) some local order is repeated over a region which is large in comparison to the order's size, (2) the order consists in the non-random arrangement of elementary parts, and (3) the parts are roughly uniform entities having approximately the same dimensions everywhere within the textured region." Although such descriptions of texture seem perceptually reasonable, they do not immediately lead to simple quantitative textural measures in the sense that the description of an edge discontinuity leads to a quantitative description of an edge in terms of its location, slope angle, and height. Because texture is a spatial property, measurements should be restricted to regions of relative uniformity. Hence it is necessary to establish the boundary of a uniform textural region by image segmentation before attempting texture measurement.

3.4 Singular Value Feature Extraction

In this section the motivation and basic theory behind a SVD feature extraction method are discussed. The SVD is particularly well suited to natural texture feature extraction because of its relative insensitivity to noise and other slight variations in texture fields.

3.4.1 Singular Value Decomposition

The SVD has been applied to signal processing problems since the late 1970's. However, the concept was developed over 100 years ago.

Vaccaro [50] explains the motivation for SVD in signal processing. Many of the problems of signal and image processing begin with a known well-defined signal, but its matrix representation is rank deficient. In other words, the signal corresponds to a proper subspace (e.g. row-space) of Euclidean space. When a noisy signal is collected, the matrix representation is no longer rank deficient. However, the noisy signal may be close to a rank deficient matrix in the sense that a small perturbation of the elements of the noisy signal would produce a rank-deficient matrix. The SVD gives the optimal rank-deficient approximation to a given full-rank matrix. Once a rank-deficient matrix is extracted from the noisy signal, the signal corresponding to the row-space of the rank-deficient matrix can be calculated. Thus the SVD can be thought of as a filter which produces a signal estimate from noisy data. This non-linear filtering process has been shown to produce better results at lower signal-to-noise ratios than classical linear techniques [50]. Another reason for the use of SVD is that it produces orthonormal bases for the row and column spaces of the matrix. As a consequence, the geometric structure of any operations involving these subspaces is revealed by the SVD.

The mathematical definition of the SVD [42] is

$$\mathbf{F} = \mathbf{U}\,\mathbf{S}\,\mathbf{V}^{T}$$

where the matrix **F** is decomposed into three matrices **U**, **S**, and **V**. Matrices **U** and **V** are unitary and **S** is a real diagonal matrix. In image processing **F** is a real *kxn* matrix, **U** is a real *kxk* orthonormal matrix, **S** is a real *kxk* diagonal matrix with non-negative elements, and **V** is a real *nxk* orthonormal matrix. The number of non-zero singular values is equal to the rank of the matrix. The higher the correlation in a texture image the lower its rank, thus resulting in a steeper singular value curve. One of the most valuable aspects of the SVD is that it enables one to deal with situations where near matrix rank deficiency pre-

vails. Rounding errors and fuzzy data make rank determination a nontrivial exercise with traditional linear algebra methods. Numerical rank deficiency is nicely characterized with SVD because the singular values indicate how near a given matrix is to a matrix of lower rank.

The singular values of a matrix can be considered as features of the matrix elements and their inter-relationships. If the image matrix is a set of random (uncorrelated) numbers, then the singular values tend to be equal. However, if the image matrix is highly structured (high correlation between elements), as in a checkerboard pattern, then usually the first and second singular values dominate and the other singular values approach zero. The singular values of image samples of natural texture generally lie between these two extremes and hence provide the motivation for this study. Figure 3.2 shows *32x32*-pixel examples of several cases. The structured scene is simply a binary checkerboard pattern. The random scene was generated from random pixel values from a normal distribution. The tree and grass scenes are examples of the natural background texture samples that are evaluated in this study. Figure 3.3 compares the distribution of singular values for the images in Figure 3.2. The structured scene has a sharp peak due to the first two (dominant) singular values, but the rest of the singular values are approximately zero. The random scene has the most even distribution of singular values (although descending gradually) of the example cases. The grass and tree scenes each have uniquely structured curves. Similar samples of grass or trees would resemble their respective classes.

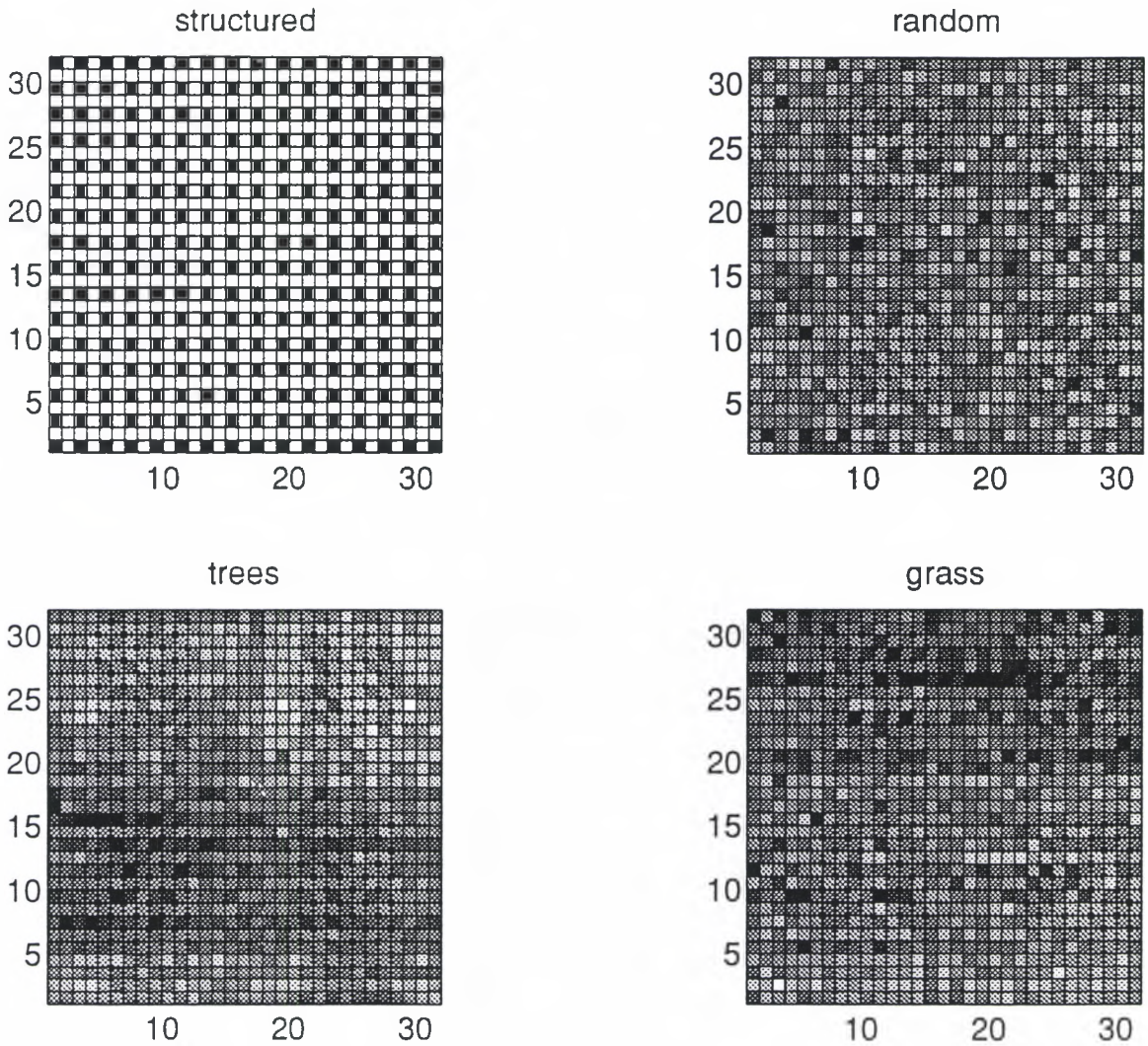

Figure 3.2. Examples of texture regions (structured, random, trees, and grass).

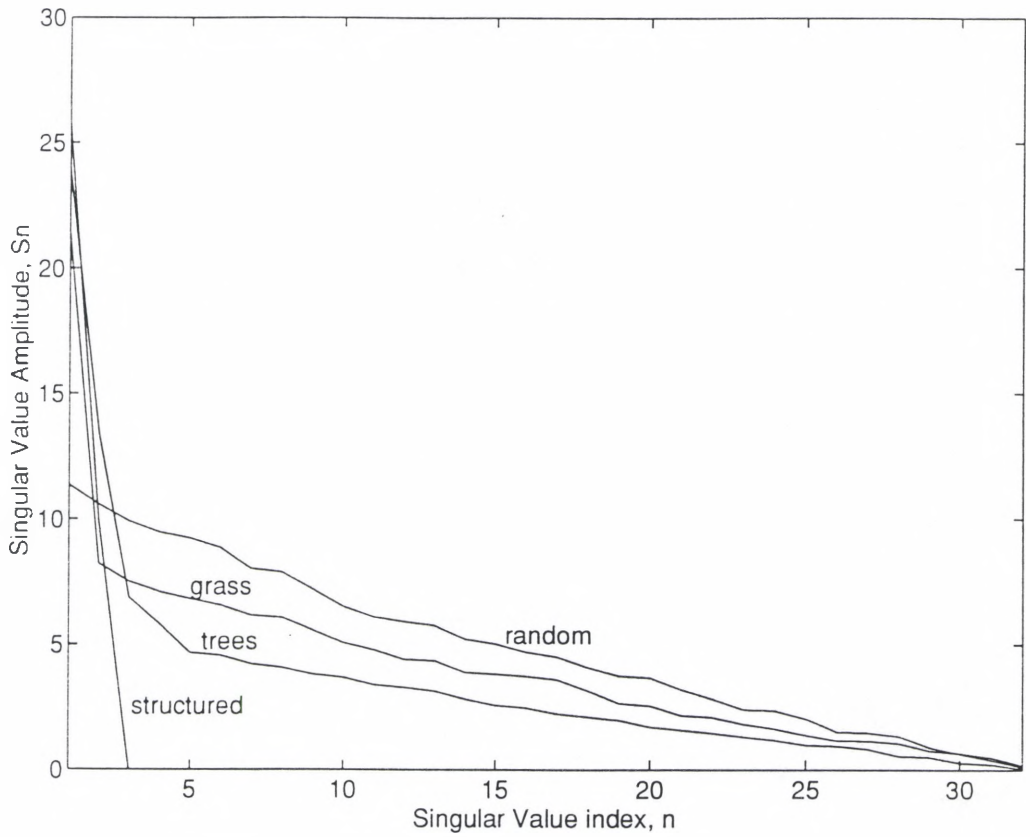

Figure 3.3. Singular value distributions for example texture regions.

Not every kind of matrix is suitable for SVD analysis. Ashjari [11] showed that highly structured image matrices from man-made objects cannot effectively be analyzed using SVD.

The next section examines several unique ways of generating features from the SVD process.

3.4.2 Singular Values Features

The SVD of the image matrices in Figure 3.2 provided essentially a vector feature containing 32 singular values ($s_1, s_2, s_3, ..., s_{32}$). These features can be analyzed in several ways to generate a reduced set of singular value texture features for the classification and evaluation of different classes of scenes. Ashjari [11] generated the following vector and scalar features to assess singular value texture features:

I. General singular value vector

$$\mathbf{z}_1 = (s_1, s_2, s_3, ..., s_k)$$

II. Normalized singular value histogram vector

$$\mathbf{z}_2 = \left(\frac{s_1}{\sum_i^k s_i}, \frac{s_2}{\sum_i^k s_i}, \cdots, \frac{s_k}{\sum_i^k s_i} \right)$$

III. Normalized singular value length vector

$$\mathbf{z}_3 = \left(\frac{s_1}{\left(\sum_i^k s_i^2\right)^{\frac{1}{2}}}, \frac{s_2}{\left(\sum_i^k s_i^2\right)^{\frac{1}{2}}}, \cdots, \frac{s_k}{\left(\sum_i^k s_i^2\right)^{\frac{1}{2}}} \right)$$

IV. Energy of singular value vector

$$\mathbf{z}_4 = \left(s_1^2,\ s_2^2,\ \cdots,\ s_k^2\right)$$

V. Normalized energy singular value vector

$$\mathbf{z}_5 = \left(\frac{s_1^2}{\sum_i^k s_i^2},\ \frac{s_2^2}{\sum_i^k s_i^2},\ \cdots,\ \frac{s_k^2}{\sum_i^k s_i^2}\right)$$

Each of these vector features ($\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3, \mathbf{z}_4, \mathbf{z}_5$) can be further reduced by generating a first-order shape histogram using the first four moments (mean, standard deviation, skewness, and kurtosis) of each. Experiments have shown that the first four moments are very accurate in representing extracted features [51]. For each vector above let h(i) = $\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3, \mathbf{z}_4$, or $\mathbf{z}_5$. The first four moments then can be derived as follows:

a. Mean

$$M_1 = \sum_{i=1}^{k} \frac{i\, h(i)}{M_0}$$

b. Standard deviation

$$M_2 = \left[\sum_{i=1}^{k} (i - M_1)^2\, h(i)\right]^{\frac{1}{2}}$$

c. Skewness

$$M_3 = \frac{1}{M_2^3} \sum_{i=1}^{k} (i - M_1)^3 \, h(i)$$

d. Kurtosis

$$M_4 = \frac{1}{M_2^4} \sum_{i=1}^{k} (i - M_1)^4 \, h(i) - \frac{3}{M_0}$$

In each case $M_0 = \sum_{i=1}^{k} h(i)$. The term $3/M_0$ makes the Gaussian histogram limit zero. For the normalized vector feature $\mathbf{z_2}$ and $\mathbf{z_5}$, $M_0 = \sum_{i=1}^{k} h(i) = 1$ and h(i) is analogous to a discrete probability density function. Thus a non-linear transformation vector can be created for each of the singular value feature vectors $\mathbf{z_1}$, $\mathbf{z_2}$, $\mathbf{z_3}$, $\mathbf{z_4}$, $\mathbf{z_5}$, i.e. $\mathbf{z_6} = (M_1(\mathbf{z_n}), M_2(\mathbf{z_n}), M_3(\mathbf{z_n}), M_4(\mathbf{z_n}))$ where n = 1, 2, 3, 4, or 5.

VI. Moment vector

$$\mathbf{z}_6 = (M_1, M_2, M_3, M_4)$$

In the evaluation and classification process, instead of using each singular value feature vector $\mathbf{z_1}$, $\mathbf{z_2}$, $\mathbf{z_3}$, $\mathbf{z_4}$, $\mathbf{z_5}$ of k numbers, the 4-dimensional vector $\mathbf{z_6}$ will be used. Again, the accuracy of the moment vector is high and its use reduces computational requirements.

In addition to the vector features above, certain scalar characteristics can be derived that are useful in further reducing the computational load, particularly when some *a priori* information is available about the image or regions of interest.

VII. Entropy scalar of $\mathbf{z_2}$ and $\mathbf{z_5}$

$$z_7 = -\sum_{i=1}^{k} h(i) \log_2 [h(i)]$$

VIII. Energy scalar of $\mathbf{z_1}$ and $\mathbf{z_2}$

$$z_8 = \sum_{i=1}^{k} [h(i)]^2$$

IX. Product of singular values

$$z_9 = \prod_{i=1}^{k} s_i$$

X. Largest singular value scalar

$$z_{10} = s_1$$

XI. Largest component of normalized histogram scalar

$$z_{11} = \frac{s_1}{\sum_{i=1}^{k} s_i}$$

XII. Largest component of normalized length scalar

$$z_{12} = \frac{s_1}{\left(\sum_{i=1}^{k} s_i^2 \right)^{\frac{1}{2}}}$$

XIII. Largest energy ratio scalar

$$z_{13} = \frac{s_1^2}{\left(\sum_{i=1}^{k} s_i^2 \right)}$$

XIV. Entropy of the largest SV for z_{11} and z_{13} scalar (where $b = z_{11}$ or z_{13})

$$z_{14} = -b \log_2 b$$

XV. Conditioning scalar

$$z_{15} = \frac{s_1}{s_k}$$

Scalar Feature Observations

Many of the scalar features above are useful only for certain kinds of texture image analysis. It is important to understand their limitations and unique applications.

Entropy scalar (z_7) - This feature is useful in determining the correlation in the texture from the normalized vectors $\mathbf{z_2}$ and $\mathbf{z_5}$. The lower the correlation of the texture of the scene, the higher the entropy.

Energy scalar (z_8) - This feature is effective only for vectors $\mathbf{z_1}$ and $\mathbf{z_2}$.

Product of SV scalar (z_9) - For highly structured (correlated) images the higher indexed singular values quickly approach zero, so multiplication produces a very small scalar value.

Largest SV scalar (z_{10}, z_{11}, z_{12}, z_{13}) - All of these scalar features provide a single scalar characteristic of the most important singular value (s_1). These features provide reasonable measures of the attribute of the image if extreme reduction (data compression) is required. They generally work well with highly structured image matrices.

Entropy of the largest SV scalar (z_{14}) - This feature only applies to z_{11} and z_{13}.

Conditioning scalar (z_{15}) - This feature is the ratio of the largest singular value to the smallest. Consequently, this feature (like others that compare the largest singular value to other singular values) has problems for a structured image where the smallest singular values are essentially zero, and it is generally used to represent more random texture field.

For natural texture patterns, Ashjari found that the scalar features generally did not perform as well as the vector features. However, for artificial texture patterns the scalar features performed better than the vector features.

3.4.3 Singular Values Features of Natural Texture

Ashjari [11] performed experiments with singular value features on several statistically significant artificial and four natural texture backgrounds (grass, raffia, sand, and wool). Sixty-four *32x32*-pixel samples were analyzed for each background (in addition, experiments were performed with 32, 128, and 196 *32x32*-pixel samples of each background; the 64 *32x32*-pixel sample size was fairly stable and statistically representative).

The natural background images were taken from Brodatz's album of natural textures. All of the natural images were close-up texture (Figure 3.4) with a high degree of homogeneity in each texture pattern.

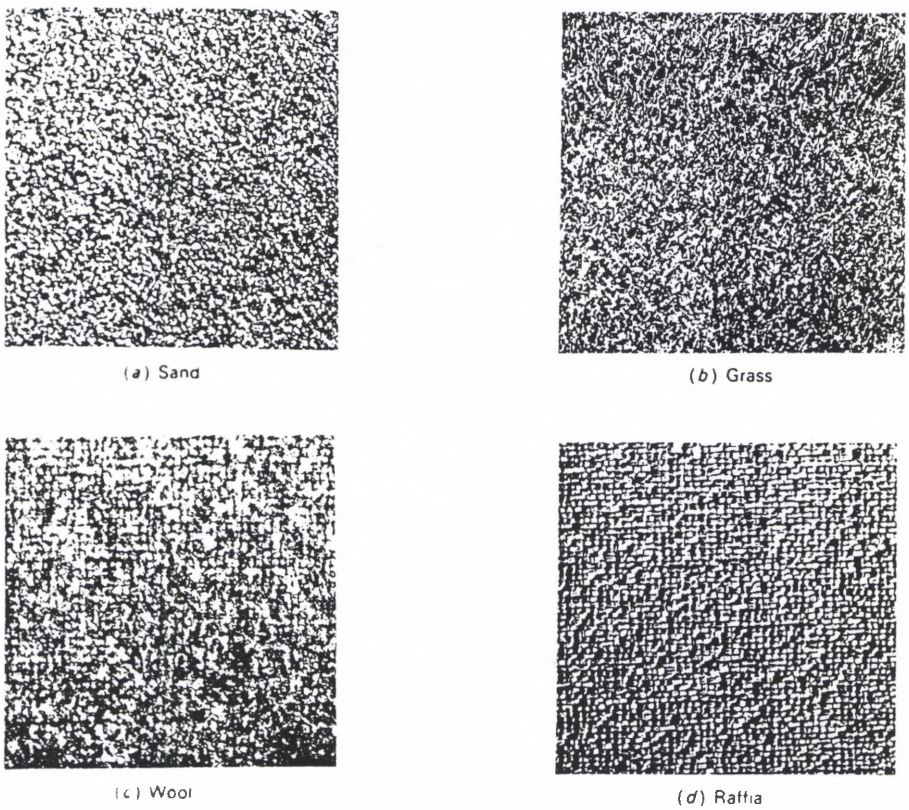

Figure 3.4. Brodatz's natural texture background used for SV analysis [11].

Figure 3.5 shows the average distribution of singular values for the four Brodatz textures calculated by Ashjari. The shape of the SV distribution was quantified by the one-dimensional moments (mean, standard deviation, skewness, kurtosis) described in the previous section. Experiment determined that the best features for natural texture backgrounds were the moment vector (M_1, M_2, M_3, M_4). The scalar features did not perform as well.

For this study, the moment vector features are used. Ashjari used Bhattacharyya distance measurements to determine the classification accuracy (described in the next section). For the four textures in Figure 3.4, the Bhattacharyya distances were relatively high and thus had high classification accuracy (Table 3.1). Note that the Bhattacharyya distances correspond well to the separability of the curves in Figure 3.5. For example, the wool and raffia texture patterns have the largest Bhattacharyya distance (Table 3.1) and have the greatest variation in their distribution curves (Figure 3.5). Conversely, the grass and sand distribution curves are the most similar and have the smallest Bhattacharyya distance.

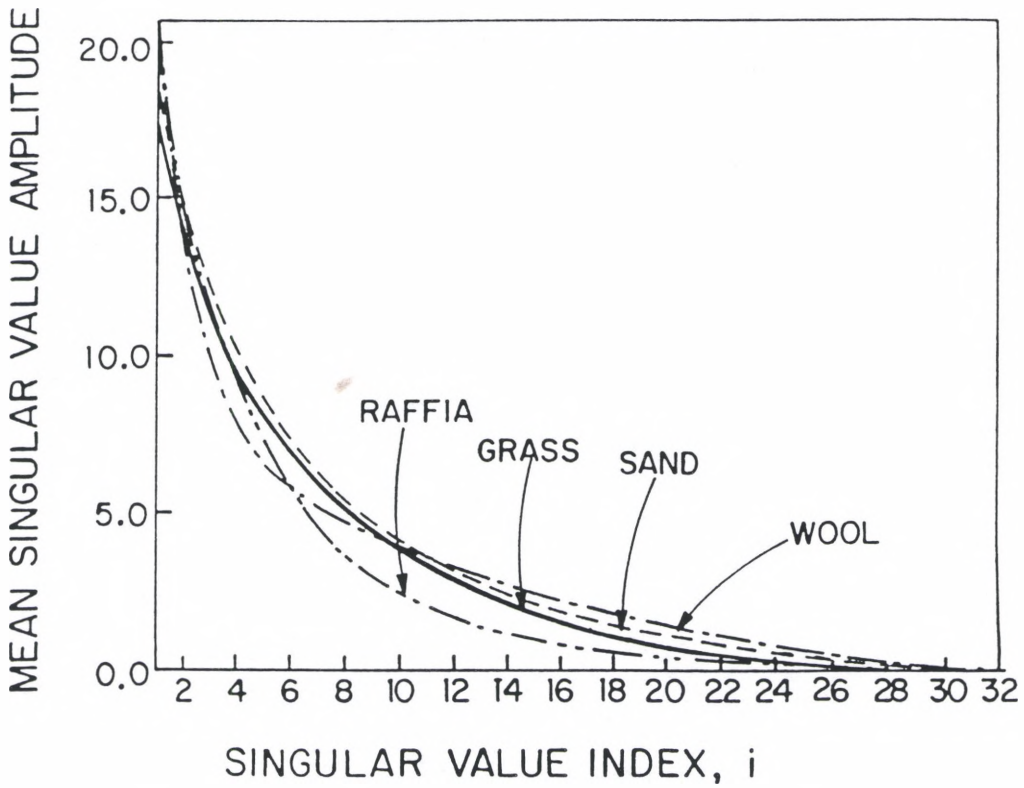

Figure 3.5. Average singular value distributions for Brodatz's natural texture background fields [11].

Table 3.1

Bhattacharyya distances and classification accuracies for Brodatz's natural texture background fields [11].

Texture Pairs		Bhattacharyya-distance	Classification accuracy %	
			Lower Bound	Upper Bound
Grass	Raffia	2.47	95.8	99.9
Grass	Sand	1.42	88.0	98.5
Grass	Wool	3.71	98.8	100.0
Raffia	Sand	7.20	100.0	100.0
Raffia	Wool	11.20	100.0	100.0
Sand	Wool	3.91	99.0	100.0

3.5 Classification of Features

This section provides background for classifying and evaluating the accuracy of the features represented by each region segmented from a scene. The Bayesian classifier is theoretically the best classifier, because it minimizes the probability of classification error. Unfortunately, its implementation is often difficult. However, if two regions are equally likely and their distribution is normal, then a simpler measure, the Bhattacharyya distance error bound, can be used to measure classification error.

3.5.1 Bayesian Classifier

The probability of error is the key parameter in pattern recognition. The error due to the Bayes classifier (the Bayes error) gives the smallest error that can be achieved from any given distribution [24].

If **X** is an observation vector and the goal is to determine whether **X** belongs to class w_1 or w_2, then a decision rule can be developed based upon probabilities. Let

$$\begin{array}{ll} q_1(\mathbf{X}) < q_2(\mathbf{X}) & \text{for } w_1 \\ q_1(\mathbf{X}) > q_2(\mathbf{X}) & \text{for } w_2 \end{array}$$

where $q_i(\mathbf{X})$ is *a posteriori* probability of w_i given **X**. For the above expression, if the probability of w_1 given **X** is larger than the probability of w_2, then **X** is classified w_1, and vice versa. In general, the decision rule does not lead to perfect classification. To evaluate the performance of a decision rule, a probability of error is determined (i.e., the probability that the sample will be assigned to the wrong class). The probability of error is the most effective measure of decision rule usefulness. It is generally quite difficult to calculate the error probability, because it requires n-dimensional integration in a complicated region. The upper bound on the Bayes error is often used as an approximate expression for the error probability. The Bhattacharyya distance error bound [52] is an upper bound of the probability of error that can be derived for a closed form expression which can be simply calculated.

3.5.2 Bhattacharyya Distance Error Bound

Minimization of the error probability to determine optimum signals or images is often difficult to carry out. Consequently, several sub-optimum performance measures that are easier than the error probability to evaluate and manipulate have been developed. In the search for a suitable criterion, distance measures between two probability distributions were introduced, such as the D^2 Mahalanobis distance [53], the linear discriminant function [54], and the Bhattacharyya distance [55]. The further apart the distributions the smaller the probability of mistaking one for the other. Kailath [55] has shown that the

Bhattacharyya distance is often easier to work with and provides more accurate results that the other methods.

To determine an expression for the Bhattacharyya distance, let

$$\mu = \text{Bhattacharyya coefficient} = \int \sqrt{p_1(X)\, p_2(X)}\, dx$$

where **X** denotes a vector containing individual image feature measurements and the conditional densities are $p_1(X)$ and $p_2(X)$. The Bhattacharyya coefficient (μ) lies between zero and unity. An exponential function is commonly used to model the distance measure [55]. The Bhattacharyya distance can be expressed as

$$B = \text{Bhattacharyya distance} = -\ln \mu$$

The Bhattacharyya distance is a measure of the separability between the feature-space conditional densities of two classes. Kailath [55] developed explicit expressions of the Bhattacharyya distance for several important distributions; multinomial, Poisson, and univariate and multivariate Gaussian distributions. The Gaussian form is the simplest of the expressions and will be used for this study. It is expressed as

$$B = \frac{1}{8}(M_2 - M_1)^T \left(\frac{\Sigma_1 + \Sigma_2}{2}\right)^{-1} (M_2 - M_1) + \frac{1}{2} \ln \left[\frac{\left|\frac{\Sigma_1 + \Sigma_2}{2}\right|}{\sqrt{|\Sigma_1||\Sigma_2|}}\right]$$

where M_1, M_2 are the mean vectors and Σ_1, Σ_2 are the covariance matrices of class 1 and class 2. This expression is the Bhattacharyya distance for Gaussian distributed feature data and is used to measure the separability of feature vectors between any two classes or

regions. In practice, features would be obtained from regions whose class has been independently determined. Sufficient feature measurements need to be taken so that the feature mean vector and covariance matrix can be accurately estimated (central limit theorem [56]).

The Bhattacharyya distance can be related to the Bayes upper error bound [24, 55] as

$$\varepsilon_u = \sqrt{P_1 P_2} \int \sqrt{p_1(X)\, p_2(X)}\, dx = \sqrt{P_1 P_2}\, e^{-B}$$

where ε is the Bayes probability of error, and P_1 and P_2 are the probability distributions of each class.

For equally likely *a priori* probability distributions ($P_1 = P_2 = 1/2$), the probability of error (Bayes upper bound) is related to the Bhattacharyya distance by

$$\varepsilon_u \leq \frac{1}{2} \exp(-B)$$

The lower bound to the probability of error is also related to the Bhattacharyya distance and is expressed as

$$\varepsilon_L \geq \frac{1}{2} - \frac{1}{2}\left[1 - \exp(-2B)\right]^{\frac{1}{2}}$$

For a Bhattacharyya distance $B = 1$ the classification error for the upper bound is 18.4% or a classification accuracy of 81.6%. High classification accuracy on the order of 99.9% have Bhattacharyya distances on the order of $B = 6$.

CHAPTER IV

APPLICATION OF THE SVD METHOD TO FLIR IMAGES

4.1 Introduction

This chapter contains results obtained by applying the SVD methods described in Chapter III to natural background scenes. The process used is illustrated in Figure 4.1. The FLIR images are segmented manually into three regions for evaluation: grass, trees, and water. A number of samples are extracted from a digital library of similar scenes. The feature selection process standardizes the images and creates SV features. Finally, the classification accuracy of the features is evaluated using the Bhattacharyya distance measure.

The FLIR images were obtained from the Air Force Wright Laboratory Automatic Target Recognition Branch (WL/AARA). Representative samplings of grass, trees, and water regions were selected for evaluation. Figure 4.2 represents a typical scene with *32x32* sample regions for grass, trees, and water in boxes.

4.2 FLIR Image Description and Selection

The images used to represent potential target background scenes were provided by WL/AARA. These scenes consisted of FLIR images taken in 1982 by Texas Instruments at the U.S. Army's Aberdeen Proving Ground in Maryland. A helicopter fire control FLIR was used to generated the images. There were approximately 130 digitized runs (available in the WL/AARA model-based vision lab) conducted from October to December; weather and daytime conditions varied from run to run.

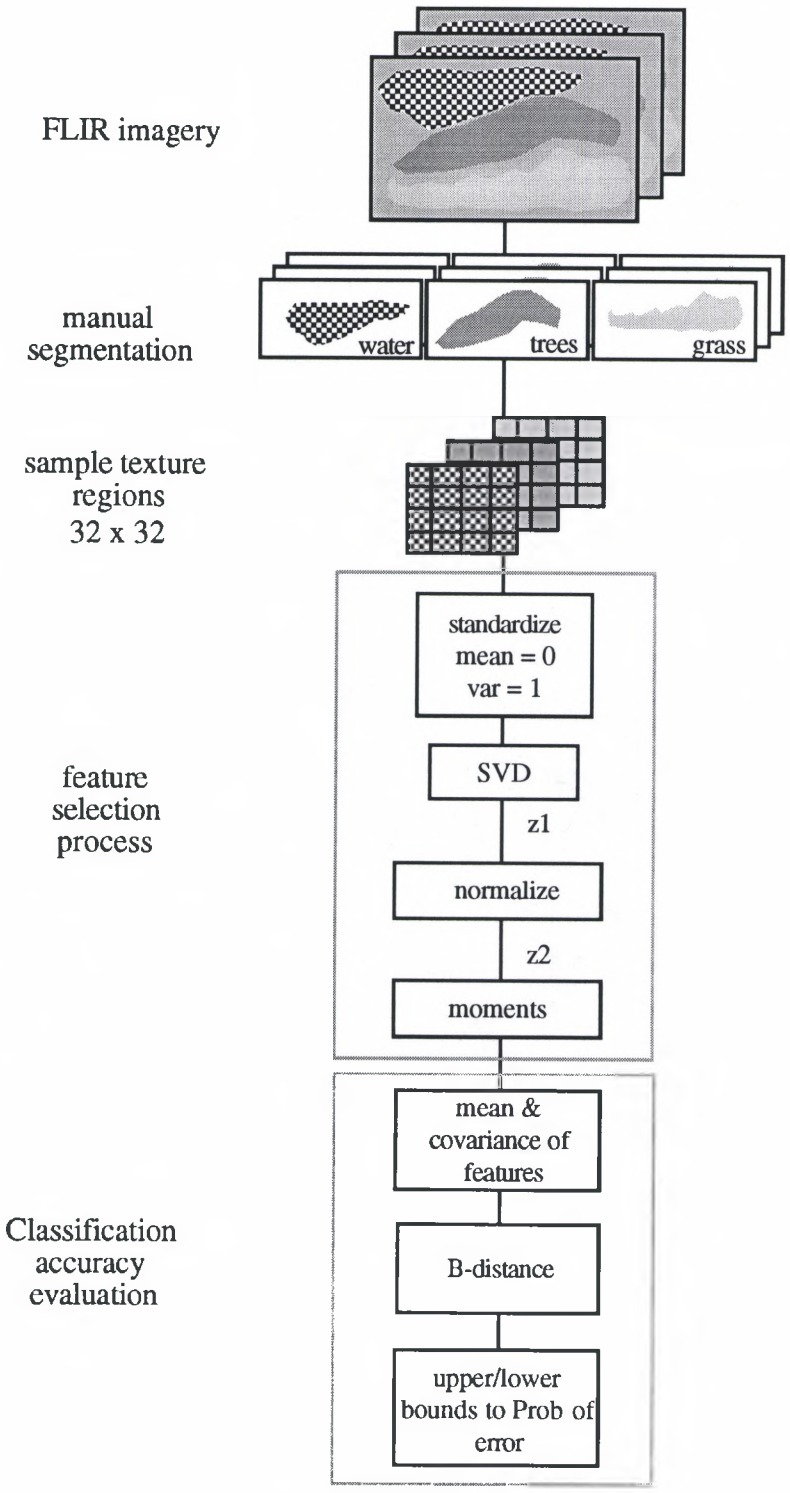

Figure 4.1. Image feature extraction and evaluation process.

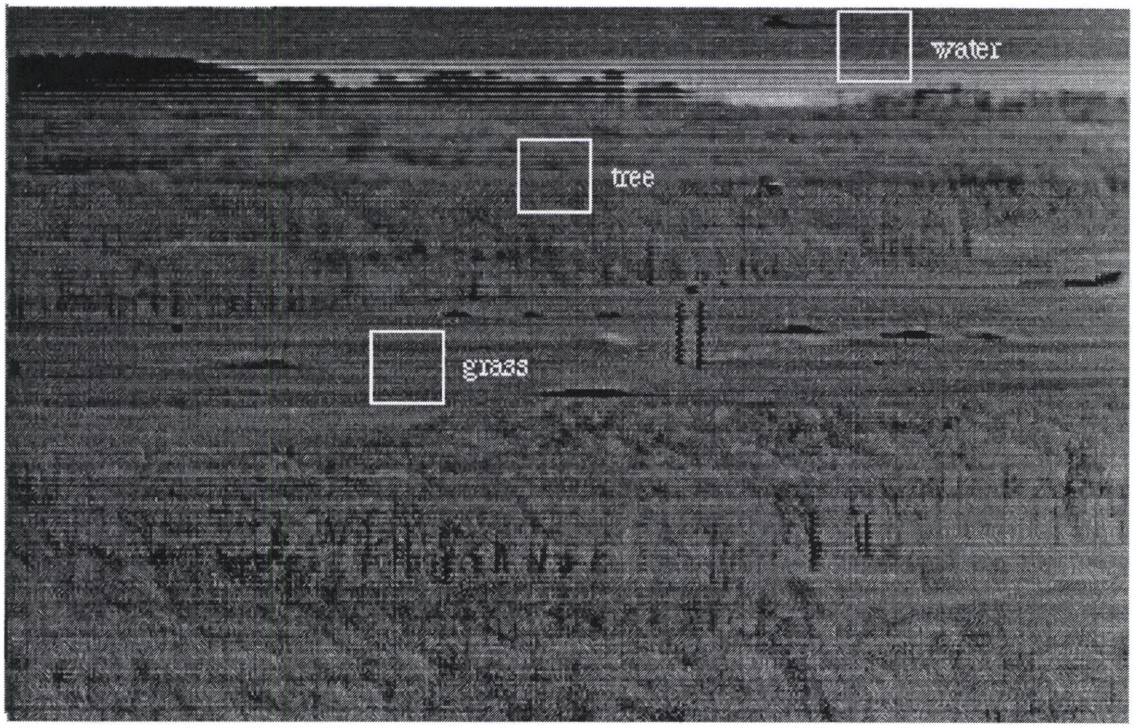

Figure 4.2. Examples of grass, tree, and water sample regions from FLIR scenes.

Figure 4.3 is the first frame of one of the runs (lant_66). Note the broad area scene consisting of a land strip surrounded by water with deciduous tree regions separated by an open grass field. The average run was approximately three minutes long. The image file for each run contained a sequence of approximately 4,000 image frames. Most of the runs began recording at ranges of approximately 6-7,000 meters from a transponder located on or next to a group of three targets situated along a treeline. The runs ended at approximately 100 meters from the targets. The recording rate was 30 frames per second. Most of the images were low contrast. It was very difficult to observe the texture patterns of trees and grass at a 7,000 meter range from the treeline. Close-up ranges also were also problematic. At ranges less than 1,000 meters from the treeline, the texture pattern of the trees and grass began to change rapidly. Figure 4.4 illustrates a typical sequence starting at frame ten (lant_66_10) and ending at approximately 1,000 meters from the targets with frame 3,300 (lant_66_3300). For this study, images were selected from frames 1,000 to 2,000 from two of the best weather condition runs (lant_66 and lant_68). Both runs began recording at approximately the same range but had different aircraft approach paths. Figure 4.5 shows the four frames from which the majority of the regional samples were extracted for analysis. Unfortunately, the transponder was inoperative during these runs, so only approximate or qualitative range values for each image are available. An Air Force analyst (WL/AARA) believes the range for frame lant_66_1000 (frame 1,000) is approximately 4,000 to 5,000 meters from the aircraft to the treeline. At a rate of 30 frames per second and given that both runs were approximately 3 minutes long, the sample images cover a range of approximately 1,000 meters (starting at approximately 5,000 meters from the aircraft to the targets).

Figure 4.3. General broad-area FLIR scene.

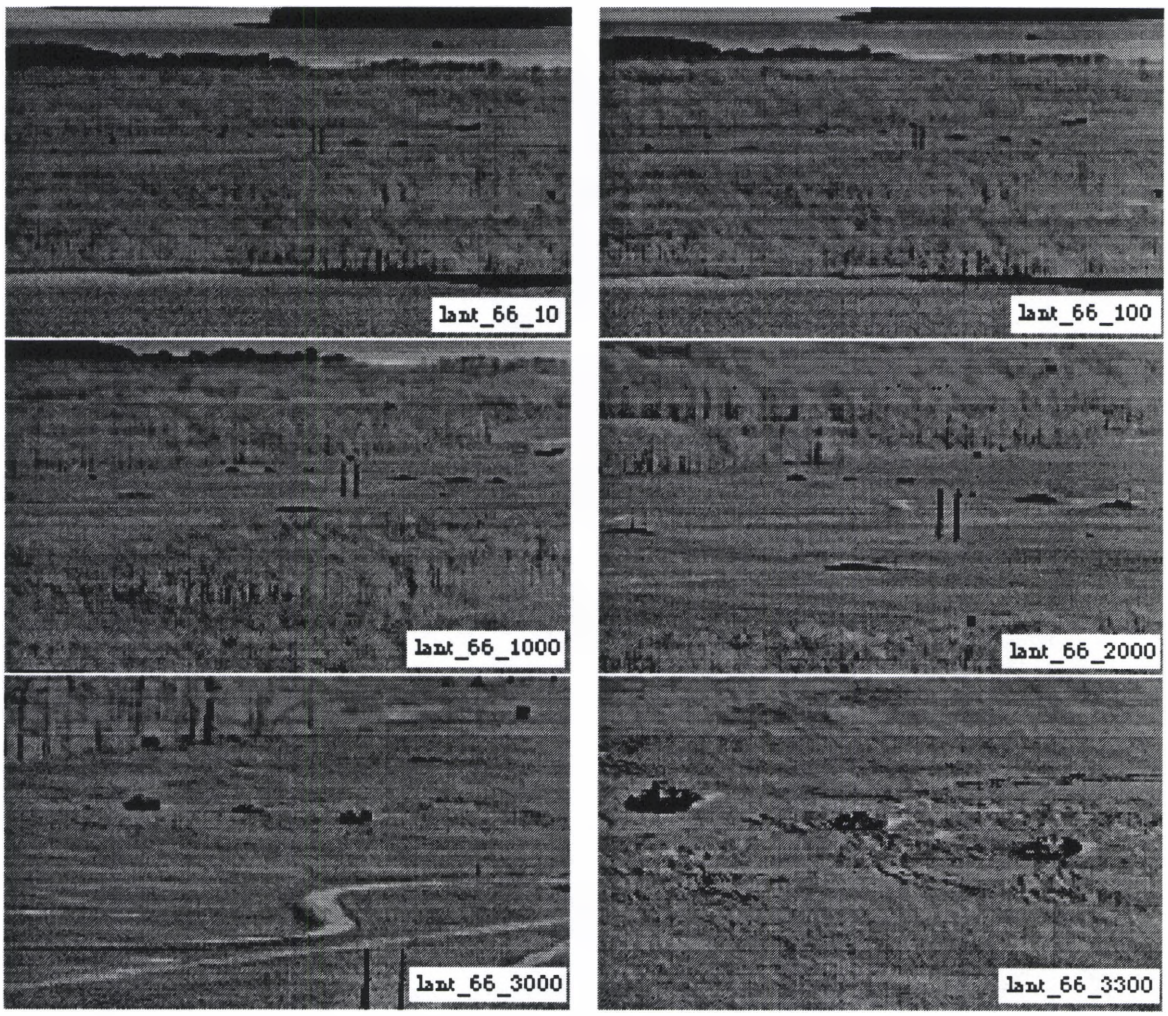

Figure 4.4. Typical image sequence of FLIR runs (frame 10, 100, 1000, 2000, 3000, 3300).

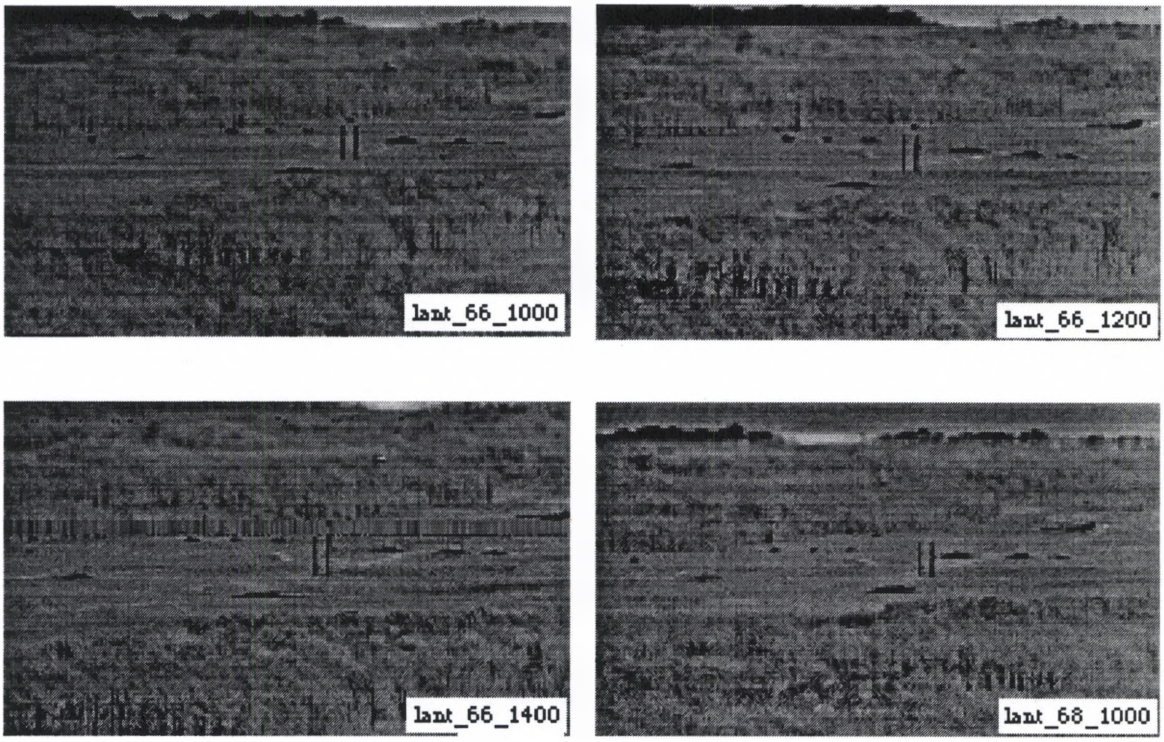

Figure 4.5. Scenes used for texture regional analysis.

It was visually determined that sample sizes of *32x32* pixels would be adequate to represent the texture of the grass, tree, and water regions. The primary concern was selecting a representative sampling of the texture within each region. Sample sizes larger than *32x32* did not typically provide enough square or rectangular samples (required for SVD matrix calculations) of each texture region for evaluation. Sample sizes smaller than *32x32* did not provide representative texture patterns for each region (particularly the tree region). Sixty-four *32x32* samples were taken from lant_66 and lant_68 (between frames 1,000 and 2,000) for regions of grass and trees (Figures 4.6 and 4.7). Few water samples were available for the same frames. Only 20 *32x32* samples of the water region were used in the analysis (Figure 4.8).

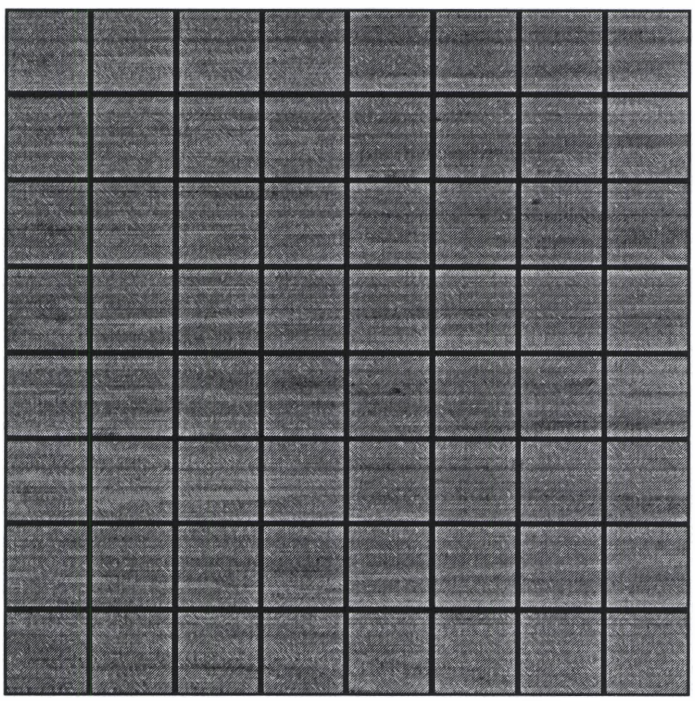

Figure 4.6. Grass region (64 *32x32* samples).

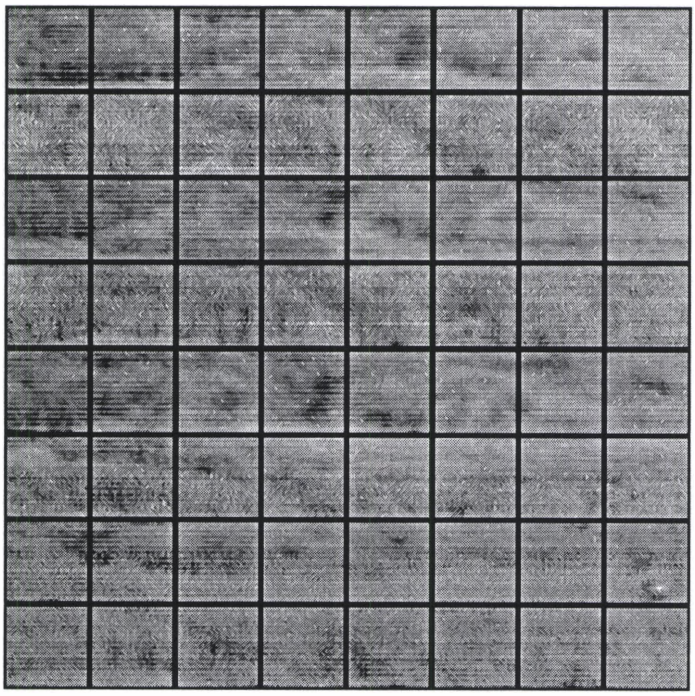

Figure 4.7. Tree region (64 *32x32* samples).

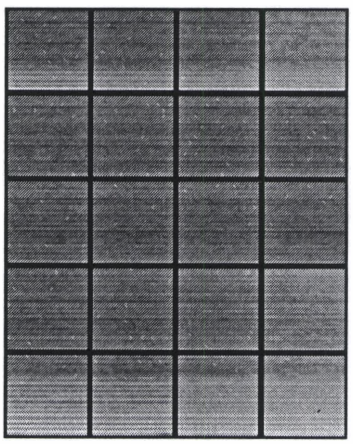

Figure 4.8. Water region (20 *32x32* samples).

4.3 Standardization of Images

Each region was standardized or normalized to remove contrast variations and biases from sample to sample. This process was a statistical rescaling. The mean of each region was scaled to zero and the variance was scaled to unity. This process insured that scaling of the axes was uniform for all samples of each region; hence, no single sample could dominate the calculation simply due to a difference in scaling. The standardization process consisted of combining all samples from a region into one file (*256x256* for the grass region, *256x256* for the tree region, and *128x160* for the water region). The mean and standard deviation were computed. The region matrix was then normalized such that the mean was zero and the variance was unity. A MATLAB algorithm called ***stand.m*** was generated to perform this function and is listed in Appendix A. For a large number of samples (for each region) the image histograms were expected to be Gaussian distribution (as described by the central limit theorem) [56]. The histograms in Figure 4.9 show the intensity distribution characteristics of each of the standardized sample regions. The grass region sample histogram displayed a Gaussian-like distribution; however, both the tree and water region histograms were not Gaussian distributions. The tree region histogram showed three major Gaussian-like distributions, each shifted slightly relative to the others. The water region had a substantially skewed and unsymmetrical distribution. Apparently the water samples near the shores had texture change due to higher water temperature, which skewed the distribution.

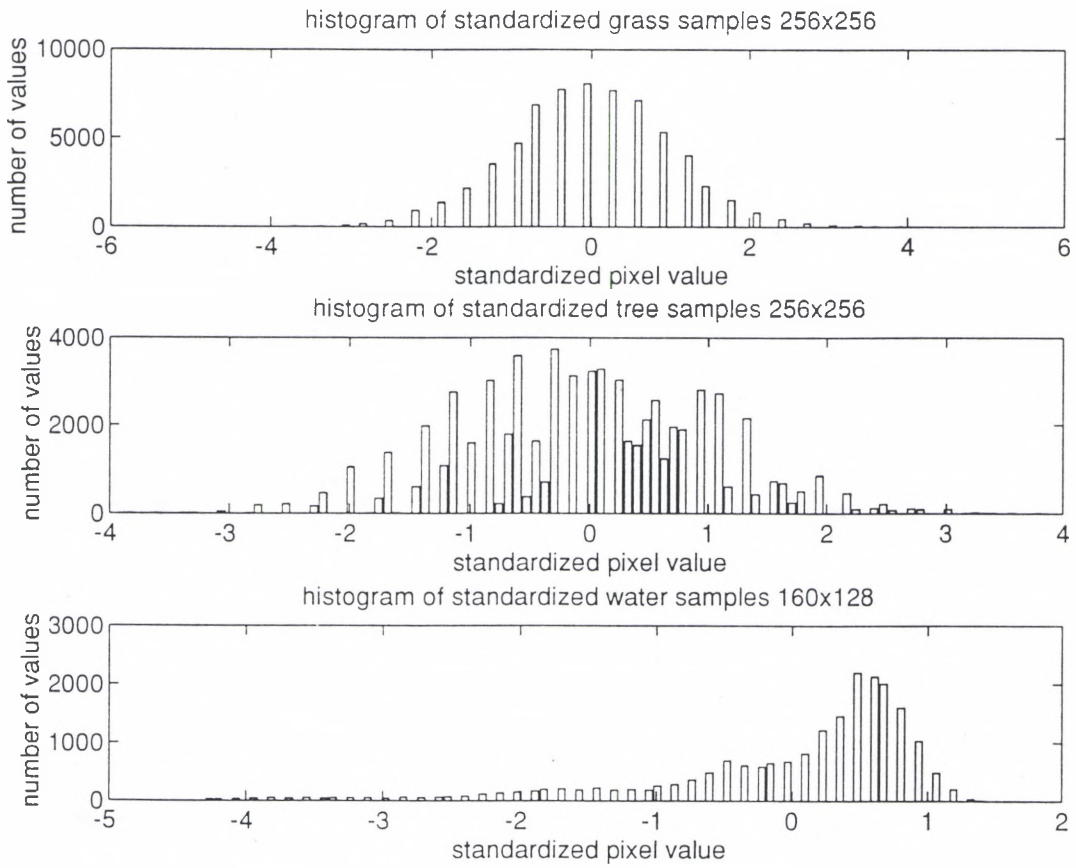

Figure 4.9. Standardized histograms of grass, tree, and water regions.

4.4 Generation of Singular Value Features

Each of the samples from the three regions was subjected to the SV feature extraction process shown in Figure 4.10 to generate feature vectors that represented the region. For each region, the singular value feature vector ($\mathbf{z_1}$), the normalized singular value feature vector ($\mathbf{z_2}$), and the moments of $\mathbf{z_2}$ were calculated. The vectors $\mathbf{z_1}$ and $\mathbf{z_2}$ each contained 32 values. The moment vector contained only four values. Although some additional experimentation was done with the scalar features discussed in the background section, they did not perform as well as the vector features for the natural background regions. Several plots are provided in Appendix B.

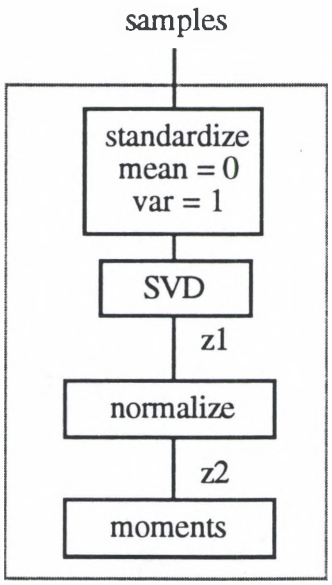

Figure 4.10. Singular value feature extraction process.

4.4.1 Singular Value Decomposition of Image Matrix

The MATLAB ***svd*** function [19] generated from the LINPACK routine [20] was used to generate the diagonal row of 32 singular values for each sample. The algorithms created to calculate and plot the singular value features were ***z1.m***, ***z2.m***, and ***z6.m*** (A listing of each is given in Appendix A).

4.4.2 Family of Singular Value Feature Vectors

Figures 4.11-4.13 show the distributions of the singular values ($\mathbf{z_1}$) for 64 samples of each of the grass and tree regions and 20 samples of the water region; their average distribution is given in Figure 4.14. Figures 4.15-4.18 show the singular value distribution curves for the normalized ($\mathbf{z_2}$) cases. The grass regions show a relatively tight clustering of curves for the 64 samples for both $\mathbf{z_1}$ and $\mathbf{z_2}$. However, both the tree and water regions have a relatively broad clustering of distribution curves for their respective 64 and 20 samples. By comparing the distribution curves of the water region, two classes of water regions can be distinguished (due to temperature differences in the samples close to the shoreline compared to the majority of the samples in cooler water).

The singular value distribution curve of each region reflects intrinsic texture characteristics. The average distribution of each region, compared with a highly structured and a random image (Figure 3.2), shows that the grass region is the most random and the water region is the most structured. The statistical variation between samples is quantified by the moments. Appendix B contains distribution plots of the moments for the grass, tree, and water regions. The relative variations of M_1, M_2, M_3, and M_4 between samples for the grass region were minor; however, for both the tree and water regions large variations were apparent.

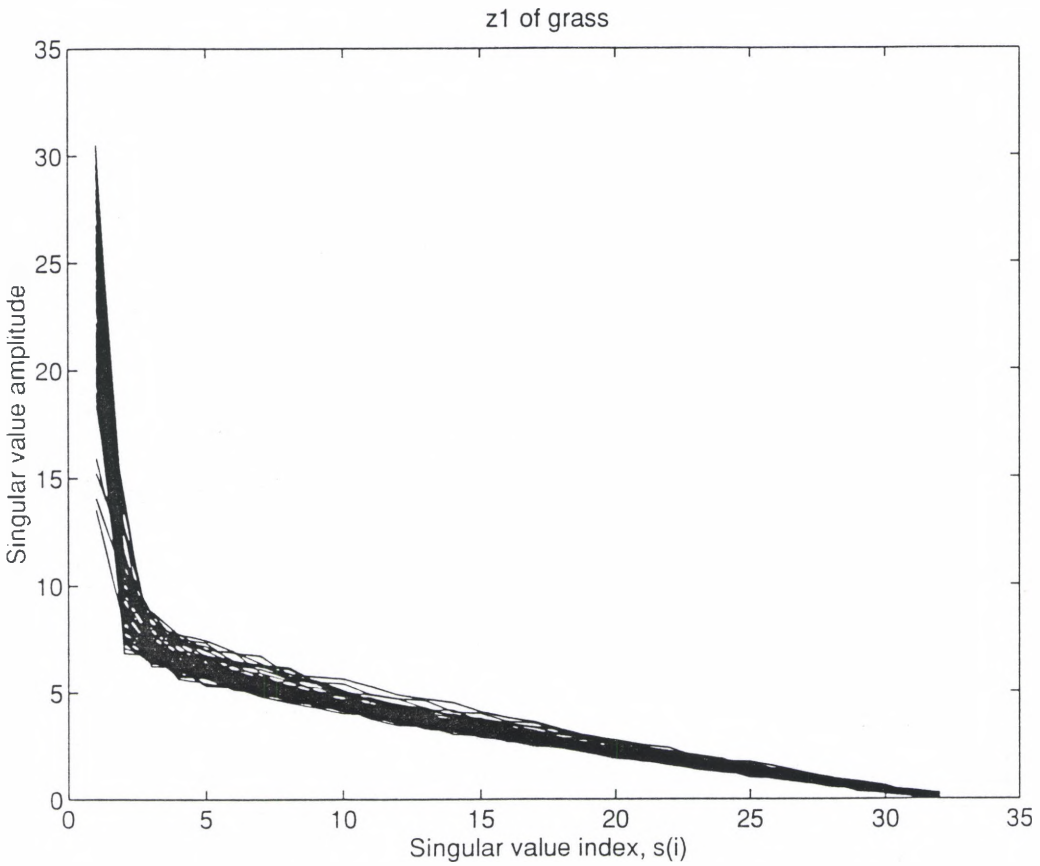

Figure 4.11. Distribution of singular values ($\mathbf{z_1}$) for 64 samples of the grass region

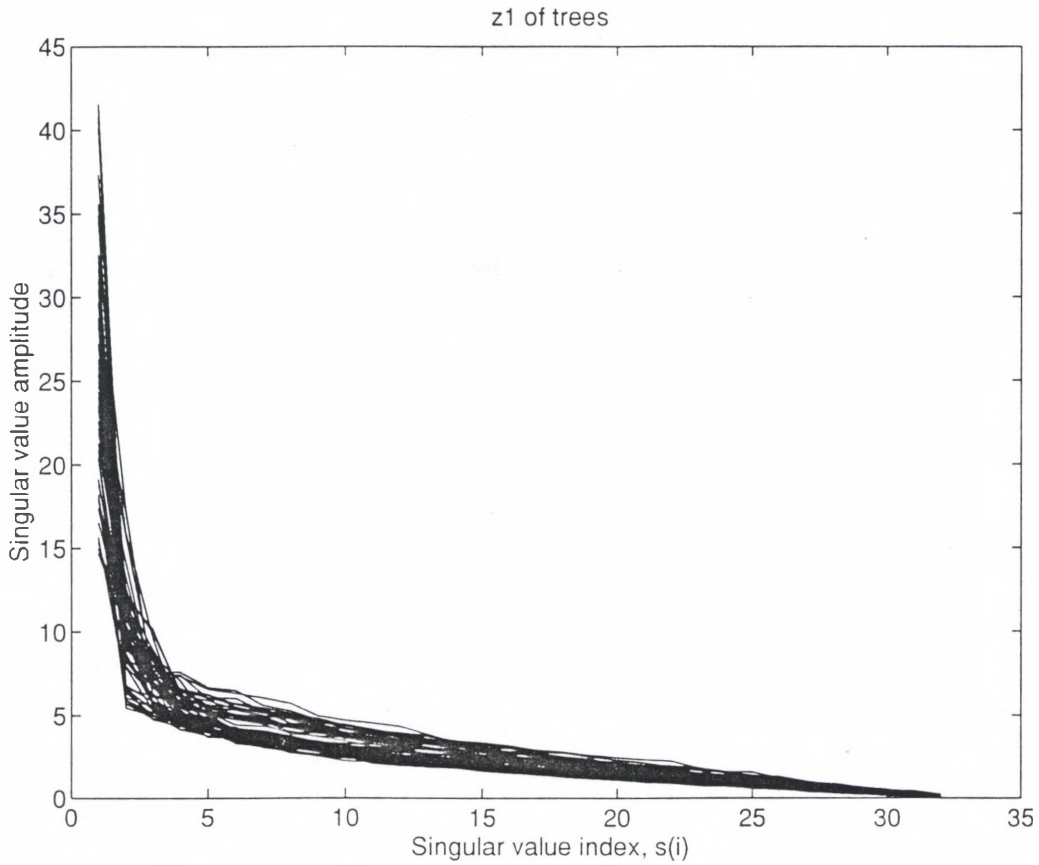

Figure 4.12. Distribution of singular values ($\mathbf{z}_1$) for 64 samples of the tree region

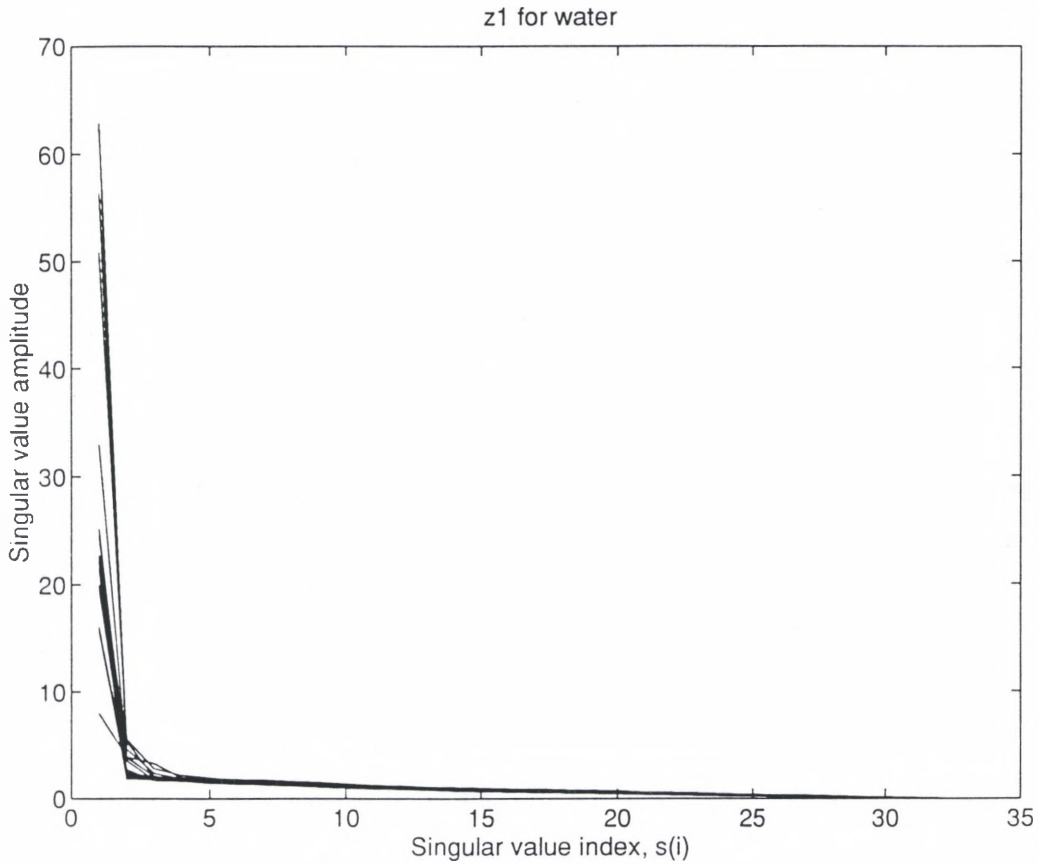

Figure 4.13. Distribution of singular values ($\mathbf{z_1}$) for 20 samples of the water region

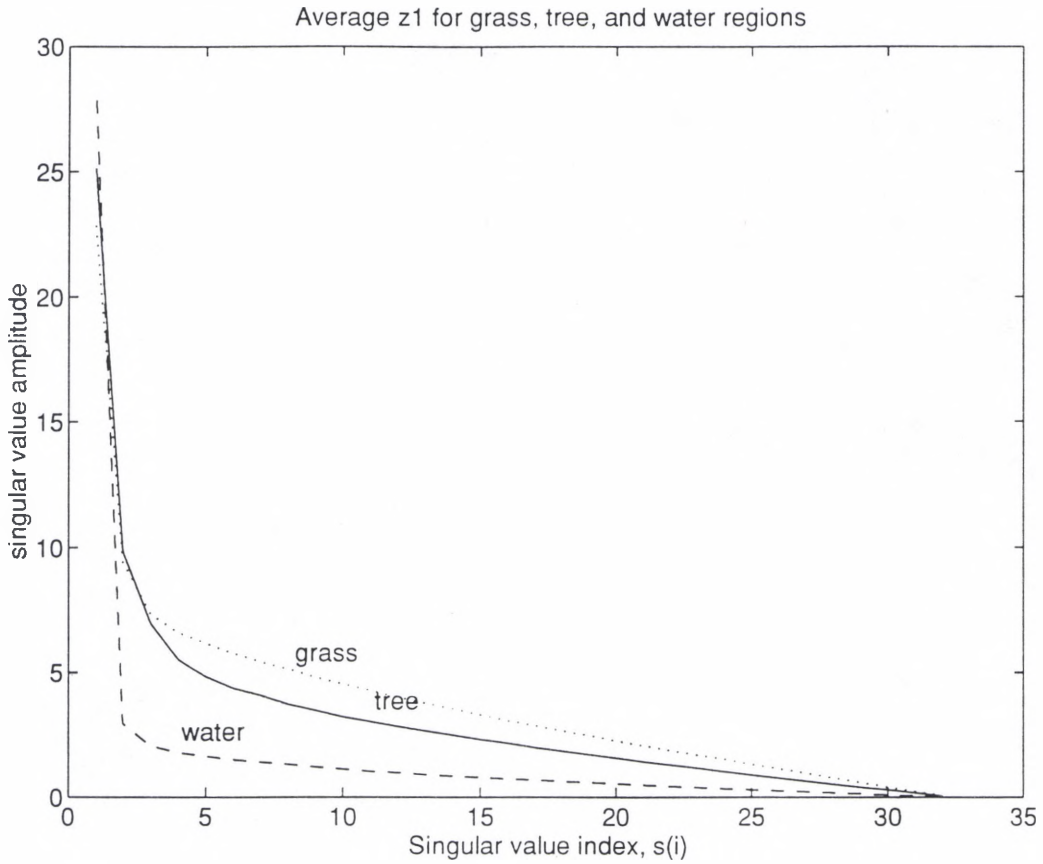

Figure 4.14. Average distribution of the singular values ($\mathbf{z}_1$) for the grass, tree, and water regions

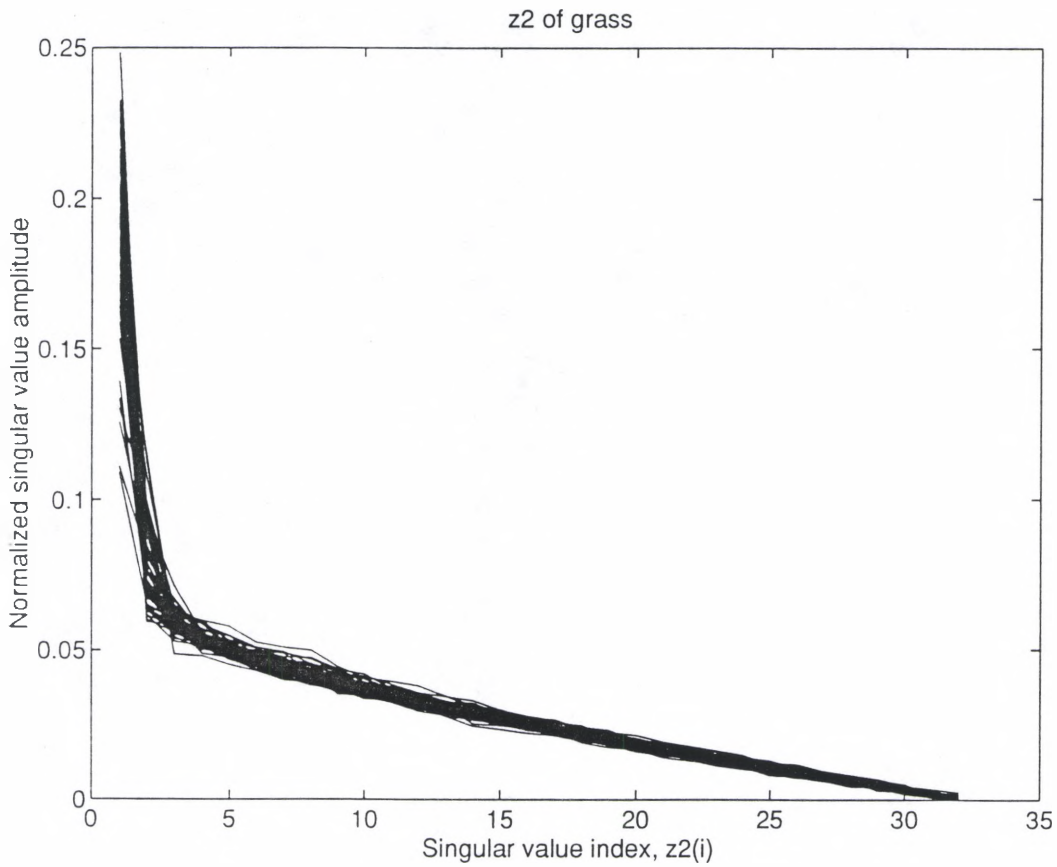

Figure 4.15. Distribution of normalized singular values ($\mathbf{z_2}$) for 64 samples of the grass region

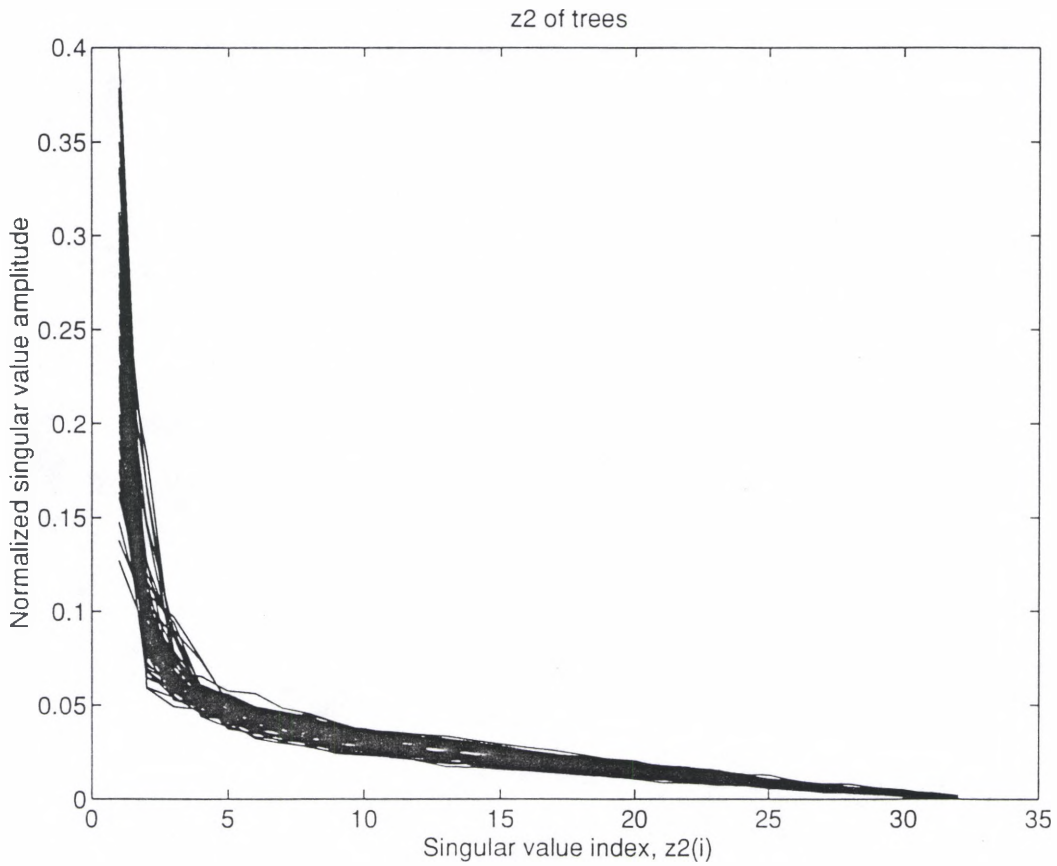

Figure 4.16. Distribution of normalized singular values ($\mathbf{z_2}$) for 64 samples of the tree region

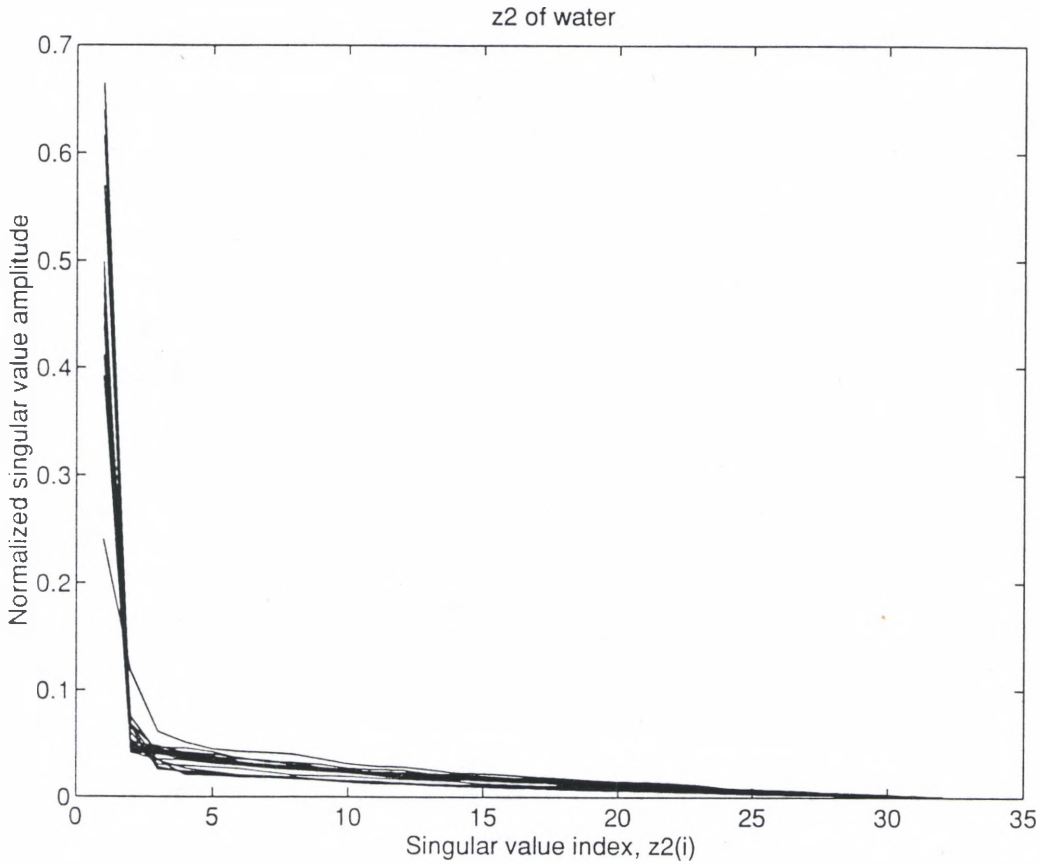

Figure 4.17. Distribution of normalized singular values ($\mathbf{z_2}$) for 64 samples of the water region

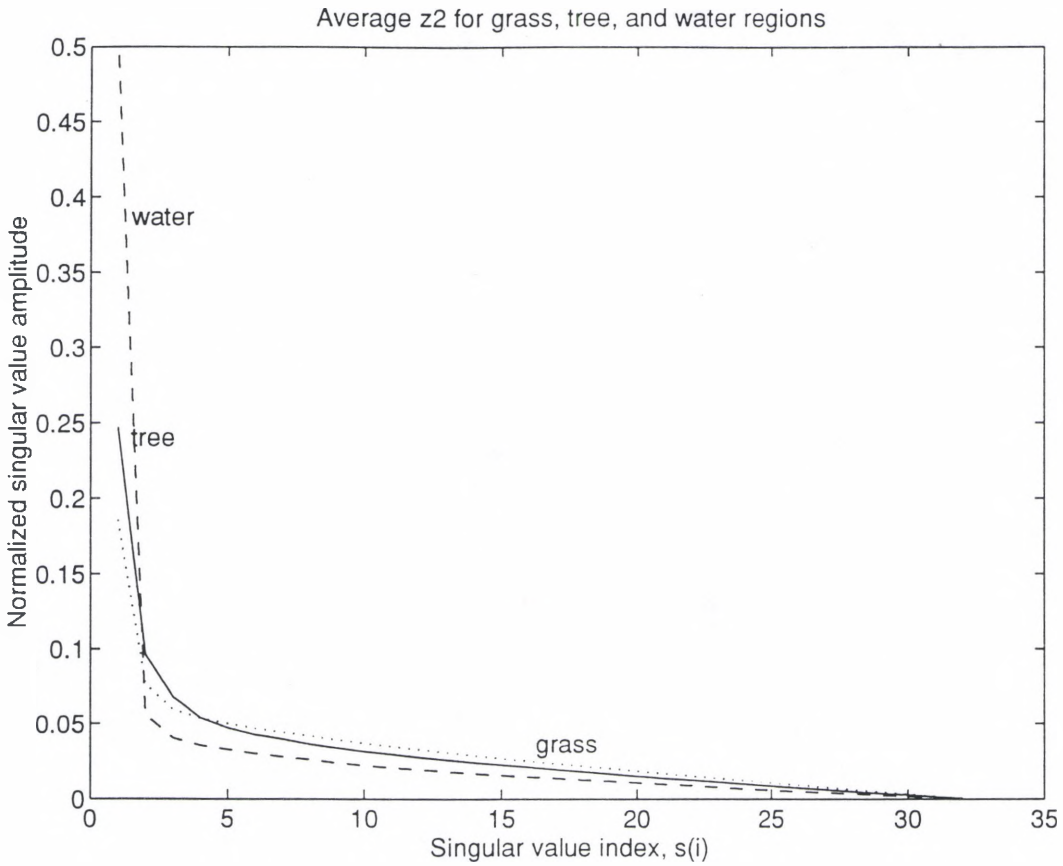

Figure 4.18. Average distribution of the normalized singular values ($\mathbf{z_2}$) for the grass, tree, and water regions

4.5 Evaluation of Classification Accuracy

The moments of the $\mathbf{z_2}$ features were evaluated using the Bhattacharyya distance criteria. Classification accuracies were determined from the upper and lower bound of the probability of error. Results for the moments of $\mathbf{z_2}$ are contained in Table 4.1. Table 4.2 shows the relative weight of the single moments for classification accuracy. The largest Bhattacharyya distances and best overall classification accuracy results came from all four moments combined (which is not surprising since they contain the most information about $\mathbf{z_2}$). The classification accuracy of M_1 was nearly as good as the four-moments vector and could be used if further reduction in feature vector size is required. The best-to-worse ranking of the individual moments of $\mathbf{z_2}$ is M_1, M_3, M_4, M_2. Combinations of lower ranking features always produced smaller Bhattacharyya distances. Scalar features did not perform as well as vector features. Finally, a visual check of the average distribution curves in Figure 4.18 revealed a direct correspondence with the regional separability of the Bhattacharyya distances.

Table 4.1

Singular value Bhattacharyya distances and classification accuracies for moments (M_1, M_2, M_3, M_4) of $\mathbf{z_2}$.

Region Pairs		Bhattacharyya-distance	Classification accuracy % (using error bound)	
			Lower Bound	Upper Bound
Grass	Tree	0.78	94.4	77.0
Grass	Water	6.88	100.0	100.0
Tree	Water	1.14	97.4	84.0

Table 4.2

Singular value Bhattacharyya distances and classification accuracies for single moments of z_2.

Moments of Regional Pairs		Bhattacharyya-distance	Classification accuracy % (using error bound)	
M_1			Lower Bound	Upper Bound
Grass	Tree	0.56	91.0	71.4
Grass	Water	2.59	99.9	96.2
Tree	Water	0.61	92.0	72.8
M_2				
Grass	Tree	0.39	86.9	66.3
Grass	Water	1.04	96.8	82.3
Tree	Water	0.33	84.9	64.2
M_3				
Grass	Tree	0.71	93.6	75.5
Grass	Water	1.66	99.1	90.5
Tree	Water	0.46	88.7	68.4
M_4				
Grass	Tree	0.79	94.5	77.3
Grass	Water	1.52	98.8	89.1
Tree	Water	0.50	89.8	69.7

CHAPTER V

COMPARISON OF SV FEATURES WITH POWER SPECTRUM FEATURES

5.1 Introduction

This chapter provides a comparison of the singular value classification method and a Fourier power spectrum method. Thirty-two Fourier power spectrum features were generated using a wedge-ring sampling approach. Bhattacharyya distances and classification accuracies were calculated and compared with the singular values results of chapter IV.

5.2 Generation of Power Spectrum Features

The Fourier transform of a image $f(x,y)$ is defined by [3]

$$F(u,v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-i\,2\pi(ux + vy)} f(x,y)\, dx\, dy$$

and the Fourier power spectrum is $|\boldsymbol{F}|^2 = \boldsymbol{F}\boldsymbol{F}^*$ (where * represents the complex conjugate).

The power spectrum can be examined radially and angularly to determine the uniqueness of an image. The radial distribution of values in $|\boldsymbol{F}|^2$ are sensitive to texture coarseness from the pattern of the image. Coarse texture will have high values of $|\boldsymbol{F}|^2$ concentrated near the origin, while $|\boldsymbol{F}|^2$ values for a fine texture image will be more spread out. Because of the symmetry of the power spectrum, coarseness of an image can be measured by sampling ring regions about the center peak, i.e.,

$$\phi_r = \int_0^{2\pi} |F(r,\theta)|^2 \, d\theta$$

Similarly, the angular distribution of values in the power spectrum are sensitive to the directionality of the texture in the image. Texture with many edges or lines in a given direction θ will have high values concentrated around the perpendicular direction θ + π/2. Thus, angular wedges are a good set of features for analyzing texture directionality and can be given as

$$\phi_\theta = \int_0^{r} |F(r,\theta)|^2 \, dr$$

For the *32x32* image samples the discrete Fourier transform (implemented with the fast Fourier transform) was used, i.e.

$$F(u,v) = \frac{1}{n^2} \sum_{j,k=0}^{n-1} f(j,k)\, e^{-i2\pi(ju+kv)}$$

This transform, however, treats the input image as periodic. If it is not, then the transform is affected by the discontinuities that exist between one edge of the image and the opposite edge. These have the effect of introducing spurious horizontal and vertical directionality, so that high values are present in the power spectrum along the u and v axes. The wedge-ring features are sensitive to size or orientation only, but not to both.

The intersection of eight wedges and four rings were selected to generate a total of 32 segments from the power spectrum. Because of symmetry, only half the power spectrum was needed; it was divided into eight wedges of 22.5° each and four ring segments of equal area (Figure 5.1). Equal area segments were selected so that each segment carried the

same weight (i.e., each segment had approximately the same number of pixels). The radii for the ring segments were $r = (0,8), (8,8\sqrt{2}), (8\sqrt{2},8\sqrt{3})$ and $(8\sqrt{3},16)$; where each radius was measured by pixels from the center of the spectrum (taken to be [16,16]). More rings were preferred, however only four rings were possible because the outer ring segments quickly became very thin (1-2 pixels wide). For the half portion of the *32x32*-pixel power spectrum used, each segment contained approximately 16 pixels. The 32 power spectrum features were generated by taking the sum of the pixel power spectral density amplitude for each segment. No attempt to was made to rank-order the features. The feature order was taken from inside the first ring, beginning with the right side wedge, rotating through the wedges to the left side, then between the first and second rings, rotating again from the right to the left side, and so on to the outer most ring (Figure 5.1). This order was chosen to maintain the uniqueness of the directionality and spreading of the texture spatial frequencies.

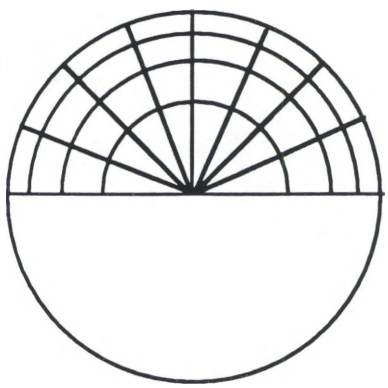

Figure 5.1. Wedge-ring filter for power spectrum feature calculations.

The MATLAB ***fft2*** [19] fast Fourier transform algorithm was used for the discrete Fourier transform calculation. A MATLAB function called ***p1.m*** (listed in Appendix A) was generated to incorporate the fast Fourier transform and power spectrum calculations,

and the wedge-ring segmenting for *32x32*-pixel samples. The input matrix is the same standardized image matrix used for each region in the singular value method. The 32 features generated using ***p1.m*** are analogous to the 32 singular value features created with ***z1.m***. An additional algorithm called ***p2.m*** (listed in Appendix A) was generated to create normalized features analogous to ***z2.m***, and ***z6.m*** was used to generate a reduced moment vector of the four features. The 32 power spectrum feature vector is thus referred to as $\mathbf{p}_1$ and the normalized power spectrum feature vector is referred to as $\mathbf{p}_2$. The power spectrum features $\mathbf{p}_1$, $\mathbf{p}_2$, and their averages are shown in Figures 5.2-5.9 for the grass, tree, and water regions.

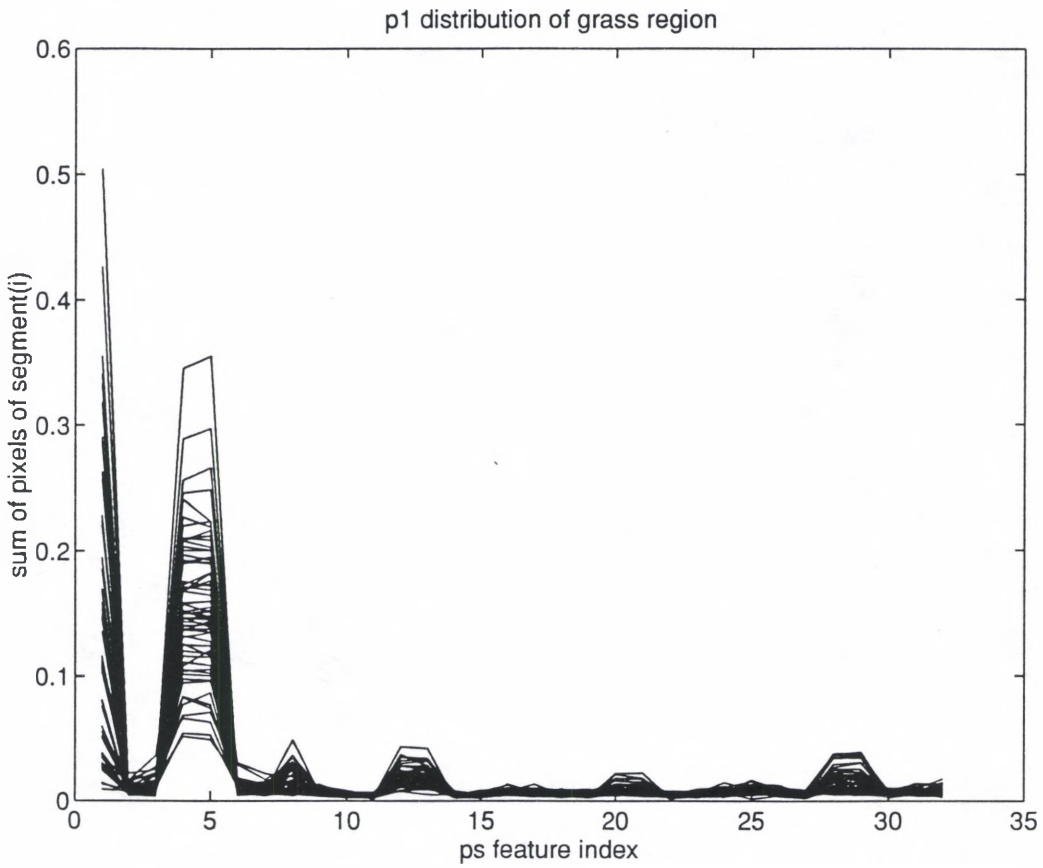

Figure 5.2. Distribution of Fourier power spectrum features ($\mathbf{p_1}$) for 64 samples of the grass region.

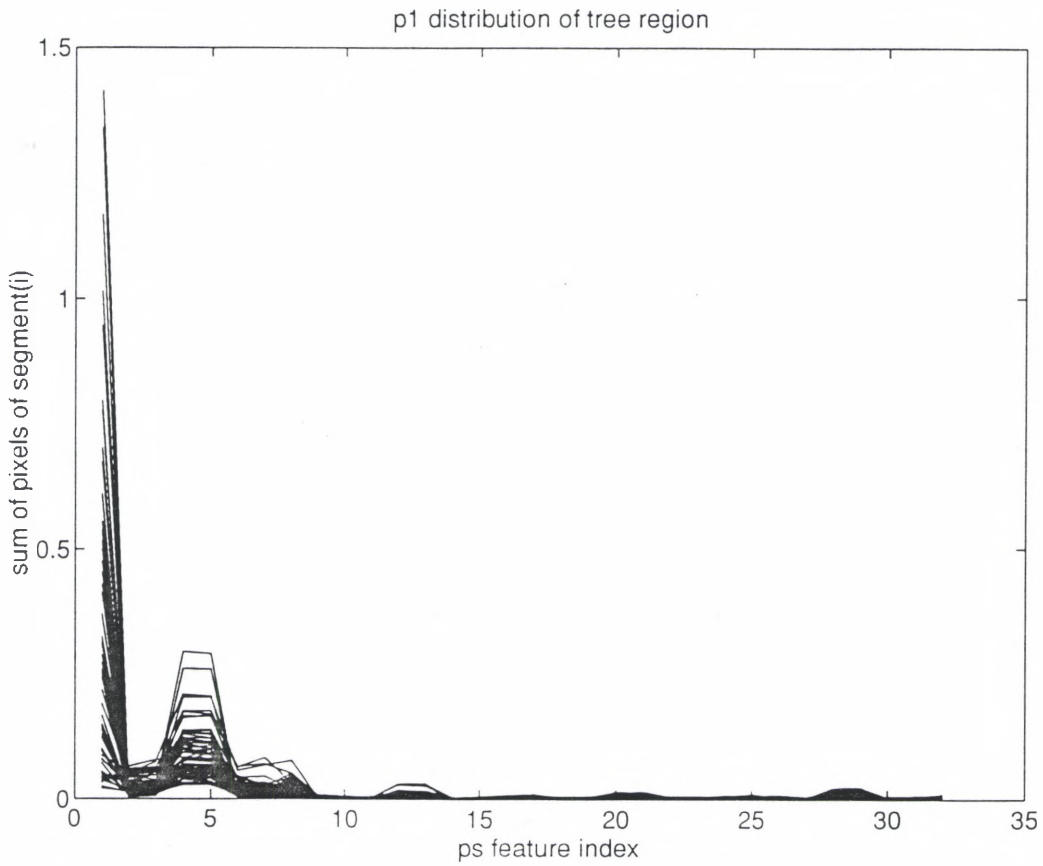

Figure 5.3. Distribution of Fourier power spectrum features ($\mathbf{p_1}$) for 64 samples of the tree region.

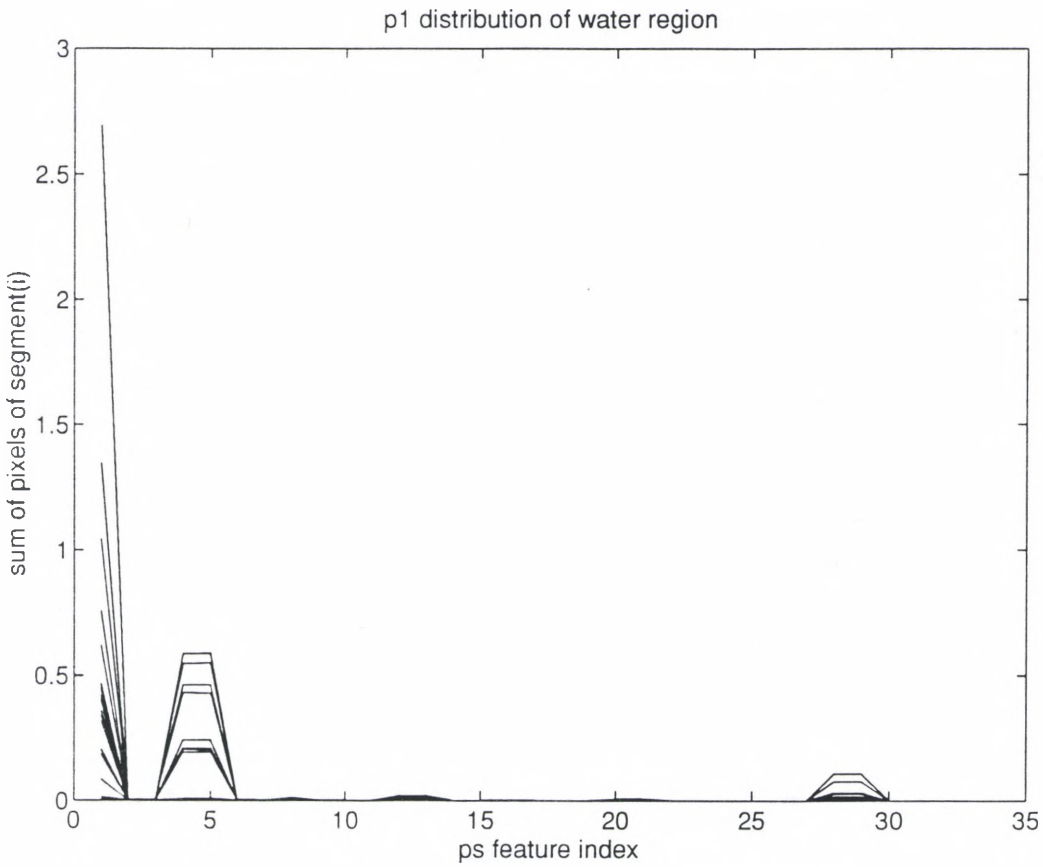

Figure 5.4. Distribution of Fourier power spectrum features ($\mathbf{p_1}$) for 20 samples of the water region.

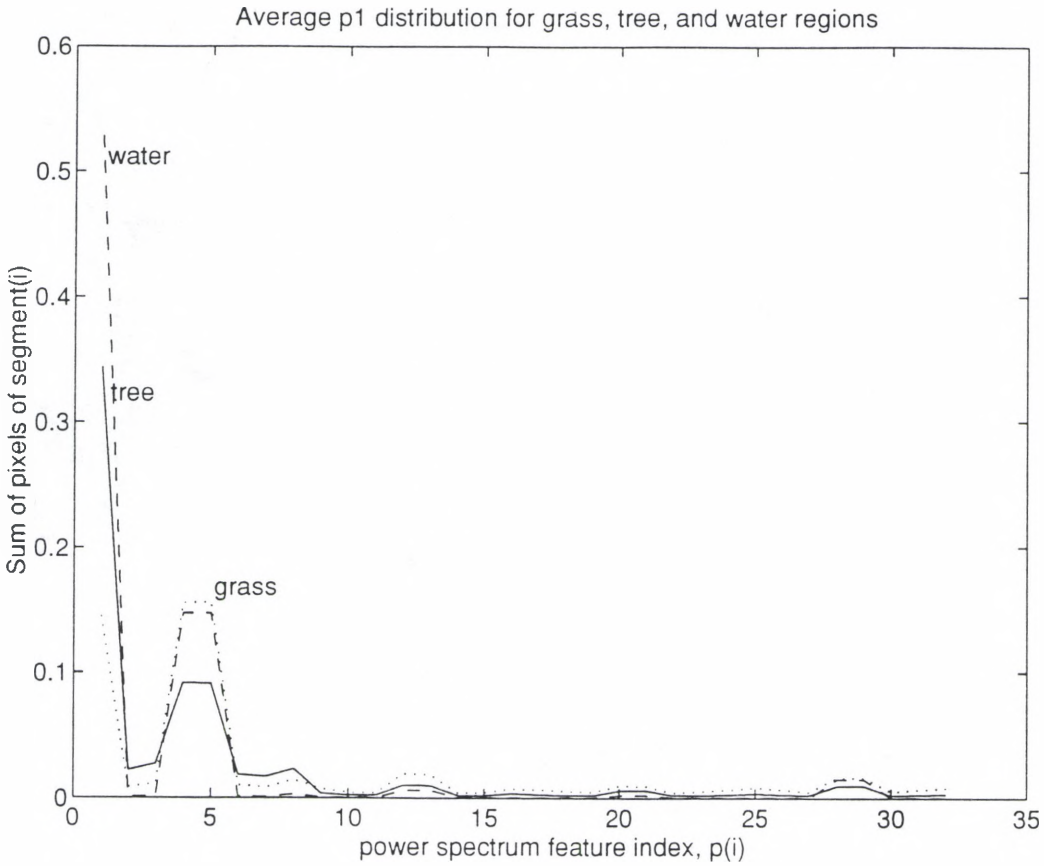

Figure 5.5. Average distribution of the Fourier power spectrum features ($\mathbf{p}_1$) for the grass, tree, and water regions.

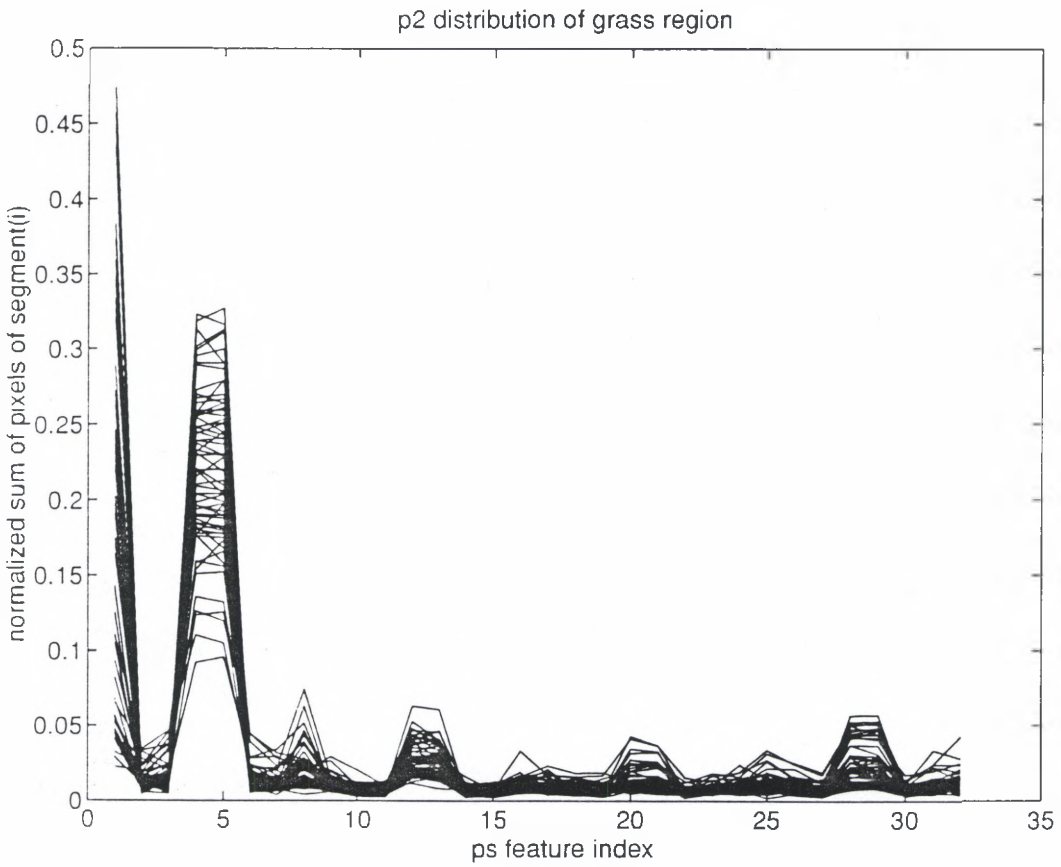

Figure 5.6. Distribution of normalized Fourier power spectrum features ($\mathbf{p_2}$) for 64 samples of the grass region.

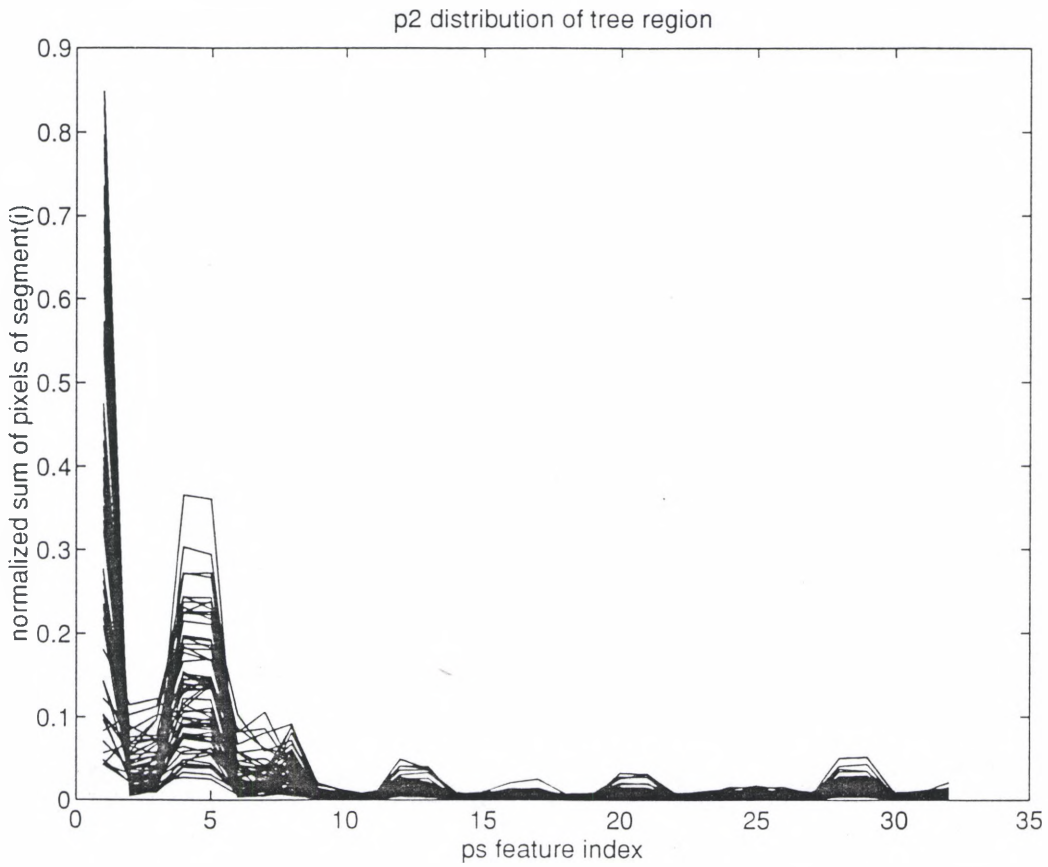

Figure 5.7. Distribution of normalized Fourier power spectrum features ($\mathbf{p_2}$) for 64 samples of the tree region.

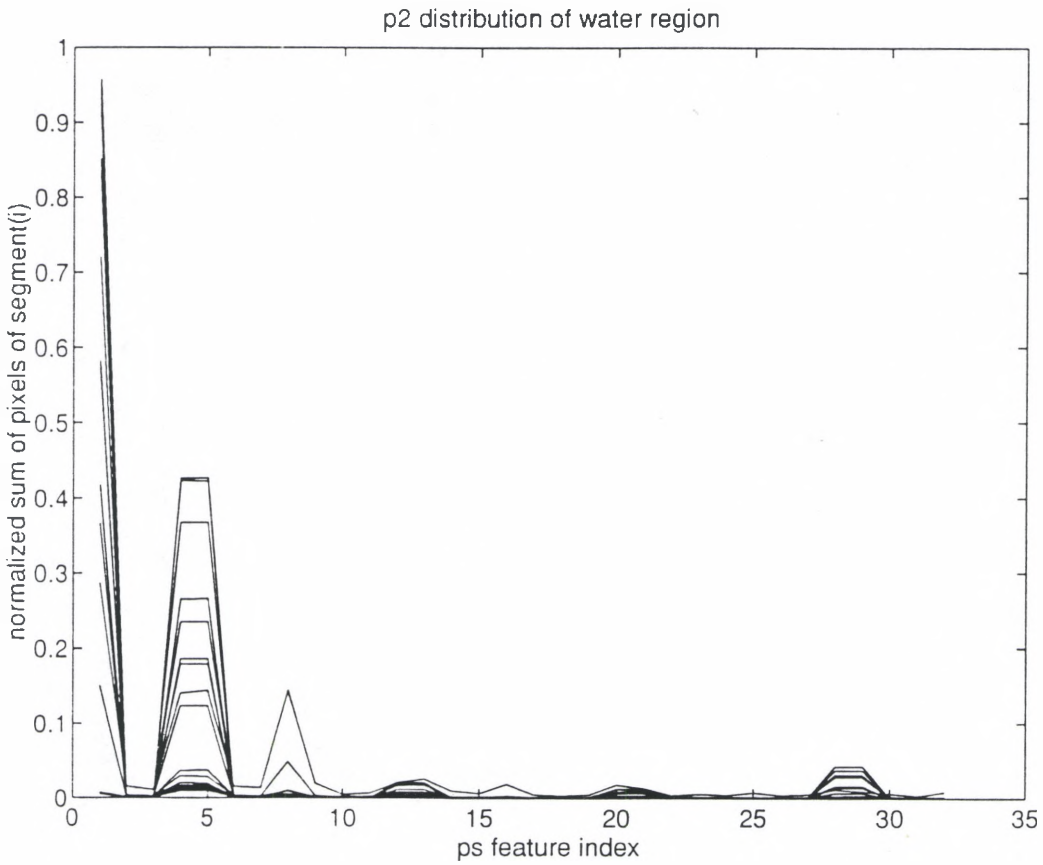

Figure 5.8. Distribution of normalized Fourier power spectrum features ($\mathbf{p_2}$) for 20 samples of the water region.

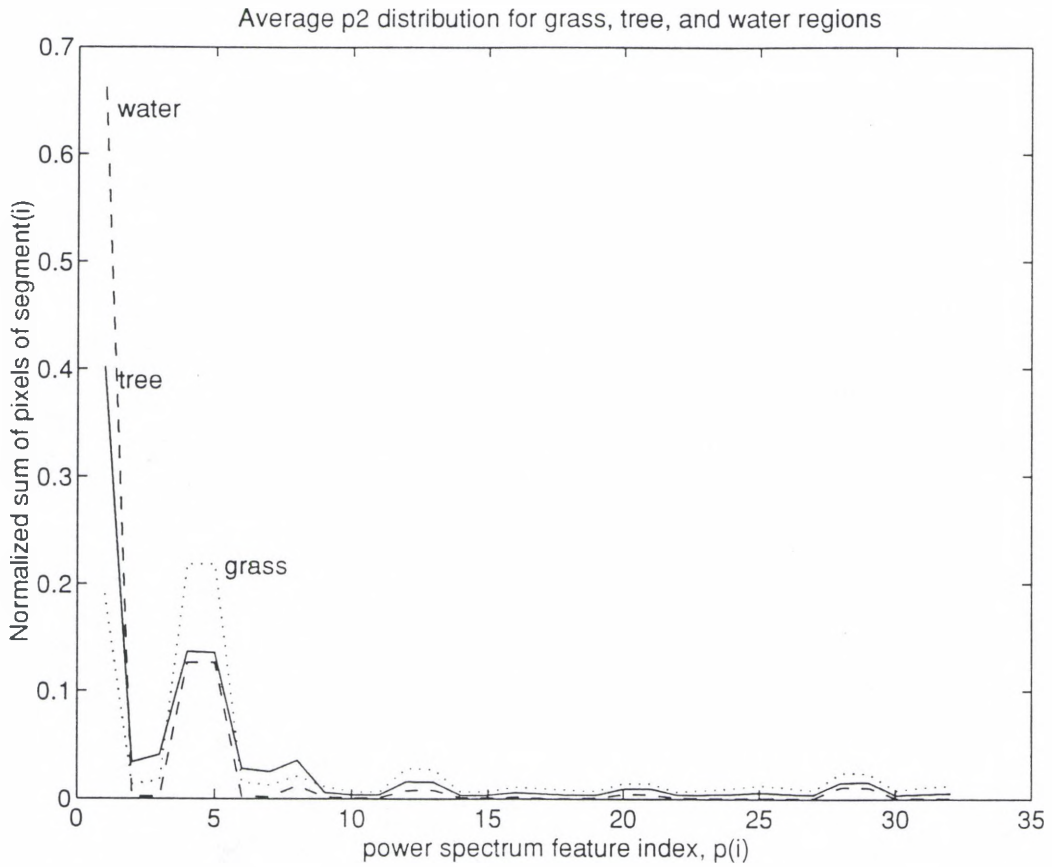

Figure 5.9. Average distribution of the normalized Fourier power spectrum features ($\mathbf{p}_2$) for the grass, tree, and water regions.

5.3 Comparison of Bhattacharyya Distance and Classification Accuracies

Table 5.1 gives Bhattacharyya distance and classification accuracy results for the moments of $\mathbf{p_2}$ for the Fourier power spectrum wedge-ring method. Table 5.2 contains the moments of $\mathbf{z_2}$ for the singular value features of Chapter IV.

Table 5.1

Power Spectrum Bhattacharyya distances and classification accuracies for moments (M_1, M_2, M_3, M_4) of $\mathbf{p_2}$.

Region Pairs		Bhattacharyya-distance	Classification accuracy % (using error bound)	
			Lower Bound	Upper Bound
Grass	Tree	0.48	89.2	68.9
Grass	Water	2.54	99.9	96.1
Tree	Water	0.87	95.4	79.1

Table 5.2

Singular value Bhattacharyya distances and classification accuracies for moments (M_1, M_2, M_3, M_4) of $\mathbf{z_2}$.

Region Pairs		Bhattacharyya-distance	Classification accuracy % (using error bound)	
			Lower Bound	Upper Bound
Grass	Tree	0.78	94.4	77.0
Grass	Water	6.88	100.0	100.0
Tree	Water	1.14	97.4	84.0

5.4 Summary

Fourier power spectrum features were generated using a wedge-ring sampling configuration for comparison with the singular value features of Chapter IV. The same standardized image regions were used; 32 features ($\mathbf{p}_1$), a normalized set of 32 features ($\mathbf{p}_2$), and the moments of each were generated. The power spectrum results were comparable to the singular value results. Only the reduced vector moments of $\mathbf{p}_2$ were generated and compared to $\mathbf{z}_2$; this feature set (moments of $\mathbf{z}_2$) had the best results for the singular value analysis. Comparing the statistically normalized features ($\mathbf{z}_2$ and $\mathbf{p}_2$), the moments of the singular values ($\mathbf{z}_2$) had better classification accuracies than the moments of the power spectrum values ($\mathbf{p}_2$) for all the combinations of regions.

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusions

This effort applied SVD methods to evaluating the texture characteristics of natural background scenes. Singular values obtained using singular value decomposition provided a reduced one-dimensional feature space of texture attributes for several natural scene regions. Further reduction was obtained by using only the first four moments of the singular value feature vectors, thus reducing the feature vector to four numbers. Although there were substantial variations within regional texture samples, good classification results were obtained using the singular value features. The highest classification accuracy (100 percent) was obtained for separating grass from water regions. The worst classification accuracy (77 percent) was obtained for separating grass from tree regions. Singular value feature results were also compared with Fourier power spectrum features. The singular value features provided slightly better overall classification results than the Fourier power spectrum features. However, the SVD technique did not provide a substantial improvement over techniques that have been presented in the literature. Large variations between samples in the water and tree regions certainly contributed to the difficulty in discriminating between regions. For the water regions, sun radiation changed the basic structure of the image texture, and samples near banks had warmer temperatures and thus different intensity and texture patterns. For the tree regions, large variations in texture resulted from variations in range and from using *32x32*-pixel sample sizes. Even *64x64*-sample sizes were not large enough to capture homogeneous texture for the tree regions.

This method could be used to provide a quantitative measure of the quality of synthetic relative to real background images for similar scenes. The singular values provide a good measure of the intrinsic characteristics of a texture pattern region. However, samples must be restricted to a homogeneous texture pattern to provide excellent discrimination results.

The statistical evaluation methods, such as the Bhattacharyya error bound worked best for normal distribution of features. The histograms of features from both the SVD and Fourier power spectrum techniques were not ideally normal. More samples in each region would provide a better statistical representation to evaluate the effectiveness of both techniques.

Computationally, the SVD method was much faster than the power spectrum method. Much of the difference could be attributed to the algorithms written (many improvements could be made to streamline the calculations and more effectively use the *MATLAB* functions and environment). In fact, using benchmark tests on a Macintosh SE (with a 33 MHz accelerator and 68882 math co-processor) the fast Fourier transform calculation was much faster than the SVD calculation (0.42 seconds vs. 3.5 seconds for a random *32x32* matrix). Optical techniques could improve the processing speed for both techniques.

The robustness of the SVD technique needs to be explored further. The test set was very limited in scope and used relatively good images. Additional samples of noisier and degraded images need to be explored. In addition, fixed noise patterns from background environment or the sensor could be a problem and should be investigated.

6.2 Recommendations

The techniques described here could provide a tool for evaluating synthetic background relative to real background scenes. Unfortunately, synthetic imagery (or software

to create synthetic imagery) of background scenes similar in texture to the FLIR imagery was not available. The **GTVISIT** [57] software developed to produce synthetic targets and scenes was examined, but it was limited to a few types of foliage scenes (coniferous trees and one kind of shrub), so deciduous trees, grass, and water texture background could not be created. A follow-on study could compare synthetic scenes with real natural scenes when images and/or software becomes available.

The equal area filter may not be the best choice for sampling the Fourier power spectrum. The outer ring segments were very narrow and consequently did not provide much information on the texture spatial characteristics of the spectrum. Other techniques like uniform radial concentric rings or constant ratio increments should be explored.

A neural network is ideally suited to use all 32 features for classification. It could potentially provide improved classification performance over the four moment feature vectors used in this study.

Application of texture feature extraction obviously extends beyond the IR FLIR images used for this study. Visible, multispectral, radar, laser radar, and other imagery provide texture patterns that could be evaluated with this technique. In fact, evaluating satellite imagery of large regional areas is probably a more attractive application for SVD techniques because variations in range at extreme distances have little impact on the homogeneity of texture patterns.

Real time applications of SVD techniques, including optical implementations, should also be explored.

APPENDICES

APPENDIX A

MATLAB SOFTWARE ALGORITHMS

This section contains the MATLAB algorithms developed to perform much of the analysis for this project. No attempt was made to optimize the algorithms for speed or efficiency. The following is a brief summary of each algorithm:

stand.m - standardizes the initial 8-bit images (i.e. grass, tree, and water) to remove biases between samples. The mean becomes zero and the variance is unity.

z1.m - calculates the singular values for *32x32* sample matrices for each region and plots the singular value amplitude vs. singular value index for all sample matrices.

z2.m - normalizes the singular values of $\mathbf{z_1}$ for each region and plots the normalized singular value amplitude vs. singular value index for all sample matrices.

z6.m - calculates the moments (M_1, M_2, M_3, M_4) for $\mathbf{z_1}$ or $\mathbf{z_2}$, and plots the distribution.

p1.m - calculates the Fourier power spectrum for each sample, samples the spectrum with a wedge-ring filter, and then generates and plots 32 power spectrum features ($\mathbf{p_1}$).

p2.m - normalizes and plots the 32 power spectrum features of $\mathbf{p_1}$ ($\mathbf{p_2}$).

B_dist.m - calculates the Bhattacharyya distance and classification accuracy between two classes of features.

stand.m

```
function y=stand(n)

% Perform standardization of images from 0-255 gray-level
% Checks for gaussian distribution
% rescales (standardizes) data by zeroing mean and
%   normalizing the variance to unity

[r,c]=size(n);

m=reshape(n,r*c,1);

% is it gaussian?
hist(m,100);
%    pause

% remove mean;
zerom=m-mean(m);

% normalize variance to one
yy=zerom/std(zerom);

% check mean,variance
mean_y=mean(yy)
var_y=cov(yy)

% look at histogram of standardized image
hist(yy,100)

% return to [r,c] form
y=reshape(yy,r,c);
```

z1.m

```
function y=z1(g)

% calculates sv (z1)
% make 32x32 segments out of large image matrix
% minimum size 32x32 matrix
% input (g) = standardize grass,trees or water

[r,c]=size(g);
m=0;
for i=1:32:r,
     for j=1:32:c,
          m=m+1;
          z1(1:32,m)=svd(g(i:i+31,j:j+31));
     end
end
% plot sv
y=z1;
semilogy(1:32,y)
```

z2.m

```
function y=z2(g)

% calculates the normalized sv (z2)
% make 32x32 segments out of large image
% minimum size 32x32 matrix
% input g = standardized grass,trees or water matrix

[r,c]=size(g);
m=0;
for i=1:32:r,
     for j=1:32:c,
          m=m+1;
          z1(1:32,m)=svd(g(i:i+31,j:j+31));
     end
end
% plot sv
xx=0:31;
% normalize to z1/sum(column)=z2
sumz1=sum(z1);
for mm=1:m,
          z2(:,mm)=z1(:,mm) ./sumz1(mm);
end
y=z2;
semilogy(xx,z2)
```

z6.m

```
function y=z6(z2)

% calculates M1, M2, M3, M4 moments of input feature vector
% moments defined in Ashjari
% input matrix z2, z1, p1 or p2
% output z6=[M1;M2;M3;M4]'

[r,c]=size(z2);
Mo=sum(z2);

% mean - M1
m=0;
for n=1:r,
  m=m+(n*z2(n,:));
end
M1=m;

% std deviation - M2
m=0;
for n=1:r,
  m=m+((n-(M1 ./Mo)) .^2) .*z2(n,:);
end
M2=sqrt(m);

% skewness - M3
m=0;
for n=1:r,
  m=m+(((n-(M1 ./Mo)) .^3) .*z2(n,:));
end
M3=(1 ./(M2 .^3)) .*m;

% Kurtosis - M4
m=0;
for n=1:r,
  m=m+(((n-(M1 ./Mo)) .^4) .*z2(n,:)-(3 ./(Mo)));
end
M4=(1 ./(M2 .^4)) .*m;

z6=[M1;M2;M3;M4]';
y=z6;
plot(z6)
title('moments of z1(water)')
xlabel('sample index')
ylabel('moment value')
text(5,z6(5,1),'M1')
text(10,z6(10,2),'M2')
text(15,z6(15,3),'M3')
text(25,z6(25,4),'M4')
```

p1.m

```
function y=p1(samp)

%  calculate Fourier power spectrum and generates
%    ring/wedge filter to create 32 features
%  samp is large matrix of n 32x32 samples of grass, tree,
%    water regions

[row,col]=size(samp);
[x,y]=meshdom(-16:16,-16:16);
r=sqrt(x.^2 + y.^2);

%  wedge calculations
th=atan2(y,x);
tha=th*180/pi;
n=22.5;

%  take one 32x32 sample at a time and calculate wedge/ring
%    features
ct=0;
for i=1:32:row,
 for j=1:32:col,
   ct=ct+1;

   ps(i:i+31,j:j+31)=fft2(samp(i:i+31,j:j+31))/32^2
.*conj(fft2(samp(i:i+31,j:j+31))/32^2);

% fftshift to work with center ps
ps(i:i+31,j:j+31)=fftshift(ps(i:i+31,j:j+31));

% only need half of spectrum
ps(i+17:i+31,j:j+31)=zeros(15,32);

% diagnostics - look at ps before calculations
mesh(ps(i:i+31,j:j+31));pause(1)

%  calculate 4 rings

for k=1:4,
     g=zeros(32);
     ri=sqrt(k-1)*8;
     ro=sqrt(k)*8;
     for ii=1:17,
          for jj=1:32,
               u=r(ii,jj);
               if u<ro & u>=ri
                    g(ii,jj)=1;
               end
          end
     end
```

```
%     calculate 8 wedges
      for w=1:8,
            au=(w*n);
            al=au-n;
            a=zeros(32);
            for ii=1:17,
                  for jj=1:32,
                        u=tha(ii,jj);
                        if abs(u)<=au & abs(u)>=al
                              a(ii,jj)=1;
                        end
                  end
            end
            cc=a .*g;
%           calculate ps for each segment
            psss=ps(i:i+31,j:j+31) .*cc;
            psff(k,w)=sum(sum(psss));
%           diagnostics
%           mesh(psss);title(['ro =',num2str(ro),' ri
=',num2str(ri),' wl =',num2str(al),' wu
=',num2str(au)]);pause(1)
      end
%     end of wedges
      end
%     end of rings
      psfr=reshape(psff',32,1);
      psf(1:32,ct)=psfr;
 end
end
p1=psf;
y=p1;
```

p2.m

```
function y=p2(samp)

%  calculates Fourier power spectrum and generates
%    ring/wedge filter to create 32 normalized features
%  samp is large matrix of n 32x32 samples of grass, tree,
%    water
%  same algorithm as p1.m, except normalizes p1 to create p2

[row,col]=size(samp);
[x,y]=meshdom(-16:16,-16:16);
r=sqrt(x.^2 + y.^2);

%  wedge calculations
th=atan2(y,x);
tha=th*180/pi;
n=22.5;

%  take one 32x32 sample at a time and calculate wedge/ring
%   features
ct=0;
for i=1:32:row,
 for j=1:32:col,
   ct=ct+1;

   ps(i:i+31,j:j+31)=fft2(samp(i:i+31,j:j+31))/32^2
.*conj(fft2(samp(i:i+31,j:j+31))/32^2);

% fftshift to work with center ps
ps(i:i+31,j:j+31)=fftshift(ps(i:i+31,j:j+31));

% only need half of spectrum
ps(i+17:i+31,j:j+31)=zeros(15,32);

% diagnostics - look at ps before calculations
mesh(ps(i:i+31,j:j+31));pause(1)

%  calculate 4 rings

for k=1:4,
     g=zeros(32);
     ri=sqrt(k-1)*8;
     ro=sqrt(k)*8;
     for ii=1:17,
          for jj=1:32,
               u=r(ii,jj);
               if u<ro & u>=ri
                    g(ii,jj)=1;
               end
          end
     end
```

```
%     calculate 8 wedges
      for w=1:8,
            au=(w*n);
            al=au-n;
            a=zeros(32);
            for ii=1:17,
                  for jj=1:32,
                        u=tha(ii,jj);
                        if abs(u)<=au & abs(u)>=al
                              a(ii,jj)=1;
                        end
                  end
            end
            cc=a .*g;
%           calculate ps for each segment
            psss=ps(i:i+31,j:j+31) .*cc;
            psff(k,w)=sum(sum(psss));
%           diagnostics
%           mesh(psss);title(['ro =',num2str(ro),' ri
=',num2str(ri),' wl =',num2str(al),' wu
=',num2str(au)]);pause(1)
      end
%     end of wedges
      end
%     end of rings
      psfr=reshape(psff',32,1);
      psf(1:32,ct)=psfr;
 end
end
p1=psf;

% calculate p2 - normalized features
sumps=sum(p1);
for mm=1:ct,
      p2(:,mm)=p1(:,mm) ./sumps(mm);
end
plot(p2)
y=p2;
```

B_dist.m

```
function b=B_dist(f1,f2)

% input class1 (f1) and class2 (f2)
% calculates the B-distance between the two classes
% upper/lower bound on classification accuracy determined
% f1 and f2 - features in columns and samples in rows
% mean creates 1 column with mean for each feature [1xn]

b=(1/8 *(mean(f1)-mean(f2)) *(inv((cov(f1)+cov(f2))/2))
*(mean(f1)-mean(f2))') +(.5*logm(norm(.5*(cov(f1) +cov(f2)))
/(norm(cov(f1)^.5)*norm(cov(f2)^.5))));

% calculate the Bayes probability of error

% upper bound
eu=.5*exp(-b);

% lower bound
el=.5-.5*sqrt(1-exp(-2*b));

% classification accuracy
ub=(1-eu)*100
ul=(1-el)*100
```

APPENDIX B

PLOTS

This appendix contains distribution plots of the moments of $\mathbf{z_2}$ and $\mathbf{p_2}$, and the distribution of the scalar features z_{10} for $\mathbf{z_1}$ and $\mathbf{z_2}$.

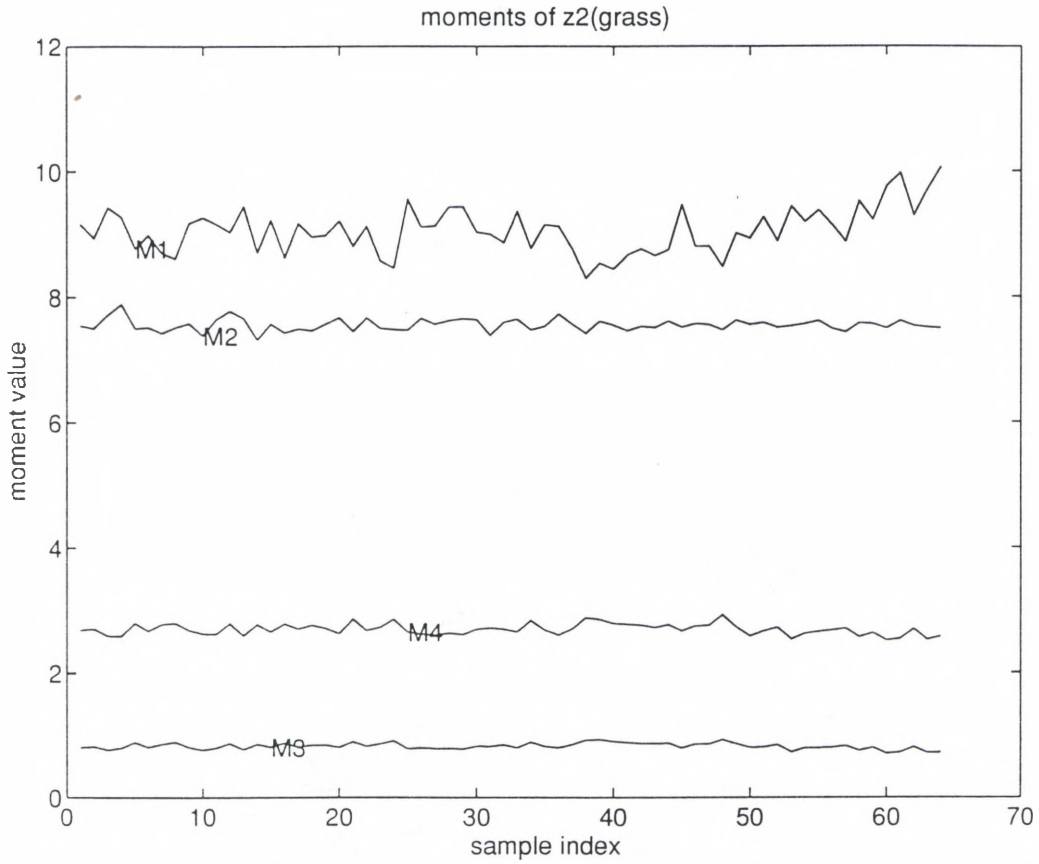

Figure B.1. Distribution of moments of $\mathbf{z_2}$ for the grass region.

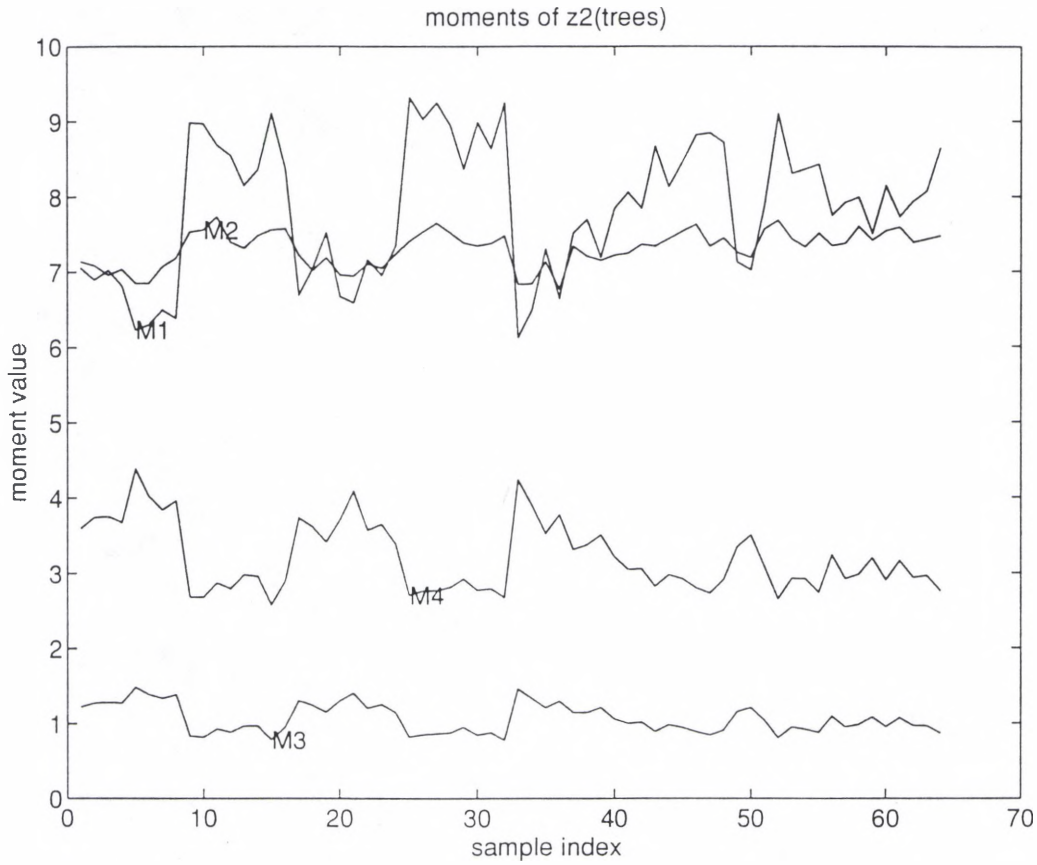

Figure B.2. Distribution of moments of $\mathbf{z}_2$ for the tree region.

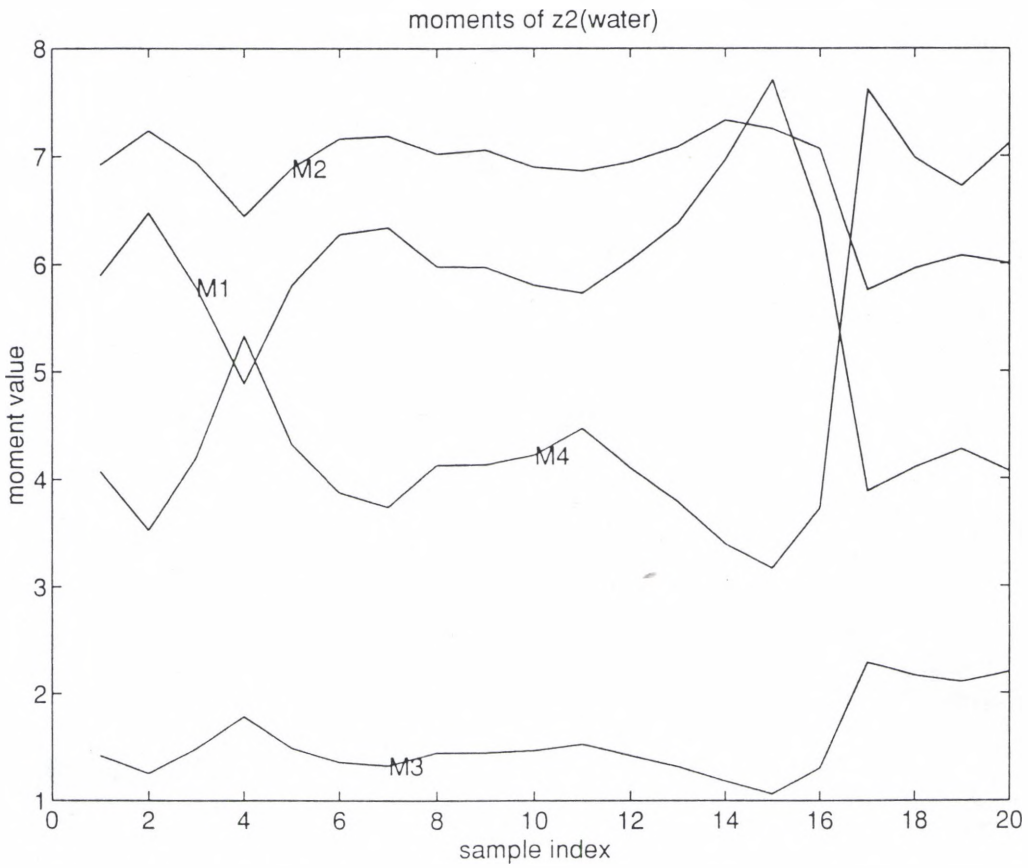

Figure B.3. Distribution of moments of $\mathbf{z_2}$ for the water region.

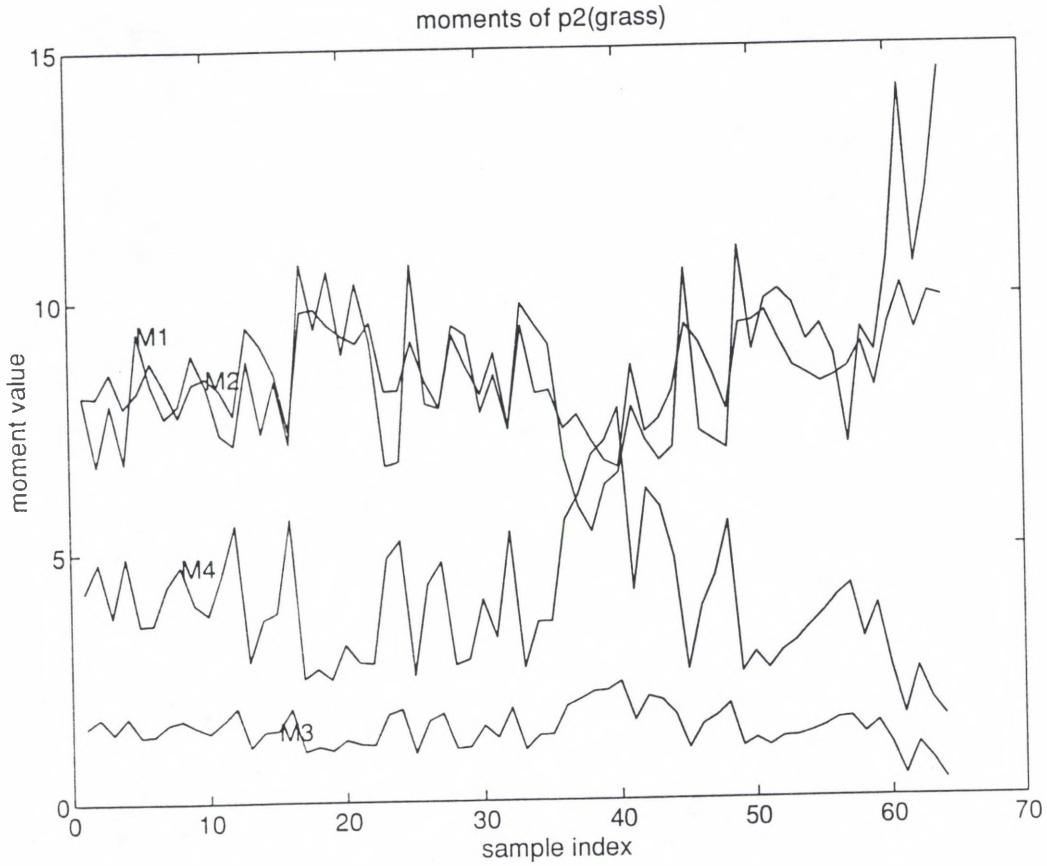

Figure B.4. Distribution of moments of $\mathbf{p_2}$ for the grass region.

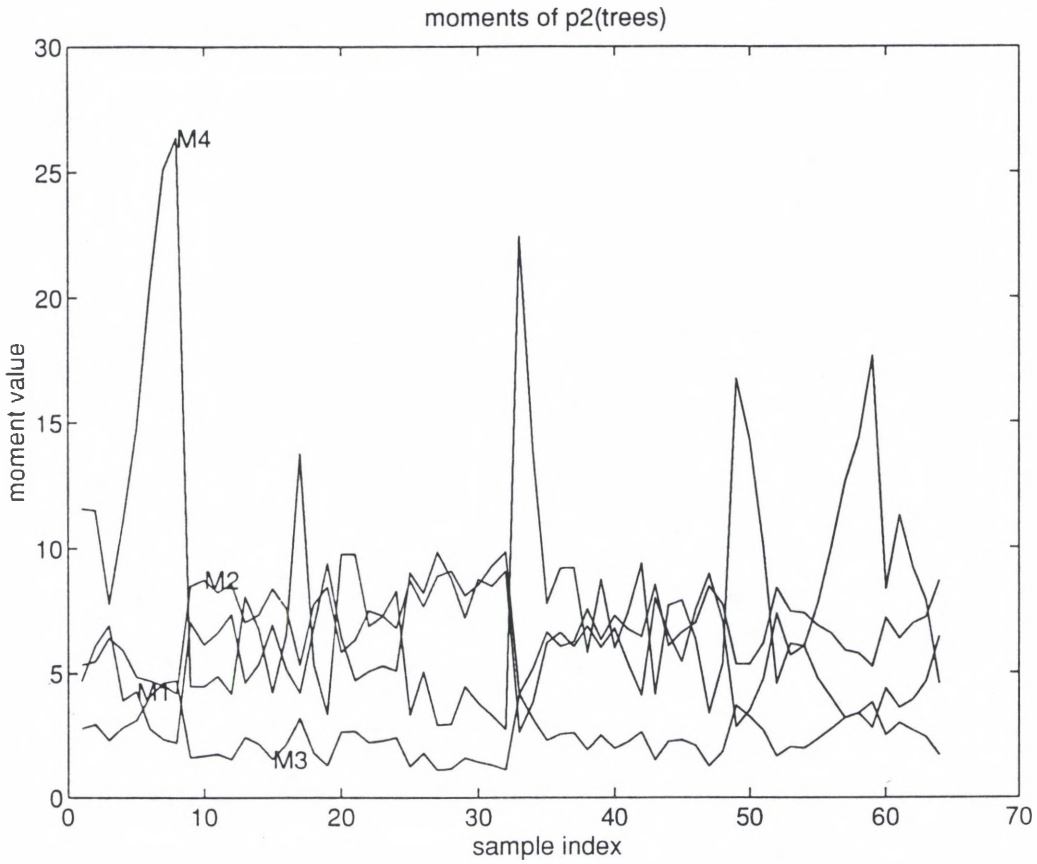

Figure B.5. Distribution of moments of $\mathbf{p_2}$ for the tree region.

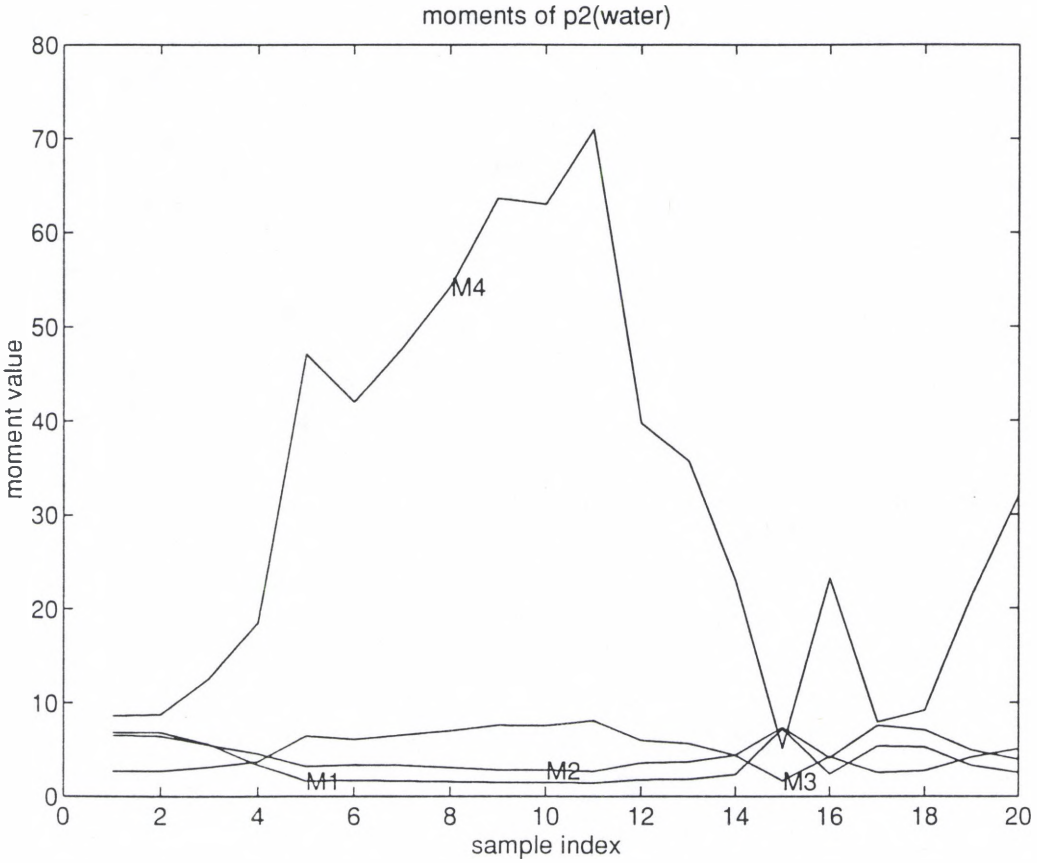

Figure B.6. Distribution of moments of $\mathbf{p_2}$ for the water region.

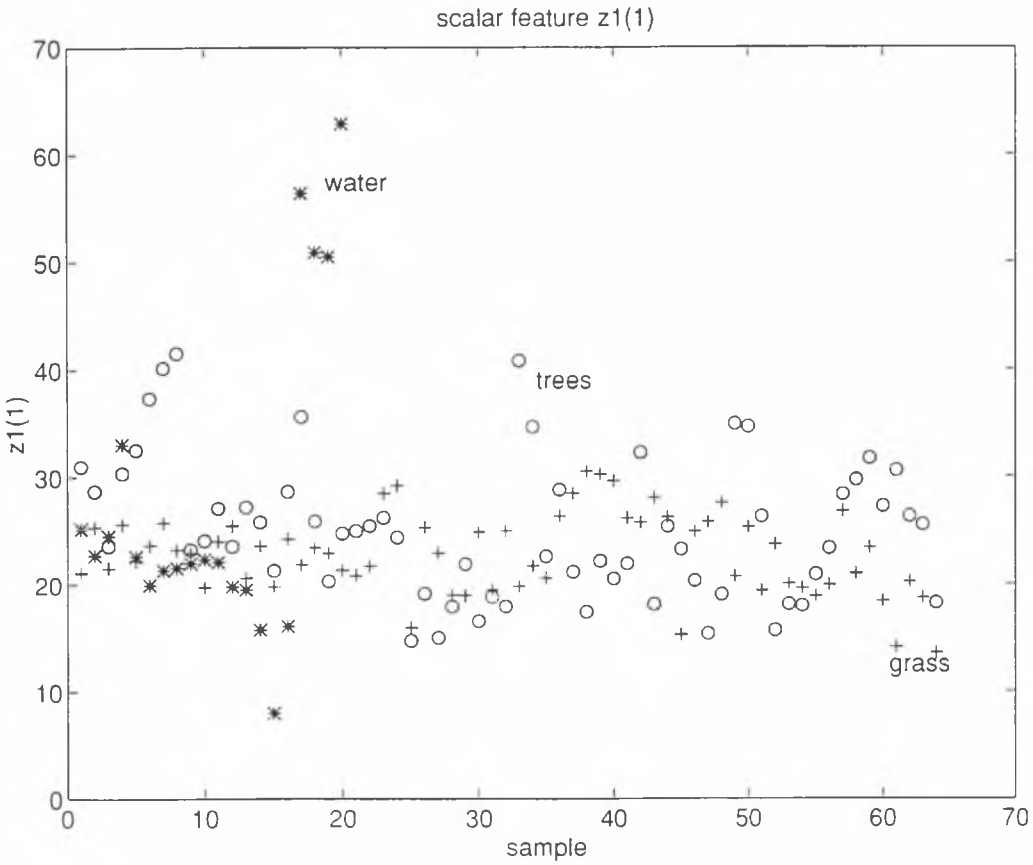

Figure B.7. Distribution of first singular value of $\mathbf{z}_1$ (scalar feature z_{10}) for grass, tree, and water regions.

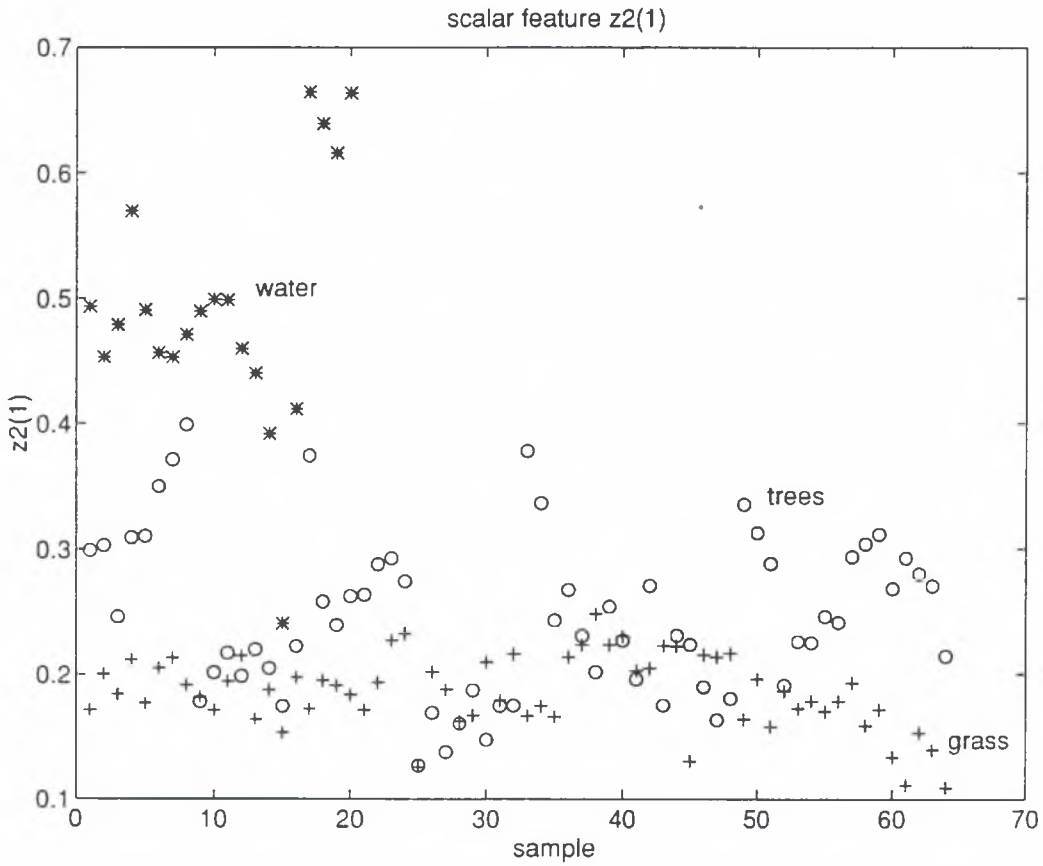

Figure B.8. Distribution of first singular value of $\mathbf{z_2}$ (scalar feature z_{10}) for grass, tree, and water regions.

REFERENCES

1. Haralick, R.M., K. Shanmugan, and I. Dinstein, "Texture Features for Image Classification," *IEEE Trans. Systems, Man, and Cybernetics*, **SMC-3**, November 1973, pp. 610-621.

2. Weszka, J.S., C. Dyer and A. Rosenfeld, "A Comparative Study of Texture Measures for Terrain Classification," *IEEE Trans. Systems, Man, and Cybernetics*, **SMC-6**, April 1976, pp. 269-285.

3. Goodman, J.W., *Introduction to Fourier Optics*, McGraw Hill, 1968.

4. Karim, M.A. and A.A.S. Awwal, *Optical Computing: An Introduction*, Wiley-Interscience, New York, 1992

5. Casasent D., "Optical Feature Extraction," *Optical Signal Processing*, Edited by J. Horner, Academic Press, San Diego, 1987, pp. 75-95.

6. Casasent, D., "Acoustooptic Linear Algebra Processors: Architectures, Algorithms, and Applications," *Proceedings of the IEEE*, **Vol. 72, No. 7**, July 1984, pp. 831-849.

7. Nilsson, N., "Adaptive Pattern Recognition: a survey." *1966 Bionics Symposium*, Dayton, OH, May 1966.

8. Duda, R., "Graphical-data-processing Research Study and Experimental Investigation," *Stanford Research Institute*, Menlo Park, CA, **Tech Report ECOM-01901-26**, March 1967.

9. Hu, M.K., "Visual Pattern Recognition by Moment Invariants," *IEEE Trans. Information Theory*, **IT-8**, 2 February 1962, pp. 179-187.

10. Pentland, A.P., "Fractal-based Description of Natural Scenes,", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **Vol. PAMI-6, No. 6**, November 1984, pp. 661-674.

11. Ashjari, B., "Stochastic Singular Value Decomposition Texture Measurement for Image Classification," Ph.D. Dissertation, University of Southern California, February 1982.

12. Pratt, W.K., *Digital Image Processing (Second Edition)*, Wiley-Interscience, New York, 1991.

13. Young, T.Y. and King-Sun Fu, *Handbook of Pattern Recognition and Image Processing*, Academic Press, New York, 1986.

14. Marinovic, N.M. and G. Eichmann, "Feature Extraction and Pattern Classification in Space-spatial Frequency Domain," *Proceedings SPIE*, **579**, 1985, pp. 19-26.

15. Sullivan, B.J. and B. Liu, "On the use of SVD and Decimation in Discrete-time band-limited signal extrapolation," *IEEE Trans. Acoustics Speech Signal Processing*, **ASSP-32**, **No. 6**, 1984.

16. Klema, V.C., "The SVD: its computation and some applications," *IEEE Trans. Automatic Control*, **AC-25**, 1980, pp. 164-176.

17. Shu-Qiu, L., "Using SVD extraction of faint electrocardiogram signal of foetus on the background of strong noise signal," *J. Data Acquisition Process* (in Chinese) **4**, 1989, pp. 12-14.

18. Hong, Zi-Quan, "Algebraic Feature Extraction of Image for Recognition," *Pattern Recognition*, **Vol. 24**, **No. 3**, 1991, pp. 211-219.

19. *MATLAB*, **The Math Works Inc.**, Prentice-Hall, New Jersey, 1992.

20. Dongarra, J.R., *LINPACK User's Guide*, **SIAM**, Philadelphia, 1979.

21. Gorecki, C., "Optical Classification of Metal Milled Samples using Fourier Spectrum Sampling," *SPIE, Industrial Inspection II,* **Vol. 1265** , 1990, pp. 206-213.

22. Stark, H. and R. O'Toole, "Statistical Pattern Recognition using Optical Fourier Transform Features," *Application of Optical Fourier Transforms*, Edited by H. Stark, Academic Press, London, 1982, pp. 465-497.

23. Duda, R.O. and P.E. Hart, *Pattern Classification and Scene Analysis*, Wiley-Interscience, New York, 1973.

24. Fukunaga, K., *Introduction to Statistical Pattern Recognition (Second Edition)*, Academic Press, New York, 1990.

25. Hall, E.L., *Computer Image Processing and Recognition*, Academic Press, New York, 1979.

26. Bow, Sing-Tze, *Pattern Recognition and Image Processing*, Marcel Dekker, New York, 1992.

27. Jain, A., *Fundamentals of Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1989.

28. Tou, J. and R. Gonzalez, *Pattern Recognition Principles*, Addison-Wesley, Reading, Massachusetts, 1981.

29. Banks, S., *Signal Processing: Image Processing and Pattern Recognition*, Prentice Hall, London, 1990.

30. Tamura, H., S. Mori and T. Yamawaki, "Texture features corresponding to Visual Perception," *IEEE Trans. on Systems, Man, and Cybernetics*, **SMC-8 No. 6**, 1978, pp. 460-473.

31. Bajcsy, R., "Computer Description of Textured Surfaces," *Proc. 3rd Int. Joint Conf. Int.*, August 1973, pp. 572-579.

32. Weszka, J. and C. Dyer, and A. Rosenfeld, "A Comparative Study of Texture Measures for Terrain Classification," *IEEE Trans. Systems, Man, and Cybernetics*, **SMC-6, No. 4**, 1976, pp. 269-285.

33. Zucker, S. and D. Terzopoulos, "Finding Structure in co-occurrence Matrices for Texture Analysis," *Image Modeling*, edited by A. Rosenfeld, Academic Press, New York, 1981.

34. Lendaris, G. and G. Stanley, "Diffraction Pattern Samplings for Automatic Pattern Recognition," *Proceedings of the IEEE*, **Vol. 58**, 1970, pp. 198-216.

35. Gramenopoulas, N., "Terrain type Recognition using ERTS-1 MSS Images," *Rec. Symp. Significant Results Obtained from the Earth Resource Technology Satellite*, **NASA SP-327**, March 1973, pp. 1229-1241.

36. Horning, R. and J. Smith, "Application of Fourier Analysis to Multispectral/Spatial Recognition," *Management and Utilization of Remote Sensing Data ASP Symposium*, Sioux Falls, SD, October 1973.

37. Bajcsy, R., "Computer Identification of Visual Surfaces," *Computer Graphics and Image Processing*, **Vol. 2**, 1973, pp. 118-130.

38. Bajcsy, R. and L. Lieberman, "Computer Description of Real Outdoor Scenes," *Proc. 2nd International Joint Conference on Pattern Recognition*, Copenhagen, Denmark, August 1974, pp. 174-179.

39. Dyer, C. and A. Rosenfeld, "Fourier Texture Features: Suppression of aperture Effects," *IEEE Trans. on Systems, Man, and Cybernetics*, **SMC-6**, 1976, pp. 703-705.

40. D'Astous, F. and M. Jernigan, "Texture Discriminant based on Detailed Measures of the Power Spectrum," *7th International Conference on Pattern Recognition, Montreal,* July 30-August 2, 1984, pp. 83-86.

41. Kirvida, L., "Texture Measurements for the Automatic Classification of Imagery," *IEEE Trans. on Electromagnetic Compatibility*, **EMC-18**, February 1976, pp. 38-42.

42. Golub, G. and C. Van Loan, *Matrix Computations (Second Edition)*, The Johns Hopkins University Press, Baltimore, 1989.

43. Klema, V. and A. Laub, "The Singular Value Decomposition: Its Computation and Some Applications," *IEEE Trans. on Automatic Control*, **AC-25, No. 2**, April 1980, pp. 164-176.

44. Brodatz, P., *Texture: A Photograph Album for Artists and Designers*, Dover, New York, 1956.

45. Faugeras, O. and W. Pratt, "Decorrelation Methods of Texture Feature Extraction," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **PAMI-2, No. 4**, July 1980, pp. 323-332.

46. Cheng, Y.Q., Y. Wu, R. Jiang, K. Liu, and J. Yang, "Aircraft Identification Based on the Algebraic Method," *Hybrid Image and Signal Processing III*, **SPIE Vol. 1702**, 1992, pp. 298-305.

47. Kumar, B., "Singular Value Decomposition using Iterative Optical Processors," *Applied Optics*, **Vol. 22, No. 7**, 1 April 1983, pp. 962-963.

48. Ruck, D., "Characterization of Multilayer Perceptrons and their Application to Multisensor Automatic Target Recognition," Ph.D. Dissertation, Air Force Institute of Technology, 1990.

49. Hawkins, J.K., "Textural Properties for Pattern Recognition," in *Picture Processing and Psychopictorics*, Academic Press, New York, 1970.

50. Vaccaro, R., *SVD and Signal Processing, II, Algorithms, Analysis and Applications*, Elsevier Science, Amsterdam, 1991.

51. Duda, R.O., "Image Data Extraction," unpublished notes, July 1975.

52. Bhattacharyya, A., "On a measure of divergence between two statistical populations defined by their probability distributions," *Bull. Calcutta Math. Soc.*, **35**, 1943, pp. 99-110.

53. Mahalanobis, P., "Analysis of Race mixture in Bengal," *J. Asiat. Soc. (India)*, **Vol. 23**, 1925, pp. 301-310.

54. Fisher, R., *Contributions to Mathematical Statistics*, New York, Wiley, 1950.

55. Kailath, T., "The Divergence and Bhattacharyya Distance Measures in Signal Selection," *IEEE Trans. on Communication Technology*, **COM-15, No. 1**, February 1967, pp. 52-60.

56. Neter, J., W. Wasserman and G. Whitmore, *Applied Statistics (Second Edition)*, Allyn and Bacon, Inc., Boston, 1982.

57. *GTVISIT User's Manual*, Electro-optics Laboratory, Georgia Tech Research Institute, Atlanta, 1991.