Segmentation of Multispectral Satellite Images based on Seeded Region Growing and Instance-Based Learning

by

Octavio Gómez Rámos

Masters Thesis

Submitted to the Program in Computer Science,

Computer Science Department

in partial fulfillment of the requeriments for the degree of

MASTER IN COMPUTER SCIENCE

at the

National Institute of Astrophysics, Optics and Electronics

Octover 2007

Tonantzintla, Puebla

Advisors:

Dr. Jesús A. González Bernal, INAOE

Dr. Eduardo F. Morales Manzanares, INAOE

Abstract

In order to reach the balance between the fulfillment of human needs and the protection of the environment, it is necessary to have detailed and accurate information about natural resources. Such information can be obtained through thematic maps, a product of remote sensing. In remote sensing, the generation of accurate thematic maps presents many research challenges, being one of them, image segmentation.

In this thesis, a novel segmentation algorithm based on seeded region growing and instance based learning is proposed. The algorithm includes a novel automatic seed generation approach that uses a histograms analysis, a new weighted instance-based learning algorithm (WIBK) which obtains one or more weights per feature per class, a novel region growing algorithm (SRG-WIBK) that uses WIBK as decision criteria, and a novel region-merging scheme based on ownership tables which allows to merge regions according to user needs. The WIBK algorithm was experimentally evaluated on several databases from the UCI repository, and compared against instance-based and non instance-based learning algorithms showing a very competitive performance. The SRG-WIBK algorithm was tested on multispectral synthetic images and compared against the algorithms implemented in the ERDAS software showing very even results.

Resumen

Para lograr el balance entre la satifacción de las necesidades humanas y la protección del medio ambiente, es necesario tener información detallada y precisa sobre los recursos naturales. Esta información puede ser obtenida mediante mapas temáticos, uno de los productos de la percepción remota. En percepción remota, la generación de mapas temáticos fiables presenta muchos retos de investigación, siendo uno de ellos, la segmentación de la imagen.

En esta tesis se propone un nuevo algoritmo de segmentación basado en crecimiento de regiones y aprendizaje basado en instacias. Dentro de las características del algoritmo se encuentran un nuevo esquema automático de obtención de semillas basado en análisis de histogramas, un nuevo algoritmo de aprendizaje basado en instacias (WIBK) que obtiene uno o más pesos por atributo por clase, un nuevo algoritmo de crecimiento de regiones (SRG-WIBK) que hace uso de WIBK como criterio de decisión y un nuevo esquema de agrupamiento de regiones basado en tablas de propiedad que permite agrupar regiones de acuerdo a las necesidades del usuario. El algoritmo WIBK fué evaluado experimentalmente en varias bases de datos del repositorio UCI, y comparado contra algoritmos de aprendizaje basados y no basados en instancias mostrando resultados muy competitivos. El algoritmo SRG-WIBK fué probado en imágenes multiespectrales sintéticas, y comparado contra los algoritmos implementados en el software ERDAS mostrando resultados muy parejos.

Acknowledgments

I wish to thank my thesis advisors, PhD Jesús Antonio González Bernal and PhD Eduardo Francisco Morales Manzanares for the guidance and the opportunity to work with them. I appreciate their support, patience, understanding, and the knowledge shared with me during these years of work.

Thanks to my synods PhD Carlos Alberto Reyes García, PhD José Francísco Martínez Trinidad and PhD Leopoldo Altamirano Robles for their valuable comments and the time dedicated to the review of this thesis.

Thanks to the National Institute of Astrophysics, Optics and Electronics (INAOE) for the opportunity of make my master's studies, and to the National Council of Science and Technology (CONACYT) for the support provided through the grant for masters studies number 201804.

Thanks to all my masters schoolmates, for all the experiences and knowledge shared.

Thanks to my wife Erika Danaé for her love, company and wise advices. I love you and I admire you.

Thanks to my parents, brothers and family, you will always be an integral part of me.

Dedicatory

To my parents and to my wife.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

In this chapter, motivation of the thesis, problem description, research issues, main objective and contributions are presented.

1.1 Motivation and problem description

There is no doubt that the environment is essential to support many aspects of human life such as health, safety, economic activity and well-being. The health of the environment determines the overall human quality of life, while an unhealthy environment considerably reduces this quality. There are many reasons to protect the environment including its own inherent value and the responsibility to leave a legacy of fully functional natural resources. This protection can be achieved through sustainable development. Sustainable development is defined as the balance between the fulfillment of human needs and the protection of the natural environment in order to meet these needs in the indefinite future.

Sustainable development requires detailed, timely, and accurate information on land resources [40], as well as changes occurring over time in the land. This information can be provided through remote sensing applications. In remote sensing, the information is commonly presented as a thematic map. A thematic map shows, over a cartographic base, characteristics or concepts (such as vegetation or a urban settlement) related to a geographic region.

Automatic generation of thematic maps with multispectral satellite images presents many research challenges, one of them being image segmentation. The quality of the thematic map has an influence on the decisions taken based on this information.

One problem in multispectral satellite image segmentation is the ambiguity in the gray value of some pixels (ambiguity means that, during segmentation, a gray value can belong to more than one region). This problem commonly appears in the borders of the regions, often caused by limitations of the image recording sensor. This ambiguity decreases the quality of the segmentation results. In addition, traditional methods in automatic generation of thematic maps put little attention to the segmentation step, considered as an independent area, mainly related with computer vision. Another problem in automatic generation of thematic maps is the small amount of labeled data, due to the effort implied in data acquisition with site studies (ground truth data). This problem forces us to employ an approach designed to obtain more profit from the labeled data to reinforce the segmentation step.

In this thesis, a novel segmentation algorithm based on seeded region growing and instance based learning is proposed. The algorithm includes a novel automatic seed generation by means of a histogram analysis, a new weighted instance-based learning algorithm (WIBK) which obtains one or more weights per feature per class, a novel region growing algorithm (SRG-WIBK) that uses WIBK as decision criteria, and a novel region-merging scheme based on ownership tables which allows to merge regions according to user needs. The proposed algorithm is used to perform multispectral satellite image segmentation.

1.2 Research issues

In seeded region growing, the selection of seeds is a key element for proper identification of regions [1]. Traditionally, this has been made using manually generated seeds

[1], random seeds [25] or seeds based on borders [11]. This thesis proposes a novel seed generation algorithm based on histogram analysis which allows to find automatically homogeneous regions. The proposed algorithm is fully automatic and free of tuning parameters. This algorithm is explained in Section 5.2.

Instance based learning algorithms are highly sensitive to noise and irrelevant features [3]. Many weighting algorithms have been developed to diminish this problem [23, 6, 43]. These weighting algorithms commonly find one weight per feature. In remote sensing it is common that, for example, a feature f is good to discriminate between class A and class B, but is not good to discriminate between class A and class C, for this reason is useful to have one weight per feature per class instead of one single weight per feature. Taking this into account, a weighted instance-based learning algorithm has been developed, that can obtain one or more weights per feature per class. The algorithm is described in Section 5.4.

It is common that some regions obtained in a segmentation process need to be merged in order to obtain the desired final results. For example, a user may need two vegetation classes to be merged to produce a single vegetation class. Traditionally, region merging is made through a user-guided process [1] or through merging rules [39] which indicate when two or more regions must be merged. A user-guided process allows the user to control the final results of the segmentation, and rules allow to perform the region merging automatically. In this thesis, a user-guided merging process through ownership tables is proposed. In an ownership table, the user indicates which regions must be merged, and the name of the resultant region. When the user gives a name to the resultant segmentation, he also gives meaning (semantic) to the region. Ownership tables combine the advantages of an user-guided process and empirical rules because the user has control over the final segmentation results and, once the ownership tables have been defined, can be applied over other similar images to automatically merge regions. Ownership tables are explained in Section 5.5.

1.3 Objective

The main objective of this work is the development of a segmentation algorithm based on seeded region growing that employs weighted instance-based learning as decision criteria.

1.4 Contributions

The contributions of this thesis are as follows:

- A new algorithm that combines seeded region growing and instance based learning for multispectral satellite image segmentation in remote sensing applications.
- A new algorithm for automatic generation of seeds based on histogram analysis. The algorithm finds the seeds required by the region growing algorithm.
- A new noise- tolerant weighted instance-based learning algorithm that uses a novel approach based on intervals of features to obtain weights useful to improve the classification task.
- The introduction of ownership tables to merge and provide semantic meaning to the segmented regions according to the user needs. Ownership tables are also useful for multilevel segmentation.

The proposed algorithm was designed to work with multispectral satellite image segmentation but also can be used in many other domains such as medical applications [15]. Although, due to the wide diversity of segmentation applications, the development of a generic segmentation algorithm is beyond the scope of this research.

1.5 Thesis organization

The rest of the thesis is organized as follows. Chapter 2 provides the main concepts of remote sensing, describes the concept of remote sensing image data, gives an introduction to remote sensing data interpretation, and presents a brief introduction to image classification.

Chapter 3 presents the algorithms from which the proposed algorithms are derived. The seeded region growing algorithm is presented first, along with a formal definition of image segmentation. Instance-based learning algorithm is presented next, along with his learning task and framework.

Chapter 4 describes the related work. In this chapter, two kinds of research are presented. First, work that make use of machine learning tools for multispectral image segmentation. Second, recent work that use seeded region growing for image segmentation.

Chapter 5 introduces the proposed algorithm in detail. This chapter begins with an overview of the algorithm, and next describes the automatic seeds generation method, the SRG-IBK algorithm, the weighting scheme, and finally the theory of ownership tables.

Chapter 6 describes the experimental results. The first part presents the results of the proposed weighting algorithm over several UCI [28] databases. In the second part, the results of the proposed segmentation algorithm over synthetic multispectral satellite images are given.

Chapter 2

Remote sensing

This chapter presents the main concepts of remote sensing. Section 2.1 gives an introduction to remote sensing, Section 2.2 presents the most common approaches to remote sensing data interpretation and Section 2.3 describes a review of quantitative analysis by classification.

2.1 Introduction to remote sensing

This section presents an introduction to the remote sensing field. Section 2.1.1 describes the remote sensing concept used in this thesis, Section 2.1.2 briefly describes the electromagnetic spectrum, Section 2.1.3 presents the main characteristics of remote sensing images, section 2.1.4 describes the wavelenghts most commonly used in remote sensing, and Section 2.1.5 gives a brief introduction to remote sensing platforms.

2.1.1 The remote sensing concept

For the purposes of this thesis, the following definition of remote sensing will be used:

> Remote sensing is the science of acquiring information about the surface of the earth without actually being in contact with it. This is done by sensing

> and recording reflected or emitted energy and processing, analyzing, and applying that information [7].

Frequently, the remote sensing process involves an interaction between incident radiation and the targets of interest. This is exemplified by the use of imaging systems where the following seven elements are involved:

1. **Energy source**. Provides electromagnetic energy to the target.
2. **Radiation and Atmosphere**. As the energy travels from its source to the target, or from the target to the sensor, it interacts with the atmosphere it passes through.
3. **Interaction with the Target**. The energy interacts with the target depending on the properties of the target and the radiation.
4. **Recording of Energy by the Sensor**. After the energy has been emitted from the target, a remote sensor is required to collect and record the electromagnetic radiation.
5. **Transmission, Reception, and Processing**. The energy recorded by the sensor has to be transmitted to a receiving and processing station.
6. **Interpretation and Analysis**. The processed image is interpreted to extract information about the target.
7. **Application**. The information is applied to solve a particular problem or to reveal new information.

Fig. 2.1 shows the seven elements of remote sensing. The sun (1) provides the electromagnetic energy, this energy interacts with the atmosphere (2) when traveling from the sun to the target (3), and from the target (3) to the sensor (4). A remote sensing station (5) receives the transmission, processes the information (6), performs an interpretation and outputs a remote sensing product (7).

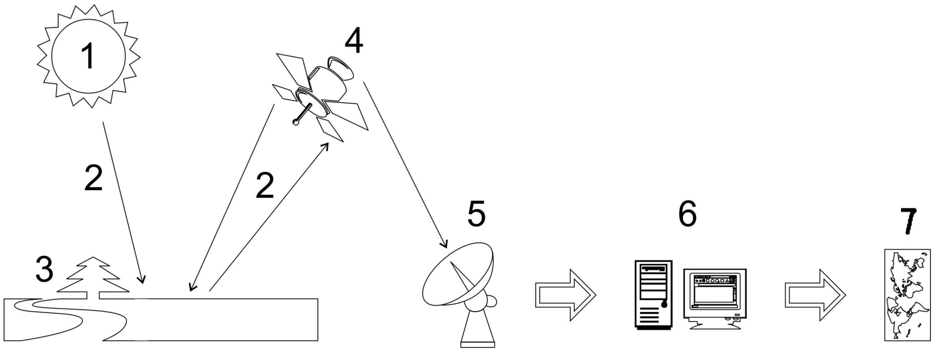


Figure 2.1: The seven elements of Remote Sensing

2.1.2 The electromagnetic spectrum

The electromagnetic spectrum (EM) is the distribution of electromagnetic radiation according to energy, organized according to frequency and wavelength. The sun, earth, and other bodies radiate electromagnetic energy of varying wavelengths. The "electromagnetic spectrum" (usually just spectrum) of an object is the frequency range of electromagnetic radiation with wavelengths from thousands of kilometers down to fractions of the size of an atom. The short wavelength limit is the Planck length, and the long wavelength limit is the size of the universe itself, so, in principle, the spectrum is infinite.

Table 2.1 shows approximate wavelengths, frequencies, and energies for selected regions of the electromagnetic spectrum. The notation eV stands for electron-volts, a common unit of energy measure in atomic physics. A graphical representation of the electromagnetic spectrum is shown in Fig 2.2.

2.1.3 Characteristics of remote sensing images

One characteristic of the images acquired by means of sensors on aircraft or spacecraft platforms is that they are available in digital format. Data that is not recorded in digital form normally needs to be converted into a digital format by means of a digitalization equipment such as scanners or cameras. Digital data can be processed by comput-

Table 2.1: Spectrum of Electromagnetic Radiation

Region	Wavelength (Angstroms)	Wavelength (cm)	Frequency (Hz)	Energy (eV)
Radio	$> 10^9$	> 10	$< 3\text{x}10^9$	$< 10^{-5}$
Microwave	$10^9 - 10^6$	$10 - 0.01$	$3\text{x}10^9 - 3\text{x}10^{12}$	$10^{-5} - 0.01$
Infrared	$10^6 - 7000$	$0.01 - 7\text{x}10^{-5}$	$3\text{x}10^{12} - 4.3\text{x}10^{14}$	$0.01 - 2$
Visible	$7000 - 4000$	$7\text{x}10^{-5} - 4\text{x}10^{-5}$	$4.3\text{x}10^{14} - 7.5\text{x}10^{14}$	$2 - 3$
Ultraviolet	$4000 - 10$	$4\text{x}10^{-5} - 10^{-7}$	$7.5\text{x}10^{14} - 3\text{x}10^{17}$	$3 - 10^3$
X-Rays	$10 - 0.1$	$10^{-7} - 10^{-9}$	$3\text{x}10^{17} - 3\text{x}10^{19}$	$10^3 - 10^5$
Gamma Rays	< 0.1	$< 10^{-9}$	$> 3\text{x}10^{19}$	$> 10^5$

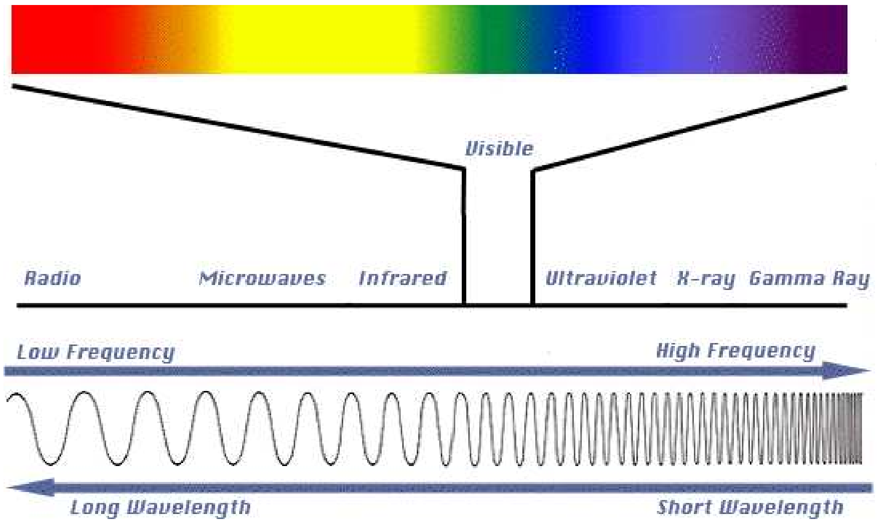


Figure 2.2: Graphical representation of the electromagnetic spectrum

ers either for machine assisted information extraction or for enhancement of its visual qualities. This thesis focuses on machine assisted information extraction.

There are mainly four kinds of resolution on remote sensing images: spectral, spatial, radiometric, and temporal. Spectral resolution is defined by the number of spectral measurements (spectral bands or channels) which conforms the image. Spatial resolution is described by pixel size in equivalent ground meters. The radiometric resolution is formed by the range and perceptible number of discrete brightness values. Radiometric resolution sometimes is called dynamic range. The radiometric resolution is commonly expressed in terms of the number of binary digits, or bits, necessary to represent the range of available brightness values, for example, data with radiometric resolution of 8 bits has 256 levels of brightness. Temporal resolution refers to how often an area can be imaged. In general, there is a trade-off between spatial resolution and temporal resolution. A sensor such as Landsat Thematic Mapper provides 30 meter pixels, but can only image a given area once every 16 days. On the other hand, the Advanced Very High-Resolution Radiometer can image the entire earth every day, but has 1.1 kilometer pixels. As with spatial resolution, the required temporal resolution is dependent on the application (domain dependent).

Usually, remote sensing systems record data from the visible and the near and mid infrared range, ultraviolet measurements are not recorded because of significant atmospheric absorption. The energy emitted by the earth itself (dominant in the thermal infrared wavelength range) can also be analyzed, although this energy is too small for most remote sensing mapping purposes. To record thermal infrared range, energy must be radiated from a platform, and the reflected energy must be recollected and recorded. Such a system is referred to as active since the energy source is provided by the platform. Remote sensing measurements that depend upon an energy source such as the sun are called passive.

2.1.4 Wavelengths commonly used in remote sensing

In remote sensing, some technical considerations exclude certain wavelengths to be considered for the images. These considerations are the selective opacity of the earth's atmosphere, the dispersion of atmospheric particles, and the significance of the data provided. The wavelengths commonly used in remote sensing applications are the visible/infrared range (between about 0.4 and 12 μm) and the microwave range (between about 30 to 300 mm). At microwave wavelengths it is common to use frequency rather than wavelength to describe the ranges, as a result, the microwave range of 30 to 300 mm corresponds to frequencies between 1 GHz and 10 GHz.

Each range of wavelength has its own characteristics in terms of the information it can contribute to the remote sensing process. This contribution depends of the interaction mechanism between the electromagnetic radiation and the materials examined. In the visible/infrared range, the energy measured depends upon properties such as the level of sedimentation of water pigmentation, moisture content, and the cellular structure of vegetation, and the mineral and moisture contents of soils. At the end of the infrared range, thermal properties of the surface and near subsurface are detected. In the microwave range, the roughness of the cover type is detected along with its electrical properties. In the range of 20 to 60 GHz, atmospheric oxygen and water vapors have a strong effect on transmission, affecting measurements in that range.

2.1.5 Remote sensing platforms

The process of image recording in remote sensing applications is carried out from satellite and aircraft platforms. This is because differences in altitude and stability lead to different image properties.

There are mainly two kinds of satellite orbits: geosynchronous and geostationary. Geosynchronous satellites have an orbit around the Earth with an orbital period matching the Earth's sidereal rotation period, which means that the satellite returns to the same place in the sky at the same time each day. Geostationary satellites have a geosynchronous orbit directly above the Earth's equator (0 latitude), with orbital eccentricity of zero. From the ground, a geostationary object appears motionless in the sky. Geostationary satellites are generally associated with climate studies and communications, and geosynchronous satellites are generally used for earth surface and oceanographic observations. The major distinction in the images that these kind of satellites provide lies in the available spatial resolutions, data acquired from geosynchronous satellites generally has pixel sizes of less than 100 m, and data acquired form geostationary satellites has a pixel resolution of the order of 1 km.

Recording sensors used in satellite remote sensing systems go from traditional cameras to *push-broom* mechanism in which a linear imaging array with sufficient detectors is placed out on the satellite, normal to the satellite's motion, such that each pixel can be recorded individually. When an image consists of two to ten spectral bands is called multispectral. Image data with more than 10 spectral bands is called hyperspectral. Hyperspectral images can register 200 spectral bands or more.

2.2 Remote sensing data interpretation

This section presents an introduction to remote sensing data interpretation. Section 2.2.1 describes the approaches used for digital images interpretation, Section 2.2.2 presents the main imagery types for photointerpretation and Section 2.2.3 gives a brief introduction to multispectral space and classes.

2.2.1 Approaches to interpretation of digital image data

Several algorithms can be used to extract information of an image available in digital form. There are two main approaches used for the interpretation of digital images. The first one is a quantitative analysis which involves the use of a computer to examine each pixel in the image individually. Inferences about pixels are specifically based on their attributes. This analysis is quantitative since pixels with equivalent attributes are often counted for estimating areas. The second approach involves a human analyst/interpreter extracting information by visual inspection of an image. This is referred to as photointerpretation or sometimes image interpretation; its success depends upon the analyst exploiting effectively the spatial, spectral and temporal elements present in the composed image product. In this approach, the analyst/interpreter notes generally large scale features and is often unaware of the spatial and radiometric features of the data. The two main approaches to digital images interpretation are often complementary. Photointerpretation is aided substantially if a certain level of digital image processing is applied to the images beforehand, while the success of a quantitative analysis depends on the information provided at key stages by an analyst (sometimes called domain expert).

In computer-based quantitative analysis the attributes of each pixel (such as the spectral bands available) are examined in order to give to the pixel a label identifying it as belonging to a particular class of pixels of interest to the user. As a result, the process is often also called classification.

2.2.2 Imagery types for photointerpretation

Remote sensing image products can be found either in photographic form or in digital format. A digital format is more flexible since photographic products can be created from the digital data, and because data can be processed by a computer for enhancement before visual interpretation and/or information extraction is performed. The wide range of image processing and image enhancement algorithms work only over a digital format.

There are two fundamental types of display products. The first is a black and white display of each band of images in gray value scale. If this display is produced from the raw digital data then black will correspond to a digital brightness value of 0 whereas white will correspond to the highest digital value. This is usually 63, 127, 255 or 4095 (for 6, 7, 8 and 12 bits data respectively). The images used in this thesis, Spot 5, have 255 gray values on each band (a gray scale of 255 values).

The second display product is a color composite (sometimes called false RGB) in which selected features or bands in multispectral data are chosen to be associated with the three additive primary colors (red, green and blue) in the display device. When data consists of more than three features or bands, a decision has to be made in order to keep only three of them, or alternatively, a mapping has to be created to allow all the features to be suitably combined into the three primary colors. When available data consists of a large number of bands (such as produced by aircraft scanners or by imaging spectrometers) only experience, and the area of application, will tell which three bands should be combined into a color product. For data with limited spectral bands however, the choice is straightforward.

2.2.3 Multispectral space and classes

Because of its ability to identify pixels based upon their numerical properties and its ability for counting pixels for area estimates, computer-based interpretation of remote sensing images is referred to as quantitative analysis. In this analysis, labels may be attached to pixels in view of their spectral character. This labeling procedure is implemented in a computer system by training it beforehand to recognize pixels with spectral similarities. In this thesis, the classification process is addressed by an instance-based learner, trained with the seeds that were automatically found by the region growing algorithm.

One way to represent multispectral images is to plot them in a multispectral vector space, with as many dimensions as spectral bands (spectral components) are in the image. Each pixel of an image is plotted as a point with coordinates given by the brightness value of the pixel in each component. This is illustrated in Fig. 2.3 for a simple two dimensional infrared space against visible red space. Fig. 2.3A shows the image, and Fig. 2.3B shows the bands which provides good discrimination. If the selected bands are good to discriminate among groups it is expected that pixels form groups in multispectral space corresponding to several cover types. The groups

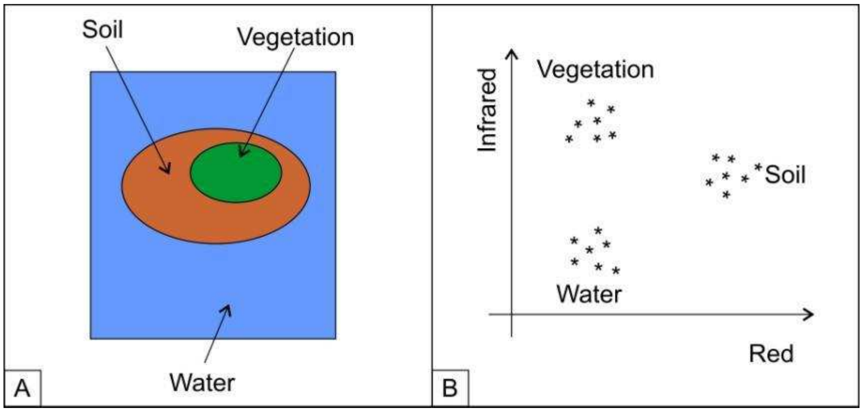


Figure 2.3: A two dimensional multispectral space

or clusters of pixel points are referred to as information classes since they belong to the actual classes which a computer will need to be able to recognize.

It is common to find that, in practice, information class groups may not be single clusters as depicted in Fig. 2.3. This situation is the result of many factors such as differences in moisture content of the target, differences in subtypes of cover, underlying vegetation, topographic influences and soil types. It is also common that the classes of interest do not form distinct groups, instead, groups are part of a continuum of data in the multispectral space. This happens, for example, when the satellite or the aircraft sensors might see a gradual variation of the land cover. In these kind of situations, it is necessary to determine appropriate sets of classes that help to discriminate among ambiguous land cover types. In this thesis, this problem is addressed by means of a weighted instance-based learning algorithm, whose weighting scheme assigns a higher weight to the spectral bands that better help to discriminate among classes.

2.3 Quantitative analysis by classification

This section presents an introduction to quantitative analysis by classification in remote sensing. Section 2.3.1 describes pixel vectors and labeling, Section 2.3.2 presents unsupervised classification and Section 2.3.3 presents supervised classification.

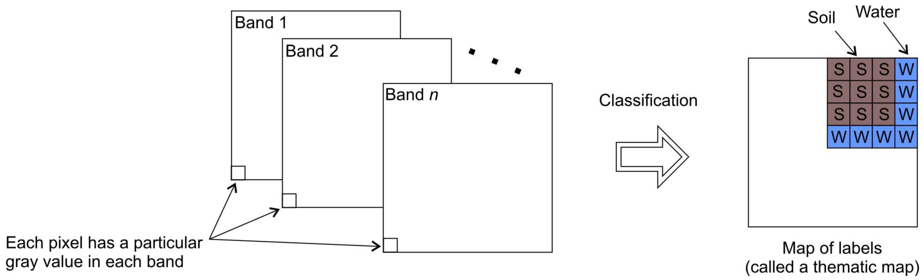

Band 1
Band 2
Band n
Each pixel has a particular gray value in each band
Classification
Soil
Water
S S S W
S S S W
S S S W
W W W W
Map of labels (called a thematic map)

Figure 2.4: Classification of remote sensing images

2.3.1 Pixel vectors and labeling

The use of machine learning tools makes possible the identification of classes corresponding to specific ground cover types. The features of each class are the pixel gray values on each spectral band. Strictly speaking, a pixel can be viewed as a vector that contains the brightness values (gray values) on each band arranged in a column:

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}$$

where x_1 to x_N are the brightness of pixel x in bands 1 to N respectively. It is simply a mathematical convention that these are arranged in a column and enclosed in an extended square bracket.

The classification process involves labeling pixels as belonging to a particular class. In the terminology of statistics this is more properly referred to as allocation rather than classification, however throughout this thesis, classification, categorization, allocation and labeling will be synonymously used.

Supervised and unsupervised classification are the two main approaches to pixel labeling. These can be used as alternative approaches, and each finds application in the analysis of remote sensing images. These approaches also can be combined into hybrid methodologies. The algorithm proposed in this thesis can be considered as a hybrid approach. The process begins with an automatic segmentation without knowing the names of the classes (unsupervised approach). Supervised and unsupervised classification are briefly explained in Sections 2.3.2 and 2.3.3.

2.3.2 Unsupervised classification

In remote sensing, unsupervised classification is a means by which pixels in an image are assigned to a class without the user's knowledge of the existence (or names) of those classes. It is generally performed using clustering methods. These procedures can be used to determine the number and location of the classes in which the data falls, and to determine the class of each pixel. The analyst then identifies those classes afterwards by associating a sample of pixels in each class with available reference data, which could include information from ground surveys. Clustering procedures are generally computationally expensive, but they are widespread used in the analysis of remote sensing images with commercial applications such as ERDAS IMAGINE.

Even when the algorithm proposed in this thesis uses instance-based learning (a supervised classifier), it is considered as unsupervised because the labels of the instances are unknown, and the training instances are automatically determined by the seeds generator of the region grower.

2.3.3 Supervised Classification

In remote sensing, there exist two main approaches to supervised classification: parametric and non parametric. Parametric methods assume that each class can be modeled by a probability distribution in multispectral space, and as a consequence, they are described by the parameters of those distributions. Nonparametric methods include machine learning approaches, and the assumptions made depend on the algorithm selected. The machine learning algorithm used in this thesis is instance-based learning. Instance based learning assumes that similar instances have similar classification.

Supervised classification depends of training data, image pixels with a previously known label. These labels can be obtained from a photointerpreter, ground studies, previous maps, etc. Supervised classification requires a set of labels for each region of interest, this means that supervised classification can not find regions for which it was not trained. The supervised classification algorithm uses the training data to build a model, and later the model is used to classify unlabeled pixels.

In the next chapter, the base algorithms of this thesis are presented. The fist section introduces the seeded region growing algorithm, along with a formal definition of image segmentation. The second section presents the instance-based learning algorithm, along with its framework and classification task.

Chapter 3

Base algorithms

This chapter presents the base algorithms used in the development of this thesis. Section 3.1 introduces the algorithm of seeded region growing and Section 3.2 presents the instance-based learning algorithms IB1 and IB*K*.

3.1 Seeded region growing

This section presents an introduction to the seeded region growing algorithm. Section 3.1.1 is an introduction, Section 3.1.2 provides a formal definition of segmentation, Section 3.1.3 describes the seeded region growing algorithm and section 3.1.4 explains the semi-interactive segmentation process.

3.1.1 Introduction

Automatic image segmentation is a very important initial step for many machine vision applications such as image description, recognition, retrieval, and object-based image compression [11]. Image segmentation is also an essential process for many remote sensing applications, and a key point to many compression standards such as MPEG-4 and MPEG-7 to perform object-based image coding and content-based image description and retrieval.

The general image segmentation problem involves the partitioning of a given image into a number of regions according to a given criteria which can be, for example, homogeneity or shape. Image segmentation can also be considered as a pixel labeling process in the sense that all pixels that belong to the same homogeneous region are assigned the same label [19].

The existing automatic image segmentation techniques can be classified into five approaches namely thresholding techniques [36], boundary-based methods [22], region based methods [18], hybrid techniques [17], and clustering-based techniques [38].

Seeded region growing (SRG) is a hybrid algorithm proposed by L. Adams and R. Bischof [1]. It starts with an initial set of seeds, and grows regions by merging a pixel into its nearest neighboring seed region. This algorithm is robust, fast, and free of tuning parameters [11], these characteristics allow the implementation of a very competitive algorithm which can be applied to a large variety of images. However, SRG algorithms require an automatic seeds generation process.

3.1.2 Segmentation definition

Previous work have been done to provide a formal definition of image segmentation [31], [13]. Because of the numerous fields of application there are several formal definitions for segmentation problem. A very general definition, used in [45] and [16] can be expressed as follows. Let R represent the entire image region (complete image). The segmentation can be viewed as a process that partitions R into n sub-regions, $R_1, R_2, \ldots, R_n$ such that:

1. $\bigcup_{i=1}^{n} R_i = R$
2. R_i is a connected region, $i = 1, 2, \ldots, n$.
3. $R_i \cap R_j = \oslash$ for all i and j, $i \neq j$.
4. If $P(R_i) = TRUE$ then $\forall j \neq i \; P(R_j) = FALSE$.

5. $P(R_i \cup R_j) = FALSE\, \forall i \neq j$.

Here, $P(R_i)$ is a logical predicate defined over the points in set R_i and $\oslash$ is the empty set.

Condition (1) indicates that the segmentation must be complete (exhaustive), which means that every pixel must belong to a region. Condition (2) requires that points in a region must be connected in any predefined sense. Condition (3) indicates that the regions must be mutually exclusive (overlapping between regions is not allowed). Condition (4) deals with the properties that must be satisfied by the pixels in a segmented region, for example, $P(R_i) = TRUE$ if all pixels in R_i have the same grey level. Finally, condition (5) indicates that regions R_i and R_j are different in the sense of predicate P.

This definition is also called *hard segmentation* because, unlike other segmentation approaches (like *fuzzy segmentation*), overlapping between regions is not allowed. Hard segmentation has been chosen for this thesis because the segmentation results are going to be used in hard classification tasks, where the examples must be labeled with only one class.

3.1.3 The seeded region growing algorithm

The seeded region growing algorithm performs a segmentation of an image with respect to a set of points, known as seeds. The seeded region growing algorithm needs n seeds sets $\{A_1\}, \{A_2\}, \ldots, \{A_n\}$. The decision of which is a feature of interest is embedded in the choice of the seeds [1]. At least one seed must exist for each region of interest. S is a multiset that contains all the seeds sets $\{A_i\}$:

$$S = \{\{A_1\}, \{A_2\}, \ldots, \{A_n\}\}$$

The process begins with the original seeds (the initial status of the sets $A_1, A_2, \ldots, A_n$). Each step of the algorithm involves the addition of one pixel to any of the sets A_i, $i \in \{1, 2, \ldots, n\}$.

Let T be the set of all unallocated (unlabeled) pixels that border at least one A_i region after m iterations:

$$T = \left\{ x \notin \bigcup_{i=1}^{n} A_i | N(x) \cap \bigcup_{i=1}^{n} A_i \neq \oslash \right\}$$

where $N(x)$ is the second-order neighborhood (8-neighbors) of pixel x. If we have that $N(x)$ intersects only one labeled region A_i, then, we define the label $i(x) \in \{1, 2, \ldots, n\}$ to be an index such that:

$$N(x) \cap A_{i(x)} \neq \oslash$$

Which means that the label i is assigned to the pixel x. If we have that $N(x)$ meets two or more regions A_i then we define $\delta(x, A_i)$ to be a measure of how different is x from a region A_i that $N(x)$ intersects:

$$\delta(x, A_i) = |g(x) - mean_{y \in A_i}(g(y))|$$

where $g(x)$ is the gray value of pixel x. The value of $i(x)$ will be the value of i such that $N(x)$ meets A_i and $\delta(x, A_i)$ is minimum. This completes step $m + 1$. The process must be repeated until all the unallocated pixels have been labeled.

3.1.4 Semi-interactive image segmentation

In multispectral satellite image segmentation it is hard to find an unambiguous separation of a given scene. In remote sensing applications, the objects or classes to be delineated depend on the application, and often the human interpreter judge which objects are significant and which are not. In this kind of applications, even a minimal level of human participation is always required. In traditional methods (such as parametric methods), human control has been reduced to parameter adjustment. Once the analyst has found a process that works for some data (commonly the training set), the process is applied to a larger number of similar images.

One simple and easy control of the segmentation process on multispectral satellite imagery can be this: the domain expert or analyst chooses the seed points, for example, by a mouse-based point-and-click mechanism, and seeded region growing completes the segmentation. This process is called semi-interactive image processing, as it is neither fully automatic nor fully manual. It has the advantage of having a human operator as quality control.

The semi-interactive seeded region growing method has the disadvantage that the final segmentation will fail if a seed falls on a noisy point, when the user makes an error in the seeds selection. In the proposed algorithm of this thesis, the semi-interactive process is performed through automatic seed generation and ownership tables. Automatic seed generation avoids problems derived from an incorrect seed selection from the user. Once the segmentation is finished, the user must generate tables that indicate which regions form a determinate concept, in other words, which regions must be merged to obtain the class of interest.

3.2 Instance-based learning

This section presents instance-based learning. Section 3.2.1 is an introduction, Section 3.2.2 provides the learning task and framework of instance-based learning, Section 3.2.3 describes the IB1 algorithm and section 3.2.4 describes the IB*K* algorithm.

3.2.1 Introduction

Up to now, different representations have been used to describe concepts for supervised learning tasks. These representations include, among others, decision trees [33], neural nets [44], and decision rules [5]. These approaches construct a specific abstraction of training examples, and this abstraction is used to classify unlabeled instances.

Instance-based learning (IBL) algorithms are derived from the nearest neighbor pattern classifier [9]. They are highly similar to edited nearest neighbor algorithms [20], which also save and use only selected instances to generate classification predictions. Edited nearest neighbor algorithms are non-incremental, and their primary goal is to maintain perfect consistency with the initial training set. Although they summarize data, they do not attempt to maximize classification accuracy on novel instances. This ignores real world problems such as noise. Instance-based learning algorithms are instead incremental and their goals include maximizing classification accuracy on subsequently presented instances. Further discussion on how IBL algorithms can solve real-world problems can be found in [2].

The design of IBL algorithms was also inspired by exemplar-based models of categorization [41]. Exemplar-based models are one of three proposed models of categorization in the psychological literature (the others are the classical and probabilistic models). Case-based reasoning (CBR) systems have been introduced to solve diagnosis and other problems [42]. Like IBL algorithms, these systems use previously processed cases to focus problem-solving activity on new cases. However, CBR systems also modify cases and use parts of cases during problem solving.

Instance-based learning algorithms are conceptually straightforward approaches to approximating real-valued or discrete-valued target functions [26]. In these algorithms, the learning step consists of storing the presented training data. When a new query instance is found, a set of similar or related instances is retrieved from memory and used to classify the new query instance. The main difference between instance-based learning and the classical approaches such as decision trees or decision rules is that instance-based learning can construct a distinct approximation to the target function for each instance that must be classified. This flexibility has significant advantages when the target function is very complex [2].

One disadvantage of instance-based learning algorithms is that the computational cost of classifying new instances can be high; this is due to all the computation that takes place during classification. An efficient indexing of the training examples is a significant practical issue to reducing the required computation at query time. Other disadvantage of instance-based learning algorithms is that they are very sensitive to noisy features. In this thesis, to diminish this problem, a new noise-tolerant weighting scheme [14] is proposed.

3.2.2 Learning task and framework

Instance-based learning addresses supervised learning (learning from examples). The approach focuses on the incremental variant of supervised learning in which the only input is a sequence of training instances.

Each instance is assumed to be represented by a set of feature-value pairs and all instances are described by the same set of n features. Missing feature values are allowed. This set of n features defines an n-dimensional instance space. Exactly one of the n features corresponds to the category attribute (class). The other features are considered as predicting features. A category is the set of all instances in an instance space that have the same value for their category attribute. In this thesis it is assumed that there is exactly one category attribute and that categories are disjoint.

The primary output of instance-based learning algorithms is a concept description. A concept description is a function that maps instances to categories: given an instance drawn from the instance space, it yields a classification, which is the predicted value for this instance's category feature.

An instance based concept description includes a set of stored instances and, possibly, some information concerning their past performance during classification. This set of instances can change after each training instance is processed. However, Instance-based learning algorithms do not construct extensional concept descriptions. Instead, on instance-based learning algorithm's, concept descriptions are determined by how the selected distance and classification functions are used in the current set of saved instances. These functions are two of the three components in the following framework that describes all instance-based learning algorithms [3]:

1. Distance function: This computes the distance between training instance i and the instances in the concept description. Distances have a numeric value.

2. Classification function: Receives the similarity function's results and the classification performance records of the instances in the concept description. It yields a classification for training instance i.

3. Concept description updater: Maintains records about classification performance and decides which instances to include in the concept description. Inputs include i, the similarity results, the classification results, and a current concept description. It yields the modified concept description.

The similarity and classification functions determine how the set of saved instances in the concept description are used to predict values for the category feature. Therefore, an instance-based learning concept description do not only contains a set of instances, but also includes these two functions.

Instance-based learning algorithms assume that similar instances have similar classification and also assume that, without prior knowledge, attributes will have equal relevance for classification decisions. This bias is achieved by normalizing each feature range of possible values.

The main difference among instance-based learning algorithms and other supervised learning methods is that instance-based learning algorithms does not construct explicit abstractions such as decision trees [33] or rules [5]. Most learning algorithms derive generalizations from training instances and use simple matching procedures to classify subsequently presented instances. Instance-based learning algorithms perform little work since they do not store nor create explicit generalizations. However, their work load is higher when subsequent instances are classified, when they compute the similarities of their saved instances with the newly presented instance.

3.2.3 The IB1 algorithm

The IB1 algorithm, described in Algorithm 1, is the simplest instance-based learning algorithm. The similarity function used here is

$$Sim(x,y) = -\sqrt{\sum_{i=1}^{n} f(x_i, y_i)}$$

where the instances are described by n attributes. We define $f(x_i, y_i) = (x_i - y_i)^2$ for numeric-valued features, and, for Boolean and symbolic-valued features $f(x_i, y_i)$ is defined as follows:

$$f(x_i, y_i) = \begin{cases} 1 & \text{if } x_i \neq y_i \\ 0 & \text{otherwise} \end{cases}$$

Missing feature values are assumed to be maximally different from the value present in the compared instance. If they are both missing, then $f(x_i, y_i)$ yields 1.

Algorithm 1 IB1 Algorithm (*CD* = Concept Description)

```
$CD \leftarrow$ all the training instances
$MaxSimilarity \leftarrow 0$
for all $x \in TestSet$ do
    for all $y \in CD$ do
        $Similarity \leftarrow Sim(x, y)$
        if $Similarity > MaxSimilarity$ then
            $MaxSimilarity \leftarrow Sim(x, y)$
            $YMax \leftarrow y$
            $AssignedClass \leftarrow class(y)$
        end if
    end for
    $CD \leftarrow CD + x$
end for
```

It is instructive to see an example of how IB1's concept description changes over time. This requires understanding how the similarity and classification functions of IB1 yield an extensional concept description from the set of saved instances. Since the nearest neighbor classification function simply assigns classifications according to the nearest neighbor policy, we can determine which instances in the instance space will be classified by each of the stored instances.

As an example, consider an instance space defined by two numeric dimensions, where 100 training instances are randomly selected from a uniform distribution and the target concept consists of four disjoints. Fig. 3.1 shows both IB1's approximation

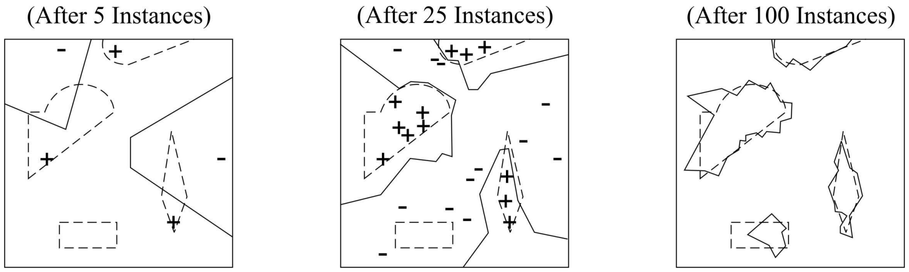


Figure 3.1: The extension of IB1's concept description, denoted by the solid lines, improves with training. Dashed lines delineate the four disjoints of the target concept.

of the target concept and the set of instances saved at three different moments during training. The predicted boundaries for the target concept, delineated using solid lines in Fig. 3.1, form a Voronoi diagram [37] which completely describes the classification predictions of the IB1 algorithm. Each predicted boundary of the target concept lies halfway between a pair of adjacent positive and negative instances. All instances on the same side of a boundary are predicted to be positive (i.e., members of the target concept). All other instances are predicted to be negative (i.e., nonmembers). These pictures indicate that IB1's approximation of the target concept description (dashed lines) improves as training continues (example and figure taken from [3]).

3.2.4 The IB*K* algorithm

The IB-*K* algorithm is an extension of the IB1 algorithm. In this thesis we use the IB-*K* algorithm as the base for the proposed learning algorithm. IB-*K* takes into account the *K* most similar instances to yield a class. IB-*K* algorithm, for the purposes of this thesis, instead of a similarity function, uses a distance function because the application domain contains only numerical-valued features without missing values. The used distance function is:

$$Dist(x, y) = \sqrt{\sum_{i=1}^{n} f(x_i, y_i)}$$

where x is a test instance, y is a training instance, x_i is the value of the i-th attribute of instance x and $f(x_i, y_i)$ is defined as follows:

$$f(x_i, y_i) = (x_i - y_i)^2$$

The instances are described by n features.

The IB-*K* algorithm is presented in Algorithm 2. To label an instance, the IB-*K* algorithm computes the distance between the test instance and the instances stored in the concept description and stores the *K* nearest instances. The class of the test instance will be the preponderant class of the *K* nearest instances previously obtained.

Algorithm 2 IB-*K* algorithm (*CD* = concept description)

```
$CD \leftarrow$ all the training instances
for all $x \in TestSet$ do
    for all $y \in CD$ do
        $Distance \leftarrow Dist(x, y)$
        if $Distance <$ of one of the $k$ lower distances stored in $KMinDistances[]$ then
            $KMinDistances[k] \leftarrow Distance$
            $YMin[k] \leftarrow y$
            $KAssignedClass[k] \leftarrow class(y)$
        end if
    end for
    $class(x) \leftarrow$ Preponderant class in $KAssignedClass[k]$
    $CD \leftarrow CD + x$
end for
```

In the next chapter, related work is presented. The first section is about machine learning algorithms used for multispectral satellite image segmentation and the second section is about segmentation algorithms based on seeded region growing.

Chapter 4

Related work

In this chapter, related work to this thesis is presented. Section 4.1 describes work that uses machine learning tools for multispectral image segmentation for remote sensing applications. Section 4.2 introduces general image segmentation algorithms based on seeded region growing.

4.1 Machine learning tools used for multispectral satellite image segmentation for remote sensing applications

This section describes two algorithms that use machine learning tools for multispectral image segmentation, the first algorithm performs segmentation of remote-sensing images by an incremental neural network proposed by M. Kurnaz et al. and the second algorithm segments multispectral remote sensing images using active support vector machines proposed by P. Mitra et al.

4.1.1 Segmentation of remote-sensing images by incremental neural network

M. Kurnaz et al. [24] proposed an incremental neural network algorithm (INeN) for remote-sensing image segmentation. The INeN is a two-layer network, the nodes in the first layer are formed by the vectors of features, and the second layer stores the labels of the output nodes. Two feature extraction methods are examined comparatively on the paper. In the first method, features correspond to the intensity of one pixel in each channel; in the second method features are formed by the intensity of a pixel along with its 8 neighbors in each channel. During the training of the INeN, the number of classes is determined automatically depending on a threshold value, given by a domain expert. Its learning algorithm computes the Euclidean distances between the nodes of the INeN and the input feature vector, and these distances are compared with the threshold. If the distance is lower than the threshold, the weight of the node nearest to the input vector is modified, if not, a new node is added to the network. The overview of the algorithm is presented next.

Algorithm overview

The algorithm first determines a pixel location and extracts the intensity of the pixels from the same location in each band of the image, then creates the feature vector associated with this pixel location and computes the distances between the feature vector and the nodes in the first layer. If the minimum distance is higher than the threshold, it adds the feature vector to the INeN as a new node in layer one, increments the index counter by one, and assigns the value in the counter as the index (label) of the new node, otherwise, it increases the *usage counter* of the node by one to form a histogram and modifies the weights of the node nearest to the input vector according to:

$$g_{ji}(k+1) = g_{ji}(k) + \mu \cdot (x_i(k) - g_{ji}(k))$$

where g_{ji} is the ith weight of the jth node nearest to the input vector X, and μ is the

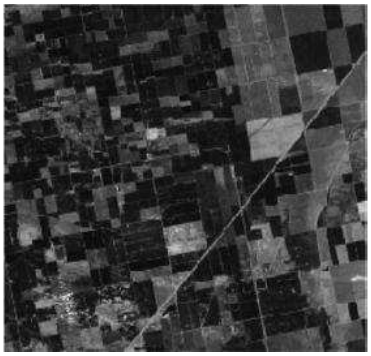


Figure 4.1: Original image (Image toked from [24].)

gain constant. Finally, go to the first step until all the pixels in the images have been labeled.

Results reported

The authors only report a visual comparison with Landsat-5 multispectral satellite images. The comparison is made against a segmentation obtained by means of a Kohonen neural network, varying the threshold value and the feature extraction method. Fig. 4.1 shows the first band of the original image. Fig. 4.2 shows a comparison against a segmentation based on a Kohonen neural network. Figures in the left column were obtained with the first extraction method, figures in the right column were obtained with the second extraction method. Figures A and B show the segmentation obtained by the Kohonen neural network, figures C, D, E and F show the segmentation obtained by the INeN. The selected threshold value for each image segmented with the INeN is 4500 (C), 3000 (D), 20 000 (E) and 18 000 (F). Threshold values were obtained experimentally.

Remarks

The authors present a novel incremental approach applied over a neural network. The algorithm seems to be efficient, however, based on the visual comparison presented it is hard to know which algorithm performs better. The proposed algorithm is also very

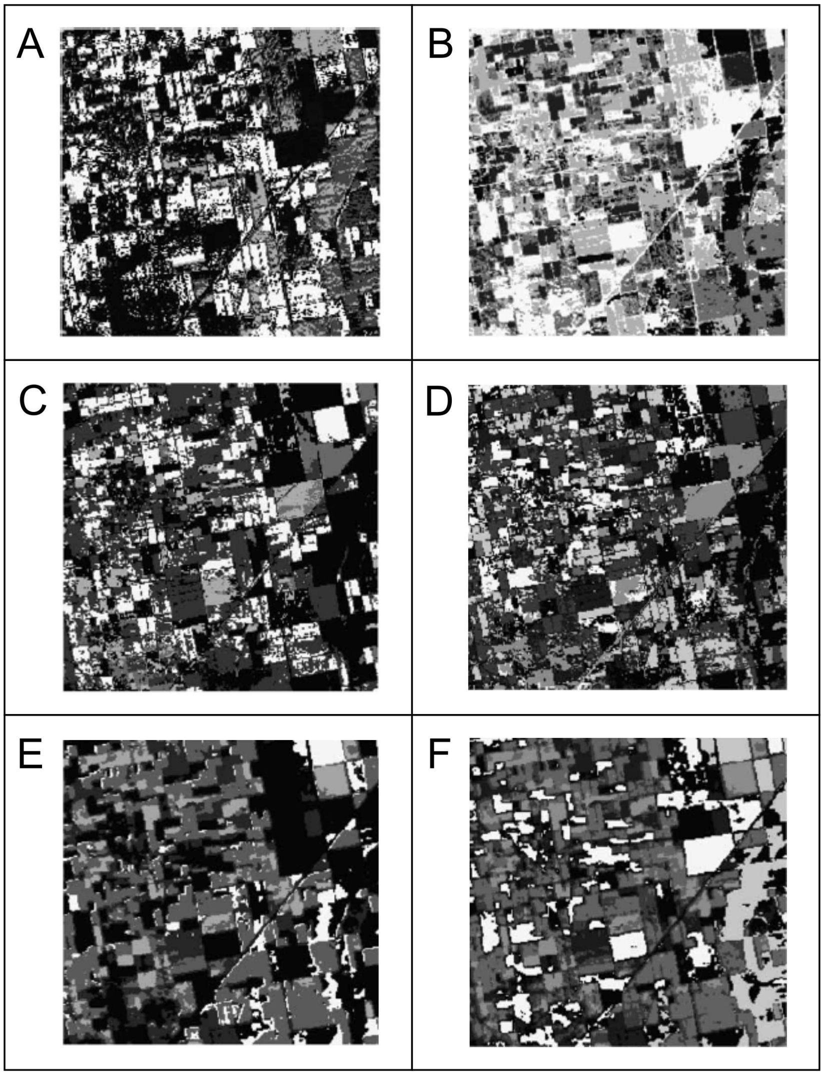


Figure 4.2: Original image (Images toked from [24].)

sensitive to the threshold value, an incorrect estimation of the threshold results in an over-segmentation or in a sub-segmentation, and the algorithm does not have built-in methods such as region merging or region splitting to overcome this problem. The proposed algorithm in this thesis is parameter-free and has a region merging scheme based on ownership tables to control the final output of the segmentation process.

4.1.2 Segmentation of multispectral remote sensing images using active support vector machines

P. Mitra et al. [27] present an active support vector learning algorithm for supervised pixel classification in remote sensing images. The goal of the work is to minimize the number of labeled points required to design the classifier. The algorithm uses an initial small set of labeled pixels to design a crude classifier, which is subsequently refined by actively querying for the labels of pixels from a pool of unlabeled data. The SVM finds the most interesting (closest of the current separating hyper surface) unlabeled pixels and asks the domain expert for a label. The algorithm employs active learning to minimize the number of labeled data used by the SVM classifier.

Algorithm overview

Let $x = [x_1, x_2, \ldots, x_d]$ represent a pixel of a d-band multispectral image, x_i represents the grey value of the ith band for pixel x, $A = \{x_1, x_2, \ldots, x_{l_1}\}$ denotes the set of pixels for which class labels are known, $B = \{x_1, x_2, \ldots, x_{l_2}\}$ denotes the set of pixels for which class labels are unknown ($l_2 >> l_1$), $SV(C)$ denotes the set of support vectors of any set C, $S_t = \{s_1, s_2, \ldots, s_m\}$ is the support vector set obtained after the ith iteration, $< w_t, b_t >$ is the corresponding separating hypersurface and Q_t is the point actively queried for at step t. First, set $t = 0$ and $S_0 = SV(A)$. The parameters of the corresponding radial basis function are $< W_0, b_0 >$. While the stopping criterion is not satisfied, obtain $Q_t = \{x | min_{x \in B} k(w_t, x)\} + b$, request for the label of Q_t, and update the sets $S_t = SV(S_t \bigcup Q_t)$, $B = B - Q_t$ and $t = t + 1$. The set S_t, where T is the

Table 4.1: Comparative Results of the SVM algorithm (Table toked from [27])

Method	$n_{labeled}$	$t_{training}(S)$	β
Active SVM	259	72.02 + time for labeling 54 pixels	6.35
SVM	198	28.15	3.45
k-means	0	1054.10	2.54

iteration at which the algorithm terminates, contains the final SV set representing the classifier. Training is stopped when none of the unlabeled points lie within the margin of the separating hypersurface $min_{x \in B} k(w_t \cdot x) + b > 1$.

Results reported

The authors present comparisons against the conventional support vector machines algorithm and against k-means. The comparison is presented in terms of labeled data needed to train the algorithm, training time and the quantitative cluster quality index (β), a cluster quality measure proposed by Pal et al. [30] Results are presented in Table 4.1.

Remarks

The active SVM algorithm provides a higher β value than the original support vector machines and K-means algorithms, which means that the clusters produced by the active SVM algorithm are more homogeneous, however, the algorithm depends of the domain expert through all the execution, and sometimes, even for an expert, labeling a single pixel can be a difficult task. The algorithm is also sensitive to wrong labeling, resulting in high performance degradation.

4.2 Algorithms based on seeded region growing

This section describes three algorithms based on seeded region growing for color image segmentation, the first work is Seeded Region Growing: an Extensive and Comparative Study performed by J. Fan et al., the second algorithm is Automatic Seeded Region Growing for Color Image Segmentation proposed by F. Y. Shih and S. Cheng and the third algorithm is A Self-Calibrating Multi-Band Region Growing Approach to Segmentation of Single and Multi-Band Images proposed by D. W. Paglieroni.

4.2.1 Seeded region growing: an extensive and comparative study

J. Fan et al. [11] proposed an automatic SRG with a boundary-oriented parallel pixel labeling framework and an automatic seeds selection method. The objective of the work is to perform automatic image segmentation for content-based image description and retrieval. The authors proposed three methods to automatically generate seeds. The first method partitions the image into a set of rectangular regions with fixed size and takes the centers of these rectangular regions as seeds. The second method first obtains a set of initial seeds from the centroid of the color edges of the image and later the neighboring similar seeds are merged. Finally, the third method is similar to the second, one difference is that an image initial filtering procedure is applied. Once the seeds are obtained, a parallel pixel labeling approach is applied.

Algorithm overview

The pixels in the same region are labeled with the same symbol. The regions are represented via two parameters: the centroid of the region and the boundary pixels. These two region description parameters are updated by adding new pixels on each step. The SRG procedure starts with all the seeds at the same time. For a seed region R_i with a set of boundary pixels $B_{R_i} = \{(x_l, y_l) | l \in [1, \ldots, L]\}$, the second-order (8 neighbors) neighboring pixels $(x_l \pm 1, y_l \pm 1)$ of the boundary pixel (x_l, y_l) must be labeled. If the unlabeled pixel at $(x_l \pm 1, y_l \pm 1)$ is similar to the adjacent boundary pixel (x_l, y_l)

of region R_i, then they are merged into the R_i region, and also replaces the boundary pixel (x_l, y_l) as the new boundary pixel of region R_i. This similarity testing and labeling procedure can be performed at the same time for all the boundary pixels for the same region. The color similarity distance $D(x_l, y_l, R_i)$, between the unlabeled pixel $(x_l \pm 1, y_l \pm 1)$ and the current testing boundary pixel (x_l, y_l) of region R_i is calculated as:

$$D(x_l, y_l, R_i) = \alpha + \beta + \gamma$$

$$\alpha = |I(x_l, y_l) - I(x_l \pm 1, y_l \pm 1)|$$

$$\beta = |u(x_l, y_l) - u(x_l \pm 1, y_l \pm 1)|$$

$$\gamma = |v(x_l, y_l) - v(x_l \pm 1, y_l \pm 1)|$$

where $I(x_l, y_l)$, $u(x_l, y_l)$, and $v(x_l, y_l)$ indicate the values of the three color components of the testing boundary pixel (x_l, y_l). $I(x_l \pm 1, y_l \pm 1)$, $u(x_l \pm 1, y_l \pm 1)$, and $v(x_l \pm 1, y_l \pm 1)$ represent the values of three color components of the unlabeled pixel $(x_l \pm 1, y_l \pm 1)$ which is adjacent to the boundary pixel (x_l, y_l). If the unlabeled pixel $(x_l \pm 1, y_l \pm 1)$ meets two or more of the labeled boundary pixels of the region R_i, it is merged into the region R_i and replaces the most similar boundary pixel (x_m, y_m) of the region R_i:

$$D(x_m, y_m, R_i) = \min_{x_l, y_l \in B_{R_i}} \{D(x_l, y_l, R_i) | l \in \{1, \ldots, L\}\}$$

If the unlabeled pixel meets two or more boundary pixels from adjacent regions, it is merged into the region R_j which has the smallest similarity distance and replaces the most similar boundary pixel as the new boundary pixel of region R_j:

$$D(x_k, y_k, R_j) = \min_{i \in \{1, \ldots, q\}} \{D(x_m, y_m, R_i) | (x_m, y_m) \in B_{r_i}\}$$

Results reported

The authors presented visual results over images from the Corel database for content-based image retrieval, medical, and people images. Fig. 4.3 shows the results of the

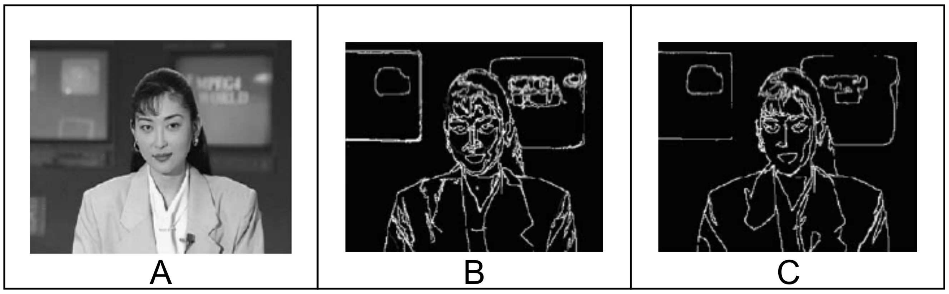


Figure 4.3: Results of the algorithm over the *Akiyo* image (Images toked from [11].)

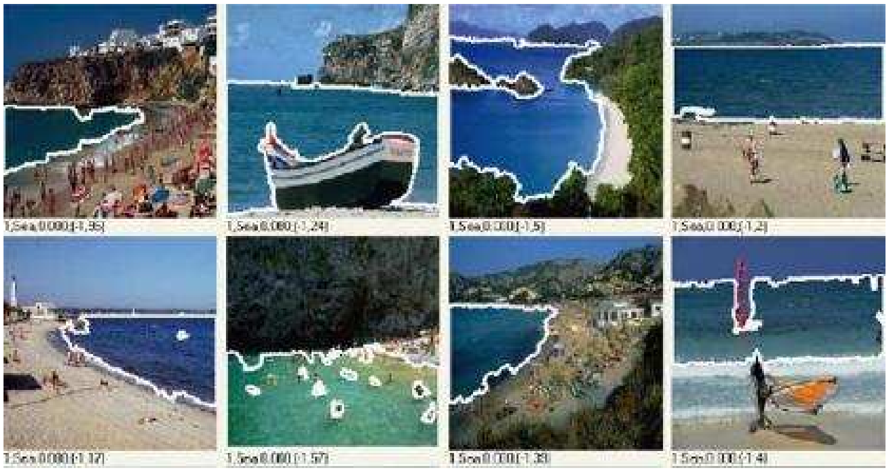


Figure 4.4: Results of the algorithm over some images of the Corel database (Images toked from [11].)

segmentation algorithm on the *Akiyo* image, (A) is the original image, (B) is the image segmented with the second method of seeds generation and (C) is the image segmented with the third method. Fig. 4.4 shows the results of the algorithm applied over images of the Corel database for content-based image retrieval with the third method of seed generation.

Remarks

The main contributions of the authors are a parallel framework for the region growing algorithm (which can be applied over many SRG-based algorithms) and three methods to automatically generate seeds. The automatic seed generation methods are good to find a general segmentation, without much detail. This is because the objective of the segmentation task is content-based image retrieval. This algorithm can not be applied if a detailed segmentation is needed because image retrieval applications do not require a highly detailed segmentation. The algorithm can not be applied over images without certain level of definition on their boundaries because the seed generation method that the algorithm employs is based on boundaries, and without these boundaries the algorithm can not find good seeds. In remote sensing, it is possible that some regions do not have well delimited boundaries, due to this limitation, the algorithm can not be successfully applied to remote sensing images.

4.2.2 Automatic seeded region growing for color image segmentation

F. Y. Shih and S. Cheng [39] present an automatic seeded region growing algorithm for color image segmentation that works over the YC_bC_r color space. First, the color image is transformed from RGB to YC_bC_r color space. Second, an automatic seeds selection method is applied to obtain initial seeds. Third, the seeded region growing algorithm is used to segment the image into regions, where each region corresponds to one seed. Finally, similar regions are merged, and small regions are merged into their nearest neighboring regions.

Algorithm overview

Seeds selection algorithm. Considering a 3 x 3 neighborhood, the standard deviations of the Y, C_b and C_r components are calculated using:

$$\sigma_x = \sqrt{\frac{1}{9}\sum_{i=1}^{9}(x_i - \overline{x})^2}$$

where x can be Y, C_b or C_r, and the mean value $\overline{x} = \frac{1}{9}\sum_{i=1}^{9} x_i$. The total standard deviation is calculated with:

$$\sigma = \sigma_y + \sigma_{C_b} + \sigma_{C_r}$$

The standard deviation must also be normalized with:

$$\sigma_N = \sigma / \sigma_{max}$$

where σ_{max} is the maximum of the standard deviations in the image.

The similarity of a pixel with its neighbors is defined as

$$H = 1 - \sigma_N$$

A seed pixel must comply with two conditions. The first condition is that a candidate seed pixel must have a similarity value higher than a threshold value. The second condition is that a candidate seed pixel must have the maximum relative Euclidean distance (MRED) to its eight neighbors lower than a threshold value. The relative Euclidean distances (in terms of YC_bC_r components) of a pixel to its eight neighbors are:

$$d_i = \frac{\sqrt{(Y - Y_i)^2 + (C_b - C_{b_i})^2 + (C_r - C_{r_i})^2}}{\sqrt{Y^2 + C_b^2 + C_r^2}}$$

$$i = 1, 2, \ldots, 8.$$

For each pixel, the maximum distance to its neighbors must be calculated with:

$$d_{max} = \max_{i=1}^{8}(d_i)$$

A pixel is considered seed if it satisfies conditions one and two.

Segmentation algorithm. $A_1, A_2, \ldots, A_i$ are the initial seeds, S_i is the region corresponding to A_i, $(\overline{Y}, \overline{C_b}, \overline{C_r})$ are the mean of all seed pixels in S_i in terms of the Y, C_b and C_r components. The algorithm proceeds as follows. First, the automatic seed selection must be carried out. Second, a label is assigned to each seed region, then, the neighbors of all regions must be recorded in a sorted list T in a decreasing order of distances. While T is not empty, remove the first point p and check its 4-neighbors. If all labeled neighbors of p have the same label, set p to this label. If the labeled neighbors of p have different labels, calculate the distances between p and all neighboring regions and classify p to the nearest region. Then update the mean of this region, and add 4-neighbors of p (which are not classified yet or are not in T) to T in a decreasing order of distances. Finally, merge regions.

Region merging algorithm. Two criteria are used for the merging step: one is the similarity and the other is the size. If the mean color difference between two neighboring regions is less than a threshold, the regions are merged and the mean of the new region is re-computed along with its neighboring regions. This process is repeated until no region has a distance less than the threshold. The color difference between two adjacent regions R_i and R_j is defined as the relative Euclidean distance :

$$d(R_i, R_j) = \frac{\xi}{\psi}$$

$$\xi = \sqrt{(\overline{Y}_i - \overline{y}_j)^2 + (\overline{C}_{b_i} - \overline{C}_{b_j})^2 + (\overline{C}_{r_i} - \overline{C}_{r_j})^2}$$

$$\psi = \min(\chi, \phi)$$

$$\chi = \sqrt{\overline{Y}_i^2 + \overline{C}_{b_i}^2 + \overline{C}_{r_i}^2}$$

$$\phi = \sqrt{\overline{Y}_j^2 + \overline{C}_{b_j}^2 + \overline{C}_{r_j}^2}$$

Finally, the size of the regions is checked. If the number of pixels in a region is smaller than a threshold, the region is merged into its neighboring region with the smallest color difference.

Results reported

The authors present visual results of the segmentation algorithm over many color nature scene images randomly collected from the Internet. The comparison was done against the JSEG algorithm [10]. Fig. 4.5 presents some of these results. The first column has the original images, the second column has the segmentation result of the algorithm proposed by the authors and the third column shows the resulting segmentation of the JSEG algorithm.

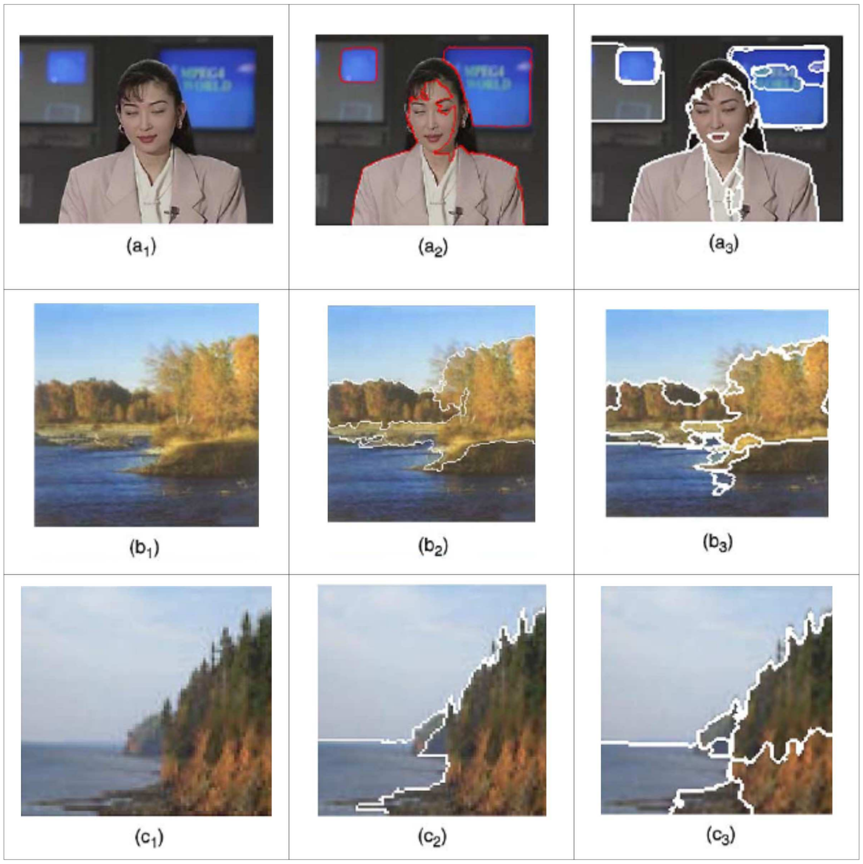


Figure 4.5: Results of the algorithm over images randomly collected from the Internet (Images toked from [39].)

Remarks

The main contribution of the authors is the adaptation of the seeded region growing algorithm to the Y, C_b, C_r color space, along with a method for automatic seeds selection and a threshold based region-merging. The major disadvantage of this algorithm is that it can only be applied over 3-band RGB images; this limitation excludes the possibility to apply this algorithm to multispectral satellite images, which commonly have more than three spectral bands.

4.2.3 A self-calibrating multi-band region growing approach to segmentation of single and multi-band images

D. W. Paglieroni [29] proposed a generalization of the classical region growing approach for single and multi-band images. The author also proposes a self-calibrating framework for automatic parameter selection to produce a segmentation that resembles a calibration edge map. In order to perform the generalization of the SRG algorithm, he proposes a local search unitary operator $\Omega_{i,j}(i', j')$ to grow regions from seed pixels (i, j). The operator is formed by two components: a spectral one and a spatial one. The spectral component obtains the spectral distance of the pixel while the spatial component obtains its neighborhood. The proposed self calibration-method framework measures the disparity between region maps and calibration edge maps to automatically adjust the parameter settings; segmentation is performed at each of several parameters settings, and the parameters that produce the most consistent region map (with the calibration edge map) are used.

Algorithm overview

Growing algorithm. MBRG segments images by employing a local search operator $\Omega_{i,j}(i', j')$, modeled as the intersection of two sets, one that imposes spectral constraints, and another that imposes spatial constraints on the unlabeled pixels (m, n)

that are assigned to $\Omega_{i,j}(i', j')$.

The spectral constraint set (spectral component) $\Omega_{spectral}(i, j)$ is based on spectral distances between a seed pixel (i, j) and pixels $(m, n) \neq (i, j)$. Consider a multi-band image with K spectral bands that contain pixel spectra $x_{i,j} \triangleq [x_{i,j}(0), \ldots, x_{i,j}(K-1)]^T$ with K spectral samples. $x_{i,j}(k)$ will refer to spectral sample k (spectral band indexes) of the pixel with $(row, column)$ coordinates (i, j). The squared spectral distance between pixels (i, j) and (m, n) can be defined as the squared norm of the difference between the spectral vector averaged over all bands, namely:

$$d^2_{i,j}(m, n) = ||x_{i,j} - x_{m,n}||^2/K$$

For MBRG:

$$\Omega_{spectral}(i, j) = \{(m, n) : d_{i,j}(m, n) < d*\}$$

where $d*$ is a spectral distance threshold.

The spatial constraint set (spatial component) of the local search operator for MBRG is:

$$\Omega_{spatial}(i', j') = \{(m, n) \neq (i', j') : |m - i'|, |n - j'| \leq w\}$$

This is the set of pixels in the square local search neighborhood of width $2w + 1$ centered on pixel (i', j'), exclusive of (i', j').

Calibration algorithm. Consider a region map R and an associated binary border map B in which pixels of value 1 correspond to borders between different regions. Let E be the calibration edge map with edge pixels of value 1 on a background of zeros. The disparity Δ_{BE} between R and E is given by:

$$\Delta_{BE} = \begin{cases} 1 & n_B = n_E = 0 \\ 0 & n_B \text{ or } n_E = 0 \text{ but not both 0} \\ (n_{BE} + n_{EB})/(n_B + n_E) & n_B, n_E \neq 0 \end{cases}$$

where n_B is the number of boundary pixels in B, n_E is the number of edge pixels in E, n_{BE} is the number of boundary pixels in B that are not associated with an edge pixel

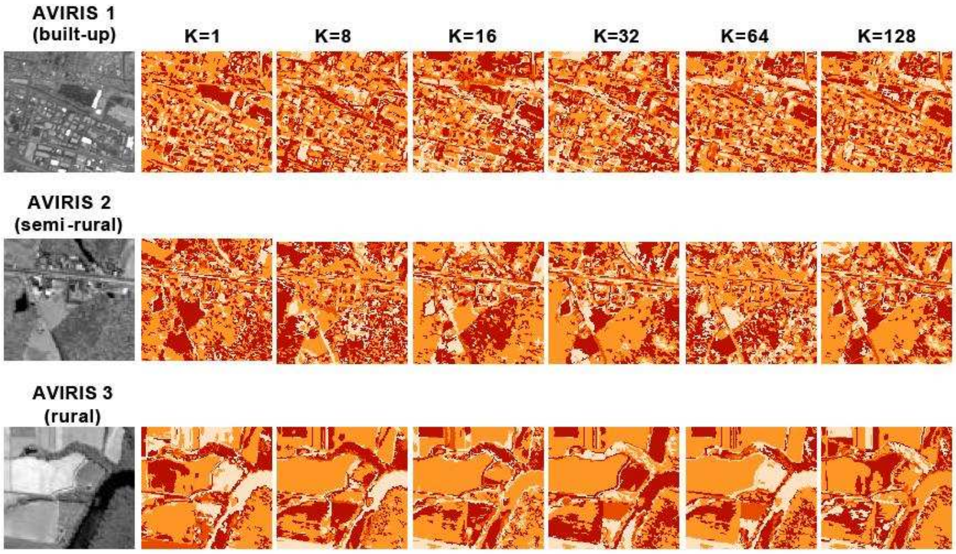


Figure 4.6: Results of the MBRG algorithm with different parameter settings (Images toked from [29].)

in E, and n_{EB} is the number of edge pixels in E not associated with a boundary pixel in B. The parameter setting that produces the lower disparity Δ_{BE} between R and E is selected.

Results reported

The author show results over hyperspectral AVIRIS images of 128 x 128 pixels with 224 bands. The images were preprocessed by removing all regions with less than 5 pixels. The author only presents visual results, and no comparison is made against other segmentation algorithms. Fig. 4.6 shows the best segmentation over three images, buildings, semi-rural and rural. K is the number of spectral bands considered in the segmentation.

Remarks

The major contribution of this work is the generalization of the seeded region growing algorithm to multi band image segmentation. A disadvantage of the algorithm is that the parameter setting calibration method can take a considerable amount of time to find the best parameter setting in a large image, and the proposed algorithm process only a segment of the image, assuming that the parameter setting found will be consistent with the rest of the image. In remote sensing images, this assumption can be erroneous because some land covers can be missed, and usually a large number of land covers are presented in satellite images. The domain expert is not involved in the process, and without an expert guide, it is possible to obtain undesirable results, since frequently distinct segmentation are desired over the same image (at different levels of detail). The author made a very simple evaluation of his method, without performing any comparison against other algorithms, and, on visual comparisons it is normally hard to see all differences to determine which is the best parameter setting.

Chapter 5

The SRG-WIBK algorithm

In this chapter, the proposed algorithms are explained. Section 5.1 is an overview of the algorithm, Section 5.2 describes the proposed automatic seed generation algorithm, Section 5.3 explains the proposed region growing algorithm based on weighted instance-based learning, Section 5.4 describes the proposed weighted instance-based learning algorithm, and Section 5.5 explains the proposed method used for region merging based on ownership tables.

5.1 Algorithm overview

The proposed algorithm can be divided in three different steps: automatic seeds selection, region growing based on weighted instance-based learning, and finally region merging via ownership tables.

1. **Automatic seeds selection**. Seeds are labeled pixels required for the region growing algorithm to segment an image. In each step, seeds are growed by adding unlabeled pixels according to certain criteria such as Euclidean distance. In the proposed algorithm, seeds are automatically generated via histogram analysis. The histogram is a graph that shows the number of pixels that have certain gray value. The histogram is divided on intervals, and the intervals with more pixels

(called representative intervals) are stored. An image pixel is considered seed if all its gray values for each band are in some representative interval. The automatic seeds selection algorithm is described in Section 5.2.

2. **Region growing based on weighted instance-based learning**. After the automatic seeds selection step, the proposed seeded region growing algorithm is applied to segment the image. This algorithm uses instance-based learning as criteria to determine to which region (seed) a pixel belongs. On instance-based learning algorithms, when a new unlabeled instance is presented, training data is used to label the new instance by assigning the label of the most seemed training instance, according to a given criteria such as similarity or distance. The proposed instance-based learning algorithm features a novel weighting scheme which assigns a weight to each band according to its discriminative power. The gray values of each region (seed) are analyzed to find the intervals in which these vales are. The overlap among intervals of distinct regions on the same band determines the weight of the band; a higher overlap means a lower weight and a lower overlap means a higher weight. The region growing algorithm is explained in Section 5.3 and the weighting scheme is described in Section 5.4.

3. **Region merging via ownership tables**. At this stage, the image has been segmented (divided) into homogeneous regions. Because information about the image (high level knowledge) has not been used, is not possible to know what represents each segment (for example, if a region is water or grass). This information is provided by means of ownership tables, in which a meaning is assigned to each region. After the region growing step, each region has a unique number that identifies it called region ID, ownership tables assigns a meaning (class) to each region ID. Ownership tables also allows to merge two or more regions into a single class. Ownership tables are fully explained in Section 5.5.

Fig. 5.1 shows a diagram of the proposed SRG-WIBK algorithm.

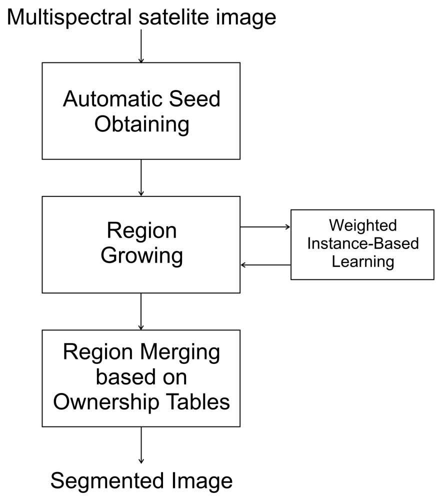

Multispectral satelite image
Automatic Seed Obtaining
Region Growing
Weighted Instance-Based Learning
Region Merging based on Ownership Tables
Segmented Image

Figure 5.1: Overview of the SRG-WIBK algorithm

5.2 Automatic seeds selection

An overview of the automatic seeds selection algorithm is shown in Fig 5.2. The algorithm is composed of four main stages. The first stage divides the histogram of each band in intervals. Let $h_b(p)$ be the histogram function, this function receives a gray value p $(0 \leq p \leq 255)$ and returns the number of pixels of band b with gray value equal to p. To divide the histogram *cut points* must be found. All the gray values p that satisfy the next two conditions will be considered as cut points:

1. $h_b(p-1) \geq h_b(p)$

2. $h_b(p+1) > h_b(p)$

The cut points indicate the end and the beginning of each interval. Table 5.1 shows the intervals S_j obtained from a given histogram function $h_b(p)$ with q cut points, where C_i is a cut point $(1 \leq i \leq q)$, S_j is a interval $(1 \leq j \leq m)$ and m is the number of resultant intervals. For example, in $h_b(1) = 10, h_b(2) = 11, h_b(3) = 12, h_b(4) = 9, h_b(5) = 10, h_b(6) = 8$ the cut points is the value 9 because completes the two conditions ($12 \geq 9$ and $10 > 9$). The algorithm is showed in Algorithm 3.

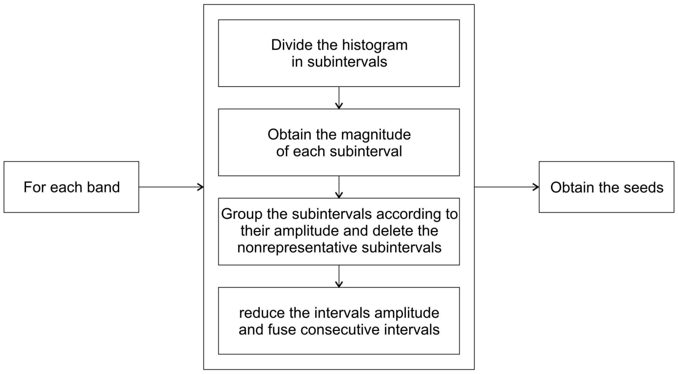

For each band
Divide the histogram in subintervals
Obtain the magnitude of each subinterval
Group the subintervals according to their amplitude and delete the nonrepresentative subintervals
reduce the intervals amplitude and fuse consecutive intervals
Obtain the seeds

Figure 5.2: Overview of the automatic seed generation algorithm

Table 5.1: intervals S_j obtained from a given histogram function $h_b(p)$ with q cut points

$S_1 = [0, C_1]$
$S_2 = [C_1 + 1, C_2]$
$S_3 = [C_2 + 1, C_3]$
$\dots$
$S_m = [C_q, 255]$

Algorithm 3 Cut points selection algorithm

```
while Index < 256 do
    Was ← Am
    if Histogram[Index] > Histogram[Index-1] then
        Am ← TRUE
    else
        if Histogram[Index] < Histogram[Index-1] then
            Am ← FALSE
        else
            Am ← Was
        end if
    end if
    if (Index = 255) OR (Was = FALSE AND Am = TRUE) then
        if Was = FALSE AND Am = TRUE then
            FinalIndex ← Index - 1
        else
            FinalIndex ← 255
            PixelCounter ← PixelCounter + Histogram[255]
        end if
        if MaximumMagnitude[(FinalIndex-InitialIndex)+1] < PixelCounter then
            MaximumMagnitude[(FinalIndex-InitialIndex)+1] ← PixelCounter
        end if
        Intervals[0, IntervalsIndex] ← InitialIndex
        Intervals[1, IntervalsIndex] ← FinalIndex
        IntervalsIndex ← IntervalsIndex + 1
    else
        PixelCounter ← PixelCounter + Histogram[Index]
    end if
    Index ← Index + 1
end while
```

The second stage obtains the amplitude of each interval. For a given interval $S_j = [C_{j-1}+1, C_j]$ the amplitude is given by:

$$amp(S_j) = (C_j) - (C_{j-1}+1) + 1$$

For example, for the interval $[6, 9]$ where $C_j = 9$ and $C_{j-1}+1 = 6$ the amplitude is $9 - 6 + 1 = 4$.

The third stage groups the intervals according to their amplitude, and stores the most representative intervals which are the intervals with more pixels. To obtain the most representative intervals, the magnitude of each interval must be computed. The magnitude of a interval S_j in the band b is:

$$mag(S_j) = \sum_{i=C_{j-1}+1}^{C_j} h_b(i)$$

For all intervals S_j with amplitude $amp(S_j) = \alpha$ the most representative interval is the one with the largest magnitude.

$$mag_{max}(\alpha) = \arg\max mag(S_j)$$

A interval S_j is non representative if its amplitude is lower or equal than half of the magnitude of the most representative interval. The value of $\frac{1}{2}$ was selected because of the quorum rule, where $\frac{1}{2} + 1$ is considered representative.

$$mag(S_j) \leq \frac{1}{2} mag_{max}(\alpha)$$

For example, if we have the intervals $S_1 = [9, 13], S_2 = [17, 21], S_3 = [34, 38], S_4 = [56 - 60]$ of amplitude 5 with magnitude $mag(S_1) = 34, mag(S_2) = 10, mag(S_3) = 25, mag(S_4) = 35$ then the largest magnitude is 35, the most representative interval is S_4 and the interval S_2 is non representative because $10 \leq (\frac{1}{2})(35)$.

The fourth step reduces the representative intervals amplitude. For a given representative interval $S_j = [C_{j-1}+1, C_j]$ of band b, the most representative gray value is:

$$g_{max}(S_j) = \arg \max_{\forall C_{j-1}+1 \leq \beta \leq C_j} h_b(\beta)$$

A gray value γ of a representative interval S_j of band b is representative if:

$$h_b(\gamma) > \frac{1}{2} g_{max}(S_j)$$

All the non representative gray values are removed from the interval, producing a reduced interval. For example, in an interval $S_j = [123, 126]$ with $h_b(123) = 18, h_b(124) = 20, h_b(125) = 25, h_b(126) = 7$ the most representative gray value is 125 and the gray value 126 is not representative because $7 \leq (\frac{1}{2})(25)$. The representative intervals selection and reduction algorithm is showed in Algorithm 4.

Depending on the application domain, consecutive resultant reduced intervals can be merged. For example, the reduced intervals [12-18], [19-25] produce the new merged interval [12-25]. Interval merging decreases the number of homogeneous seeds, and must be avoided if the application needs the highest separation among seeds (i.e., the user needs the maximum level of homogeneity in the regions). In this thesis, interval merging is always performed. The interval merging algorithm is described in Algorithm 5.

In the final step we generate the seeds. A pixel x is considered as a seed if its gray values on each band are in a representative interval. If all the gray values of two seed pixels are in the same representative intervals, the pixels will be labeled with the same region ID. The region ID is a number that identifies each region. The output of the seeds generator is a set with n seeds $A_1, A_2, \ldots, A_n$.

5.3 Region growing based on weighted instance-based learning

The automatically generated seeds are used to construct the classifier using the region ID as the class of the pixel. All the seeds must be grouped according to their region ID;

Algorithm 4 Representative intervals selection and reduction algorithm

```
InsertPosition $\leftarrow$ 0
for Index = 0 TO IntervalsIndex do
    if IntervalMagnitude[Index] $>$ MaximumMagnitude * 0.5 then
        HigherPosition $\leftarrow$ Intervals[0, Index]
        High $\leftarrow$ Histogram[HigherPosition]
        for interval = Intervals[0,Index]+1 TO Intervals[1,Index] do
            if Histogram[interval] $>$ High then
                High $\leftarrow$ Histogram[interval]
            end if
        end for
        NewBegin $\leftarrow$ Intervals[0, Index]
        NewEnd $\leftarrow$ Intervals[1, Index]
        while Histogram[NewBegin] $<$ High*0.5 AND Histogram[NewEnd] $<$ High*0.5 do
            if Histogram[NewBegin] $<$ High*0.5 then
                NewBegin $\leftarrow$ NewBegin + 1
            end if
            if Histogram[NewEnd] $<$ High*0.5 then
                NewEnd $\leftarrow$ NewEnd - 1
            end if
        end while
        Intervals[0, InsertPosition] $\leftarrow$ NewBegin
        Intervals[1, InsertPosition] $\leftarrow$ NewEnd
        InsertPosition $\leftarrow$ InsertPosition + 1
    end if
end for
```

Algorithm 5 Interval Merging algorithm

```
LeftLimit $\leftarrow$ Intervals[0,0]
RightLimit $\leftarrow$ Intervals [1,0]
for Index $=$ 0 TO IntervalsIndex do
    TemporalRightLimit $\leftarrow$ Intervals[0, Index]
    if RightLimit + 1 $=$ TemporalRightLimit then
        RightLimit $\leftarrow$ Intervals[1,Index]
    else
        Intervals[0, TemporalIndex] $\leftarrow$ LeftLimit
        Intervals[1, TemporalIndex] $\leftarrow$ RightLimit
        LeftLimit $\leftarrow$ TemporalRightLimit
        TemporalIndex $\leftarrow$ TemporalIndex + 1
    end if
end for
Intervals[0, TemporalIndex] $\leftarrow$ LeftLimit
RightLimit $\leftarrow$ Intervals[1, IntervalsIndex]
Intervals[1, TemporalIndex] $\leftarrow$ RightLimit
IntervalsIndex $\leftarrow$ TemporalIndex
```

the pixels with the same region ID must be in the same region set R. Before the region growing step, another two sets must be defined. The first set is P which is the set of pixels to label and is initially empty:

$$P = \{\}$$

The second set is Q which is the set of unlabeled pixels, this set must be filled with the remaining pixels (pixels that are not in R or in P).

$$Q = \left\{ x | x \notin \bigcup_{i=1}^{m} R_i \cup P \right\}$$

where m is the magnitude of R:

$$m = |R|$$

Regions sets R are used to construct the weighted instance-based learner (Section 5.4). In the region growing process, P and Q are updated on each iteration. P is filled with all the unallocated pixels x which in his 8-neighborhood has a labeled pixel:

$$P = \left\{ x | x \in Q \wedge N(x) \cap \bigcup_{i=1}^{m} R_i \neq \oslash \right\}$$

where $N(x)$ represents the 8-neighborhood of x.

The set of unallocated pixels Q must be updated by deleting all the elements previously added to P. In the classification step, the weighted instance-based algorithm is used to classify the pixels in P using the regions ID's as labels. Each labeled pixel is added to its corresponding region (a region defined by the same region ID). After all the pixels in P have been labeled, the weighted instance-based learner is reconstructed to consider the newly labeled instances. The region growing procedure ends when Q is empty (i.e., when all the pixels have a label). The region growing algorithm is shown in Fig. 5.3.

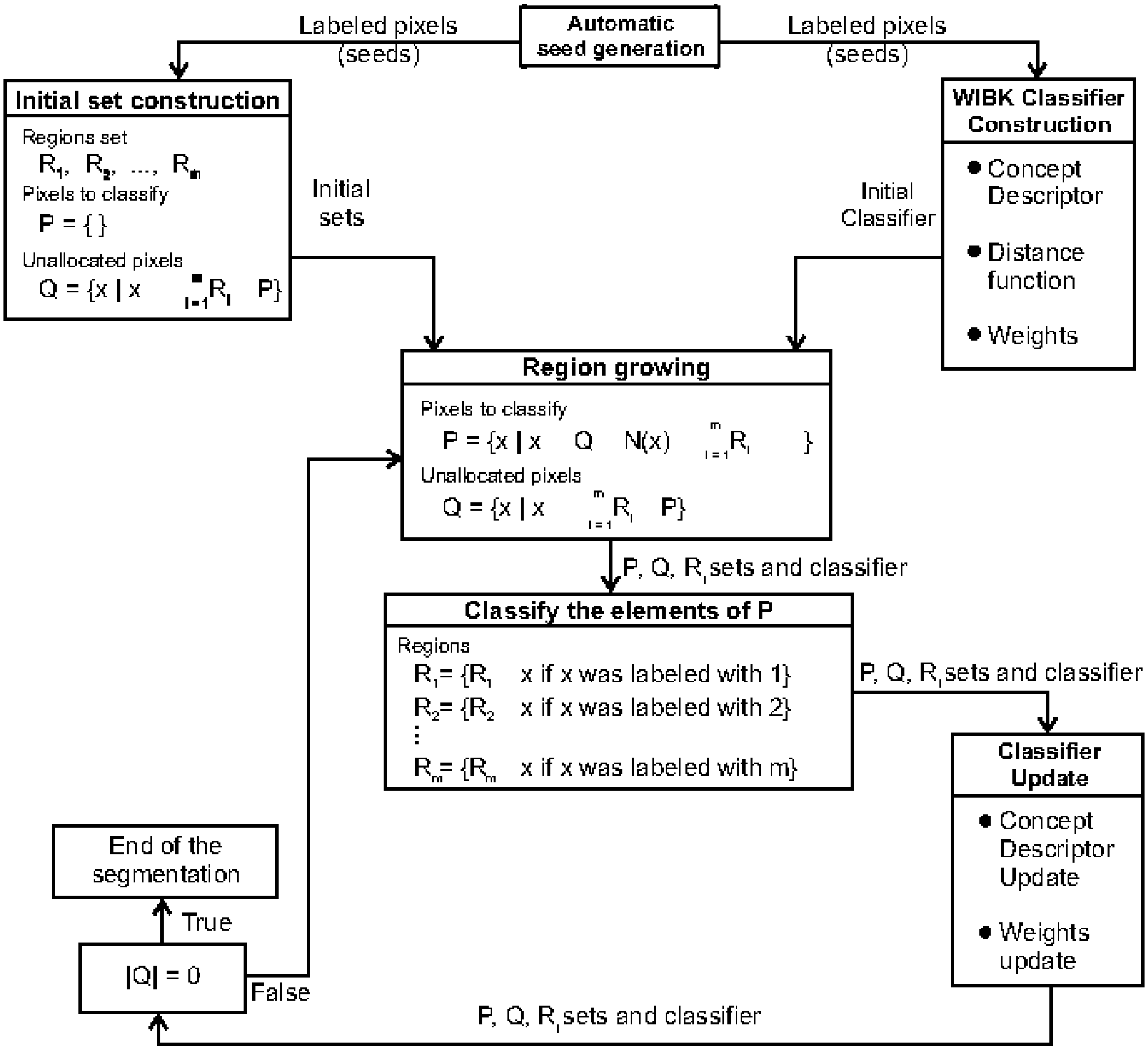


Figure 5.3: Region growing algorithm.

5.4 Feature Weighting Based on Representative Intervals

This section explains the proposed weighted instance-based learning algorithm based on representative intervals. Section 5.4.1 presents the initial definitions and Section 5.4.2 describes the weighted instance-based learning algorithm.

5.4.1 Initial definitions

- Ω is the instance space formed by n instances and m features.
- $x_i \in \Omega$ represents the i-th instance, $1 \leq i \leq n$.
- $x_{i,j}$ represents the value of the j-th feature of the i-th instance, $1 \leq j \leq m$.
- C^α_β is a multiset (a set of sets) that contains all the values of feature β for all the instances $x_i \in \Omega$ with $class(x_i) = \alpha$.

  $$C^\alpha_\beta = \{x_{i,j} | class(x_i) = \alpha \wedge j = \beta\}$$

- D^α_β is the set that contains all the values contained in C^α_β, but without repeated values. This set is ordered under the relation $<$. For example, if $C^\alpha_\beta = \{3, 5, 7, 4, 9, 3, 9, 5, 3, 5, 4, 6\}$ then $D^\alpha_\beta = \{3, 4, 5, 6, 7, 9\}$.
- $f(a)$ is the frequency function, it returns the number of times that a value $a \in D^\alpha_\beta$ appears in C^α_β, and is defined as follows

  $$f(a) = \sum_{\forall b_l \in C^\alpha_\beta} g(a, b_l)$$

  where $1 \leq l \leq |C^\alpha_\beta|$ and $g(a, b_l)$ is defined as

  $$g(a, b_l) = \begin{cases} 1 & \text{if } a = b_l \\ 0 & \text{otherwise} \end{cases}$$

  For example, with the previous sets C^α_β and D^α_β, $f(5) = 3$. This function can be viewed as an histogram of the image.

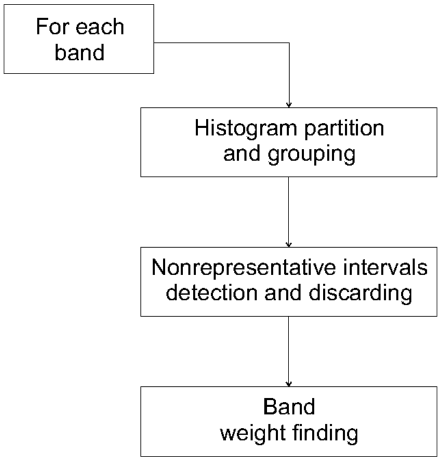

For each band
Histogram partition and grouping
Nonrepresentative intervals detection and discarding
Band weight finding

Figure 5.4: Weighting algorithm overview

5.4.2 Representative intervals and weights

An overview of the weighting algorithm is shown in Fig. 5.4. In the first step, the D_{β}^{α} set must be partitioned into mutually exclusive subsets $D_{\beta,\gamma}^{\alpha}$, where γ is the index of the partition. The partition must be done in such a way that all the consecutive values are grouped in exactly one partition, for example, in $D_{\beta}^{\alpha} = \{1, 2, 3, 5, 6, 7\}$ there are two sets of consecutive values which are $1, 2, 3$ and $4, 5, 6$, then resultant partitions are $D_{\beta,1}^{\alpha} = \{1, 2, 3\}$ and $D_{\beta,2}^{\alpha} = \{5, 6, 7\}$.

The magnitude of a partition $D_{\beta,\gamma}^{\alpha}$ is:

$$Magnitude(D_{\beta,\gamma}^{\alpha}) = \sum_{\forall t \in D_{\beta,\gamma}^{\alpha}} f(t)$$

The amplitude of a partition $D_{\beta,\gamma}^{\alpha}$ is:

$$Amplitude(D_{\beta,\gamma}^{\alpha}) = |D_{\beta,\gamma}^{\alpha}|$$

All the partitions $D_{\beta,\gamma}^{\alpha}$ are grouped according to their amplitude.

The second step proceeds as follows. Let $E_{\beta,\eta}^{\alpha}$ be the set formed by all the partitions $D_{\beta,\gamma}^{\alpha}$ with the same amplitude η, then, the maximum magnitude ψ of the set $E_{\beta,\eta}^{\alpha}$ is:

$$\psi = argmax(Magnitude(D_{\beta,\gamma}^{\alpha}))$$

Table 5.2: The four levels of confidence defined to discriminate noise

Level of confidence	Interval	Left value	Right value
High	$[H_i, H_f]$	$H_i = (H_f\%2) + 1$	$H_f = \psi$
Medium	$[M_i, M_f]$	$M_i = (M_f\%2) + 1$	$H_i - 1$
Low	$[L_i, L_f]$	$L_i = (L_f\%2) + 1$	$M_i - 1$
Null	$[0, N_f]$	0	$L_i - 1$

where $D^{\alpha}_{\beta,\gamma} \in E^{\alpha}_{\beta,\eta}$.

In order to discard the non representative intervals, considered as noise (outliers), it is necessary to define levels of confidence. This characteristic allows the algorithm to be noise-tolerant.

If ψ is the maximum amplitude of $E^{\alpha}_{\beta,\eta}$ then the levels of confidence are shown in Table 5.2, where $\%$ represents the integer division.

The sets $D^{\alpha}_{\beta,\gamma} \in E^{\alpha}_{\beta,\eta}$ whose magnitude falls in the null level of confidence are discarded because they are considered noise (outliers). The remaining sets $D^{\alpha}_{\beta,\gamma}$ are the representative intervals of feature β for class α.

Finally, in the third step, the percentage of values inside a representative interval of a feature β that is not overlapped with any other value inside all the representative intervals of the same feature β for all the remaining classes is the weight of the interval. The weight must be in the range $[0, 1]$. For example, if an interval of 30 values has 10 overlapped values, its weight is $(30 - 10)/30$. Fig 5.5 shows the four types of overlap between two intervals: totally overlapped(*A*) where *non-overlapped area* $= 0$, partially left-overlapped (*B*) where *non-overlapped area* $= (D_2 - I_2) - (D_1 - I_2)$, partially right-overlapped (*C*) where *non-overlapped area* $= (D_2 - I_2) - (D_2 - I_1)$ and partially center-overlapped (*D*) where *non-overlapped area* $= (D_2 - I_2) - (D_1 - I_1)$. In general, a given interval is overlapped by combinations of these base overlaps.

The obtained weights are used in the distance function of WIB-k:

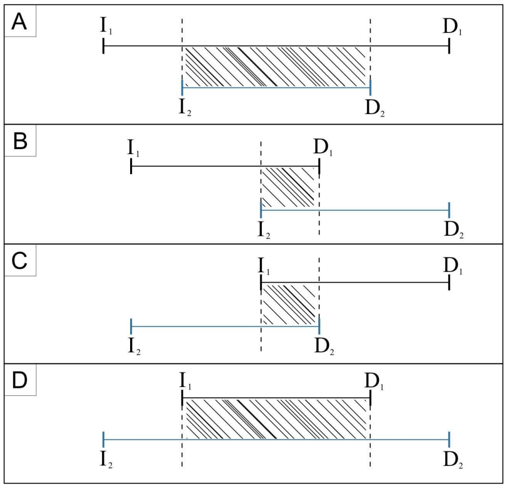


Figure 5.5: The four types of overlap between two intervals

$$Distance(x, y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2 w(y_i)}$$

where x is the example to label and y is the labeled example stored in the concept description of WIB-K (the concept description of an instance-based learning algorithm is described in section 3.2.2). If y_i falls within a representative interval, its weight will be the weight of the interval. If $x_{i,j}$ does not fall in any representative interval, its weight will be zero. The weights must be normalized before they can be used. The overlap finding algorithm is shown in Algorithm 6. This algorithm find the overlap between Interval1 ([InitInt1, EndInt1]) and Interval2 ([InitInt2, EndInt2]).

5.5 Region merging based on ownership tables

In this section, the region merging method based on ownership tables is explained. Section 5.5.1 introduces the concept of heterogeneous regions with semantic and Section 5.5.2 presents ownership tables.

Algorithm 6 Overlap finding algorithm

```
for i $=$ 0 TO NumberOfClasses do
    for j $=$ 0 TO NumberOfClasses do
        if i $\neq$ j then
            for k $=$ 0 TO NumberOfAttributes do
                if (InitInt2 $\leq$ InitInt1) AND (EndInt2 $\geq$ EndInt1) then
                    Weight $\leftarrow$ MinimumWeight
                else
                    if (InitInt2 $\leq$ InitInt1) AND (EndInt2 $\geq$ InitInt1) then
                        Update(StoredRightOverlap)
                    else
                        if (EndInt2 $\geq$ EndInt1) AND (InitInt2 $\leq$ EndInt1) then
                            Update(StoredLeftOverlap)
                        else
                            if (InitInt2 $>$ InitInt1) AND (EndInt2 $<$ EndInt1) then
                                Update(StoredInitCenterOverlap)
                                Update(StoredEndCenterOverlap)
                            end if
                        end if
                    end if
                end if
            end for
        end if
    end for
end for
```

5.5.1 Heterogeneous regions with semantic information

In many real-world remote sensing applications, a land cover that might have a meaning (semantic) to the end user can be composed by a number of homogeneous regions that, among them, are heterogeneous. For example, a vegetation cover can be formed by grass and trees; both covers are homogeneous, but the grass cover it does not look like the tree cover. The formal definition proceeds as follows. Let X and Y be two homogeneous regions:

$$X = \{x | \forall x \in X | F(x) = True\}$$

$$Y = \{y | \forall y \in Y | G(y) = True\}$$

where F(x) and G(y) are homogeneity predicates that describe respectively the pixels of X and Y. These regions are heterogeneous between them when:

$$\neg \exists x \in X | G(x) = true$$

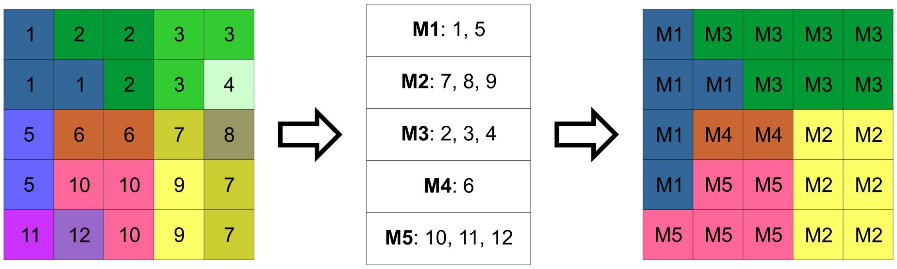
and

$$\neg \exists y \in Y | F(y) = true$$

The combination of two or more homogeneous regions results in an heterogeneous region, and when this region has a meaning for the end user it is called a heterogeneous region with semantic. For example, one user might want a city as a single cover whereas other user may want to separate roads from the city to make two different covers. Ownership tables are used to merge the distinct homogeneous regions in heterogeneous regions with semantics according to the user needs.

5.5.2 Ownership tables

To determine which regions conform a heterogeneous region with semantic, it is necessary to have high-level knowledge about the image provided by a domain expert or



1	2	2	3	3
1	1	2	3	4
5	6	6	7	8
5	10	10	9	7
11	12	10	9	7

M1: 1, 5

M2: 7, 8, 9

M3: 2, 3, 4

M4: 6

M5: 10, 11, 12

M1	M3	M3	M3	M3
M1	M1	M3	M3	M3
M1	M4	M4	M2	M2
M1	M5	M5	M2	M2
M5	M5	M5	M2	M2

Figure 5.6: The region merging via ownership tables

previous maps. The scheme for merging homogeneous regions proceeds as follows. Using the high-level knowledge about the image, a table is created to indicate which regions ID's correspond to the same heterogeneous region according to the semantic. This can be achieved through a semi-interactive process. In the simplest way to generate an ownership table, the user reviews the image, and based on personal judgment, chooses which single regions conform a composed region, this could be done using a mouse-based point and click mechanism.

Ownership tables must comply with the following requirements:

1. An ownership table must be defined for each region.

2. Two distinct regions can not have the same ownership table (formed by the same regions). A given ownership table completely defines a region.

3. If two distinct ownership tables share one or more regions, these regions must be re-labeled to eliminate ambiguity.

Region merging via ownership tables is illustrated in Fig. 5.6. The left figure shows the result of the segmentation. In the center image are the five ownership tables generated by the domain expert, which groups the regions shown in the left figure and gives semantic to them. The right figure shows the result of the region merging process.

Ownership tables also allow the user to define abstraction levels, regions with specific meaning for the user can be merged to form a higher level concept. For example, the initial segmentation can throw two crop covers, a corn crop and a chili crop, these

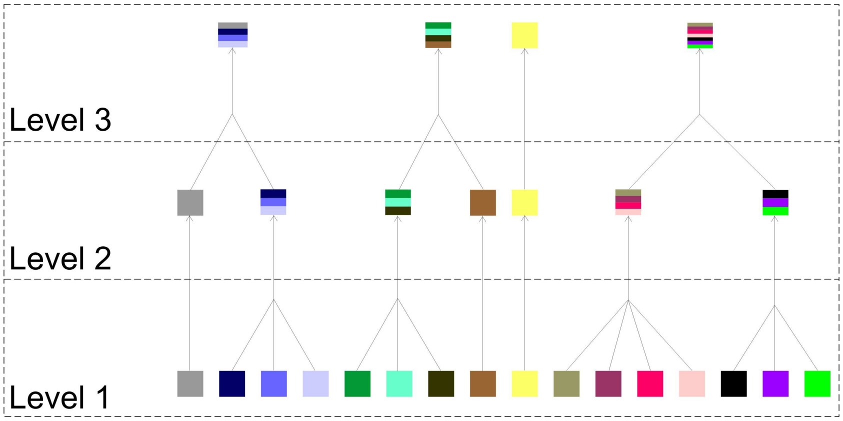


Figure 5.7: Hierarchical region merging

covers can be merged in a more general (higher lever) cover called crop fields, and in the next level, the crop fields cover can be merged with the tree cover to form a vegetation cover, which is more general than trees and crop fields.

Fig. 5.7 on Level 1 (level 1 of abstraction) shows several homogeneous regions resulting of the segmentation process. Level 2 groups some of the homogeneous regions into heterogeneous regions, with some meaning to the user. Level 3 shows the previous regions into a higher level of abstraction (more general covers), also with a meaning to the user.

Chapter 6

Experimental results

In this chapter, experimental results of the proposed learning and segmentation algorithms are presented. Section 6.1 shows the experiments with the WIBK algorithm, Section 6.2 shows the experiments with the SRG-WIBK algorithm and Section 6.3 presents a brief discussion of the results.

6.1 Experiments and results with WIBK

In this section, the performance of the WIBK algorithm is showed. Section 6.1.1 is an overview, in Section 6.1.2 the data sets used for the experiments are described, Section 6.1.3 briefly explains the ANOVA test, Section 6.1.4 presents the results of the comparison against instance-based and weighted instance-based algorithms and Section 6.1.5 shows the results of the comparison against well-known classifiers.

6.1.1 Overview

This section describes the experiments carried out to show the performance of the proposed WIBK algorithm over real world databases. Comparisons were made against several classifiers (instance-based and non instance-based).

The experiments were divided in two parts. The first part shows a comparison against weighted and non weighted instance-based algorithms. The second part shows a comparison against several well-know machine learning approaches. The comparisons in both cases show that WIBK has very competitive performance.

6.1.2 Data sets

For this experiment, the UCI machine learning repository [28] was selected as source of real world datasets. Databases with integer-valued features were selected, without concerning about the type of the class (numerical or nominal). Table 6.1 shows a brief description of the datasets with their name, number of instances, features, and classes.

All the data sets were randomly partitioned in ten disjoint sets for 10-fold cross validation, even the LC database where 27 examples where used for training and 5 for testing. The same sets for training and testing were used for all the algorithms (instance and non instance-based algorithms). Instances with missing values were removed because information-absence treatment is outside the scope of this research, although we could use a publicly available algorithm. The compared algorithms were taken from the Weka class library [47] and used with their default parameters, except for the *K* value used on some instance-based approaches that was always the same value used for WIBK.

6.1.3 The ANOVA test

The analysis of variance (ANOVA) is a statistical test for heterogeneity of means by analysis of group of variances [46]. To apply the test, it assumes random sampling of a variable Y with equal variance, independent errors and a normal distribution. Let n be the number sets of identical observations within each of K treatment groups, and y_{ij} be the jth. observation within factor level i. Also assume that the ANOVA is "balanced" by restricting n to be the same for each factor level. First, the sums of squared terms are defined:

Table 6.1: Data sets description

Name	Instances	Features	Classes
Balance Scale (BS)	625	4	3
Breast Cancer (BC)	699	11	2
CMC	1473	10	3
Dermatology (D)	366	34	6
Haberman (H)	306	4	2
Hayes Roth (HR)	162	6	3
Lung Cancer (LC)	32	57	3
TAE	151	6	3

$$SST = \sum_{i=1}^{k}\sum_{j=1}^{n} y_{i,j}^2 - \frac{(\sum_{i=1}^{k}\sum_{j=1}^{n} y_{i,j})^2}{Kn}$$

$$SSA = \frac{1}{n}\sum_{i=1}^{k}(\sum_{j=1}^{n} y_{i,j})^2 - \frac{1}{Kn}(\sum_{i=1}^{k}\sum_{j=1}^{n} y_{i,j})^2$$

$$SSE = \sum_{i=1}^{k}\sum_{j=1}^{n}(y_{i,j} - \overline{y}_i)^2$$

which are the total, treatment, and error sums of squares. Here, $\overline{y}_i$ is the mean of observations within factor level i. Compute the entries in Table 6.2 to obtain the F-ratio of the mean squared values:

$$Fratio = \frac{MSA}{MSE}$$

If the F-ratio is smaller that the value present in a F-values table with $(K-1, n-1)$ degrees of freedom, the difference among the compared quantities is not statistically significant, otherwise, the difference among the compared quantities is statistically significant.

Table 6.2: ANOVA entries to obtain the F-ratio of the mean squared values

Category	∘ freedom	SS	mean squared	F-ratio
model	$K-1$	SSA	$MSA = \frac{SSA}{K-1}$	$\frac{MSA}{MSE}$
error	$K(n-1)$	SSE	$MSE = \frac{SSE}{K(n-1)}$	
total	$Kn-1$	SST	$MST = \frac{SST}{Kn-1}$	

6.1.4 Comparison against instance-based and weighted instance-based algorithms

This subsection shows the results of the comparison of WIBK against other weighted and non weighted instance-based learning algorithms. IB1 and IB-*K* are the implementations of the original instance-based learning algorithms proposed by D. Aha et al. [3]. (1/d)IB*K* and (1-d)IB*K* are the IB-*K* algorithms weighted by the distance of the nearest neighbors (closest neighbors have more weight). (1/d) means that the weight is obtained from the inverse of the distance ($1/distance$) whereas (1-d) means that the weight is obtained from the complement of the distance ($1 - distance$). LWL is the implementation of the locally weighted learning algorithm proposed by Atkenson et al. [4]. This algorithm assigns a weight to each training observation, this weight depends upon the location of the training point in the input variable space relative to that of the point to be predicted, training observations closer to the prediction point generally receive higher weights. Finally, K-Star is the implementation of the instance-based learner K* proposed by J. C. Cleary and L. E. Trigg [8], this algorithm employs entropy as a distance measure.

Table 6.3 shows the comparison of classification rate accuracy (in %) between WIBK and other weighted and non weighted instance-based learners. WIBK achieves the higher accuracy in all the data sets except for Dermatology and Hayes Roth. Even for the Dermatology and Hayes Roth data sets, WIBK is better than other classifiers. Thus, in 6 out of 8 data sets, WIBK achieved the best performance. Table 6.3 also shows

Table 6.3: Accuracy comparison between WIBK and other weighted and not weighted instance-based learners

DB	K	WIBK	IB1	IB-*K*	(1/d)IB-*K*	(1-d)IB-*K*	LWL	K-Star
BS	24	**90.396**	79.027	89.436	89.436	89.436	53.932	88.474
BC	5	**97.216**	96.04	97.079	**97.216**	**97.216**	92.09	95.458
CMC	16	**55.126**	43.312	48.404	48.264	48.4	48.47	49.553
D	1	90.238	**95.269**	**95.269**	**95.269**	**95.269**	82.642	94.126
H	27	**75.483**	65.709	74.494	73.537	74.494	72.505	71.204
HR	2	75.604	76.538	62.087	67.417	67.417	**79.395**	61.263
LC	1	**70**	48.333	48.333	48.333	48.333	56.666	56.666
TAE	1	**72.958**	63.583	62.291	62.291	62.291	52.916	64.291

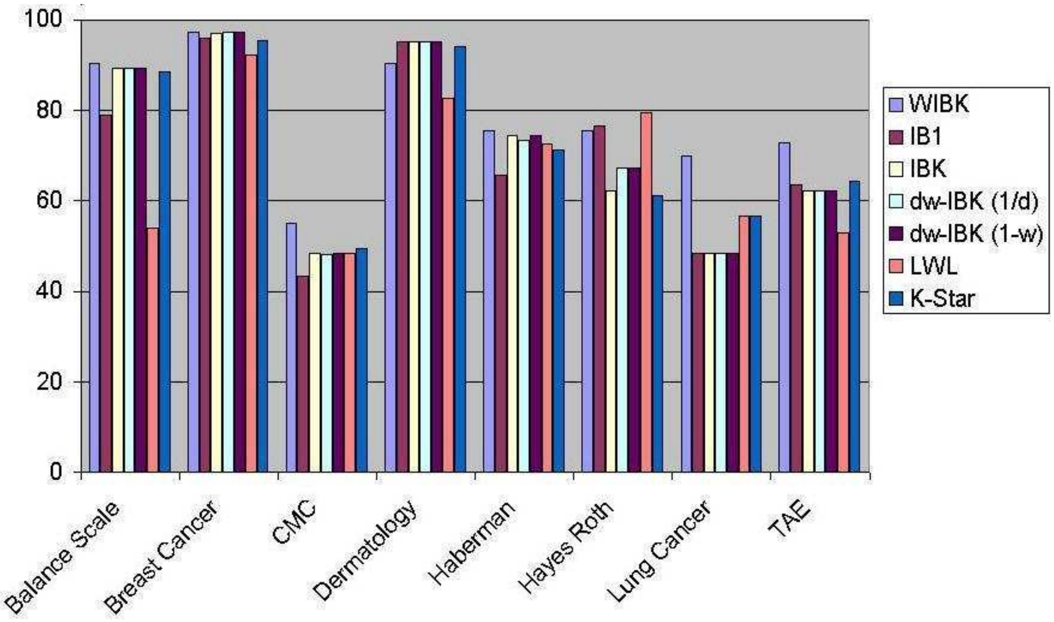


Figure 6.1: Bar Graph of the results of Table 6.3

Table 6.4: 95% ANOVA test of Table 6.3

DB	IB1	IBK	dw-IBK(1/d)	dw-IBK(1-w)	LWL	K-Star
BS	+	-	-	-	+	-
BC	-	-	-	-	+	-
CMC	+	+	+	+	+	+
D	-	-	-	-	-	-
H	-	-	-	-	-	-
HR	-	-	-	-	-	-
LC	-	-	-	-	-	-
TAE	-	-	-	-	+	-

that the proposed algorithm, WIBK, is consistently better than the original algorithms IB1 and IB-*K*.

Fig. 6.1 shows the classification rate with a bar graph. From the bar graph, it is clear that the proposed algorithm is superior in most of the tests in terms of classification rate.

Table 6.4 shows the results of a 95% ANOVA test with the results showed in Table 6.3. The symbol + means that the difference between the results is statistically significant while the symbol - means that the difference between the results is not statistically significant. In the experiments $K = 2$ and $n = 10$. From Table 6.4 we can see that only in the CMC database all the results are statistically significant. We can see too that in the BS, BC, CMC and TAE databases the results are statistically significant against the LWL classifier. In the D and HR databases, where other algorithms have the better performance, the difference is not statistically significant. In this table it is possible to see that statistically the results are very even.

6.1.5 Comparison against well-known classifiers

This subsection shows the results of the comparison of WIBK against well-known and representative machine learning algorithms. NB is a Naïve Bayes classifier that uses

Table 6.5: Accuracy comparison between IBK and other well-known algorithms

DB Name	K	WIBK	NB	SMO	MP	J48	PART
Balance Scale	24	90.39	90.04	87.68	**90.72**	76.64	83.52
Breast Cancer	5	**97.21**	96.33	97.07	96.04	96.04	95.46
CMC	16	**55.12**	49.28	50.98	54.51	53.22	50.10
Dermatology	1	90.23	97.48	**97.76**	97.48	95.25	93.29
Haberman	27	**75.48**	74.83	73.52	72.87	71.89	72.54
Hayes Roth	2	75.60	74.24	53.78	74.24	**80.30**	75.75
Lung Cancer	1	70	**70.37**	48.14	51.85	48.14	48.14
TAE	1	**72.95**	52.98	54.30	54.30	59.60	58.27

estimator classes [21]. SMO is the implementation of the algorithm for training a support vector classifier proposed by J. Platt [32]. MP is the implementation of a neural network that uses back-propagation as the training algorithm. J48 is the implementation of the C4.5 decision tree proposed by R. Quinlan [33]. Finally, PART is a decision rule-based algorithm proposed by E. Frank and I. H. Witten [12].

Table 6.5 shows the accuracy (in %) achieved by WIBK and other representative machine learning algorithms. The WIBK algorithm achieves the highest accuracy in the Breast Cancer, CMC, Haberman, and TAE data sets. For the remaining data sets, WIBK performed better than other well-known machine learning algorithms. The other algorithms obtained the best result in at most one data set whereas WIBK did it in four.

Figure 6.2 shows a bar graph of the data presented in Table 6.5. From the bar graph we can see that the WIBK algorithm is highly competitive.

Table 6.6 shows the results of a 95% ANOVA test with the results showed in Table 6.5. The symbol + means that the difference between the results is statistically significant while the symbol - means that the difference between the results is not statistically significant. In the experiments $K = 2$ and $n = 10$. From Table 6.6 we can see that in the BS, BC, CMC, D, and HR databases the algorithm has at least one statistically

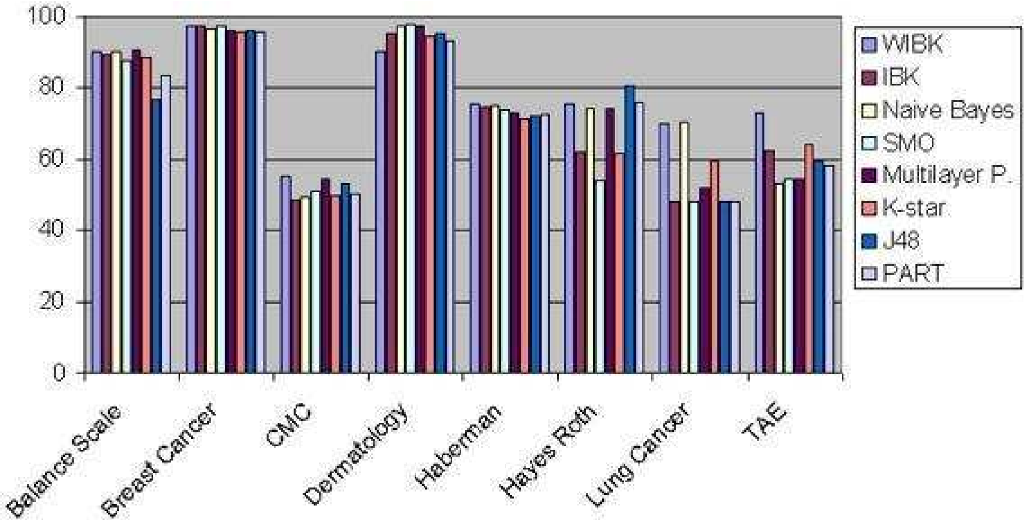


Figure 6.2: Bar Graph of the results of Table 6.5

Table 6.6: 95% ANOVA test of Table 6.5

DB	NB	SMO	MP	J48	PART
BS	-	-	-	+	+
BC	-	-	-	-	+
CMC	+	-	-	-	-
D	+	+	-	-	-
H	-	-	-	-	-
HR	-	+	-	-	-
LC	-	-	-	-	-
TAE	-	-	-	-	-

🟥 Class 1: Mangrove

⬜ Class 2: Urban Zone

⬛ Class 3: Roads

🟦 Class 4: Water

🟩 Class 5: Vegetation

🟧 Class 6: Uncovered Soil or Low Vegetation

Figure 6.3: Classes of interest selected for the comparison, along with the color code selected for each class

significant difference. In the BS, HR and LC databases, where other algorithms have the better performance, the difference is not statistically significant. Once again, in this table it is possible to see that the results are very even.

6.2 Algorithm experiments and results with SRG-WIBK

In this section, the performance of the SRG-WIBK algorithm is showed. In Section 6.2.1 the classes used for the experiments are showed, Section 6.2.2 briefly explains the method used to generate the multispectral synthetic images used in the experiments and Section 6.2.3 shows the comparison against the ERDAS IMAGINE remote sensing commercial tool with five multispectral synthetic images.

6.2.1 Selected classes

For the experiments carried out in this section, six classes of interest were selected. The classes of interest along with the color code selected are showed in Fig. 6.3.

6.2.2 Multispectral synthetic images

For this set of experiments, multispectral synthetic images were used to perform a quantitative evaluation because of the unavailability of thematic maps for reference. The method for generating these images was proposed and used in [35]. In this method the multispectral satellite images must be segmented. In this thesis we used the ISODATA implementation of the ERDAS IMAGINE software to segment the image. After that, the pixels of the segmented regions were substituted with labeled pixels of known classes (ground truth data). This ensures that the regions are truly constituted with pixels that belong to the class that the region represents.

The data used to generate the synthetic images was take from multispectral SPOT-5 satellite images of the Veracruz port. SPOT-5 images have 10m spatial resolution, and a cover of 60 x 60 km (3600 km^2). These images register four spectral bands: green $(0.5 - 0.59\mu m)$, red $(0.61 - 0.68\mu m)$, near infrared $(0.78 - 0.89\mu m)$ and medium infrared $(1.58 - 1.75\mu m)$. The SPOT-5 images that we used have a 2A processing level, which means radiometric and geometric corrections.

Five multispectral synthetic images were generated. Fig. 6.4 shows the four bands of the first multispectral synthetic image (MSI-1). The six regions of interest are present in this image. Mangrove, vegetation, and uncovered soil / low vegetation are the predominant classes. Fig 6.5 shows a false-color composition of MSI-1 that combines bands four, one, and three. False color composition was explained in Section 2.2.2.

6.2.3 Comparison against commercial remote sensing tools

In order to demonstrate the performance of the proposed SRG-WIBK algorithm, a comparison against the ERDAS IMAGINE software was carried out. ERDAS IMAGINE is a powerful and highly specialized analysis tool for satellite imagery developed by Leek Geosystems. Its current price (commercial edition) is USD $2200.

ERDAS IMAGINE has five segmentation/classification algorithms. Four algorithms perform supervised classification and one performs unsupervised classification. The supervised algorithms implemented are Minimum Distance, Mahalanobis Distance, Maximum Likelihood, and Parallelepiped. These algorithms are broadly used in many other software packages for remote-sensing image analysis such as Multispec, ENVI, and ERMapper. The unsupervised algorithm implemented in the ERDAS IMAGINE soft-

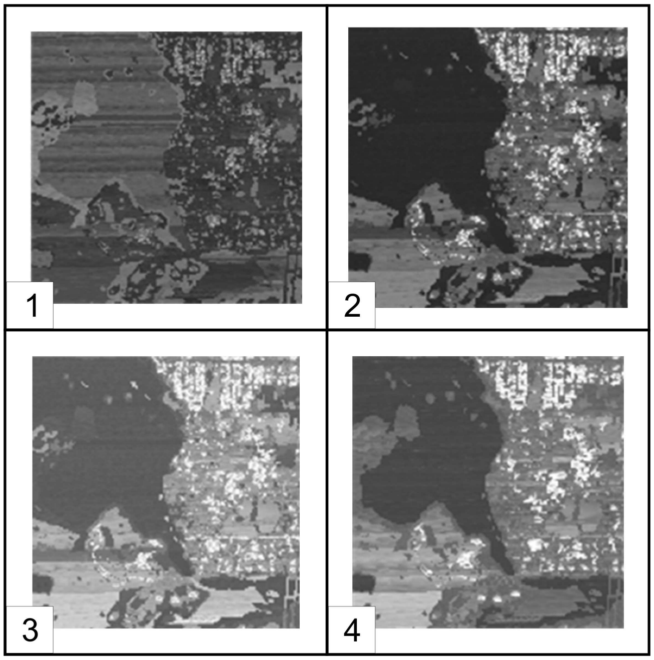


Figure 6.4: The four bands of the Multispectral Synthetic Image 1 (MSI-1)

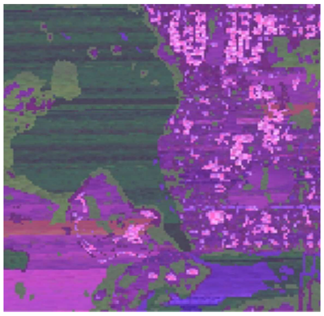


Figure 6.5: A false-color composition of the multispectral synthetic image 1 (MSI-1)

Table 6.7: Ownership tables of MSI-1

Mangrove: 1, 2, 3, 4, 6, 10, 13, 9, 24
Vegetation: 5, 23
Water: 31, 28, 26, 15, 17, 27, 29, 30
Urban Zones: 7, 11, 14, 18, 19
Roads: 12, 16, 21, 20
Uncovered Soil: 8, 25, 22

ware is ISODATA. More information about these algorithms can be found in [34].

It is not possible to apply statistical significance tests (such as the ANOVA test) to the experiments carried out in this section because statistical test need a number n of replicates of the experiment, in this case, the segmentation. Several segmentations of the same image with the algorithms compared in this section always produce the same results.

Experiments with MSI-1

This section describes the experiments carried out with MSI-1. In the experiments, the same image (MSI-1) was processed with all the algorithms. The thematic map resulting from each algorithm segmentation was compared against the thematic map of the image (reference segmentation) by means of intersections between regions.The segmented region was intersected with the reference region, all the pixels which fall inside the intersection are correctly segmented pixels and all the pixels which not fall inside the intersection are incorrectly segmented pixels.

SRG-WIBK has obtained 31 homogeneous regions on this image, Table 6.7 shows the ownership tables of MSI-1, each row is an ownership table that contains the concept formed and the region ID's that conforms it. Table 6.8 shows the acronyms of the ERDAS IMAGINE algorithms.

Table 6.9 shows the comparison among SRG-WIBK and the ERDAS IMAGINE al-

Table 6.8: Acronyms for the ERDAS IMAGINE algorithms

Name	Acronym
Mahalanobis Distance	MHD
Minimum Distance	MD
Maximum Likelihood	ML
Parallelepiped	P

Table 6.9: Comparison of SRG-WIBK against the ERDAS IMAGINE algorithms on MSI-1

Cover	SRG-WIBK	MHD	MD	ML	P	ISODATA
Mangrove	99.88	99.97	99.7	99.97	100	100
Urban Zone	99.74	100	99.43	100	99.91	98.99
Roads	93.12	100	91.51	100	100	77.01
Water	100	100	42.61	100	72.5	32.4
Vegetation	99.66	99.18	99.66	99.35	99.18	0
Uncovered Soil	100	100	100	100	98.69	100
Overall Accuracy	98.52	99.84	93.55	99.87	97.41	71.61

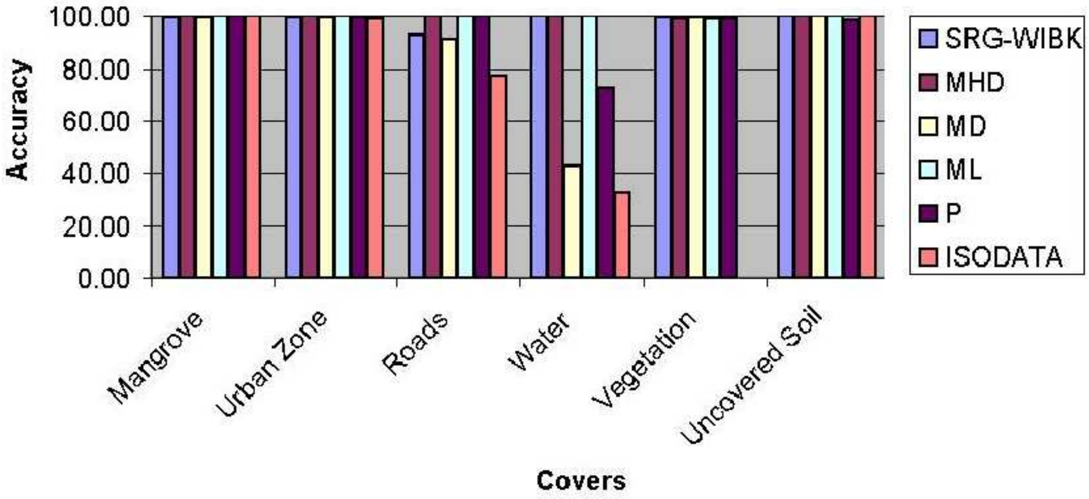


Figure 6.6: Graph bar for Table 6.9

gorithms in terms of correctly labeled pixels percentage (accuracy). Each column represents an algorithm and each row represents a segmented class; the last row presents the overall accuracy obtained by each algorithm. As we can see from the table, SRG-WIBK had better performance than the Minimum Distance, Parallelepiped, and ISODATA algorithms in a 4.97%, 1.11% and 26.91% respectively. Although the Mahalanobis Distance and the Maximum Likelihood algorithms performed better than the SRG-WIBK in 1.32% and 1.35% respectively, SRG-WIBK has the advantage that, contrary to these algorithms, it does not need training data.

Mangrove, Urban Zone, and uncovered soil were, in general, well classified by all the algorithms. The Vegetation cover was well classified by all the methods except for ISODATA, which was not able to find this cover. The water cover presented problems to the Minimum Distance, Parallelepiped, and ISODATA algorithms while SRG-WIBK achieved 100% of accuracy on this class. For the Road cover, SRG-WIBK obtained its lowest accuracy, which was 93.12%.

Fig. 6.6 shows the data of Table 6.9 in a graph bar. From this graph it can be seen that the SRG-WIBK algorithm is very even. In five of six covers (Mangrove, Urban Zone, Water, and Vegetation), the performance of SRG-WIBK is higher than 99 %.

Table 6.10 shows the confusion matrix obtained from the segmentation of MSI-1 with the SRG-WIBK algorithm; from this table, the omission error can be obtained. The omission error is widely used in remote sensing literature, this error refers to those pixels belonging to the class of interest that the classifier has failed to recognise. The omission error is obtained from the columns; for example, in the column of mangrove, the classifier has failed to recognise 7 pixels, that means 0.12% of the total in the column (5742 pixels). The mangrove cover has few omission errors with the vegetation cover in a 0.12%, the roads cover was mistaken with the uncovered soil, water and vegetation covers in a 1.75%, 4.44% and 0.67% respectively. The urban zone cover was little mistaken with the uncovered soil and water covers in a 0.17% and 0.08% respectively. Finally, the vegetation cover was mistaken with the uncovered soil and mangrove covers in a 0.26% and 0.07% respectively. In general, all the covers have

Table 6.10: Confusion matrix of the segmentation of MSI-1 with SRG-WIBK

	Row %	total	Unc. S.	Mang.	Roads	Water	Urban Z.	Vegetation
Unc. S.	97.79	4208	4115	0	78	0	4	11
Mang.	99.95	5738	0	5735	0	0	0	3
Roads	100	4144	0	0	4144	0	0	0
Water	89.96	1993	0	0	198	1793	2	0
Urban Z.	100	2264	0	0	0	0	2264	0
Vegetation	99.11	4153	0	7	30	0	0	4116
Total		22500	4115	5742	4450	1793	2270	4130
Column %			100	99.88	93.12	100	99.74	99.66
Kappa stat.	98.18							
Global %	98.52							

been well classified, with an accuracy higher than 93%.

Experiments with MSI-2

This section shows the experiments performed over the multispectral synthetic image 2 (MSI-2). This image only contains four covers which are Urban Zone, Roads, Water and Uncovered Soil. The vegetation and Mangrove covers are not present in this image. Urban Zone and Water are the predominant classes. The conditions of the experiments were almost the same than in the experiments carried out with MSI-1. The only difference is that the Mangrove and Vegetation training data were removed from the training data of the ERDAS IMAGINE algorithms to avoid confusion of these algorithms during classification and to avoid the noise that this data can introduce. SRG-WIBK has obtained 11 homogeneous regions on this image, Table 6.11 shows the ownership tables of MSI-2.

Table 6.12 shows the accuracy comparison among SRG-WIBK and the ERDAS

Table 6.11: Ownership tables of MSI-2

Water: 7, 9
Urban Zones: 10, 6, 5, 4, 3, 1
Roads: 11, 8
Uncovered Soil: 2

Table 6.12: Comparison of SRG-WIBK against the ERDAS IMAGINE algorithms on MSI-2

Cover	SRG-WIBK	MHD	MD	ML	P	ISODATA
Urban Zone	99.43	99.01	96.24	98.79	97.11	99.36
Roads	99.95	100	97.89	100	100	0
Water	81.88	100	65.36	100	83.71	100
Uncovered Soil	97.13	100	100	100	99.60	100
Overall Accuracy	96.73	99.52	93.03	99.41	96.30	80.11

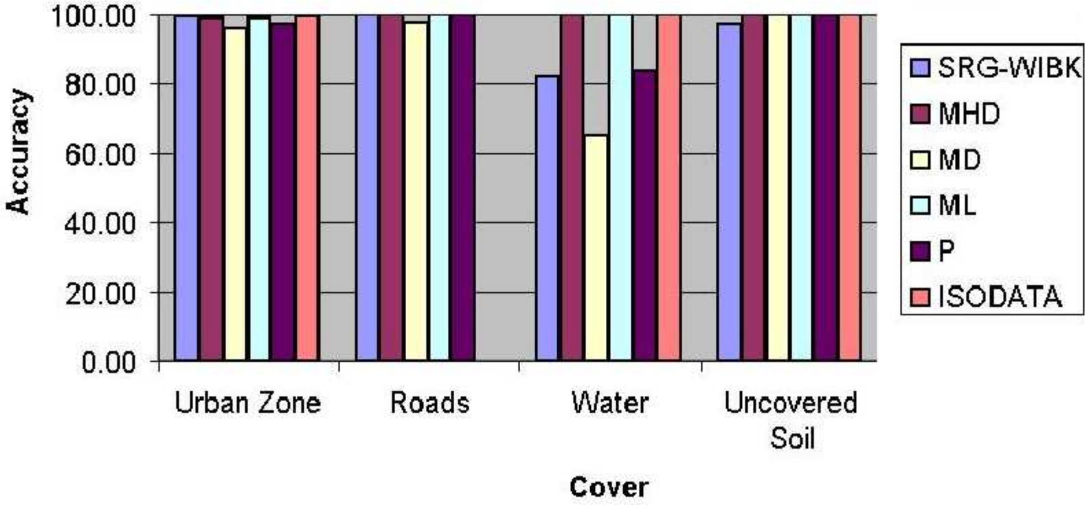


Figure 6.7: Graph bar of Table 6.12

Table 6.13: Confusion matrix of the segmentation of MSI-2 with SRG-WIBK

	Row %	total	Unc. S.	Roads	Water	Urban Z.
Unc. S.	99.31	3922	3895	2	0	25
Roads	86.5	5090	115	4403	534	38
Water	100	2508	0	0	2508	0
Urban Z.	99.81	10980	0	0	21	10959
Total		22500	4010	4405	3063	11022
Column %			97.13	99.95	81.88	99.43
Kappa stat.	95.13					
Global %	96.73					

IMAGINE algorithms for the multispectral synthetic image 2. The results were similar to the results obtained over MSI-1. SRG-WIBK again performed better than the Minimum Distance, Parallelepiped, and ISODATA algorithms in a 3.7%, 0.43% and 16.62% respectively. The Mahalanobis Distance and Maximum Likelihood algorithms performed better than SRG-WIBK in a 2.79% and 2.68% respectively.

Urban zone and uncovered soil covers were, in general, well classified by all the algorithms. The road cover was well classified by all the methods except by ISODATA, which again was not able to find that cover. The water cover presented many problems to the minimum distance algorithm while SRG-WIBK achieved 81.88% of accuracy on this class. For the water cover, SRG-WIBK obtained its lowest accuracy.

Fig. 6.7 shows the data of Table 6.12 in a graph bar. From this graph we can see that in three of four covers (urban zone, roads and uncovered soil), the performance of SRG-WIBK is higher than 97%, and in two of these three covers the performance is higher than 99%.

Table 6.13 shows the confusion matrix obtained from the segmentation of MSI-2 with SRG-WIBK algorithm. The uncovered soil cover was mistaken with the road cover in a 2.86%. The road cover was little mistaken with the uncovered soil cover in a

Table 6.14: Ownership tables of MSI-3

Mangrove: 2, 3, 4, 5, 11, 15
Vegetation: 20, 10, 9
Water: 26, 25, 24, 23, 22, 21, 12
Urban Zones: 19, 18, 17, 14
Roads: 16, 8, 7, 6
Uncovered Soil: 1, 13

0.4%. The water cover was mistaken with the road and urban zone covers in a 17.58% and 0.69%, and finally the urban zone cover was little mistaken with the uncovered soil and road covers in a 0.22% and 0.34% respectively. In general, all the covers except water were well classified, with an accuracy higher than 97%.

Experiments with MSI-3

This section shows the experiments performed over the multispectral synthetic image 3 (MSI-3). This image contains all the covers. Mangrove, Vegetation and Water are the predominant classes. The conditions of the experiments were the same than in the experiments carried out with MSI-1. SRG-WIBK has obtained 26 homogeneous regions on this image, Table 6.14 shows the ownership tables of MSI-3.

Table 6.15 shows the accuracy comparison among SRG-WIBK and the ERDAS IMAGINE algorithms. As we can see from the table, SRG-WIBK performed better than the minimum distance and ISODATA algorithms in a 2% and 36.58% respectively. The Mahalanobis distance, maximum likelihood and parallelepiped algorithms performed better than the SRG-WIBK in a 5.55%, 5.58% and 2.45% respectively.

Mangrove, urban zone, roads and vegetation covers were, in general, well classified by all the algorithms. The Vegetation cover was well classified by all the methods except for ISODATA, which was not able to find this cover. For the water cover, SRG-WIBK obtained its lowest accuracy, which was 64.94%.

Table 6.15: Comparison of SRG-WIBK against the ERDAS IMAGINE algorithms on MSI-3

Cover	SRG-WIBK	MHD	MD	ML	P	ISODATA
Mangrove	99.28	99.92	99.44	99.92	100	100
Urban Zone	100	100	100	100	100	100
Roads	98.81	100	93.71	100	100	0
Water	64.94	100	57.65	100	80.04	100
Vegetation	98.1	98.89	99.48	99.06	98.79	0
Uncovered Soil	100	100	100	100	97.91	100
Overall Accuracy	94.23	99.78	92.23	99.81	96.68	57.65

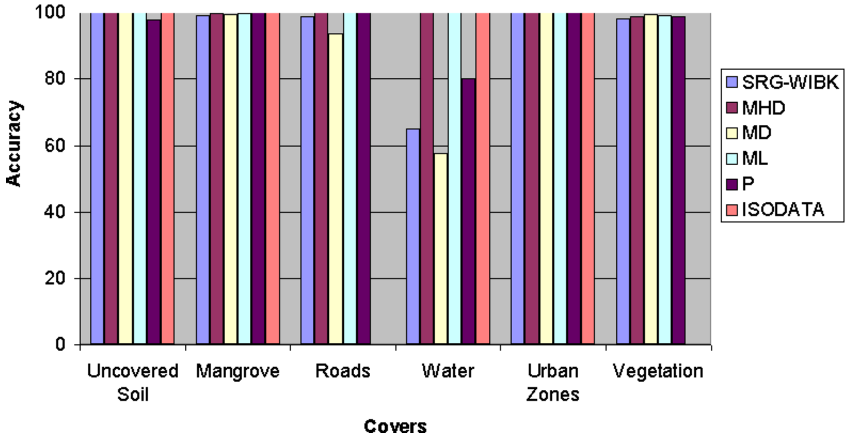


Figure 6.8: Graph bar for Table 6.15

Table 6.16: Confusion matrix of the segmentation of MSI-3 with SRG-WIBK

	Row %	total	Unc. S.	Mang.	Roads	Water	Urban Z.	Vegetation
Unc. S.	99.02	2856	2828	0	12	0	0	16
Mang.	98.84	4993	0	4935	0	0	0	58
Roads	82.82	6542	0	0	5418	1121	0	3
Water	97.92	2120	0	0	44	2076	0	0
Urban Z.	100	1975	0	0	0	0	1975	0
Vegetation	98.88	4014	0	36	9	0	0	3969
Total		22500	2828	4971	5483	3197	1975	4046
Column %			100	99.28	98.81	64.94	100	98.1
Kappa stat.	92.88							
Global %	94.23							

Fig. 6.8 shows the data of Table 6.15 in a graph bar. From this graph we can see that in five of six covers (mangrove, urban zone, uncovered soil, and vegetation), the performance of SRG-WIBK is higher than 98%, and in two of these five covers the performance is 100%.

Table 6.16 shows the confusion matrix obtained from the segmentation of MSI-3 with SRG-WIBK algorithm. From this table we can see that in the uncovered soil and urban zone covers the classifier did not make any mistake. The mangrove cover was little mistaken with vegetation in a 0.72%. The road cover was mistaken with the uncovered soil, water and vegetation covers in a 0.21%, 0.80% and 0.16% respectively. The water cover was frequently mistaken with the road cover in a 35.06%. Finally, the vegetation cover was mistaken with the uncovered soil, mangrove and road covers in a 0.39%, 1.43% and 0.07% respectively. In general, all the covers except water were well classified, with an accuracy higher than 98%.

Table 6.17: Ownership tables of MSI-4

Vegetation: 25, 24, 22, 21, 20, 18, 13, 10, 7, 6 ,2
Urban Zones: 17, 16, 15, 14, 9, 8, 1
Roads: 23, 19, 12, 5, 4
Uncovered Soil: 11, 3

Table 6.18: Comparison of SRG-WIBK against the ERDAS IMAGINE algorithms on MSI-4

Cover	SRG-WIBK	MHD	MD	ML	P	ISODATA
Urban Zone	100	100	99.63	100	99.94	99.44
Roads	99.47	100	97	100	100	98.91
Vegetation	99.6	96.6	99.71	97.9	99.09	99.64
Uncovered Soil	100	100	100	100	98.77	100
Overall Accuracy	99.74	98.84	99.15	99.28	99.33	99.54

Experiments with MSI-4

This section shows the experiments performed over the multispectral synthetic image 4 (MSI-4). This image only contains four covers which are urban zone, roads, vegetation and uncovered soil. The water and mangrove covers are not present in this image. Urban Zone and roads are the predominant covers. The conditions of the experiments were almost the same than in the experiments carried out with MSI-1. The only difference is that the Mangrove and Water training data were removed from the training data of the ERDAS IMAGINE algorithms to avoid confusion of these algorithms during classification and to avoid the noise that this data can introduce. SRG-WIBK has obtained 25 homogeneous regions on this image, Table 6.17 shows the ownership tables of MSI-4.

Table 6.18 shows the accuracy comparison among SRG-WIBK and the ERDAS IMAGINE algorithms for the multispectral synthetic image 4. In this image SRG-

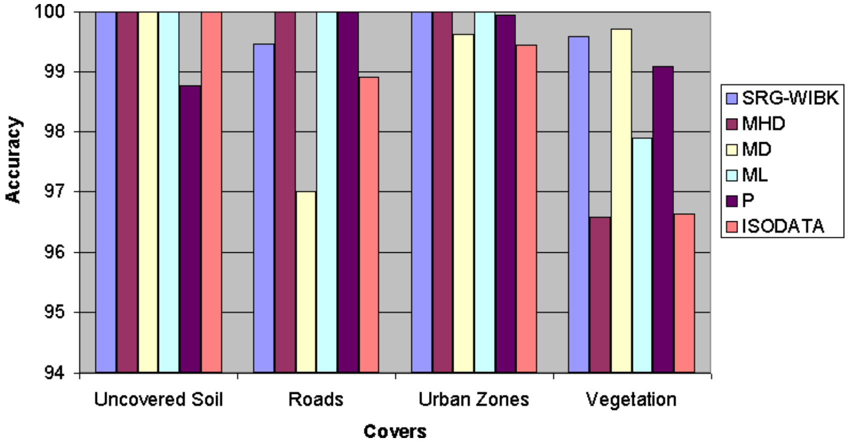


Figure 6.9: Graph bar of Table 6.18

WIBK has achieved his best performance, being better than all the other algorithms. SRG-WIBK performed better than the Mahalanobis distance, minimum distance, maximum likelyhood, parallelepiped, and ISODATA algorithms in a 0.9%, 0.59%, 0.46%, 0.41% and 0.2% respectively.

Fig. 6.9 shows the data of Table 6.18 in a graph bar. From this graph we can see that in all covers the performance of SRG-WIBK is higher than 99%, and in two of these three covers the performance is 100%.

Table 6.19 shows the confusion matrix obtained from the segmentation of MSI-4 with the SRG-WIBK algorithm. Uncovered soil and urban zone covers were perfectly classified by SRG-WIBK. The road cover was little mistaken with the uncovered soil, urban zone and vegetation covers in a 0.13%, 0.21% and 0.19% respectively. The vegetation cover was mistaken with the uncovered soil and road covers in a 0.28% and 0.11% respectively. In general, all the covers were well classified, with an accuracy higher than 99.4%.

Experiments with MSI-5

This section shows the experiments performed over the multispectral synthetic image 5 (MSI-5). This image contains all the covers, with mangrove and uncovered soil as

Table 6.19: Confusion matrix of the segmentation of MSI-4 with SRG-WIBK

	Row %	total	Unc. S.	Roads	Urban Z.	Vegetation
Unc. S.	99.54	6352	6323	7	0	22
Roads	99.83	5215	0	5206	0	9
Urban Z.	99.66	3253	0	11	3242	0
Vegetation	99.87	7680	0	10	0	7670
Total		22500	6323	5234	3242	7701
Column %			100	99.47	100	99.6
Kappa stat.	99.64					
Global %	99.74					

Table 6.20: Ownership tables of MSI-5

Mangrove: 2, 6
Vegetation: 5
Water: 1, 4, 8, 9, 10, 11, 12
Urban Zones: 3, 7
Roads: 13, 15
Uncovered Soil: 14

predominant. The conditions of the experiments were the same than in the experiments carried out with MSI-1 and MSI-3. SRG-WIBK has obtained 15 homogeneous regions on this image, Table 6.20 shows the ownership tables of MSI-5.

Table 6.21 shows the accuracy comparison among SRG-WIBK and the ERDAS IMAGINE algorithms. As we can see, SRG-WIBK is most accurate than the Minimum Distance and ISODATA algorithms in a 3.84% and 15.52% respectively. The Mahalanobis Distance, Maximum Likelihood and parallelepiped algorithms performed better than the SRG-WIBK in a 3.86%, 3.86% and 0.02% respectively.

All the covers except roads were, in general, well classified by all the algorithms.

Table 6.21: Comparison of SRG-WIBK against the ERDAS IMAGINE algorithms on MSI-5

Cover	SRG-WIBK	MHD	MD	ML	P	ISODATA
Mangrove	97.05	99.9	99.33	99.9	100	100
Urban Zone	99.07	100	98.89	100	99.81	98.45
Roads	78.59	100	94.05	100	100	58.31
Water	100	100	62.94	100	79.38	41.37
Vegetation	100	100	100	100	100	0
Uncovered Soil	100	100	100	100	99.26	100
Overall Accuracy	96.1	99.96	92.26	99.96	96.12	80.58

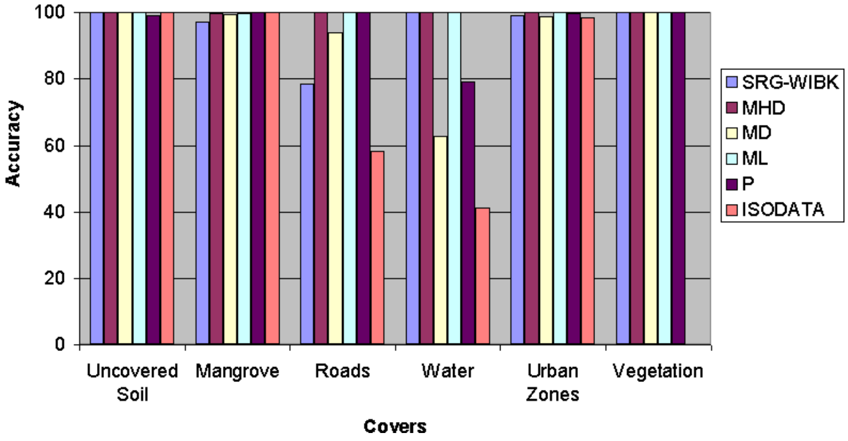


Figure 6.10: Graph bar for Table 6.21

Table 6.22: Confusion matrix of the segmentation of MSI-5 with SRG-WIBK

	Row %	total	Unc. S.	Mang.	Roads	Water	Urban Z.	Vegetation
Unc. S.	100	5246	5246	0	0	0	0	0
Mang.	99.9	7684	0	7676	8	0	0	0
Roads	99.61	2322	0	0	2313	0	9	0
Water	87.03	4635	0	0	595	4034	6	0
Urban Z.	100	1601	0	0	0	0	1601	0
Vegetation	74.31	1012	0	233	27	0	0	752
Total		22500	5246	7909	2943	4034	1616	752
Column %			100	97.05	78.59	100	99.07	100
Kappa stat.	94.92							
Global %	96.1							

The Vegetation cover was well classified by all the methods except for ISODATA, which was not able to find this cover. For the road cover, SRG-WIBK obtained its lowest accuracy, which was 78.59%.

Fig. 6.10 shows the data of Table 6.21 in a graph bar. From this graph we can see that in five of six covers the performance of SRG-WIBK is higher than 90%, and in four of these five covers the accuracy was higher than 96%.

Table 6.22 shows the confusion matrix obtained from the segmentation of MSI-5 with SRG-WIBK algorithm. From this table we can see that in the uncovered soil, water and vegetation covers the classifier did not make any mistake. 2.95% of mangrove pixels were mistaken with vegetation. The road cover was mistaken with the mangrove, water and vegetation covers in a 0.27%, 20.21% and 0.91% respectively. The road cover was the worse classified. The urban zone cover was little mistaken with road and water in a 0.55% and 0.37%. In general, all the covers except roads were well classified, with an accuracy higher than 97%.

6.3 Discussion

In the experiments performed with the instance-based algorithm WIBK, an improvement was shown over the base algorithms IB1 and IBK. The results also demonstrate that the weighting scheme used in WIBK is better than other weighting schemes based on distance. In the comparison of WIBK against well-known machine learning algorithms it was empirically shown that the proposed algorithm is highly competitive. No algorithm performed consistently better than the others on all the databases. For a particular application, many approaches are normally tested before deciding which one performs better.

The ERDAS IMAGINE segmentation algorithms are a hard proof for all remote-sensing applications. This software, with 17 years in the market, have achieved high performance and maturity on his algorithms, however, the proposed SRG-WIBK algorithm showed high competitiveness, performing better than three of the five algorithms. The algorithms that were superior than SRG-WIBK have the disadvantage that, unlike the proposed algorithm, they need training data because they are supervised approaches. The ISODATA algorithm, that does not need training data because it is an unsupervised approach, has been beaten in all the experiments. It is well known that the ERDAS IMAGING algorithms were highly improved by the Leica Geosystems company because the same experiments with all the multispectral synthetic images were carried out with local implementations of the Mahalanobis distance, mean distance, maximum likelihood, and parallelepiped algorithms. These implementations were coded in Matlab by Juan F. Robles for his master's thesis [35], and are based on the theory explained in [34]. The results of these experiments with MSI-1 and MSI-2 are shown in Table 6.23 and Table 6.24. In all the cases, SRG-WIBK obtained the highest overall accuracy by a large margin. The results of the experiments with MSI-3, MSI-4 and MSI-5 were not shown because the local implementations have obtained percentages lower than 50%.

Table 6.23: Comparison of SRG-WIBK against local implementations of the MHD, MD, ML and P algorithms with MSI-1

Cover	SRG-WIBK	MHD	MD	ML	P
Mangrove	99.88	83.75	95.45	74.03	95.47
Urban Zone	99.74	99.91	99.16	99.07	99.16
Roads	93.12	13.48	45.57	10.18	45.37
Water	100	74.68	50.64	79.53	50.64
Vegetation	99.66	8.55	40.73	14.96	40.70
Uncovered Soil	100	58.01	92.78	71.42	92.78
Overall Accuracy	98.52	52.25	71.86	53.05	71.82

Table 6.24: Comparison of SRG-WIBK against local implementations of the MHD, MD, ML and P algorithms with MSI-2

Cover	SRG-WIBK	MHD	MD	ML	P
Urban Zone	99.43	99.96	95.79	98.92	95.79
Roads	99.95	45.72	50.99	29.99	52.74
Water	81.88	72.64	92.36	77.41	91.61
Uncovered Soil	97.13	12.34	35.36	40.77	35.36
Overall Accuracy	96.73	70.01	75.78	72.13	76.02

Conclusions and future research

In this section conclusions, summary of contributions and future research directions are given.

Conclusions

Segmentation through seeded region growing is widely used in images because it is fast, robust and free of tuning parameters. However, the seeded region growing algorithm requires an automatic seed generator. This thesis introduces a new automatic seeded region growing algorithm called SRG-WIBK that performs the segmentation of color (RGB) and multispectral images. The seeds are automatically generated via a threshold analysis; the histogram of each band is analyzed to obtain intervals of representative pixel values. An image pixel is considered to be a seed if its gray values for each band fall in some representative interval. After that, our new seeded region growing algorithm is used to segment the image. This algorithm uses instance-based learning as its distance criteria. Using instance-based learning can take advantage of all the labeled data because it is an incremental learning approach that uses all the labeled pixels to update the classifier. The proposed weighting scheme is used to improve the classification results. Finally, according to the user needs, the regions are merged using ownership tables. The algorithm was tested on synthetic multispectral satellite images showing good results compared to the most commonly used parametric methods.

Instance-based learning algorithms are widely used due to their capacity to approximate complex target functions; however, the performance of this kind of algorithms degrades significantly in the presence of irrelevant features. This thesis also introduces a new noise-tolerant Instance-based learning algorithm, called WIBK that uses one or more weights, per feature per class, to classify integer-valued databases. A set of intervals that represent the rank of values of all the features is automatically created for each class, and the nonrepresentative intervals are deleted. The remaining intervals (representative intervals) of each feature are compared against the representative intervals of the same feature in the other classes to assign a weight. The weight represents the discriminative power of the interval, and is used in the distance function to improve the classification accuracy. The algorithm was tested on several real-world datasets from the UCI repository, and compared against other representative machine learning algorithms showing very competitive results.

This research work produced two papers:

- Octavio Gómez, Eduardo F. Morales and Jesús A. González. “Weighted Instance-Based Learning Using Representative Intervals”. Accepted to be published on: *Lecture Notes on Artificial Intelligence: Proceedings of the 6th Mexican International Conference on Artificial Intelligence*, 2007, Vol. 4827.

- Octavio Gómez, Jesús A. González and Eduardo F. Morales. “Image Segmentation Using Automatic Seeded Region Growing and Instance-Based Learning”. *Lecture Notes on Computer Science: Proceedings of the 12th Iberoamerican Congress on Pattern Recognition*, 2007, Vol. 4756, Pages 192 - 201.

Summary of contributions

The main contribution of this research is the design and development of a new seeded region growing algorithm based on instance-based learning to perform segmentation of multispectral satellite images for remote sensing applications. The most relevant issues addressed in this research can be summarized in the following key aspects:

- A new and competitive algorithm that combines seeded region growing and instance based learning to multispectral satellite image segmentation for remote sensing applications.

- A new algorithm for automatic generation of seeds based on histogram analysis. The algorithm finds the seeds required by the region growing algorithm.

- A new and competitive noise-tolerant weighted instance-based learning algorithm that uses a novel approach based on intervals of features to obtain weights useful to improve the classification task.

- The introduction of ownership tables to merge and provide semantic meaning to the segmented regions according to the user needs. This tables also are useful for multilevel segmentation.

Further research directions

This research has been mainly focused on the integration of seeded region growing and instance based learning, without taking into account some implementation issues related to data structures to improve performance or indexing schemes to help instance-based learning to find faster the closest example. Further research may include all these practical aspects in order to improve the implementation and the algorithm in general.

Other research direction is related with the weighting scheme. The proposed weighting scheme only works over integer-valued features; further research include the development of discretization methods to allow the weighting scheme to be applied to a wider range of data (domains). The region merging procedure can also be improved with the creation of an automatic method to obtain ownership tables; for example, a classifier trained with previous labeled maps.

Bibliography

[1] Adams, R., Bischof, L.: "Seeded region growing". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16. IEEE Computer Society, Los Alamitos, California (1994) 641–647

[2] Aha, D.W., Kibler, D.: "Noise-tolerant instance-based learning algorithms". Proceedings of the Eleventh International Joint Conference on Artificial intelligence. Morgan Kaufmann, San Francisco (1989) 794–799

[3] Aha, D.W., Kibler, D., Albert, M.K.: "Instance-based learning algorithms". Machine Learning, Vol. 6. Elsevier, Amsterdam (1991) 37–66

[4] Atkeson, C.G., Moore, A.W., Schaal, S.: "Locally weighted learning". Artificial Intelligence Review, Vol. 11. Springer-Verlag, Netherlands (1997) 11–73

[5] Blaszczynski, J., Graco, S., Slowinski, R.: "Multi-criteria classification - A new scheme for application of dominace-based decision rules". European Journal of Operational Research, Vol. 181. Elsevier, Amsterdam (2007) 1030–1044

[6] Blansché, A., Gancarski, P., Korczak, J.J.: "MACLAW: A modular approach for clustering with local attribute weighting". Pattern Recognition Letters, Vol. 27. Elsevier, Amsterdam (2006) 1299–1306

[7] Canada Center of Remote Sensing: Fundamentals of remote sensing On Internet: http://www.canadaremotesensing.org/remote Canada Center of Remote Sensing, Toronto (1998)

[8] Cleary, J.G., Trigg, L.E.: “K*: an instance-based learner using an entropic distance measure”. Lecture Notes in Computer Science: Proceedings of the 12th International Conference on Machine Learning, Vol. 2225. Springer-Verlag, Netherlands (1995) 108–114

[9] Cover, T.M., Hart, P.E.: “Nearest neighbor pattern classification”. IEE Transactions on Information Theory, Vol. 13. IEEE Computer Society, Los Alamitos California (1967) 21–27

[10] Deng, Y., Manjunath, B.S.: “Unsupervised segmentation of colortexture regions in images and video”. IEEE Transactions of Pattern Analysis and Machine Intelligence, Vol. 23. IEEE Computer Society, Los Alamitos California (2001) 800–810

[11] Fan, J., Zeng, G., Body, M., Hacid, M.: “Seeded region growing: and extensive and comparative study”. Pattern Recognition, Vol. 26. Elsevier, Amsterdam (2005) 1139–1156

[12] Frank, E., Witten, I.H.: “Generating accurate rule sets without global optimization”. Machine Learning: Proceedings of the Fifteenth International Conference Morgan Kaufmann, San Francisco (1998) 2343–2353

[13] Fu, K., Mui, J.: “A survey on image segmentation”. Pattern Recognition, Vol. 13. Elsevier, Amsterdam (1981) 3–16

[14] Gomez, O., Morales, E.F, González, J.A.: “Weighted instace-based learning using representative intervals”. Lecture Notes on Artificial Intelligence: Proceedings of the 6th Mexican International Conference on Artificial Intelligence (MICAI) 2007, Vol. 4827. Springer-Verlag, Berlin Heidelberg (2007) 420–430

[15] Gomez, O., González, J.A., Morales, E.F.: “Image segmentation using automatic seeded region growing and instance-based learning”. Lecture Notes on Computer Science: Proceedings of the 12th Iberoamerican Congress on Pattern Recognition (CIARP) 2007, Vol. 4756. Springer-Verlag, Berlin Heidelberg (2007) 345–251

[16] Gonzalez, R., Woods, R.: *Digital image processing*. 2nd edition. Horton, M. (Ed.) Prentice Hall, New Jersey (2002)

[17] Haddon, J., Boyce, J.: "Image segmentation by unifying region and boundary information". IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 12. IEEE Computer Society, Los Alamitos California (1990) 929–948

[18] Haralic, R., Shapiro, L.: "Survey: image segmentation techniques". Computer vision, graphics and image processing, Vol. 29. University of Utah, Salt Lake (1985) 100–132

[19] Haris, K., Efstratiadis, S., Maglaveras, N., Katsaggelos, A.: "Hybrid image segmentation using watersheds and fast region merging". IEEE Transactions on image processing, Vol. 7. IEEE Computer Society, Los Alamitos California (1998) 1684–1699

[20] Hart, P.E.: "The condensed nearest neighbor rule". IEEE Transactions on Information Theory, Vol. 14. IEEE Computer Society, Los Alamitos California (1968) 515–516

[21] John, G.H., Langley, P.: "Estimating continuous distributions in bayesian classifiers". Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, Vol. 11. Morgan Kaufmann Publishers, San Mateo (1995) 338–345

[22] Kass, M., Witkin, A., Terzopoulos, D.: "Snakes: active contour models". Proceedings of the 1st International Conference on Computer Vision IEEE Computer Society, Los Alamitos California (1987) 259–267

[23] Kononenko, I.: "Estimating atributes: analysis and extensions of RELIEF". Lecture Notes on Artificial Intelligence: Proceedings of the European Conference on Machine Learning Springer Verlag, Heidelberg (1994) 1935–1945

[24] Kurnaz, M.N., Dokur, Z., Olmez, T.: "Segmentation of remote sensing images by incremental neural network". Pattern recognition letters, Vol. 26. Elsevier, Amsterdam (2005) 1096–1104

[25] MVTec Software.: Halcon release 7.0 Munchen, GR. MVTec Software GmbH (2006)

[26] Mitchell, T.M.: *Machine learning*. Liu, C.L., Tucker, A.B. (eds.) McGraw-Hill, New York (1997)

[27] Mitra, P., Shankar, B.U., Pal, S.K.: "Segmentation of multispectral remote sensing images using active support vector machines". Pattern recognition letters, Vol. 25. Elsevier, Amsterdam (2004) 1067–1074

[28] Newman, D.J., Hettich, S., Blake, C.L., Merz, C.J.: UCI Repository of machine learning databases. On Internet: http://mlearn.ics.uci.edu/MLRepository.html University of California, California (1998)

[29] Paglieroni, D.W.: "Design considerations for image segmentation quality assessment measures". Pattern Recognition, Vol. 37. Elsevier, Amsterdam (2004) 1067–1617

[30] Pal, S.K., Ghosh, A., Shankar, B.U.: "Segmentation of remotely sensed images with fuzzy thresholding, and quantitative evaluation". International Journal of Remote Sensing, Vol. 21. Elsevier, Amsterdam (2000) 2269–2300

[31] Panl, N., Pal, S.: "A review on image segmentation techniques". Pattern Recognition, Vol. 26. Elsevier, Amsterdam (1993) 1277–1294

[32] Plat, J.: "Fast training of support vector machines using sequential minimal optimization". Advances in Kernel Methods - Support Vector Learning. MIT Press, Cambridge (1998) 767–776

[33] Quinlann, J.: "Induction of decision trees". Machine Learning, Vol. 1. Elsevier, Amsterdam (1986) 81–106

[34] Richards, J.A., Jia, X.: "Remote sensing digital image analysis". Springer Verlag, Heidelberg (1999) 854–894

[35] Robles, J.F., González, J.A.: Extracción de mapas temáticos a partir de la clasificación en imágenes satelitales. Masters Thesis Instituto Nacional de Astrofísica, Óptica y Electrónica (2007)

[36] Sezgin, M.: "Survey over image thresholding techniques and quantitative performance evaluation". Journal of electronic imaging SPIE, California (2004) 146–165

[37] Shamos, M.I., Hoey, D.: "Closest point problems". Proceedings of the Sixteenth Annual Institute of Electrical and Electronic Engineers Symposium on the Foundations of Computer Science. IEEE Computer Society, Los Alamitos California (1975) 151–162

[38] Shen, X., Spann, M., Nacken, P.: "Segmentation of 2D and 3D images through hierarchical clustering". Pattern Recognition, Vol. 31. Elsevier, Amsterdam (1998) 1295–1320

[39] Shih F.Y., Cheng, S.: "Automatic seeded region growing for color image segmentation". Image and Vision Computing, Vol. 23. Elsevier, Amsterdam (1998) 877–886

[40] Skidmore, A.: *Environmental modelling with GIS and remote sensing*. 2nd Edition. Taylor & Francis, New York (2002)

[41] Smith, E.E., Medin, D.L.: *Categories and concepts*. Harvard University Press, Cambdrige (1981)

[42] Stanfill, C., Waltz, D.: “Toward memory-based reasoning”. Communications of the ACM, Vol. 29. Association for Computing Machinery, New York (1986) 1213–1228

[43] Tahir, M. A., Bouridane, A., Kurugollu, F.: “Simultaneous feature selection and feature weighting using Hybrid Tabu Search/K-nearest neighbor classifier”. Pattern Recognition Letters, Vol. 28. Elsevier, Amsterdam (1997) 438–446

[44] Tesauro, G.: “Programming backgammon using self-teaching neural nets”. Artificial Intelligence, Vol. 134. Elsevier, Amsterdam (2002) 181–199

[45] Tremeau, A., Borel, N.: “A region growing and merging algorithm to color segmentation”. Pattern Recognition, Vol. 30. Elsevier, Amsterdam (1997) 1191–1203

[46] Weisstein, E.W.: The ANOVA test. From MathWorld–A Wolfram Web Resource. http://mathworld.wolfram.com/ANOVA.html

[47] Witten, I.H., Frank, E.: *Data mining: practical machine learning tools and techniques*. 2nd Edition. Morgan Kaufmann, San Francisco (2005)