Interactive Visualisation Techniques for Data Mining of Satellite Imagery

By

Samuel John Welch, BComp

A dissertation submitted to the

School of Computing

In partial fulfilment of the requirements for the degree of

Bachelor of Computing with Honours

University of Tasmania

November 2006

Declaration

I, Samuel John Welch declare that this thesis contains no material which has been accepted for the award of any other degree or diploma in any tertiary institution. To my knowledge and belief, this thesis contains no material previously published or written by another person except where due reference is made in the text of the thesis.

Samuel John Welch

Abstract

Supervised classification of satellite imagery largely removes the user from the information extraction process. Visualisation is an often ignored means by which users may interactively explore the complex patterns and relationships in satellite imagery. Classification can be considered a “hypothesis testing” form of analysis. Visual Data Mining allows for dynamic hypothesis generation, testing and revision based on a human user’s perception. In this study Visual Data Mining was applied to the classification of satellite imagery.

After reviewing appropriate techniques and literature a tool was developed for the visual exploration and mining of satellite image data. This tool augments existing semi-automatic data mining techniques with visualisation capabilities. The tool was developed in IDL as an extension to ENVI, a popular remote sensing package.

The tool developed was used to conduct a visual data mining analysis of high-resolution imagery of Heard Island. This process demonstrated the positive impacts of visualisation and visual data mining when used in the analysis of satellite imagery. These impacts consist of: increased opportunity for understanding and hence confidence in classification results, increased opportunity for the discovery of subtle patterns in satellite imagery and the ability to create, test and revise hypotheses based on visual assessment.

Acknowledgements

Firstly, I would like to thank my supervisors, Dr. Ray Williams and Dr. Arko Lucieer. Ray and Arko’s feedback and guidance this year helped immensely. Arko’s work in preparing the Heard Island data for Humphrey and I was very much appreciated. Thanks go to both Arko and Ray for weathering my storm of last minute re-writes.

Thanks are due to my fellow honours room residents, Luke, Tristan, Matt, Jaidev, and Mathew, and to the rest of the Hobart honours crew. Some humour and a few games made for an enjoyable work environment. Thanks also to Humphrey in Launceston. It’s been good working with you this year and it’s been interesting to see how our two projects developed independently and yet, in the end, each has much to contribute to the other.

Many thanks are also due to my parents. Mum and Dad, your support and encouragement have been greatly appreciated.

Finally, my thanks go to my girlfriend, Jessica. Thank you for helping me make it through this year. Your comforting support kept me sane and helped me through this stressful time.

Table of Contents

List of Figures

List of Tables

List of Equations

Chapter 1 Introduction

1.1 Problem Description

Satellite imagery is increasingly being used to monitor the surface of the Earth. There are numerous satellites orbiting the Earth equipped with powerful instruments for remote observation of entities and events on the ground. The data collected by these instruments is used for various purposes, such as monitoring of climate change and resource mapping. The science and art of acquiring and interpreting these images is known as remote sensing (Lillesand & Kiefer 2000; Richards 1986; Schowengerdt 1997). The extraction of spatial features is usually carried out by image interpretation and/or quantitative analysis (Richards 1986, p. 69). Image interpretation (also known as photointerpretation) requires a human analyst to visually inspect the imagery and extract information based on their experience and expert knowledge. This process can become extremely tedious due to the manner in which the data is presented and even the most expert analysts occasionally make avoidable oversights. Quantitative analysis aims to automate the interpretation stage and is generally favoured over manual image interpretation.

Classification is a quantitative analysis technique in which each pixel is assigned a thematic class based on its spectral values. This process involves statistical comparison of each pixel’s properties to that of a reference class. The problem with classification and other quantitative methods is that the user is removed from the interpretation process. This results in decreased confidence in, and understanding of, the results of such analyses (Keim, D A 2002).

Visual Data Mining (VDM) (Soukup & Davidson 2002) involves the user in the classification process. Visualisation and VDM can play an important role in exploring patterns and relationships in satellite imagery. Visualisation can also reveal the functioning of classification algorithms (Thearling et al. 2001). Most, if not all, commercial remote sensing software packages are lacking tools for VDM and explorative visual analysis. The provision of visualisation and visual data mining tools in a remote sensing software package would allow the exploration of the use of these technologies for analysis of satellite imagery.

1.2 Research Objectives

1.2.1 Hypothesis

It is hypothesised that combining interactive visualisation capabilities with existing techniques for data mining of remotely sensed imagery can enhance understanding of the image classification process, reveal trends in the data and produce more insightful image analyses.

1.2.2 Aim

The aim of this study is to test the hypothesis by:

- Designing and implementing a prototype visualisation system for data mining of remotely sensed imagery, and
- Applying the visualisation prototype to a classification study based on high-resolution satellite imagery of Heard Island to showcase and assess the value of visualisation in semi-automated image classification.

Chapter 2 Background

2.1 Remote Sensing

2.1.1 Introduction

Remote sensing is broadly defined as the acquisition of *information* regarding a particular entity or event without requiring the data acquisition device to be in close proximity to the entity or event (Lillesand & Kiefer 2000). Most commonly the term refers to the use of satellite and aircraft mounted sensors for mapping and monitoring of the Earth's surface. Numerous remote sensing satellites are orbiting the Earth. Each satellite is referred to as a *platform* and is home to one or more *instruments* or *sensors*. Multiple satellites with identical sensors are sometimes referred to as a system, eg: the Landsat system (Richards 1986).

Remote sensing is more than remotely obtaining data regarding the Earth's surface, eg: taking photographs from space. It is the science of drawing useful conclusions from the raw data, or put simply, extracting *information*. This process of extracting information is known as *interpretation*. There are two approaches to interpretation: image interpretation and quantitative analysis. These are discussed in section 2.1.4.

2.1.2 Remotely Sensed Data

Remotely sensed data is usually available in a digital format suitable for computer processing. The data is composed of discrete picture elements or *pixels*. A 2-dimensional (2D) array of pixels makes up an *image*. Each of these pixels contains one or more data components (one per *band*) relating to one or more wavelengths in the electromagnetic spectrum. This is an important characteristic of remotely sensed imagery. A remotely sensed image comprises one or more *bands*. A band is a 2D array of values recording a measure of energy in some part of the electromagnetic spectrum. Bands may represent reflected visible light, infrared, microwave wavelengths or any other part of the spectrum. Bands in an image have equal sizes in each dimension and represent the same spatial area. By simultaneously indexing each band array with an (x,y) coordinate pair the component values for the pixel with coordinates (x,y) of the image can be determined.

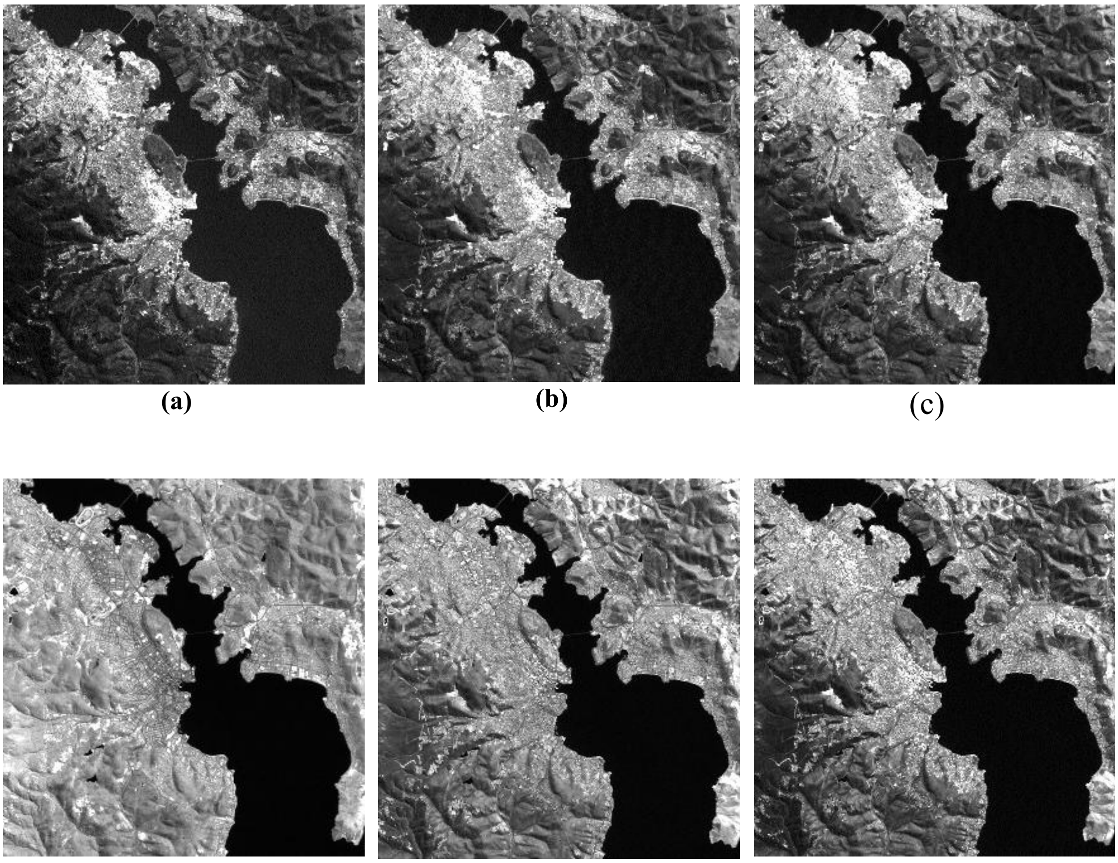
Properties of remotely sensed data that are commonly discussed are its spectral, spatial and radiometric resolutions. Spectral resolution refers to the number and range of the bands in the image. Spatial resolution refers to the size, in ground units (square kilometres or metres), that each pixel represents. Finally, radiometric resolution refers to the number of discrete brightness levels available for components of each band. This is usually stated in terms of the number of binary digits used to represent the range of values.

For example, the Landsat Thematic Mapper (TM) instrument measures 6[1] distinct wavelength ranges; blue (0.45-0.52 μm), green (0.52-0.60 μm), and red (0.63-0.69 μm) visible light bands, a near infrared (0.76-0.90 μm) and two middle infrared (1.55-1.75 μm and 2.08-2.35 μm) bands (Richards 1986). Images from this instrument have 6 bands, one for each wavelength range. Each of these bands is recorded with a spatial resolution of 30 m; that is, each pixel represents a 30x30 m square on the ground. Each band is recorded with a radiometric resolution of 8 bits which means each element of each band can only be one of 256 discrete brightness levels.

Bands for different wavelengths within an image are illustrated in Figure 2-1. The figures shown are Landsat TM images of Hobart, Tasmania, Australia acquired on 28/9/1999. Each band is displayed independently of the others in a grey scale image composition of the data for that band. The brighter any pixel in the image is, the higher its reflectance in that particular band.

---

[1] The Landsat Thematic Mapper does in fact have a seventh band representing thermal energy with a spatial resolution of 120m which has been excluded for the sake of simplicity



(a) **(b)** (c)

(d) **(e)** (f)

Figure 2-1: Reflectance in each of the Landat TM satellites 6 bands: (a), (b), (c) Blue, green and red visible light, (d) near infrared, (e) middle infrared 1, (d) middle infrared 2.

Bands can be combined and viewed as a (false) colour composite. In this process three bands are selected and allocated to represent the red, green and blue components of pixels on the display. By allocating bands 3, 2 and 1 of the Landsat TM image in Figure 2-1cba to represent red, green and blue colour components, the image in Figure 2-2 is composed. This is known as a colour composite. A false colour composite is one in which higher wavelength bands are used or visible light bands are assigned out of order. Figure 2-3 shows a false colour composite of the Hobart image where red represents near infrared, green represents blue visible light and blue represents middle infrared. False colour composite images enable visual identification of spectrally distinct land cover features.

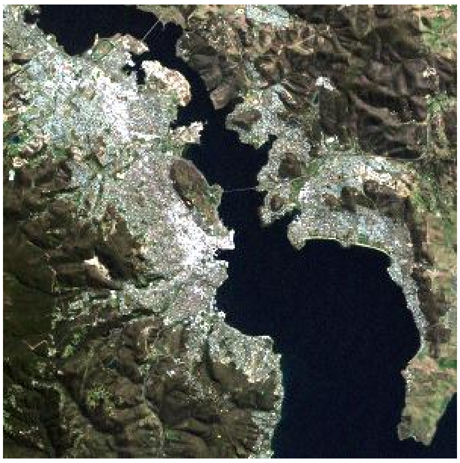


Figure 2-2: Landat TM image of Hobart, Tasmania in visible light (bands 3, 2 and 1).

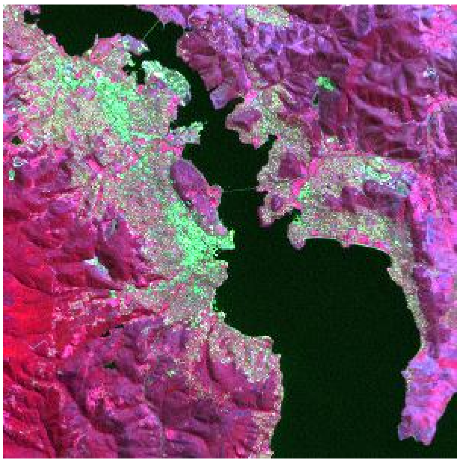


Figure 2-3: False colour composite using bands 4, 1 and 5 of the Landsat TM image of Hobart.

2.1.3 Pre-processing

Prior to interpretation, remotely sensed imagery is often subjected to rectification and/or enhancement. Rectification of imagery is important for obtaining accurate results. Radiometric correction is used to counter the effects of sensor noise and atmospheric refraction. Geometric correction is used rectify the geometric distortions cause by the curvature of the Earth, the angle of acquisition and topographic relied distortion. A summary of rectification techniques can be found in (Lillesand & Kiefer 2000; Richards 1986; Schowengerdt 1997). Enhancement refers to the use of image filters to stretch, smooth and sharpen imagery for visual analysis (Gonzalez & Woods 1983).

2.1.4 Interpretation

Interpretation is the key stage in the process of remote sensing. Up until this point no actual information has been gained. This section presents the two exclusive yet complementary, traditional approaches to interpretation of remotely sensed data and introduces the use of a third non-traditional approach.

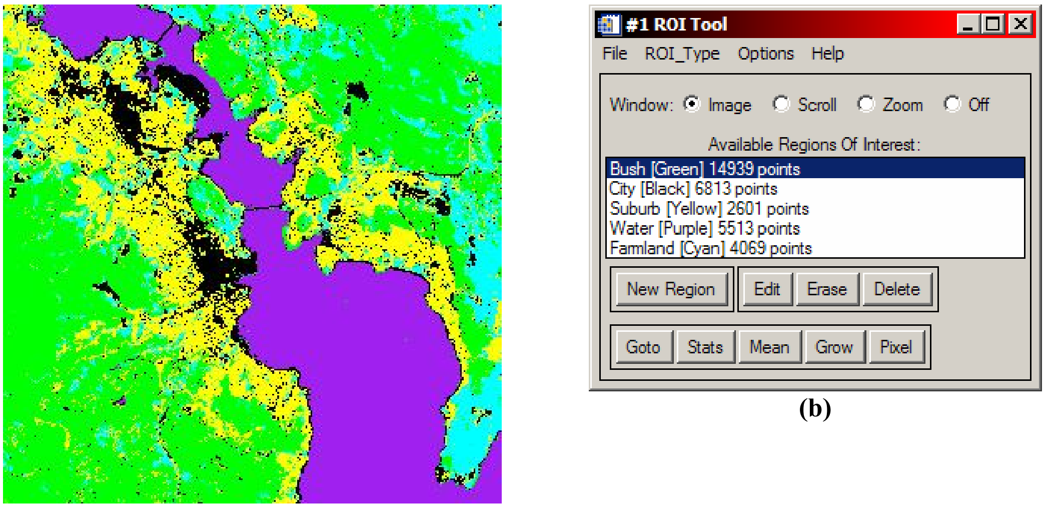
Traditional image interpretation (sometimes referred to as photointerpretation) involves a human analyst/interpreter extracting information by visual inspection. The data is usually displayed as a false colour composite and the analyst makes decisions about what information is contained in the image based on their experience and expertise. This visual inspection based approach is usually quite effective in the process of segmentation. Segmentation involves extracting objects (road, buildings, lakes, rivers, etc) from the image and displaying their extent in the image. Image interpretation is not suited to the task of accurately determining the area or extent of land cover classes. Not only is it tedious for an analyst to examine every pixel in an image (as would be needed to obtain an accurate area estimate) but it is difficult for them to take into account the full reflectance profile of each pixel as only three bands are being displayed at once. Multiple false colour composites can be used to somewhat alleviate the latter issue but the problem still exists.

Whilst it is impractical for a human analyst to examine every pixel in an image, and when doing so to take into account the full dimensionality of (i.e., the number of bands in) the image the task is well suited to a computational approach. A quantitative analysis may consist of *classifying* each pixel in the image. Classification is a method by which labels may be attached to pixels according to their spectral characteristics (Richards 1986). This is discussed in Section 2.3. Having obtained a classification for each pixel, a colour coded *thematic map* of the image can be constructed (Figure 2-4). From this map the area of coverage for a given land cover class can be extracted, as well as many other measurements.

Quantitative analysis more readily yields the information that remote sensing analysts require. However, interpretation can take another key form; Visual Data Mining (VDM) or more generally scientific visualisation. This is essentially the middle ground between quantitative analysis and image interpretation. Visual data mining presents the user with a more appropriate representation of their data so that they can visually explore the relationships and patterns it contains. Like image interpretation VDM is highly subjective. This does not detract from its usefulness in remote sensing. Indeed Lillesand & Kiefer highlight the artistic nature of remote sensing:

> Remote Sensing is the science and art of obtaining information about an object, area, or phenomenon through the analysis of data acquired by a device that is not in contact with the object, area, or phenomenon under investigation. (Lillesand & Kiefer 2000)

Visualisation was used in combination with remote sensing successfully by Lucieer (2004). Visualisation and Visual Data Mining are discussed extensively in section 2.6. Its use in remote sensing is discussed in section 2.7.



Bush [Green] 14939 points
City [Black] 6813 points
Suburb [Yellow] 2601 points
Water [Purple] 5513 points
Farmland [Cyan] 4069 points

(a)

(b)

Figure 2-4: (a) Thematic map of the Landsat TM Hobart image, (b) Land cover classes by colour

2.2 Data Mining

2.2.1 Introduction

Witten & Frank (2005) define data mining as follows:

> Data mining is defined as the process of discovering patterns in data. The process must be automatic or (more usually) semiautomatic. The patterns discovered must be meaningful in that they lead to some advantage, usually an economic advantage. The data is invariably present in substantial quantities.

Machine Learning (ML) is the common set of methods or algorithms applied to computationally extract these patterns. ML techniques require data to be represented in an attribute-value style format. An example ML dataset is given in Table 2-1. The dataset describes the properties of various specimens of Iris plants. In this dataset there are four *instances* – one for each specimen –, and four *attributes*. Each instance has a *value* for each attribute as well as a *class label* indicating the species of Iris to which an expert has decided it belongs.

Table 2-1: Sample from the Iris Plants Database (Fisher 1988)

Instance No.	Sepal Length	Sepal Width	Petal Length	Petal Width	Class
1	5.1	3.5	1.4	0.2	Iris Setosa
2	4.6	3.1	1.5	0.2	Iris Setosa
3	6.1	2.8	4.7	1.2	Iris Versicolor
4	6.3	2.9	5.6	1.8	Iris Virginica

There exist many methods for machine learning; decision trees, classification rules, artificial neural networks and instance based methods (Witten & Frank 2005). These methods all perform the operation of classification. The result of this operation is the construction of a classifier. A classifier embodies the knowledge extracted from the dataset and can be used to classify new unclassified instances. For example, if a new specimen of Iris needs to be classified, the values of the specimen's attributes (sepal length/width, petal length/width) are given to the classifier and a class label is returned.

In the case of satellite imagery, instances are pixels. The attributes for each pixel are the bands in the image. The values are the reflectance values in each band. Class labels are assigned by an analyst to a small set of pixels in the image known as the training pixels. A classifier can be constructed and used to classify the rest of the pixels in the image.

ML methods can be split in three ways: supervised versus unsupervised, parametric versus non-parametric and hard classifying versus soft classifying. Supervised ML describes cases in which the algorithm is given the class labels of its training instances. Unsupervised ML describes cases in which the algorithm clusters the instances and presents the unlabelled clusters to the user for interpretation. The user then assigns, based on their perception, the most appropriate label to the cluster (Witten & Frank 2005, p. 43). Parametric classifiers are those that summarise their training data statistically according to certain assumptions regarding the distribution of the class and make classification decisions based on those statistics and assumptions. Non-parametric classifiers make no assumptions regarding the statistical distribution for each class. Hard classifiers provide a single class label for a new instance. Soft classifiers may provide multiple, weighted classifications for a new instance.

Specific classifiers are described in section 2.3. The descriptions refer specifically to remotely sensed data rather than the general case. Many remote sensing authors tend to discuss these techniques as if they exist purely for the field of remote sensing; this of course is not true, they may be applied to virtually any classification problem.

2.2.2 Feature Space

Feature space is an abstract *n*-dimensional space representing the classification problem at hand. Each instance in a dataset can be plotted as a point in this space. The value of *n* is determined by the number of attributes in the dataset. The dimensionality can be reduced by simply ignoring certain attributes. If *n* is reduced to 2 or 3 the 2D or 3D feature space can be visualised.

It is useful when describing various classification algorithms to use some form of visual aid. A 3D feature space allows visualisation of the training data clusters and/or the decision boundaries used by the classifier. A 3D feature space plot can be constructed in the following manner: select three attributes from the dataset to assign to the axes of 3D Euclidean space, each instance can then be plotted according to the values of these three selected attributes. Likewise, a 2D feature space plot uses two attributes from the data set. In Figure 2-5 each instance from a sample dataset is plotted as a single point. Points are usually grouped into clusters based on class. Shapes can then be used to summarise the points in each cluster (Figure 2-6). Shapes and points allow visual explanation of classifier parameters.

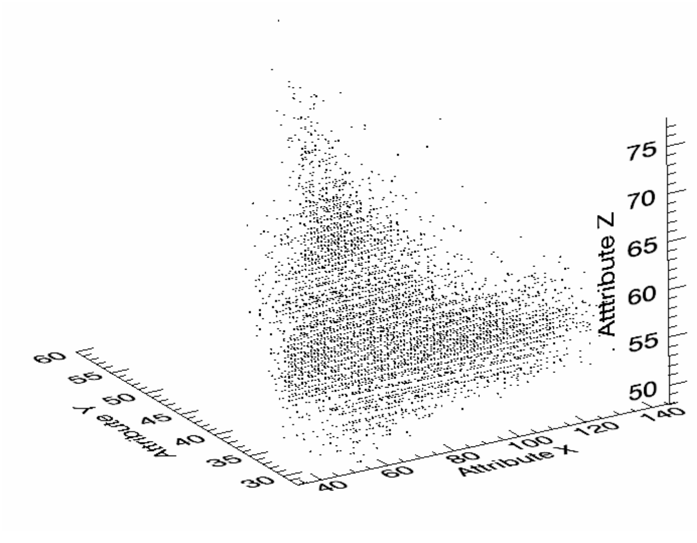
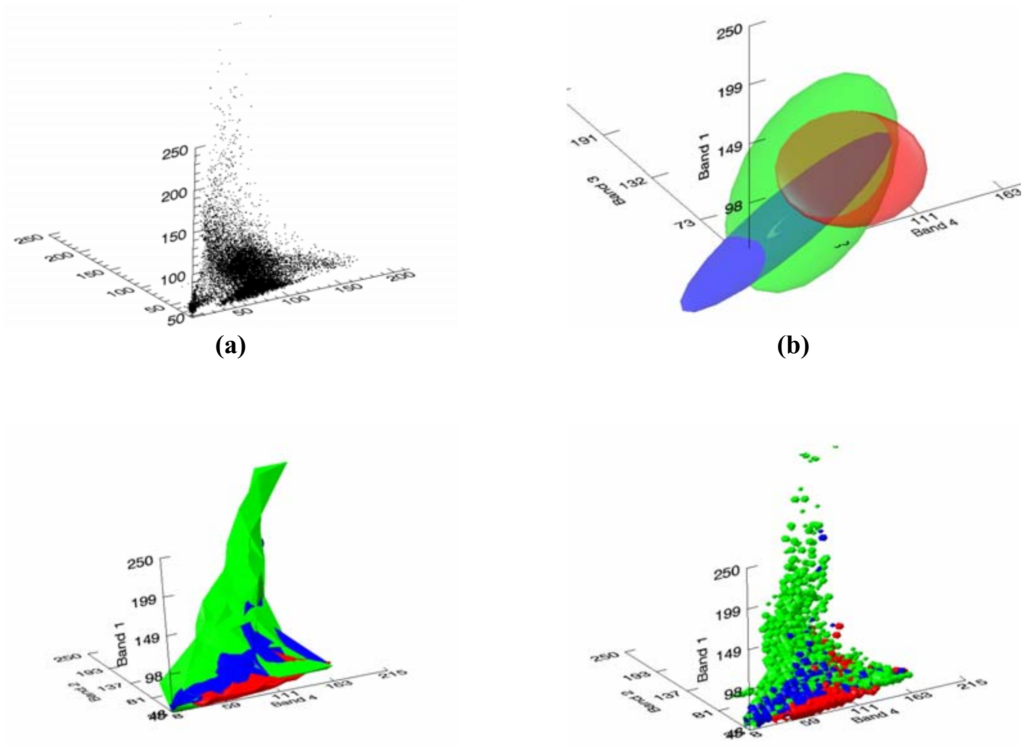


Figure 2-5: A sample feature space plot.



(a) (b)

(c) (d)

Figure 2-6: Feature space plots; (a) Greyscale scatter plot of a 3 class dataset, (b) Ellipsoids statistically representing the 3 classes, (c) α-shapes representing the extent of classes, (d) isosurfaces representing the extent and density of classes.

2.3 Supervised Remote Sensing Image Classification

2.3.1 Introduction

Supervised classification is the most commonly applied quantitative analysis technique for remotely sensed imagery. Image classification is the process of labelling pixels in the image as representing particular land cover types, or classes based on reference or training samples. Many algorithms exist for performing this classification. Regardless of the algorithm used, the process of supervised image classification is as follows:

1. Determine and specify the classes into which the image is to be classified.

2. For each class a training set of pixels must be defined. These pixels are deemed to be representative of the class. The location of these pixels can be derived in many ways: the user may perform a subjective photointerpretation and make a corresponding manual selection of pixels, or the pixels may be

selected based on field data (ground truth). This second option is often preferred as it removes any subjectivity from the classification.

3. The classifier is trained on these training pixels. For parametric classifiers the data is summarised statistically, for some non-parametric classifiers the training data may simply be recorded verbatim.

4. The trained classifier is then used on every pixel for which classification is desired (training pixels are usually excluded from this process). Every pixel in the image will now have an associated class label.

5. A colour coded thematic map or tabular summary can be produced.

This section presents a selection of classification algorithms popular in remote sensing applications. The most popular of these algorithms are parametric. Their popularity is largely due to the fact that non-parametric classifiers (excluding the parallelepiped) tend to be computationally intensive. Remote sensing data sets are becoming larger because of increasing image resolutions causing longer run times for these algorithms.

2.3.2 Maximum Likelihood

The maximum likelihood classifier is historically the most commonly used supervised classifier in remote sensing (Richards 1986). This parametric classifier uses the covariance matrix of the training set combined with an assumption of a normal distribution for pixel values belonging to this class. In simple terms the maximum likelihood classifier works by generating a set of probabilities, one for each class. These probabilities are an estimate of how likely the pixel is to belong to each class, like a Bayesian classifier (Witten & Frank 2005). The maximum probability is selected and the pixel is assigned the corresonding class label. The probabilities are based on how close a pixel in feature space lies to the mean of any class, weighted according to the covariance matrix (as a measure of the class' spread). In other words the decision boundary for a maximum likelihood classifier is a parabola (in 1D feature space), a circle or ellipse (in 2D feature space) or an ellipsoid or hyper-ellipsoid (in 3+-dimensional feature space). The closer the pixel is to the centre of the ellipsoid, relative to the length of the radial line on which the pixel lies, the higher the likelihood. A pixel on the exact boundary of two decision surfaces has the same probability for each class.

Maximum likelihood classifications usually impose some probability threshold below which a pixel will be labelled as unclassified; this avoids the situation of selecting the highest from a very low set of likelihoods. In a five class problem where one class has probability of 0.201 and the others have probability 0.19975 the first will be selected and the relatively high probability of the other classes will be totally ignored. The reason for its popularity is the fact that, as a parametric classifier, it is fast to classify but still maintains some important properties of the training data through the inclusion of the covariance matrix.

2.3.3 Minimum Distance

Possibly one of the simplest parametric classifiers is the minimum distance to mean, also known as the nearest-mean classifier (Schowengerdt 1997). During training the mean vector of each class is calculated and stored. Classification then simply consists of finding the nearest mean for a pixel in feature space based on its Euclidean distance. The pixel is then assigned the class of the closest mean. Decision boundaries for the minimum distance classifier are lines or (hyper-) planes dividing feature space. Minimum distance classifications are sometimes performed with a maximum distance threshold; pixels outside this range of any mean vector will be labelled as unclassified.

2.3.4 Level-Slice and Parallelepiped

The parallelepiped classifier is a simple non-parametric classifier. Training consists of recording the upper and lower bounds of the training data’s histogram for each class, in other words the minimum and maximum values for each band for each class. Classification consists of checking if a pixel falls within the bounds of each class. In feature space this simply means checking if the pixel falls within a given parallelepiped, that is a rectangle or (hyper-) box. The parallelepiped classifier, like the minimum distance classifier, is extremely fast to execute. A very minimal number of computations need to be carried out. The parallelepiped classifier has several drawbacks. Unless a pixel falls in a parallelepiped it cannot be classified (if a pixel falls in more than one parallelepiped a random decision must be made between the two classes). By contrast the minimum distance and maximum likelihood classifiers can classify any pixel (unless thresholding is applied) (Richards 1986).

The level-slice classifier is a special case of the parallelepiped. In this instance the parallelepipeds are strictly aligned with axes of feature space, as opposed to the general algorithm in which the (hyper-) shapes are only restricted to having parallel opposing sides (hence they are parallelepipeds). This is illustrated in Schowengerdt (1983, p. 178).

2.3.5 Instance Based Learning

Instance based classifiers are essentially the pinnacle of non-parametric classification. Many of these classifiers simply record their data verbatim. One of these types of instance based classifiers is the k-nearest neighbour. During training every pixel in the training set is recorded. During classification the k nearest neighbours in feature space (based on the Euclidean distance) in the training set are found for the pixel to be classified. k is a user defined parameter. The class labels of these neighbours are inspected and the most frequently occurring class is assigned to the pixel. Instance based methods are not commonly applied in remote sensing as they are very computationally expensive. An unoptimised k-nearest neighbour classification algorithm has time complexity of $O(n^2)$. The k-nearest neighbour classifier can produce accurate results when used on remotely sensed data as shown by Murray (in-press). Decision boundaries for the k-nearest neighbour are complex and vary depending on the value of k.

2.3.6 Soft Classification

The classifiers described above all assign a hard class label during classification. Fuzzy classification employs fuzzy set theory to provide a classified pixel with variable degrees of membership to each class in the image. In this way the user is provided with some measure of certainty of the classification. Numerous authors have published on the use of supervised fuzzy classification in remote sensing: (Lucieer 2004; Schowengerdt 1997; Wang 1999b, 1999a; Zhang & Foody 2001). Fuzzy classification allows the exploration of uncertainty in classification. If a pixel has near equal fuzzy membership values to multiple classes, this represents a high degree of uncertainty of classification. This certainty level can be recorded for each pixel and visualised post-classification (Lucieer 2004)(see Section 2.7). The decision boundaries for the fuzzy classifier can be visualised as spheres (Lucieer 2004, p. 23) or ellipsoids.

2.4 Texture

Classification of remotely sensed imagery as discussed in section 2.3 was limited to the use of a single pixel's spectral properties (its value in each band) to classify the pixel. The use of *texture measures* allows the interpretation of spatial relationships between pixels in a local area in classification.

Texture is a difficult concept to define. Gonzales & Woods (1983) define texture as a 'descriptor [that] provides measures of properties such as smoothness, coarseness, and regularity.' Another way of thinking of texture is as an attribute that represents the spatial arrangement of the spectral values of the pixels in a region. We can quantise texture in a region by a variety of texture measures.

A popular approach to quantising texture is the grey-level co-occurrence matrix (GLCM) (Haralick, Shanmugan & Dinstein 1973). This approach examines how often grey level values co-occur within a user specified neighbourhood in a single image band. Of course this matrix is of no use in itself as a texture measure but it can be statistically summarised using a variety of measures. Common measures include:

Table 2-2 - Common texture measures derived from the co-occurrence matrix

Entropy	$\sum_{i=1}^{n}\sum_{j=1}^{n} -Ci_j \log C_{ij}$
Mean	$\sum_{i=1}^{n}\sum_{j=1}^{n} C_{ij}$
Dissimilarity	$\sum_{i=1}^{n}\sum_{j=1}^{n} Cij\lvert i-j\rvert$

where the matrix C represents the grey level co-occurrence matrix with i and j being column and row indices respectively. Murray (in-press) used GLCM measures in the classification of sub-Antarctic vegetation types.

Other well known methods for quantising texture are Fourier and wavelet transform based approaches. Numerous authors explain these transforms: (Castleman 1996;

Gonzalez & Woods 1983; Randen & Husøy 1999; Richards 1986; Wahl 1987). Randen & Husøy (1999) discuss and compare these and a variety of other texture based classification techniques. Texture has been shown to improve classification results (Haralick, Shanmugan & Dinstein 1973; Lucieer 2004; Murray in-press). Visualisation can be used be determine the effects of particular measures before they are used for classification.

2.5 Scientific Visualisation

2.5.1 Introduction

Scientific Visualisation can be defined as the process of, and field of research relating to, the representation of data graphically as a means of gaining a deeper understanding of a complex system. There are subtle (and much debated) differences between the field of Scientific Visualisation and the field of Information Visualisation. Whilst Scientific Visualisation is considered to be related to visualisation of "natural" or spatial data, Information Visualisation is commonly regarded as visualisation of non-inherently spatial data, for example email traffic flow or relational databases (Ferreira de Oliveira & Levkowitz 2003). Some Scientific Visualisation purists would argue that Information Visualisation is only about presenting known ideas while Scientific Visualisation can be used to gain new understanding of complex problems.

Scientific Visualisation can take many forms. Parallel coordinate plots, feature space plots, surface and contour plots, volume renderings and a myriad of other visualisations are commonly used. Dynamically linked views are often used to represent multi-dimensional data in different ways. The dynamic link between visualisation and the original data enables exploration of patterns and relationships.

This study makes frequent use of feature space plots. Chapter 3 describes the use of a dynamic link between feature space and geographic image space to visualise relationship Sections 2.5.3 and 3.5 describe techniques from the field of 3D computer graphics which can be employed to represent class clusters in 3D feature space.

The use of *interactive* scientific/information visualisation for visual analysis of data is known as Visual Data Mining (Section 2.6).

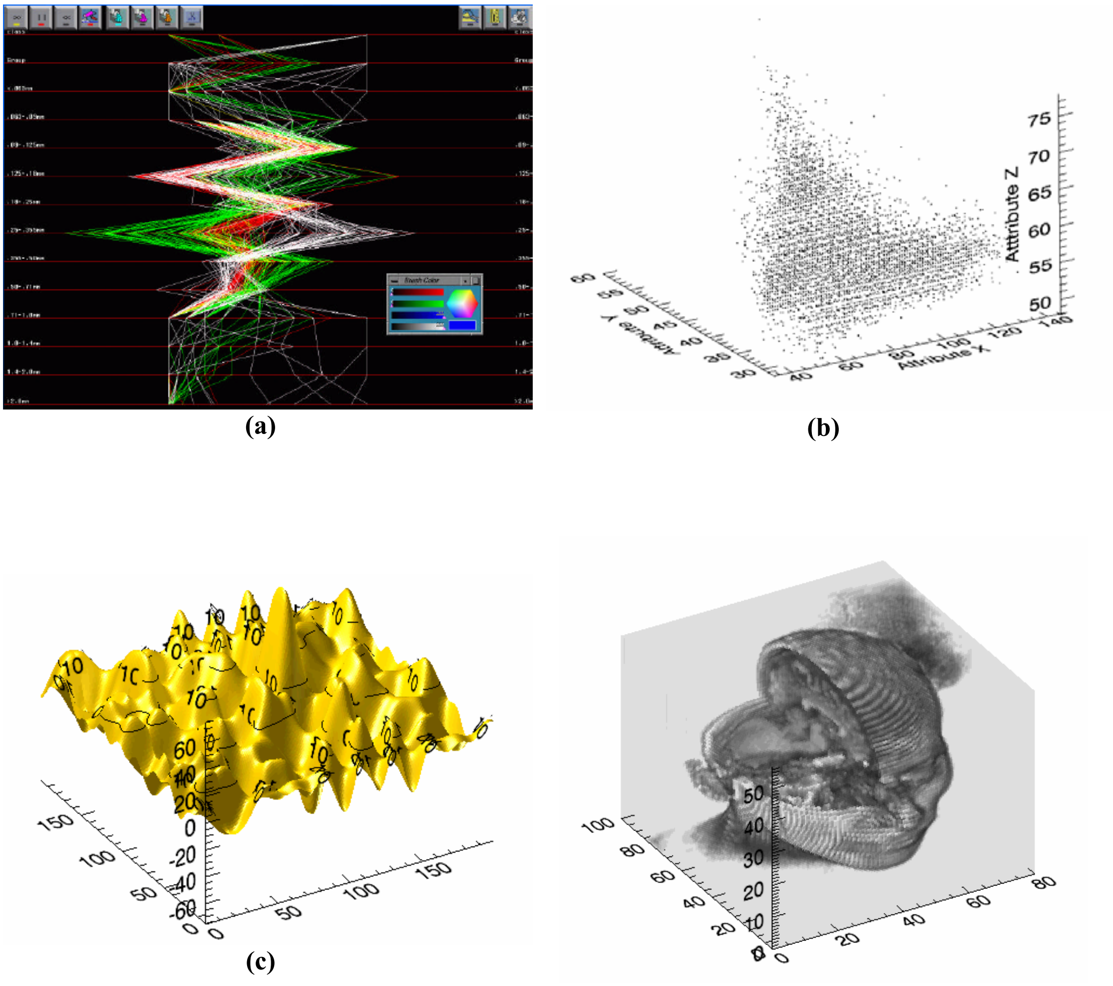


Figure 2-7: Examples of different scientific visualisations: (a) Parallel coordinate plot (Carr, Wegman & Luo), (b) feature space scatter plot, (c) surface and contour plot, (d) volume rendering.

2.5.2 Volumes

A 3D array of data is referred to as a volume. A volume is made up of volume elements or *voxels*. Each voxel is represented by a density value. Volume data is prevalent in fields such as medical imaging, engineering, archaeology and many others. Medical imaging techniques such as Magnetic Resonance Imaging (MRI) and Computed tomography (CT) produce volume data. For example an MRI machine can be used to examine tissue within a human head at small, regular intervals. The machine produces a set of measurements as 2D slices; these are then stacked together to form a 3D volume (Figure 2-8).

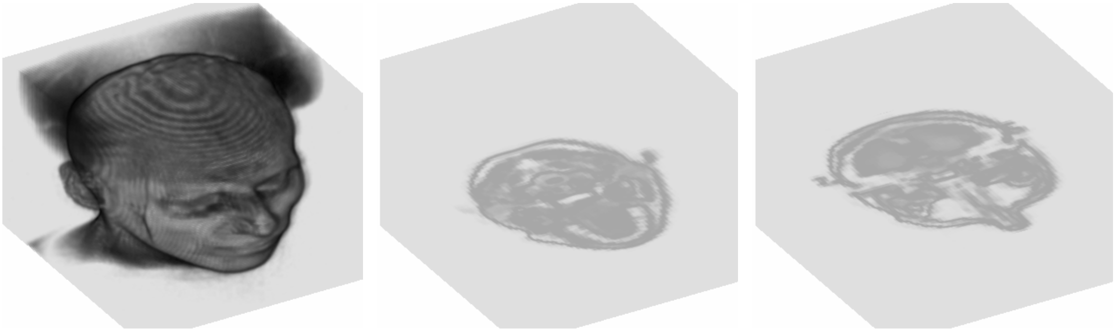


Figure 2-8: MRI of the human head and 2 slices extracted from the volume. MRI data by (RSI 2003a) rendering and slice view created using iVolume (RSI 2003b).

2.5.3 Computer Graphics for Displaying Feature Space

The field of 3D computer graphics is too large to discuss in detail in a study such as this. Numerous authors have published on the subject (Firebaugh 1993; Foley et al. 1993; Hearn & Baker 2003; Mortenson 1999). In this study the primary concern is with displaying 3D objects with volume such as ellipsoids and polyhedra. These shapes can be used to represent class clusters in feature space. These shapes are usually represented by a boundary representation (Hearn & Baker 2003). In a boundary representation a shape is defined by a list of vertices in 3D space which define its points and a table defining how these vertices are linked to form faces. These are known as the vertex array and connectivity (or polygon) table respectively (Hearn & Baker 2003). By appropriately defining vertices and the linkages between them, shapes can be represented in 3D space. The construction of these shapes for visualisation of class clusters is described in section 3.5.

As shown in Figure 2-5 feature space can be represented by a simple scatter plot. Facilities for plotting points in 3D space are common in graphics libraries (Hearn & Baker 2003). However, plotting a point for every pixel in a region or image may be too computationally expensive. Plotting a large number of points can cause unacceptable degradation of response times for interactive applications. The high and ever increasing resolution of satellite imagery means a large number of points need to be plotted.

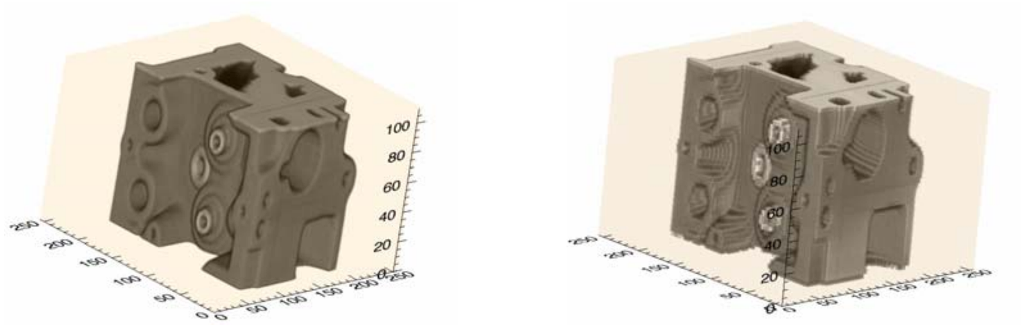
Feature space can also be feasibly represented by one or more volumes. This volume representation is usually a *generalisation* of true feature space. This allows a trade off between visual accuracy and processing speed to be obtained, the higher the degree of generalisation, the less accurate and less computationally intensive the scene is to process. Specific means of constructing feature space volumes and the issues involved in this performance trade off are discussed in section 3.2.

Volumes may either be rendered directly or have surface representations extracted from them. Extraction of surfaces for boundary representation is discussed in section 3.5. Many techniques exist for the direct rendering of volumes (Drebin, Carpenter & Hanrahan 1988). Two popular methods are volumetric ray-casting and texture mapped volume rendering. Others of interest include: splatting and shear warp (Lacroute & Levoy 1994).

Volumetric ray-casting produces high quality volume renderings as demonstrated in Figure 2-9a. However, it is computationally intensive and cannot normally be used in real time processing[2]. Texture mapped volume rendering can produce results rivalling ray-casting with significantly lower computational cost, allowing its use in real time processing. Whereas ray casting samples the volume many times for each ray, texture mapping samples, or rather *slices* the volume at a similar number of regular intervals but only once. The trade off comes when extracting the slices. The slices may be either viewport aligned or volume aligned. Viewport aligned slices produce high quality output but require the use of graphics hardware supporting 3D textures. Volume aligned slices produce poorer quality output as demonstrated in Figure 2-9b but have no hardware requirements.

---

[2] Modern graphics hardware (GPUs supporting pixel and vertex shaders) is now being employed to accelerate and parallelise traditional techniques such as ray-casting making them practical for real time processing.



(a) **(b)**

Figure 2-9: Volume rendering of a CT scan of an engine component. (a) Ray-casting, (b) Volume aligned texture mapping.

2.6 Visual Data Mining

2.6.1 Introduction

Visual Data Mining (VDM) can be thought of as the melding of (interactive) scientific or information visualisation (Section 2.5) with data mining (Section 2.2). The term is believed to have first been used by Cox et al. (1997). This study limits the definition of VDM to the use of scientific visualisation because remotely sensed data is spatial in nature.

VDM uses interaction to allow a human user to visually extract and explore patterns in data. When conducting a non-visual data mining, no matter how unbiased it may seem, the fact is that by simply choosing to carry out an automated analysis a priori assumptions have been made about what form the important results will take before analysis has actually begun (Simoff 2002). By visually mining the data this prior bias can be removed. Whilst the bias is removed, subjectivity of the analysis is vastly increased as it is based on a user’s perception, a point highlighted by many machine learning purists. However, this increased subjectivity is compensated for by a vastly increased degree of confidence in the analysis (Keim, D A 2002). VDM not only seeks to allow a human user to visually mine data but also to augment the non-visual data mining process. This augmentation usually takes the form of making the automated process more transparent to the user, hence providing increased confidence.

As stated above, when conducting a non-visual analysis, bias is inherently present in the choice of conducting a particular quantitative analysis. When conducting such analyses a hypothesis must be pre-determined. The analysis is then run to test the hypothesis. By contrast VDM can be seen as a hypothesis generation process (Keim, D A 2002). Visualisation of the data helps the user create new hypotheses. These hypotheses may then be tested by purely visual or non-visual data mining, or by a combination of the two. The results of the test can also be visualised. This allows for effective revision of the hypothesis.

The *Information Seeking Mantra* (Shneiderman 1996) defines the normal process of visual data mining: ‘Overview first, zoom and filter, and then details-on-demand’ (Keim, D A 2002). This is the visual analogy of a divide-and-conquer style algorithm. However, only interesting parts of the data are analysed.

Visual Data Mining is more than the application of scientific visualisation to attribute-value style data. There must be interaction that allows the user to mine the data for patterns (Ferreira de Oliveira & Levkowitz 2003). This usually means the incorporation of non-visual, automated data mining, into the visualisation system. The system should allow the user to create new hypotheses and test them using the automated tools. The tests may be applied at varying scales and to various subsets of the original data.

A true visual data mining system has one more key component: the visual representation of the automated data mining algorithms to be applied. Many users of data mining have no need to be concerned with the underlying implementation of the quantitative analysis they are performing. However, a basic understanding of *what the algorithm does* to achieve its goal is important in order to understand what effect the analysis will have on a given set of data. From a classification perspective this is the visualisation of decision boundaries and parameters in feature space.

Visual Data Mining can deal with inhomogeneous and noisy data. Many classification algorithms struggle to deal with noise and unexpected structure in training data. Human visual perception can easily deal with these issues. The user’s visual perception can be used to reduce possible sources of classifier confusion or to visually conduct the analysis themselves.

2.6.2 Exploratory Data Analysis

Visual Data Mining is a type of Exploratory Data Analysis (EDA). Non-visual data mining is a form of Confirmatory Data Analysis (CDA). This is essentially formalises the statements in Section 2.6.1; EDA in the form of Visual Data Mining is used to explore the data prior to the use of, or in place of, CDA in the form data mining. Wegman (2001) states:

> In the original EDA framework, both visual and analytical tools were used to explore data and confirm that the data conformed to the assumptions underlying the confirmatory statistical methods employed to analyze the data. With the advent of high performance personal computing, more aggressive exploration of data has come into vogue. No longer was EDA simply used to verify underlying assumptions, but also was used to search for unanticipated structure.

2.6.3 Summary

The key benefits of Visual Data Mining can be summarised as follows:

- The ability to deal with inhomogeneous and noisy data,
- Light expectations concerning user knowledge of data mining algorithms,
- The ability to educate users regarding the operation of these algorithms,
- The removal of prior notions regarding the form of the important information in the data, and;
- Higher confidence factors in findings as the user has been involved in the process and understands the steps carried out by any automated analysis that may be performed.

Visual Data Mining's major drawback is the subjective a nature of any results obtained. In a purely visual situation, it may become tedious to conduct a large number of analyses, but instances where purely visual DM is the optimal strategy are relatively few. Most analyses would combine visual and automated techniques.

2.7 Visual Data Mining of Remotely Sensed Data

Interpretation is an important stage in the process of remote sensing (Section 2.1.4). Visual Data Mining can be used at this stage to aid in image interpretation or quantitative analysis. It can also be employed in conjunction with both of these traditional techniques. Few studies have investigated and discussed the use of VDM in remote sensing: Braverman & Kahn (2004), Keim, D. A. (2001), Keim, D. A. et al. (2004), Lucieer (2004), Lucieer & Kraak (2004). Many prior studies have focused largely on the visualisation rather than the data mining element. Lucieer (2004) visualised the fuzzy classification of remotely sensed imagery. A tool was developed whereby a user could visually adjust parameters of the fuzzy classifier. The use of such a system improved insight into fuzzy classification of remotely sensed imagery. Also, Lucieer (2004) investigated the use of visualisation of object segmentation and related uncertainty. This part of the study can be considered post-processing VDM; quantitative analysis was applied and results visualised.

The aim of this study is to investigate the effect providing VDM tools within a remote sensing software package has on analyses performed using such a system.

Chapter 3 Methods

3.1 Overview

In order to assess the value of Visualisation and Visual Data Mining of satellite imagery, a system with these features has been constructed. Most commercial remote sensing software packages lack visual exploratory analysis tools. ENVI (the *Environment for Visualising Images*) is a leading remote sensing package with an extensible framework. This framework has been used to create a visual data mining environment for satellite imagery.

This chapter details the algorithms and technologies used in the construction of a prototype visualisation system. The system features a volume based feature space representation. The volumes are derived from regions of interest (ROIs) within a digital image. Direct volume rendering can be applied to directly visualise these volumes. Ellipsoids are used to show the mean and variance of a region. Surface representation is used to show the full extent of regions in feature space. This representation is by means of isosurfaces and α-shapes. The prototype is written in IDL (the *Interactive Data Language*, created by *Research Systems Inc.*) as an extension to ENVI; a leading commercial remote sensing package written in IDL. Combined with ENVI, the prototype forms a visual data mining system for satellite imagery.

Figures in this chapter have been generated by the prototype visualisation software unless otherwise cited. Unless otherwise stated the dataset used to generate the figures is a Landsat TM image of Hobart, Tasmania, Australia captured on 28/9/1999. This image has 6 bands as described in Section 2.1.2. The spatial resolution is 30 m and the radiometric resolution is 8-bit.

The axes for most of the figures in this chapter were set to use 5 numbered tick marks regardless of volume size in use. The tick marks are only present to provide a reference between figures and are not indicative of volume size (see section 3.2). Unless otherwise stated, figures have been generated with a volume size of 64.

3.2 Constructing Volume Representations

Pixels in an image can be thought of as being a 2D array of size b x n (b = number of bands, n = number of pixels). One way plot this data in 3D feature space is to create a 3D frequency histogram of the 2D data based on three of the image bands. This 3D histogram is a *volume* representing the pixels which can be directly rendered in 3D feature space. In this way the frequency of certain pixel values becomes the density of the corresponding voxels in the volume. Figure 3-1 shows a volume in which every second voxel has been assigned the value zero in order to illustrate individual volume elements or *voxels*. The value at every other voxel could be a frequency at which pixels with certain values occur. A volume representation such as this is a summary of the set of pixels; each voxel represents the frequency at which pixels with certain values occur. For example, a voxel with coordinates (10, 5, 14) might represent the frequency of pixels with the exact value (10, 5, 14) in three image bands. More commonly each voxel represents a range of pixel values. This is a *generalisation* of feature space. The degree of generalisation is defined by the volume size. The volume size is the number of voxels along the x, y and z axes of the volume. In Figure 3-1 the volume size is 20 (with only half the voxels visible).

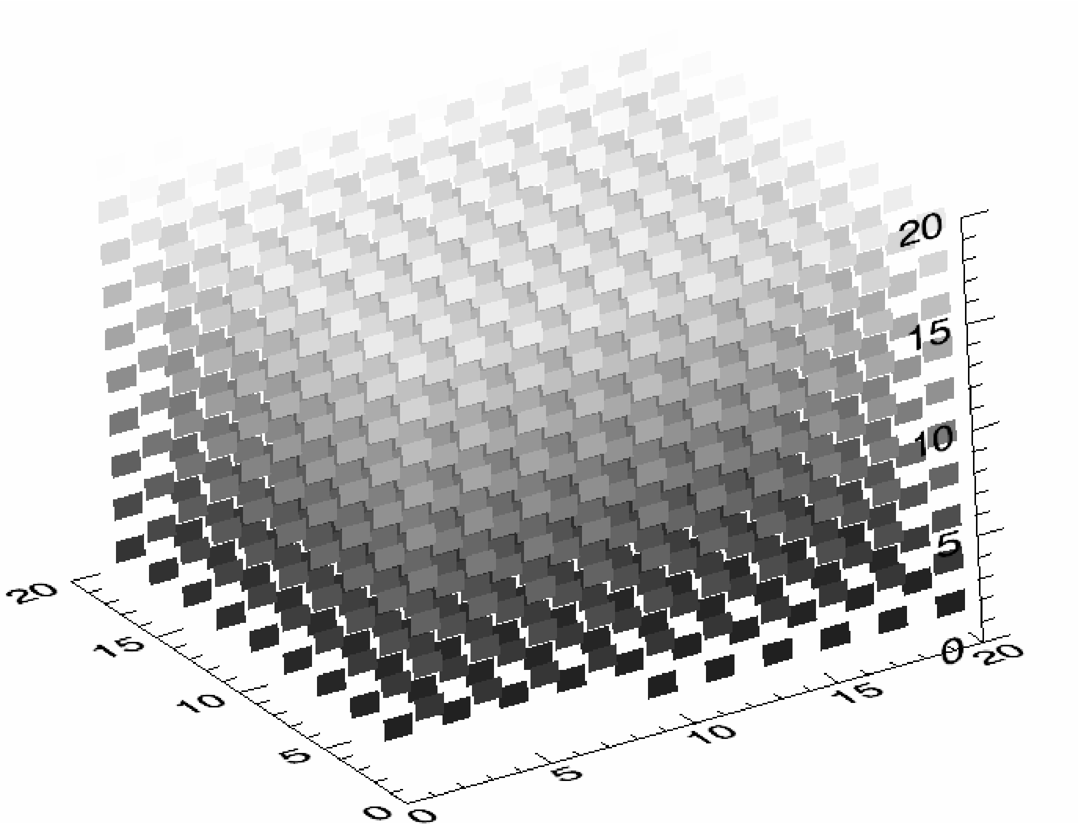


Figure 3-1: Example volume rendering showing varying voxel values. The darker voxels have higher values, lighter voxels have lower values. Every second voxel has a value of zero. Voxels with zero values are rendered as fully transparent in order to illustrate individual voxels.[3]

Volume size determines the degree of generalisation by adjusting the granularity of the volume. For example, take the X dimension of a volume. This dimension represents the columns of the volume. Each column is made up of a number of rows (in the Y dimension) and slices (in the Z dimension). This X dimension is assigned to one band of a satellite image. The radiometric resolution is 8-bits, so the data range is 0 to 255. However only values between 8 and 215 are used in this band, so empty regions of feature space are removed by scaling the pixel values to lie between the minimum and maximum. The volume size determines what ranges of pixels will be grouped together.

With a volume size of 5, the columns of the volume might look like[4]:

---

[3] This volume was rendered with texture mapped direct volume rendering using nearest neighbour interpolation for crisp edges. The volume itself was created with the following IDL commands:

```
IDL> vol = replicate(1, 20,20,20)
IDL> vol[*,*,0:*:2] = 0
IDL> vol[*,0:*:2,*] = 0
IDL> vol[0:*:2,*,*] = 0
IDL> vol[where(vol)] = uindgen(1000)
```

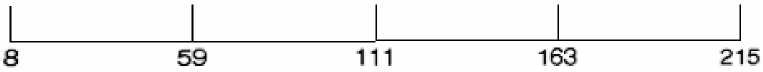


Figure 3-2: Ranges of pixel values to be sorted to five columns.

This means that all pixels with values ≥8 and <59 will be sorted to the first column, pixels with values ≥59 and <111 will be sorted to the second, and so on. If the volume size is increased to 10, the granularity becomes finer, hence less generalisation:

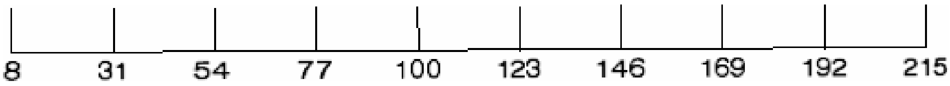


Figure 3-3: Ranges of pixel values to be sorted to ten columns.

This doubling in volume size doubles the detail expressed by the volume. This also doubles the compute time for operations on the volume. In the extreme case the volume size may bet set equal to the range of the data. In this case a single pixel value will be mapped to each column. The opposite extreme is a volume size of 1, in this instance all pixels are sorted to the one existing column, indeed for a cube volume, to the one existing voxel. This concept of generalisation can be scaled up to the 3 dimensions of feature space by simultaneously applying similar reasoning to two more dimensions and bands.

Rendering one voxel unique pixel value combination, resulting in no generalisation, is often too computationally intensive to be considered feasible. Remotely sensed data is often of 8-bit, 16-bit integer or 32-bit floating-point type. A volume with one voxel per pixel value combination for an unsigned 16-bit image would be approximately 32766GB in size. Few remotely sensed images will use this full range of the data type in any of their bands. Constructing volumes based on the full data type range of the image often gives poor visual results with the volume being largely empty and interesting areas being too small to visually explore. For these reasons the prototype constructs volumes of a user defined size (the same in each dimension), and scales pixel values to fit the user defined size. This scaling takes into account the minimum and maximum values in each band for the given set of pixels. This usually

---

[4] This range of values (215 – 8 = 207) does not divide evenly. Labelled values have had their decimal portions truncated for simplicity.

produces a relatively full feature space and increases the users' opportunity to observe different patterns in the data.

As stated above, volumes constructed in this study are of equal size in each dimension. The default size for volumes in the prototype software is 64 in each dimension. The user may choose any size they wish, however, sizes larger than 256 are considered too large for interactive volume rendering. As the array is three dimensional, the increase in size from 64 to 255 causes an exponential growth in compute time and memory allocation; 64^3 = 512kB (64^3 x 2 bytes per voxel) vs 256^3= 32mB (256^3x 2 bytes per voxel).

Volumes are implemented as 3D arrays of 16-bit unsigned integers, yielding a maximum frequency in any one voxel of 65535. This should be enough for most practical purposes and reasonable volume sizes (volumes with few voxels relative to the range of the pixel values would be considered unreasonable). Measures were taken to check for overflow and report this to the user. If overflow should occur the value will be capped at the maximum representable 16-bit unsigned integer.

Pixel values must be scaled such that the range of values for each attribute becomes equal to the range between zero and the size of the dimensions of the array. Scaling is carried out on a per-dataset, per-band basis. The dataset refers to a number of sets of pixels to be visualised simultaneously in 3D feature space.

This process of constructing a volume from a set of pixels is condensed into the function `bin_volume(volume-size, dataset,...)`[5] which returns a volume of given size in each dimension representing the pixel data given.

3.3 Intersection of Volumes

One of the purposes of the prototype is to explore the interactions between training regions for different land cover classes. The prototype allows the user to generate and visualise intersections of multiple volumes. Intuitively this involves examining where volumes intersect in 3D feature space, which in geographic space translates to examining pixels in different training sets for similar values. These similar values represent spectral intersection between ROIs. Comparing each pixel in a training set

---

[5] Refer to Appendix B for full function header and code listing.

to every pixel in every other training set with some measure of similarity (eg: Euclidean distance) has unsatisfactory time-complexity. This brute force style method described above would achieve the most accurate result and would allow the use of a similarity threshold at which pixels are included in the intersection. However a more efficient method is available.

The existing volumes for each region can be used to perform the intersection operation. Having already constructed the volumes it is simply a matter of preforming a logical AND operation on every voxel of a volume with the same voxel in other intersecting volumes. Although this method is performing a similar operation to that described above it is preferred due to the fact that time-complexity is constrained by the size of the volume not the range of pixel data. The measure of similarity is essentially defined by the volume size. Using an AND style operation also allows the process to short-circuit as soon as a zero value is found in a voxel. This method has the disadvantage of performing the intersection based on values that have been scaled to fit the volume size; potentially causing greater intersection than would be desirable. This problem fades as the size of the volume approaches the range of the data. This problem is discussed in 3.6.2.

The intersection routine implemented in the prototype creates a new volume consistent with those to be intersected and finds voxels where intersection occurs. Each voxel in the new volume where intersection occurs is assigned the value of the sum of that voxel over all volumes being intersected (overflow is handled by capping at the maximum value and notifying the user). All other voxels have value zero.

3.4 Direct Volume Rendering

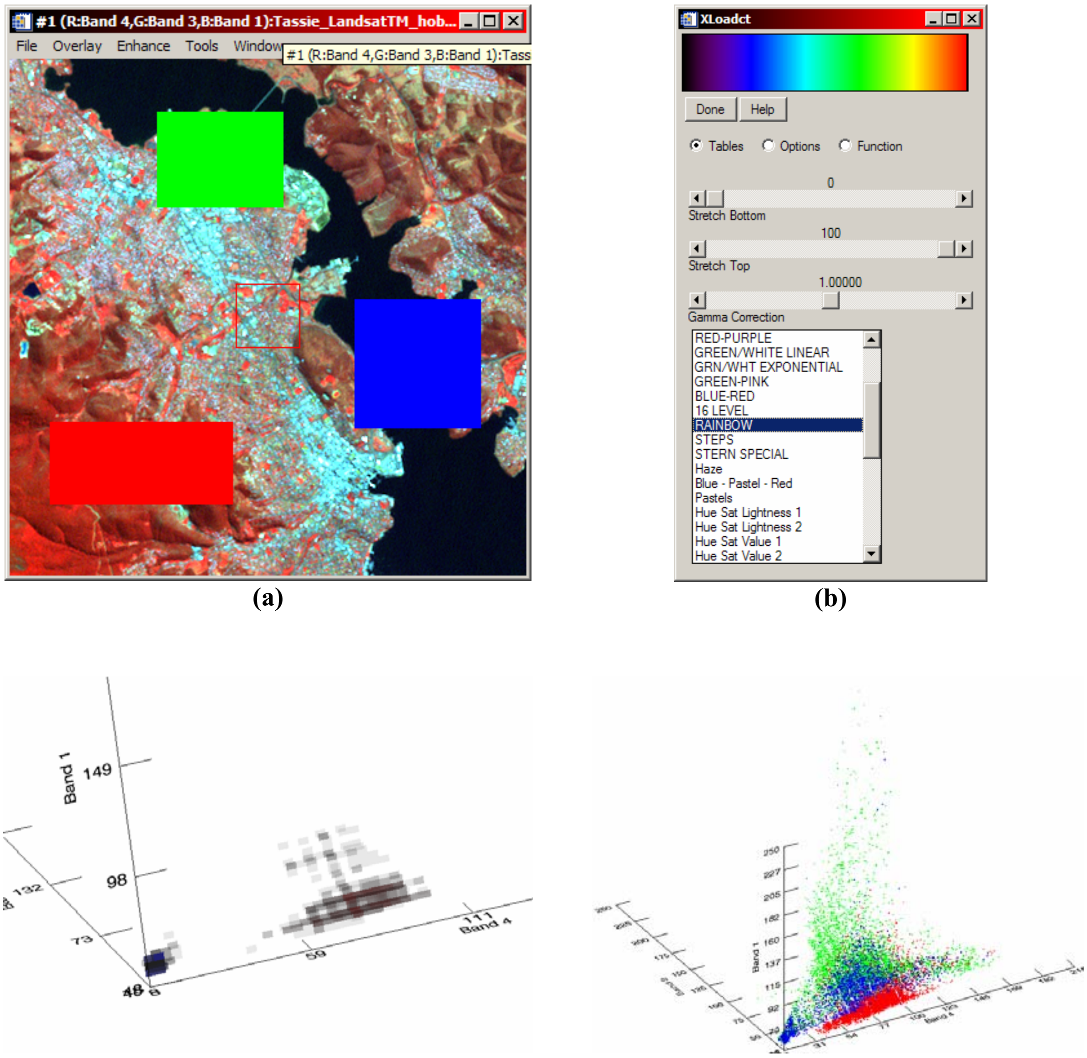
Section 3.2 describes the technique used to construct a volume representation of a set of pixel data. In this study those pixels are extracted from regions of interest in a satellite image. The prototype allows for direct volume rendering of one independent volume or three consistent volumes with a fourth volume defining opacity. These are referred to as single-channel rendering and four-channel rendering respectively. In single channel rendering the density at each voxel is used to index a grey-scale colour table. This determines the output colour for that voxel. For four-channel rendering each of three volumes index one element of an RGB colour table. A fourth volume indexes an opacity table. The outputs of all these volumes are merged to form the final rendering. There are two renderers available; a fast, low quality, 2D texture-mapped approach and an accurate ray-casting approach. The user may switch between these renderers at any time.

3.5 Boundary Representation and Surfaces

This section describes 3D surface and boundary representation methods used to enhance visualisation of class clusters in feature space. Section 3.5.2 describes how basic ellipsoids are formed from statistical measurement of a set of pixels. Sections 3.5.3 and 3.5.4 describe methods used for constructing 3D meshes which closely approximate the true shape of the cluster represented by a volume.

3.5.1 Problems with Direct Volume Rendering and Point Clouds

Direct volume rendering is most useful in fields in which the data is inherently in volume form, as in the example given of medical MRI. It is less visually useful when data is not inherently in this form as is the case when plotting geospatial data in feature space. The reason geoscientists usually want to visualise multiple volumes at once is to examine how they interact with one another. Without some very ingenious use of the colour table it is difficult to show this interaction. For example, consider the following scenario referring to Figure 3-4: the user wishes to display the red region as clusters tending toward red and the blue and green regions in a similar fashion. The alpha channel is defined as the sum over all colour channels per voxel. They select a rainbow style colour table in order that each channel will show the appropriate colour in varying intensities. The resulting rendering of the volumes in feature space is disappointing, with many of the voxels having frequencies too low to plot. The blue region volume contains many dark pixels; 2643 of the 9800 pixels in the region have values which occur in voxel (1,1,2) of feature space. This is by far the maximum frequency in the plot and causes small values to be scaled even lower, in some cases to zero. Voxel values are scaled to byte values so that they may be used to index the colour and opacity tables. Large values cause smaller values to be truncated to zero after scaling. The direct volume rendering of these volumes shows the high frequency of blue (water) in the lower corner of the plot and the medium concentration of red (vegetation) in the mid-range of band 4 (infrared) and low ranges of two visible light bands. The green region, representing mostly man-made land cover, is too sparsely distributed to be recognisable by visual inspection alone.



● Tables

Stretch Bottom: 0

Stretch Top: 100

Gamma Correction: 1.00000

Selected: RAINBOW

(a) (b)

(c) (d)

Figure 3-4: Screen captures from an example usage scenario. (a) false colour display of Landsat TM image of Hobart, TAS. (b) selection of a predefined colour table, (c) unsatisfactory volume rendered output (alpha-enhanced), (d) original region data shown as a point cloud (ungeneralised feature space).

It is predicted that standardising the volume prior to direct volume rendering would alleviate the scaling issue but this has not been implemented in this prototype. The issue of visualising the relationships between clusters still remains. Other means such as boundary representations can easily achieve this.

Point clouds provide a simple representation of clusters in feature space. The disadvantages of point clouds were discussed in section 2.5.3. The prototype allows the visualisation of regions as point clouds for the purposes of comparison to other surfaces.

3.5.2 Ellipsoids

Historically parametric classifiers have been popular in remotely sensed image classification. It is useful to visualise the parameters that will be used for classification. By visualising a statistics based ellipsoid of a data set against the actual shape of the volume as represented by an isosurface or α-shape useful insight into maximum likelihood classification results can be obtained (see Figure 4-13).

A 3D ellipsoid can be constructed based on the statistics of the data set. The mean vector of the data represents the centre of the ellipse. Computing the covariance matrix allows the calculation of *eigenvalues* and *eigenvectors*. Eigenvectors can be used represent the direction of the axes of the ellipsoid and eigenvalues to represent the length of those axes. Obtaining these values is known as *eigen-decomposition* and can only be carried out on square matrices such as the covariance matrix. Abdi (2007 (in-press)) provides an introduction to eigen-decomposition. The use of these values to reasonably approximate the shape of a cluster relies on the assumption of a normal distribution of points in that cluster.

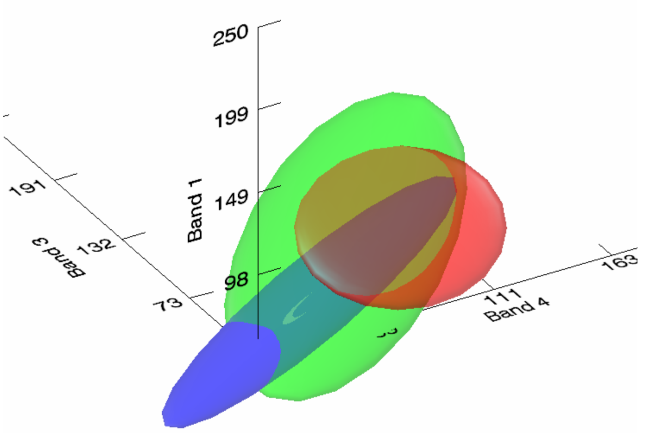
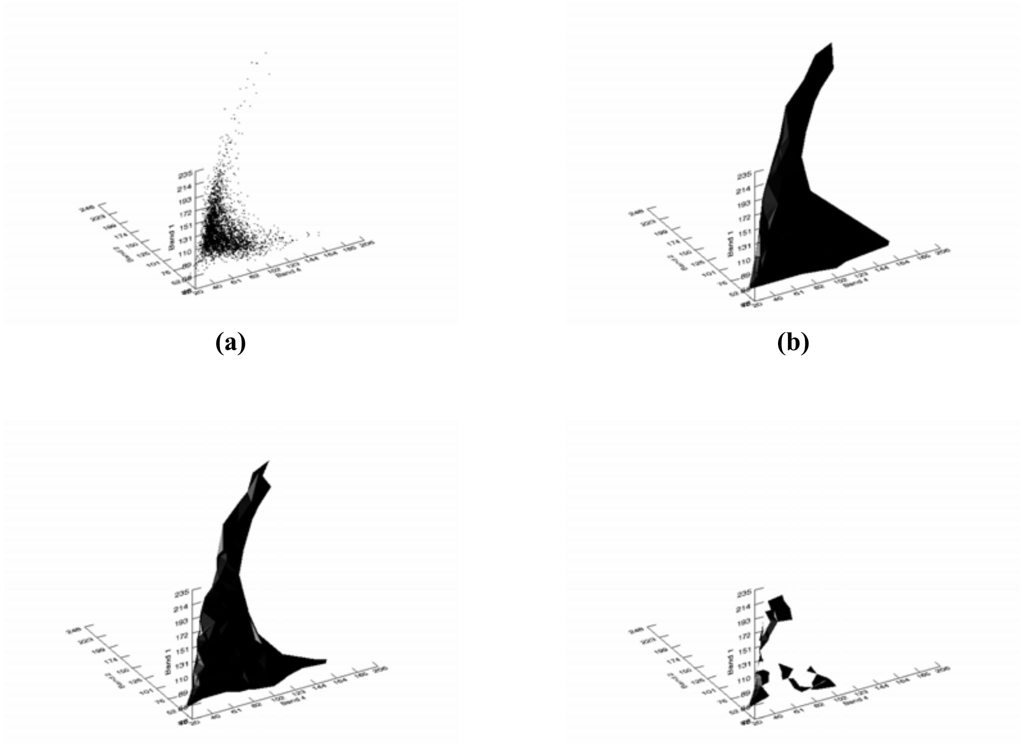


Figure 3-5: Ellipsoids for regions defined in Figure 3-4a at 60% transparency

3.5.3 α-shapes

Basic shapes such as ellipsoids do not provide an adequate representation of the extent of a cluster. Ellipsoidal representations make the assumption of a normal distribution within the data. This is frequently not the case in satellite imagery.

α-shapes can be used to accurately represent the true extent of irregularly shaped or concave clusters. Edelsbrunner & Mücke (2000) state ‘The geometric notion of “shape” has no associated formal meaning.’ They introduced α-shapes as a formal definition and computation of the shape of finite point sets in Euclidean space. For a variable parameter α, shapes produced by this computation range from crude to fine representations of the cluster. This relationship is such that for $\alpha = \infty$ the α-shape of a point set *S* is equal to the convex hull of *S*. As α decreases the shape shrinks and moulds to the concavities of the point set. Tunnels and holes may appear and the connectivity of the shape may be broken as shown in Figure 3-6.



(a) **(b)**

(c) **(d)**

Figure 3-6: α-shapes for a sample point set (a), with α values of (b) 784, (c) 100, (d) 50

Formally, if S is a finite point set in R^d (where d is the number of dimensions) and α a real number such that $0 \leq \alpha \leq \infty$ then the α-shape of *S* is a polytope (in 2D a polygon or in 3D a polyhedron) that is neither necessarily convex nor connected. Fischer (1994) provides an excellent metaphor for describing α-shapes:

> One can intuitively think of an α-shape as the following. Imagine a huge mass of ice-cream making up the space R^d and containing the points *S* as “hard” chocolate pieces. Using one of these sphere-formed ice-cream spoons we carve

> out all parts of the ice-cream block we can reach without bumping into chocolate pieces, thereby even carving out holes in the inside (eg. parts not reachable by simply moving the spoon from the outside). We will eventually end up with a (not necessarily convex) object bounded by caps, arcs and points. If we now straighten all "round" faces to triangles and line segments, we have an intuitive description of what is called the α-shape of *S*.

In 3D feature space $R^d = R^3$ as the shape is limited to three dimensions. A graphical illustration of Fischer's metaphor for a point set in R^2 is shown in Figure 3-7. In this case the ice cream spoon is replaced by a simple circle.

Edelsbrunner & Mücke's (Bloomenthal & Wyvill 1997) algorithm for generating α-shapes takes as input a list of points in Euclidean space; that is, a vertex list and the parameter α. It returns a polygon table defining the faces of the calculated shape.

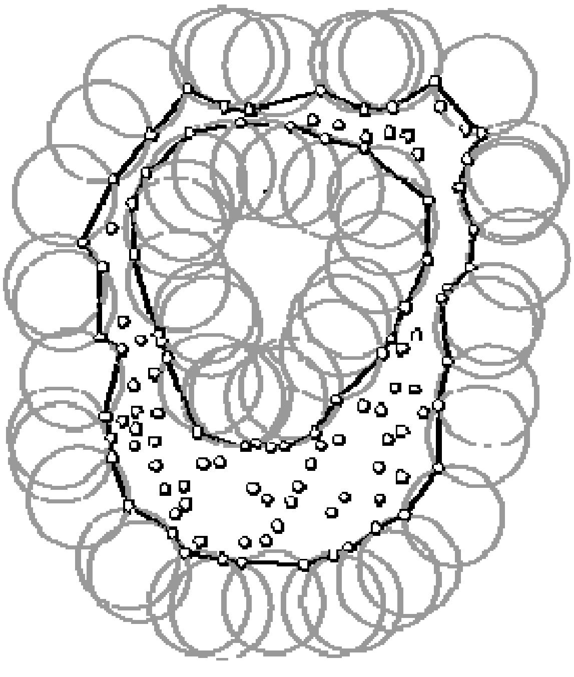


Figure 3-7: Example formation of a 2D α-shape. The parameter α determines the radius of the disc connecting any two points on the edge of the final shape (Fischer 2000).

As stated above, α-shapes can be used to accurately represent concave and irregular shapes. This property can be used to accurately visualise clusters in 3D feature space. Each voxel with value greater than zero and address (x,y,z) can be considered a point in the finite set Z^3 (representing Euclidean space) with coordinates (x,y,z). The set is denoted Z^3 rather than R^3 as the points will be strictly integers. These points make up a cluster which defines the “shape” of the set. In this way α-shapes in the prototype are still tied to the volume representation. This is important as α-shapes are computationally expensive to construct (Lucieer 2004, p. 126). The trade off between detail and processing time based on volume size must still be considered.

α-shapes are implemented in the prototype using *hull* (Clarkson 2004), an external program written in C. Hull takes as input a list of points in 3D space, that is, a vertex array, and value for α. It produces a list of faces which defines the polygon connectivity for the shape. Each face consists of the three indices of the vertex array in right-handed[6] order.

3.5.4 Isosurfaces

A different approach to volume rendering is the extraction of polygonal meshes from the volume data. This gives a surface representation of the volume. Isosurfaces (Bloomenthal & Wyvill 1997) are a popular means of doing this. The surface generated varies depending on a user specified isovalue. Like α-shapes, isosurfaces can be used to represent the extent of clusters in feature space. They also provide a visual measure of the density of the clusters. Unlike α-shapes, isosurfaces require the initial data to be in volume form. The prototype fulfils this requirement by generalising feature space into volume form.

An isosurface can be thought of as a three dimensional isogram (a line connecting points in space having the same numerical value of some variable). A useful analogy is that of isobars on a weather map. Barometric pressure is sampled at various points across the Earth. A grid of dots could be drawn where each dot represents some area on the ground (eg. x km^2), Some of the dots would have barometric pressures associated with them (where a measurement has been made). Lines are then drawn through points of equal pressure forming an isobar. Obviously there is not a measurement of pressure for every dot, values are interpolated according to the values of neighbouring recordings. Expanding this to three dimensions (a 3D grid of dots, each representing an area of x km^3) it is possible to imagine a value being

---

[6] In a right-handed system the Z (depth) axis increases toward the viewer. Vertices are ordered anti-clockwise in polygons tables. Hearn, D & Baker, PM 2003, *Computer Graphics with OpenGL*, 3rd edn, Prentice Hall.

available for every point in that 3D space; essentially a volume of barometric pressure readings.

Isosurfaces can be calculated using the Marching Cubes algorithm (Drebin, Carpenter & Hanrahan 1988). The algorithm builds a polygonal mesh according to the specified isovalue. The mesh will represent a surface around or through the volume where voxels have values equal to the isovalue. The effect of the isovalue on the extent of an isosurfaces is demonstrated in Figure 3-8. This can be exploited to visualise where the more dense portions of a cluster lie.

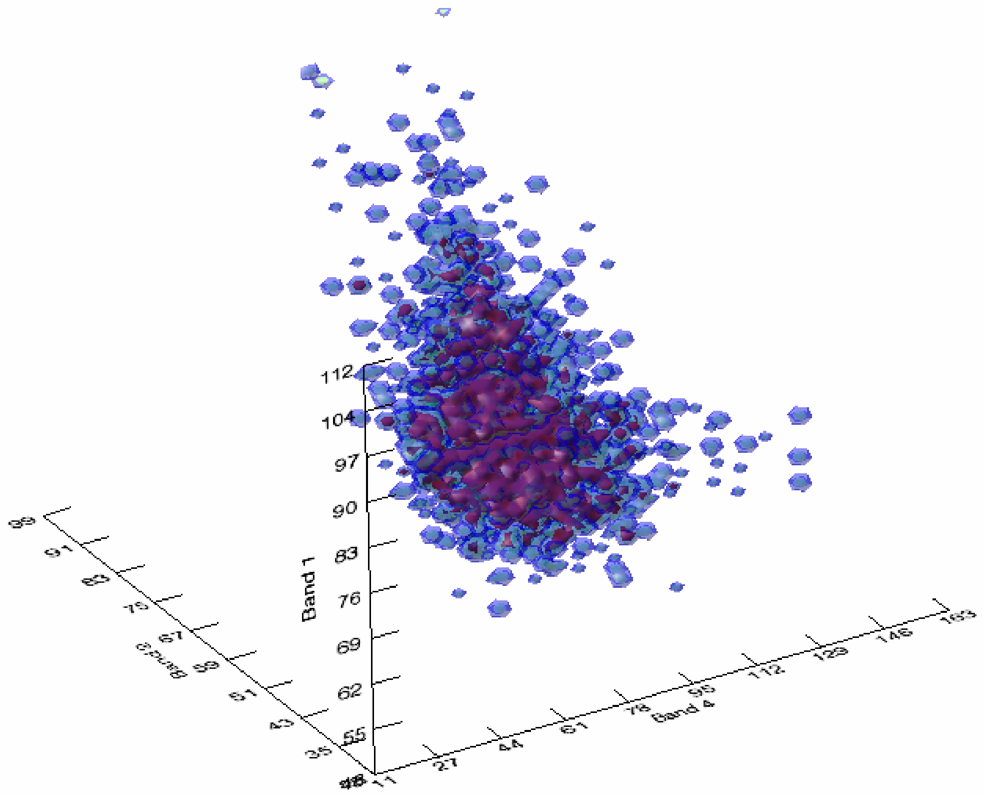


Figure 3-8: Three isosurfaces for a cluster of points in feature space. The Blue surface has an isovalue of 0.1, the green, 0.5 and the red 1.0.

Recall that the prototype can already construct volumes from ROIs in the image. Some voxels in these volumes may contain frequencies greater than zero. These voxels can be used to determine *potential* for each voxel in the volume. The construction of an isosurface then is the process of extracting a polygonal mesh from the volume representing voxels of equal potential. The isovalue is the user selected potential at which to generate the surface.

In order to build an isosurface each point in (ungeneralised) feature space must have a sensible numeric value. This value can be derived in the following manner, as used by (Lucieer 2004): Each pixel when plotted in feature space can be considered a source of gravitational potential energy, analogous to a planet in the universe. In the general case, potential energy (or simply *potential*) *Ug* is calculated:

$$U_g = -G\frac{m_1 m_2}{r}$$

Equation 3-1: Potential energy in a gravity field

where *G* is the gravitational constant[7], *m1* and *m2* are masses and *r* is the distance between them. This can be simplified for an artificial environment such as a feature space plot:

$$U_g = \frac{m_1}{r^c}$$

Equation 3-2: Potential energy in feature space

where $G = 1.0$, $m_2 = 1.0$ and U_g is the potential for a given point in feature space. m_1 is the mass of a source determined by the distance to the closest neighbouring source. *c* may be used to adjust the power of the potential energy. The potential then for any point in feature space is the sum over all sources of potential for that point. Potential will be high for points near sources and low for points distant from sources. This approach works for feature space plots based on point clouds or other individual pixel representations. This study uses volumes to represent feature space. A small adjustment is needed to generate sensible isosurfaces on these volumes.

The variable m_1 can be thought of slightly differently for constructing isosurfaces based on volumes (eg: a 3D frequency histograms). In this case each non-zero valued voxel in 3D space already represents a cluster of one or more pixels, that is, it has a value that can be treated as the voxel's mass. Voxels with higher frequencies will cause higher potentials in surrounding voxels of feature space.

---

[7] http://physics.nist.gov/cgi-bin/cuu/Value?bg|search_for=Gravitational+Constant

Isosurfaces are commonly generated using the marching cubes algorithm (Lorensen & Cline 1987). This divide-and-conquer algorithm divides space into cubes, similar to the voxels in a volume. Each cube references eight points in feature space, the points at each corner of the cube (Figure 3-9). For each cube, the intersection between the cube and the isosurface is calculated. Intuitively, if one or more of the points has a value less than the isovalue and one or more other points have values equal to or greater than the isovalue then the given cube must contribute some component to the final isosurface. Given this there are an enumerable number of possible intersections within the cube:

> Since there are eight vertices in each cube and two slates, inside and outside, there are only $2^8 = 256$ ways a surface can intersect the cube. By enumerating these 256 cases, we create a table to look up surface-edge intersections, given the labeling of a cubes vertices. The table contains the edges intersected for each case. (Lorensen & Cline 1987)

However, there are two important symmetries of the cube that reduce the 256 cases to just 14. First, the topology of the intersecting surface is unchanged if the relationship between the points in the cube is inverted. Removing these complementary cases reduces the number of cases to 128. Secondly, the cube possesses rotational symmetry; Lorensen & Cline further reduced the number of cases to 14 by inspection (Figure 3-10). An 8-bit index is used to index a table of edge intersections for a given cube configuration. Each bit in this index corresponds to one vertex of the cube and its relationship to the isovalue, thus describing a unique cube configuration.

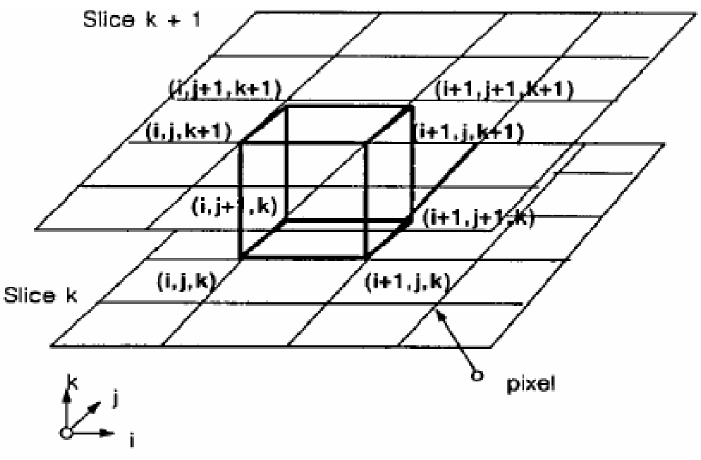


Figure 3-9: Marching cube (Lorensen & Cline 1987)

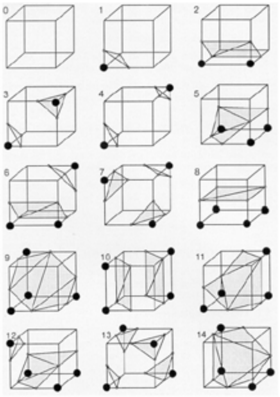


Figure 3-10: Symmetrically distinct triangulations (Lorensen & Cline 1987)

Having determined the cube configuration, polygons are generated and the process repeats for the next cube. After all cubes have been processed the generated polygons are connected into a single mesh by standard triangulation.

In the prototype system, feature space is represented by a volume. Each voxel in the volume can be thought of as a point in the previous description of the algorithm. Isosurfaces are implemented in the prototype using IDL’s built in isosurface construction routine. This routine uses a slight variation on the marching cubes algorithm; until recently marching cubes was covered by a U.S. patent (Cline & Lorensen 1985). The user may visualise as many regions as isosurfaces as they desire.

3.6 Highlighting Conflicting Training Data

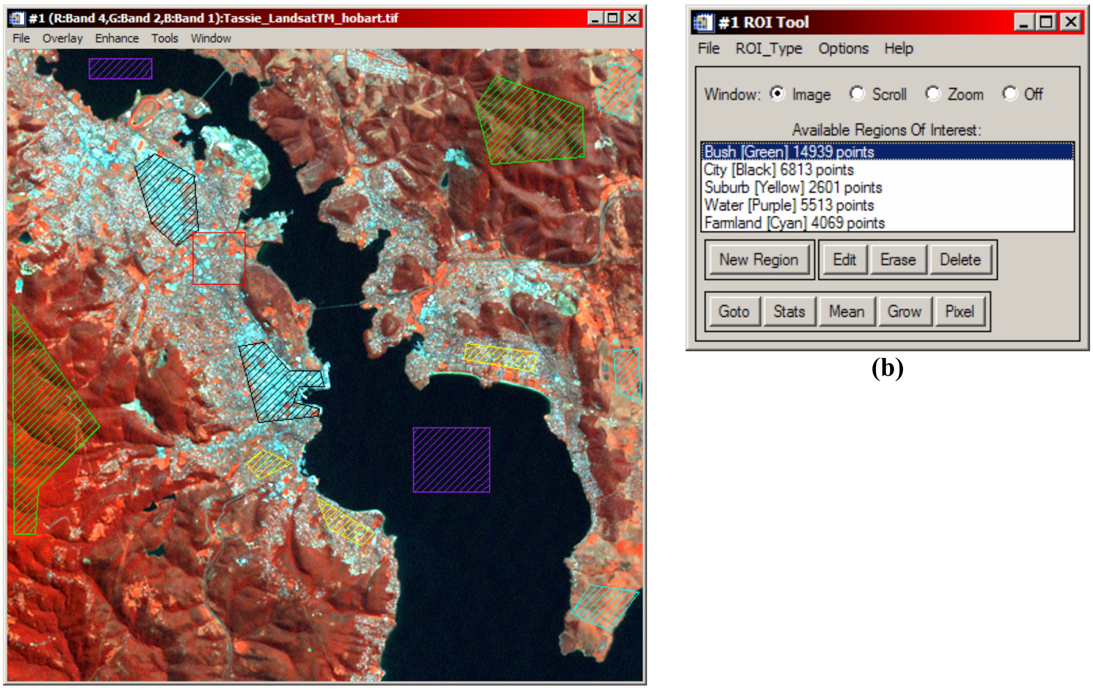
3.6.1 Linking Data Spaces

Section 3.3 described the technique used by the prototype to intersect multiple volumes. These volumes can then be visualised alongside others in 3D feature space. This feature is designed to allow the user to explore the relationships in their data. Whilst they may gain some understanding of their data through this visualisation it is difficult to appreciate the impact of this overlap on geographic space (ie: a colour-composite image view). This section describes the technique used to highlight pixels causing overlap in feature space, referred to as *offenders* in this document, in a colour-composite view.

The figures shown in this section are derived from regions within the Hobart Landsat TM image as shown in Figure 3-11. These regions are selected by visual inspection and bear no association with any ground truth data for the region. The purpose is not to verify the correctness of any classification based on these regions, but is to serve as sample data for visualisation in feature space. Indeed figures in this section highlight the vast overlap between the regions.

3.6.2 Limitations

Recall that the prototype has been constructed so that the processing layer is closely tied to the visualisation layer. For example, volume size is a visualisation parameter trading rendering performance for detail. This same trade off will occur for all operations on this volume; visual and non-visual. In this instance concern lies with the intersection operation. Take for example the yellow (Suburb) and black (City) regions in Figure 3-11. These regions overlap in feature space (Figure 3-12). However, the range of pixels falling in this overlap is controlled by the volume size as illustrated in Figure 3-13. The reduction in volume size will cause more pixel values to be captured in the intersection. This in turn will cause more pixels to be labelled in geographic space as causing intersection. The user must strike an acceptable balance between volume size and performance. This is discussed in Section 3.2.



Window: ● Image ○ Scroll ○ Zoom ○ Off

Available Regions Of Interest:

Bush [Green] 14939 points
City [Black] 6813 points
Suburb [Yellow] 2601 points
Water [Purple] 5513 points
Farmland [Cyan] 4069 points

(a)

(b)

Figure 3-11: ENVI display showing: (a) Landsat TM image of Hobart, TAS annotated with various regions. (b) Colours of regions and their assigned labels.

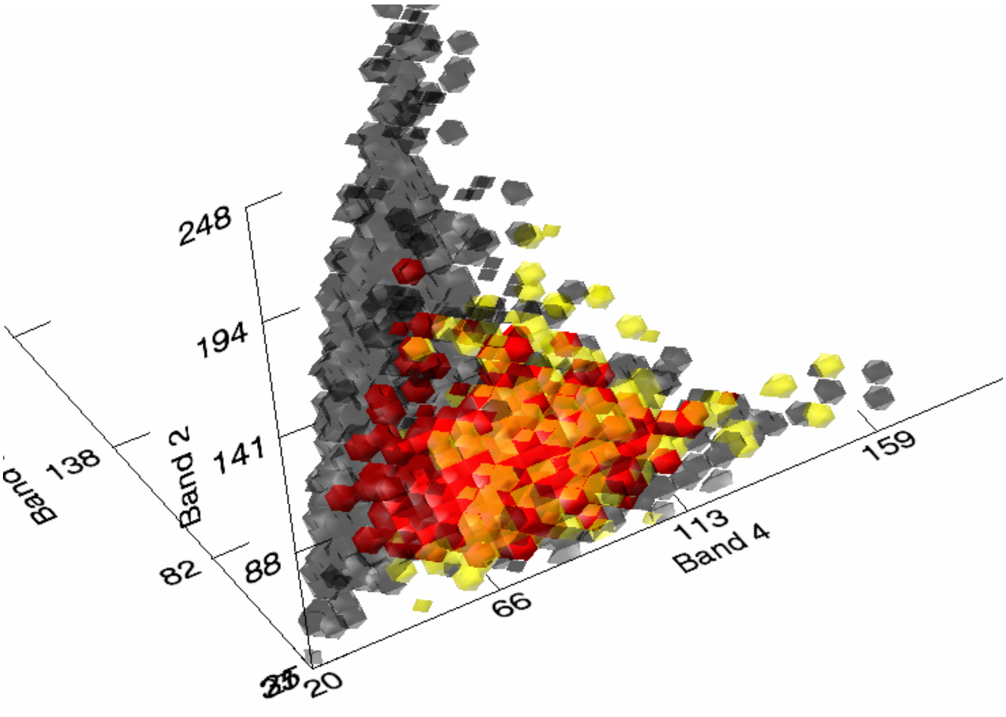


Figure 3-12: Feature space intersection between yellow and black regions shown as red (all as isosurfaces with isovalue 0.1) with a volume size of 64.

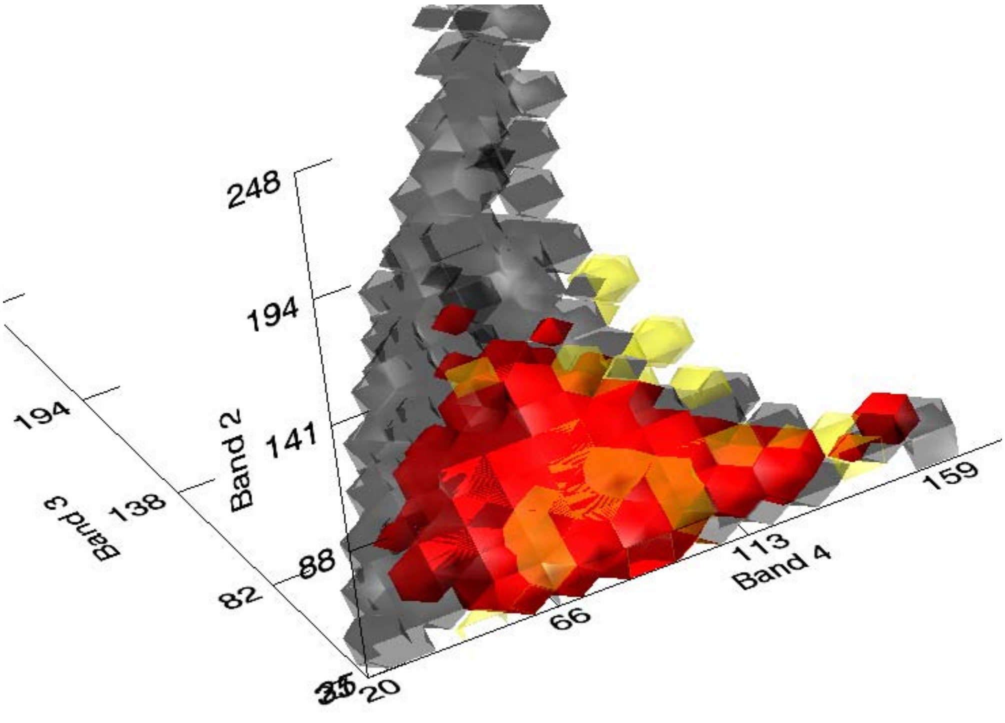


Figure 3-13: Feature space intersection between yellow and black regions shown as red (all as isosurfaces with isovalue 0.1) with a volume size of 32.

3.6.3 Determining Pixel Values

Recall that volumes are constructed from an array of pixels of size 3 x *n* where *n* is the number of pixels in a given ROI. For a given pixel *x* in the region, its three component values will be [0,*x*], [1,*x*] and [2,*x*]. This pixel will be referred to as pixel number *x* in the region. To construct the volume each pixel component must be scaled to lie between zero and the volume size and be converted to an integer value. This produces an array of 3D indices into the volume, one triple for each pixel in the pixel array, which is stored for later use if the user requests the intersection to be mapped to the image display. If a voxel in an intersection volume (the result of intersecting two or more volumes) has a value greater than zero then all pixels falling in that voxel are deemed to have caused that intersection. Pixel values causing intersection (*offenders*) can then be obtained by the following steps:

1) Use the voxel indices to search the scaled pixel value arrays (one array per volume intersected).

2) Retrieve the pixel numbers whose scaled indices match.

3) Retrieve the pixels values by indexing the original array using the pixel numbers

Having obtained the list of offenders the prototype then, based on the user's selection, either:

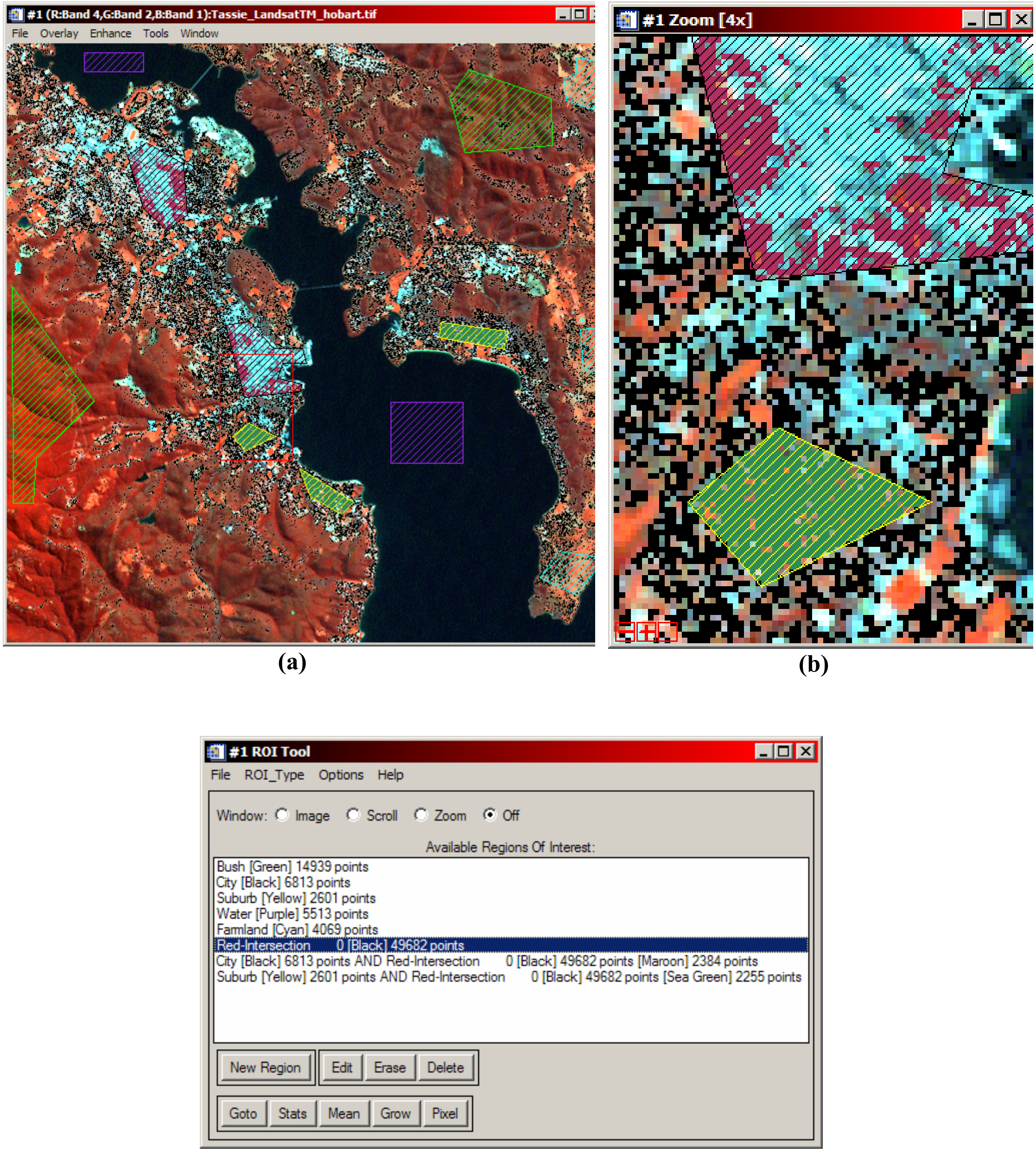
- Searches the entire image for pixels with values matching any of those in the list, or;
- Searches the regions involved in the intersection for such pixels.

Obviously searching only those regions of concern is vastly more efficient than searching every pixel in the image. The user may wish to use the entire image option to find all occurrences of such offenders. This image searching process is essentially that of Table Look Up Classification (Richards 1986, p. 186).

3.6.4 Displaying Offenders

Section 3.6.3 describes the process of finding the location of offenders. This yields a list of 2D coordinates (x,y) into the image. Pixels at these locations need to be highlighted as they represent uncertain areas in the user's region selection. The prototype does this by creating a new ROI containing the offending points. The prototype uses ENVI's built-in routines for creating ROIs from lists of points. This feature is implemented using these functions. The new ROI is assigned an arbitrary colour and unique descriptive label.

The user may then manipulate the new region like any other ENVI region; it may be merged or intersected (spatially) with others, statistics calculated and details exported to other formats. Figure 3-14 shows the result of applying this procedure to the spectral intersection of the City and Suburb regions as defined in Figure 3-11, followed by a full image search for offending pixels (highlighted in black). This new black region is then spatially intersected with each of the City and Suburb regions in turn producing two new regions coloured dark red and green. Pixels in these regions have values which cause the intersection of these regions in feature space. In this instance the conclusion could be drawn that pixel values in the Suburb region are largely a subset of the values in the City region.


#1 (R:Band 4,G:Band 2,B:Band 1):Tassie_LandsatTM_hobart.tif
File Overlay Enhance Tools Window
#1 Zoom [4x]
#1 ROI Tool
File ROI_Type Options Help
New Region
Edit
Erase
Delete
Goto
Stats
Mean
Grow
Pixel

Window: ○ Image ○ Scroll ○ Zoom ● Off

Available Regions Of Interest:

```
Bush [Green] 14939 points
City [Black] 6813 points
Suburb [Yellow] 2601 points
Water [Purple] 5513 points
Farmland [Cyan] 4069 points
Red-Intersection      0 [Black] 49682 points
City [Black] 6813 points AND Red-Intersection      0 [Black] 49682 points [Maroon] 2384 points
Suburb [Yellow] 2601 points AND Red-Intersection      0 [Black] 49682 points [Sea Green] 2255 points
```

(a)

(b)

(c)

Figure 3-14: (a) Highlighted conflict between regions. Pixels with values that cause intersection in feature space between the City (black) and Suburb (Yellow) classes are highlighted as black points. Spatial intersection is shown as dark red in the City region and green in the Suburb Regions. (b) Zoomed view of boxed area of the image showing spectral intersection between regions. (c) ROI details.

3.7 Configuring the Visualisation

The user configures and launches the visualisation prototype using a graphical user interface (GUI). The first configuration step is the selection of the 3 bands to use as axes in 3D feature space. This step is in the form of selecting three bands to use in a colour composite image display. The next step is to define ROIs to visualise in feature space. ENVI provides the image display and ROI management features. A menu item on the colour composite display launches the visualisation configuration GUI.

This visualisation configuration GUI allows users to select the regions they wish to visualise and the manner in which they visualise them (ellipsoids, isosurfaces, direct volume rendering, etc). It also allows the volume size parameter to be set. Depending on the choice of visualisation type the user may need to specify additional information, eg: isovalues, α values and colour tables for direct volume rendering. The user then clicks a button and the initial visualisation is presented. This visualisation can be rotated, zoomed and panned. The properties (colour, outline style, etc) can be changed from the *Visualisation Browser* window. This window is part of the iTools framework within which the prototype operates. The user may revise any of their decisions through the configuration GUI at any time.

3.8 Implementation of the Prototype

The prototype system is implemented in IDL using ENVI library functions for image display functionality and classification. The prototype makes use of the *iTools* (*Intellegent Tools*) framework. This means that new visualisations can be easily added; for example, new shapes for cluster representations. In addition to the *iTools* ***framework****, RSI* also supply a number of existing visualisation programs with the *iTools* package. One such program is *iVolume*, a tool for volume rendering and related operations on volume data. This was used a basis for the prototype, with features such as ellipsoids and α-shapes being incorporated as additional visualisation classes.

The code for the prototype can be found in Appendix B on the accompanying CD-ROM. Instructions for running the prototype under IDL 6.0 and ENVI 4.0 are included on the CD-ROM.

Chapter 4 Case Study

4.1 Introduction

This case study discusses the features of the prototype software and demonstrates its capabilities for:

- Presenting the opportunity for increased understanding of classification of satellite imagery,
- Communicating patterns in the image to the user,
- Assessing the quality of the user's assumptions and conveying possible revisions to those assumptions, and;
- Explaining the results of classification.

The study has been carried out using the ENVI environment within which the prototype operates. Several of ENVI's built-in supervised classifiers are used to classify an image. A k-Nearest Neighbour classifier implemented by Murray (in-press) is also used. ENVI's GLCM based texture measures are applied to produce texture bands which are used in a feature space visualisation. A Principal Components transform is carried out using ENVI. The first three derived principal components are used in a feature space visualisation. ENVI provides built-in basic statistics and plotting functions and these are used to plot statistics of regions highlighted by the prototype. All other VDM activities described are carried out by the visualisation prototype as described in Chapter 3.

4.2 Study Area

4.2.1 Paddick Valley, Heard Island, Australia

Heard Island is a sub-Antarctic island situated roughly 4100km south-west of Perth, Australia and 1500km north of the Antarctic continent. Remote, unpopulated, sub-Antarctic islands such as Heard are of particular interest as they are key indicators for climate change. Heard Island, along with the neighbouring MacDonald islands, is isolated by vast stretches of ocean. No permanent human settlement exists on Heard Island. These factors in combination with the island's highly distinct ecosystems make for a very interesting study area. Paddick Valley is a small valley, roughly 2 km long, on the south eastern side of the island and is the study area for this case study. Further details and maps are attached in Appendix A.

4.2.2 Dataset

Imagery

Imagery of the eastern side of the island was acquired by the IKONOS satellite (http://www.geoimage.com.au/) in January 2004. The IKONOS satellite provides images with four multi-spectral bands and a panchromatic band. Spectral bands have a spatial resolution of 4 m and the panchromatic band 1 m. The panchromatic band uses a very wide range of wavelengths (0.45-0.90 μm) to achieve its finer spatial resolution. The spectral bands are pansharpened using the panchromatic band, improving their spatial resolution to 1 m (Lucieer 2006). The resulting image to be analysed contains 4 bands with 1 m spatial resolution representing: blue, green and red visible light, and near-infrared (Table 4-1).

Table 4-1: Multi-spectral wavelengths recorded per band by the IKONOS satellite.

Band	Wavelength (μm)
Blue	0.445-0.516
Green	0.506-0.595
Red	0.632-0.698
Near-IR	0.757-0.853

A subset of the image containing Paddick Valley was extracted. This image was then geometrically rectified using differential GPS coordinates and a rubber-sheeting transformation (Lucieer 2006). The visible light bands of the image are shown as a colour composite in Appendix A. The image origin is at 73°31′51″ S, 58°8′13″ E.

Regions of Interest

Regions of interest have been defined within the image representing different land cover classes. The definition of these regions is based on ground truth from a site survey conducted by a team from the University of Wollongong (Brandner 2005). The survey team examined ten 1 m^2 plots within several 30 m diameter areas. The centres of the areas were recorded and vegetation classes associated with each area based on the ten plots examined. Using this field data (Lucieer 2006) constructed disc regions representing the centre of the area and a 30 m buffer zone about that point. Three sets of regions were created each with a higher degree of detail regarding vegetation type. These are summarised in Table 4-2.

Table 4-2: The three classification levels.

Region Set	Non-vegetation	Vegetation
1c	Rock Water Wave	Vegetation
3c	Rock Water Wave	Dense Medium Sparse
6c	Rock Water Wave	FellField Tussock ClosedCushion OpenCushion MossField HerbField

From the 30 m discs a random sample of 200 points were extracted for each class. These comprise the training regions for each class, denoted as `Xc_training` where *X* is the detail level for the vegetation regions (1,3 or 6). A further 200 independently randomly sampled points were extracted for each class for validation purposes. These comprise the testing regions for each class, denoted as `Xc_testing`. The region sets comprising the entire 30 m discs are denoted as `Xc_all`.

4.2.3 Statistics

For reference, basic statistics for the image are in Table 4-3 and plotted in graphical form in Figure 4-1.

Table 4-3: Statistics for Paddick Valley image.

```
Filename: E:\HeardData\iko_jan04_ps_rsrc_paddick.bsq
Dims: Full Band (3043501 points)

Band          Min          Max         Mean        Stdev
   1    50.000000  1594.000000   361.020952    62.640973
   2     5.000000  1924.000000   342.313533    76.998076
   3     1.000000  1649.000000   217.338314    68.547503
   4     7.000000  1479.000000   269.702219   244.077425
```

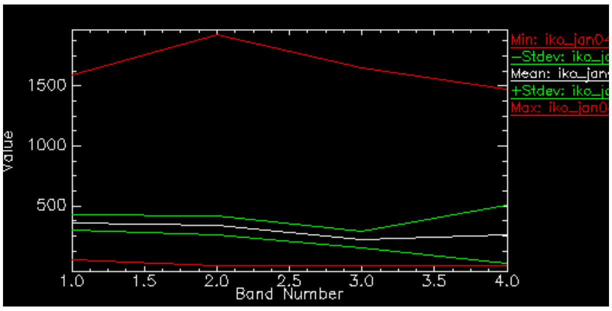


Figure 4-1: Image band statistics of the study area

4.3 *Simple Spectral Classification*

4.3.1 Overview

This section describes a simplified classification scenario. Only 4 classes are defined, Vegetation, Water, Rock and Wave. The *1c-all* ROI set will be used to examine the relationships between regions. The visualisation prototype is used prior to classification to explore these relationships. Assumptions such as band selection are revised according to the output of the visualisation prototype. The *1c-training* ROI set is then used to classify the image with various classifiers. The overall process involves several iterations of: assumptions being formed or revised, the effects visualised, and results discussed.

4.3.2 Iteration 1

Process

An ENVI RGB display of bands 4, 3 and 2 is loaded and the *1c-all* ROI set is overlayed on to the display (Figure 4-2). This choice of bands is based on Figure 4-1; band 4 has the largest variance and would yield the most information, bands 1, 2 and 3 all have similar standard deviation properties. The choice between these bands is largely arbitrary except for the larger overall range of value in band 2. Visualisations and classifications will be derived from these 3 bands alone. Visualisations will assign the red display band to the X axis, green to Y (deeper into the scene) and blue to Z (higher in the scene).

The visualisation prototype is invoked and configured to use a volume size of 64 and display all four ROI clusters as isosurfaces in feature space, coloured accordingly (Figure 4-3). The wave region has high reflectance in all bands. This distorts feature space in order to accommodate the cluster. The wave region is of little interest in this process and is disregarded in further visual analyses. This produces a more satisfactory feature space rendering, as shown in Figure 4-4.

Visual inspection of 3D feature space from Figure 4-4 shows some potential interaction between the Rock and Water ROIs highlighted in Figure 4-5. The visualisation is reconfigured to display any intersection between these regions as a new isosurface. Figure 4-6 shows the result of this operation. From visual inspection in geographic space this is not what would be expected.

Whilst both water and rock would be expected to share low reflectance values in band 4 (infrared) and relatively low reflectance in other visible light bands, they would still be discernable, especially on sensitive equipment like IKONOS's sensors. If plotted in feature space, individually, as shown in Figure 4-7, there is no intersection as feature space is not distorted by the presence of other classes. However, this intersection in distorted feature space and close proximity otherwise still has implications for classification.

One feature of the prototype is the mapping of pixel values causing overlap in feature space to points in geographic space with those values. Figure 4-8 shows the result of this operation on the intersection between the Rock and Water regions (Figure 4-6). The whole image is searched for pixels with values causing intersection between these regions. These locations of these pixels are used to create a new ROI, shown in orange. There are 146 points in this new ROI. All pixels in this region have values which cause them to fall in the highlighted region of feature space in Figure 4-6 (see also Figure 4-7). This is reflected in the following statistics table for the orange region:

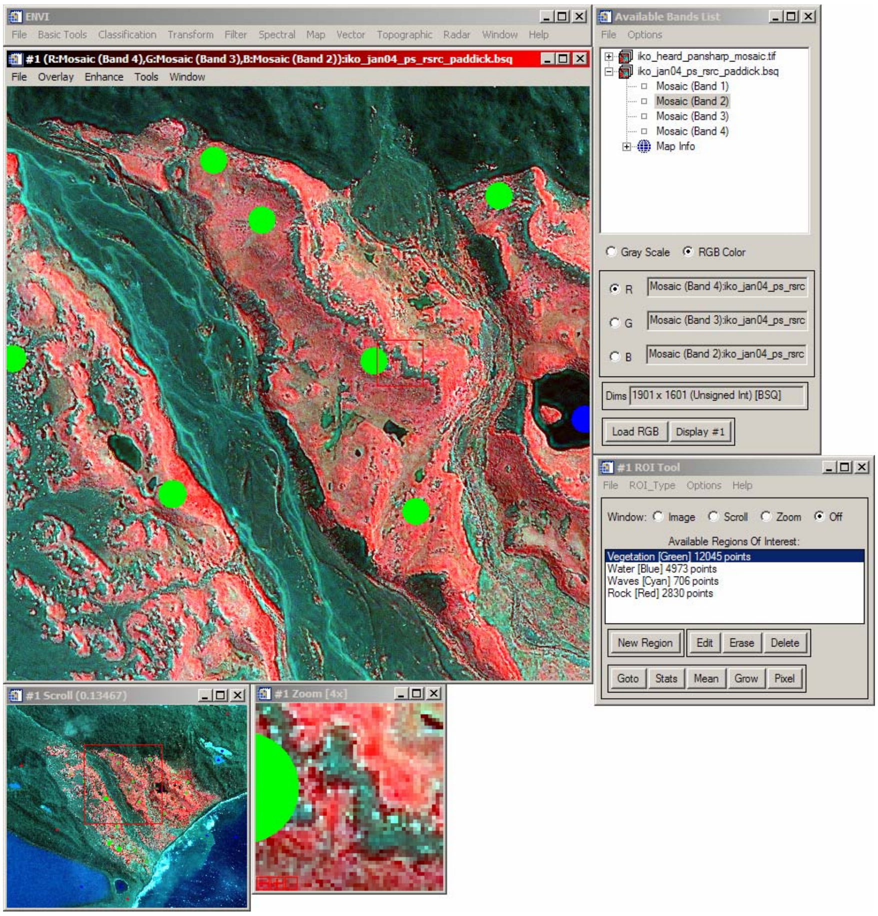
Table 4-4: Statistics for orange region shown in Figure 4-8.

```
Filename: E:\HeardData\iko_jan04_ps_rsrc_paddick.bsq
Region: Red-Intersection       0 [Orange1] 146 points

Band         Min         Max        Mean       Stdev
   1  298.000000  320.000000  308.671233    4.915997
   2  260.000000  268.000000  264.472603    2.729646
   3  153.000000  160.000000  157.047945    2.696952
   4  118.000000  136.000000  125.609589    5.490271
```

Table 4-4 shows the effect the inclusion of the vegetation (green) region in the feature space plot has on the intersection. This region increases the range of reflectance values in band 4 (infrared). In feature space this means more pixels from the Water and Rock regions, which have a uniformly lower reflectance in band 4 occur within a small area of feature space. Figure 4-10a plots the data in Table 4-4 graphically. Compare this graph with that in Figure 4-10b. Perhaps the initial choice of bands has contributed somewhat to this intersection in feature space. For the orange region, comprised of pixels with overlapping values in feature space, very little variance occurs in two of the bands plotted as axes in feature space: bands 2 and 3. Band 1, however has variance near that of band 4. Using this band as one of our axes in feature space may minimise this overlap.

For later comparison a spatial intersection is performed between the orange overlap region with each of the Water and Rock regions. This results in two new regions shown in Figure 4-9. These regions contain the actual pixels responsible for the intersection in feature space. In the Water intersection (Figure 4-9a) contains only a single pixel whilst the Rock intersection (subset in Figure 4-9b) contains 16 pixels. These 16 pixels overlap in feature space with just a single offending pixel from the Water region.



#1 (R:Mosaic (Band 4),G:Mosaic (Band 3),B:Mosaic (Band 2)):iko_jan04_ps_rsrc_paddick.bsq

iko_heard_pansharp_mosaic.tif
iko_jan04_ps_rsrc_paddick.bsq
- Mosaic (Band 1)
- Mosaic (Band 2)
- Mosaic (Band 3)
- Mosaic (Band 4)
- Map Info

○ Gray Scale ● RGB Color

● R Mosaic (Band 4):iko_jan04_ps_rsrc
○ G Mosaic (Band 3):iko_jan04_ps_rsrc
○ B Mosaic (Band 2):iko_jan04_ps_rsrc

Dims 1901 x 1601 (Unsigned Int) [BSQ]

Window: ○ Image ○ Scroll ○ Zoom ● Off

Available Regions Of Interest:
Vegetation [Green] 12045 points
Water [Blue] 4973 points
Waves [Cyan] 706 points
Rock [Red] 2830 points

Figure 4-2: ENVI user interface with false colour composite display of Paddick Valley overlayed with 1c-all regions.

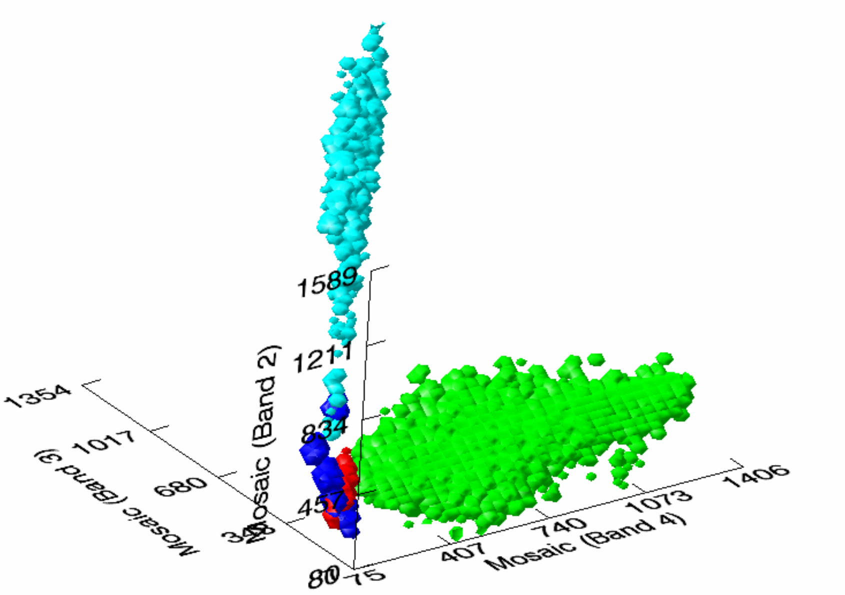


Figure 4-3: Isosurfaces for all four ROIs in the 1c-all ROI set with isovalues of 0.1. The light blue surface represents the wave ROI which has high reflectance in all bands.

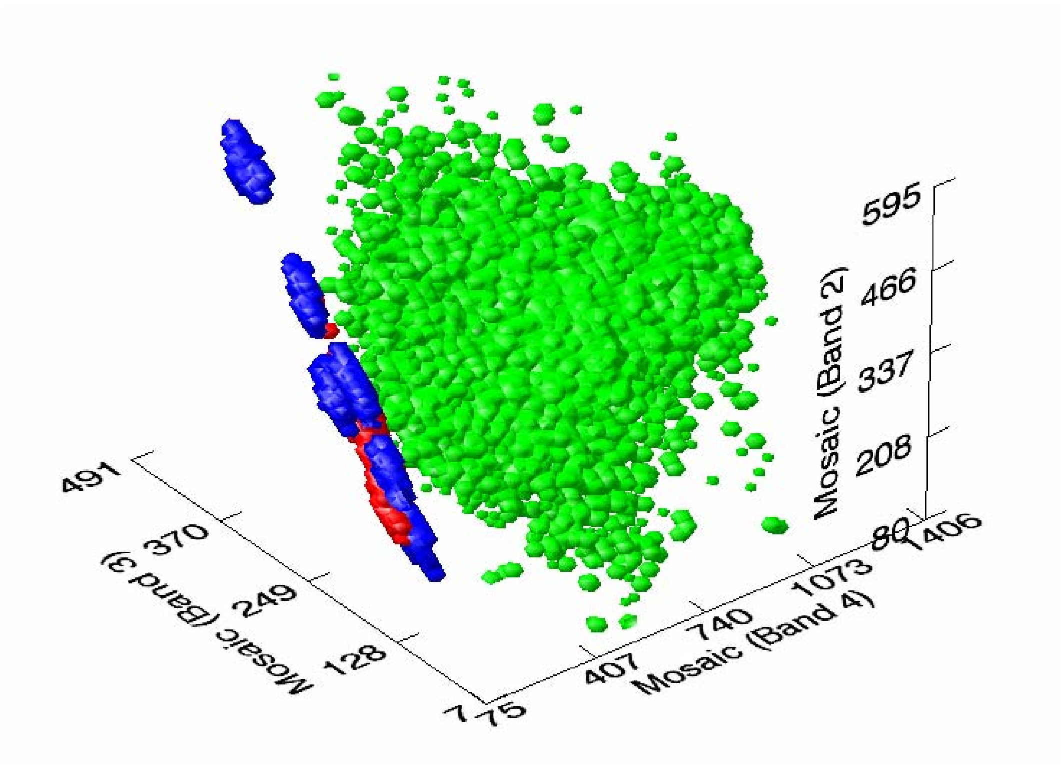


Figure 4-4: Isosurfaces for three ROIs in the 1c-all ROI set with isovalues of 0.1

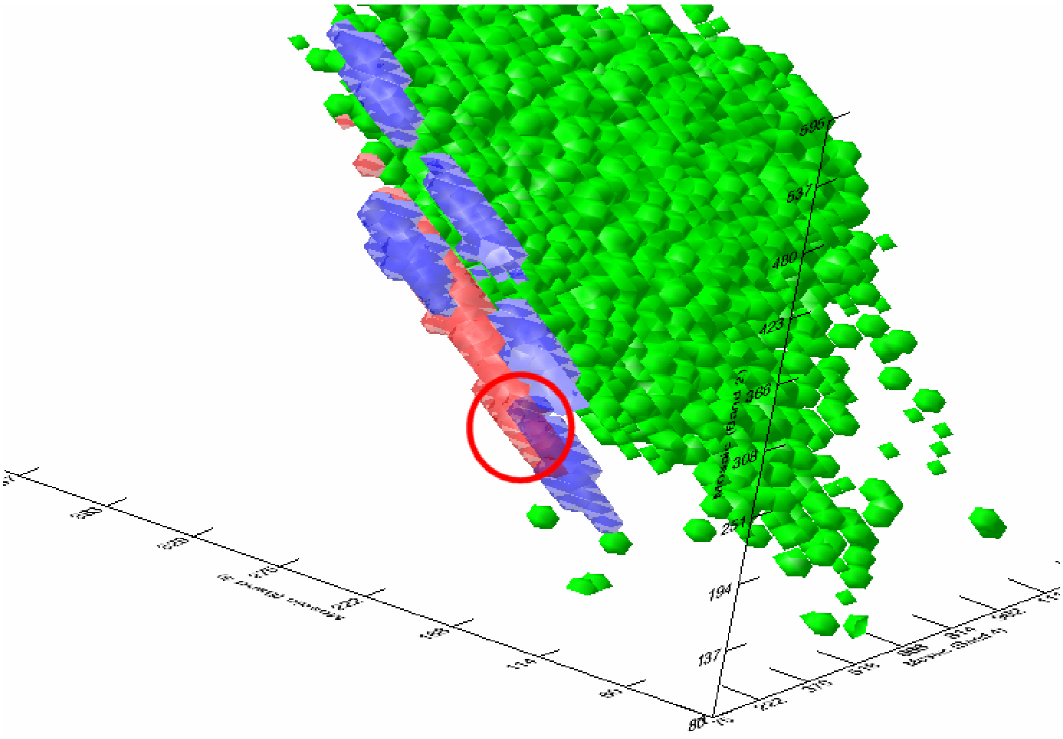


Figure 4-5: Potential intersection in feature space between the Rock and Water ROIs.

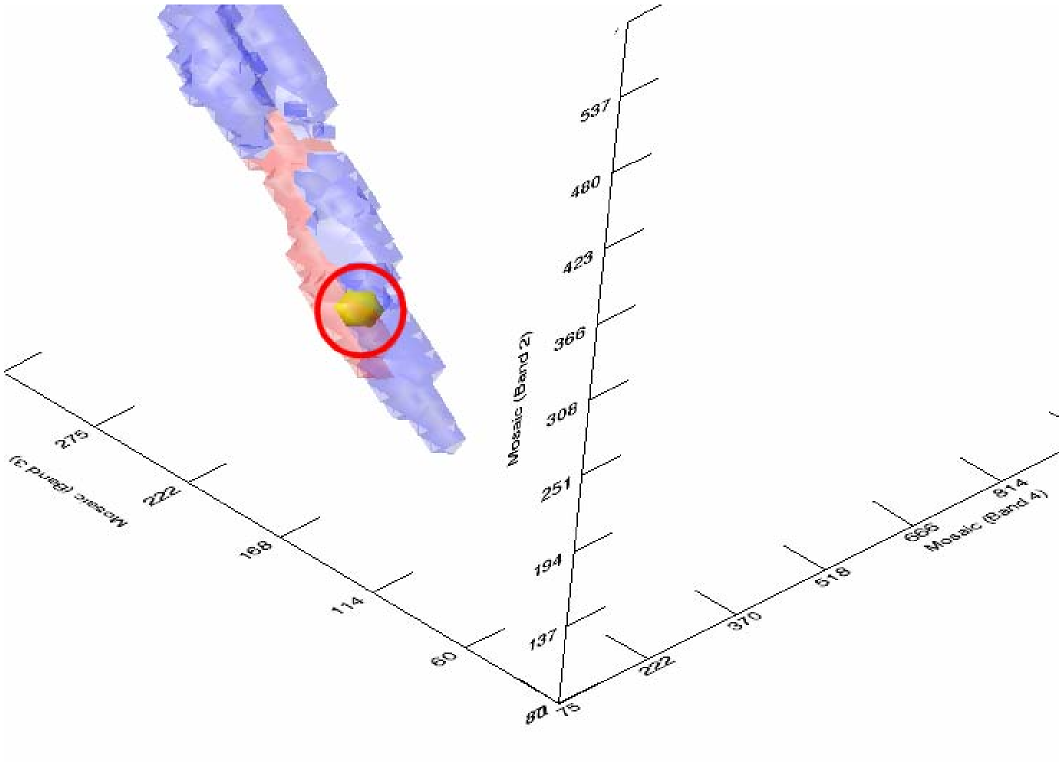


Figure 4-6: Isosurface (yellow) representing the intersection between Rock and Water ROIs in feature space. The Vegetation ROI (Green) has been hidden.

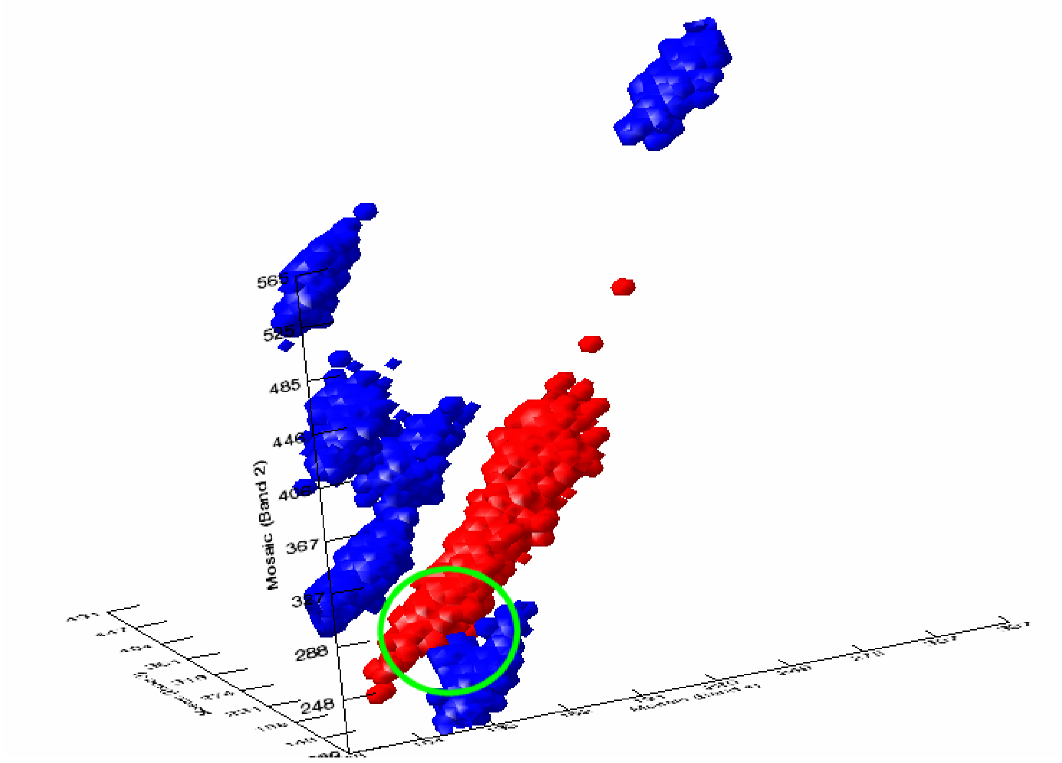


Figure 4-7: Feature space plot showing no intersection between ROIs. The portion of the volumes causing intersection in Figure 4-6 is highlighted in green.

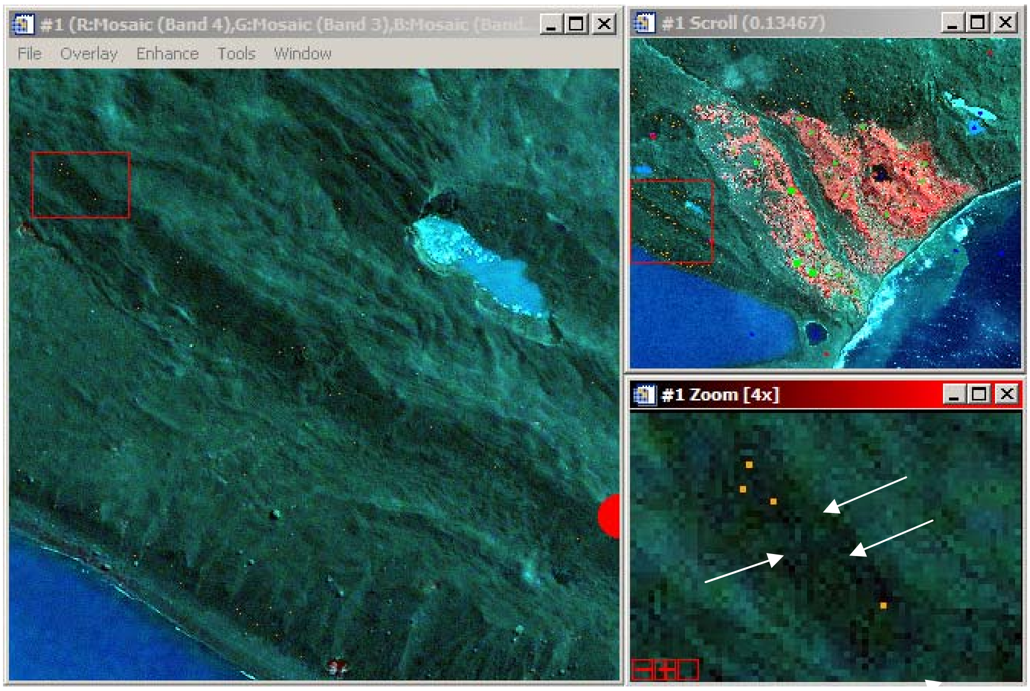
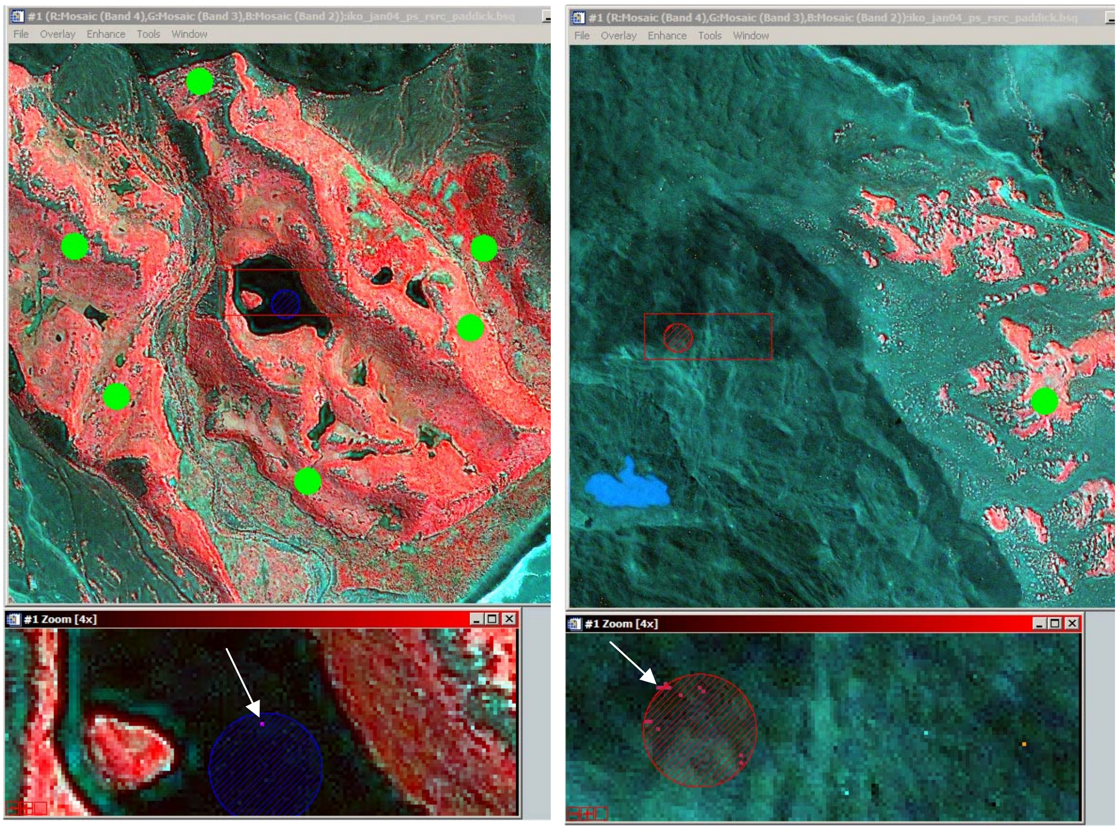


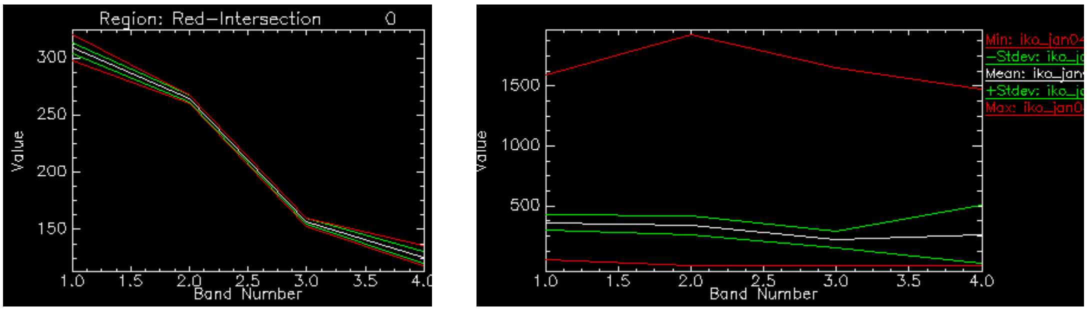
Figure 4-8: Overlayed false colour composite of Paddick Valley demonstrating highlighting of conflicting pixel values. Orange points show pixels with values matching those causing intersection between the Rock (Red) and Water (Blue) regions.


#1 (R:Mosaic (Band 4),G:Mosaic (Band 3),B:Mosaic (Band 2)):iko_jan04_ps_rsrc_paddick.bsq
File
Overlay
Enhance
Tools
Window
#1 Zoom [4x]
#1 (R:Mosaic (Band 4),G:Mosaic (Band 3),B:Mosaic (Band 2)):iko_jan04_ps_rsrc_paddick.bsq
File
Overlay
Enhance
Tools
Window
#1 Zoom [4x]

(a)

(b)

Figure 4-9: (a) Maroon point in Water representing pixel with values causing overlap in feature space with pixels highlighted in (b) (subset shown) as magenta points in Rock.



(a) (b)

Figure 4-10: (a) Average spectrum plot of orange region showing higher variance in bands 4 and 1. (b) Average spectrum plot for Paddick Valley image (Figure 4-1).

Summary

In this iteration the prototype has:

- Highlighted a possible flaw in the assumption used during band selection for classification,
- Revealed conflict between regions and the pixels causing that conflict, and;
- Showcased the effects of distorted feature space in visual data mining.

The next iteration will revise the band selection assumption, repeat the visualisation and compare the results of several classifications. Visualisation will be used to explain any misclassification.

4.3.3 Iteration 2

Process

The visualisation prototype is reconfigured to use bands 4, 2 and 1. The intersection between Rock and Water is recalculated and visualised in feature space (Figure 4-11). This overlap is mapped back into geographic space. The prototype creates a new region containing 25 points in the image with pixel values matching those causing intersection. This region is again intersected with each of the Water and Rock regions producing regions shown in Figure 4-12. The new band selection produces a slightly larger, but less concentrated overlap in feature space. This translates to a more even ratio of pixels from each region and fewer pixels overall causing intersection.

Classification is now carried out using the minimum distance to mean, maximum likelihood and k-nearest neighbour classifiers. For each of these classifiers a representation of the decision surface or parameters can be shown in feature space. Classifiers are trained on 200 randomly sampled pixels for each region (*1c-training* regions). Testing of classifier accuracy is carried out using a further 200 independently, randomly sampled pixels for each region (*1c-testing* regions). Visualisation will continue to use all of the regions' pixels to construct shapes (*1c-all* regions). Where possible a classification will be forced for each pixel; pixels will not be labelled as unclassified. For the maximum likelihood classifier this means there is no probability threshold set. For the Euclidean distance based minimum distance and k-nearest neighbour classifier this means there is no maximum distance threshold from a mean or neighbour at which classification cannot occur. Thematic maps for the entire image produced by these classifiers are shown in Appendix A.

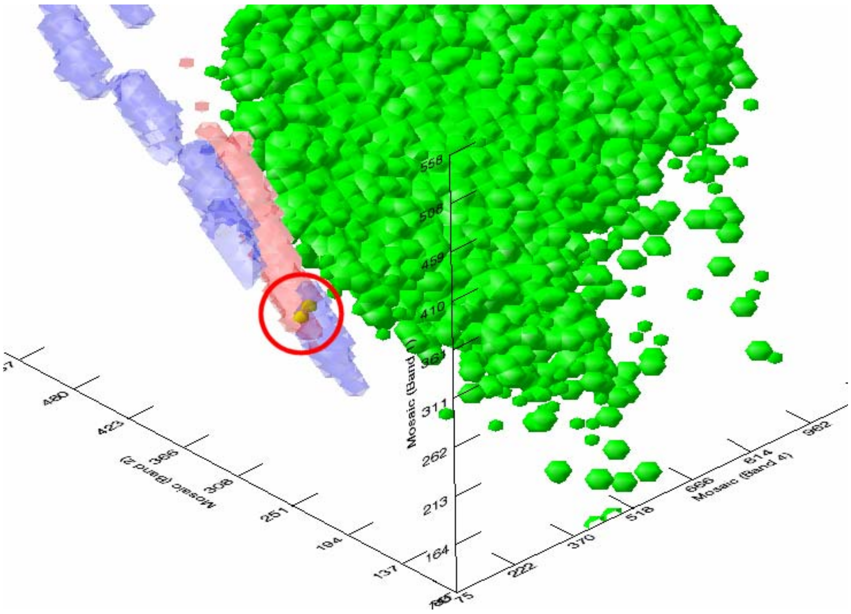
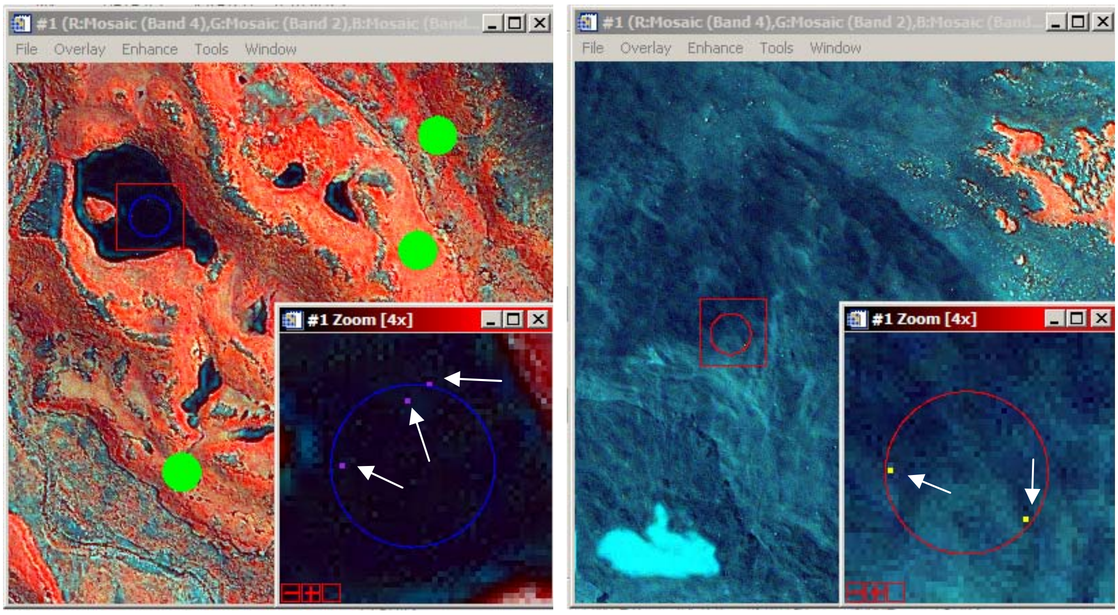


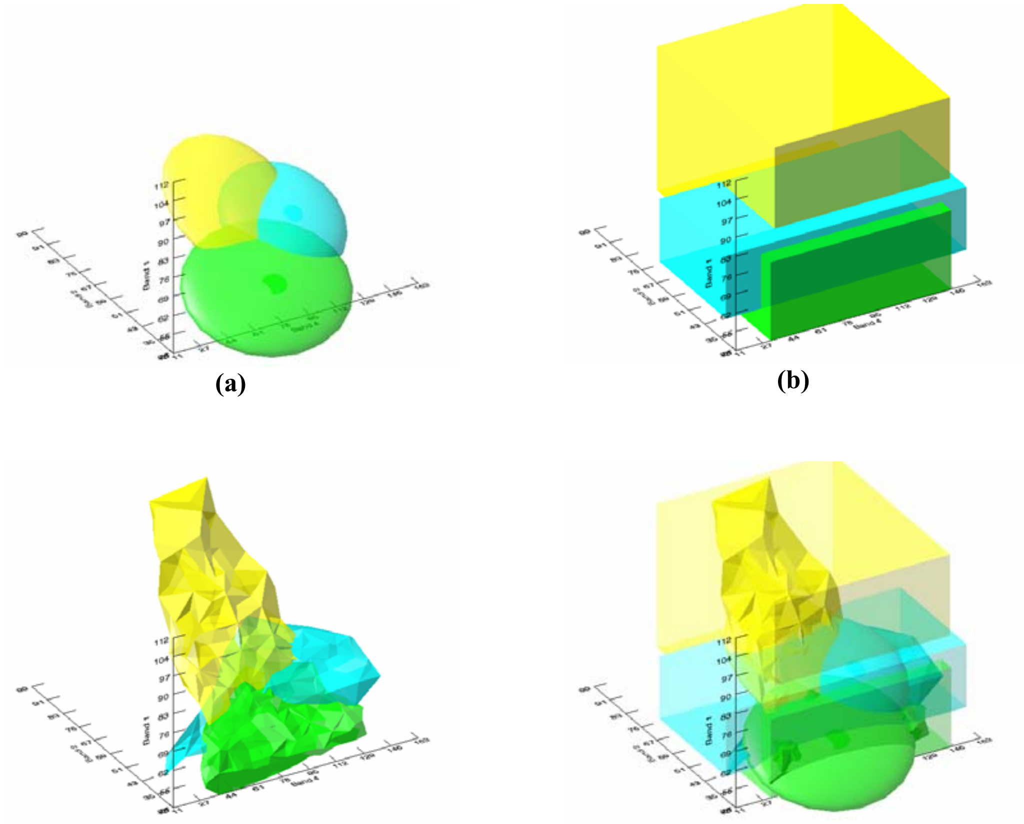
Figure 4-11: Intersection between Rock and Water based on bands 4, 2 and 1. Note now that the intersection forms two blobs in space (when rendered as an isosurface) implying two voxels of overlap.


#1 (R:Mosaic (Band 4),G:Mosaic (Band 2),B:Mosaic (Band...
File
Overlay
Enhance
Tools
Window
#1 Zoom [4x]
#1 (R:Mosaic (Band 4),G:Mosaic (Band 2),B:Mosaic (Band...
File
Overlay
Enhance
Tools
Window
#1 Zoom [4x]

(a) **(b)**

Figure 4-12: (a) Pixels causing overlap in the Water region; (b) Pixels causing overlap in the Rock region.

The prototype has the ability to visualise decision boundaries and parameters. Figure 4-13 presents a comparison of the decision surfaces used by several classifiers described in section 2.3. The minimum distance classifier can be represented by Figure 4-13a; the centre of the ellipsoids is the class mean. The k-nearest neighbour classifier is represented by an α-shape in Figure 4-13c, this is not a precise visualisation of the way the classifier works, it merely serves to highlight the true maximum extent of a class' training data in feature space.



(a) (b) (c) (d)

Figure 4-13: Comparison of decision surfaces. (a) maximum likelihood, (b) level-slice, (c) k-nearest neighbour (representative only of extent), (d) all.

Figure 4-14 shows the mean points in feature space for each region. The mean is the only property of the data used by the minimum distance classifier. Pixels are classified according to the closest mean point in feature space. Given this, it can be seen that some misclassification may be expected to occur between the Water and Rock classes. In fact, we can expect misclassification between all classes. This is reflected in Figure 4-15 and Figure 4-16. The lake in the centre left of the image is almost completely misclassified as water. Of all figures presented thus far, Figure 4-7 most clearly highlights the nature of the Water and Rock regions if used for classifier training. When plotted in feature space the Water region clearly consists of five distinct clusters. Pixels lying in the cluster closest to the origin of feature space and those toward the lower end of the next nearest cluster will be misclassified as rock due to the Rock class' mean lying closer to this cluster than the true Water class mean. This is visualised in Figure 4-17 in which the ellipse region component from the Water region shown in Figure 4-12a has been isolated using ENVI's ROI manipulation tools and shown as a mauve isosurface.

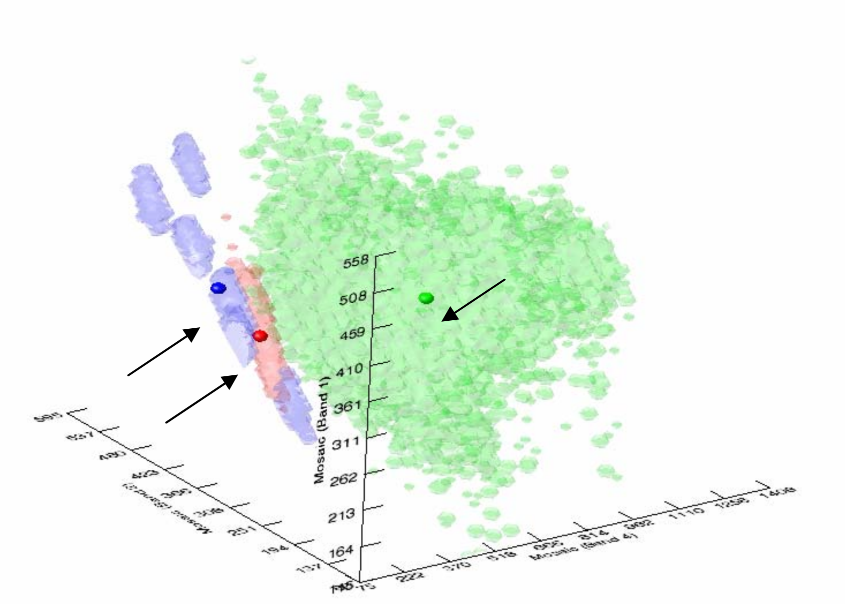


Figure 4-14: Mean points for each region in feature space shown, as small spheres. Note the close proximity of means for the Rock and Water classes. Isosurfaces are transparently rendered indicating the true extent of each region.

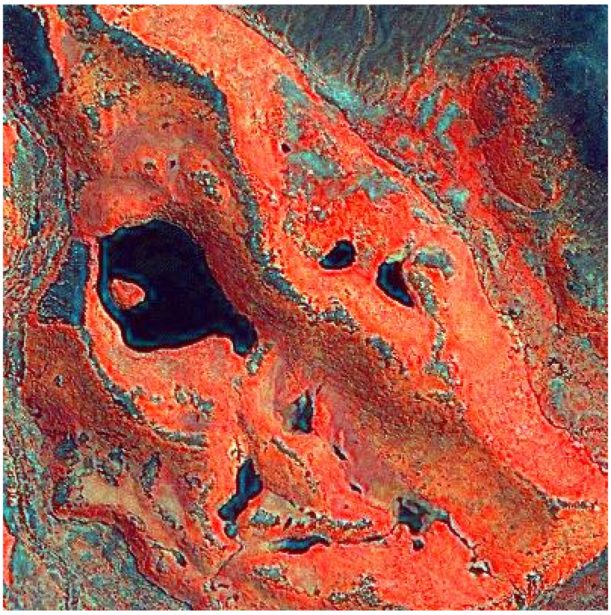


Figure 4-15: False colour composite showing bands 4, 2 and 1 of a subset of the Paddick Valley image.

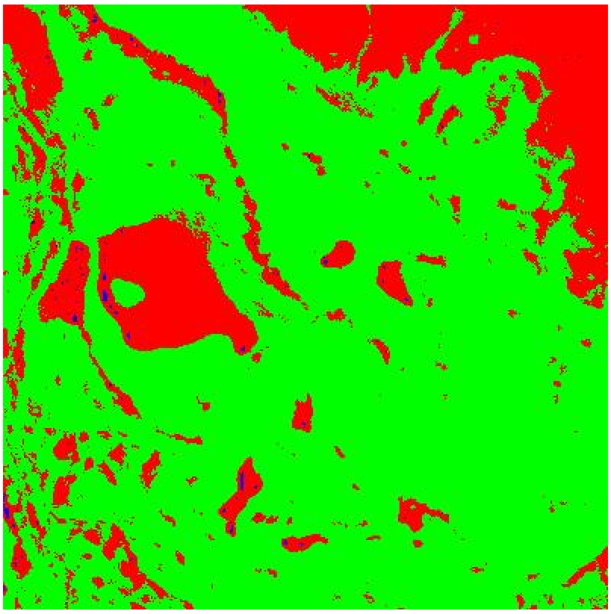


Figure 4-16: Thematic image showing the result of a minimum distance classification of the image subset in Figure 4-15. Red pixels have been classified as Rock, blue as Water and green as Vegetation.

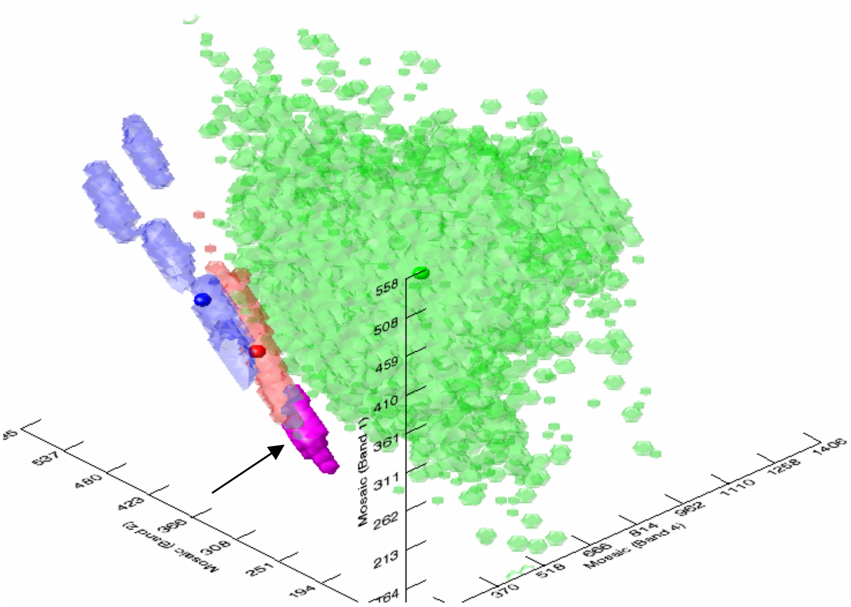


Figure 4-17: Means and extent for each region. Pixel values from the ellipse in Figure 4-12a are shown as a purple isosurface.

Figure 4-18 shows ellipsoids representing the statistical distribution of pixels in the regions. Configured appropriately, these ellipsoids can represent the decision boundaries of a maximum likelihood classifier. The closer a pixel lies toward the centre of a class ellipsoid relative the boundary of the ellipsoid the higher the likelihood that it belongs to that class. A pixel on the exact edge of two ellipsoids would have exactly equal likelihoods for each class. In this way a pixel has multiple likelihoods associated with it. It is classified according to the maximum of these likelihoods. Figure 4-18 shows that the Rock and Water classes have much potential for misclassification. The distinct clusters within the Water region cause a high variance which the classifier takes into account. This causes the decision boundary for the Water class to almost completely engulf the Rock class. In this way, pixels falling within the Rock class, but on the left side of the Rock ellipsoid may in fact have a higher likelihood for the Water class. This is reflected in Figure 4-19; the lake is correctly classified but much of the rock in the top right of the subset image is incorrectly labelled as water.

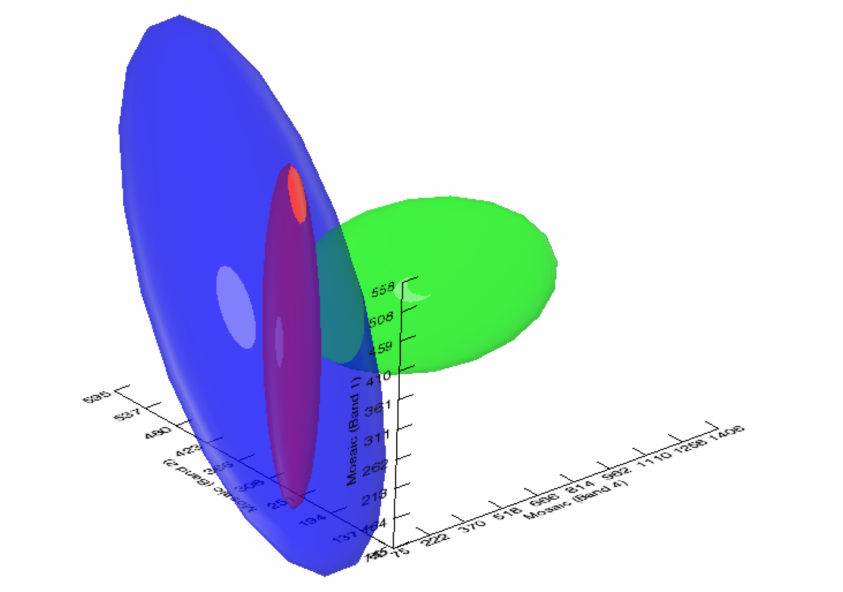


Figure 4-18: Ellipsoids representing the decision boundary of a maximum likelihood classifier. Note that the disjoint clusters in the Water class cause a large amount of variance for that class.

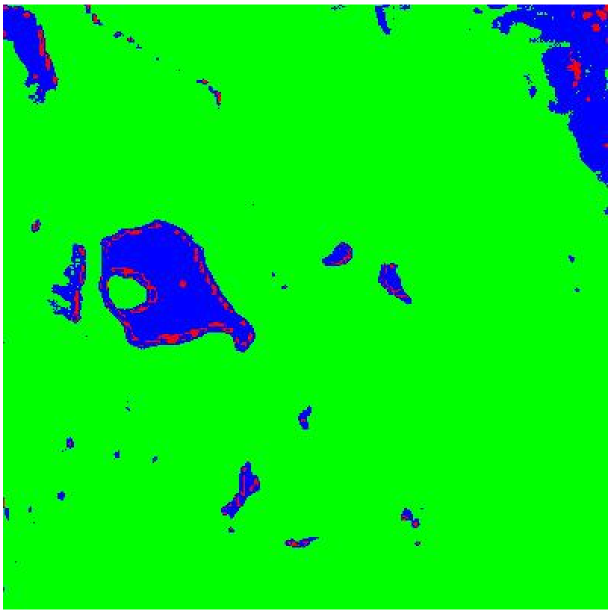


Figure 4-19: Thematic image showing the result of a maximum likelihood classification of the image subset in Figure 4-15. Red pixels have been classified as Rock, blue as Water and green as Vegetation.

Visualising a decision boundary for the k-nearest neighbour classifier is more complex than the preceding classifiers. The boundary will vary depending on the value of k selected. The users own perception will need to be employed to make these adjustments. The regions can be displayed as isosurfaces, α-shapes, or simple point clouds. Point clouds make it difficult to distinguish the extent of one region from another so are not useful in this instance. Figure 4-20 compares the α-shapes and isosurfaces for the regions being examined. Isosurfaces provide some visual representation of density of occurrence which is important for k values greater than 1.The α-shape representation however provides a clear cut decision boundary. Figure 4-20 provides a visualisation of the factors influencing a k-nearest neighbour classification.

Classification is performed with a k value of 7 with no maximum distance threshold. The result of this classification can be seen in Figure 4-21. This classifier seems to provide the most acceptable result with a balance being struck between the Water and Rock classifications.

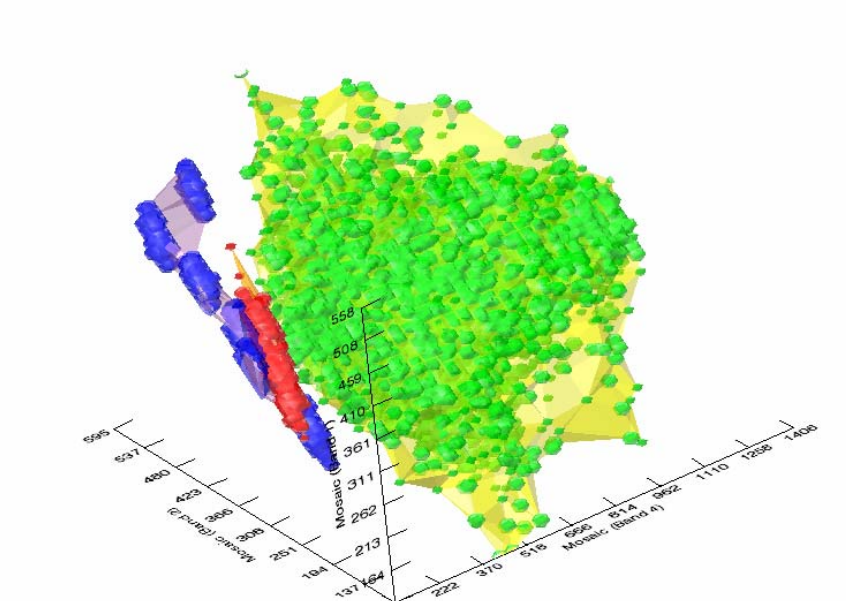


Figure 4-20: Comparison of α-shapes (α computed for joined shapes) and isosurfaces (isovalue of 0.1) for the regions. Yellow, orange and purple shapes are α-shapes. Green, red and blue are isosurfaces. The shapes represent Vegetation, Rock and Water respectively.

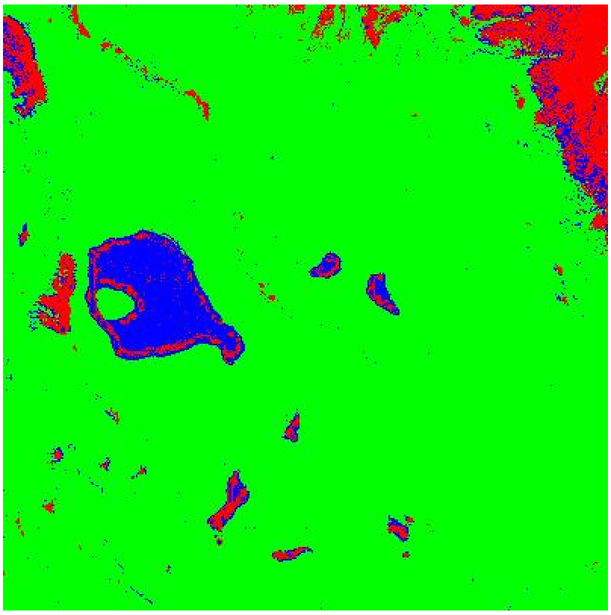


Figure 4-21: Thematic image showing the result of a k-nearest neighbour classification of the image subset in Figure 4-15 with a k value of 7. Red pixels have been classified as Rock, blue as Water and green as Vegetation.

An important step in any classification based analysis is accuracy assessment. In the above example we have assessed the classifier's accuracy based purely on visual inspection of the thematic images. Table 4-5 presents the error matrices obtained when classifying pixels from randomly sampled (exclusive of training pixels) testing regions for each class (*1c-testing* regions). Overall accuracies reported by the maximum likelihood and k-nearest neighbour classifiers are excellent. From the confusion matrices alone it would seem maximum likelihood performs almost as well as the k-nearest neighbour. However, there is a vast difference between the thematic maps produced by these algorithms. As the prototype was used to visualise these regions in feature space, the results in the error matrices can be explained and easily understood.

Misclassification occurred for the minimum distance classifier due to the close proximity of class means to the cluster boundaries of other classes. This is reflected in the error matrix for this classification. There is a significant amount of misclassification between all classes in the testing set. The maximum likelihood classification yielded an error matrix which seems close to perfect, however inspection of the feature space plot and thematic map produced shows otherwise. These qualitative measures suggest revision of training regions may be necessary if this classifier is to be used. Both qualitative (feature space, thematic map) and quantitative accuracy assessments of the k-nearest neighbour classification show that of the three classifiers it provides the most acceptable results.

Table 4-5: Error matrices for each classification. Key: MD = Minimum Distance, ML = Maximum Likelihood, 7-NN = k-nearest neighbour using 7 nearest neighbours.

MD

```
Overall Accuracy = (670/800)  83.7500%
Kappa Coefficient = 0.7833

                 Ground Truth (Percent)
    Class       Vegetation        Water        Waves         Rock        Total
 Unclassified         0.00         0.00         0.00         0.00         0.00
Vegetation [G        89.00         0.00         0.00         0.00        22.25
Water [Blue]          0.00        56.00         1.50         8.50        16.50
Waves [Cyan]          0.00         0.00        98.50         0.00        24.63
Rock [Red] 20        11.00        44.00         0.00        91.50        36.63
        Total       100.00       100.00       100.00       100.00       100.00
```

ML

```
Overall Accuracy = (795/800)  99.3750%
Kappa Coefficient = 0.9917

                 Ground Truth (Percent)
    Class       Vegetation        Water        Waves         Rock        Total
 Unclassified         0.00         0.00         0.00         0.00         0.00
Vegetation [G        99.50         0.00         0.00         0.00        24.88
Water [Blue]          0.50        99.50         0.50         1.00        25.38
Waves [Cyan]          0.00         0.00        99.50         0.00        24.88
Rock [Red] 20         0.00         0.50         0.00        99.00        24.88
        Total       100.00       100.00       100.00       100.00       100.00
```

7-NN

```
Overall Accuracy = (796/800)  99.5000%
Kappa Coefficient = 0.9933

                 Ground Truth (Percent)
    Class       Vegetation        Water        Waves         Rock        Total
 Unclassified         0.00         0.00         0.00         0.00         0.00
Vegetation [G        99.50         0.00         0.00         0.00        24.88
Water [Blue]          0.50        99.50         0.00         1.00        25.25
Waves [Cyan]          0.00         0.00       100.00         0.00        25.00
Rock [Red] 20         0.00         0.50         0.00        99.00        24.88
        Total       100.00       100.00       100.00       100.00       100.00
```

Summary

In this iteration the prototype has, in general:

- Shown decision parameters in feature space for several classifiers,
- Revealed the cause of misclassification and;
- Revealed the true implications of overall accuracy figures.

More specifically it has provided precise reasons for misclassification between two of the defined classes. Visualisation of feature space assisted in determining the pixel values that would cause misclassification and how these values change considerably for different classifiers.

4.4 Vegetation Classification

Thus far a relatively simple analysis has been carried out. A more in-depth analysis of the Heard Island imagery is the classification of different vegetation types (Murray in-press). However, this is complex problem as illustrated in Figure 4-22.

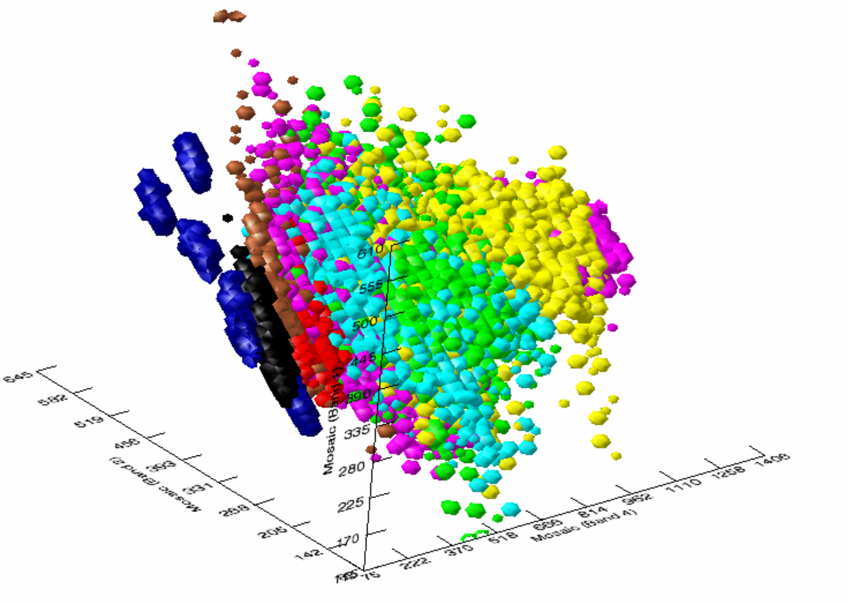


Figure 4-22: Feature space plot showing extent of regions for a six vegetation class analysis.

The full visual data mining of such a problem would be difficult to document as copious amounts of user interaction would be necessary. This is not to say that VDM applies only to simple problems. The constraint exists purely in the form of reporting operations performed.

4.5 Visualising Texture

Texture has been shown to be useful in classification of vegetation on Heard Island (Murray in-press). Texture bands can be produced using grey level co-occurrence matrix measures (see section 2.4). The prototype can be employed to visualise regions using texture – optionally combined with spectral – bands as axes in feature space. ENVI allows texture bands to be displayed as false colour composites. It is difficult to predict based on this image the effect these texture bands will have on classification. A feature space plot will make visually apparent the implication of using these bands with certain classifiers.

Murray (in-press) found that of the grey level co-occurrence measures the three most useful measures for classification were mean, dissimilarity and entropy. A feature space plot based on these bands and the *6c-all* ROI set is shown in Figure 4-23.

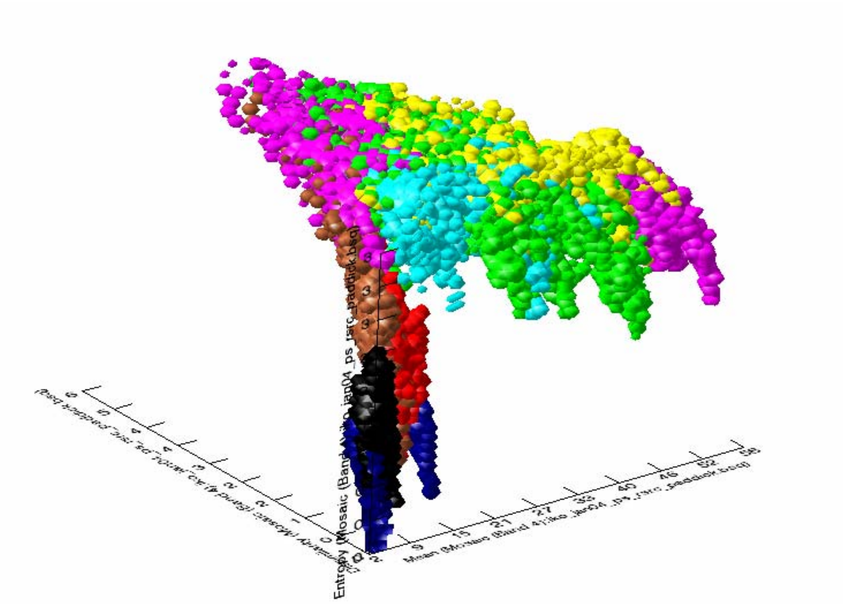


Figure 4-23: Feature space plot showing the effect of texture on classification of vegetation types.

4.6 Visualising Principal Components Bands

Principal Component (PC) bands may be derived in ENVI. As is the case with bands derived from texture, the prototype can be employed to visualise the results of Principal Components Analysis (PCA) (Richards 1986, p. 214). Figure 4-25 shows a feature space plot based on 3 PC bands displayed as a false colour composite in Figure 4-24. The interesting point to draw from Figure 4-25a is that is it visually similar to the initial plot in Figure 4-25b. Greatly simplified, the aim carrying out PCA on an image is to remove correlation between bands by generating a new uncorrelated set of bands. This is effectively a rotation of feature space (Richards 1986, p. 134), as shown by comparing Figure 4-25a and Figure 4-25b. The fact that the PCA produces a plot with much the same structure as the selected spectral bands indicates low correlation between the initial selection of bands. This has positive effects for classifier efficiency and accuracy.

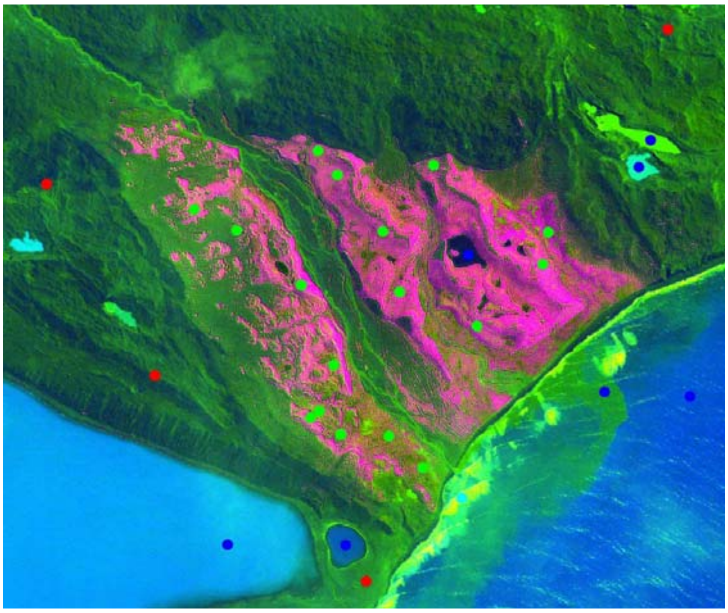
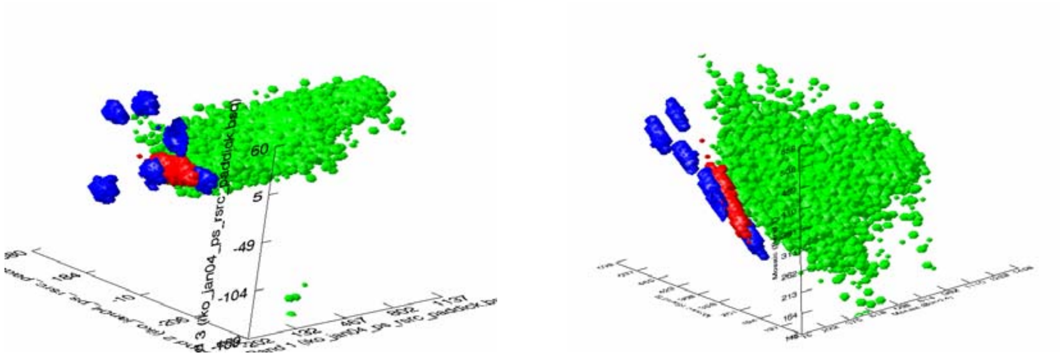


Figure 4-24: False colour composite of three derived principal component bands. PCA carried out on all four bands of the image.



(a) (b)

Figure 4-25: (a) Feature space plot based on 3 generated principal component bands, (b) Feature space plot based on bands 4, 2 and 1 of the image.

Chapter 5 Discussion

Chapter 3 discussed issues regarding volume representations of feature space. This representation is advantageous as performance and quality can be traded according to the user's needs. The disadvantage comes with the need for the user to understand the implications of their choice of volume size. The interpretations of their visualisations will need to take this variable into account.

This case study in Chapter 4 presented a usage scenario for the visualisation prototype described in Chapter 3. The prototype was used to perform a visual data mining analysis of high-resolution IKONOS imagery of Paddick Valley, Heard Island. The visual analysis was combined with non-visual classifications and used to explain the underlying algorithms and results of those classifications.

Interactive visualisation of training region overlap has allowed the discovery of local patterns in the data that would otherwise not be immediately apparent. Visualisation of this conflict allowed explanation of classification results and revision of classifier parameters such as band selection. The link between geographic and feature spaces can be used to highlight uncertainty in training regions. The ability to highlight and manipulate these uncertain pixels as a region of interest allowed investigation of the spatial and statistical properties of these pixels as a single group. This is advantageous as overlap in feature space can correspond to classification uncertainty (Lucieer 2004).

It has been shown that different shapes in feature space can be used to visualise the decision boundaries and parameters of classification algorithms. Thearling et al. (2001) highlight the importance of 'letting the user understand what is going on'. Visualisation of decision boundaries provided insight into the internal functioning of the classifiers and presented the opportunity for greater understanding of classifications' results. This not only applies to instances where misclassification occurred but also where classification was correct. Insight was gained regarding classification and this resulted in increased confidence in the results of the classification and understanding of why classification errors occurred. Visualisation of classifier decision parameters highlighted the need for careful consideration when defining training regions and the importance of considering revision of those training regions.

The visualisation of the effects of texture and PCA on feature space allows prior knowledge of the effects these filters and transformations will have on classification. In the case of texture, this allows a qualitative visual assessment of the usefulness of different measures before the time consuming process of classification. Although Murray (in-press) found particular measures were more effective when used in classification of sub-Antarctic vegetation, the usefulness of different measures will vary for different imagery. Therefore the ability to visualise the effects of different texture measures in feature space is useful.

Visualising PC bands also yields important information about the nature of the spectral bands in an image. By comparing a feature space plot based on PC bands to a plot based on spectral bands, key information regarding the correlation of those spectral bands can be qualitatively assessed.

Chapter 6 Conclusion

Chapter 2 discussed relevant literature regarding visualisation and visual data mining. These theories and technologies were then applied in Chapter 3 to create a visualisation prototype. The prototype allows the visual exploration and mining of patterns in the data and visualisation of classifier decision boundaries.

Chapter 3 discussed the issues of volume based feature space representation and surface representations. It also presented a technique for linking feature space and geographic (image) space. This linkage plays a key role in the visual data mining process and is a valuable addition to the prototype.

The core VDM component of the system was the intersection routine. By computing and visualising the intersection between clusters in feature space the prototype allowed the user to explore the nature of their training data. This allowed for directed revision of hypotheses regarding the imagery.

During the study various shapes were used to represent clusters in feature space. It was shown that different 3D shapes can be used to represent the decision boundaries and parameters of popular classifiers. Visualisation of these properties allowed increased understanding of the image classification process and the results obtained.

The case study presented in Chapter 4 yields specific evidence for evaluation of the hypothesis. It is argued that this evidence supports the hypothesis. The incorporation of interactive visualisation with existing software for data mining of satellite imagery:

- provided opportunity for enhanced understanding of the image classification process,
- showed possible revisions to current hypotheses,
- revealed subtle patterns in satellite image data, and;
- resulted in more insightful analyses than non-visual data mining alone allowing for greater confidence in the results (Keim, D A 2002).

In conclusion, the provision of a visual data mining system for satellite imagery results in added value for analyses and increased understanding and confidence for users. This study highlights the positive impact of visual data mining on satellite image classification. Further work, involving feedback from the remote sensing community, should be undertaken to provide further support for the hypothesis. This work may take the form of a focus group user test.

Chapter 7 Further Work

This study has revealed many promising paths for future research. They can be categorised as those relating to the extension of the prototype and those relating to the use of visual data mining in satellite image classification.

7.1 Prototype Enhancements

- It would be interesting to examine the effect of the reversal of the feature space to geographic space link. In this scenario the user would highlight pixels in the image and be simultaneously shown that pixel's location in feature space. This may be useful in defining training regions based on visual inspection.

- A tighter incorporation of visualisation and automated data mining could be achieved by allowing the classification process to be visualised pixel by pixel. As a pixel is classified it is shown as a point in a feature space plot. The decisions made by the classifier are visualised and the point assigned a colour based on its classification. For example, a minimum distance classifier could visually show the measurement of Euclidean distance to the class means.

7.2 Visual Data Mining in Remote Sensing

- A focus group test of a complete (including features described in Section 7.1) visual data mining system for satellite imagery should be conducted. This group should comprise both experienced imagery analysts and novice users. The group could be divided in two. One group would conduct a non-visual analysis and the other conduct an enhanced visual analysis. Questionnaires and group discussion could be used to gauge the usefulness of the system.

- Lucieer (2004) mentions the need for research into visualisation for remote sensing education. Students may benefit by having the ability to visually predict and explain their results. This study has shown that visualisation of classifier decision parameters presents the opportunity for greater understanding of the underlying algorithm.

References

Abdi, H 2007 (in-press), 'Eigen-decomposition: eigenvalues and eigenvecteurs.' in NJ Salkind (ed.), *Encyclopedia of Measurement and Statistics*, Sage, Thousand Oaks (CA).

Bloomenthal, J & Wyvill, B 1997, *Introduction to Implicit Surfaces*, Morgan Kaufmann Publishers Inc.

Brandner, S 2005, 'The feasibility of using remote snesing to map and monitor vegetation change on sub-Antarctic Heard Island', Honours thesis, University of Wollongong.

Braverman, A & Kahn, B 2004, 'Visual Data Mining for Remote Sensing Data Sets', *AGU Fall Meeting Abstracts*, vol. 43, p. 0442.

Carr, DB, Wegman, EJ & Luo, Q *ExplorN*, <http://science.gmu.edu/~rmoustaf/explorn.html>.

Castleman, KR 1996, *Digital Image Processing*, Prentice-Hall.

Clarkson, K 2004, *hull - convex hulls, delauney triangulations, alpha shapes.*, viewed Oct 10 2006, <http://netlib.bell-labs.com/netlib/voronoi/hull.html>.

Cline, HE & Lorensen, WE 1985, *System and method for the display of surface structures contained within the interior region of a solid body*, 06/741,390, General Electric Company, US, June 5, 1985, 4,710,876.

Cox, KC, Eick, SG, Wills, GJ & Brachman, RJ 1997, 'Brief Application Description; Visual Data Mining: Recognizing Telephone Calling Fraud', *Data Mining and Knowledge Discovery*, vol. V1, no. 2, pp. 225-31.

Drebin, RA, Carpenter, L & Hanrahan, P 1988, 'Volume rendering', in *Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, ACM Press, pp. 65-74.

Edelsbrunner, H & Mücke, EP 1994, 'Three-dimensional alpha shapes', *ACM Trans. Graph.*, vol. 13, no. 1, pp. 43-72.

Ferreira de Oliveira, MC & Levkowitz, H 2003, 'From visual data exploration to visual data mining: a survey', *Visualization and Computer Graphics, IEEE Transactions on*, vol. 9, no. 3, pp. 378-94.

Firebaugh, MW 1993, *Computer graphics: tools for visualization*, William C. Brown Publishers.

Fischer, K 2000, *Introduction to Alpha Shapes*, viewed 10/10/2006 <http://n.ethz.ch/student/fischerk/alphashapes/as/>.

Fisher, RA 1988, *Iris Plants Database*, Machine Learning Dataset.

Foley, JD, van Dam, A, Feiner, SK, Hughes, JF & Phillips, RL 1993, *Introduction to Computer Graphics*, 1st edn, Addison-Wesley Professional.

Gonzalez, RC & Woods, RE 1983, *Digital Image Processing*, Addison-Wesley.

Haralick, RM, Shanmugan, K & Dinstein, I 1973, 'Textural features for image classification', *IEEE Transactions on Systems, Man and Cybernetics*, no. 2, pp. 610-21.

Hearn, D & Baker, PM 2003, *Computer Graphics with OpenGL*, 3rd edn, Prentice Hall.

Keim, DA 2001, 'Visual exploration of large data sets', *Commun. ACM*, vol. 44, no. 8, pp. 38-44.

---- 2002, 'Information Visualization and Visual Data Mining', *IEEE Transactions on Visualization and Computer Graphics*, vol. 7, no. 1, pp. 100-7.

Keim, DA, Panse, C, Sips, M & North, SC 2004, 'Visual Data Mining in Large Geospatial Point Sets', *IEEE Computer Graphics and Applications*, vol. 24, no. 5, pp. 36-44.

Lacroute, P & Levoy, M 1994, 'Fast Volume Rendering Using a Shear-Warp Factorization of the Viewing Transformation', paper presented to SIGGRAPH'94, Orlando, Florida.

Lillesand, TM & Kiefer, RW 2000, *Remote sensing and image interpretation*, 4 edn, John Wiley & Sons, New York.

Lorensen, WE & Cline, HE 1987, 'Marching cubes: A high resolution 3D surface construction algorithm', in *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, ACM Press, pp. 163-9.

Lucieer, A 2004, 'Uncertainties in Segmentation and their Visualisation', PhD thesis, Utrecht University.

---- 2006, Heard Island Dataset, Regions of Interest to S Welch, H Murray & R Williams.

Lucieer, A & Kraak, MJ 2004, 'Alphas-shapes for visualising irregualr shapes class clusters in 3D featuer space for classifcation of remotely sensed imagery'.

Mortenson, ME 1999, *Mathematics for Computer Graphics Applications*, Industrial Press Inc.

Murray, H in-press, 'Texture For Satellite Imagery Classification', Honours thesis, University of Tasmania.

Randen, T & Husøy, JH 1999, 'Filtering for texture classification: A comparative study', *IEEE Transactions on Pattern Ananlysis and Machine Intelligence*, vol. 21, no. 4, pp. 291-310.

Richards, JA 1986, *Remote Sensing Digital Image Analysis*, Springer-Verlag.

RSI 2003a, *IDL Online Help*, Research Systems Inc.

---- 2003b, *IDL iTools*.

Schowengerdt, RA 1983, *Techniques for Image Processing and Classification in Remote Sensing*, Academic Press.

---- 1997, *Remote Sensing, Models and Methods for Image Processing*, Academic Press.

Shneiderman, B 1996, 'The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations', paper presented to IEEE Symposium on Visual Languages.

Simoff, SJ 2002, 'VDM@ECML/PKDD2001: the International Workshop on Visual Data Mining at ECML/PKDD 2001', *SIGKDD Explor. Newsl.*, vol. 3, no. 2, pp. 78-81.

Soukup, T & Davidson, I 2002, *Visual Data Mining: Techniques and Tools for Data Visualization and Mining*, 1 edn, Wiley.

Thearling, K, Becker, B, DeCoste, D, Mawby, B, Pilote, M & Sommerfield, D 2001, 'Visualizing Data Mining Models', in U Fayyad, G Grinstein & A Wierse (eds), *Information Visualization in Data Mining and Knowledge Discovery*, Morgan Kaufman.

Wahl, FM 1987, *Digital Image Signal Processing*, Artec House.

Wang, F 1999a, 'Improving remote sensing image analysis through fuzzy infromation representation', *Photogrammetric Engineering and Remote Sensing*, vol. 56, no. 8, pp. 1163-9.

---- 1999b, 'Fuzzy supervised classification of remote sensing images', *IEEE Transactions on Geoscience and Remote Sensing*, vol. 31, no. 1, pp. 136-45.

Wegman, EJ 2001, *Visual Data Mining*, Center for Computational Statistics, George Mason University, VA, viewed 30 Oct 2006, <http://www.galaxy.gmu.edu/stats/syllabi/inft979/VisualDataMining.pdf>.

Witten, IH & Frank, E 2005, *Data Mining - Practical Machine Learning Tools and Techniques*, 2nd edn, Morgan Kaufmann.

Zhang, J & Foody, GM 2001, 'Fully-fuzzy supervised classification of suburban land cover from remotely sensed imagery: Statistical and artifical neural network approaches', *International Journal of Remote Sensing*, vol. 22, no. 4, pp. 615-28.

Appendix A Imagery

This appendix contains (false) colour composites of the imagery used in this study. Each image is shown as a visible light colour composite and a false colour composite. Greyscale images of Paddick Valley overlayed with Regions of Interest are provided for reference. Classification images for Paddick Valley are also provided. These are based on the *1c_training* pixels and bands 4, 2 and 1 of the image.

Hobart Landsat TM Image

Description	Hobart, Tasmania, Australia and surrounds.
Source:	Centre for Spatial Information Science (CenSIS), School of Geography and Environmental Studies, University of Tasmania
Capture Date	28/9/1999
Size	600 x 600 pixels
Radiometric Resolution	8-bit

Band	Wavelength Range (μm)
1	0.45 – 0.52 (blue)
2	0.52 – 0.60 (green)
3	0.63 – 0.69 (red)
4	0.76 – 0.90 (near infrared)
5	1.55 – 1.75 (mid infrared)
6	2.08 – 2.35 (mid infrared)

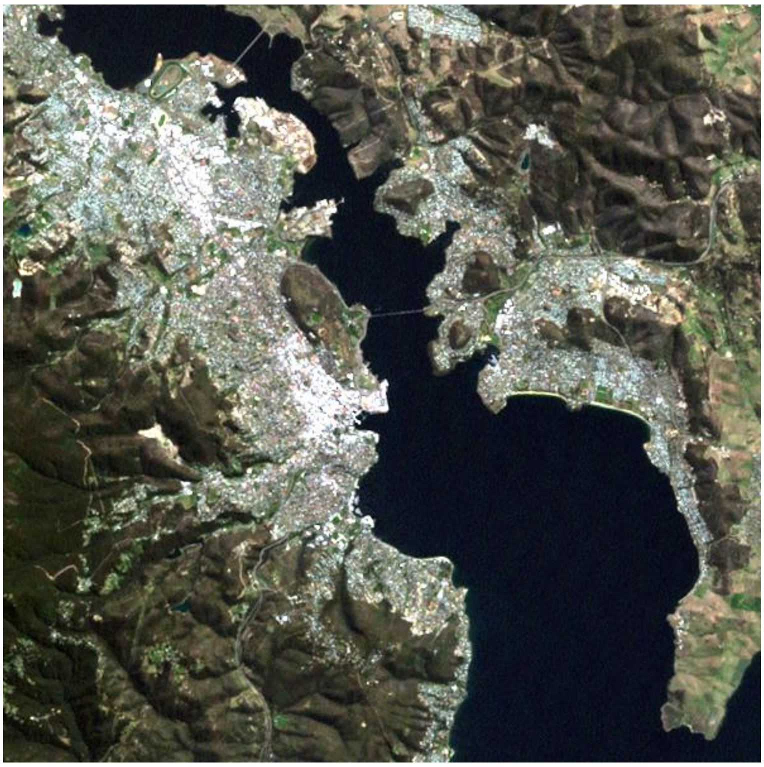


Figure A-1: Colour composite of visible light bands of the Hobart Landsat TM image.

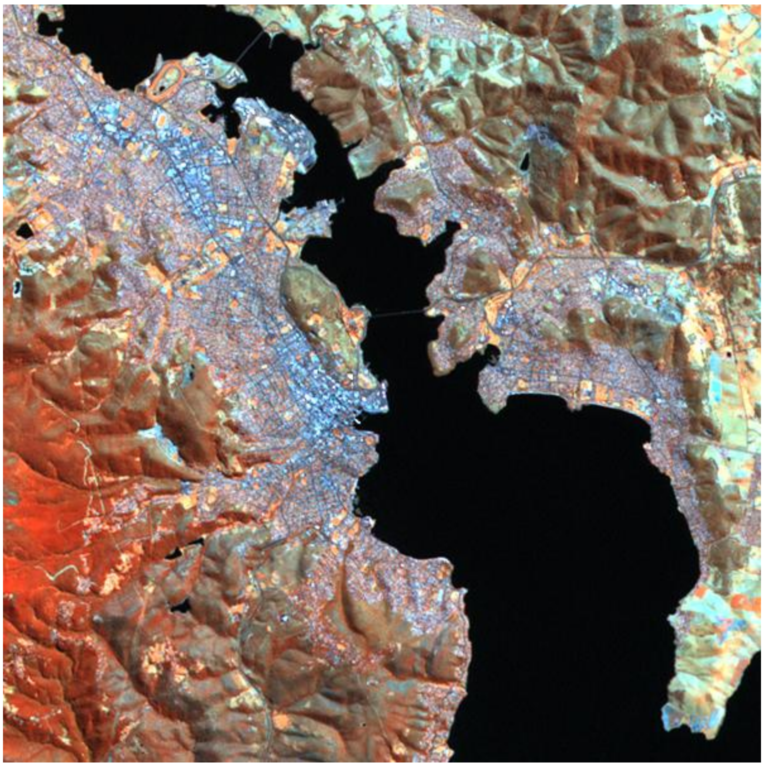


Figure A-2: False colour composite of bands 4,5 and 6 of the Hobart Landsat TM image.

Paddick Valley IKONOS Image

Description	Paddick Valley, Heard Island, Australian Territory.
Source:	Original source: www.geoimage.com.au Subset extracted and pre-processed by Lucieer (2006), Centre for Spatial Information Science (CenSIS), School of Geography and Environmental Studies University of Tasmania
Capture Date	January 2004
Size	1901 x 1601 pixels
Radiometric Resolution	16-bit

Band	Wavelength Range (µm)
1	0.445 – 0.516 (blue)
2	0.506– 0.595 (green)
3	0.632 – 0.698 (red)
4	0.757 – 0.853 (near infrared)

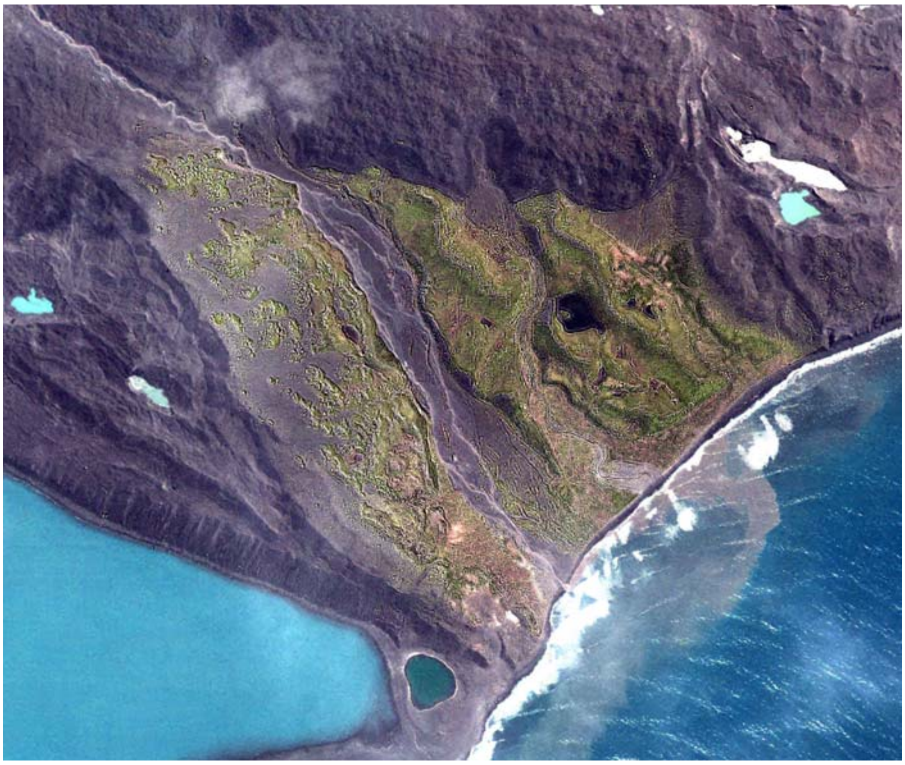


Figure A-3: Colour composite showing visible light bands of the Paddick Valley IKONOS image.

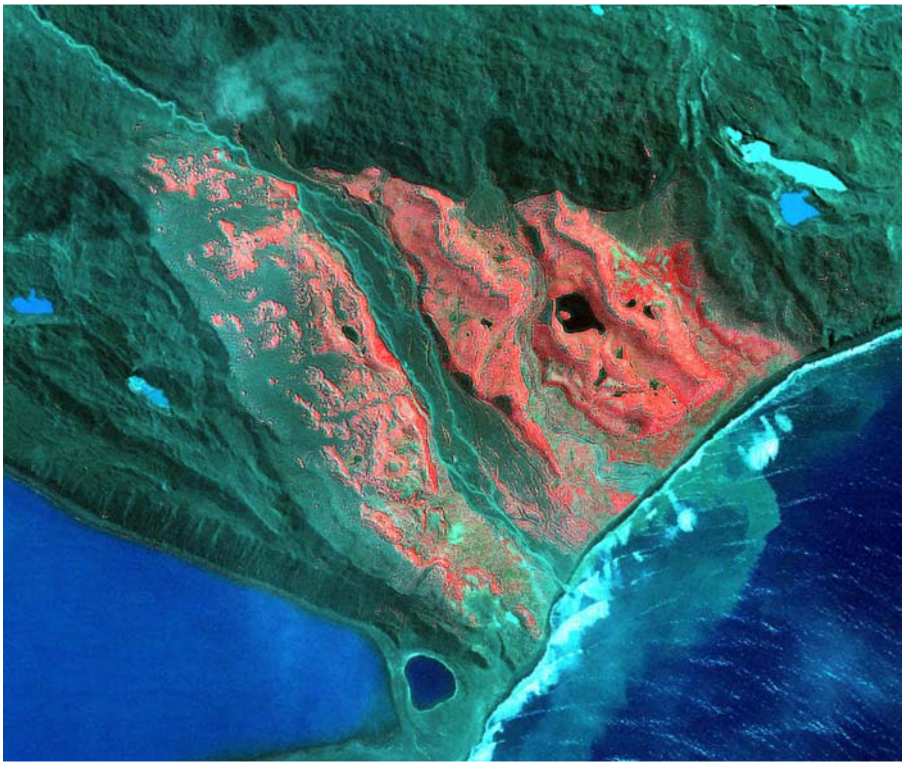


Figure A-4: False colour composite of bands 4,3,2 of the Paddick Valley IKONOS image.

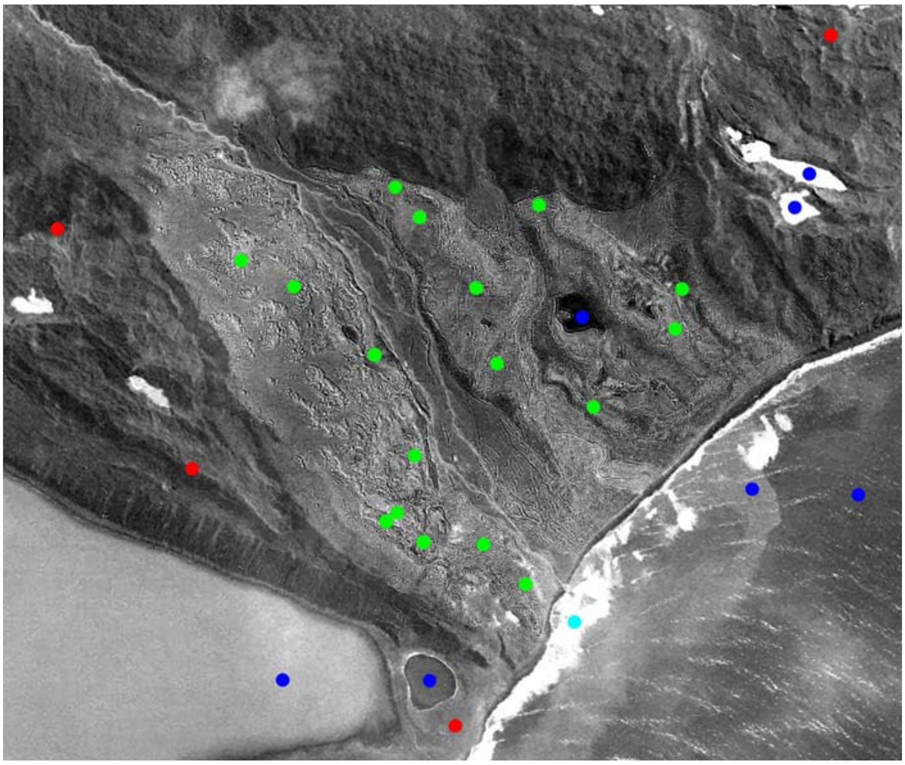


Figure A-5: The 1c-all regions.

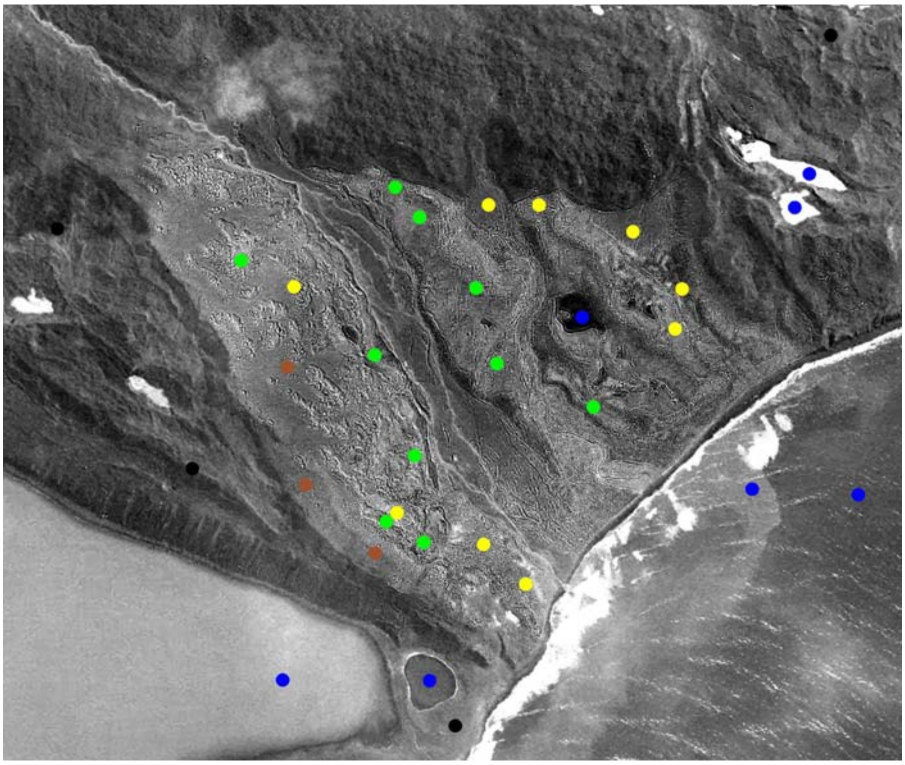


Figure A-6: The 3c-all regions.

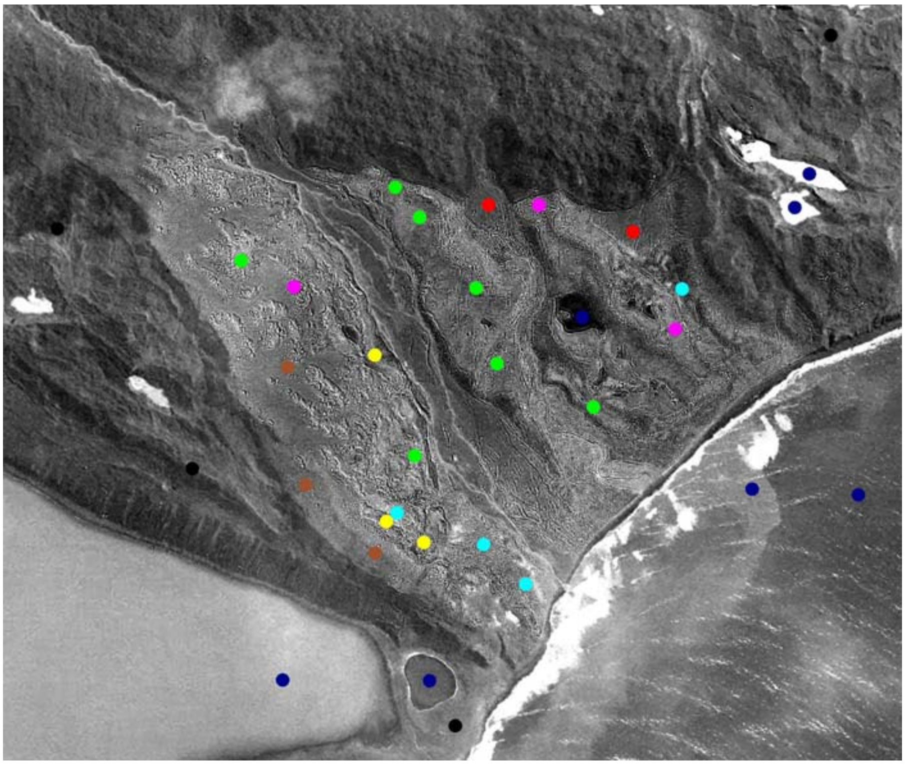


Figure A-7: The 6c-all regions.

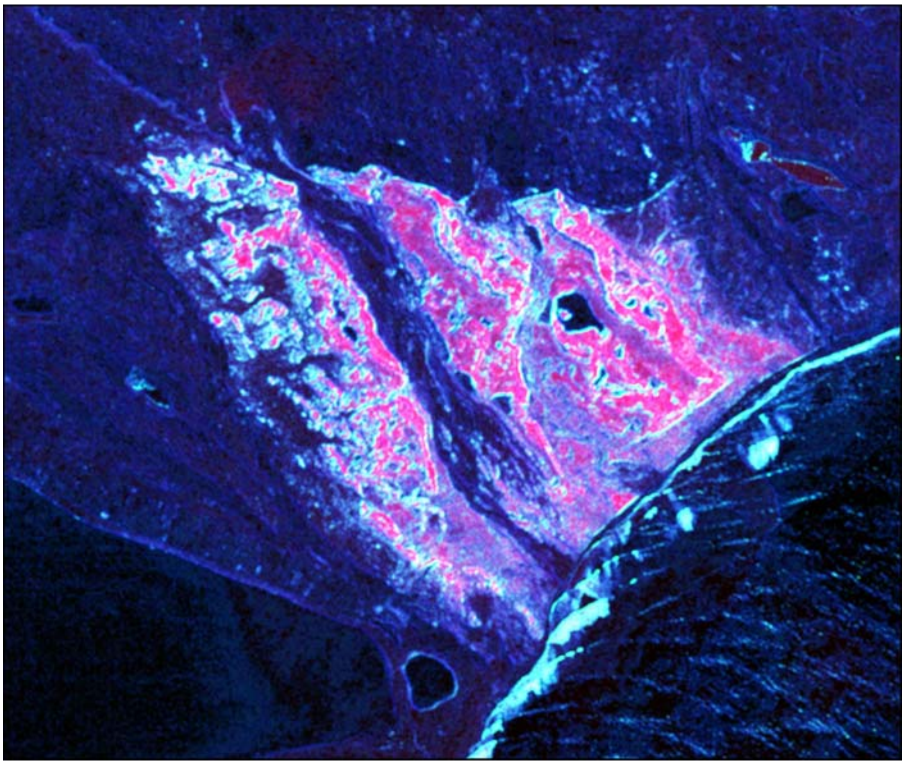


Figure A-8: False colour composite of mean, dissimilarity and entropy texture measures.

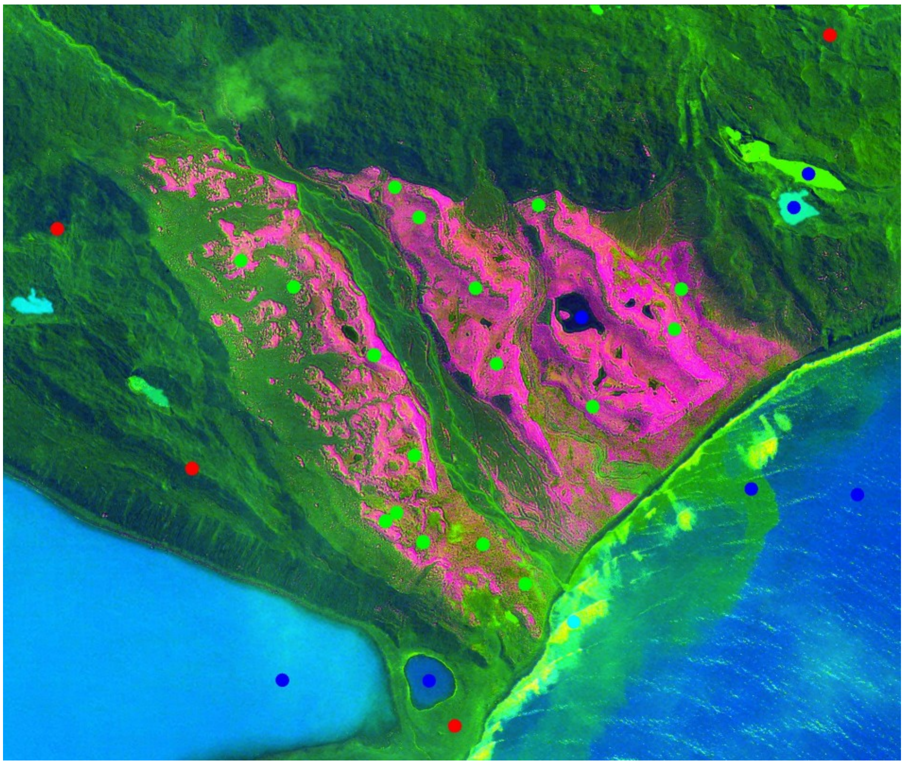


Figure A-9: False colour composite of three principal component bands overlayed with 1c-all regions.

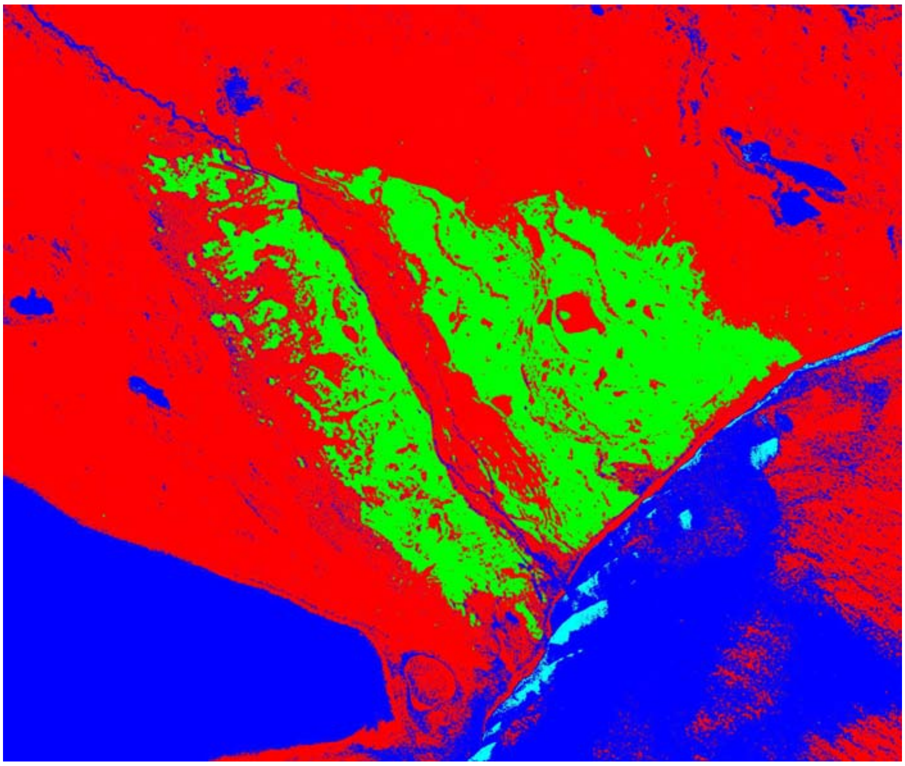


Figure A-10: Thematic map produced by the minimum distance classifier.

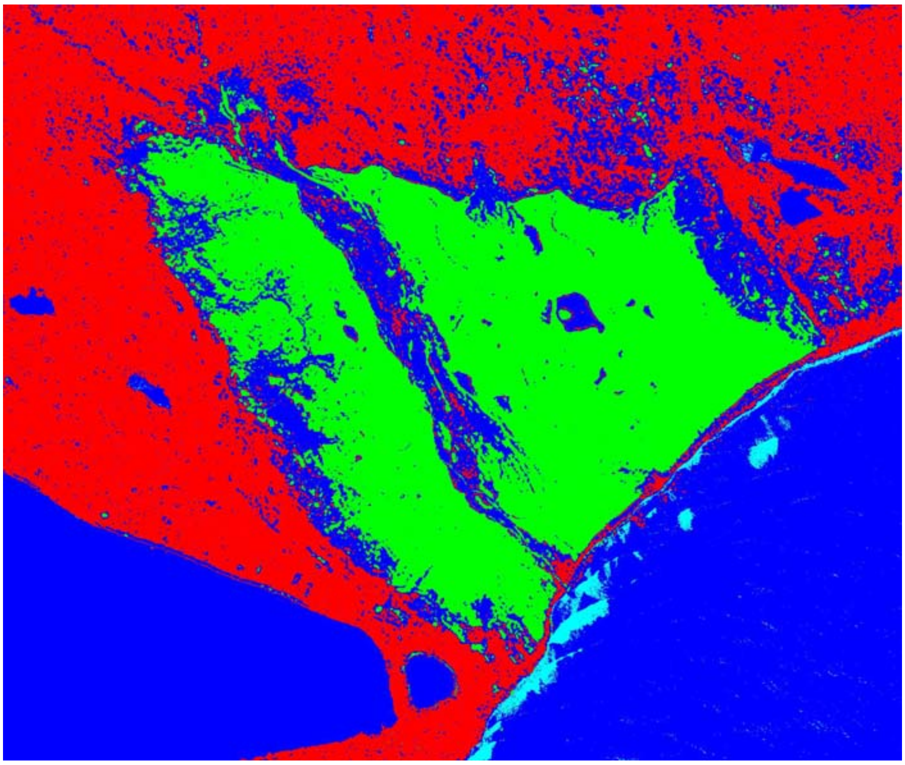


Figure A-11: Thematic map produced by the maximum likelihood classifier.

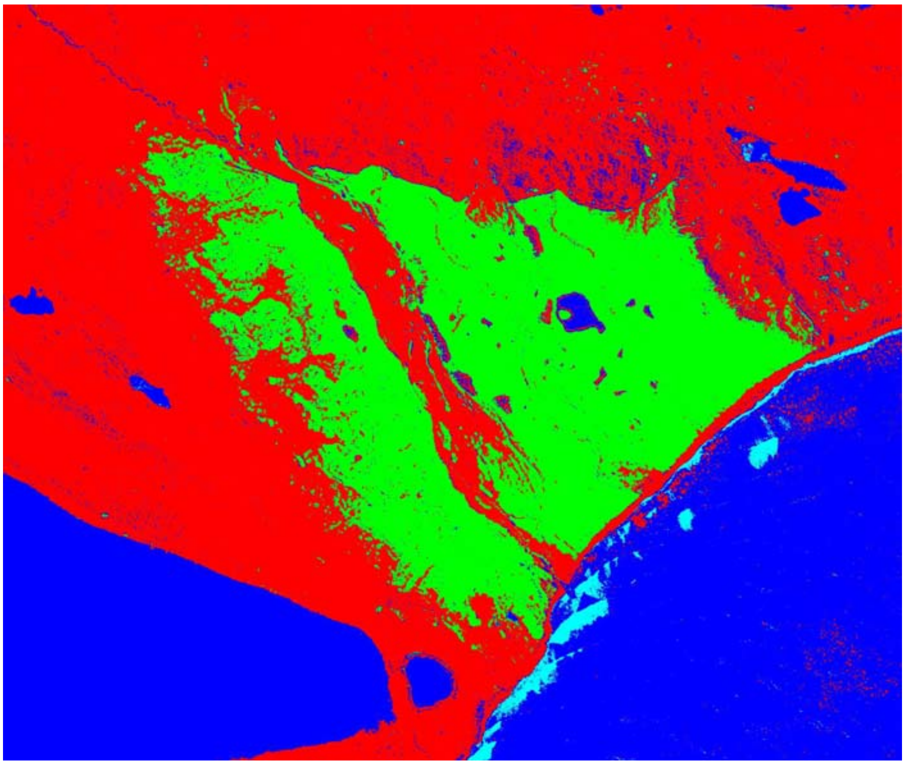


Figure A-12: Thematic map produced by the k-nearest neighbour classifier using 7 nearest neighbours.

Appendix B Program Code

IDL code for the visualisation prototype is included on the CD-ROM accompanying this thesis. Instructions for installing and running the prototype are included on the disc. In order to run the prototype a licensed installation of IDL 6.0 and ENVI 4.0 is required.