Concepts Elaboration and System Architectures for Mining Very Large Image Archives

vom Fachbereich Elektrotechnik und Informatik der
Universität Siegen
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
(Dr.-Ing.)

genehmigte Dissertation

von

Diplom-Ingenieurin Inés María Gómez Muñoz

1. Gutachter: Prof. Dr.-Ing. habil. Otmar Loffeld
2. Gutachter: Prof. Dr.-Ing. habil. Mihai Datcu
Vorsitzender: Prof. Dr. rer. nat. Rainer Brück

Tag der mündlichen Prüfung: 08.07.2009

gedruckt auf alterungsbeständigem holz- und säurefreiem Papier

To my grandparents

Abstract

During the last decades, satellite technology has been outstandingly improved, providing huge amounts of Earth Observation (EO) data to be processed and stored. The availability of very high resolution sensors has encouraged the birth of new domains for remote sensing applications. Relatively new fields in this frame are Image Information Mining (IIM) and Content Based Image Retrieval (CBIR). These fields are born to provide solutions for querying very large EO archives by content. This dissertation tries to contribute on the IIM domain, providing new image processing algorithms and optimization processes for mining image databases.

The study of an IIM system can be focused on signal processing methods, data compression, semantic knowledge discovery, human-machine interaction or system architecture design. Thus, the system can be divided in three modules: on one hand, we have the off-line tasks, consisting of signal and image processing methods. The extracted information of these algorithms is based on a hierarchical Bayesian representation, and usually, is very time consuming. On the other hand, we explore the on-line actions that are performed at real time through an interaction with the user. Finally, an optimal software architecture where all these concepts are merged has to be studied. In this thesis, contributions on these three modules are provided.

We begin studying multi temporal high resolution image analysis under different illumination conditions and strong background clutter. The aim is to build a target detection map through a synergy of image processing methods. However, we can be faced with a common problem while extracting information from EO data, which is the estimation of parameters. Often the accuracy of the methods is strongly dependent on the selection of parameters and it is difficult to a priori know the optimum one. This is the motivation for the second contribution that deals with this problem. To cope with it, we implement an algorithm based on clustering features that uses information and rate distortion theories to help in the assessment of parameters.

One of the main characteristics of an IIM system, is its potential to learn though human interaction. The user provides some examples of his interests, and based on them, the system learns his preferences, searches for them in large archives, and returns similar contents to the user provided ones. In this framework, we developed a multiple classifier, that enables the user to provide more than one example type. Thus, the system will be

queried for different features, refining the query results and search accuracy.

In order to be an operable and useful system, all new features proposed in this dissertation have to be accomplished in a modular system architecture. The system, from the software design point of view, must be opened, standard compliant and accessible though Internet. The software architecture design of the IIM system is the last contribution of this thesis. For building the system, we have to consider the following aspects:

- how to manage the large data volume of original and processed images;
- the automatization of tasks as loading new data, extracting features or generation of thematic maps;
- how to adapt the system to the user knowledge, that is, the image interpretation has to be adapted to the symbols the users are able to recognize and to the specific semantics of their domains;
- how to perform the man-machine communication through a continuous interaction and exchange of knowledge.

To achieve these aims, the new system will be composed of web services and Java interfaces orchestrated by a workflow engine.

Zusammenfassung

In den letzten Jahrzehnten ist die Satellitentechnologie auerordentlich verbessert worden und riesige Fernerkundungdateien werden heutzutage prozessiert und abgespeichert. Die Verfügbarkeit von Sensoren mit sehr hoher Auflösung hat die Erstellung neuer Bereiche für Fernerkundungsanwendungen gefördert. Relativ neue Felder sind in diesem Rahmen Image Information Mining (IIM) und inhaltsbasierte Bildsuche. Beide wurden konzipiert um eine Lösung für inhaltsbasierte Abfragen auf Bilddatenbanken formulieren zu können. Der Beitrag dieser Dissertation zum IIM Bereich besteht aus neuen Algorithmen für Bildanalyse und Prozessoptimierung von Bilddatenbaken.

Die Studie von einem IIM System enthält verschiedene Module: Prozesssignale, Datenkomprimierung, semantische Kenntnissuche, Mensch-Maschine Interaktion oder Aufbau der Systemarchitektur. Somit kann das System in drei Komponenten aufgeteilt werden: auf einer Seite haben wir die offline-Aufgaben, das sind Methoden zur Signal- und Bildverarbeitung. Die mit diesen Methoden extrahierte Information ist auf einer Bayesschen hierarchischen Darstellung basiert, und diese Algorithmen sind normalerweise sehr zeitaufwendig. Auf der anderen Seite, untersuchen wir die Online-Aufgaben, die in Echtzeit durch ein benutzerinteraktives Lernen ablaufen. Zuletzt muss eine optimale Softwarearchitektur, wo alle die Konzepte integriert sind, studiert werden. Beiträge zu diesen drei Modulen sind im Rahmen dieser Doktorarbeit geleistet.

Wir fangen mit der Implementierung neuer Algorithmen für Erkennung des Targets durch Änderungsanalyse von Mehrzeit- hyperspektralen Hochauflösungsbilder unter verschiedenen Lichtbedingungen und starken Hintergrundstördaten an. Unser Ziel ist es, eine Zielerkennungskarte durch die Synergie von Bildanalysealgorithmen zu bilden. Wenn wir Eigenschaften von Fernerkundungsdateien extrahieren, werden wir allerdings das Problem der Parameterabschätzung auflösen müssen. Die Genauigkeit der Methoden ist oft von der Parameterauswahl stark abhängig, und normalerweise ist es sehr kompliziert, von vornherein die optimale Parameterkombination zu kennen. Das ist die Motivation für den zweiten Beitrag dieses Dissertation: wir implementieren einen Algorithmus, der durch Informations- und Rate Distortionstheorie die Parameterqualität auswerten kann.

Eine der Haupteigenschaften eines IIM Systems ist seine Fähigkeit, über Benutzerinteraktion zu lernen. Der Benutzer liefert Beispiele seiner Interessen und das System lernt die Benutzerbevorzugungen, sucht nach Bildern mit ähnlichem Inhalt auf der Datenbank.

In diesem Rahmen, steht die Entwicklung eines neuen Mehrfach-klassifizierers, die es dem Benutzer erlaubt, mehr als einen Beispielstyp in das System einzugeben. Das System wird dementsprechend über unterschiedliche Eigenschaften abgefragt, was die Ergebnisse der Abfrage und die Suchgenauigkeit erheblich verbessert. Das bildet den dritten Beitrag dieser Doktorarbeit.

Um ein betriebsbereites und anwendbares System zu bilden, sind die obengenannten Beiträge in eine modulare Systemarchitektur zusammen zu bringen. Das System muss von dem Standpunkt des Software Design offen, standardkonform und über das Internet zugänglich sein. Das Design der Softwarearchitektur des IIM Systems ist der letzte Beitrag dieser Dissertation. Um das System erfolgreich aufzubauen, müssen wir folgendes beachten:

- wie riesige Datenmengen von originalen und prozessierten Bilddaten zu verwalten sind;
- wie die Automatisierung der Aufgaben, z.B. das Aufladen einer neuen Datei, die Eigenschaftenextraktion oder die Erzeugung von thematischen Karten zu verwirklichen ist;
- wie das Systemanpassung an der Benutzerkenntnis ermöglicht wird (d.h. die Bildinterpretation muss an den Systemsymbolen angepasst werden, die der Benutzer erkennt, und an der spezifischen Semantik seines Bereiches);
- wie die Mensch-Maschine Kommunikation durch eine duchgehende Interaktion und Kenntnisaustausch durchgeführt wird.

Um diese Ziele zu erreichen, wird das neue System aus Web Services und Java Schnittstellen bestehen, die von einem Workflow-Engine orchestriert sind.

Resumen

En las últimas décadas, la tecnología satelital ha mejorado considerablemente, y hoy en día ofrece ingentes cantidades de datos de Observación de la Tierra para su posterior proceso y almacenamiento. La disponibilidad de sensores de muy alta resolución ha permitido el nacimiento de nuevos campos de investigación y aplicación de la teledetección. Algunos de estos nuevos ámbitos son la Minería de Información sobre Imágenes, en inglés Image Information Mining (IIM), y la Recuperación de Imágenes Basada en su Contenido, en inglés Content Based Image Retrieval (CBIR). Estos nuevos campos ofrecen soluciones de consulta sobre el contenido de una imagen almacenada en grandes sistemas de almacenamiento. Este doctorado contribuye en el campo de IIM, creando nuevos algoritmos de procesado de imágenes, así como optimizando los procesos de minería sobre bases de datos que albergan grandes cantidades de imágenes satélite.

En el estudio de un sistema IIM podemos distinguir varias tareas: procesado de la señal, descubrimiento de conocimiento semántico, interación hombre-máquina o diseño sobre la arquitectura del sistema, entre otros. De esta forma, el sistema lo podemos dividir en tres módulos: por un lado tenemos las tareas que se ejecutan fuera de línea, o también llamadas en modo off-line, que son aquellos métodos de procesado de señal e imagen. Estos algoritmos extraen información de los datos basándose en una representación Bayesiana y jerarquizada de los mismos que, a menudo, necesitan mucho tiempo de ejecución. Por otro lado, tenemos las tareas que se ejecutan en línea, o en modo on-line, que son aquellas acciones en las que el usuario interacciona directamente con el sistema. El último módulo consistirá en diseñar la arquitectura software óptima del sistema IIM para que albergue todos estos conceptos. Esta tesis contribuye en los tres módulos descritos anteriormente.

Empezaremos realizando un análisis de imágenes multi-temporales de alta resolución tomadas con diferentes condiciones de iluminación y que, además, poseen gran distorsión. El objetivo de este primer estudio es la obtención de un mapa de detección de cambios, aplicando una sinergía de distintos métodos de procesado de imagen. En los procesos de extracción de información de imágenes satélite, la estimación de parámetros suele representar un problema a tener en cuenta. La precisión de este tipo de procesos es extremadamente dependiente de la selección particular de los parámetros y, a priori, es difícil conocer el valor óptimo de los mismos. La segunda contribución de esta tesis intenta dar solución a este problema. Para ello, hemos implementado un algoritmo basado

en técnicas de clustering que, aplicando la teoría de la información e índices de distorsión, es capaz de estimar el valor óptimo de uno de los parámetros.

Una de las principales características de un sistema de IIM es su potencial de aprendizaje mediante la interacción con el usuario. Éste introduce ejemplos de su propio interés, y así el sistema aprende las preferencias del usuario. Seguidamente, el sistema realiza una búsqueda en su repositorio de imágenes que contienen la misma información que los ejemplos introducidos por el usuario, y le devuelve a éste los resultados. En este marco, hemos desarrollado un clasificador múltiple que permite al usuario introducir más de un tipo de ejemplo de interés. De esta forma, el sistema buscará en su archivo imágenes que contengan distintas características, refinando, por tanto, las búsquedas.

Por último, para construir un sistema que sea operable y utilizable, todas los algoritmos propuestos anteriormente en esta tesis deben ser integrados en una arquitectura modular. El sistema, desde el punto de vista de diseño software, debe ser abierto, estandarizado y accesible a través de Internet. El diseño de la arquitectura software de nuestro nuevo sistema de IIM constituye la última contribución de este doctorado. Para construir dicho sistema, debemos tener en cuenta los siguientes aspectos:

- cómo gestionar el gran volumen de datos, tanto de imágenes originales como procesadas;
- la automatización de tareas, como por ejemplo, la carga de datos, la extración de características o la generación de mapas temáticos;
- cómo adaptar el sistema al conocimiento específico del usuario, o dicho de otra forma, la interpretación de imágenes ha de adaptarse a la simbología que el usuario es capaz de interpretar, así como a la semántica específica del campo de trabajo;
- cómo ejecutar la comunicación hombre-máquina a través de una interación continua y un intercambio de conocimiento.

Para conseguir estos objetivos, el nuevo sistema de IIM se basará en servicios web e interfaces Java que serán orquestrados por un motor de flujos de trabajo.

Contents

List of Figures

List of Tables

1

Introduction

Classical systems for Earth Observation (EO) data access allow querying the image catalogues by metadata, i.e. geographical position, sensor type, etc. The feature extraction and classification is usually performed by a visual inspection of the data or by using suitable but not automatic algorithms. Available datasets are very small due to the valuable price, and experts in EO belonging to an specific application domain are demanded. Nevertheless, this becomes very expensive, due to the strong requirement of human intensive activity.

Nowadays, Image Information Mining (IIM) and Content-Based Image Retrieval (CBIR) fields are producing an important development in querying huge EO archives by image content. The standard procedure in a CBIR supports an interactive query based retrieval where the user takes part in the process. Through training examples of image content, the user can refine the query making decisions by visual similarity. Thus, these systems avoid the ambiguity of text annotation, encouraging the visual content query and definition of semantic concepts. Examples of these systems are Query by Image Content (QBIC)[60] from IBM and Photobook [8] developed by MIT, or more recent ones, VisiMine [48][49] and IKONA [70] systems. However, a CBIR system does not always retrieve the desired image due to the semantic gap existing between image signal and semantic concepts. Thus, a still open issue is to find visual attributes that match correctly with image features and to use them in the retrieval process. Although there exist several CBIR systems in the literature, this field is still in its infancy.

The concept envisaged in this thesis, shall provide solutions how to access to large image datasets through information mining, and content based image retrieval. In there, the user-defined semantic image content interpretation shall be linked through Bayesian networks to a completely unsupervised content index. Based on this stochastic link, the user will be able to query the archive for relevant images obtaining a probabilistic classification of the entire image archive as an intuitive information representation. This semantic knowledge will be adapted to the user providing image interpretation symbols according to the specific domain. Finally, we have to take care on the design of such a system in order to be operable, maintainable, accessible and user oriented.

In this chapter, after this first introduction of IIM and CBIR fields, in Sec. 1.1, the motivation of writing this dissertation is presented. Once the need of building such a system is described, we focus on the different perspectives of the system, and therefore, of this thesis: from the signal processing viewpoint and from the system architecture aspect. This will be detailed in Sec. 1.2. Sec. 1.3 introduces the main contributions of this dissertation, giving a representative example of each one. We conclude this chapter presenting the outline of the organization of this dissertation.

1.1 Motivation

In the past, EO image archives resulted in huge repositories where less than 5% of the data where used. The access to this information was based on large metadata files through sensor type, geographical information or image name, and the way to obtain the data was mainly in optical supports. With the new generations of high resolution EO sensors and the big amount of data a ground station receives per day, new applications to help in image classification and image understanding have to be built. The progress in information retrieval, computer vision, and image analysis makes it possible to establish very complete bases of algorithms and operators. An specialist in remote sensing or image processing now has the tools that allow him to configure applications solving complex problems of image understanding. However, in reality, EO data analysis is still performed in a very laborious way at the end of repeated cycles of trial and error.

To overcome this, systems where images are retrieved by content in an interactive and user adapted form are needed. Systems where users coming from diverse disciplines can contribute with different knowledge being this knowledge shared among domains. Such a system should provide novel advanced approaches for EO information processing based on Human-Centered Concepts (HCCs). New features and functions allowing improved feature extraction, search at semantic level, availability of collected knowledge, interactive knowledge discovery and new visual user interfaces should be integrated.

Furthermore, the system should be simple, concise, reliable, timely, understandable, compliant with user processes and able to deliver real-time knowledge in response to natural language queries. These qualities let the system provide large benefits to users, easy scientific investigations, and support the growth of value-adding industry, service providers and market, by permitting the creation and provision of new services based on information and knowledge.

1.2 Positioning this Dissertation

This dissertation aims at the development of methods to enlarge Knowledge-driven Information Mining (KIM), the IIM system of the German Aerospace Center (DLR), and at the improvement of some already operational approaches. Because an IIM system is based on a compendium of theoretical fields, this thesis has been developed under the study of theories from parameter estimation, Bayesian inference or machine learning. In addition, building an operational system, basics on machine communication, software technologies and system architectures are required. Thus, this dissertation can be seen from two perspectives: a signal processing one to overcome the new methods in IIM domain, and a second one dealing with the design of a new version of KIM.

SIGNAL PROCESSING PERSPECTIVE

The signal processing perspective analyzes the thesis from an information hierarchy point of view. This hierarchy is structured at three levels. In the first one, we review the state of the art in signal processing, statistics, mathematical modeling and information theory. Theories coming from these fields as parameter estimation, clustering, learning, Bayesian inference and rate distortion theory are studied, building the theoretical background. At a second level, we find the main contributions in signal processing domain: we develop a change detection algorithm for multi temporal high resolution images by applying parameter estimation theory. A second contribution tries to measure the quality of parameters in extracting spatial information as is the analyzing window size. Rate distortion, clustering and parameter estimation theories are behind this method. The last contribution builds a multiple classifier for the learning process allowing the user to obtain maps defined by several semantics. In this case, Bayesian inference and learning theories are used. Once the novel approaches are implemented, they are integrated in the IIM system establishing the third level of information. Fig. 1.1 summarizes the signal processing perspective of this dissertation.

SYSTEM ARCHITECTURE PERSPECTIVE

From the system architecture point of view, we have to integrate the novel contributions in the IIM system in an operable, reliable and automatic way. In this case, the contribution of the dissertation focuses on the study, analysis and design of a new IIM system concept in terms of technology, storage, accessibility to data, modularity of functionalities, interoperability and user-oriented tasks. Fig. 1.2 depicts how the system represents the study object analyzing the best form to develop and integrate all the modules.

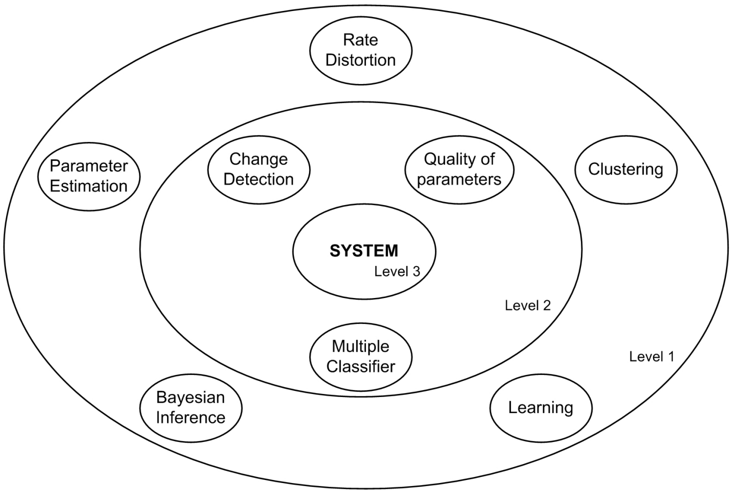


Figure 1.1: Signal processing perspective of the dissertation.

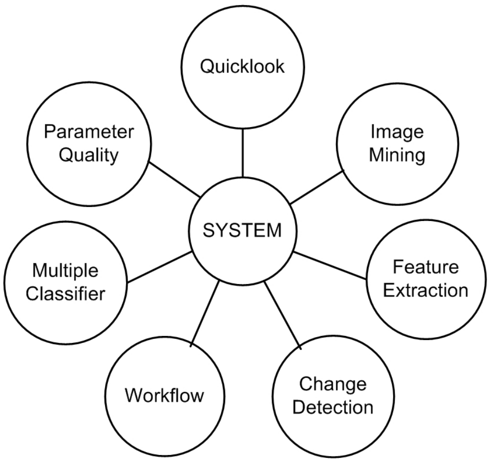


Figure 1.2: System architecture perspective of the dissertation.

1.3 Contributions

This dissertation presents contributions in different fields of research always preserving the objective of building an operational and user oriented IIM system. Typically, an IIM system is composed of a chain of processes, starting with the original image until it is ingested in the repository. These processes consist of a feature extraction phase and a clustering or classification module. Once processed images are stored in the database, a graphical interface allows the user to provide training examples in order to query the archive for similar content images. In this section, we present the contributions of this thesis in each module of an IIM system for a better understanding of the flow of this dissertation.

1.3.1 Signal Processing Perspective Contributions

In this section, we present some examples of the contributions in the signal processing perspective presented in previous section.

- **Target Detection in Multi Temporal Images**

  In the feature extraction process of the ingestion chain a target detection by change analysis of multi temporal images method is developed. Observing the content of images, we perform different algorithms to extract vegetation or shadows from them, and them, a synergy of the results is completed. Thus, this method builds map of changes from multi temporal images. Fig. 1.3 shows a map of changes generated for a Daedalus multi temporal image.

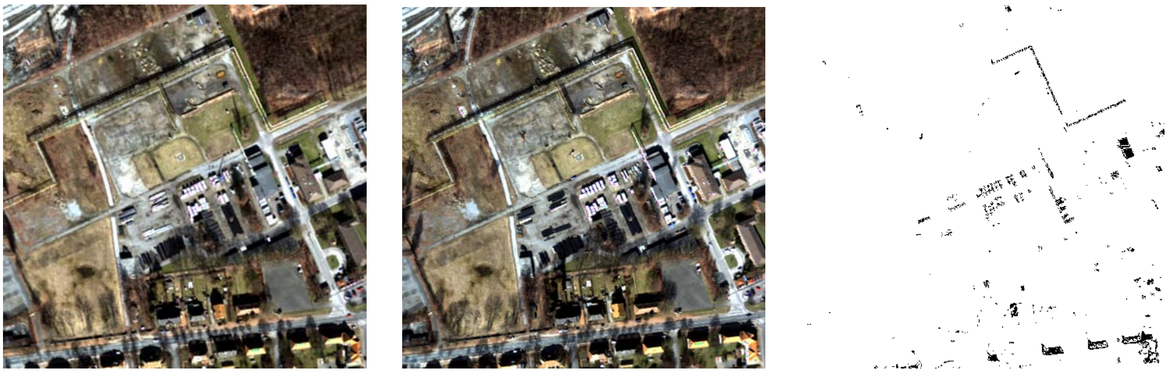


Figure 1.3: Example of target detection in multi temporal images contribution. Images from left to right: original Daedalus image at time 1; original Daedalus image at time 2; map of changes between images.

- **Quality of Parameters**

  Following the chain of processes, presented before, we arrive at the clustering phase. In this module, we measure the quality of the parameter estimation by applying rate distortion theory. It helps to evaluate the optimum size of the analyzing window which is a parameter of a spatial information extraction algorithm. The result will be dependable on the spatial diversity presented on the image. It is placed at clustering phase, due to the need of executing a classification algorithm to measure distortion between generated clusters. Fig. 1.4 displays how the distortion values differ depending on the number of clusters have been considered. The example shows

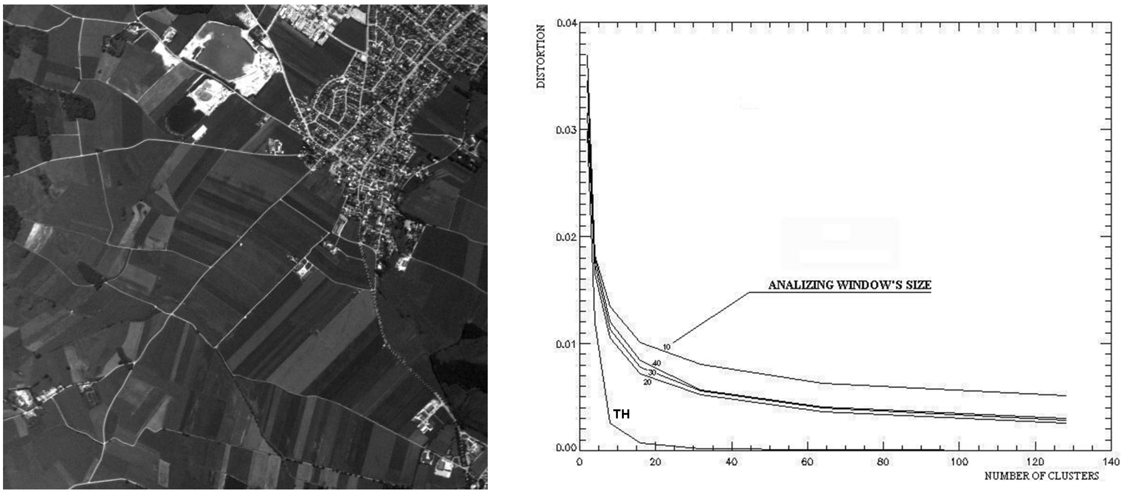


Figure 1.4: Example of estimating the optimum parameter. Image on the left corresponds to original Daedalus image from Oberpfaffenhoffen (Germany), and on the right, distortion curves for that image are represented. The lowest curve plots the theoretical bound (TH) computed for a window of 20 x 20 pixels.

the result of the method applied to a Daedalus image, whose optimum window size is 20 times 20 pixels. The lowest curve plots the theoretical bound (TH) computed for a window of 20 x 20 pixels.

- **Multiple Classifier**

  Once images are processed and ingested in the repository, the user can provide examples to the system to show its interest. But, usually the user is interested in searching in catalogues for different contents or building thematic maps consisting in more than only one feature. The multiple classifier has been developed to provide the IIM system the ability to be trained for more than one semantic class, and therefore, give the possibility to search in the archive for images containing several kinds of content. In addition, this can be used to provide thematic maps to the user. Fig. 1.5 shows an example of this contribution.

1.3.2 System Architecture Perspective Contributions

Once the modules of the ingestion chain have been developed, they have to be assembled building a functional IIM system for EO. The contribution of the dissertation in this perspective is the design of a software architecture studying the available technology, communication protocols, workflow tools and interoperable standards. Fig. 1.6 displays the software architecture design based on web services and orchestrated by a workflow

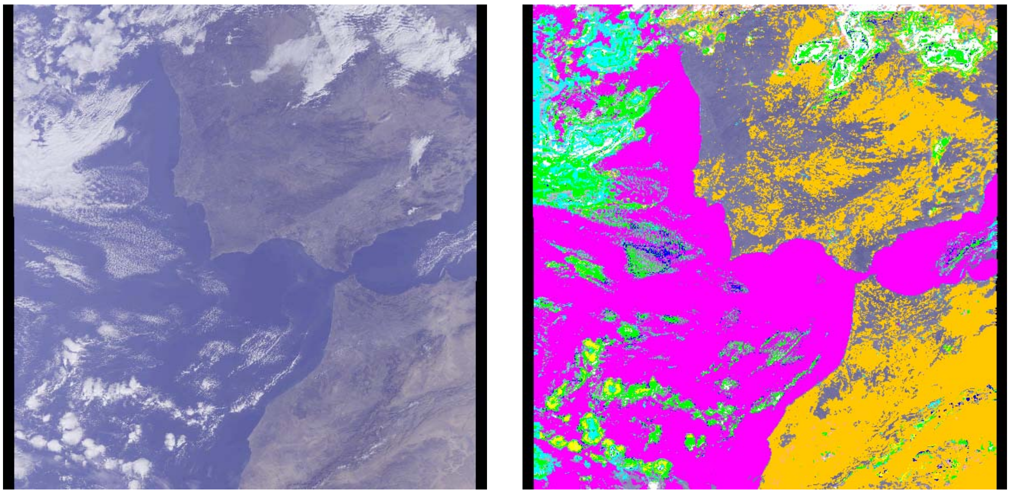


Figure 1.5: Example of the multiple classifier. The left image corresponds to the original Meris image, and the image on the right is the thematic map of the original image for several semantic classes as land, water or cloud.

engine which decides the flow of processes and data over the IIM system.

1.4 Outline of the Dissertation

Chapter 1 has given a brief introduction of IIM systems pointing out the motivation of building such a system. In addition, the two perspectives of the dissertation are described mentioning the main contributions.

Chapter 2 outlines the different modules that constitute a generic architecture of an IIM system presenting the state of the art of each one. Then, an overview of existing systems for CBIR is provided.

The basis of this thesis is KIM, an IIM system that provides solutions to access image archives by content. Such a system is based on a hierarchical representation of semantic abstraction levels: feature extraction, content index and user defined semantic concepts. In **chapter 3**, KIM and its posterior versions are presented.

In **chapter 4**, the theoretical background in stochastic modeling, Bayesian inference and information theory are studied. We discuss probabilities, stochastic processes, parameter estimation and basic measures of information that will be applied in the following chapters.

Chapter 5 focuses on the feature extraction level of the information hierarchy. Start-

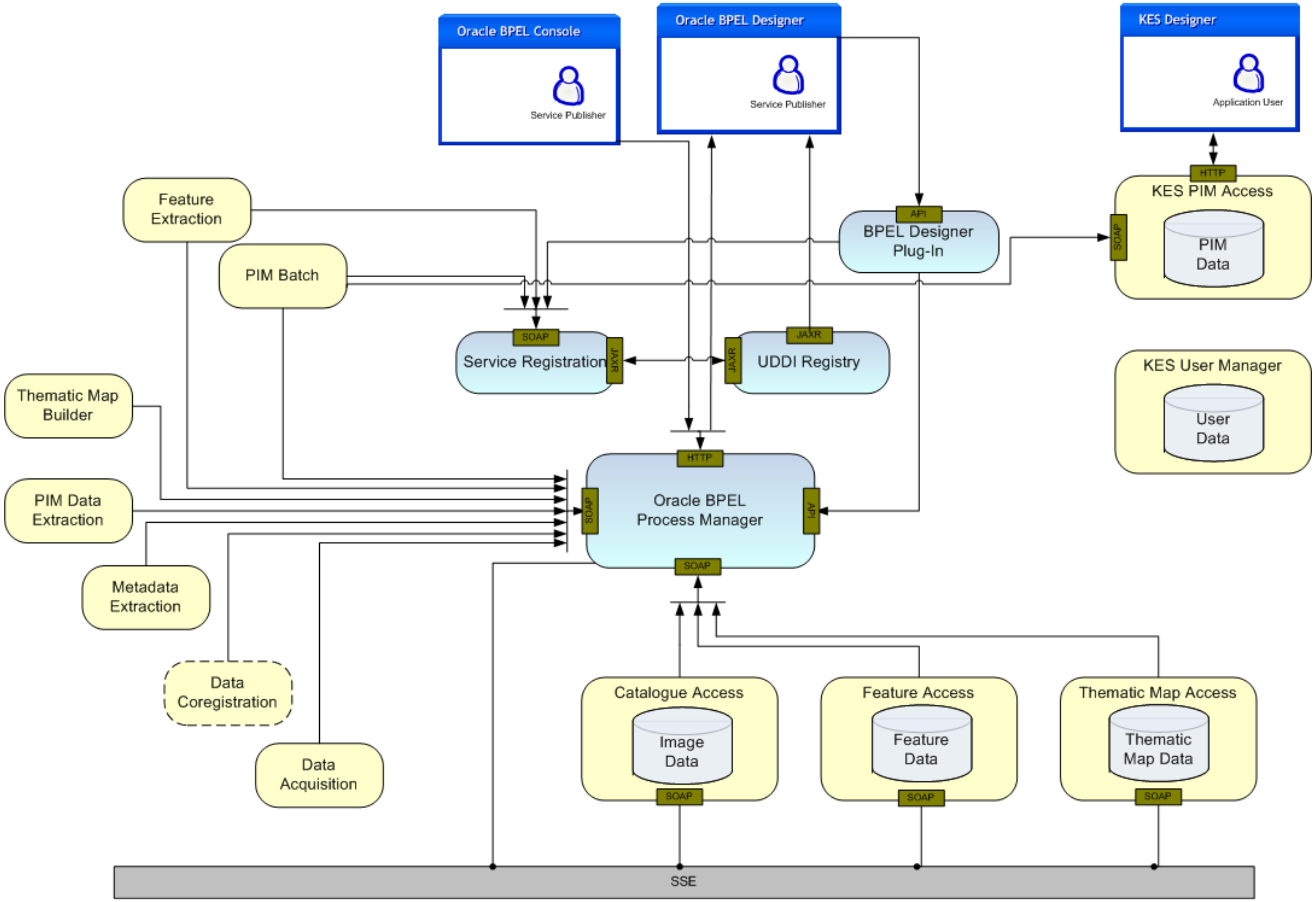


Figure 1.6: Software architecture design of an IIM system for EO.

ing from the original image, new algorithms will be detailed for target detection by change analysis of multi temporal images. Other algorithms for linear and spatial information extraction and data compression are also described.

The feature space represents a big amount of information that must be compressed to achieve an optimized performance saving storage capacity. Thus, features are grouped together through an unsupervised classification technique. In **chapter 6**, we present different variations of the well known k-means algorithm.

In **chapter 7** concepts introduced in chapters 5 and 6 are applied trying to estimate optimum parameters for a spatial information extraction method. To overcome this, once content indexes are obtained from texture parameters, several concepts from information theory are studied resulting in an optimization of parameters adjusted to data resolution.

Once features are compressed and indexed, we obtain a vocabulary of signal classes containing no semantic meaning. **Chapter 8** details how these unsupervised classes are provided with user defined semantic labels through an interactive learning process. In this process, the user interacts with the system by giving positive and negative examples showing his interest. In this chapter, a novel multiple classifier will be described, which provides the system the ability of training over one image for more than one label.

Chapter 9 discusses the design of the overall architecture of a new version of our IIM system. In this new system, all previous concepts are integrated. In this chapter, the requirements are pointed out and the newest technologies in distributed architectures and workflow engines are studied. Finally, a new system design for an optimized IIM system is proposed.

Finally, **chapter 10** exposes different applications, where our IIM system can be applied. We describe two examples of relevance: a multi temporal analysis for a target detection application, and a feature classification of medium resolution images.

We conclude this dissertation in **chapter 11**, summarizing the main results and itemizing the value of the contributions of this thesis.

Appendix A presents the characterization of the data that has been used through this dissertation; **appendix B** points out the details of two mathematical demonstrations about rate distortion and mutual information; **appendix C** summarizes the acronyms used in this thesis.

2

Overview of Existing Mining Systems

Image satellite sensors acquire huge volumes of imagery to be processed and stored in big archives. An example of such an archive is the German Remote Sensing Data Center (DFD) at Oberpfaffenhofen, Germany, that receives about hundreds of GigaBytes per day of data entailing 10^4 GigaBytes in the repository. To provide access to this data, web applications have been developed, e.g. the DLR EOWEB [30], to retrieve images according to meta information such as date, geographical location or sensor. Alexandria Digital Library [1] is another example of accessing remote sensed imagery through its meta information providing a distributed searching mechanism for retrieving geospatial referenced data collections. It is able to search different types of databases placed at different locations. The software enables to implement web clients as Globetrotter [4] or Gazetteer [2]. These systems based on meta information retrieval allow only constrained queries giving no information about the content, and consequently, no content based retrieval is offered.

At the conference on database techniques for pictorial applications that took place in 1979 in Florence, Italy, the pursued aim was the integration of databases with image processing. This idea evolved, in 1990, promoting a new field, called Content Based Image Retrieval (CBIR). In 1998, CBIR got married with Data Mining and Knowledge Database Discovery (KDD) emerging, in 2000, the Image Information Mining (IIM) field. This new domain requires expertise in image processing, database organization, pattern recognition, content-based retrieval and data mining: image processing indicates the understanding and extraction of patterns from a single image; content-based retrieval is characterized by retrieving images from the archive based on their semantic and visual contents; spatial data mining denotes the extraction of spatial relationships and patterns from remote sensed images not explicitly stored in an spatial database. An IIM system provides users the capability to deal with large collections of images by accessing into large image databases and also to extract and infer knowledge about patterns hidden in the images, so that the set of relevant images is dynamic, subjective and unknown. It enables the communication between heterogeneous source of information and users with diverse interests at high semantic abstraction.

In general, an IIM system presents two fundamental modules: a computationally expensive component where image processing and classification algorithms are executed, and an interactive part, where queries are introduced by the user and relevant images are retrieved. Fig. 2.1 represents the typical flow of a data in an IIM system: original data arrive at a feature extraction module, where main image characteristics are computed; then, these features are compressed and indexed in a database; in a second module, the archive is queried by the user for similar features computing similarity measurements for an optimal image retrieval.

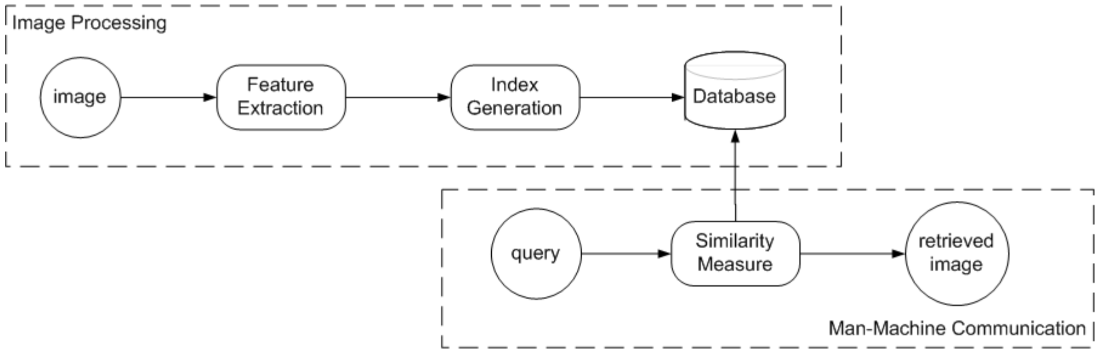


Figure 2.1: Architecture of an Image Information Mining system.

This thesis is organized following this generic concept of an IIM system, so that each chapter corresponds to one component. For this reason, in Sec. 2.1 we will focus on a general presentation of the modules and methods, leaving the state of the art and further details about them for the following chapters. Sec. 2.3 presents existing experimental and commercial CBIR systems.

2.1 Image Information Mining System Architecture

As depicted in Fig. 2.1, the generic concept of an IIM system requires several processing modules: extraction of properties from images, reduction and content indexation and communication between users and system. In this section, we will present the state of the art of these modules giving an overview of existing techniques in these fields.

2.1.1 Feature Extraction

In general, by image we understand picture, thus relating it to the (human) visual perception and understanding. A picture is characterized by its primitive features such as color, texture or shape at different scales. Thus, an image will be represented as a multidimensional feature vector acting as signature. Some classical techniques to characterize an image are the following:

- Color: Color information has been an important feature in image processing and computer vision. There exist different color models or color spaces, each one being useful for a specific application. A digital imaging system, typically represents color images in red, green, blue using the RGB space. Another one related with the perception of the colors by human beings is the HSV (Hue, Saturation and Intensity) color space. This one describes the property of the surface reflecting the light (hue), measures the colorfulness or whiteness (saturation) and the brightness (intensity) of colors. Often a full color image providing the three colors (RGB) in each pixel is needed, being essential to interpolate missing colors with the information of neighboring pixels. There are nonadaptive algorithms [79] as nearest neighbor replication and bilinear interpolation, and adaptive algorithms [79] based on pattern matching or edge sensing interpolation. On the other side, a common practise in image processing is the statistical analysis of color histograms, due to the strong correlation between objects and color in an image.

- Texture: Texture is a very interesting feature to characterize the spatial structure of an image. This is an active research field where parametric and non parametric methods are applied. Haralick's co-occurrence [67] technique based on the computation of the gray-level co-occurrence matrix for several values of displacement, orientation and image quantization levels is an effective method in texture analysis. Other algorithms based on wavelet transformations as the computation of Gabor filter [64] can also be applied.

- Shape: Shape of objects must be invariant to translation, rotation and scale of the image and is characterized in two senses: boundary-based, that considers the object outer contour, and region-based, where the whole shape region of the object is analyzed. In this sense, Fourier descriptors are suitable for transforming boundaries into shape features, and moment invariants for the extraction of geometric object region. A modified Fourier descriptor that preserves the invariance of geometric transformations and noise is proposed in [84]. A common practise before applying shape techniques, is to segment the image in small regions. Comaniciu [20] presents this approach based on the mean shift method for density gradients estimation.

- Topology: topological properties of an image such as number of connected or disconnected components, do not change when an image is rotated, scaled, translated, stretched or deformed. One example of characterizing an image through its topological properties is the computation of the Euler number [79]. It is defined as the difference between number of connected components and number of holes in a binary image. An extension of the Euler number defined for binary images, is the Euler vector [79] that can be applied to gray-level images. Segmentation techniques may also help in the extraction of topological features.

2.1.2 Multidimensional Indexing

In the CBIR and IIM domain, the concept of multidimensional indexing differs from the one in a traditional database management system. In here, an index consists of the structure that provides access to the database in terms of record organization. In IIM, once an N-dimensional feature vector is obtained, images are assigned to a suitable content based description extracted from these features. These content descriptors are then organized into a data structure for retrieval.

In multidimensional indexing, the following items must be considered:

- Reduction of dimensionality: Due to the huge amount of images and extracted features, normally the dimensionality of the information at the indexing step is very high. This complicates the management of the feature vector rendering its computation very expensive. For this reason, mechanisms for reducing the dimension of the feature space must be considered. Among these methods, Karhunen-Loève transform [79] and the Discrete Cosine Transform (DCT) [50][82] are often considered.

- Clustering: Extracted features with similar content must be grouped together through a classification algorithm. In this case, pixels containing similar features belong to the same class. Existing clustering techniques can be classified into two main groups: distance-based and model-based [87] approaches. In the first group, we mention methods based on Euclidean and Mahalanobis distances, and to the second group belong algorithms based on an a priori specified model, such as Gaussian mixture models or Markov chains. A widely used method in IIM is the unsupervised k-means algorithm that will be further studied in chapter 6.

- Data structure for content based retrieval: Once a clustering algorithm is performed, a data structure for indexing descriptors to semantic content must be selected. The common used methods are tree-based indexing techniques, as multidimensional binary search trees or R-trees, and hashing-based ones.

2.1.3 Content-Based Image Retrieval

Usually CBIR is limited by the semantic gap existing between signal classes and semantic labels. Li et al. [55] propose a context sensitive Bayesian network to infer the semantic concept of regions or classes. Semantic score functions based on region features (spectral and texture) are computed to link semantic concepts to regions. Aksoy et al. [71] suggest a Bayesian framework to cope with the semantic gap problem. They introduce a visual grammar that builds a hierarchical semantic model from pixel level to region and scene levels. Pixel-level characteristic provides classification by automatic fusion of primitive features; then, at region-level through a segmentation algorithm land cover labels are defined; and scene-level represents the spatial relationship among regions. Thus, the visual grammar consists of two learning steps, where naive Bayesian classifiers are applied: a probabilistic link between features and semantic labels, and a fuzzy modeling to link regions and scenes. Once the visual grammar is built, the image classification process aims at finding representative region groups that describe the scene. The procedure consists of modeling the labeled regions by a Dirichlet distribution based on the number of training examples containing a certain region group, and then, assigning the best matching class to image by using the maximum a-posteriori rule.

In order to provide the system the ability to search at query-time for images with similar features, a similarity metric for the comparison of objects or image properties must be defined. If we want a realistic measure, computer and human judgments of similarity should be generally correlated. If this condition is not met, images returned by the system will not be those desired by the user. These techniques are often based on distances or on an specific domain as histogram intersection, neural networks, shape measures or graph matching. Queries like ”retrieve images containing an specific content” or ”retrieve images that do not contain a particular object” can be asked to a CBIR system.

2.2 Semantic Learning for Content-based Image Retrieval

Apart from primitive features, images can also be interpreted based on knowledge such to be appropriately understood in some context. The knowledge consists of the ensemble of existing information, known causalities and other types of associations between information and concepts. This knowledge is represented in form of semantic concepts assigned to signal features.

Later, when querying images with similar semantic content, some comparison between query image target and other images in the archive must be performed. This comparison, which results in similarity measurements, can be carried out at different levels. At pixel level, Euclidean or angular distances that usually require an expensive computation time.

At a higher level, in the multidimensional space, we find features extracted from data representing the main characteristics of the image as color, texture or shape. At this level, the distance measure is computed between the feature vector and the query feature vector. Because we stay in the multidimensional space and geometric distances demand a high computation time, probabilistic calculations are used. The main problem of using feature vectors for querying images with similar content, is that often, the appearance of an image does not correspond to its semantic meaning, making the returned images only partially respond to the users query. Therefore, at object or region level, the highest level of abstraction, an image is represented by its objects, and a semantic label is assigned to each of them.

A common used technique to provide regions with semantic meaning is the manual annotation that, combined with a powerful segmentation method, can result in a good meaningful classification. Comaniciu [20] proposes a color image segmentation algorithm based on the mean shift that estimates density gradients, using a simple nonparametric procedure. Then, the users interactively identify the segmented regions by labeling the features. Because of hand-annotating images is tedious and human expensive, methods for learning image representations directly from data are investigated.

Fei-Fei and Perona [33] propose a Bayesian hierarchical model to learn and recognize natural scene categories through intermediate "themes". In there, the most complete scene category dataset found in the literature is used. An image is modeled as a collection of local patches (regions). Each patch is represented by a codeword from a large vocabulary of them obtained from all categories training examples. For each codewords in each category, a Bayesian hierarchical model is learnt, building a collection of Bayesian models. Then, to provide semantic meaning to an unknown image, first the image codewords are extracted, and then, they are compared with the predefined models, assigning the one which fits best. The main problem of the proposed algorithm is that, although it can learn intermediate themes of scenes with neither supervision nor human intervention, the categories are fixed, being not able to assign semantic meaning to other ones.

Another method that uses predefined lexicon of semantic concepts as trained data, is the semantic pathfinder for multimedia indexing [15]. In here, given a pattern x, part of a camera shot, the aim is to detect a semantic concept ω from shot i using probability $p(\omega|x_i)$. Each step in the semantic pathfinder analysis extracts x_i from data, and learns $p(\omega|x_i)$ for all ω in the semantic lexicon.

Hudelot et al. [16] propose a learning approach based on two steps: a feature selection step that chooses the most characterizing features for better visual concept detection, and a training phase using a Support Vector Machine (SVM), where positive and negative samples are required. Trying to solve the weaknesses of the learning approach like the lack of learning the spatial structure of semantic concepts, a further step is given, storing the visual knowledge that is the link between semantic concepts and sensor data in a symbol. This link is modeled as a fuzzy linguistic variable that enables the representation of imprecision, thus the image features are fuzzified a priori by a human expert, providing spatial relation representations and spatial reasoning.

In these articles, we find two facts that we try to avoid. On one hand, the lack of generalization by using a predefined lexicon when trying to link data with semantic classes. The use of a semantic lexicon is useful when we arrange an a priori and limited knowledge. On the other hand, the need of experts in the application domain to manually label the regions of interest.

An important issue to arrange while assigning semantic meaning to a combination of classes is the data fusion. Li and Bretschneider [56] propose a method where combination of feature vectors for the interactive learning phase is carried out. They propose an intermediate step between region pairs (clusters from k-means algorithm) and semantic concepts, called *code pairs*. To classify the low-level feature vectors into a set of codes that form a codebook, the Generalised Lloyd Algorithm is used. Each image is encoded by an individual subset of these codes, based on the low-level features of its regions.

Signal classes are objective and depend on feature data and not on semantics. Sheikholeslami et al. [36] propose a semantic clustering. This is a parallel solution considering semantics in the clustering phase. In the article, a first level of semantics dividing an image in semantic high category clusters, as for instance, grass, water and agriculture is provided. Then, each cluster is divided in feature subclusters as texture, color or shape. Finally, for each subcluster, a semantic meaning is assigned.

In terms of classification of multiple features in an interactive way, there exist few methods in the literature. Sheikholeslami et al. [36] describe the design of a multilayer neural network model to merge the results of basic queries on individual features. The input to the neural network is the set of similarity measurements for different feature classes and the output is the overall similarity of the image. To train the neural network and find the weights, a set of similar images for the positive examples and a set of non similar ones for the negative examples must be provided. Once the network is trained, it can be used to merge heterogeneous features.

To finish this review in semantic learning, we have to mention the kind of semantic knowledge we can extract from EO data. The semantic knowledge depends on image scale, and the observability of scale is limited by sensor resolution. It is important to understand the difference between scale and resolution. The term of sensor resolution is a property of the sensor, while the scale is a property of an object in the image. Fig. 2.2 depicts the correspondence between knowledge that can be extracted for an specific image scale, corresponding small objects with a scale of 10 meters and big ones with a scale of thousands of meters. The hierarchical representation of extracted knowledge enables answering questions like which sensor is more accurate to a particular domain or which are the features that better explain the data.

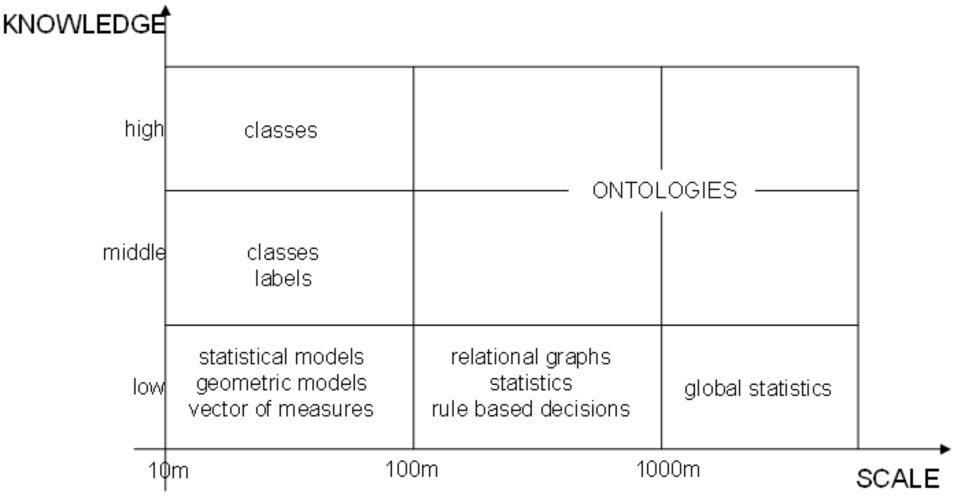


Figure 2.2: Knowledge level in the hierarchy to be extracted depending on the image scale.

2.2.1 Relevance Feedback

Often an IIM system requires a communication between human and machine while performing interactive learning for CBIR. In the interaction loop, the user provides training examples showing his interest, and the system answers by highlighting some regions on retrieved data, with a collection of images that fits the query or with statistical similarity measures. These responses are labeled as *relevance feedback*, whose aim is to adapt the search to the user interest and to optimize the search criterion for a faster retrieval.

Li et al. [55] propose a composite relevance feedback approach which is computationally optimized. At a first step, a pseudo query image is formed combining all regions of the initial query with the positive examples provided by the user. In order to reduce the number of regions without loosing precision, a semantic score function is computed. On the other hand, to measure image-to-image similarities, they perform an integrated region matching.

In order to reduce the response time while searching in large image collections, Cox et al. [22] developed a system, called *PicHunter*, based on a Bayesian relevance feedback algorithm. This method models the user reaction to a certain target image and infers the probability of the target image on the basis of the history of performed actions. Thus, the average number of man-machine interactions to locate the target image is reduced, speeding up the search.

2.3 Existing Image Information Mining Systems

As IIM field is nowadays in its infancy, there are only a few systems that provide CBIR being under evaluation and further development. Aksoy [10] provides a survey of CBIR systems prior to 2001, and a more recent review is provided by Daschiel [23]. In this section, we present several IIM systems for retrieval of remote sensed images, most of them being experimental ones.

Li [54] proposes a system, able to retrieve integrated spectral and spatial information from remote sensing imagery. Spatial features are obtained by extracting textural characteristics using Gabor wavelet coefficients, and spectral information by Support Vector Machines (SVM) classification. Then, the feature space is clustered through an optimized version of k-means approach. The resulting classification is maintained in a two schemes database: an image database where images are stored, and an Object-Oriented Database (OODB) where feature vectors and the pointers to the corresponding images are stored. The main advantage of an OODB is the mapping facility between an object oriented programming language as Java or C++, and the OODB structures through supported Application Programming Interfaces (API). The system has the ability of processing a new image in online mode, in such a way that an image which is not still in the archive is processed and clustered in an interactive form.

Feature extraction is an important part of IIM systems, however, it is computationally expensive, and usually generates a high volume of data. A possible solution would be to compute only those relevant features for describing a particular concept, but how to discriminate between relevant and irrelevant features? The *Rapid Image Information Mining (RIIM)* prototype [73] is a Java based framework that provides an interface for exploration of remotely sensed imagery based on its content. Particularly, it puts a focus on the management of coastal disaster. Its ingestion chain begins with the generation of tiles and an unsupervised segmentation algorithm. Once tiles are segmented, a feature extraction composed of two parts is performed: a first module consists of a genetic algorithm for the selection of a particular set of features that better identifies a specific semantic class. A second module generates feature models through genetic algorithms. Thus, if the user provides a query with a semantic class of interest, feature extraction will be only performed over the optimal features for the prediction, speeding up the ingestion of new images. The last step consists of applying a SVM approach for classification. While executing a semantic query, the system computes automatically the confidence value of a selected region and facilitates the retrieval of regions whose confidence is above a particular threshold.

The *IKONA system* [70] is a CBIR system based on a client-server architecture. The system provides the ability of retrieving images by visual similarity in response to a query that satisfies the interest of the user. The system offers the possibility to perform region based queries in such a way that the search engine will look for images containing similar parts to the provided one. A main characteristic of the prototype is the hybrid text-image retrieval mode. Images can be manually annotated with indexed keywords, and while retrieving similar content images, the engine searches by keyword providing a faster computation. IKONA can be applied not only for EO applications, but also for face detection or signature recognition. The server-side architecture is implemented in C++ and the client software in Java, making it independent from the platform where it runs. The only prerequisite on the client is to have installed a Java Virtual Machine.

The *Query by Image Content (QBIC) system* [60] is a commercial tool developed by IBM that explores content-based retrieval methods allowing queries on large image and video databases. These queries can be based on selected color and texture patterns, on example images or on user-made drawings. QBIC is composed of two main components: database population and database query. The former deals with processes related to image processing and image-video database creation. The latter is responsible for offering an interface to compose a graphical query and for matching input query to database. Before storing images in the archive, they are tiled and annotated with text information. The manual identification of objects inside images can become a very tedious task, and trying to automatize this function, a full automatic unsupervised segmentation technique based on foreground/background models is introduced. Another method to automatically identify objects, also included in this system, is the flood-fill approach. This algorithm starts from a single pixel and continues adding neighbor pixels, whose values are under a certain threshold. This threshold is calculated automatically and updated dynamically by distinguishing between background an object.

Photobook [8] developed by MIT, is another content-based image and image sequences retrieval, whose principle is to compress images for a quick query-time performance, preserving essential image similarities. Reaching this aim, the interactive search will be efficient. Thus, for characterization of object classes preserving its geometrical properties, an approach derived from the Karhunen-Loève transform is applied. However, for texture features a method based on the Wold decomposition that separates structured and random texture components is used. In order to link data to classes, a method based on color difference provides an efficient way to discriminate between foreground objects and image background. After that, shape, appearance, motion and texture of theses foreground objects can be analyzed and ingested in the database together with a description. To assign a semantic label or multiple ones to regions, several human-machine interactions are performed, and through a relevance feedback, the system learns the relations between image regions and semantic content.

VisiMine system [48][49] is an interactive mining system for analysis of remotely sensed data. VisiMine is able to distinguish between pixel, region and tile levels of features, providing several feature extraction algorithms for each level. Pixel level features describe spectral and textural information; regions are characterized by their boundary, shape and size; tile or scene level features describe the spectrum and textural information of the whole image scene. The applied techniques for extracting texture features are Gabor wavelets and Haralick's co-ocurrence, image moments are computed for geometrical properties extraction, and k-medoid and k-means methods are considered for clustering features. Both methods perform a partition of the set of objects into clusters, but with k-means, further detailed in chapter 6, each object belongs to the cluster with nearest mean, being the centroid of the cluster the mean of the objects belonging to it. However, with k-medoid the center of the cluster, called medoid, is the object, whose average distance to all the objects in the cluster is minimal. Thus, the center of each cluster in k-medoid method is a member of the data set, whereas the centroid of each cluster in k-means method could not belong to the set. Besides the clustering algorithms, general statistics measures as histograms, maximum, minimum, mean and standard deviation of pixel characteristics for regions and tiles are computed. In the training phase, naive Bayesian classifiers and decision trees are used. An important factor of VisiMine system is its connectivity to S-PLUS, an interactive environment for graphics, data analysis, statistics and mathematical computing that contains over 3000 statistical functions for scientific data analysis. The functionality of VisiMine includes also generic image processing tools, such as histogram equalization, spectral balancing, false colors, masking or multiband spectral mixing, and data mining tools, such as data clustering, classification models or prediction of land cover types.

GeoIRIS [17] is another IIM system that includes automatic feature extraction at tile level, such as spectral, textural and shape characteristics, and object level as high-dimensional database indexing and visual content mining. It offers the possibility to query the archive by image example, object, relationship between objects and semantics. The key point of the system is the ability to merge information from heterogeneous sources creating maps and imagery dynamically.

Finally, *KIM* [29][57] and later versions of *KES* and *KEO* [80][81][34][65] are perhaps the most enhanced systems in terms of technology, modularity and scalability. They are based on IIM concepts where several primitive and non-primitive feature extraction methods are implemented. In the last version, new feature extraction algorithms can easily plugged in, being incorporated to the data ingestion chain. In the clustering phase, a variant of k-means technique is executed generating a vocabulary of indexed classes. To solve the semantic gap problem, KIM computes a stochastic link through Bayesian networks, learning the posterior probabilities among classes and user defined semantic labels. Finally, thematic maps are automatically generated according with predefined cover types. Currently, a first version of KEO is available being posterior versions under further development. A full description of these systems is presented in chapter 3. They represent the basics of this dissertation being the main contributions integrated in KEO system.

3

The Knowledge-driven Information Mining System: Concept and Overview

In 1992, two years after the CBIR field was born, the idea arose to build a system, able to be queried by image content from a database of remote sensed images. Thus, the theoretical concept was studied by IMF-DLR (German Aerospace Center) and ETHZ (Swiss Federal Institute of Technology, Zurich). In 2002, the first prototype of Knowledge-driven Information Mining System (KIM) was developed. This prototype is based on the hierarchical Bayesian modeling of information representation.

However, with the evolution of satellite sensors, an increasing and heterogeneous volume of data are received per day. Then, a system able to process data from different sensors for a better image interpretation was needed. Thus, in 2004, a Knowledge Enabled Services (KES) system, based on KIM Bayesian information representation was built. This novel IIM system fulfils the requirement of a user knowledge centered environment helping to the understanding and interpretation of heterogeneous data.

Meanwhile, the Internet is widely extended making possible the communication with the other side of the world sharing all kind of information through the network. Technology evolves towards pieces of software residing on servers being accessible through standard-based interfaces. With this premises, the KES system was provided with new features arising Knowledge-centered Earth Observation (KEO) in 2008. The novel architecture of KEO based on the newest technology allows users to share algorithms, image data, processed data, etc. Fig. 3.1 shows the evolution of the system through the years, from the beginning of KIM concept to the newest version of KEO.

This chapter begins presenting the hierarchical Bayesian information representation on which this dissertation is based. Then, Sec. 3.2 describes the evolution of KIM towards a more user oriented and better performance system called KES. Finally, Sec. 3.3 introduces a KEO system, whose design and architecture will be detailed further in chapter 9.

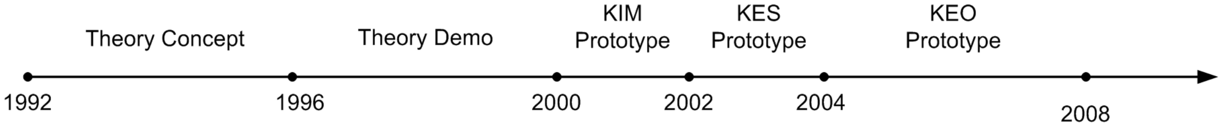


Figure 3.1: Evolution of the Image Information Mining concept: KIM - KES - KEO systems.

3.1 Knowledge-driven Information Mining

A Knowledge-driven Information Mining System (KIM) [59] builds a theoretical framework of collaborative methods for the extraction and exploration of the image content in large EO data repositories, for establishing the link between user knowledge and information content of images, and for communicating at high semantic abstraction between users coming from different disciplines and heterogeneous sources of information. It provides solutions how to access large image datasets through IIM and CBIR. User-defined semantic image content interpretation is linked through Bayesian networks to a completely unsupervised content-index. Based on this stochastic link, the user can query the archive for relevant images obtaining a probabilistic classification of the entire image archive as an intuitive information representation. This concept was first developed by Schröder [62] in the CBIR domain, and extended by Daschiel[24], who defined a higher level of semantic concepts.

From the computation and system architecture point of view, the hierarchy of information is separated in two components: an expensive, off-line and unsupervised computational module which is responsible for extracting information from data, grouping features in classes and indexing classes in the database, and a supervised, online and fast computational component to define user provided semantic labels. In other words, KIM is partitioned in an unsupervised grouping section and a supervised learning part. Thus, the flow of information in the stochastic inference from original image to semantic labels decreases the amount of information lightening the computation factor.

The information representation hierarchy of KIM is based on a five-level Bayesian learning model (see Fig. 3.2), allowing going forwards and backwards in the hierarchy applying Bayes$'$ rule.

From Level 0 to Level 1: Primitive feature extraction

The Bayesian learning model represents original data as level 0 in the information hierarchy. Starting at level 0, spatial and spectral features θ are extracted using stochastic models M. These extracted image features characterize a particular aspect of the data depending on the model as texture parameters or spectral signature, and form the level 1 of Bayesian hierarchy. Models M are given in a parametric form by the likelihood $p(D|\theta, M)$, which is assigned to a realization of the data D for a particular parameter

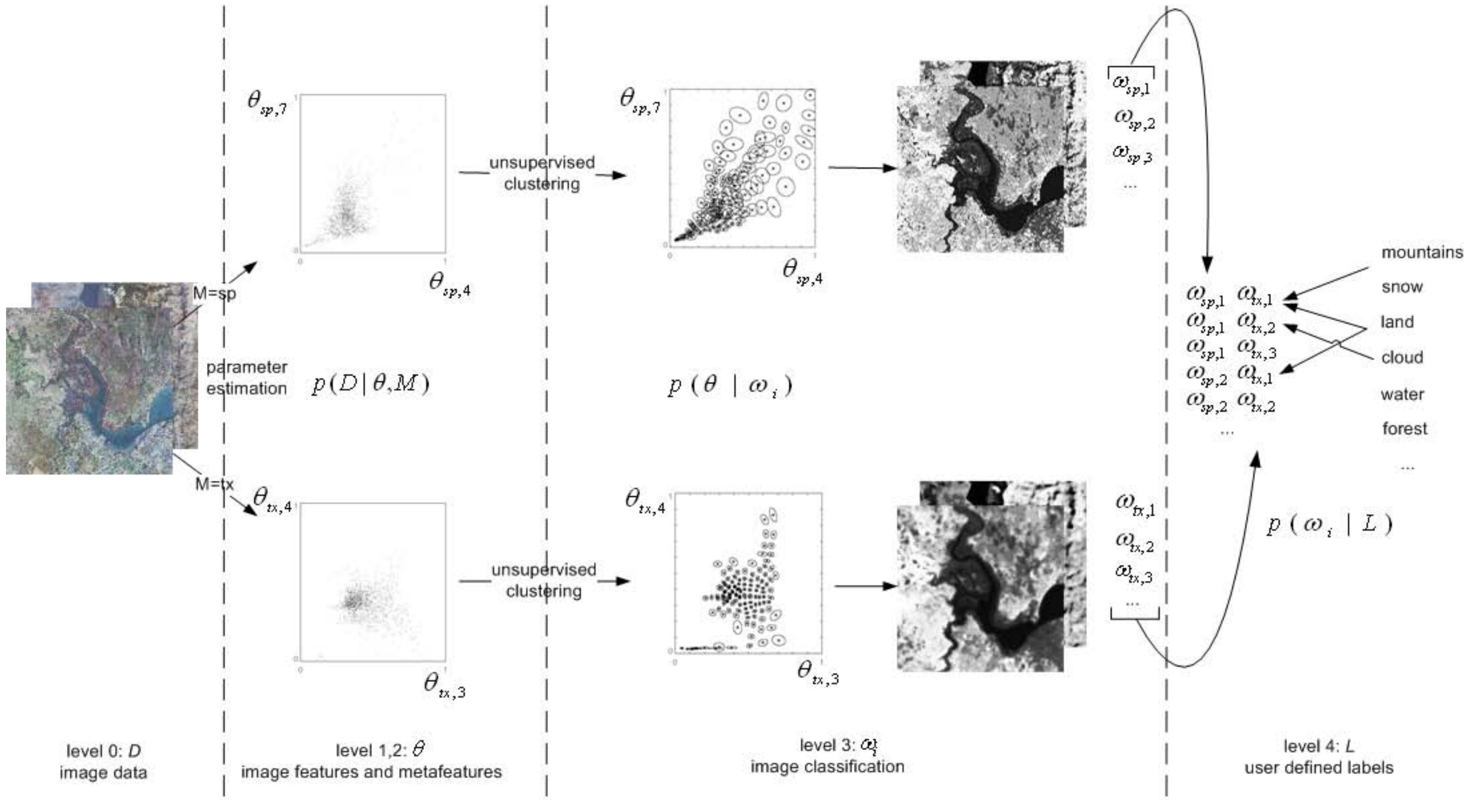


Figure 3.2: Hierarchical modeling of image content and user semantic. First, primitive image features (level 1) and metafeatures (level 2) are extracted from image data (level 0) based on different parametric signal models. Through an unsupervised clustering of the features, we obtain a vocabulary of signal classes (level 3). Finally, semantic cover types (level 4) are defined by the user. Semantic labels L are linked to the signal classes ω_i through the probabilities $p(\omega_i|L)$ using Bayesian networks.

vector θ.

From Level 1 to Level 2: Metafeatures

The level 2 in the information hierarchy corresponds to metafeatures extracted from image data D (level 0) using different signal models. Some examples of metafeatures are the scale of images or the most evident model found in data by performing the Bayesian technique.

From Level 1/2 to Level 3: Clustering

Extracted features (level 1) and metafeatures (level 2) together with original data produce a large volume of data. In order to reduce the dimension of feature space trying to keep the properties of data, compression algorithms are needed. Thus, performing a completely unsupervised clustering algorithm to the extracted image parameters θ, a vocabulary of signal classes ω_i is obtained. Generated classes, also called clusters, describe characteristic groups of points separately for each model and are indexed in a relational database. This task is implemented through a variant approach of the k-means method called dyadic k-means further detailed in Sec. 6.1.

From Level 3 to Level 4: Interactive learning

From level 1 to level 3 computation is performed in a completely unsupervised and application-free way. Data are processed and ingested in the database with no interaction with the user. Providing semantic meaning to the indexed classes, the specific user interest is defined building level 4 in the information hierarchy.

The user gives positive and negative examples for a supervised learning of features through a simple graphical interface. This knowledge is then stored in form of a semantic label, L, that is linked to combinations of vocabularies of signal classes, ω_i, by probabilities $p(\omega_i|L)$ through Bayesian networks. Thus, information at level 4 is interactively defined by users with a learning paradigm that links (objective) signal classes and user (subjective) labels, solving the semantic gap problem.

From the technology point of view, KIM is based on a server-client architecture to be operated over internet. At server side, primitive feature extraction and clustering algorithms developed in C++ programming language are computed. At client side, the interface learning module allows the user to define and refine a semantic label though a Java applet. The probabilistic link between semantic concepts and classes is computed and temporary saved in the client cache in order to speed up the response. Finally, the search results are presented to the user in HTML format.

3.2 Knowledge Enabled Services

KIM offers the possibility of ingesting EO data in a relational database for a later retrieval of images by their content. However, the new generations of high resolution EO sensors and the increasing volume of available data encourage the upgrading of the system. Knowledge Enabled Services (KES) is a distributed information system, designed to provide a framework for exploring EO data based on user provided knowledge to be run over Internet. It keeps the Bayesian hierarchy of the information from KIM adding a new level of information (level 5) that defines complex semantics by aggregating labels to higher-level semantic concepts. In a similar way as defining a label through positive and negative examples, an aggregated label is defined by training samples. Thus, the definition of ”water” is performed by giving as positive examples ”sea”, ”river” and ”lake” and as negative examples ”streets”, ”mountains” and ”houses”.

As KES is a KIM-based system, it offers the ingestion chain module of KIM, and additionally, it provides following new capabilities:

- Automatic upload of data to be ingested: data can arrive in the system manually or via FTP at a data acquisition module. Thus, it is possible to connect this module to a Ground Segment feeding the information system in a periodic and automatic way with no human interaction.
- The acquired knowledge is stored in two new catalogues: the Probabilistic Information Mining containing the *a-posteriori* maps built by the user through training examples, and the Thematic Map catalogue that stores generated thematic maps containing several semantic labels.
- Knowledge transfer: knowing the user domain of interest and observing its interactions with the system, it is possible to infer data likely of user interest, and link this data with the domain. With this potential, the user can navigate through a taxonomically structured knowledge. Going further, measuring the positive feedback of the system for weighting interest, the degree of correlation between concepts and domains can be inferred, enabling the prediction of interesting data for a certain user.
- Multiple classifier module: this novel multi-class function enables through a friendly graphical interface to build interactive classification maps by selecting defined semantic cover-types.
- Available accessory data as Geographical Information System (GIS) layers or associated texts can be attached to extracted knowledge.

As described in chapter 1, this dissertation can be seen from two perspectives, one based on signal processing aspects, and a second one related with system architectures.

KIM and already presented KES concepts, concentrate on the first perspective features. Now, we present the system perspective, which includes the design of an optimal software architecture for building an operational IIM system. From this point of view, KES can be decomposed in three separated subsystems: the server, where ingestion chain modules are computed, the client, that provides a graphical user interface to perform the interactive learning tasks, and a database where original data together with extracted features and classification maps are stored. Going deeper in technological aspects, we look at:

- Client: through Java Web Start (JWS) technology, the client is automatically downloaded and installed in the computer of the user. Files are cached at client side in order to have them available to be launched at any time. The client provides the end user a rich graphical interface. The communication with server side will be carried out over internet using the HTTP protocol.

- Server: the server can be separated in two tiers: at a first tier we find the ingestion chain modules, formed by a collection of executable programs that allow the extraction of primitive features and their classification. These components are developed in C++ and run on a Linux cluster in order to process the huge data volume at a reasonable time. At a second tier, we find those modules that communicate with the client, which are based on Java 2 Enterprise Edition (J2EE) technology according with standard design patterns. The J2EE defines the standard for developing multitier enterprise applications. The J2EE platform simplifies enterprise applications by basing them on standardized, modular components, by providing a complete set of services to those components, and by handling many details of application behavior automatically, without complex programming. At this level, we distinguish between modules responsible for the communication in terms of proxies, security and formatting response which are developed through servlets, and modules in charge of processing client requests that are implemented through Web Services and Enterprise Java Beans (EJB).

- Database Management System (DBMS): The database is managed by an IBM/Informix Dynamic Server with a Geodetic Datablade module in charge of performing geotemporal queries. The communication between the server and the database is provided through different connectors depending on the server component to be accessed: an ODBC connector for the ingestion chain modules and JDBC one for J2EE applications.

KES also provides an administration tool that consists of a dedicated console to manage all the system functionalities being not downloadable from the Internet. It interacts with the server through the RMI protocol, and access to the database through JDBC connector.

3.3 Knowledge-centered Earth Observation

Continuing in the system perspective of the dissertation, the Knowledge-centered Earth Observation (KEO) system was born to adapt the previous versions of KIM and KES systems in order to be remotely accessible, modular, standardized and easily expandable. With these requirements a new architecture has to be designed, considering existing technology and user scenarios. Thus, KEO extends the capabilities of KIM and KES by including following features:

- Ability of accepting EO data coming from Ground Segments and other data providers: KIM and KES systems are fed manually, by writing incoming data to an specific location of the local disk or network. KEO can be fed remotely, with the only requirement of having Internet connection.
- Output of the system in easily accessible format: KEO generates Geographical Information System (GIS) data, as maps compliant with Web Map Service (WMS) from the Open Geospatial Consortium.
- Leverage on open standards: In order to be accessible and understandable by other systems, KEO provides standard data formats of the generated outputs. For accessing information, KEO accepts and understand also standard services.
- Modular and distributed architecture based on Service-Oriented Architecture (SOA): this software architecture allows the localization and provision of web services via Internet in a scalable environment.

From the software architecture point of view, KEO is fully developed in Java, using J2EE components for the server components and JWS for the client. Each server component will be implemented as a web service, being accessible through the network. These web services will take part of different workflows for processes automatization. The workflow will be orchestrated by a process manager, which orders and monitors the correct behavior of the processes. Finally, KEO catalogues containing original images, extracted features and generated thematic maps are also wrapped in a web service and will be available in the Service Support Environment (SSE) portal, provided by the European Space Agency (ESA) for accessing image repositories. A full description of KEO software architecture is presented in chapter 9, being the design part of this dissertation.

4

Basics of Inference and Stochastic Image Analysis

The KIM system, presented in the previous chapter consists of different levels of image content abstraction, modeled through a hierarchical Bayesian network. These levels representing image features, metafeatures, clusters or semantic labels are considered as realizations of stochastic models. In order to achieve robust results or to evaluate some steps of the hierarchy, concepts of information theory are applied. This chapter concentrates on the theoretical aspects of this dissertation building the basis of the contributions.

This chapter begins with a brief introduction of general concepts in inferential statistics, as probabilities or statistical moments. Sec. 4.2 focuses on the principles of Bayesian inference, and Sec. 4.3 presents several concepts related with information theory as entropy, Fisher information or Rate Distortion Theory.

4.1 Stochastic Image Analysis

In statistics, there are two main lines of research: descriptive and inferential statistics. The first one analyzes, summaries and interprets the sample without inferring any property of the population from which the sample was extracted. In contrast, inferential statistics introduces the concepts of probability and hypothesis and deduces (infers) properties of the population from the analysis of the sample. In addition, the inferential statistics is divided in two groups: the *frequentist* approach, that assumes constant values of unknown parameters of the population, and the *bayesian* approach, which supposes that the unknown parameters are continually revised in light of new data by using a weight assigned to a previous assumption. Thus, they are random variables with an associated probability distribution called *prior*.

In this section, general concepts of inferential statistics as probability, random variables and stochastic processes are presented.

4.1.1 Probability

According to inferential statistics approaches, two definitions of probability can be found:

Definition 1. *The probability of one event is the ratio of the number of cases favorable to it, to the number of all cases possible when nothing leads us to expect that every one of these cases should occur more than any other, which renders them, for us, equally possible.* Frequentist approach definition (Laplace 1812).

Definition 2. *The probability is a representation of degrees of plausibility by real numbers.* Bayesian approach definition (Jeffrey, today's understanding).

The probability of an event E, $Pr(E)$, must satisfy the following axioms:

1. Positivity

$$Pr(E) \geq 0 \tag{4.1}$$

2. Certain event

$$Pr(I) = 1 \tag{4.2}$$

3. Sum

$$Pr(E_1 \cup E_2) = Pr(E_1) + Pr(E_2) \qquad if E_1 \cap E_2 = \emptyset \tag{4.3}$$

4. Product

$$Pr(E_1 \cap E_2) = Pr(E_2|E_1)Pr(E_1) = Pr(E_1|E_2)Pr(E_2) \tag{4.4}$$

We say that two events are statistically independent if

$$Pr(E_1 \cap E_2) = Pr(E_1)Pr(E_2) \tag{4.5}$$

then $Pr(E_2|E_1) = Pr(E_2)$ and $Pr(E_1|E_2) = Pr(E_1)$

4.1.2 Random Variable

A random variable (r.v.) is a function from S, the sample space, to R, the real line [19]. It is characterized by the probability distribution of the elements in the sample space.

Distribution function : the distribution function $F(x)$ of a continuous r.v. X is the probability that X value is equal or lower than x: $F(x) = Pr(X \leq x)$

Probability density function (p.d.f.) : the probability distribution of a continuous r.v. X is characterized by the p.d.f., $p(x)$, and determines the probability in a given interval. It has the following properties:

1. $p(x) \geq 0, -\infty < x < \infty$
2. $\int_{-\infty}^{\infty} p(x)dx = 1$
3. $Pr(a \leq X \leq b) = \int_a^b p(x)dx$

The relation between the distribution function $F(x)$ and the p.d.f. $p(x)$ is $F(x) = \int_{-\infty}^{x} p(v)dv$

Other interesting p.d.f. that will be applied in the following chapters are:

- Joint p.d.f., $p(x, y)$: the joint p.d.f., also called multivariate distribution, is often summarized by a vector of parameters, which may be sufficient to characterize the distribution completely. It is defined as $p(x, y) = p(x|y)p(y) = p(y|x)p(x)$
- Marginal p.d.f., $p(x)$: given a joint p.d.f. of two r.v. x and y, the marginal p.d.f. is determined by the probability distribution of x. It is computed by integrating the joint distribution of x and y: $p(x) = \int_y p(x, y)dy$
- Conditional p.d.f., $p(x|y)$: a joint p.d.f. is conditional relative to y, if y has a predefined value. It is computed by $p(x|y) = Pr(X \leq x|Y = y)$

Statistical Moments

Moments are applicable to many image processing aspects, describing image content (or distribution) with respect to its axes. They capture both global and detailed geometric information, extract properties with analogous statistics or characterize a grey level image.

The moments of a random variable X are the expected values of certain functions of it. From all moments, only three are mentioned here. The first moment, called *mean*, is defined by

$$\mu = E(X) = \int_{-\infty}^{\infty} xp(x)dx \tag{4.6}$$

The second one, called *variance*, is defined by

$$\sigma^2 = V(X) = \int_{-\infty}^{\infty} x^2 p(x)dx \tag{4.7}$$

The third moment, called *covariance*, measures the strength of the correlation between two random variables. The covariance between two random variables X and Y is defined as

$$cov(X, Y) = C_{XY} = E\{[X - E(X)][Y - E(Y)]\} \tag{4.8}$$

When a feature increases and the other one also increases, the covariance value is positive. Contrary, if the covariance value is negative, when a feature increases, the other one decreases. If there is no correlation between the two features, the covariance becomes zero, indicating that the two features are independent of each other.

Karhunen-Loeve Transform or Principal Component Analysis

Principal Component Analysis (PCA) is a classical statistical method, widely used in data analysis and compression. It is a technique to reduce multidimensional data to lower dimensions for an easily analysis of the image. Computation of PCA requires the calculation of the eigenvectors and eigenvalues of the covariance matrix.

Generally, a vector v, called *eigenvector*, is scaled when it is multiplied by a matrix A, and the scale value λ is called *eigenvalue*. For the PCA computation case, a NxN image is represented as a one-dimensional vector of $N \cdot N$ elements, considering each row of the image one after another. Then, we compute the covariance matrix of the data and we extract the eigenvalues from this covariance matrix. Thus, the eigenvectors corresponding to the most significant eigenvalues will be the principal components.

4.1.3 Stochastic Processes

A stochastic process is a process that can be described by the evolution of a r.v. over some parameter. Examples of stochastic processes are *time series* parameterized by time and *random fields* in the spatial domain. Stochastic image analysis considers images as particular realizations of a spatial stochastic process and aims at the local and neighborhood analysis of the statistical dependencies of the pixel intensities.

A stochastic process is determined by the knowledge of all p.d.f.'s

$$\lim_{n \to \infty} p(x(t_1), ..., x(t_n)) \tag{4.9}$$

where $x(t_i)$ is the characterization of a r.v. at time t_i

Stationary Process

A stochastic process is said to be *strict-sense stationary*, if its statistical properties are invariant respect to a translation of the origin.

A stochastic process is said to be *wide-sense stationary*, if its mean is constant and its autocorrelation depends only on the time shifting $t_1 - t_2$.

$$\begin{aligned} E[x(t)] &= constant \\ C_{XX}(t_i, t_{i+1}) &= C_{XX}(t_i - t_{i+1}) \end{aligned} \tag{4.10}$$

Ergodic Process

A stochastic process is called ergodic if all its statistical properties can be determined from a single realization. In this case, temporal averages equal the stochastic expectations.

4.1.3.1 Markov Random Fields

An image as a realization of a 2-dimensional stochastic process defined on a regular rectangular lattice $\Omega = \{(i,j)|1 \leq i,j \leq N\}$ is also called *random field*. An internal site (i,j) has four nearest neighbors as $\mathcal{N}_{i,j} = \{(i-1,j),(i+1,j),(i,j-1),(i,j+1)\}$.

A Markov Random Field (MRF) is defined on a lattice Ω with respect to a neighborhood system $\mathcal{N}$ if and only if:

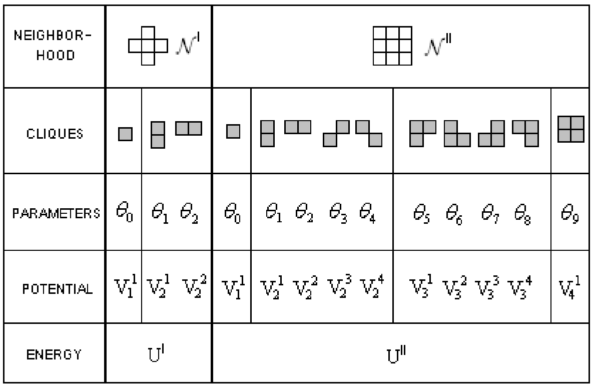
$$p(x_s|x_r; r \neq s) = p(x_s|x_r; r \in \mathcal{N}_s) \tag{4.11}$$

where r and s are two sites in Ω. Only neighboring pixels have a direct interaction. Markovianity is a *local* characterization of a random field.

4.1.3.2 Gibbs Random Fields

The definition of a Gibbs Random Field (GRF) has two aspects: topological and stochastic. Fig. 4.1 shows a graphical definition of the topological aspect, where a GRF is described by a system of cliques.

The clique system specifies the possible topology of the random structures in different realizations of the stochastic process: θ are the model parameters associated to each clique,



NEIGHBOR-HOOD	$\mathcal{N}^{I}$		$\mathcal{N}^{II}$			
CLIQUES						
PARAMETERS	θ_0	θ_1 θ_2	θ_0	θ_1 θ_2 θ_3 θ_4	θ_5 θ_6 θ_7 θ_8	θ_9
POTENTIAL	V_1^1	V_2^1 V_2^2	V_1^1	V_2^1 V_2^2 V_2^3 V_2^4	V_3^1 V_3^2 V_3^3 V_3^4	V_4^1
ENERGY	U^{I}		U^{II}			

Figure 4.1: $\theta_0, ..., \theta_9$ are the model parameters associated to the corresponding cliques, V is the potential function and U is the energy function. $\mathcal{N}^i$ is the dimension of the neighborhood for a model order of i.

V is the potential function characterizing the interaction between random field samples inside the clique, and U is the energy function for the corresponding neighborhood.

A stochastic model of the data X using a set of parameters θ is equivalent to assign to each state of X a conditional probability $p(X|\theta)$. This probability depends on the particular clique and on the shape of cliques, and is given by the exponential distribution:

$$p(X|\theta) = \frac{1}{Z} e^{-U(X,\theta)} \tag{4.12}$$

where Z is a normalization factor of the distribution and $U(X, \theta)$ is defined as the sum of the potential functions $V_c(X, \theta)$ over the set of all cliques $\mathcal{C}$ as $U(X, \theta) = \sum_{c \in \mathcal{C}} V_c(X, \theta)$.

Different computations of potential function can be applied depending on the selected model [75]. Thus, we can choose between following alternatives:

1. Autobinomial Model

   If the image X is assumed to be a realization of a Markovian stochastic process, then the p.d.f. of a single pixel intensity x_s can be modeled as

   $$p(x_s|\mathcal{N}_s; \theta) = \frac{1}{Z_s} e^{-H(x_s|\mathcal{N}_s;\theta)} \tag{4.13}$$

   with a local energy function $H(x_s|\mathcal{N}_s; \theta)$ depending only on x_s and its neighbors $\mathcal{N}_s$

(see Fig. 4.2), and Z_s, being the normalization over all states of a single pixel x_s, e.g., for 8bit data over 256 terms. In this model the energy function is defined as

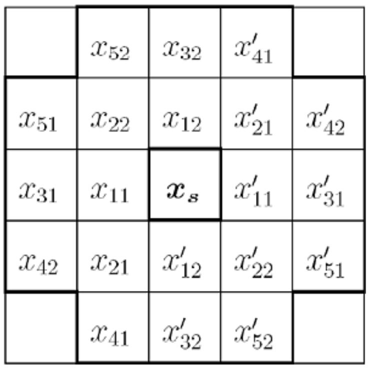
$$H(x_s|\mathcal{N}_s, \theta) = -log\binom{G}{x_s} - x_s \cdot \eta \tag{4.14}$$

where G is the maximum grey level value of a pixel, x_s is the central pixel, $\mathcal{N}_s$ the dimension of the neighborhood and $\eta = a + \sum_{ij} b_{ij} \frac{x_{ij}+x_{ij}^{'}}{G}$ the joint influence of the neighbors with the parameter vector

$$\theta = [a, b_{11}, b_{12}, b_{21}, b_{22}, ...]^T \tag{4.15}$$

The interaction of x_s with the neighborhood pair x_{ij} and $x_{ij}^{'}$ is weighted by the component b_{ij} in θ. a represents the probability distribution of x_s without spatial interaction. θ determines the p.d.f. $p(x_s|\mathcal{N}_s;\theta)$ for a given Gibbs local energy function $H(x_s|\mathcal{N}_s;\theta)$ over all cliques.

The model parameters shown in Fig. 4.1 correspond with the parameter vector θ with $\theta_0 = a, \theta_1 = b_{11}, \theta_2 = b_{12}, \theta_3 = b_{21}, ...$



	x_{52}	x_{32}	$x_{41}^{'}$	
x_{51}	x_{22}	x_{12}	$x_{21}^{'}$	$x_{42}^{'}$
x_{31}	x_{11}	$\boldsymbol{x_s}$	$x_{11}^{'}$	$x_{31}^{'}$
x_{42}	x_{21}	$x_{12}^{'}$	$x_{22}^{'}$	$x_{51}^{'}$
	x_{41}	$x_{32}^{'}$	$x_{52}^{'}$	

Figure 4.2: Example of neighborhood $\mathcal{N}_s$. The central pixel x_s is described by the pixels belonging to its neighborhood x_{ij} and $x_{ij}^{'}$.

2. Gauss Markov Random Field Model

   The Gauss Markov Random Field (GMRF) model is defined by the energy function

$$H(x_s|\mathcal{N}_s;\theta) = \frac{a^2}{2}(x_s - \eta)^2 \tag{4.16}$$

with $\eta = \sum_{ij} b_{ij}(x_{ij} + x_{ij}^{'})$ as interaction with the pairs of neighbors x_{ij} and $x_{ij}^{'}$.

3. Multi-Level Logistic Model

   For each clique c, the associated value of the potential function is determined by

$$V_c(x) = \begin{cases} -\theta_c & \text{if all pixels in the clique are equal} \\ \theta_c & \text{otherwise} \end{cases}$$

Finally, we have to mention the local view of GRF given by the equivalence between Gibbs random fields characterized by its global property (Gibbs distribution) and MRF characterized by its local property (the Markovianity). Any MRF can be modeled by a GRF with a large clique system, and any GRF can be considered as Markovian, if a large enough neighborhood system is applied.

4.2 Bayesian Inference

The Bayesian inference considers the unknown parameter θ as a representation of a random variable been described by a certain probability distribution called *a priori* distribution or *prior*, $p(\theta)$. The *prior* represents the probabilistic behavior of the parameter without any previous knowledge. Once a random sample x is provided, we are aware if the assumption of the *prior* was wrong and has to be modified or not. This modified probability distribution taking into consideration the sample knowledge is called *a posteriori* distribution and is represented by $p(\theta|x)$.

Applying the Bayes' theorem, the corresponding a posteriori probabilities can be determined for each value of θ as:

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)} \tag{4.17}$$

where $p(x|\theta)$ is the density function of the observed data x for a certain θ, $p(\theta)$ the prior, and $p(x)$ the marginal distribution of x defined in the parameter space Θ as $p(x) = \int_\Theta p(x,\theta)d\theta$.

In signal processing and CBIR systems, the Bayes' theorem can be applied to extract features from original data or to incorporate information as prior knowledge due to the stochastic nature of the signal models. As detailed in the previous chapter, the system can be explained at different levels of information: at a low level, primitive image features are extracted applying stochastic models; at a higher level, signal classes are described by their stochastic behavior; at the highest level, semantic image content is connected to a completely unsupervised content indexes through a stochastic link.

From this perspective of various levels of image information representation, the Bayesian hierarchical model is introduced. The basic idea in a hierarchical model is that, looking at the likelihood function $p(x|\theta)$, it may be appropriate to use priors that depend on other parameters not mentioned in the likelihood. These parameters themselves will require priors which depend on new parameters, and so on. The process finishes when no more new parameters are introduced. Formally, a non hierarchical Bayesian model is represented by

$$\{p(\theta), p(x|\theta)\} \tag{4.18}$$

with $p(\theta)$ the prior and $p(x|\theta)$ the likelihood, in the sense that knowing x, θ can be updated. Contrary, a hierarchical Bayesian model is described by

$$\{p(\theta), p(x|\xi_1), p(\xi_1|\xi_2), ..., p(\xi_k|\theta)\} \tag{4.19}$$

where knowing x, a value of ξ_1 can be learned; the knowledge of ξ_1 deduces the value of ξ_2, and so on till the knowledge of θ is reached. Fig. 4.3 shows an example of how Bayesian networks are applied at different levels of information representation.

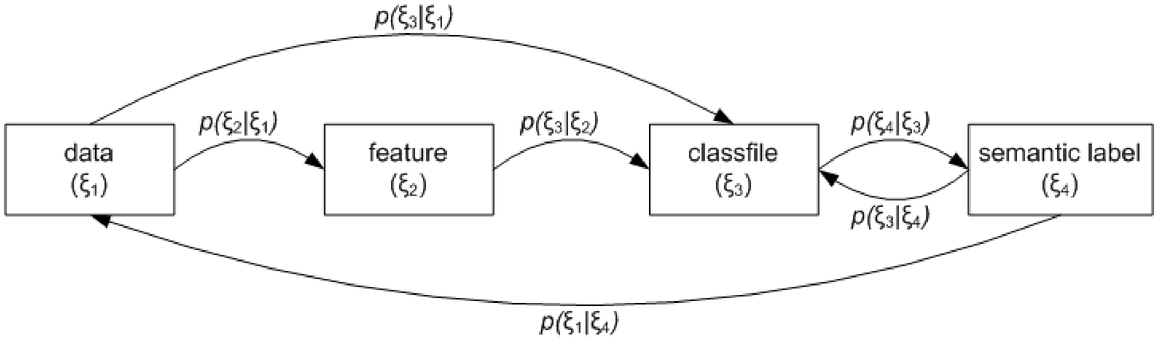


Figure 4.3: Hierarchical Bayesian Model. $p(\xi_2|\xi_1)$ represents the stochastic link between features and data; $p(\xi_3|\xi_2)$ learns the unsupervised signal classes knowing the features; $p(\xi_4|\xi_3)$ infers the semantic content from classfiles; $p(\xi_3|\xi_4)$ deduces the signal cluster of a known semantic label; $p(\xi_3|\xi_1)$ enables extracting signal classes from original dataset; $p(\xi_1|\xi_4)$ obtains the image data from its semantic content.

At this point, it is necessary to explain the image formation process, for a later description of how Bayesian inference can be applied on the image understanding problem.

4.2.1 Image Understanding

The scene is illuminated by the sun, and its reflected radiation x, called cross-section image, is propagated to the sensor. But in this propagation, physical effects are also taking part as humidity, roughness, etc. This scattering information is represented as a noisy signal n in the resulting image. Thus, the observed image y is model as

$$y = x_\theta + n \tag{4.20}$$

being x_θ the cross-section image depending on scene parameters θ. Image understanding task consists of reconstructing the cross-section image x and extracting the structural parameters θ from y.

4.2.1.1 Parameter Estimation

We consider an estimate $\hat{x}$ of the unknown grey-level values x, and the hypothetic error ϵ produced in the estimation process

$$\epsilon = x - \hat{x} \tag{4.21}$$

In order to estimate the error, a cost function $c(\epsilon)$ has to be measured. The most common ones are the quadratic computed as $c(\epsilon) = (x - \hat{x})^2$, and the uniform error function defined by

$$c(\epsilon) = \begin{cases} 0 & |\epsilon| \leq \frac{\delta}{2} \\ 1 & |\epsilon| > \frac{\delta}{2} \end{cases}$$

where δ is the interval where $x - \hat{x} = 0$. The expectation of the cost function R, called *Bayes risk*, is defined by

$$R = E(c(x - \hat{x})) = \int \int c(x - \hat{x}) p(x, y) dx dy \tag{4.22}$$

Therefore, applying Bayesian decision theory to the parameter estimation problem, means trying to minimize the Bayes risk. The Minimum Mean Square Error (MMSE) and Maximum A-Posteriori (MAP) Bayesian estimators aims at minimizing R by applying different cost error functions.

Minimum Mean Square Error estimator

The MMSE estimator uses the quadratic cost error function and is computed as

$$\hat{x}_{MMSE} = \int xp(x|y)dx \tag{4.23}$$

where $p(x|y)$ is the posterior probability approximated by $p(x|y) \propto p(y|x)p(x)$ and $p(x)$ the prior of x. The MMSE estimator is the conditional mean.

Maximum A-Posteriori estimator

The MAP estimator incorporates a prior distribution making inference possible using Bayes' rule. It asumes a uniform cost function and is estimated as

$$\hat{x}_{MAP} = \arg\max_{x} p(x|y) \tag{4.24}$$

Graphically, MMSE estimator is positioned in the center of mass of the posterior p.d.f., while MAP estimator represents the mode of the p.d.f. In Bayesian inference, the MAP estimator is often used due to the prior knowledge introduced in its definition.

4.2.1.2 Bayesian Two-Level Information Extraction

A second task of the image understanding problem lies in the estimation of the unknown parameters θ that characterize the structures in the image. Two levels of inference are introduced in order to build the model that better represents the data: a first level to fit a model to the data from a library of models, and a second one to select the model that better explains the data.

Level I: Model Fit

In order to be able to describe the image data y, a library of parametric models $\{M\}$ for a better understanding of the image content is used. A model $M_i \in \{M\}$ describes the data y through a vector of unknown parameters θ, and the task consists of fitting the model to the data. Thus, the probability of the model parameter θ given the data y and a certain model M_i is defined applying the Bayes' rule as

$$p(\theta|y, M_i) = \frac{p(y|\theta, M_i)p(\theta|M_i)}{p(y|M_i)} \tag{4.25}$$

with $p(y|\theta, M_i)$ the likelihood of the data knowing the model and the parameters, and $p(\theta|M_i)$ the prior of θ according to the model M_i. The term $p(y|M_i)$ is the evidence of the model M_i and will be studied in the second level of the Bayesian inference. After fitting the model to the data, we obtain an estimation $\hat{\theta}$ of the parameters θ.

Level II: Model Selection

From the library of parametric models, which is the one that better explains the data? The selection between them is not automatic, and usually we need more than one model to represent the data. This, together with the spatial diversity of the images, makes the task even more complicated.

A set of models $\{M\}$ representing the data y can be compared computing the posterior probability of each one using Bayes' rule as

$$p(M_i|y) = \frac{p(y|M_i)p(M_i)}{p(y)} \tag{4.26}$$

where $p(y|M_i)$ is the evidence defined as the probability of the data y given the model M_i, $p(M_i)$ denotes the prior of the model and $p(y)$ the marginal distribution of y. The evidence is obtained by marginalization

$$p(y|M_i) = \int p(y|\theta, M_i)p(\theta|M_i)d\theta \tag{4.27}$$

The ratio of the widths of the probability functions $p(y|\theta, M_i)$ and $p(\theta|M_i)$ is called *Occam factor* and will be the decision term for or against a model. Good fitting models will reflect a higher width of the likelihood (or higher Occam factor) than bad fitting models. So that, the best model will be the one that maximize the width of the likelihood.

4.3 Elements of Information Theory

In this section, several concepts of information theory are outlined. We begin defining entropy, mutual information and Fisher information as well as the equivalencies between them. In a second part, concepts of rate distortion theory are presented.

4.3.1 Shannon / Differential Entropy

Shannon entropy, also called information entropy, is a measure of the uncertainty associated with a random variable $X = x_1, ..., x_n$. Without any knowledge of the physical data source, it is not possible to know the entropy, which is estimated based on the outcome of the source by observing the structure of the data as source output. Hence, estimation of the entropy depends on the observation and assumptions about the structure of the source data sequence. These assumptions are called *model* of the sequence. Thus, entropy can be interpreted as the amount of information of X contained in the model $p(x)$.

The entropy of a discrete random variable X, called **Shannon entropy** is defined as

$$H(X) = -\sum p(x) \log p(x) \tag{4.28}$$

while the entropy of a continuous random variable X, called **differential entropy**, is defined by

$$H(X) = -\int_S p(x) \log p(x) dx \tag{4.29}$$

where $p(x)$ is the p.d.f. of X and S the support set of the random variable.

4.3.2 Kullback Leibler divergence or distance

The relative entropy or Kullback Leibler (KL) divergence or distance $D(p||q)$ between two p.d.f. $p(x)$ and $q(x)$ of a continuous random variable X is defined by

$$D(p||q) = \int p(x) \log \frac{p(x)}{q(x)} dx \tag{4.30}$$

The KL divergence measures the amount of information of X contained in the model $p(x)$ compared to a second alternative model $q(x)$. Thus, Shannon entropy can be interpreted as the amount of information in a model $p(x)$ of X compared to the maximum incertitude model, being the uniform probability distribution the one with maximum entropy.

4.3.3 Mutual Information

Information theory gives us a general definition of mutual information, described as the amount of information shared by two random variables, e.g. between input and output of noisy channels. But mutual information can be also applied to parameter estimation measuring how much information of the data Y is contained in the estimated parameter X. Formally, the mutual information of two continuous random variables X and Y is defined as

$$I(X;Y) = \int_Y \int_X p(x,y) \log \Big(\frac{p(x,y)}{p(x)p(y)}\Big) dxdy \tag{4.31}$$

where $p(x,y)$ is the joint probability of x and y, and $p(x)$ and $p(y)$ the marginal p.d.f. of x and y, respectively.

$I(X;Y)$ is a measure of dependence between X and Y, so that, if X and Y are independent, $p(x,y) = p(x)p(y)$ and therefore $\log(p(x,y)/p(x)p(y)) = \log 1 = 0$.

Mutual information can be equivalently expressed as

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ &= -\int_S p(x) \log p(x) dx - \int\int p(x|y) \log p(x|y) p(y) dx \end{aligned} \tag{4.32}$$

where $H(X)$ denotes the entropy of the estimator X, and $H(X|Y)$ the conditional entropy for continuous distributions computed as $H(X|Y) = \int\int p(x|y) \log p(x|y)p(y)dxdy$. In Bayesian terms, Y represents the prior information about X. Interpreting Eq. 4.32, mutual information reduces the uncertainty of the estimated parameter due to the knowledge of data.

4.3.4 Cram$\acute{e}$r-Rao Lower Bound and Fisher Information

Assuming an unbiased estimator, that is, the expectation value of the unknown parameters coincides with true parameters, the variance can be limited by the Cram$\acute{e}$r-Rao Lower Bound (CRLB) [58] computed as

$$\sigma_{\mathrm{x}}^2 \geq \frac{1}{E\Big[\frac{\partial}{\partial \mathrm{x}} \log \mathrm{p}(\mathrm{y}|\mathrm{x})\Big]^2} \tag{4.33}$$

where the conditional probability $p(\mathrm{y}|\mathrm{x})$ is the likelihood function.

The CRLB is the lower bound on the conditional variance and helps to take a decision for or against an estimator. However, often more than a single parameter is estimated and the CRLB has to be computed for each one. This situation leads to replace the variance for the covariance matrix, Σ_x, and to compute the Fisher Information, $J(x)$, instead the CRLB. $J(x)$ measures the minimum error in estimating a parameter of a distribution being defined only for a continuous random variable. Hence, Eq. 4.33 is transformed into

$$[\Sigma]_x \geq \frac{1}{[J(x)]} \tag{4.34}$$

being each element of $J(x)$ obtained by

$$J_{k,l}(x) = E\left[\Big(\frac{\partial}{\partial \mathrm{x_k}} \log \mathrm{p(y|x)}\Big)\Big(\frac{\partial}{\partial \mathrm{x_l}} \log \mathrm{p(y|x)}\Big)\right] \tag{4.35}$$

Therefore, Fisher Information gives a lower bound on the error in estimating X from the data.

Equivalent Measures

One equivalency between information measurements is denoted by mutual information and KL divergence. Observing Eq. 4.30 and Eq. 4.31, it can be confirmed that the mutual information and the KL divergence are equivalent by

$$I(X;Y) = D\big[p(x,y)||p(x)p(y)\big] \tag{4.36}$$

A second equivalency is found between Fisher information and the differential entropy applying the *de Bruijn's identity*

$$\lim_{t\to 0} \frac{\partial}{\partial t} H(x+\sqrt{t}z) = \frac{1}{2}J(x) \tag{4.37}$$

where x is a random variable with finite variance and density function $p(x)$, z an independent normally distributed random variable and H the differential entropy. The demonstration of Bruijn's identity is given in Appendix B.2.

According to the Bruijn's identity, when a given random variable is perturbed by adding gaussian noise, the change rate of Shannon's entropy is proportional to the Fisher information [86]. Thus, Shannon information defines a global measure meanwhile Fisher information describes a local one for coding the variable of interest.

4.3.5 Rate Distortion Theory

Information theory studies how to encode data in order to transmit it in a secure and robust way for a later reproduction of the data at receiver side. A source produces a sequence $X_1, X_2, \ldots, X_n = X^n$ of n independent identically distributed random variables with probability density function $p(x)$, x belonging to the source alphabet set $\mathscr{X}$. On one hand, the encoder describes the source sequence X^n by a function $f_n(X^n) \in \{1, 2, \ldots, 2^{nR}\}$, where R is the number of bits used to represent X and n is the length of the sequence. On the other hand, the decoder represents X^n by an estimator $\hat{X}^n \in \hat{\mathscr{X}}$ with $\hat{\mathscr{X}}$ the reproduction alphabet set. Usually $\hat{X}^n$ is called *vector quantization*, *representation*, *source code*, or *estimate* of X^n. Fig. 4.4 shows a typical architecture of an encoder-decoder system.

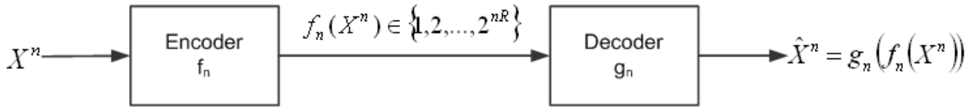

X^n
Encoder
f_n
$f_n(X^n) \in \{1,2,\ldots,2^{nR}\}$
Decoder
g_n
$\hat{X}^n = g_n(f_n(X^n))$

Figure 4.4: Encoder and decoder system.

The representation of a continuous random variable would need an infinite number of bits, and due to the use of a quantized vector to represent data, some information is missing introducing, consequently, an error [21]. Rate distortion theory tries to measure how much information is lost in the compressing process, in other words, the distance between the random variable X^n and its representation $\hat{X}^n$. Thus, the main goal is to compress data minimizing the average distortion.

The distance function $d = d(x, \hat{x})$ is a cost measure of representing the symbol x by the symbol $\hat{x}$, and is bounded if the maximum value of the function is finite:

$$d_{max} \overset{def}{=} \max_{x \in \mathscr{X}, \hat{x} \in \hat{\mathscr{X}}} d(x, \hat{x}) < \infty \tag{4.38}$$

Definition. The **information rate distortion function** $R^I(D)$ for a source X with distance measure $d(x, \hat{x})$ is defined as

$$R^I(D) = \min_{p(\hat{x}|x): \sum_{(x,\hat{x})} p(x)p(\hat{x}|x)d(x,\hat{x}) \leq D} I(X; \hat{X}) \tag{4.39}$$

where the minimization is over all conditional distributions $p(\hat{x}|x)$ for which the joint distribution $p(x, \hat{x}) = p(x)p(\hat{x}|x)$ satisfies the expected constraint:

$$\lim_{n \to \infty} E\Big[d(x, \hat{x})\Big] \leq D \tag{4.40}$$

where $E\Big[d(x,\hat{x})\Big] = \sum_x d(x,\hat{x})p(x)$, $p(x)$ is the p.d.f. of X and D is a fixed distortion value.

Definition. The **rate distortion function** $R(D)$ is the minimum rate at which is possible to transmit for a fixed distortion value D.

Definition. The **Rate Distortion Theorem** says that the rate distortion function for an i.i.d. source X with distribution $p(x)$ and bounded distance function $d(x,\hat{x})$ is equal to the associate information rate distortion function. Thus

$$H(X) \geq R(D) = R^{(I)}(D) = \min_{p(\hat{x}|x):\sum_{(x,\hat{x})} p(x)p(\hat{x}|x)d(x,\hat{x})\leq D} I(X;\hat{X}) \tag{4.41}$$

is the minimum achievable rate at distortion D.

This theorem determines the minimal amount of information (or entropy) $H(X)$ that should be communicated over a channel, so that the source can be approximately reconstructed at the receiver without exceeding a given distortion D.

Rate distortion theory gives theoretical bounds for how much compression can be achieved using lossy data compression methods. The proof of the theorem is presented in the Appendix B.1.

4.4 Conclusions

In this chapter, the following items have been discussed:

- Theoretical backgrounds in terms of probability and stochastic processes oriented to image interpretation have been shown. We have seen, how an image can be considered as stochastic process by examining spatial features. Once basics on statistics are fixed, Bayesian inference is studied pointing out the levels of hierarchy. We have described, how prior knowledge can be introduced to infer image characteristics and to select the best model fitting.

- Several information theoretic quantities for measuring system information quality as Shannon's entropy, Fisher information and Rate Distortion are itemized. This theoretical background will be applied in further chapters to real applications for estimation of optimum parameters.

5

Earth Observation Image Feature Extraction

One of the main research fields in EO field is the extraction of features for different purposes. Analyzing the spectral signature of images, several features can be obtained as contextual spatial characteristics, object's geometry, color, shapes, segmentation of images, etc. These features can be fused to merge results useful for change detection, data compression or thematic map generation.

In this chapter several feature extraction techniques will be detailed. Sec. 5.1 proposes a novel method for change detection, based on the study of mathematical combinations of spectral bands to detect interesting and non-interesting changes on airbone images. Continuing extracting features, Sec. 5.2 describes too ways of extracting contextual spatial features based on stochastic models: one applying co-occurrence matrices, and another one modeling image by Gibbs Random Fields. Changing to geometric features, Sec. 5.3 expects to detect lines longer than a certain length, and Sec. 5.4 investigates how to compress data without loosing quality in the frequency domain.

5.1 Multi Temporal Analysis of High Resolution Images

Multi temporal satellite and airborne images provide a practical way to monitor changes in the environment. They can be useful to evaluate the evolution of changes on the Earth surface, like the growth of cities, movements of the population, catastrophes detection, evolution of the vegetation in different seasons, etc. To monitor these phenomena, several images captured from the same place at different times are required. Some studies demand images taken in consecutive days and others with a difference of years.

This kind of applications of EO data requires two main issues: consistence in the illumination condition (solar angle or radar geometry) to obtain coherent and comparable results in a thematic classification, and calibrated data for reducing clutter while detecting changes.

For our study, two flight campaigns, one in winter and a second one in summer, of the airborne sensor Daedalus have been carried out capturing high resolution hyperspectral images within a period of three hours. For Daedalus sensor data characterization see Appendix A. The test sites obtained at two times are the following:

- Dorsten test site located in the region of Nordrhein-Westfallen in Germany. It includes a pipeline of natural gas with vegetation and various constructions (houses). This scene is imaged from several observation angles for the target spots.
- Borculo test site located in Netherlands. It is a typical agriculture environment with scattered buildings. The scene is captured under two different view angles.
- Vicinity of Marseille in France. It comprises different land use from rural over industrial to urban with moderate to median topography changes.

Fig. 5.1 displays the test sites showing the location of targets at both times. Images from top to bottom: A central sub-image of 466 x 466 pixels containing the relevant targets from a full scene of 600 x 600 pixels of Dorsten site; a relevant tile of 466 x 466 pixels from a large dataset of 55 cuts of the Marseille scene; and the full scene of Borculo with a size of 1000 x 625 pixels.

As it has been briefly mentioned, these data contain two main difficulties to cope with. The first one is related with the illumination conditions of the scenes. In several flights, the test sites have been captured from different view angles causing, therefore, many shadows over all data. The second problematic is the fact that data are not well calibrated. Due to the nature of targets like cars, trucks or work areas, which are relatively small and close to buildings, the uncalibrated data constitute a real problem concerning their detection. Fig. 5.2 shows the ground truth of the Dorsten site.

Knowing the problematic, a novel target detection by change analysis of multitemporal high resolution images algorithm is proposed in this thesis. The requirements on data preprocessing are realistic, i.e. they reflect the state of the art refinement which could be expected to be implemented from very large volumes of data as to be acquired from a satellite sensor in an operational system. The proposed method can be applied not only to the high resolution multispectral data presented in this chapter, but also to hyperspectral or panchromatic data.

This chapter begins in Sec. 5.1.1, presenting the preprocessing tasks that must be performed and the particularities of the change detection problem. Sec. 5.1.2 reviews the state of the art in change detection and in illumination invariant object recognition, for an easier understanding of the proposed method. In Sec. 5.1.3, a novel approach for unsupervised feature extraction for the target detection problem by classifying the

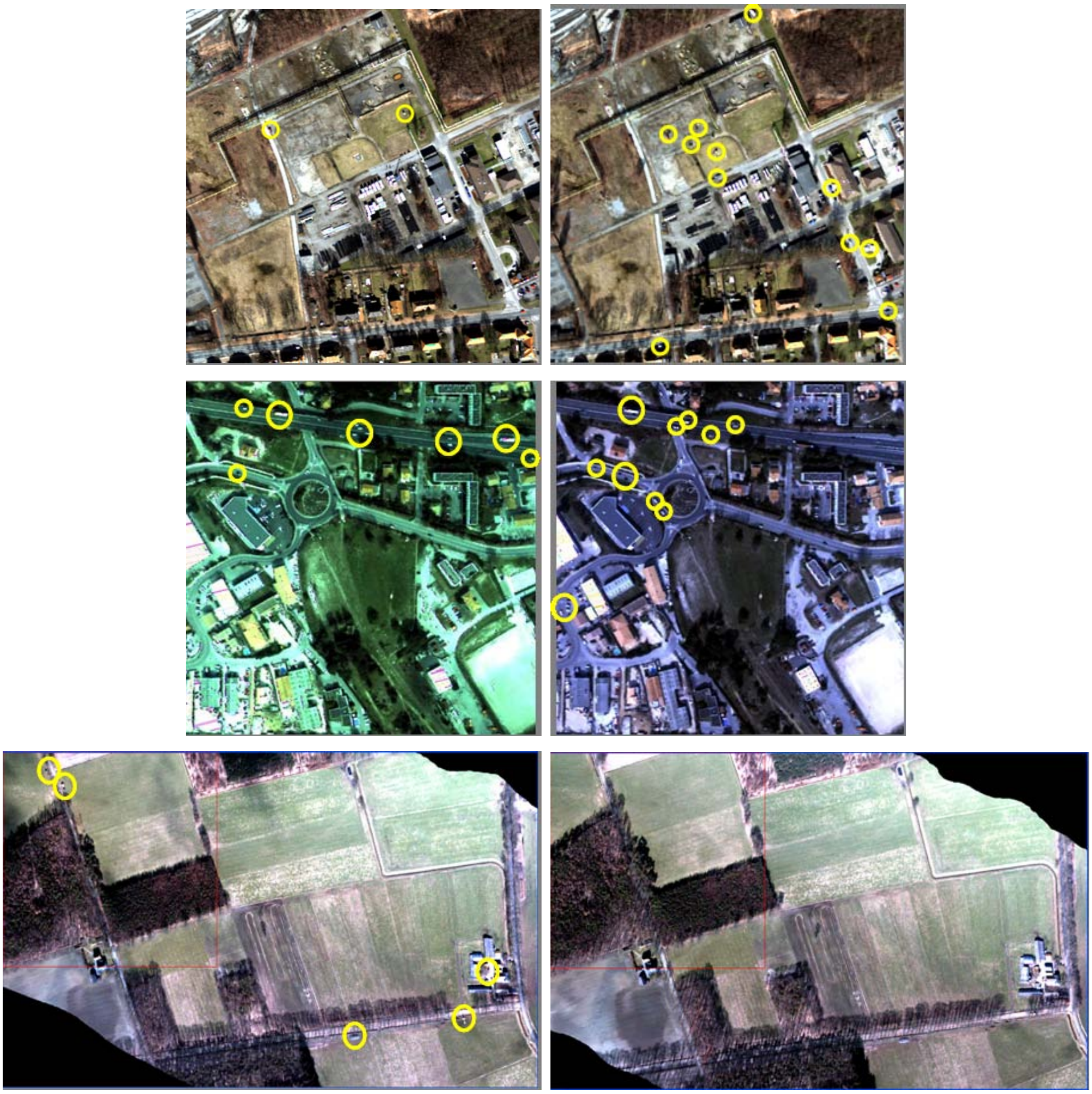


Figure 5.1: Test sites. Images from left to right, from top to bottom. Dorsten site taken at a particular time t_1, Dorsten site taken at a later time t_2; Marseille site taken at time t_1, Marseille site taken at time t_2. Borculo site taken at time t_1, Borculo site taken at time t_2. The location of targets to be detected are marked in yellow.

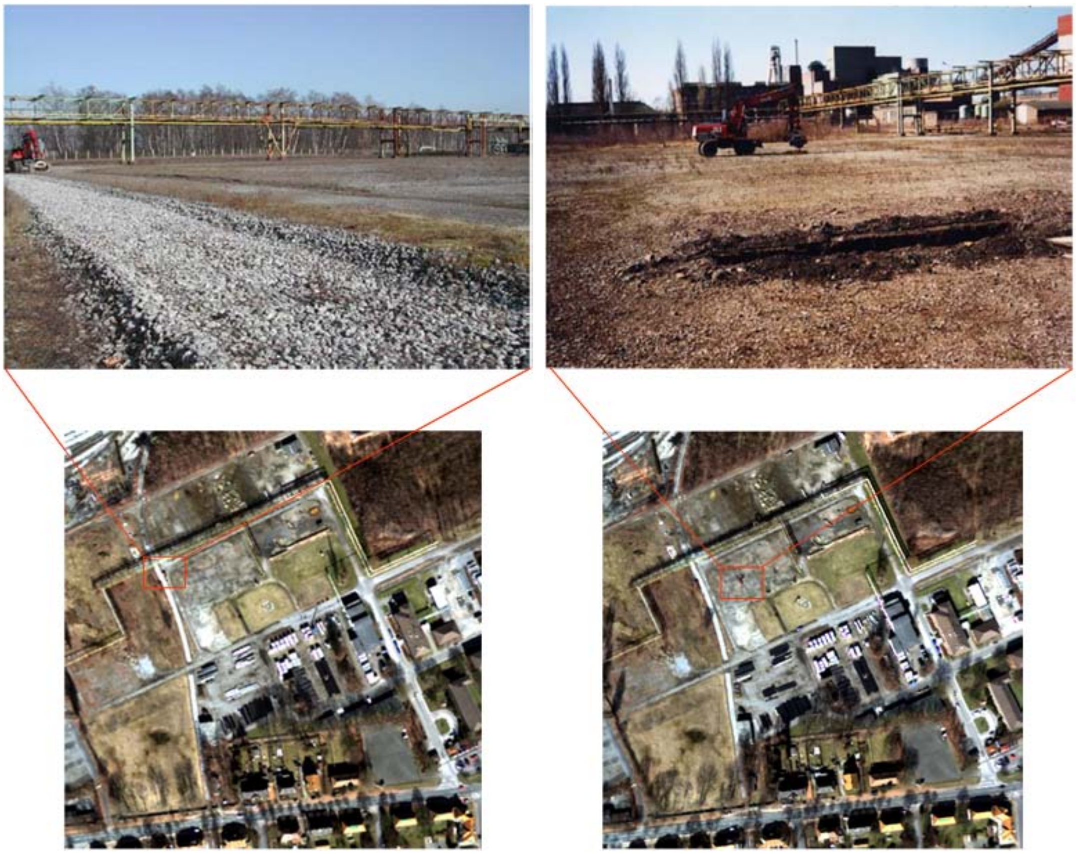


Figure 5.2: Ground truth of a target (Dorsten site): the excavator showing the position of it at the first flight (image on the left) and at the second flight (image on the right).

different changes that can be present in the bitemporal images is detailed. These are based on the analysis of the spectral bands applying different change detection operations, as differences, ratios or temporal spectral angular distance. Apart from mathematical operations between spectral bands, we search other techniques more related with the composition of spectral signature of each band that could help in target detection. The attention is focused on illumination invariant indexes. On the other side, the extraction of contextual spatial features as texture may also help.

5.1.1 Problem Statement

Before describing the proposed algorithms for the detection of targets, we have to analyze and classify the types of changes that can be found in an image, and if necessary, to analyze the preprocessing tasks to be applied to the original products.

Preprocessing

EO data must be preprocessed by performing some corrections before using them in a change detection algorithm. The most widely used rectifications are the following:

1. System corrections: The system corrections include a lost lines procedure, which uses spline interpolation with the neighboring lines. Also, it accounts for the scan angle dependent sensitivity of the sensor, which has been measured in the laboratory before the flight. The measured values are used in a multiplicative way, to adapt the whole line to the nadir value.

2. Radiometric corrections: The radiometric correction procedure is an (empirical) method to account for the view angle dependent properties of the image data in case of large fields of view (e.g. the Daedalus field of view is 86 degrees). The radiance at any view angle (off-nadir) is normalized to the radiance at the selected reference angle, usually in nadir direction.

3. Geometric rectification: The line scanner imagery is geometrically distorted with respect to a mapping frame. The upcoming high precision direct geo-referencing systems can be used for orthoimage production. The utilization of direct measurements of the image exterior orientation parameters by a GPS/IMU system for image rectification is called Direct Geo-referencing and allows a fast automatic ortho-rectification of the remotely sensed data.

Particularities of Observation of Scene Changes Phenomenology

Data registration has been carried out manually by selecting tie points. The registration error is of one pixel maximum, and non uniformly distributed over the scene. This clutter on the images together with the illumination differences hide the targets and provide shadows that are not considered as desirable changes. These items lead us to classify the changes into the following categories:

- Relevant Targets: Targets are all changes that are likely to be of risk. The targets in Dorsten, Marseille and Borculo sites are mostly cars or lorries.

- Non-Relevant Changes: These are those changes in which we are not of interest, also called false alarms. These changes come from shadows of trees, roofs or houses, etc. Depending on the orientation of the sun, the objects will experience varying illumination.

- Scene Temporal Clutter: These changes correspond to those errors coming from the system, radiometric or geometric correction algorithms explained above.

5.1.2 State of the art in Multi Temporal Analysis

Change detection in EO images has been widely investigated for many reasons, such as monitoring of vegetation changes, seasonal changes or damage assessment, among others. The most used unsupervised techniques for change detection are those based on the differences between the multi temporal images. This method processes the two multispectral images acquired at two times in order to generate a further image. After that, changes are identified by thresholding the resulting image. The univariate image differencing technique, generates the difference image by subtracting, pixel by pixel, a single spectral band of the two images under analysis. The choice of the spectral band depends on the specific type of change to be detected. Another similar technique, but using more than one spectral band is the Change Vector Analysis (CVA) [66]. In this case, for each pair of corresponding pixels, a spectral change vector is computed as the difference between the feature vectors at the two times.

Dealing with multispectral data, Principal Component Analysis (PCA) [66][6] can be applied. After computing eigenvectors and corresponding eigenvalues (see Sec. 4.1.2 for details), areas experiencing a change will be represented by small eigenvalues. A further step is the computation of the PCA of the simple image difference. Considering each pixel of the hyperspectral image as a vector:

$$X = [X_1, ..., X_n]^T, t_1 \tag{5.1}$$

$$Y = [Y_1, ..., Y_n]^T, t_2 \tag{5.2}$$

with n the number of spectral bands, X_i image pixel in band i at time 1, Y_i image pixel in band i at time 2, and t_i the capture time, a simple change detection transformation is:

$$X - Y = [X_1 - Y_1, ..., X_n - Y_n]^T \tag{5.3}$$

Finally, we compute the PCA of this difference.

The problem of this approach is, on one side, the selection of the principal component that represent a 'change' without performing a visual inspection of the data, and on the other side, having more than three channels, how to visualize them simultaneously. To overcome this problems, Multivariate Alteration Detection (MAD) [6] is introduced to compensate the radiometric and atmospheric differences by automatically adjusting the coefficients of a linear image transformation in the spectral space. The method consists of a linear transformation that maximizes the detection of change after differencing both images. Thus, being X and Y two images with a different number of spectral bands, p and q, respectively $(p \leq q)$ and

$$a^T X = a_1 X_1 + ... + a_p X_p \tag{5.4}$$

$$b^T Y = b_1 Y_1 + ... + b_q Y_q \tag{5.5}$$

linear combinations of them, the relevant changes are estimated by maximizing the variance of the difference of Eq. 5.4 and Eq. 5.5 by

$$max\{Var(a^T X - b^T Y)\} \tag{5.6}$$

with the constraints $Var(a^T X) = Var(b^T Y) = 1$. This formulation is implemented using standard canonical correlations analysis. The result of applying the MAD transformation in Dorsten site is displayed in Fig. 5.3. Observing the results, MAD is able to detect mainly changes on large regions including those that do not correspond to real targets as tree shadows. Fig. 5.4 shows in detail the results of this transformation. In yellow the detected targets (mainly lorries) are marked, and in red we display the false alarms due to difference in sun orientation over buildings or shadows.

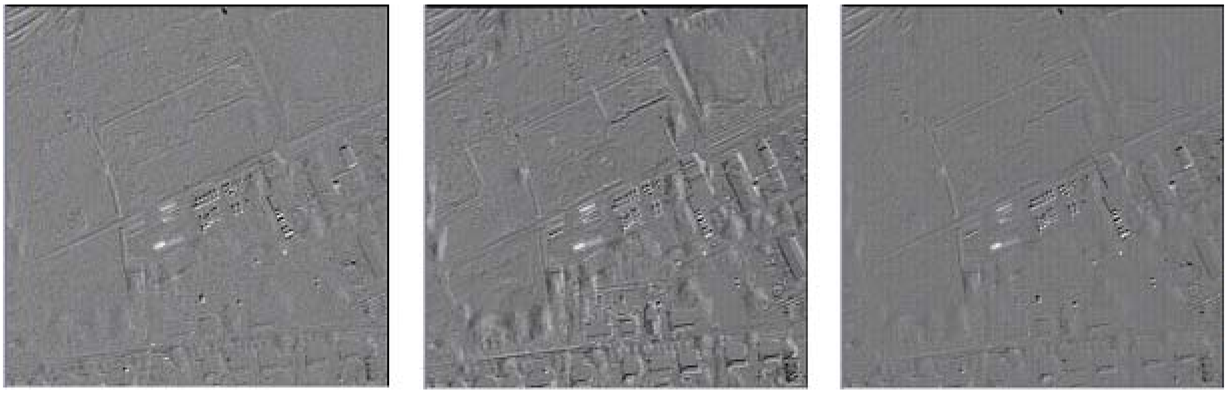


Figure 5.3: Results of applying different change detection techniques over band 0 of Dorsten bitemporal site. Images from left to right. Simple difference; Principal Component Analysis (PCA); Multivariate Alteration Detection transformation.

Bruzzone and Fernández Prieto [13] proposed two automatic unsupervised techniques based on Bayes theory for the analysis of the difference image and selection of an appropriate threshold. The first one assumes that pixels in the difference image are independent

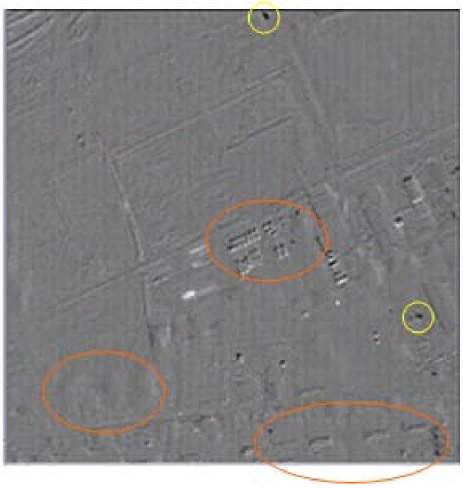


Figure 5.4: Detected targets (yellow) and false alarms (red) obtained by applying the MAD transformation.

one to each other, and the second one considers the spatial contextual information included in the neighborhood of each pixel. This method is based on Markov Random Field theory. Both techniques require the knowledge of the statistical distributions of the changed and unchanged pixels.

Trying to apply these approaches to our special case of high resolution images, we found that Bruzzone and Prieto's approach requires that changes to be identified should be large enough to be detected by the sensor. With these techniques, we could not estimate the changed position of a car in multi temporal images, for example. The same occurs with the MAD algorithm, that is not well adapted due to the specific nature of the target, as can be seen in Fig. 5.4.

Some of these methods assume that two or more images captured at different times have been exactly coregistered, enabling pixel by pixel comparison. But in satellite or aerial images, the differences in camera angles and position produce differences in distortion, and therefore, pixels of misregistration will reduce the accuracy of remotely sensed multi temporal analysis, leaving this an open issue.

Another problem in detecting targets is the radiometric variation between images of the same scene, causing differences in illumination conditions. In this field, Finlayson et al. [35] propose a method to remove shadows from an image. To reach their objective, first a shadow free grey-scale image is computed by illumination invariance at a pixel. Then, shadow edges are removed, and finally, edge maps are integrated getting the original image without shadows. If the sensor's spectral response can be represented as delta functions (they respond only at a single wavelength) and illumination is restricted to the Planckian locus, then it is possible to find a 1-D coordinate, a function of image chromaticities, which is invariant to illumination spectral signature and intensity. This gives a greyscale representation of the original image, but without shadows. From this result, edges not included in the original image and also in the invariant one are removed. Finally, Finlayson et al. apply the Weiss Method [35] to reintegrate the image. This method uses a sequence of time varying images of a fixed scene to determine the reflectance edges of the scene. The main idea is to analyse the image sequence with respect to determining edges, which correspond to a change in reflectance (those edges which persist throughout the sequence). Thus, given reflectance edges, the Weiss method is able to reintegrate the information for deriving a reflectance image.

5.1.3 Feature Extraction Methods for Target Detection

After reviewing the literature and analyzing the special problem of detecting small targets by using high resolution images, novel unsupervised techniques are developed in order to extract the primitive features, characterizing the temporal changes, both assuming independent pixel values and considering spatial contextual information. In the following sections, these techniques are detailed. In this section, we describe the methods and we present the results, but it will be in chapter 10, where we analyse and evaluate the results showing the accuracy of them for target detection.

5.1.3.1 Temporal Spectral Angular Distance

The aim of the Temporal Spectral Angular Distance (TSDA) algorithm is the detection of targets getting illumination independence. Observing test images, scene temporal clutter due to non calibrated data, and many shadows due to different illumination conditions are found. The angle between the hyperspectral signature at the two times reduces the dependence on the intensity, thus pixels with similar signature but in different illumination conditions will have small angles. Contrary, a target will be characterized by large angle values.

The method consists of a pixel-based technique, where a feature vector per pixel is formed, considering each value in each band. This operation is performed for two images taken from the same area but at different times, and then, the angle that both vectors form is computed. This is the main difference with the widely known CVA technique, that computes the simple difference of the two vectors. Proceeding in this way, a map is generated, representing unchanged pixels with small gray-level values, and changed areas with larger pixel values.

Formally, a multi temporal image is described by $B(x, y, \lambda, t)$, where x and y represent the dimensions of the image, n the number of bands, λ the corresponding wavelength and t the temporal parameter. The provided test data has been captured at two times, therefore, each pixel p_{ij} of a scene will be considered as a vector by

$$p_{ij}(t_1) = \{(i, j, \lambda_1, t_1), (i, j, \lambda_2, t_1), ..., (i, j, \lambda_n, t_1)\} \tag{5.7}$$

$$p_{ij}(t_2) = \{(i, j, \lambda_1, t_2), (i, j, \lambda_2, t_2), ..., (i, j, \lambda_n, t_2)\} \tag{5.8}$$

where i and j represents the (i, j) image pixel. The cosines of the angle α that both vectors form is computed as:

$$\cos(\alpha) = \frac{p_{ij}(t_1) \cdot p_{ij}(t_2)}{||p_{ij}(t_1)|| \cdot ||p_{ij}(t_2)||} \tag{5.9}$$

Fig. 5.5 shows a graphical representation of the angular distance α between band 1 and 2.

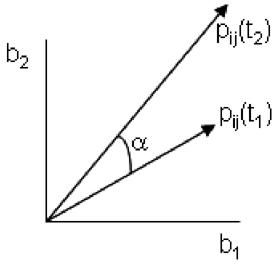


Figure 5.5: Graphical representation of the angular distance α between band 1 and band 2.

The result of applying the TSAD method (Eq. 5.9) over the test sites is displayed in Fig. 5.6. In the resulting images after applying the TSAD algorithm (third row of Fig. 5.6), we can appreciate, that some targets have been detected but also many false alarms. In Sec. 10.1, we will see, how to remove these false alarms without loosing detection accuracy.

5.1.3.2 Normalized Difference Vegetation Index

A widely extended method to discriminate two covers, presenting different reflective behaviors in two or more bands is the computation of vegetation indexes. They enable the possibility to enhance soil and vegetation and to reduce the effect of the relief in the spectral signature of different covers. It provides independence of calibration errors reducing the temporal clutter. Because Dorsten and Borculo sites are agricultural scenes, vegetative indexes are investigated.

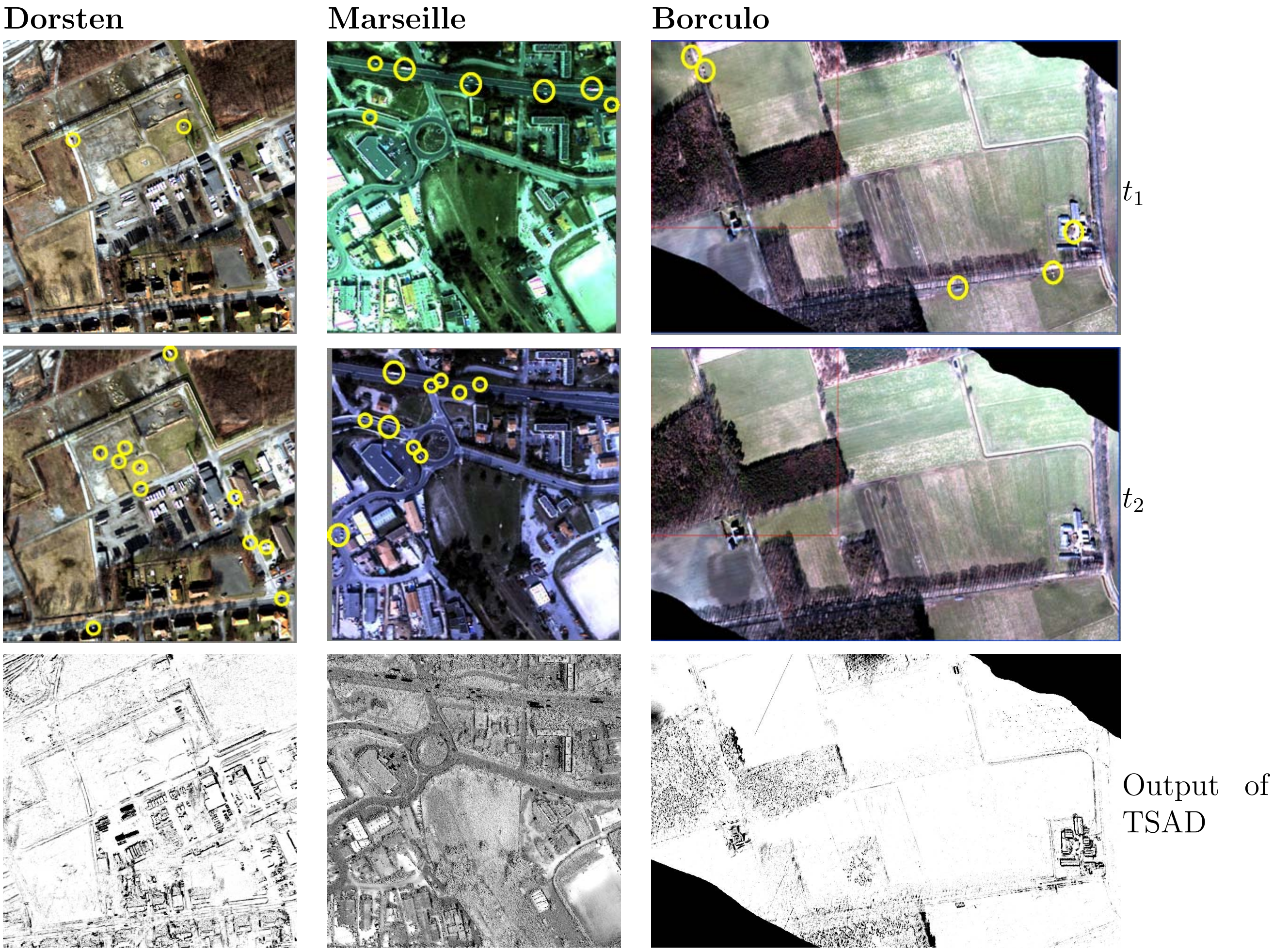


Figure 5.6: Output of the Temporal Spectral Angular Distance algorithm. Images from top to bottom, from left to right: Dorsten, Marseille and Borculo original images in time 1 showing the position of targets; Dorsten, Marseille and Borculo original images in time 2 showing the position of targets; output of the TSAD algorithm for Dorsten, Marseille and Borculo, respectively. We appreciate, that some targets have been detected but also many false alarms.

A vegetative index is a value that is calculated for EO datasets used to quantify the vegetative cover on the Earth's surface. Although many vegetative indexes exist, the most widely used is the Normalized Difference Vegetation Index (NDVI) [3]. The NDVI is computed as a ratio between measured reflectivity in the red and near infrared portions of the electromagnetic spectrum. They are mostly affected by the absorption of chlorophyll in leafy green vegetation and by the density of green vegetation on the surface. Also, in the red and near-infrared bands, the contrast between vegetation and soil is at a maximum, and it helps separating vegetated and built-up areas.

The NDVI is a type of product, known as a transformation, which is created by transforming raw image data into an entirely new image using mathematical formulas to calculate the vegetation index of each pixel. Since transformations can be created that highlight relationships and differences in spectral intensity across multiple bands of the electromagnetic spectrum, this type of product is especially useful to be applied to our multi spectral remote sensing data Dorsten and Borculo.

Being $B(t_i) = \{b_1(x, y, \lambda_1, t_i), b_2(x, y, \lambda_2, t_i), ..., b_n(x, y, \lambda_n, t_i)\}$ the formal representation of a multispectral image, captured at time t_i, where b_j represents the j band (for $1 \leq j \leq n$) of the image B, x the width of the image, y the height, n the number of bands and λ the corresponding wavelength, the NDVI of the image at time t_i is calculated as the ratio between the red (band 4) and near infrared (band 7) spectral bands (see Appendix A for data characterization):

$$NDVI(t_i) = \frac{b_7 - b_4}{b_7 + b_4} \tag{5.10}$$

where b_j corresponds to $b_j(x, y, \lambda_j, t_i)$.

The NDVI equation produces values in the range of -1.0 to 1.0, where vegetated areas will present values greater than zero, and negative values indicate non-vegetated surface features such as water, ice, snow or clouds. In order to maximize the range of NDVI results providing appropriate pixel values to be displayed in an 8 bit image, the NDVI figure must be scaled. This scaling converts a number between -1.0 and 1.0 into a pixel value more appropriated for a gray tone display. We obtain the scaled image by

$$Scaled\_NDVI(t_i) = 100(NDVI(t_i) + 1) \tag{5.11}$$

After scaling the NDVI images in both times, the pixel value range is between 0 and 200, thus those values smaller than 100 will represent clouds, snow, water, and other non-vegetative surfaces, meanwhile values greater than 100 will correspond to vegetative areas.

The NDVI results in a good invariant transformation of the data, however it uses only a part of the hyperspectral signature. This contribution enhances probability of detection in areas covered by vegetation. Fig. 5.7 displays the resulting image of this algorithm for Dorsten, Marseille and Borculo scenes. In the resulting images after applying the NDVI algorithm (third row of Fig. 5.7), we can appreciate, that some targets have been detected but also false alarms. In Sec. 10.1, we will see, how to remove these false alarms without loosing detection accuracy.

5.1.3.3 Color Normalization

Due to the difference in illumination conditions at the time, scenes are captured, some strong shadows become visible and are confused with targets. Kosugi [83] proposes an approach to detect building shadows based on a matching score in RGB vector space, considering the brightness and blue intensity. Because our data are mainly vegetative scenes and we are interested in a more generic method to model illumination, we propose a linear model for compensating the radiometric variation between images of the same scene.

The algorithm consists of a color based technique, called Color Normalization (CN), [5] that combines multispectral images with high resolution data. Each band in the hyperspectral signature is multiplied by a ratio of the high resolution data divided by the sum of the red, green and blue spectral bands. In Daedalus' spectral sensor, the red band corresponds to channel 5, the green one to channel 3 and the blue to band number 2 (see Appendix A for data characterization). The three spectral bands are automatically resampled to the high-resolution pixel size using a nearest neighbor procedure.

This sharpening technique is mostly used for data fusion, however, we apply this technique in order to obtain a remarkable color contrast between bands at aiming to detect strong shadows. We adapt the method, replacing the high resolution data by the thermal infrared band (channel 11), looking not only for contrast in the color, but also contrast in the temperature.

Considering $B(t_i) = \{b_1(x, y, \lambda_1, t_i), b_2(x, y, \lambda_2, t_i), ..., b_n(x, y, \lambda_n, t_i)\}$ a multispectral image with x width, y height, n bands, with the wavelength λ and captured at time t_i, the CN image at time i is represented by

$$CN_i = \{CN_5, CN_3, CN_2\} = \{b_5(x, y, \lambda_5, t_i), b_3(x, y, \lambda_3, t_i), b_2(x, y, \lambda_2, t_i)\} \tag{5.12}$$

where CN_5, CN_3 and CN_2 corresponds to the CN of band five, three and two, respec-

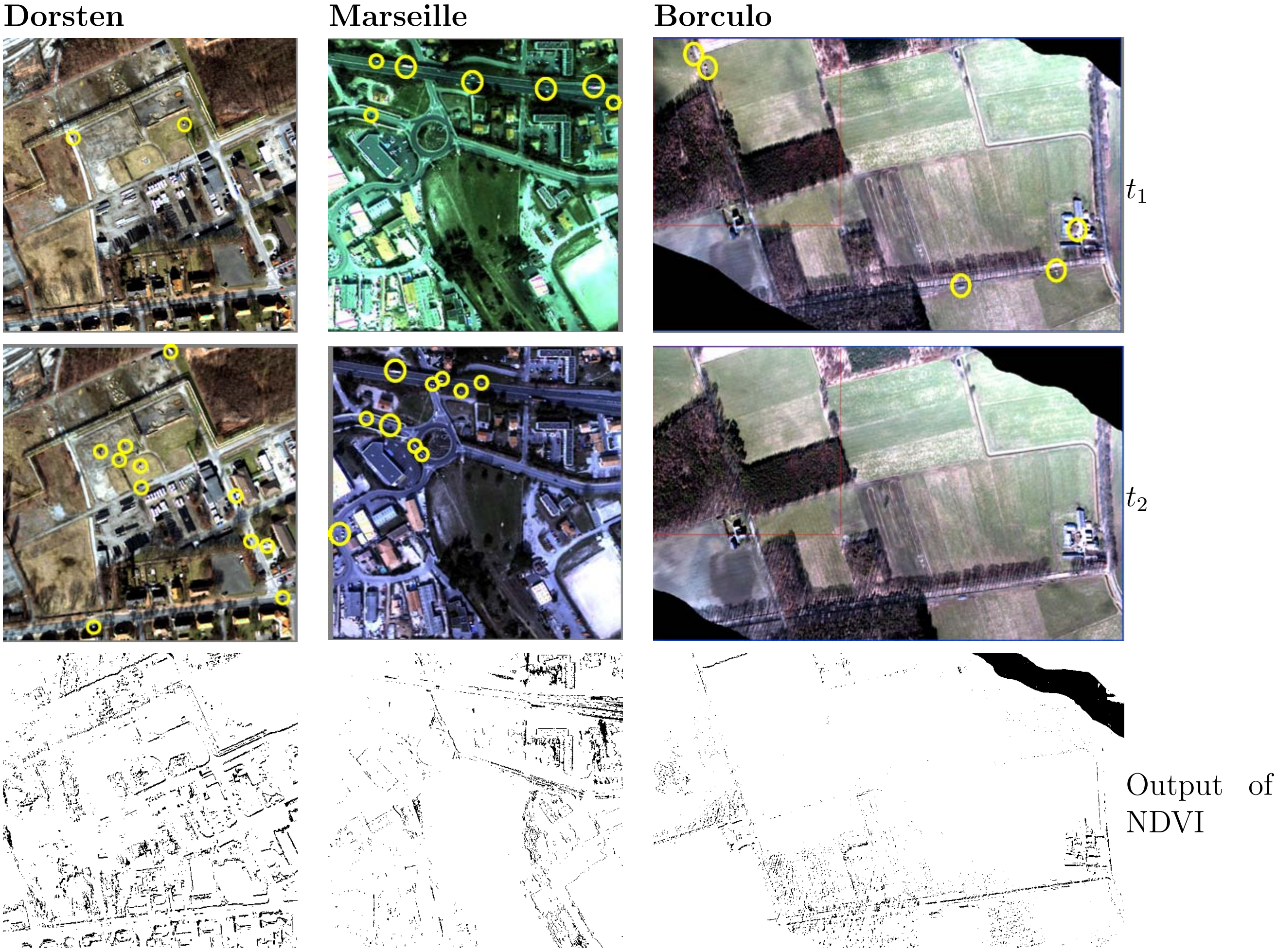


Figure 5.7: Output of the NDVI algorithm. Images from top to bottom, from left to right: Dorsten, Marseille and Borculo original images in time 1 showing the position of targets; Dorsten, Marseille and Borculo original images in time 2 showing the position of targets; output of the NDVI algorithm for Dorsten, Marseille and Borculo, respectively. This contribution enhances the probability of detection in areas covered by vegetation. We appreciate, that some targets have been detected but also many false alarms.

tively being computed as:

$$CN_j = b_j * (\frac{b_{11}}{b_5 + b_3 + b_2}) \tag{5.13}$$

where $j \in \{2, 3, 5\}$, b_5, b_3 and b_2 corresponds to the RGB channels of the original image, and b_{11} to the thermal infrared band of the data.

After normalizing the image color by using Eq. 5.13, the statistical distribution of pixels is analyzed. It consists of computing to each pixel, the cutoff value V in a standard Gaussian distribution with zero mean and a unit variance such that the probability that a random variable X is greater than V is equal to a user-supplied probability P. This threshold will be applied to the CN results, and this new output will be called Color Normalization Gaussian (CNG). Fig. 5.8 displays the CNG of Dorsten, Borculo and Marseille scenes. This contribution enhances the probability of detection of strong shadows and, therefore, provides a mechanism to build a mask of false alarms (see Sec. 10.1).

5.2 Texture Analysis

In Sec. 5.1 we have explained different methods for the detection of targets in multi-temporal high resolution images in an unsupervised way. In this section, we concentrate our study in the spatial information trying to extract the textural parameters from EO datasets.

Texture is very easy to detect visually, but very difficult to define and even to extract from an image in an automatic way. Texture analysis is a very valuable tool for image analysis in EO, and the existing algorithms for extracting texture are often linked with an application. Texture is widely used in industrial image processing for the inspection of materials, in medical image analysis to distinguish between types of blood cells, and in document processing for location of bar codes, as example. In the EO field, texture is used to detect complex cover-types such as 'cumulus clouds', 'young forest', 'city' or 'fields'.

In the literature there exist several algorithms for extracting texture parameters. Here, we want to describe one based on co-occurrence matrices (GLCM) applied to SAR imagery and a second one based on Gibbs Random Fields (GRF) for optical sensors.

5.2.1 Gray Level Co-Occurrence Matrices

A classical approach to quantitatively evaluate textural parameters is the computation of the Gray-Level Co-occurrence Matrix (GLCM) that represents the spatial dependence

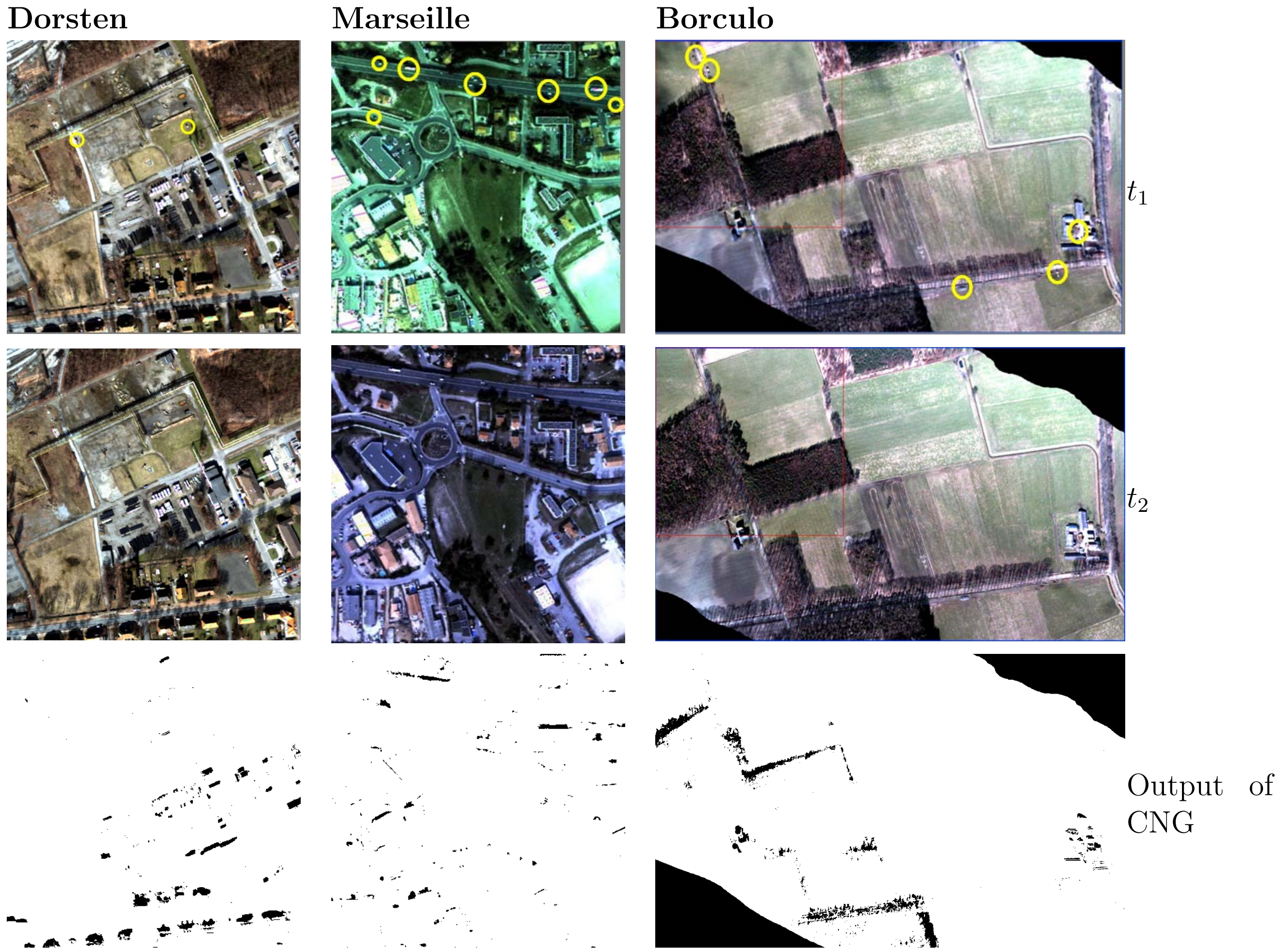


Figure 5.8: Output of the CNG algorithm. Images from top to botom, from left to right: Dorsten, Marseille and Borculo original images in time 1 showing the position of targets; Dorsten, Marseille and Borculo original images in time 2 showing the position of targets; output of the CN algorithm after applying the threshold for Dorsten, Marseille and Borculo, respectively. This contribution enhances the probability of detection of strong shadows.

of texture in an image. This idea came from Haralick [67], who proposed 14 different measures, including the features detailed on the next pages. The GLCM is a tabulation of how often different combinations of pixel brightness values (grey levels) occur in an image. A GLCM is created in the following way:

1. Definition of the parameters:

    - window size, W: dimensions of the region where GLCM is computed.
    - quantization levels of the image, N_g: number of gray levels.
    - displacement, d: distance between pixels.
    - orientation, θ: spatial relation between reference pixel and neighbors: north, south, east, west and the for diagonals (0°, 45°, 90° and 135°) (See Fig. 5.9).

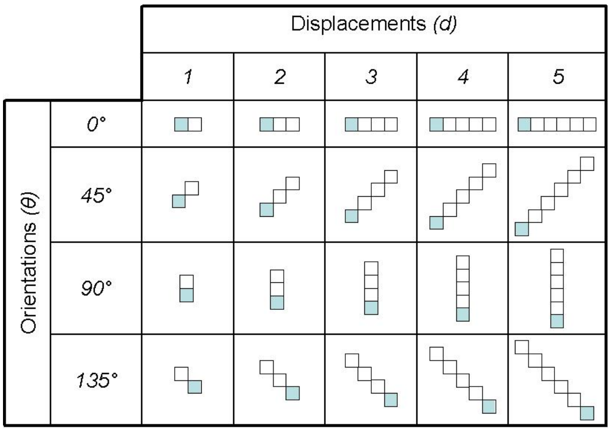


Figure 5.9: Displacements d and orientations θ between reference pixel (marked in blue) and its neighbors.

2. Considering only the samples in W, the element i, j of the GLCM will be defined as the number of times two samples of intensities i and j ($0 \leq i, j < N_g$) occur in an spatial relationship (considering displacement, d, and orientation, θ).

3. GLCM symmetric: To the obtained GLCM, add a transposed copy of it. This produces a symmetric matrix, where each element is denoted as $V_{i,j}$.

4. Normalization of the GLCM: Each element, $P_{i,j}$, of the final GLCM is calculated as

$$P_{i,j} = \frac{V_{i,j}}{\sum_{i,j=0}^{N_g-1} V_{i,j}} \tag{5.14}$$

Each element of the GLCM, $P_{i,j}$, is considered as the probability of finding the relationship i, j (or j, i) in W. If the image contains a high radiometric resolution, the number of grey levels will be large, causing zero value in most of the relationship between pixels. Therefore, the image data must be pre-scaled before computing the GLCM to reduce the number of grey levels. The loss of information while scaling the pixel values leads to a low number of signal classes.

Once we have computed the normalized GLCM, we can obtain the mean, μ_x and μ_y, and variances, σ_x^2 and σ_y^2, for rows and columns of the matrix as:

$$\mu_x = \sum_i \sum_j i P_{i,j} \qquad \mu_y = \sum_i \sum_j j P_{i,j} \tag{5.15}$$

$$\sigma_x^2 = \sum_i \sum_j (i - \mu_x)^2 P_{i,j} \qquad \sigma_y^2 = \sum_i \sum_j (j - \mu_y)^2 P_{i,j} \tag{5.16}$$

The following equations define the extracted texture measures:

1. Energy:

$$f_1 = \sum_i \sum_j P_{i,j}^2 \tag{5.17}$$

2. Contrast:

$$f_2 = \sum_i \sum_j P_{i,j}(i,j)^2 \tag{5.18}$$

3. Correlation:

$$f_3 = \frac{(\sum_i \sum_j (ij) P_{i,j}) - \mu_x \mu_y}{\sigma_x \sigma_y} \tag{5.19}$$

4. Homogeneity:

$$f_4 = \sum_i \sum_j \frac{1}{1 + (i-j)^2} P_{i,j} \tag{5.20}$$

5. Entropy:

$$f_5 = -\sum_i \sum_j P_{i,j} log(P_{i,j}) \tag{5.21}$$

6. Autocorrelation:

$$f_6 = \sum_i \sum_j (ij) P_{i,j} \tag{5.22}$$

7. Dissimilarity:

$$f_7 = \sum_i \sum_j |i - j| P_{i,j} \tag{5.23}$$

8. Cluster Shade:

$$f_8 = \sum_i \sum_j (i + j - \mu_x - \mu_y)^3 P_{i,j} \tag{5.24}$$

9. Cluster Prominence:

$$f_9 = \sum_i \sum_j (i + j - \mu_x - \mu_y)^4 P_{i,j} \tag{5.25}$$

10. Maximum Probability:

$$f_{10} = MAX_{i,j} P_{i,j} \tag{5.26}$$

Soh et al. [77] describe three implementations for the extraction of texture parameters based on GLCM computation: the Mean Displacement and Mean Orientation (MDMO) matrix, the X^2-Optimal Displacement and Mean Orientation (ODMO) matrix, and the X^2-Optimal Displacement and X^2-Optimal Orientation (ODOO) matrix. Comparing these implementations, clusters through the MDMO method are more selective than those obtained with ODMO and ODOO. MDMO detects the difference between low and high

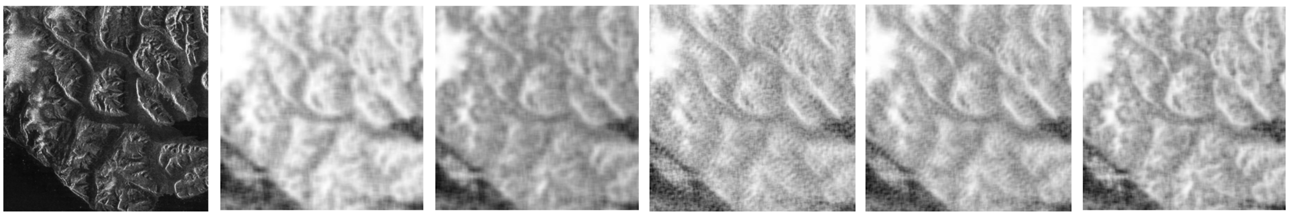


Figure 5.10: Texture analysis of SAR sea ice imagery using Gray Level Co-occurrence Matrices. Images from left to right: Original image; energy; entropy; contrast; dissimilarity; autocorrelation features.

contrast, and is able to capture local and global details of a texture, due to the combination of features of all displacement values. For these reasons, we select the MDMO implementation for extraction of textures being computed as:

$$F_i^{MDMO} = \frac{1}{|\Delta|}\sum_{d\epsilon\Delta}\frac{1}{|\Theta|}\sum_{d\epsilon\Theta} f_i(d,\theta) \tag{5.27}$$

where d is the displacement, θ the orientation, $|\Delta|$ the number of displacements, $|\Theta|$ the number of orientations, and $f_i(d,\theta)$ the computed feature i with displacement d and orientation θ.

Our case study aims at determining, which parameter values and representation are best for mapping sea ice texture using ERS-1 SAR images. For achieving this purpose, we compute the GLCM of an 8-bit image data, that means 256 grey levels (matrix of 256 x 256). But, as explained before, this introduces too many grey levels yielding in a matrix with a mayority of zero values. Therefore, we pre-scale the data obtaining a 16 x 16 GLCM, where each element corresponds to 8 grey levels, increasing the probability distribution of the relationship between pixels. We compute all features, but despite pre-scaling the image, only five of them offer good results due to the nature of the data. These measures are energy, entropy, contrast, dissimilarity and autocorrelation computed for all orientations: 0° (horizontal), 45° (east diagonal), 90° (vertical), 135° (west diagonal), and displacements, 1 to 5. Fig. 5.10 displays the MDMO computed for those features with $|\Theta| = 4$ and $|\Delta| = 5$.

We have described a classical method for extracting spatial information based on co-occurrence matrices for SAR images. But, a limitation of using co-occurrence matrices is the appearance of multiple zeros in the matrix due to the lack of relationship between neighboring pixels with different color, resulting in a low number of textures, we can extract. However, this could be useful to prohibit large classes to be formed. Although GLCM is still in use due to its fast implementation often applied when textures are known a priori, the loss of information due to the scaling of the data, leads to think in other methods providing further spatial features.

5.2.2 Gibbs Random Fields Texture Models

An alternative method to extract structural information from remotely sensed images is to model data by stochastic processes as parametric models $p(X|\theta, M)$, where X is the data, θ a set of parameters, and M the model. Finding a good texture model is equivalent to find a good image model. Because we do not have any application in mind, and we are interested in describing the image content, we select the closest model to data that needs a minimum number of parameters. This corresponds to the Gibbs Random Fields (GRF) family of stochastic models [28][75].

GRF considers an image as a set of pixels assumed to be a realization of a random field. If the probability density function (p.d.f.) of a single pixel x_s can be determined by its local neighbors, $\mathcal{N}_s$, it can be characterized by a Gibbs distribution as follows:

$$p(x_s|\mathcal{N}_s;\theta) = \frac{1}{Z_s} e^{-H(x_s|\mathcal{N}_s;\theta)} \tag{5.28}$$

where Z_s is a normalization factor, $H(x_s|\mathcal{N}_s;\theta)$ a local energy function with θ a vector parameter that weights the contributions of each neighbor to the central pixel. This parameter vector θ will characterize the spatial information of the image. This algorithm has been explained in Sec. 4.1.3.2.

The energy function, $H(x_s|\mathcal{N}_s;\theta)$, is computed over an analyzing window to estimate the interaction between pixels in the neighborhood. The size of this analyzing window fixes the number of samples available for the estimation and will directly affect the accuracy of the extracted texture parameters. Then, which is the optimal analyzing window size? Which is the size of the analyzing window containing more spatial information? How to estimate it? These questions will be answered in chapter 7 while describing the estimation of parameters by using Rate Distortion methods.

As result of computing a pixel based energy function over the analyzing window, an estimated parameter vector $\hat{\theta}$ is obtained. From $\hat{\theta}$, the norm, variance, evidence and mean are estimated, getting the texture parameters. The norm $|\hat{\theta}|$ provides a measure of the spatial structures in the image; the variance measures the grey value variations; the mean is calculated over the grey values of the image, and the evidence $p(x|M)$ represents the probability of the data x given the model M.

Fig. 5.11 shows an example of the extracted texture parameters. The original image covers the town of Oberpfaffenhofen in the south of Germany and vicinity. It is a Daedalus

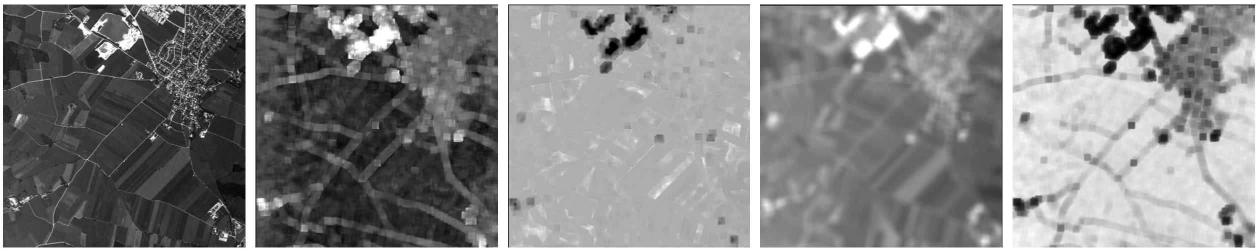


Figure 5.11: Estimated texture parameters. Images from left to right. Band 7 of the original Daedalus image from Oberpfaffenhofen (Germany); norm; variance; mean; evidence extracted parameters.

image of 605 x 605 pixels consisting of 12 spectral bands. Band 7 has been selected for the extraction of the texture parameters due to the gaussianity distribution present in this band.

GRFs are very suitable to automatically extract a large set of different textures for remotely sensed images. In particular, the autobinomial model seems to be a useful descriptor for EO applications and for object recognition due to the strength and randomness provided by the norm and evidence measurements, respectively.

5.3 Linear Feature Extraction

The algorithms described in previous sections give an impression of how to extract spatial and spectral features from EO images. However, as the spatial resolution grows, also geometric features should be included in our collection of methods. In this section, a linear feature extraction algorithm is proposed, where only signal characteristics of the data are considered.

In the literature, we can find many filters to detect lines, as Sobel filter, Roberts and Canny edge detector or the zero crossing detector. Because, the feature extraction algorithms, described in this dissertation will be operated in a real time scenario, a fast computation of the methods is a critical factor. Therefore, among the filters we select the fastest one, which is the Roberts edge detector.

Roberts edge detector

The Roberts operator performs a 2-D spatial gradient which often corresponds to edges on a greyscale image based on a pair of convolution masks defined as:

$$\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

These kernels enable the edge detection in two different and perpendicular orientations. They are applied separately to the original image, computing the gradient in each orientation, calling this G_x and G_y, respectively. Then, they are combined together to find the gradient at each point. The magnitude of the gradient is computed by:

$$|G| = \sqrt{G_x^2 + G_y^2}$$

often approximated by

$$|G| = |G_x| + |G_y|$$

and the angle of the orientation (relative to the pixel grid orientation) is defined as:

$$\theta = \arctan(G_y/G_x) - 3\pi/4 \qquad (5.29)$$

To speed up this calculation, Roberts method approximates this computation applying a pseudo-convolution mask to the original data as

$$|G_{jk}| = |F_{jk} - F_{j+1,k+1}| + |F_{j,k+1} - F_{j+1,k}| \qquad (5.30)$$

where $|G_{jk}|$ is the absolute magnitude of the gradient for the pixel (j, k), and (j, k) is the coordinate of each pixel F_{jk} of the image.

Fig. 5.12, shows the result of executing the Roberts edge detector to a Daedalus image covering Borculo area. Exploring the results, geometric features of the image are enhanced, i.e. paths between agricultural fields or walls of the buildings. Nevertheless, growing fields or grass are also highlighted, and consequently, considered as lines. To discriminate edges from other geometric objects, a further step is given by performing a Hough transformation.

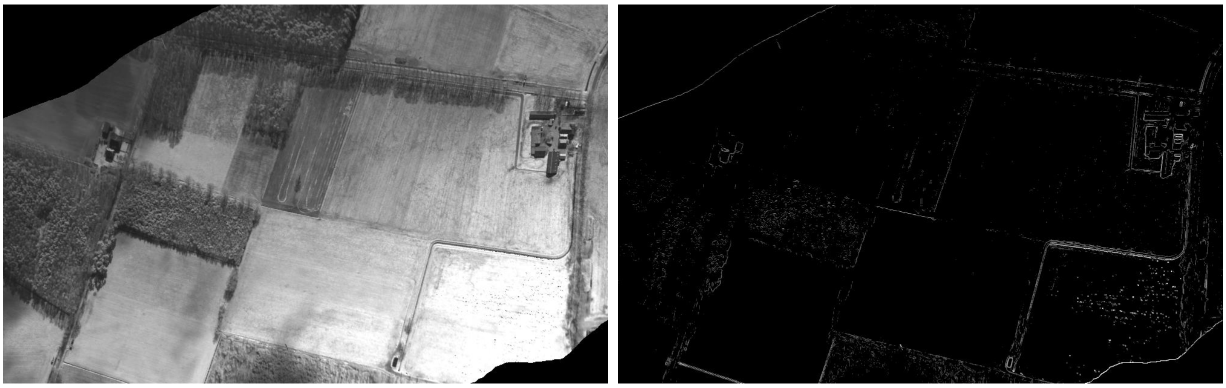


Figure 5.12: Roberts edge detector. Images from left to right. Daedalus image covering Borculo and result of performing the Roberts edge detector.

Hough Transformation

The principle of the Hough transformation is based on the idea that an infinite number of lines passes through any point. The transformation tries to determine, which of these theoretical lines fits most closely to the data. To do that, each line is represented by two parameters: the length ρ and the angle θ in the Hough domain (see Fig. 5.13). Thus, each pixel (m, n) in the original image A, is transformed into a sinusoid ρ. Therefore, a line in the image can be unique defined by (ρ, θ) where $\theta \in [0, 2\pi]$ and $\rho \geq 0$. After performing this step, each nonzero pixel in the image is transformed in a sinusoid in the Hough domain. Then, if the curves of two pixels are superimposed, the location where they cross in the Hough space corresponds to a line through both pixels in the image space. Fig. 5.13 shows the line transformation from image domain to Hough one, and the representation of a sinusoid in the Hough domain.

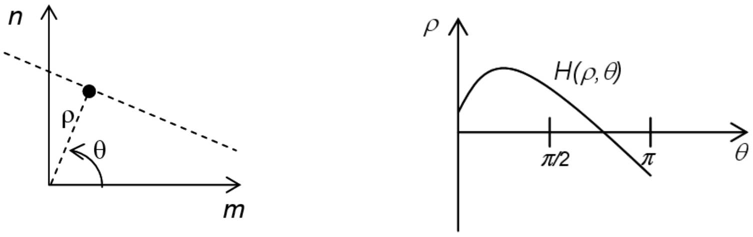


Image domain *Hough domain*

Figure 5.13: Image on the left shows the line transformation from image domain to Hough domain, and the image on the right displays the transformation from Hough domain to image domain.

The formal representation of the Hough transformation for an image A of m by n dimensions is:

$$H(\rho,\theta) = \sum_m \sum_n A_{mn}\delta(\rho,\rho^{'}) \tag{5.31}$$

where

$$\rho^{'} = (m\Delta x + x_{min})\cos\theta + (n\Delta y + y_{min})\sin\theta$$

and

$$\delta(\rho,\rho^{'}) = \begin{cases} 1 & \rho = \rho^{'} \\ 0 & otherwise \end{cases}$$

After transforming the image to the Hough domain, we also need to transform back to the spatial domain in order to build the final map of detected lines. To do that, the Hough backprojection is performed. Since indiscriminants backprojection will result in numerous false lines, the length of the lines to be detected must be limited. This decision is taken by inspecting the original image. For instance, looking at the Borculo original scene shown in Fig. 5.12, we are interested in discriminating paths or lines of the buildings from grass. Knowing the spatial resolution of the image, we can derive the pixel length, and consequently, the extension of the interesting lines, that for the case of Borculo will be of 30.

Hough Backprojection

Once this decision is taken, we proceed with the Hough backprojection image B_{mn}, that transforms the (ρ,θ) points in the Hough domain into straight lines in the spatial domain. This transformation is computed as:

$$B_{mn} = \begin{cases} \sum_\theta \sum_\rho H(\rho,\theta)\delta(n,[am+b]) & |sin\theta| > \frac{\sqrt{2}}{2} \\ \sum_\theta \sum_\rho H(\rho,\theta)\delta(m,[a^{'}n+b^{'}]) & |sin\theta| \leq \frac{\sqrt{2}}{2} \end{cases}$$

where

$a = -\frac{\Delta x}{\Delta y}\frac{\cos\theta}{\sin\theta}$; $b = \frac{\rho - x_{min}\cos\theta - y_{min}\sin\theta}{\Delta y \sin\theta}$; $a^{'} = \frac{1}{a}$; $b^{'} = \frac{\rho - x_{min}\cos\theta - y_{min}\sin\theta}{\Delta x \cos\theta}$

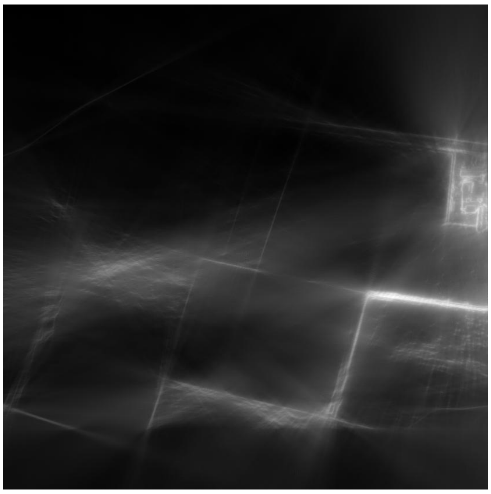


Figure 5.14: Detected lines after Hough Backprojection in Borculo scene.

Fig. 5.14 displays the result of the Hough backprojection method showing the detected lines for Borculo scene.

5.4 Discrete Cosine Transform Based Dimension Reduction

The presented feature extraction algorithms based on the spectral signature and contextual information stay in the spatial domain by evaluating some equations over the spectral bands. In this section, another algorithm for obtaining characteristics is presented, but in this case, looking at the frequency domain with the unique purpose of compressing the data. We will explain, how the Discrete Cosine Transform (DCT) can help us to select some spectral bands from the data without compromising the quality of the resulting image. Transform coding relies on the premise that image pixels exhibit a certain level of correlation with their neighborhood. Consequently, these correlations can be exploited to predict the value of a pixel from its neighbors. Therefore, a transformation is defined to map this spatial (correlated) data into transformed (uncorrelated) coefficients.

DCT [50][82] was developed in 1974 by Ahmed, Natarajan and Rao, and its application to image compression was pioneered by Chen and Pratt in 1984. DCT helps to separate an image into spectral sub-bands exploiting the redundancies in image data to provide compression, reducing, therefore, the entropy. It is similar to the Discrete Fourier Transform (DFT) because it transforms a signal from the spatial domain to the frequency domain. However, DFT computes complex numbers resulting in a more expensive computation method than DCT, which deals with real numbers. We can also compare DCT with PCA. In this case, using PCA, each time new data is introduced, the covariance matrix has to be recomputed, rendering the process very expensive. If data is a stationary Markov-1 process, Discrete Cosine Transform (DCT) can be computed, and many real signals behave statistically like a stationary Markov-1 signal. DCT stays invariant to the amount of data rendering the computation of DCT more efficient than the PCA one.

Because DCT attempts to decorrelate image data, each transform coefficient can be encoded independently without losing compression efficiency after decorrelation. Then, the coefficients of the output image B extracted from input image A are:

$$B(u,v) = \alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} A(x,y)\cos[\frac{\pi(2x+1)u}{2N}]\cos[\frac{\pi(2y+1)v}{2N}] \tag{5.32}$$

for $u, v = 0, 1, 2, ..., N-1$ and $\alpha(u)$ and $\alpha(v)$ defined as

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}} & \text{if u=0} \\ \sqrt{\frac{2}{N}} & \text{if u} \neq 0 \end{cases}$$

where N represents the height and width of the square input image A.

The inverse transform, used to rebuild original image starting with the DCT coefficients is defined as

$$A(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} \alpha(u)\alpha(v)B(u,v)\cos[\frac{\pi(2x+1)u}{2N}]\cos[\frac{\pi(2y+1)v}{2N}] \tag{5.33}$$

for $x, y = 0, 1, 2, ..., N-1$. If we compare the resulting image of performing the inverse transform with the original one, we would observe, that the error between them is only of about 5%, ensuring that the DCT keeps the energy invariant.

For most images, much of the signal energy lies at low frequencies (see Fig. 5.15). The lower right values represent higher frequencies, and are often small enough to be neglected with little visible distortion. With this property, compression is achieved.

DCT has the following properties:

- Decorrelation: The principal advantage of image transformation is the removal of redundancy between neighboring pixels. This leads to uncorrelated transform coefficients which can be encoded independently.

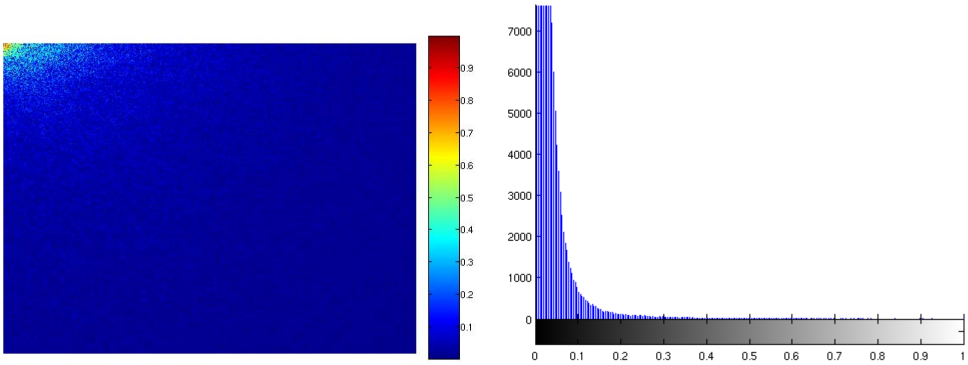


Figure 5.15: Image on the left shows the magnitude of the spectrum. The right plot shows the distribution of the energy at different frequencies. The signal energy concentrates most of the power on the lower frequencies.

- Energy Compaction: DCT exhibits excellent energy compaction for highly correlated images (see Fig. 5.15).
- Separability: $B(u, v)$ can be computed in two steps by successive 1-D operations on image rows and columns.
- Symmetry: Rows and columns operations in $B(u, v)$ are functionally identical, called symmetric transformation. This enables to process separately and offline the image, reducing the computation time.
- Orthogonality: DCT basis functions are orthogonal, thus none of the basis functions can be represented as a combination of other basis functions, waveforms are independent.

For the experiments, a medium resolution image dataset taken from the Meris sensor is selected. The dataset contains twenty Meris products Level 1b of 14848 by 1121 pixels and 15 bands (see Appendix A for data characterization). The image processing begins by performing a *smile correction* that consists of the normalization of the wavelengths within one spectral band to one reference wavelength. This is necessary due to the small variations of the spectral wavelength of each pixel along the Meris product level 1b. Next correction that must be performed before applying the DCT compression, is a *radiometric conversion*. Radiances are converted to reflectance, using the sun zenith angle cosine, interpolated at the pixel and the spectral flux read from the Level 1b product.

After performing these corrections, we extract the DCT coefficients from data. Then, images are reconstructed by performing the inverse DCT operation on the coefficients. Based on the heterogeneous bandwidth of the original data, DCT coefficients can be

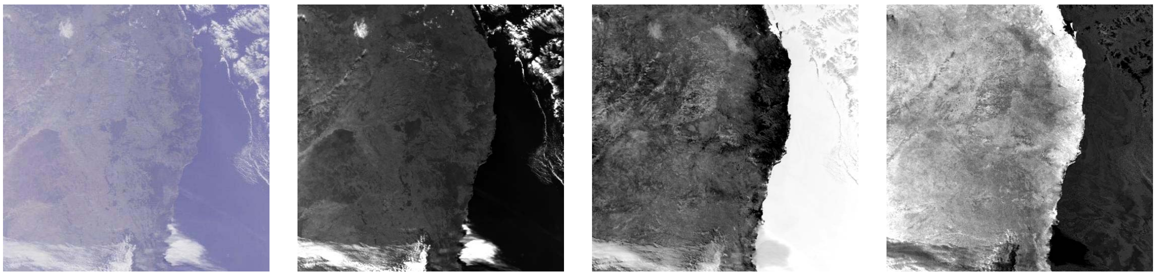


Figure 5.16: Discrete Cosine Transform. Images from left to right: Original Meris product Level 1b quicklook; Band 1; Band 2; Band 6 after DCT compression.

discarded (bands 3, 4, 5 and from 7 to 15) while rendering acceptable quality. Thus, analyzing the statistics of the spectral signature in the DCT domain, it can be observed, that the highest energy is concentrated in bands 1, 2 and 6. Therefore, some of the high frequency content (from band 7 to band 15) can be discarded without significant quality degradation, and we go from the fifteen dimensional space, to the three dimensional space. Fig. 5.16 shows bands 1, 2 and 6 resulting from the computation of the inverse DCT once the DCT coefficients have been extracted.

5.5 Conclusions

In this chapter, the following feature extraction algorithms have been discussed:

- Multi temporal analysis: this novel contribution proposes a method, able to detect small targets from bitemporal high resolution images not dependent of illumination conditions and co-registration errors. Three techniques have been developed: one based on detection of targets getting illumination independence, a second one that quantifies the vegetative cover on the Earth's surface, and the last one focusing on shadow detection. In chapter 10, these methods will be combined and evaluated to quantify the accuracy of them.

- Texture analysis: an important feature in EO data is the spatial information that considers the relationship between neighboring pixels. We have studied two approaches to extract textural characteristics: a non parametric one, based on grey level pixel vicinity and a second one, based on the statistical analysis of local Gibbs Random Fields. This technique is mainly useful in object recognition.

- Linear features: besides textural information, linear features become interesting, when dealing with segmented areas, paths recognition or detection of buildings. An approach based on Roberts filter and Hough transformation is described to extract linear features longer than a certain threshold.

- Dimension reduction of feature space: an important characteristic in image processing is the huge volume of data generated after extraction of features. A compression algorithm based on DCT is developed. With this method, spectral resolution can be drastically reduced without loosing quality on data.

6

Clustering

Since the feature extraction algorithms produce large volumes of data that are burdensome in practice, estimated image parameters have to be compressed and reduced. Vector quantization permits to find the value that better represents a set of grouped pixel in the realization space. This procedure can be viewed as a compression process. As information theory says, data processing cannot increase the information, and at each step in the hierarchical scheme, a certain amount of information is lost. The clustering technique, which is similar to vector quantization, reduces the accuracy of the system, but is justified due to its ability of large data reduction. In order to reflect existing structures of data in the feature space and to avoid the time-consuming calculation of similarity functions, unsupervised clustering is performed across the archive.

Model based image parameter estimation can be interpreted as a data coding, where the code depends on the assumed data model. If the parameter space is compressed using a vector quantization, it will be possible to use Rate Distortion (RD) theory to assess the model quality as properties of the data.

Continuing with the information hierarchy described in Sec. 3.1, we begin this chapter, presenting the theoretical concepts of the clustering phase of this hierarchy. In Sec. 6.2, a classical and widely used clustering method, called k-means is presented, and in Sec. 6.3, an enhanced version of k-means algorithm in terms of computation time. This chapter builds the basis of next one, where distortion calculations will be applied to a set of classes obtained from a clustering approach for optimization of parameter estimation.

6.1 Clustering Phase in Information Hierarchy

Following the information hierarchy presented in chapter 3, extracted features have to be compressed by clustering techniques. The clustering method consists of an unsupervised classification together with a Bayesian method for determining the optimal number of classes. This approach is called Bayesian classification or AutoClass [45]. Thus, the probabilistic link between θ_j, the jth data point in the feature space, and ω_i, the ith cluster, is modeled by

$$p(\theta_j|\theta_j \in \omega_i, V, T) \tag{6.1}$$

where T represents the classification model, and V the parameter of the model (in our case, number of classes). This probability computes for a given cluster ω_i, how probable is that the θ_j data point belongs to that cluster. The parametric class model, applied to represent a cluster is a fully covariant Gaussian with μ_i mean and Σ_i variance. With this model, probabilities are described by using the full covariance matrix Σ as:

$$p(\theta_j|\theta_j \in \omega_i, V, T) = \frac{1}{(2\pi)^{\frac{k}{2}}|\Sigma_i|^{\frac{1}{2}}} e^{-\frac{1}{2}(\theta_j-\mu_i)^t \sum_i^{-1}(\theta_j-\mu_i)} \tag{6.2}$$

where k is the dimension of the feature space.

Because we are interested in the posterior probability $p(\omega_i|\theta_j)$, the probability of a cluster ω_i can be expressed using Bayesian inference by

$$p(\omega_i|\theta_j) = \frac{p(\theta_j|\omega_i)p(\omega_i)}{\sum_{i=1}^{r} p(\theta_j|\omega_i)p(\omega_i)} \tag{6.3}$$

where $p(\omega_i)$ is the prior probability of the ith cluster and r the total number of clusters.

Because the generated vocabulary of signal classes is valid for the entire dataset, images can be clustered together, performing the so called across-image classification.

6.2 *K*-means: Generalized Lloyd Algorithm

Clustering divides the data space into subspaces (classes), where all the points belonging to a class are described by a representative point. The task consists of finding the space partitioning, and the best points that minimize a cost function, i.e. Euclidean distance, mean square error or distortion. Classes are modeled by a mixture of gaussians aiming at parameter estimation as mean or variance, that better represent the data. The generalized Lloyd Algorithm, called k-means in data analysis literature, is one of the most commonly used algorithm for mixture density estimation.

The following pseudo-code corresponds to the k-means algorithm:

```
Input: number of $k$ clusters, dataset consisting of n samples
Output: $k$ clusters which minimize the cost function $E$
Method:
    1. randomly guess $k$ cluster center locations
    2. do
        (a) each datapoint finds out the closest center
        (b) each center finds the centroid of the cluster
        (c) center = centroid
    3. until no significant change in $E$
```

6.3 Dyadic *k*-means

To generate classes in the feature space from a set of estimated parameters, the dyadic k-means classification method is applied. It is a modified and enhanced version of the traditional k-means clustering tool. The algorithm starts with a random selection of two clusters ω_1 and ω_2 with centers μ_1 and μ_2, respectively. Then, each point of the dataset θ_j is associated with one of the two initial clusters. To achieve an optimization of the error-cost function E, the initial state is updated like in k-means, and if convergence is reached, each of the clusters ω_1 and ω_2 is divided into two new ones. E is optimized for the new configuration in a way that points from one cluster can only fall into one of the split clusters. This process is repeated, until E or the cluster membership does not change significantly or until the defined number of clusters k is reached. Both, splitting each cluster into two new ones and optimizing this configuration is one level in the algorithm.

The error-cost function E measures the similarity by some criterion between two pixels and influences strongly in the cluster shape. The proposed criteria are:

- Euclidean distance, that measures the distance d between center of the cluster μ_i and the considered point of the dataset θ_i. This criterion is computed as:

$$d = ||\theta_j - \mu_i||^2 \tag{6.4}$$

- Spectral angular distance, that measures the angle α formed between the cluster center μ_i and each point in the feature space θ_j. It is computed as:

$$\alpha = \frac{\theta_j * \mu_i}{||\theta_j|| \cdot ||\mu_i||} \tag{6.5}$$

In general, the spectral angular distance groups pixels belonging to the same feature independently of radiometrical changes which affect the amplitudes. These are mainly produced by atmospheric effects and nature of the relief affecting strongly in some bands of the electromagnetic spectrum. The spectral angular distance will be able to classify a feature, avoiding illumination conditions, while the results of applying the euclidean distance will be strongly affected by radiances. But using spectral angular distance as cost function requires a very high computation time, due to the trigonometric functions. This is a critical factor to classify the feature space, and finally the similarity measurement based on euclidean distance is often chosen.

The pseudo-code of this variation of k-means algorithm is as follows:

```
Input: number of $k = 2^q$ clusters (q=1,...,n)

Output: $k$ clusters which minimize the cost function $E$

Method:

    1. initialization: select 2 clusters $\omega_1$ and $\omega_2$ and obtain their cluster centers $\mu_1$
       and $\mu_2$, respectively
    2. for i=1, ..., q-1
        (a) split each cluster $\omega_1$, ...,$\omega_{2^i}$ into two new ones. This is performed applying
            $k$-means for $k = 2$ to each cluster.
        (b) recompute cluster centers $\mu_1$, ...,$\mu_{2^{(i+1)}}$
        (c) classify the samples of clusters $\omega_1$, ..., $\omega_{2^i}$ separately to one of the two new
            (split) ones
    3. until no significant change in $E$ or in $\mu_1$, ..., $\mu_{2^{(i+1)}}$ or number of clusters $k = 2^q$
       is reached.
```

Fig. 6.1 plots two iterations of dyadic k-means approach for $k = 2$ and $k = 4$, showing, how a cluster splits in two new ones at each iteration, and Fig. 6.2 displays an example of application of dyadic k-means method using the spectral angular distance as cost function E. The image has been provided by the Meris sensor and covers Italy and north African areas. The algorithm is able to classify the features in three main classes: clouds, represented in white, see water, in blue, and land, displayed in dark.

The reason for modifying k-means is to obtain an enhanced and accelerated tool for clustering huge datasets. The most time-consuming computation in k-means is the spectral angular distance calculation in the multi-dimensional feature space, because it must be performed to all cluster centers for each point. The time required for I/O is less important in comparison to the distance computation but increases linearly with the amount of data. Whereas the computational complexity for k-means is $O(kn)$ with k clusters and n samples, it is *O(n log k)* for the dyadic algorithm. Especially for a large

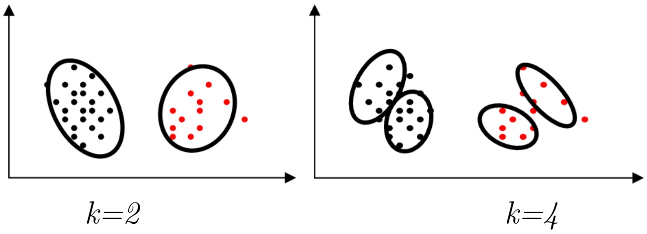


Figure 6.1: Dyadic k-means algorithm: Example of how pixels are grouped forming clusters for k=2 and k=4 number of clusters.

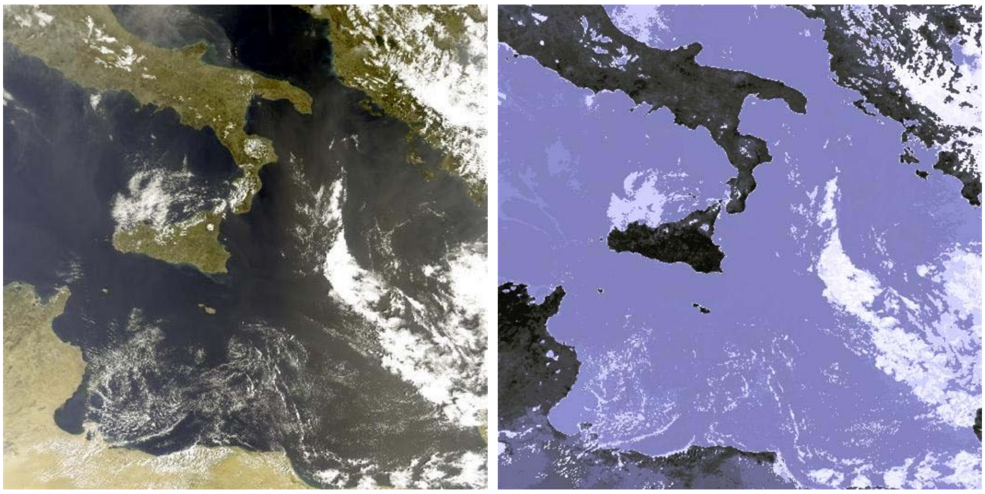


Figure 6.2: Images from left to right. Meris original image; result of dyadic k-means using the spectral angular distance as similarity measure. The classified features are: clouds, represented in white, see water, in blue, and land, displayed in dark.

number of clusters the algorithm has proven to be very efficient. From the processing point of view, it is consequent to speed up the algorithm as much as possible. Since the power of a single processor is limited to a few GHz, it is challenging to run the clustering tool on a multi-processor machine. For the dyadic k-means, the iterative application of k-means enables the direct efficiently utilization of a single instruction multiple data architecture.

K-means and its variations seem to be an efficient method for compressing large volumes of data. However, this is subject to two main constraints: the number of clusters, needed to classify the data, and how to proceed if the archive enlarges. To answer the first question, 128 clusters seems to be a compromise between detailed data representation and computational cost. To cope with the second problem incremental clustering algorithms are being developed.

6.4 Conclusions

In this chapter, we have learned, how feature space is compressed in an unsupervised way, building a vocabulary of signal classes. With the variant of the classical k-means algorithm, an efficient mining tool is developed to reduce the amount of available information. This process may be expensive in terms of computation time, but could speeded up by parallel computing. Once classes are built, they will be the catalogue entries for the database.

7

Optimization of Feature Extraction based on Rate Distortion Theory

As previously mentioned, the cluster space, obtained for the extracted features can be considered as a code building an abstract vocabulary of signal classes for each feature. For coding, Rate Distortion (RD) theory is one of the most successful ideas in information transmission theory. Coding tries to get an optimal digital representation of the signal, leading to a certain compression degree. Through compression, the processed signal is distorted. RD theory ensures a balance between an acceptable distortion (error) and a satisfactory compression rate. In this chapter, we will see, how to estimate the optimal analyzing window in a textural parameter extraction algorithm by RD analysis of the cluster space.

Sec. 7.1 proposes a theoretical framework based on RD theory for the estimation of parameters. Then, we present a practical application of RD for selecting the optimum size of the window. We begin, describing the evaluation procedure in Sec. 7.2, and in Sec. 7.3, we present the results of the method for two datasets coming from different sensors.

7.1 Rate Distortion Theory

The computational cost of extracting features from the image sometimes becomes very expensive. In other occasions, the user is more interested in computing local features than global ones, or the algorithm itself requires inspecting the image by small windows. In those situations, the optimum size of the analyzing window becomes an important factor that not only depends on the data, but also on the feature itself.

For our study, we take the estimation of texture parameters as an example. In Sec. 5.2.2, we learned, how to extract contextual spatial information from EO data by using parametric models. In this frame, the estimated parameter vector represents the relationship between a pixel and its neighborhood obtained by computing the Gibbs local energy function over a fixed analyzing window. The size of this window strongly influences the accuracy of the texture parameters. The larger the analyzing window, the more diversity of spatial information is collected inside the window. Therefore, in order to extract the maximum amount of information from data, and consequently, to define a good model, the optimal size for the analyzing window must be estimated.

The autobinomial model, computed for estimating the texture parameters, provides two quality measurements: the evidence and the variance. The former is often used for model selection, and consequently, to measure the model quality; the latter depends on the number of samples considered for the estimation, and therefore, will be useful for deriving the optimal size of the analyzing window.

Obtaining good quality of the parameters requires a small variance of the estimator being limited by the Cramér-Rao Lower Bound in large volume of data (Eq. 4.33). However, in chapter 4 we explained that if we have to consider a parameter vector instead a single parameter, the covariance matrix has to be minimized being limited by the Fisher information matrix $J(x)$ (Eq. 4.34). Thus, we can say that trying to minimize the variance is equivalent to maximize Fisher's information:

$$\min \sigma_x^2 \Rightarrow \max J(x) \tag{7.1}$$

On the other side, Eq. 4.32 defines the mutual information as

$$I(X;Y) = H(X) - H(X|Y) \tag{7.2}$$

and knowing the relationship between Fisher information and the conditional differential entropy given by *Bruijn's identity* (Eq. 4.37), we can consider the following hypothesis

$$H(X|Y) = J(x). \tag{7.3}$$

Substituting in Eq. 7.2 and aiming at maximizing Fisher's information, we infer that trying to maximize $H(X|Y)$ the mutual information has to be minimized

$$\max H(X|Y) \Rightarrow \min I(X;Y) \tag{7.4}$$

Finally, the RD theorem (Eq. 4.41) determines the minimal amount of information to transmit, ensuring that the output signal does not exceed a given distortion D from the input signal. Under this hypothesis, we can conclude that in order to obtain a good quality of parameters, a small variance is required, and this can be reached by computing the RD of data

$$\min \sigma_x^2 \Rightarrow \max J(x) \Rightarrow \min I(X;Y) = R(D) \tag{7.5}$$

In this section, we have introduced, how the analyzing window size affects the accuracy of the textural information extraction algorithm. Applying RD theory, we can estimate the optimum window size for specific data. In the following, we present the evaluation procedure and the results of applying the theory to a real case.

7.2 Evaluation Protocol

For the evaluation of the approach and estimation of the optimum analyzing window for extracting texture parameters, the information hierarchy presented in chapter 3 is followed. First, features (level 1) and meta features (level 2) are extracted from the original image (level 0). The hierarchy continues by running a classification or compression algorithm of the feature space to reach level 3 of information formed by clusters. Finally, RD theory is applied over cluster space to optimize the size of the analyzing window.

1. Feature Extraction

   Before extracting texture parameters, we have to select the band that best approximates a Gaussian distribution. For the band selection, we can proceed in two ways: heuristically, so that we select the band containing the most details (in general, it is the blue channel); or computing the PCA over all bands and choosing the first component. The selected band is cut to produce squared tiles manageable by the method. Then, texture parameters are estimated through GRF algorithm and norm, variance, evidence and mean are extracted.

2. Clustering

   Performing the dyadic k-means approach, the information represented in the feature space is grouped by forming a set of clusters.

3. Distortion

   Using a distortion, measure the clusters of feature space can be analyzed. Distortion $d = d(x, \hat{x})$ is the cost measure of representing the symbol x by $\hat{x}$. Through dyadic k-means algorithm, data are modeled by the center or by the covariance matrix of the cluster. The results of measuring distortion over both models coincide, therefore only the center of the cluster will be considered.

Distortion measure defined on a pixel to pixel basis, that is, it measures the distortion between one single pixel and its representation by the center of the cluster. But, to compute the local distortion, the average of all distortions in a cluster must be measured. Thus, this local distortion function is computed as:

$$D = E[d(x^{M_i}, \hat{x}^{M_i})] = \frac{1}{M_i} \sum_{j=1}^{M_i} d(x_j \in c_i, \hat{x}_i) \tag{7.6}$$

where M_i is the number of pixels belonging to the class c_i, x_j is the j-th pixel of i-th class, and $\hat{x}_i$ represents the center of the class c_i.

To get the global distortion instead the local one, all clusters in the feature space must be considered. Hence, in case of discrete random variable and choosing the mean squared error as distortion function, the global distortion will be obtained by

$$D = \frac{1}{M} \sum_{i=1}^{n} \sum_{j=1}^{M_i} |x_{i,j} - \hat{x}_i|^2 \tag{7.7}$$

with M the total number of pixels in the feature space, n number of clusters and $x_{i,j}$ the j-th pixel of i-th class.

4. Theoretical Bound

The obtained results will be compared with the distortion theoretical bound for a mixture of Gaussians. This bound will be computed as

$$D = \sum_{i=1}^{m} \mu_i \cdot D_i(b_i) \tag{7.8}$$

with

$$D_i(b_i) = n \cdot K \cdot \Lambda_i \cdot 2^{-2\frac{b_i}{n}} \tag{7.9}$$

and

$$\Lambda_i = \left(\prod_{j=1}^{n} \lambda_{ij} \right)^{\frac{1}{n}} \quad \text{for} \quad i = 1, \ldots, m \tag{7.10}$$

where μ_i is the weight, b_i the number of bits assigned to i-th cluster, m the number of cluster, n the space dimension, λ_{ij} the j-th variance of cluster i, and K a constant which is approximately $\frac{\pi\sqrt{3}}{2}$ for Gaussian sources.

Fig. 7.1 shows the block diagram of the evaluation method.

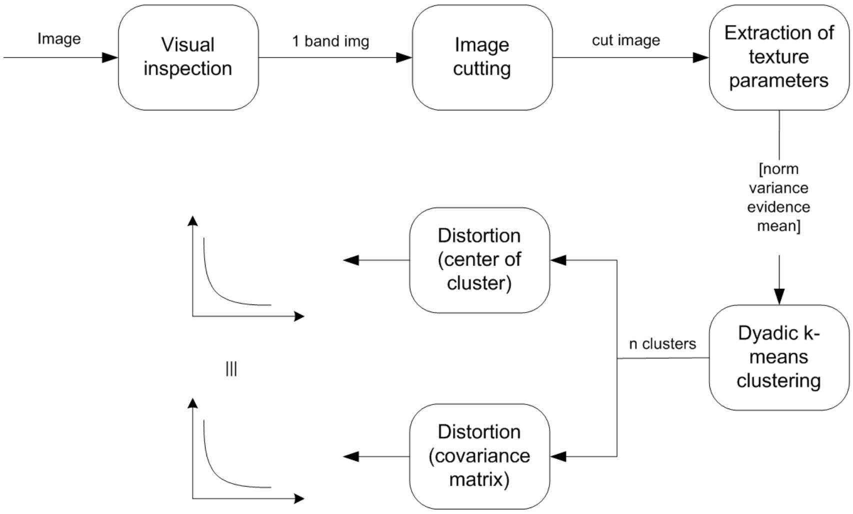


Image
Visual inspection
1 band img
Image cutting
cut image
Extraction of texture parameters
[norm variance evidence mean]
Dyadic k-means clustering
n clusters
Distortion (center of cluster)
Distortion (covariance matrix)
|||

Figure 7.1: Block diagram of the evaluation approach.

7.3 Method Assessment

For the evaluation of the proposed approach, a dataset of homogeneous images from the Brodatz texture archive is selected. With this dataset, the theoretical behavior of the method can be analyzed for a later comparison with real data to see the effectiveness of the research.

a) Homogeneous Data

For the study, twelve images from the Brodatz texture database [69] are selected. Images consist of one band of 640 pixels height and 640 pixels width. Fig. 7.2 displays six of them as example.

To extract texture parameters from this dataset, the algorithm is run with analyzing window sizes of 10 × 10, 20 × 20, 30 × 30, 40 × 40, 50 × 50 and 60 × 60 pixels. Fig. 7.3 plots the behavior of the norm, variance and evidence in the feature space for different size of the analyzing window. In this theoretical scenario we can analyze, how the variance of data decreases when the analyzing window size increases.

After inspecting the behavior of the variance in the feature space, the mean squared error distortion is computed to test our approach. Fig. 7.4 displays the distortion curves for different window size and number of clusters. We observe, how distortion values decrease when the size of analyzing window increases, ensuring the expected theoretical behavior.

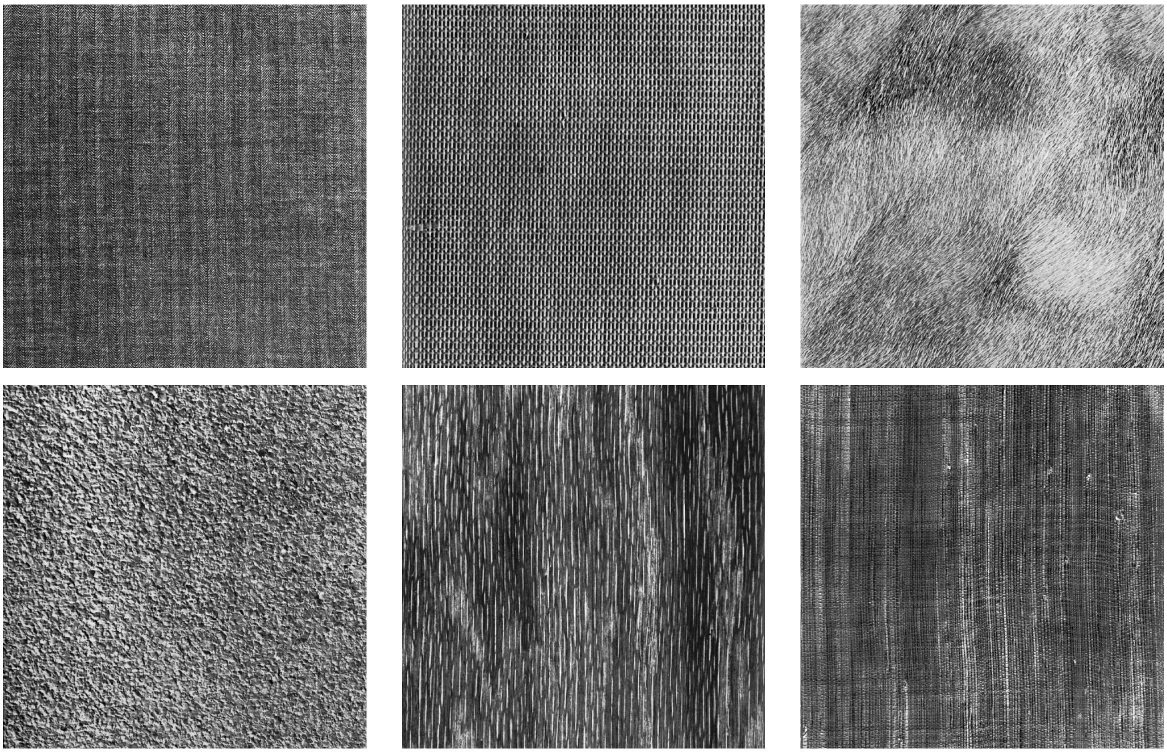
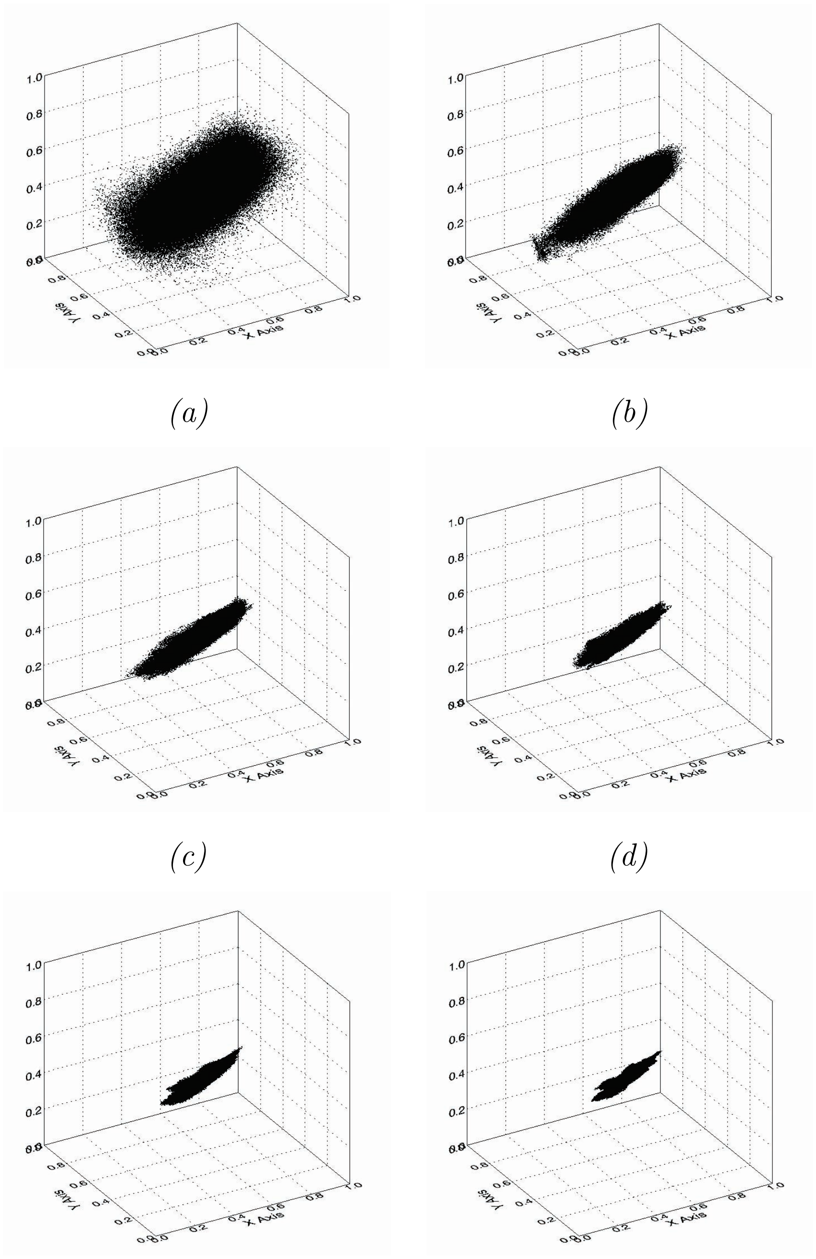


Figure 7.2: Text data from Brodatz texture database.



(a) *(b)*

(c) *(d)*

(e) *(f)*

Figure 7.3: Behavior of the estimated parameters (x-axis the norm, y-axis the variance and z-axis the evidence) in the feature space for analyzing window of size 10 × 10 *(a)*, 20 × 20 *(b)*, 30 × 30 *(c)*, 40 × 40 *(d)*, 50 × 50 *(e)* and 60 × 60 *(f)*. The variance becomes smaller when increasing the size of the analyzing window.

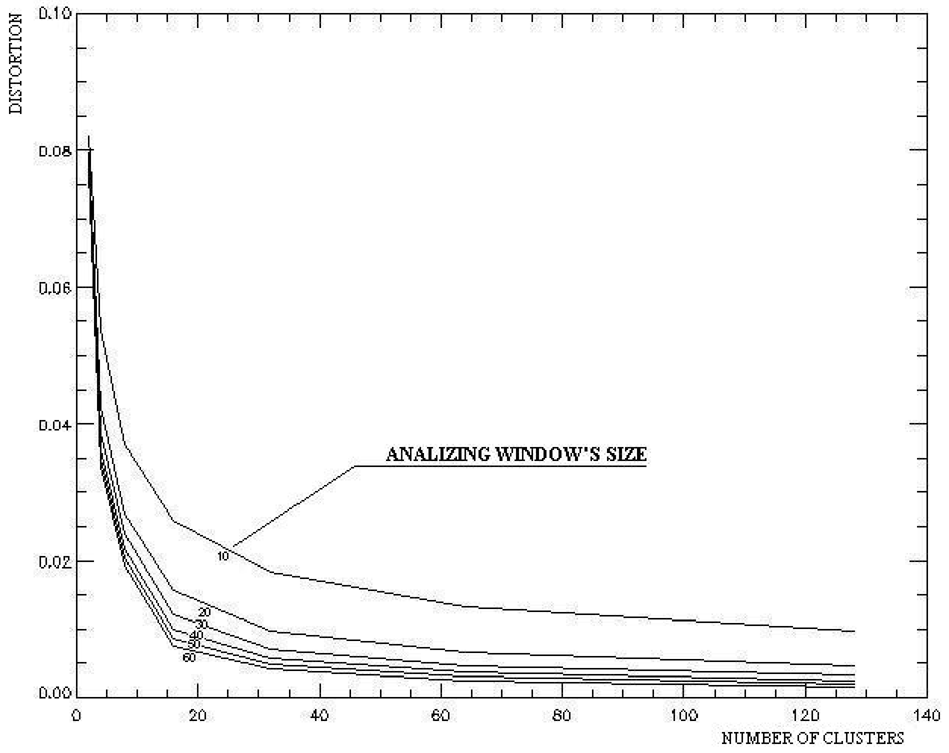


Figure 7.4: Distortion curves in the stationary case for different analyzing window size.

b) Heterogeneous Data

Distinctly from homogeneous data and continuing analyzing texture parameters, real data are usually non-stationary and present spatial diversity content. To evaluate the behavior of RD and therefore the optimum analyzing window size, two datasets are selected. The first one corresponds to a Landsat image, covering the area of Mozambique with 30m resolution. The original image was cut into squared images of 466 × 466 pixels to compute texture. The second one is taken from the Daedalus sensor of 0.8m resolution and includes the Oberpfaffenhofen (Wessling, Germany) town and surroundings. In this case, it was cut into square tiles of 605 × 605 pixels. Fig. 7.5 displays band five of the Landsat image and band seven of the Daedalus sensor.

The distortion curves for window sizes of 10 × 10, 20 × 20, 30 × 30, 40 × 40, 50 × 50 and 60 × 60 pixels are plotted in Fig. 7.6 and Fig. 7.7 for Landsat and Daedalus, respectively. Analyzing Landsat image results, distortion values decrease when the window size increases (the distortion curve plotted between the 20 and the 30 ones corresponds to the computation of distortion with an analyzing window of 40 × 40 pixels). But when the minimum distortion is reached, values begin to increase again, being 30 for the Landsat image and 20 for the Daedalus dataset. This occurs when

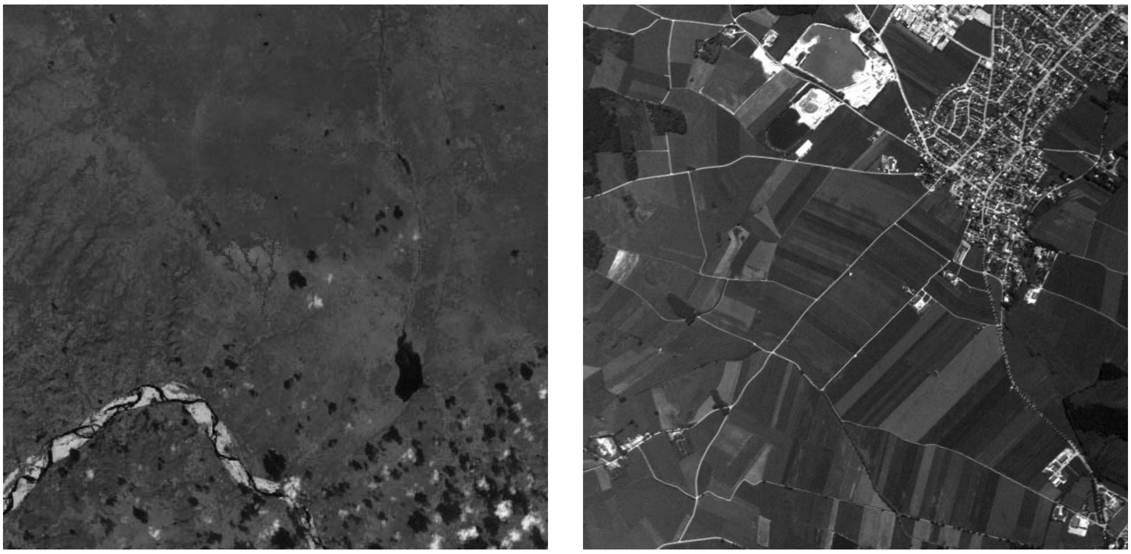


Figure 7.5: Images from left to right. Band five of Landsat image covering Mozambique area; band seven of Daedalus image taken from Oberpfaffenhofen (Wessling, Germany) town and surroundings.

spatial diversity starts to be included in the window, thus increasing the cluster variance. Thus, before including spatial diversity in the window, the optimum window size is obtained. Tab. 7.1 and Tab. 7.2 contain the distortion values D (Eq. 7.7), for different sizes of the window computed for 4, 8, 16, 32, 64 and 128 clusters. The theoretical bound (TH), evaluated for the minimum distortion is displayed in both tables and figures (lowest curve). This is obtained, assuming gaussianity on data, so that, if this hypothesis is not fulfilled, a difference between the minimum distortion and the theoretical bound curve will appear.

Why is the optimum size of the window different for both datasets? Distortion values depend on the image content and on the extracted feature. When different features are included in the same window, the distortion will be higher than when the analyzing window contains only one feature. The Daedalus image contains more details, hence more variety of textures than the Landsat one, thus, the optimum size of the window is reached earlier.

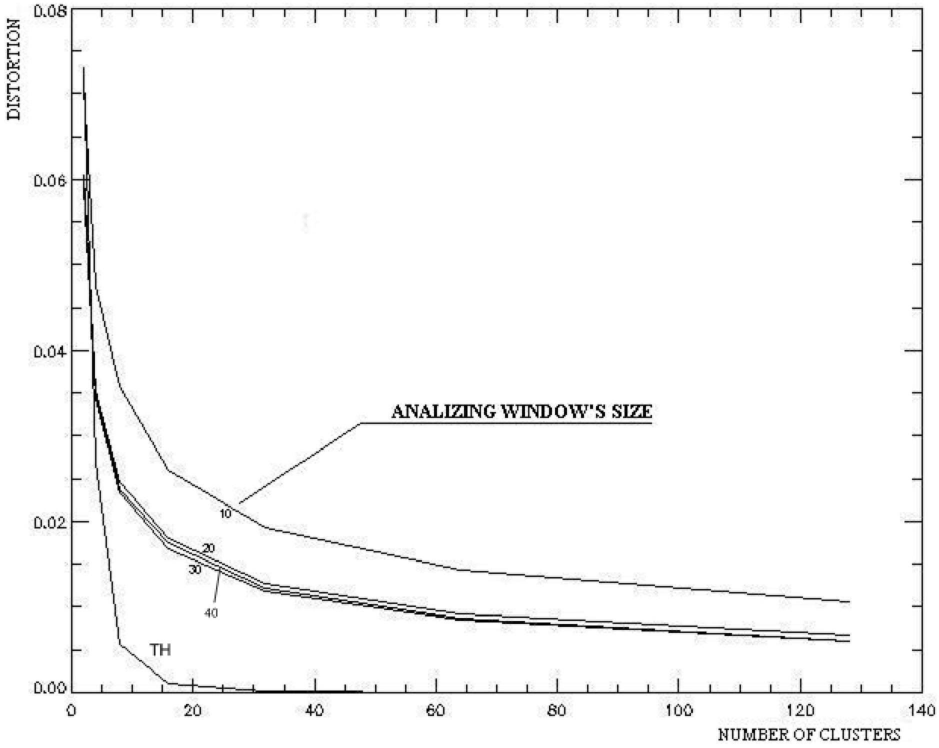


Figure 7.6: Distortion curves on Landsat data. The minimum distortion is obtained for the analyzing window of size 30 × 30 pixels. (Curve between 20 and 30 corresponds to the distortion for an analyzing window of 40 × 40 pixels). The lowest curve plots the theoretical bound (TH) computed for a window of 30 × 30 pixels.

WIN	NUMBER OF CLUSTERS					
SIZE	**4**	**8**	**16**	**32**	**64**	**128**
10	0.0476070	0.0357953	0.0260905	0.0193684	0.0143299	0.0105636
20	0.0351714	0.0245875	0.0180811	0.0126777	0.00918795	0.00674138
30	0.0343124	0.0234149	0.0168479	0.0119416	0.00851491	0.00607247
40	0.0350622	0.0236877	0.0174588	0.0121431	0.00861382	0.00608281
50	0.0359135	0.0239821	0.0179190	0.0121242	0.00849565	0.00581971
60	0.0364942	0.0240030	0.0172197	0.0119933	0.00822319	0.00571007
TH	0.022823207	0.0052907453	0.0015284197	0.00029337619	5.6862104e-005	1.5112676e-005

Table 7.1: Distortion values D (Eq. 7.7), of the Landsat image for different sizes of the analyzing window, computed for 4, 8, 16, 32, 64 and 128 clusters: the optimum size of the analyzing window is 30 × 30 and is marked in red; in blue displays the theoretical bound (TH), computed for 30 × 30 window size.

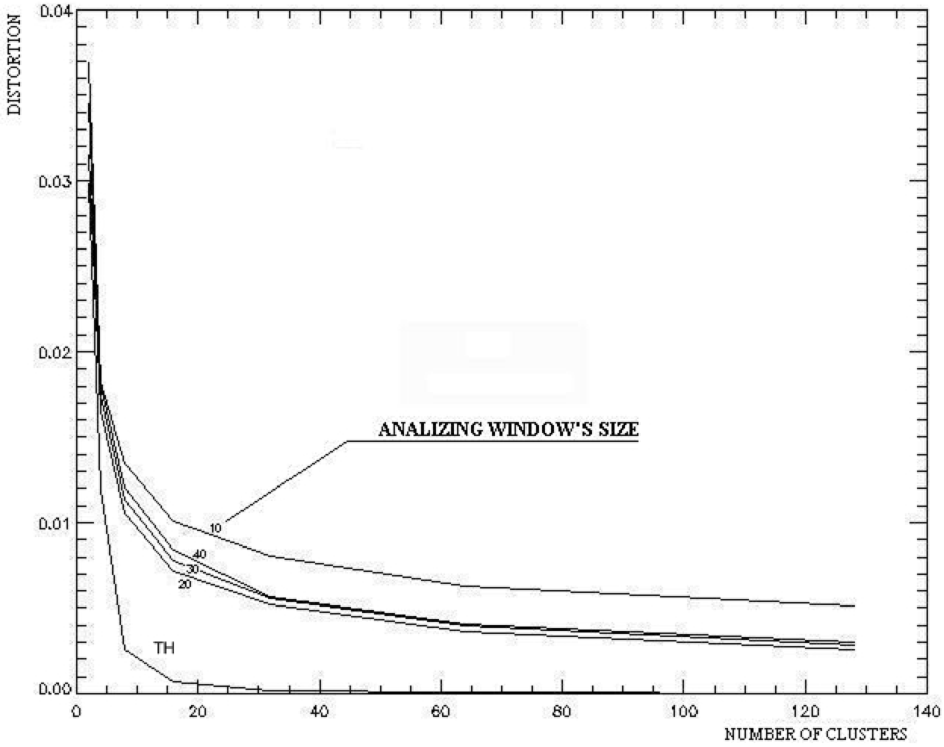


Figure 7.7: Distortion curves of Daedalus data. The minimum distortion is obtained for the analyzing window of size 20 × 20 pixels. The curve with no number plots the theoretical bound (TH), computed for a window of 20 × 20 pixels.

WIN	NUMBER OF CLUSTERS					
SIZE	4	8	16	32	64	128
10	0.0181957	0.0134270	0.0100542	0.00802099	0.00631736	0.00510492
20	0.0165385	0.0105246	0.00716277	0.00520682	0.00362008	0.00259755
30	0.0174057	0.0113578	0.00775441	0.00555624	0.00396684	0.00283382
40	0.0180810	0.0120574	0.00839687	0.00569983	0.00410801	0.00301704
50	0.0187720	0.0125151	0.00829058	0.00581014	0.00403086	0.00276101
60	0.0193045	0.0128178	0.00857343	0.00582683	0.00402654	0.00281745
TH	0.0075298511	0.0016249744	0.00057176973	0.00014384566	-	-

Table 7.2: Distortion values D (Eq. 7.7), of the Daedalus image for different size of the analyzing window, computed for 4, 8, 16, 32, 64 and 128 clusters: the optimum size of the analyzing window is 20 × 20, is marked in red; in blue we display the theoretical bound (TH), computed for 20 × 20 window size.

7.4 Conclusions

In this chapter, we have seen, how RD theory builds the theoretical background for the estimation of parameters, and in general, it can be useful in the image processing output quality evaluation. Until now there were no methods, except visual inspection, to assess the goodness of the extracted parameters in Image Information Mining on large data archives. The approach, presented here, describes the parameter estimation as a coding, depending on the size of the analyzing window and classification as compression. The observed behavior is that increasing the size of the analyzing window, distortion decreases until spatial diversity is included in the window. Thus, the smaller the optimum window, the more diversity on data.

8

Interactive Learning

There is a strong need to build up applications that help the user in image interpretation, permitting to query the archives in a content based mode. In this chapter we propose a synergy between stochastic modeling, knowledge discovery, and semantic representation. To do that, we associate semantic labels to a combination of primitive image features. The user-defined semantic image content interpretation is linked with Bayesian networks to the results of a completely unsupervised classification.

The proposed probabilistic approach can be applied for dynamic search of new features on the ingested images, for content based image selection, for information discovery and for scene understanding. It provides the solution for defining and searching not only pair wise disjoint cover types (e.g., "lake" and "not lake"), but different features, contained in the same image (e.g., "lake" and "mountain" and "cloud" and "not lake, not mountain, not cloud"). This new paradigm for the interaction with EO archives can provide several applications for users, coming from different domains, as change detection, agricultural field classification, environment monitoring, atmosphere effects or urbanization.

Thus, Sec. 8.1 begins, describing, how the interactive learning is performed, and how the acquired knowledge is stored in a catalogue in form of semantic labels. Once the user has given training examples on an image, the system can retrieve images with similar features from the archive by performing a searching task. This probabilistic search is detailed in Sec. 8.2. In Sec. 8.3 an extension of the system is presented, detailing how a novel module, called multiple classifier, enables the system to aggregate labels on the same image obtaining several thematic maps.

8.1 Interactive Learning

In chapter 6, we have applied an unsupervised clustering to obtain characteristic signal classes from the extracted features. This vocabulary of classes must be somehow linked to the user specific interest creating semantic labels. Due to the fact, that Bayesian networks provide a probabilistic classification of image content, they are consistent and appropriate to be used in this context. They are also suitable to retrieve content based images from the archive.

We try to link the subjective elements L, called semantic labels or cover-types, to the objective signal classes ω_i through learning probabilities $p(\omega_i|L)$. This means, that the cover-types are modeled in terms of image properties and not in terms of objects. Therefore, a cover-type will be linked with one or more signal classes depending on the nature of the image, i.e. a label 'forest', will be linked with 'texture' and 'spectral signature', and the intersection of both will describe the semantic label. Knowing this, a signal class ω_{jk} is decomposed into

$$\omega_{jk} = \omega_{sp,j} \cap \omega_{tx,k} \tag{8.1}$$

representing $\omega_{sp,j}$ the spectral signal class j, and $\omega_{tx,k}$ the texture signal class k. Any other combination of signal models including signal models coming from different sensors can be used. The data fusion procedure is performed while computing the learning probabilities $p(\omega_i|L)$ under the assumption of conditional independence between models. Then, $p(\omega_{jk}|L)$ is computed as

$$p(\omega_{jk}|L) = p(\omega_{sp,j}|L) \cdot p(\omega_{tx,k}|L) \tag{8.2}$$

In the following, we will use ω_i to represent ω_{jk}, knowing that we are referring to a combination of signal classes.

But before making inference from the image data D to the label L, the system has to learn the probabilistic link $p(\omega_i|L)$ based on training examples. Bayesian Inference provides an elegant way of learning probabilities from data [44]. Considering a set of user-supplied training data T with $T = \{N_1, ..., N_r\}$ and N_i the number of occurrences of ω_i in T, and the parameter vector $\phi = \{\phi_1, ..., \phi_r\}$ as a model of a set of probabilities $p(\omega_i|L, \phi) = \phi_i$, where r indicates the number of signal classes ω_i, we can model our problem as a Dirichlet distribution by

$$p(\phi_1, \phi_2, ..., \phi_{r-1}; \alpha_1, \alpha_2, ..., \alpha_r) = \frac{1}{B(\alpha)} \Pi_{i=1}^{r} \phi_i^{\alpha_i - 1} \tag{8.3}$$

where $B(\alpha)$ is the multinomial beta function, and $\alpha = \alpha_1, ..., \alpha_r$ is the hyperparameter vector. This vector represents the user interaction, providing positive and negative examples. Thus, two hyperparamenter vectors are created: one for positive and another one for negative examples, and are stored in the database for a future probability search. The initial state of the hyperparameters is $\alpha_0 = \{1, ..., 1\}$ and with the first interaction, they are updated by $\alpha_i = 1 + N_i$.

Then, let $\phi = (\phi_1, ..., \phi_r) \sim Dir(\alpha)$, meaning that the first $r - 1$ components present the densities of Eq. 8.3, and $\phi_r = 1 - \phi_1 - ... - \phi_{r-1}$, and knowing that in Bayesian statistics, the Dirichlet distribution is the conjugate prior of the multinomial distribution, then the posterior probability is defined as

$$p(\phi|T) \sim Dir(1 + N_1, ..., 1 + N_r) = Dir(\alpha + T) \tag{8.4}$$

where T is the training data set. We will used this posterior probability to estimate the hidden parameters, ϕ. Fig. 8.1 explains the previous notations in a Unified Modeling Language (UML) sequence diagram. This diagram displays the interaction between the user and the system for training a semantic label, L, showing the parameters that are computed or updated in each step: A positive or negative training sample (element N_i of T) is given by the user to the client (learning applet). A new posterior map is computed and the hyperparameter vector, α, is updated. Finally, the user defines a new label based on the posterior map and it is stored in the database, together with its hyperparameter vector.

For the second and following training actions, we observe another training set $T^{'}$ considered independent on T. In this case, the hyperparameters are updated by

$$p(\phi|T^{'}) \sim Dir(\alpha_1 + N_1^{'}, ..., \alpha_r + N_r^{'}) = Dir(\alpha^{'}) \tag{8.5}$$

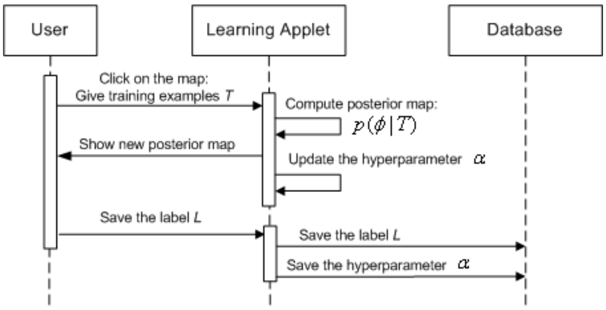


Figure 8.1: Explanation for the notations used in the interactive learning phase through a UML sequence diagram: T denotes the training examples provided by the user; $p(\phi|T)$ is the a posteriori probabilities computed for building the posterior map; α defines the hyperparameter vector; L denotes the user defined label.

where

$$\alpha_i^{'} = \alpha_i + N_i^{'} \tag{8.6}$$

Once the hyperparameters are defined, the initial probabilities can be finally calculated as the expectation over all possible values of ϕ_i (number of positive examples belonging to cluster i over all positive examples in the image) as

$$p(\omega_i|L,T) = E[\phi_i] = \frac{\alpha_i}{\sum_i \alpha_i} \tag{8.7}$$

and for a second and the following training samples, these probabilities are updated by

$$p(\omega_i|L,T) = \frac{\alpha_i^{'}}{\sum_i \alpha_i^{'}} \tag{8.8}$$

We have seen, how to infer the user-defined semantic labels L (level 4) from the image data D (level 0). Now, we need to provide the user a feedback of accuracy of the semantic cover type. This feedback is given by a visual map, called posterior map, computed as

$$p(L|\omega_i, D) = p(L) \sum_i \frac{p(\omega_i|L)p(\omega_i|D)}{p(\omega_i)} \tag{8.9}$$

where $p(L)$ is the prior probability of the semantic labels L, $p(\omega_i|L)$ the stochastic link provided by Eq. 8.7, $p(\omega_i|D)$ the posterior probability obtained through unsupervised classification, assuming that the signal features of semantic labels are fully represented by ω_i, and $p(\omega_i)$ the prior of signal classes defined by

$$p(\omega_i) = p(\omega_i|L)p(L) + p(\omega_i|\neg L)p(\neg L) = \sum_v p(\omega_i|L_v)p(L_v) \tag{8.10}$$

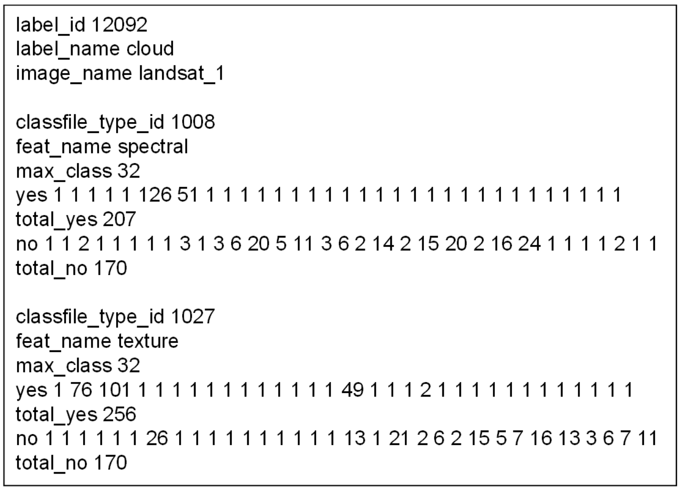
Each time, the user gives a training sample on the image, positive or negative, the probabilities $p(\omega_i|L)$ are calculated and the posterior probabilities $p(L|\omega_i)$ are updated, providing a quick feedback to the user. If this posterior probability is over a certain threshold, the training example is classified as label L, otherwise as label $\neg L$. Computing this pixel based method over the entire image, the posterior map is drawn.

Once a semantic label is defined, its associated hyperparameter vector is stored in the database in a binary large object (blob) format. A blob is a collection of binary data, stored as a single entity in a database management system. This file contains the following information:

- label_id: each time a new cover-type is created, a new identifier is provided.
- label_name: semantic concept given by the user.
- image_name: name of the image where the label has been defined.
- classfile_type_id: mining model identifier.
- feat_name: mining model name.
- max_class: total number of clusters in the mining model.
- yes: hyperparameter positive examples for the selected model (α_i). Each value corresponds to one cluster in the model, representing the positive examples given by the user in each cluster in the label.
- total_yes: sum of the positive examples ($\sum_i \alpha_i$).
- no: hyperparameter negative examples for the selected model (α_i). Each value corresponds to one cluster in the model, representing the negative examples given by the user in each cluster in the label.
- total_no: sum of the negative examples ($\sum_i \alpha_i$).

Fig. 8.2 shows an example of the information stored in the database for the semantic label 'cloud' using the spectral and textural information as mining models.

The user owns an interface to choose first, the collection of images to be trained, and second, the mining models. He selects an image in form of a quicklook to play with, and performs the supervised classification. Fig. 8.3 shows examples of performing interactive learning over Landsat TM images covering several areas of Italy. The mining models that have been chosen, are spectral signature and texture at scale zero. In each example, different features are trained: coast line, a kind of land and kind of water on the ocean.



```
label_id 12092
label_name cloud
image_name landsat_1

classfile_type_id 1008
feat_name spectral
max_class 32
yes 1 1 1 1 1 126 51 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
total_yes 207
no 1 1 2 1 1 1 1 1 3 1 3 6 20 5 11 3 6 2 14 2 15 20 2 16 24 1 1 1 1 2 1 1
total_no 170

classfile_type_id 1027
feat_name texture
max_class 32
yes 1 76 101 1 1 1 1 1 1 1 1 1 1 1 1 49 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1
total_yes 256
no 1 1 1 1 1 1 26 1 1 1 1 1 1 1 1 1 1 13 1 21 2 6 2 15 5 7 16 13 3 6 7 11
total_no 170
```

Figure 8.2: Example of information stored about the user defined label 'cloud'.

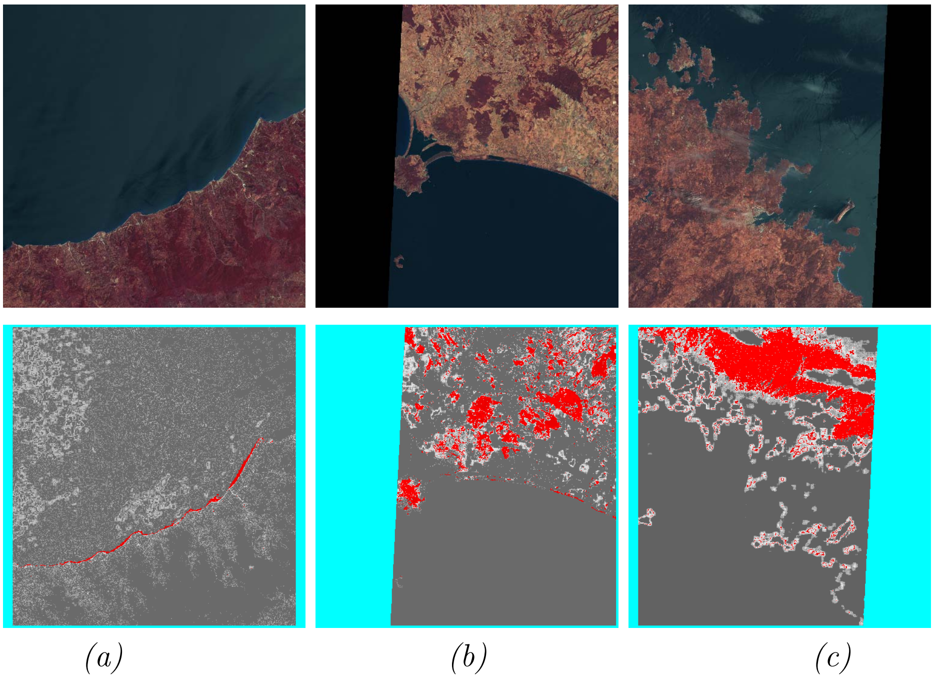


Figure 8.3: Interactive learning using Landsat TM images. *(a)* original image (top) and posterior map (bottom) of coast line; *(b)* original image (top) and posterior map (bottom) searching for a kind of land; *(c)* original image (top) and posterior map (bottom) of a kind of water.

8.1.1 Quality of the Stochastic Link

To measure the quality of the stochastic link $p(\omega_i|L)$, information theory is applied [75][23]. We have to measure the distance between the probability distributions $p(\omega_i|L)$ and $p(\omega_i|\neg L)$ by computing the Kullback Leibler divergence $D(L, \neg L)$ (see chapter 4) as

$$D(L, \neg L) = \sum_{i=1}^{r} (p(\omega_i|L) - p(\omega_i|\neg L)) log \frac{p(\omega_i|L)}{p(\omega_i|\neg L)} \tag{8.11}$$

The divergence D is automatically calculated while training on a label, and returns a relevance feedback to the user during the interactive learning phase, showing the most relevant features in the label.

8.2 Probabilistic Retrieval

In the previous section, the man-machine interaction by defining semantic labels has been presented. These user provided cover-types, however, can be used for a further search for similar content in the archive.

Before explaining the different measures for retrieving imagery, a new probabilistic link must be defined. In a similar way the posterior probability of a semantic label given a particular image is estimated by $p(L|\omega_i, D)$, the posterior probability of L given a collection of images I_ζ can be defined as

$$p(L|I_\zeta) = \sum_i p(L|\omega_i) p(\omega_i|I_\zeta) \tag{8.12}$$

where the following approximation is used

$$p(\omega_i|I_\zeta) \equiv p(\omega_{m,j}, \omega_{n,j}|I_\zeta) \sim p(\omega_{m,j}|I_\zeta) \cdot p(\omega_{n,j}|I_\zeta) \tag{8.13}$$

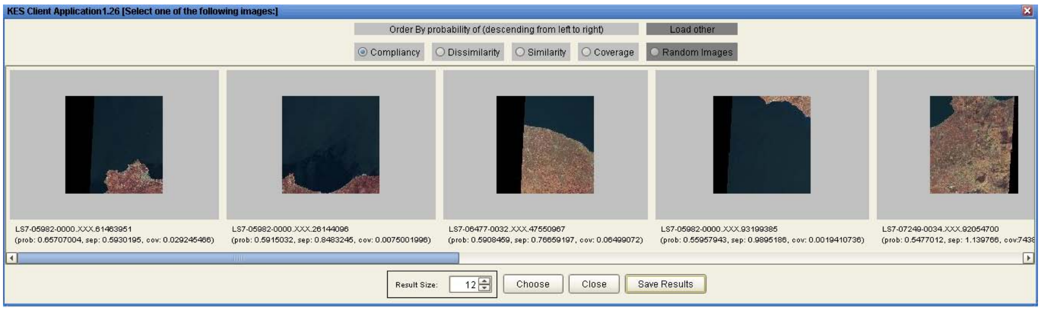
This approximation allows us to estimate $p(L|I_\zeta)$, avoiding the calculation of joint histograms of signal classes of each image in the collection, speeding up the search.

The probability index $p(\omega_{m,j}|I_\zeta)$ represents the jth class of the mth model in the ζth image, which is stored when new data is ingested in the archive together with the classfiles (level 3 in the hierarchy of information). This probability will be used to index signal classes of the images for a fast retrieving process, and to measure the quality of the results.

When searching in the catalogue for similar features, the system computes four types of measurements for retrieving images:

- Compliancy or posterior probability: Measure of how probable an image is of a particular cover-type. This search criterion returns images with the highest probability of containing the same or similar pixels to the positive training examples, defined in the feature.
- Dissimilarity or high separability: Measure of how well L is separated from $\neg L$ in a particular image I_ς. This criterion shows images with the highest distance between the pixel with positive and negative matches according to the feature. They are the images containing the most counter examples from the original one.
- Similarity or low separability: This returns images with the lowest distance between the pixel with positive and negative matches according to the feature.
- Coverage: It measures the approximate percentage of the image containing only the desired cover-type. This criterion shows the images whose coverage matches better with the provided examples.

Continuing the example of Fig. 8.3, searching for coast lines in Landsat TM catalogue, Fig. 8.4 displays images with the highest probability of containing coast lines, Fig. 8.5 images with the highest probability of not containing coast lines, and Fig. 8.6 images where coast lines cover the highest percentage of the image.


KES Client Application1.26 [Select one of the following images:]
Load other
Choose
Close
Save Results

Order By probability of (descending from left to right)

● Compliancy ○ Dissimilarity ○ Similarity ○ Coverage ○ Random Images

LS7-05982-0000.XXX.61463951
(prob: 0.65707004, sep: 0.5930195, cov: 0.029245466)

LS7-05982-0000.XXX.26144096
(prob: 0.5915032, sep: 0.8483245, cov: 0.0075001996)

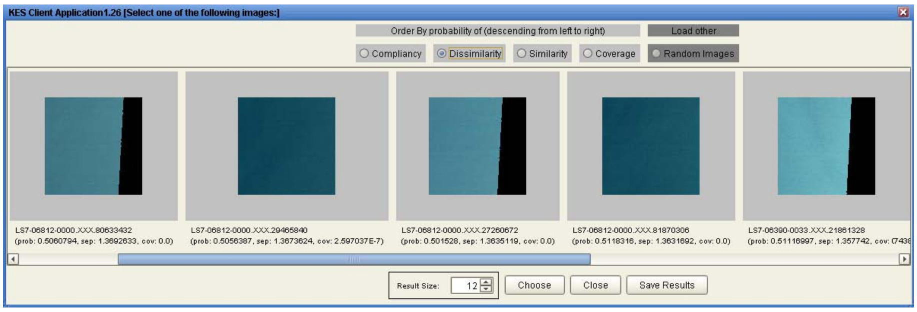
LS7-06477-0032.XXX.47550967
(prob: 0.5908459, sep: 0.76659197, cov: 0.06499072)

LS7-05982-0000.XXX.93199385
(prob: 0.55957943, sep: 0.9895186, cov: 0.0019410736)

LS7-07249-0034.XXX.92054700
(prob: 0.5477012, sep: 1.139766, cov:7438

Result Size: 12

Figure 8.4: Posterior probability of searching for coast lines in the Landsat catalogue. All returned images contain a coast line inside with a high probability.



KES Client Application1.26 [Select one of the following images:]

Order By probability of (descending from left to right)

○ Compliancy ● Dissimilarity ○ Similarity ○ Coverage ○ Random Images

LS7-06812-0000.XXX.80633432
(prob: 0.5060794, sep: 1.3692633, cov: 0.0)

LS7-06812-0000.XXX.29465840
(prob: 0.5056387, sep: 1.3673624, cov: 2.597037E-7)

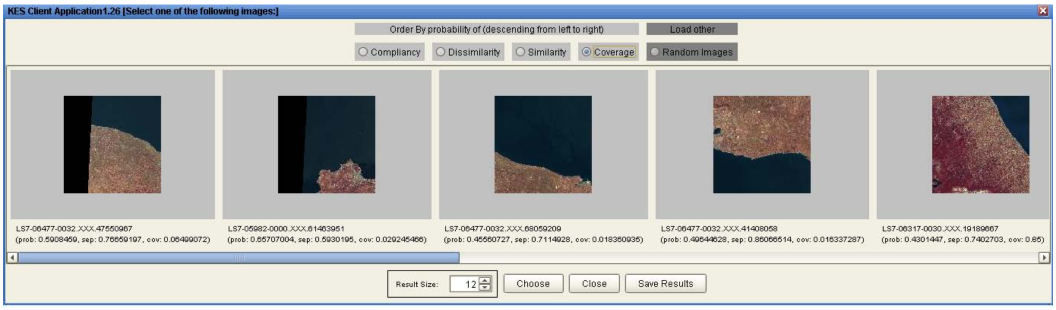
LS7-06812-0000.XXX.27260672
(prob: 0.501528, sep: 1.3635119, cov: 0.0)

LS7-06812-0000.XXX.81870306
(prob: 0.5118316, sep: 1.3631692, cov: 0.0)

LS7-06390-0033.XXX.21861328
(prob: 0.51116997, sep: 1.357742, cov: 0.743

Result Size: 12

Figure 8.5: High separability of searching for coast lines in the Landsat archive. None of the returned images contain coast lines; all of them shows water inside. (Dark areas correspond to those areas that have not been captured by the sensor, and therefore, are not included in the imagery).



KES Client Application1.26 [Select one of the following images:]

Order By probability of (descending from left to right)

○ Compliancy ○ Dissimilarity ○ Similarity ● Coverage ○ Random Images

LS7-06477-0032.XXX.47550967
(prob: 0.5908459, sep: 0.76659197, cov: 0.06499072)

LS7-05982-0000.XXX.61463951
(prob: 0.65707004, sep: 0.5930195, cov: 0.029245466)

LS7-06477-0032.XXX.68059209
(prob: 0.45560727, sep: 0.7114928, cov: 0.018360935)

LS7-06477-0032.XXX.41408058
(prob: 0.49644628, sep: 0.86066514, cov: 0.016337287)

LS7-06317-0030.XXX.19189667
(prob: 0.4301447, sep: 0.7402703, cov: 0.65)

Result Size: 12

Figure 8.6: Coverage of searching for coast lines in the Landsat catalogue. Results are ranked from images containing high percentage of coast lines to a lower percentage of them.

8.3 Multiple Classifier

In Sec. 8.1 a pairwise classification by performing an interactive learning has been detailed. The lack of that module is the capability of defining more than two classes (one - versus - rest scheme). Usually, a human trainer is interested in defining different cover types (e.g., "cloud", "ground" and "see") over the same image or set of images in full resolution, and is useful to visualize the classification of multiple cover types content. Therefore, to train and store a multi-label, we have to redefine the hyperparameter α_i, the probabilistic link $p(\omega_i|l_1, ..., l_n)$ and the posterior probabilities $p(l_v|l_{v-1})$.

For the first training set T, the behavior will be the same as described in Sec. 8.1. Thus, the hyperparameter vector will be initialized to a set of 1 and updated as $\alpha_i = 1+N_i$ and the stochastic link $p(\omega_i|l_1) = \frac{\alpha_i}{\sum_i \alpha_i}$.

For the definition of the second label, previously defined cover type must be considered, excluding labeled pixels from the candidates to be trained. The hyperparameter vector of the new label assuming $N_i^{'} \neq N_i$ is defined by

$$\alpha_i^{'} = \alpha_i + N_i^{'} \tag{8.14}$$

and the stochastic link assuming the prior cover type is

$$p(\omega_i|l_1, l_2) = \frac{\alpha_i^{'}}{\sum_i \alpha_i^{'}} \tag{8.15}$$

For the general case, the hyperparameter vector is defined as

$$\alpha_i^n = \alpha_n^{n-1} + N_i^n \tag{8.16}$$

assuming $N_i^n \neq N_i^{n-1} \neq ... \neq N_i$, and the stochastic link as

$$p(\omega_i|l_1, ..., l_n) = \frac{\alpha_i^n}{\sum_i \alpha_i^n} \tag{8.17}$$

Computing the posterior map, previously user-provided labels are considered, and in terms of Bayes' formula is defined as

$$p(l_v|D, l_{v-1}) = \sum_i p(l_v|\omega_i, D)p(\omega_i|D, l_{v-1}) = p(l_v|D)\sum_i \frac{p(\omega_i|l_v, D)p(\omega_i|D, l_{v-1})}{p(\omega_i|D)} \tag{8.18}$$

In Sec. 8.1, we have listed the information, stored in the database together with the hyperparameter vector. In the case of multiple classifications, a new hyperparameter vector is built, but the information does not come directly from the user, but from the previously defined labels. Thus, the positive examples of the new hyperparameter are computed as the sum of all positive examples, which are not equal to 1, of each label that forms the multi-label, and for negative examples, the sum of all negative examples, which are not equal to 1, is calculated. Fig. 8.7, graphically shows an example of the definition of a multi-label L, formed by the user defined semantic labels l_1, l_2 and l_3 from a set of positive and negative samples. In the example, first element of positive examples of L is 2, which corresponds to the positive value of the first element of l_2. In this case, first elements of l_1 and l_3 are 1, that means, no positive example has been given to them. However, the forth element of positive examples of L is 9, which is the sum of 5 (the forth element of positive examples of l_1) and 4 (the forth element of positive examples of l_3). In this case, the forth element of positive examples of l_2 is 1, that means, no positive example has been provided on this label.

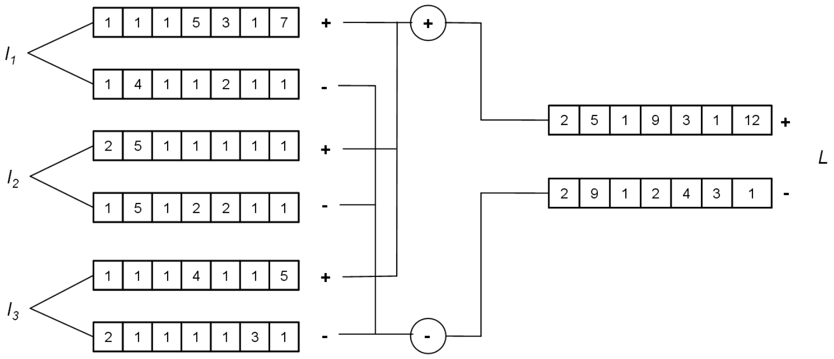


Figure 8.7: Definition of positive and negative examples of the multi-label L.

The multi-label L is saved in a file enclosing the following information:

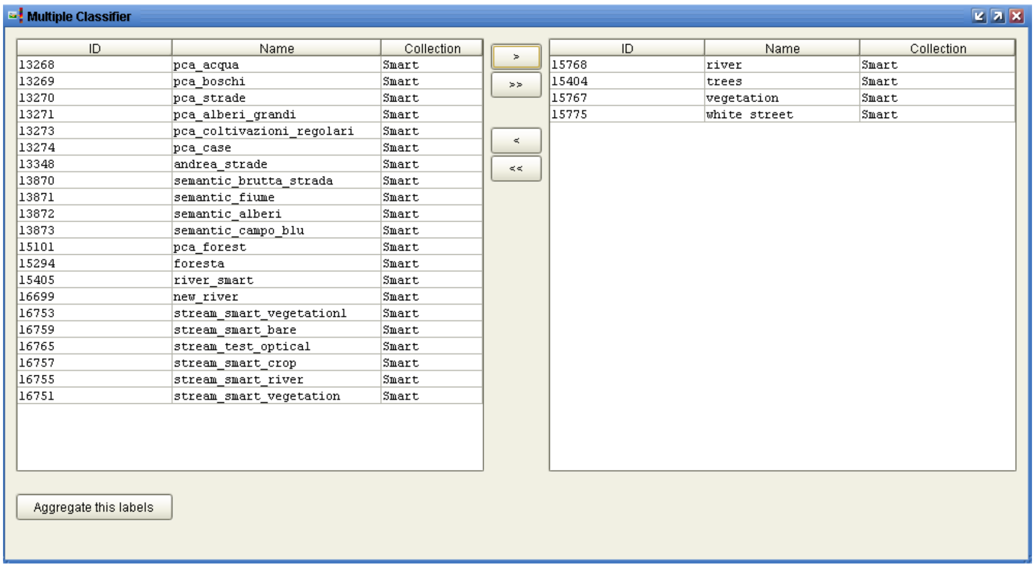
- label_names: name of the defined label provided by the user.
- classfile_type_id: mining model identifier.
- feat_name: mining model name.
- max_class: total number of clusters in the mining model.
- yes: sum of the positive examples in all labels composing the multi-label for the selected model (α_i^n).
- total_yes: sum of the positive examples in all labels ($\sum_i \alpha_i^n$).
- no: sum of the negative examples in all labels composing the multi-label for the selected model (α_i^n).
- total_no: sum of the negative examples in all labels ($\sum_i \alpha_i^n$).

Fig. 8.8 shows an example of the information associated to a multi-label.

The multiple classifier approach is available through an interface for choosing the predefined labels, which the user wants to combine. Fig. 8.9 displays the available

```
label_names river cloud land mountain
classfile_type_id 1070
feat_name spectr
max_class 64
yes
1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 1 5 4 9 4 2 5 8 1 7 1 1 4 6 22 3 7 38 1 1 33 11 5
1 2 2 1 1 1 1 1 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
total_yes 227
no
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 6 10 1 1 1 1 9 1 1 1 12 35 11 1 29 1 1 36
3 11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
total_no 216
classfile_type_id 1071
feat_name GRF
max_class 64
yes
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 8 12 1 1 1 8
17 9 8 12 1 4 1 8 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
total_yes 141
no
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2
15 11 44 1 8 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
total_no 139
```

Figure 8.8: Example of information, stored for a multi-label formed by 'river', 'cloud', 'land' and 'mountain' cover-types.

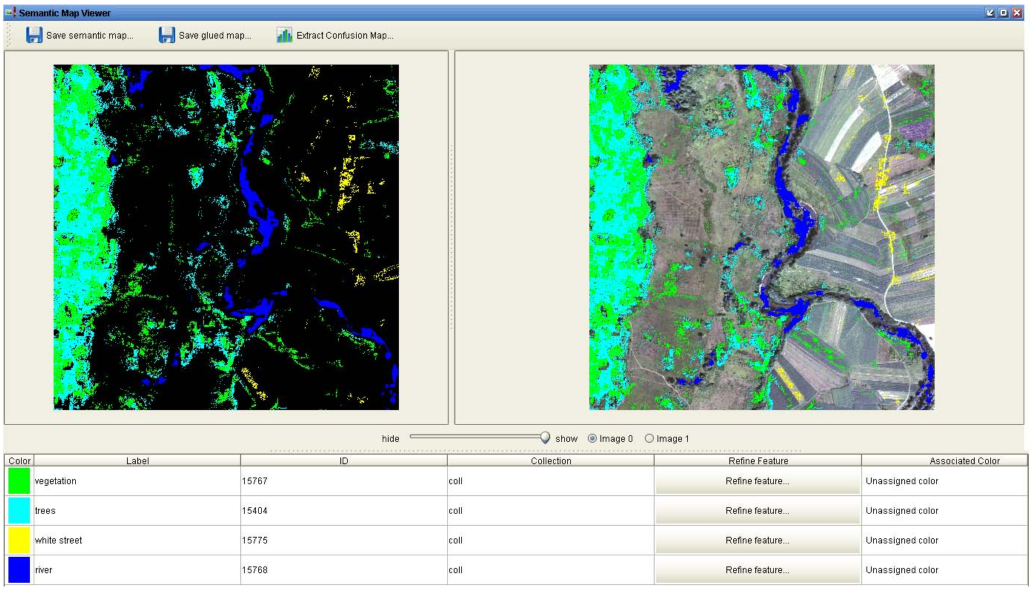


ID	Name	Collection
13268	pca_acqua	Smart
13269	pca_boschi	Smart
13270	pca_strade	Smart
13271	pca_alberi_grandi	Smart
13273	pca_coltivazioni_regolari	Smart
13274	pca_case	Smart
13348	andrea_strade	Smart
13870	semantic_brutta_strada	Smart
13871	semantic_fiume	Smart
13872	semantic_alberi	Smart
13873	semantic_campo_blu	Smart
15101	pca_forest	Smart
15294	foresta	Smart
15405	river_smart	Smart
16699	new_river	Smart
16753	stream_smart_vegetationl	Smart
16759	stream_smart_bare	Smart
16765	stream_test_optical	Smart
16757	stream_smart_crop	Smart
16755	stream_smart_river	Smart
16751	stream_smart_vegetation	Smart

ID	Name	Collection
15768	river	Smart
15404	trees	Smart
15767	vegetation	Smart
15775	white street	Smart

Figure 8.9: GUI of the multiple classifier for choosing the available predefined labels for the selected image.

predefined labels for a Daedalus dataset covering the area of Glinska Poljana (Croatia), where the user will select those labels, he wants to aggregate, in this case 'river', 'trees', 'vegetation' and 'white street'. In Fig. 8.10, the result of the aggregation of the multi-label to test data is shown. In this interface, the image on the left corresponds to the semantic map (posterior map) of the composite label: ’vegetation’, ’trees’, ’white street’ and ’river’. This map gives the user a feedback of how strong and accurate the multi-label has been defined, representing each label in a different color. This is the result of applying the posterior probability of Eq. 8.18 pixel based to the defined labels. Since the unsupervised signal classes have already been extracted on the server, and the probabilistic link $p(\omega_i|l_{v-1})$ has been previously computed and stored temporarily on local memory, only the probabilistic link $p(\omega_i|l_v)$ has to be computed in real time by the client. This allows a fast computation of the posterior probabilities and provides a quick feedback to the user. The image on the right side of Fig. 8.10 displays a hybrid composition of original image and posterior map. Fig. 8.11 shows more examples of the multi-label map, obtained by the multiple classifier for LandsatTM, SPOT5 and Meris images.



Color	Label	ID	Collection	Refine Feature	Associated Color
🟩	vegetation	15767	coll	Refine feature...	Unassigned color
🟦	trees	15404	coll	Refine feature...	Unassigned color
🟨	white street	15775	coll	Refine feature...	Unassigned color
🟦	river	15768	coll	Refine feature...	Unassigned color

Figure 8.10: GUI of the multiple classifier showing the posterior map of a multi-label composed by semantic labels ’river’, ’trees’, ’vegetation’ and ’white street’. The image on the right displays an hybrid composition of the original image and posterior map.

8.4 Human Machine Communication and Relevance Feedback

An IIM is characterized by the human-machine interaction. There is an interactive communication between the user, providing training examples, and the system, searching in the archive for images with similar content with the user interest. After the system returns ranked images (posterior probabilities, separability and coverage), the user can continue training till he is satisfied with the retrieved results. During this action, the

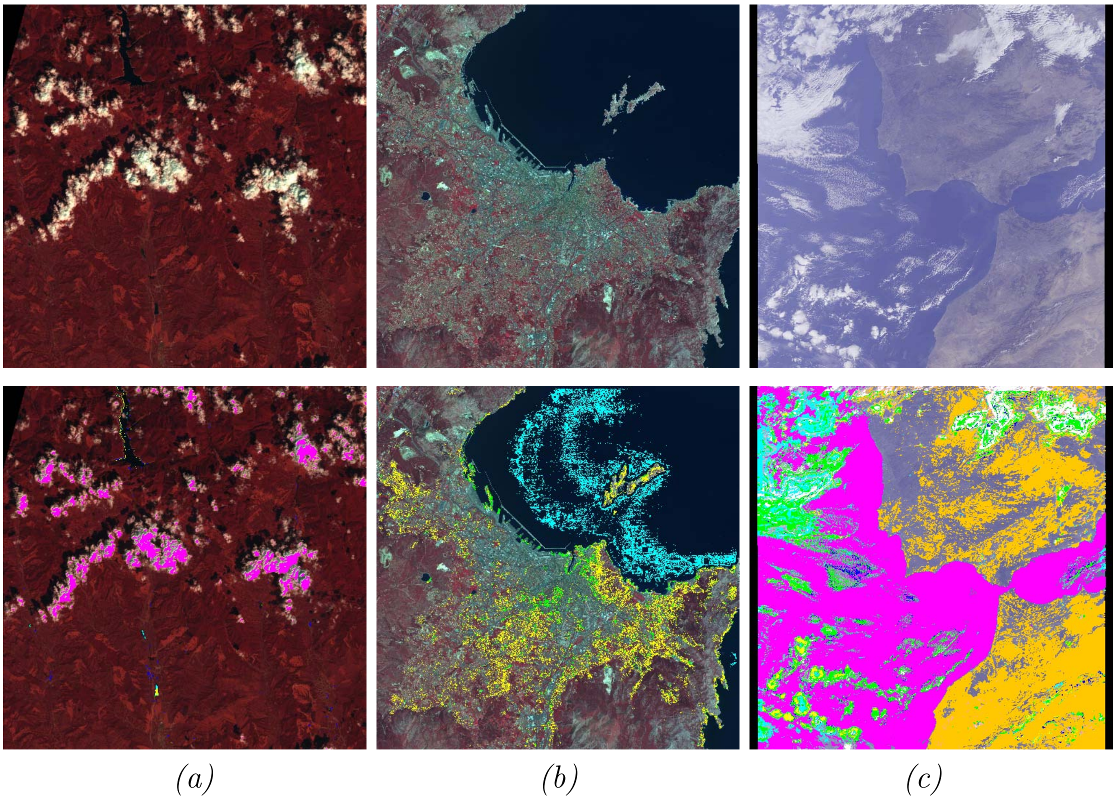


Figure 8.11: Multiple classifier applied to different sensors. *(a)* original image (top) and multiple classification map (bottom) of 'cloud' in pink, 'river' in yellow and 'city' in blue for a Landsat TM image; *(b)* original image (top) and multi-label map (bottom) of 'building areas' in yellow and 'water' in blue for a SPOT5 image; *(c)* original image (top) and multiple classification map (bottom) of 'water' in pink, 'land' in yellow and two types of 'cloud' in green and blue for a Meris image.

system learns the interest of the user by two mechanisms: incremental learning, where positive examples, provided by the user, are translated into a hyper-parameter vector in form of positive values; and incremental unlearning, where negative examples are modeled by negative values in the hyper-parameter vector.

Classical feedback is given only in one direction, that is, from user to the system in form of positive or negative examples building a training dataset. However, our system approach also returns an answer to the user in form of:

- Multiple classifier: The novel multiple classifier approach enables the definition of a diversity of categories at the level of human learning, providing a relevance feedback to the user. Allowing to manage multiple classified features helps people to perform a better selection in the learning process.

- Search categories: The probabilistic search module lets the user look in the archive for similar image content to the provided training dataset. The system answers, categorizing the results with respect to posterior probability, separability and coverage ranking the outputs from higher to lower probability.
- Signal model quality measure: Each signal model provides a quality measurement, showing the divergence between positive and negative examples. This feedback is plotted in form of graphical bars.

8.5 Conclusions

In this chapter, man-machine interaction has been addressed, focusing on the following items:

- We have seen, how a user can train the system, providing positive or negative samples, depending on his interest. Then, the system computes a probabilistic link, based on a Bayesian learning algorithm, to return a feedback in form of a posterior map. In this visualization, the user can test and decide, if he performed a good training or not, refining the samples, if necessary. This link between provided examples and cluster indexes does not demand an expensive computational performance, since images are processed and indexed in the database at ingestion time. Only the update of the hyperparameter vector is computed at real time at the client's site, and its complexity is rather low.
- Starting with training samples, the system is able to search in the archive for similar features, ranking returned images based on similarity or not similarity measurements.
- An important characteristic of an IIM system is the ability to present thematic maps. We have seen, how the multiple classifier is able to build maps, formed by several features, giving a clear visualization of the classification. This novel property can help to search in the archive for images containing more than one feature.

9

Optimal System Design Approach

In the previous chapters, we have studied, how to reduce the huge volume of extracted features through clustering techniques in order to compress data to be stored. But, generally, the repository of an IIM system needs a capacity of hundreds of Terabytes to archive original and processed data. However, the flow of data per day, uploaded from the satellite is about 100 Gigabytes. In this chapter, we concentrate on the design of an optimal system architecture for processing such a huge amount of data, including the design of a data model to provide quick access to the stored information in real time.

Once images arrive in the system, features are extracted and classified, reducing the amount of data to be stored by a factor of 10^4. However, generated classes contain no meaning and must be linked with user defined semantics. When the user queries for similar images, the system has to return an answer in a reasonable time. Moreover, the search is usually performed through an interactive training, refining the query in real time. Fig. 9.1 summarizes the flow of data from satellite acquisition, until it is ready to be used and accessible by the end user, showing how the amount of data decreases, while processing image data. We can distinguish three main archives: the first one contains 1000 of therabytes, and archives the original remote sensed images; a second repository of 100 gigabytes of data, stores the extracted features and the classification images; the third archive collects in 1 gigabyte the labels defined by the user.

This process requires an optimal system architecture, adapted to data and user driven operations providing solutions for information representation and human-machine communication. This chapter deals with the design of such an architecture, to be operational and user oriented. We begin, giving an introduction in web service technology, in web architectures, and describing the importance and application of this technology in the proposed architecture. Then, in Sec. 9.2, the system concept is defined, and the design variables and existing technologies are analyzed. Finally, Sec. 9.3 proposes an optimal software architecture describing the modular functionality by drawing several use cases where the system is involved.

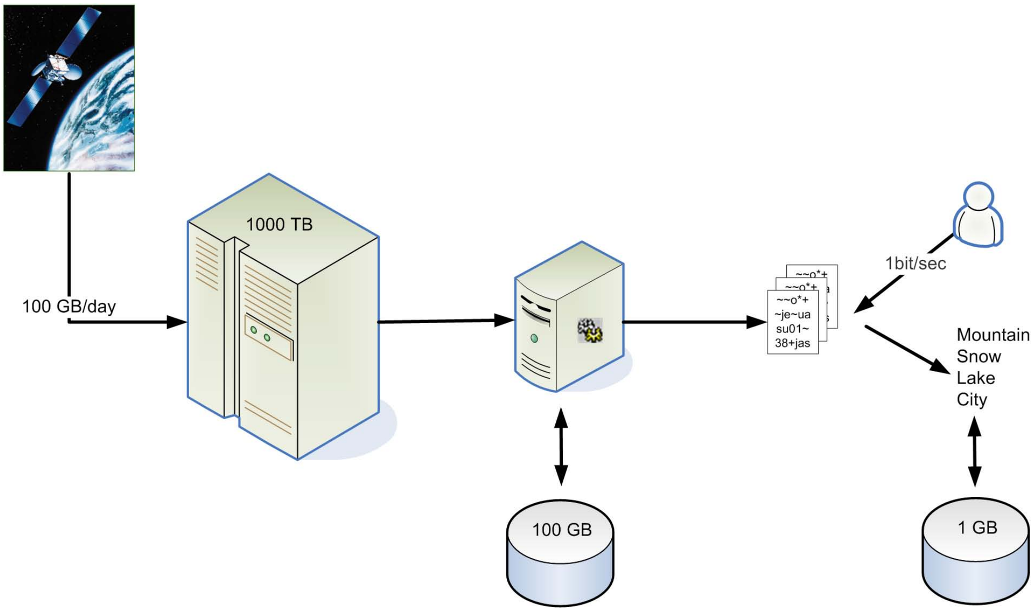


Figure 9.1: Overview of complete data flow from satellite to end user. Three main archives are considered: the first one contains 1000 of therabytes and archives the original remote sensed images; a second repository of 100 gigabytes of data stores the extracted features and the classification images; the third archive collects the labels defined by the user in 1 gigabyte.

9.1 Web Service Technology

Web services are one of the most recent technologies for the communication between machines. They can be spread over the network exposing their behavior through an XML-based file, called Web Service Definition Language (WSDL) and deployed in a web container. Web services can be used by distributed and heterogeneous architectures, being independent from the platform where they are installed and being developed in different programming languages. The communication protocol is called Simple Object Access Protocol (SOAP), that governs the format and processing rules of messages exchanged between applications. Additionally, a Universal Description Discovery and Integration (UDDI) registry can be provided for registering, publishing and locating web services. If this registry does not exist, the WSDL file has to be known by the hosts. A Service Oriented Architecture (SOA) allows different services to exchange data by using WSDL for service description and SOAP for service invocation.

A formal definition of a web service is given by the World Wide Web Consortium (W3C): *A Web Service is a software system, designed to support interoperable machine-to-machine interaction over a network. It has an interface, described in a machine-processable format (specifically WSDL). Other systems interact with the web service in* *a manner prescribed by its description, using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other web-related standards.*

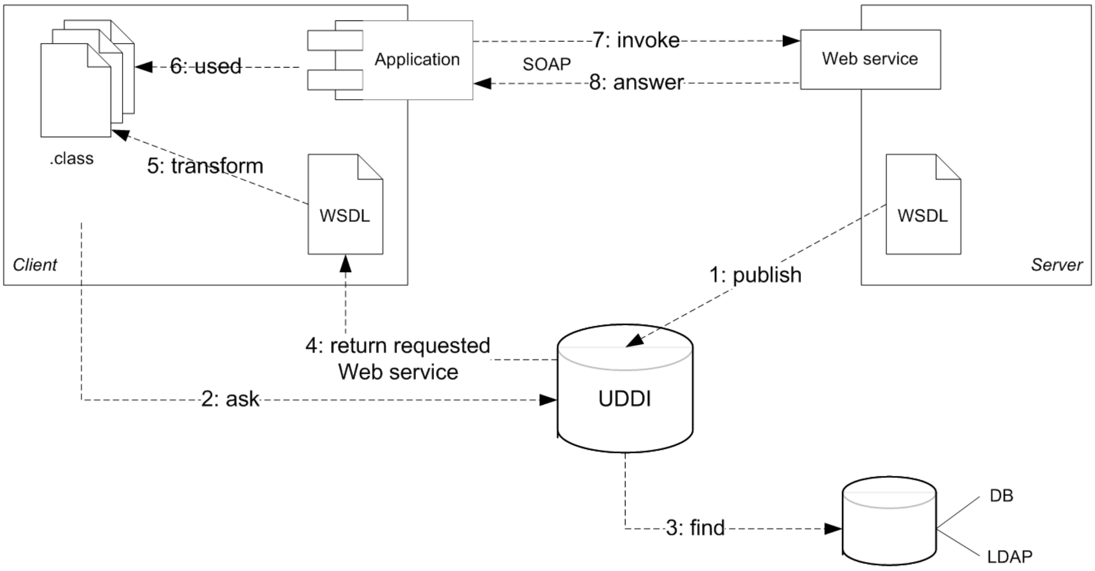


Figure 9.2: SOA Infrastructure.

Fig. 9.2 shows the general behavior of communication between a client and a web service. A web container (server) hosts several web services, whose functionalities are exposed in a WSDL file. The WSDL file is published (1) via HTTP in a UDDI registry in order to be localized. When a client needs to execute any action of a web service, it must have access to the UDDI registry. If this is the case, the client asks (2) for the WSDL file to the UDDI registry, this looks for it in its repository (this can be implemented as a database or as LDAP directory (3)) and returns the requested WSDL file to the client (4). On the client application, the WSDL file must be transformed (5) in the language understandable by the client, e.g. if the client is running an application developed in Java, the WSDL file must be transformed to Java format, that is, in *.class* files. Once the WSDL is transformed, the client application can invoke the functions exposed by the Web service (7), and therefore, exchange SOAP messages with the server.

Due to the simple implementation and the efficient communication between heterogeneous platforms, web services have been widely spread in the community.

Maamar et al. [85] show an example, where the web service composition by using software agents and context is addressed. A software agent is an autonomous program that carries out tasks on behalf of users deciding, how and when to be executed; the circumstances are the context, objects and conditions around the user. With these concepts, they try to design a system, based on composing web services. The idea is that agents are aware of their services context and build composite services on-the-fly.

Another example of application, based on web services is presented by Kurz [51]. Here, Kurz presents a theoretical framework for the communication of two repositories: the Data Information Management System (DIMS) and KES, the Information Mining System presented in chapter 3. The idea is to automatically upload data in KES, each time new data enters in DIMS. On the other side, thematic maps, processed by KES, are stored in DIMS. The communication bridge is based on a web service architecture, applying different interface protocols: SOAP and CORBA for inside and outside communication, and FTP for transferring data. The transactions will be managed by an Oracle BPEL workflow manager, based on a composed web service architecture.

Web services can also be applied for connecting mobile devices. In such an scenario, the applications must cope with dynamic environments, where both data and source availability vary over location and time. In addition, data objects, residing on mobile devices actively move throughout the system, and each device is continually processing the incoming data while looking for information of interest to its user. Such an environment is described by Perich et al. [32], where devices are grouped in information providers, devices containing data, information consumers, devices, able to query and update data, and information managers for administration functions. They constitute a framework to access heterogeneous data in an intensive way from a mobile device, based on profiles and context. They use a semantic cache to predict user behavior, where future desires are stored, and therefore, when querying the data, it provides a faster answer. This idea of semantic caching can be useful for any environment (not only mobile devices) to overcome both, the spatial-temporal variation of data, and data source availability in case of disconnection.

Instead of anticipating the intention of a user, it could be interesting, to share knowledge between domain specific users. For that, heterogeneous databases must be shared creating semantic conflicts in consequence. In this case, each database uses its own vocabulary at data and schema level, but they define the same concepts. For instance, in remote sensing archives, the common entities are image, sensor and feature, and common attributes are sensor name, image coordinates or features characteristics. To provide access to different ground segments archives, the semantic conflicts must be solved in order to offer a uniform vision of the repositories.

Ram and Park [68] present an ontology for detecting and resolving data and schema-level semantic conflicts. They propose a domain, schema and application independent ontology, based on capturing and representing similarities of data objects by analyzing its context. They introduce several techniques as semantic modeling approaches, formal languages, knowledge-based systems and classifications of terminology to establish ontology mappings between schemas. The mapping is encoded by transformation rules converting any data value to any data units already defined representing the same concept. This solution can be also applied to a system that is accessible by users, coming from several domain expertises, and trying to define their own vocabulary for the same concept.

Finally, the problem of data availability and replication must be looked at. Gao et al. [52] propose a solution for consistency and data replication. The idea is, to design distributed objects, which share application specific semantics. For that purpose, database and business logic is replicated, being communicated with the central cluster through a message layer. The approach provides high availability and excellent performance.

Classical solutions are the synchronization of the device with the central database when they are connected, or the installation of databases in cluster to guarantee data availability.

9.2 Concept and Design of a Knowledge Centered Earth Observation System

There is a strong need for more automated and human centered methods to extract information from data. In this direction, we aim at designing the KEO software system able to:

- maintain an archive for at least 10^5 to 10^6 Terabytes of scenes;
- automatically upload data from heterogeneous sources as ground segments or data provider repositories. About 100 satellite scenes per day will be automatically processed and ingested in the archive;
- extract relevant features from EO data through IIM techniques or specific processing algorithms provided by third parties;
- provide, as an output, valuable information in an easily accessible format (e.g. Geographic Information Systems (GIS) data);
- support end-users in finding, through an interrogation process, based on natural language, EO products and/or services best suited for his needs;
- provide a modularized architecture for easiness of maintenance and upgrading, be flexible enough to progressively enrich its functionalities;
- leverage on open standards for catalogue exploitation.

The business value of such a system would be, on one hand, the foreseen feature extraction mechanisms that will permit identification and classification of archived images via their information content, the direct access to the extracted features (without the need to access again the images) and the possibility to combine them in form of thematic maps. On the other hand, an open and distributed architecture allows the localization and the provision of loosely coupled services via Internet in a modular and scalable environment.

Moreover, a certain number of users, like researchers, service and data providers or end users will benefit from such a system, as it will provide the capability to identify relevant data and services at semantic level, to process their own data for information extraction, to plug-in data processing modules, or even to schedule self-made business processes. The system offers the possibility to share knowledge between individuals, belonging to a particular domain ontology or even across different domains.

When trying to define the concept of the system, it can be thought of as a 'black box', providing views of itself and interfaces with external actors (see Fig. 9.3). The idea behind is to hide the particularities of the architecture, not interesting for the user or machine. These views are the following:

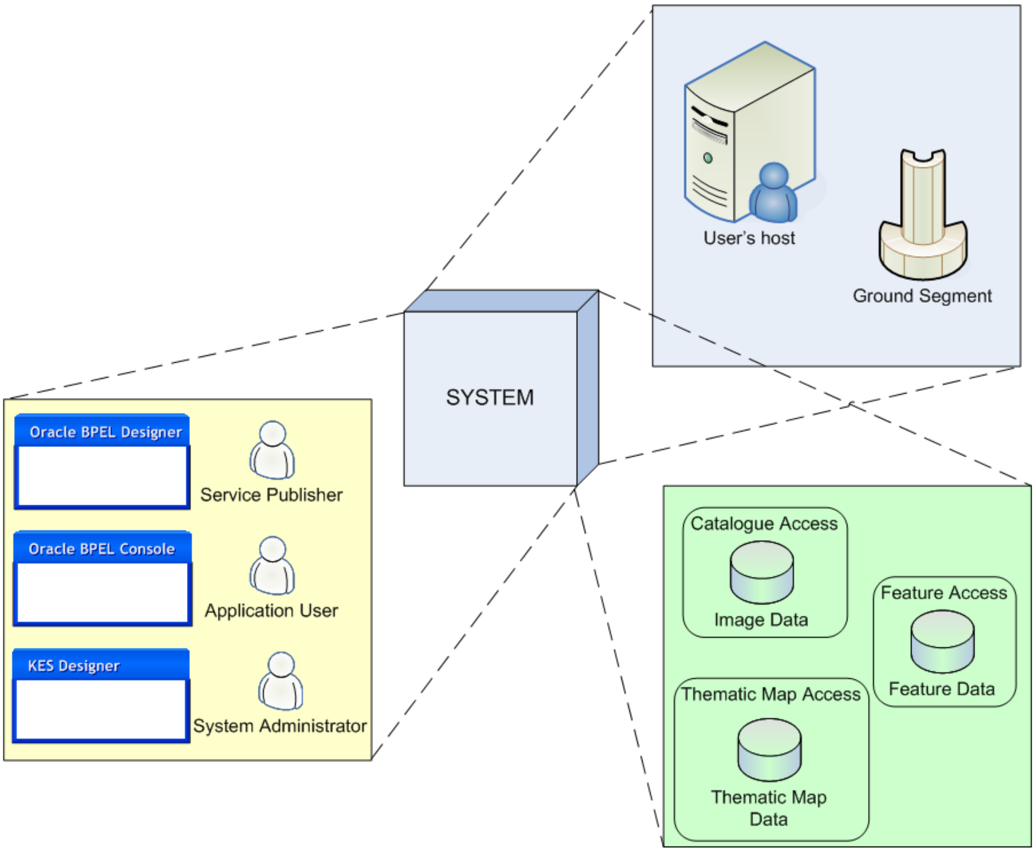


Figure 9.3: KEO external views.

- Machine Interface: KEO will be interfaced with the Service Support Environment (SSE) on one end and with one or more European Space Agency (ESA) Ground Segments on the other. The SSE provides an infrastructure, available on the Internet, which is neutrally managed by ESA. It empowers service providers to deploy and deliver web services easily. The services might be integrated directly into the

SSE framework, or remaining within the infrastructure of the service provider, being accessible through the SSE Business Process Management (BPM) platform.

KEO has to provide SSE compliant web services to be easily accessed though the operational SSE portal. In this way, KEO will define itself as an open, distributed system, able to create business value also in a form accessible from a broader context. On the other hand, to be able to process data to get valuable information and services, KEO will be fed by external data sources, represented by either a ground segment facility or a third-party user machine for those end-users, interested in using ingestion and processing services offered by the system. Some functionalities externalized as web services are data ingestion and data processing.

- Human Interface: KEO will be perceived as a human-oriented advanced system, offering functionalities, like service authoring, service orchestration, data and information browsing, global data management, etc. through the provision of a human interface view of itself. There are several actors, who can interact with the system depending on their action:
  - Service Publisher: he will be in charge of creating, deploying and monitoring the designed workflows and publishing feature maps.
  - Application User: he will have the capability of training the system for the creation of new feature labels, based on IIM techniques. This actor can be an expert on sensor characteristics or image interpretation, or a user from fields like meteorology or social geography.
  - System Administrator: he will perform administrative tasks, such as creating user domains, defining user privileges to access the system, ingesting new data if necessary, defining collection of images, etc.
- Catalogue Interface: KEO will expose the access to catalogues about original data, extracted features and thematic maps. It also offers OGC (Open Geospatial Consortium) compliant services like Web Map Service (WMS), Web Feature Service (WFS) and Web Coverage Service (WCS).

SYSTEM PROPERTIES

To design this concept, first the requirements must be evaluated and prioritized. There are some system properties, called systemic qualities, that establish the quality of the service that the system can deliver. Thus, an architecture is designed, minimizing constraints on those qualities. To fulfill the system needs, KEO architecture must be designed considering following main qualities:

- Scalability: Measure of the work and cost, required to modify the system to provide higher throughput. This means that KEO must be designed, on one side, to grow in the future, providing new functionalities, and on the other side, knowing that the number of users accessing simultaneously will increase over the time.

- Availability: Measure of the probability that the system will operate at the moment a user attempts to use it, and that the system continues operating till the completion of user's request. Usually, to guarantee availability of systems, clusters to balance the hardware and to restart the machines, if necessary, are installed.
- Extensibility: Measure of the work and cost of adding new features of functionality to the system.
- Performance: Measure of the response time as seen by the user.
- Throughput: Measure of the total amount of work that can be performed at a given time. This quality is relevant to measure, how many data can be processed and ingested in KEO archive at a given time.

From the system computational point of view, the following variables must be also considered:

- Capacity: Dimension of raw power in an element, perhaps a powerful CPU, fast network connection, or a capacious storage medium to archive at least 10^5 to 10^6 scenes in KEO.
- Modularity: Dimension, in which a computational problem is divided into separate elements and spread across multiple computer systems.
- Tolerance: Dimension of time available to fulfill a request from a user.
- Heterogeneity: Dimension of diversity in technologies that is used within a system or one of its subsystems. KEO has to communicate with different environments, platforms and modules implemented in different programming languages.

STORAGE DESIGN

Before implementing a system, the data model must be designed. For that end, the following notations will be used:

I *Image* entity

C *Collection* entity

F *Feature* entity

L *Label* entity

T *Thematic_Map* entity

id_image primary key of *Image* table.

id_col primary key of *Collection* table.

id_feat primary key of *Feature* table.

id_label primary key of *Label* table.

id_thematic primary key of *Thematic_Map* table.

The KEO database consists of feature images F, identified by an id_feat, generated by primitive features *pf* from the original image *I*. The user can train and define labels L on selected images, by giving positive and negative examples. These labels are characterized by the identifier *id_label*. Then, the system calculates measures of believes *believe* for all feature images, belonging to one collection, to provide the most relevant images regarding the trained label. Once the label is trained and validated, the measures of believes can be used to transform each feature image into a thematic map T. Thus, a thematic map identified by an *id_thematic* is generated by assigning the user defined label to a feature image.

To specify the relationships between entities of the KEO repository, the Entity Relationship (ER) model, as shown in Fig. 9.4, is designed.

According to this model, a collection C contains one or more images I and has defined several labels. In addition, the extracted features F belong to one collection, and one feature F, together with one or more labels L generate a single thematic map T. This ER model is implemented on a Database Management System (DBMS) based on IBM/Informix Dynamic Server with the Geodetic Datablade Module to perform geospatial queries.

A DBMS provides mechanisms to deal with data integrity, called constraints, that enforce the integrity of the database. We define three types of constraints at table level:

1. Domain constraints: deal with one or more columns that meet particular criteria. When a row is inserted or updated in a table, the constraint is applied without respect to any other row in the same table. This set of constraints contains check, default and rules constraints.

2. Entity constraints: focus on a particular row. This ensures that for a particular row, the same value does not exist in some other row. Primary key and unique constraints belong to this category.

3. Referential integrity constraints: created when a value in one column must match the value in another column, being either in the same table or in a different table.

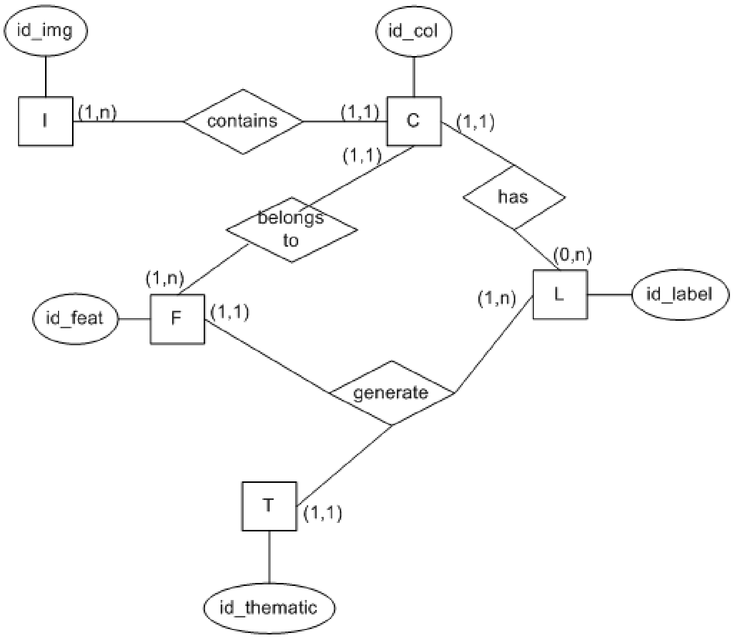


Figure 9.4: Entity relationship model for KEO repository.

In KEO DBMS all these constraints are defined, but we concentrate only on the primary key and referential integrity constraints due to their importance. To cope with the first one, a primary key is defined for each entity identifying univocally each row. The second ones are specified as follows:

- Referential integrity between *Image* and *Collection* tables: all images i must be an element in a collection c. Therefore, the *Image* table contains a collection identifier as foreign key.

$$\forall i \epsilon I (\exists c \epsilon C : (i.id\_col = c.id\_col)) \tag{9.1}$$

- Referential integrity between *Feature* and *Collection* tables: all features f belongs to one collection c. Therefore, the *Feature* table contains a collection identifier as foreign key.

$$\forall f \epsilon F (\exists c \epsilon C : (f.id\_col = c.id\_col)) \tag{9.2}$$

- Referential integrity between *Label* and *Collection* tables: all defined labels l are defined over one collection c. Therefore, the *Label* table contains a collection identifier as foreign key.

$$\forall l \epsilon L (\exists c \epsilon C : (l.id\_col = c.id\_col)) \tag{9.3}$$

- Referential integrity between *Thematic_Map*, *Label* and *Feature* tables: thematic maps t are generated from a feature map f and a collection of labels l. Therefore, the *Thematic_Map* table contains a feature and label identifiers as foreign keys.

$$\forall t \epsilon T(\exists l \epsilon L : (t.id\_label = l.id\_label) \wedge \exists f \epsilon F : (t.id\_feat = f.id\_feat)) \qquad (9.4)$$

9.3 KEO System Architecture

After studying the prerequisites of KEO, a modular architecture, based on web services is proposed, considering constraints and merging existing technologies: the modular components shall be developed by using object-oriented programming, the communication between them shall follow standard protocols and they shall be orchestrated by a higher-level module. With these premises, the KEO global architecture consists of:

- Application Server: All software components running in KEO system will be deployed in one or more GlassFish Application Servers distributed between the KEO server stations. These components are based on the Java 2 Enterprise Edition (J2EE) technology:
  - Servlet: A Java application that runs in a Web server or application server and provides server-side processing, such as accessing a database. Widely used for Web processing, servlets are designed to handle HTTP requests (get, post, etc.) and because they are written in Java, servlets are portable between servers and operating systems. In KEO, user requests from the client will arrive at a servlet that acts as a facade between client and server. The servlet will pass the request to Enterprise JavaBeans (EJB) modules to access the database.
  - Web service: A web service is a software component, designed to communicate via Internet. It enables clients and services to communicate, regardless of the object model, programming language, or runtime environment, used on either side of the communication link. They are identified by an URI and its description and communication protocols are based on XML standards:
    - WSDL: is an XML format for describing network services as a set of endpoints, operating on messages, containing either document-oriented or procedure-oriented information. It specifies the location of the web service and the exposed operations.
    - SOAP: is an XML-based communication protocol to let applications exchange information via Internet through HTTP protocol. It is platform and language independent, allowing to get around firewalls and developed as a W3C standard. It is an extensible protocol, building an scalable environment.

        * UDDI registry: provides a standardized method for publishing and discovering information about web services. It attempts to create a platform independent, open framework for describing services, discovering businesses, and integrating business services. UDDI focuses on the process of discovery in the SOA. Being a web service itself, UDDI is invoked using SOAP. In addition, UDDI also defines, how to operate servers and how to manage replication among several servers. Just registering a new web service in KEO UDDI registry, as for example, a new feature extraction algorithm, wrapped as a web service, it will be accessible for any user.
    - Enterprise JavaBeans (EJB): A software component that resides in a runtime engine called EJB Container, which provides a host of common interfaces and services to the EJB, including security and transaction support. In KEO will be used session beans providing the business logic and entity beans to access the database.

- Catalogue Database Server: The catalogues of KEO will reside in an Informix database. These catalogues build the main repositories of original imagery, thematic maps and features, that are accessible from outside through SSE.

- OGC compliant Database Server: The OGC catalogues are stored in a PostgreSQL database. The OGC services WMS, WFS, and WCS are available to SSE according to the SSE specifications for OGC catalogues.

- Client Application: The KEO client will be implemented with Java Web Start (JWS) technology. JWS is a client-side technology that enables Java applications to launch either from a desktop or from a Web page. It ensures that the most current version of the application will be deployed, as well as the correct version of the Java Runtime Environment (JRE). On the other side, while interacting with a web application, the user waits two or three seconds for an answer. Using this technology, the system can make use of local resources performing some tasks on the client, and therefore providing a quick answer to the user. Using this technology, KEO client will be installed on the local machine and will provide a user interface to perform the interactive learning phase.

Fig. 9.5 shows the global architecture of KEO in a physical UML diagram. Basically, it consists of an application server, containing several web services that externally expose some functionalities, servlets to process client HTTP request, and EJBs to perform the business logic and to connect to KEO' main repository. This connection with the database is carried out by using data sources, ensuring the availability, while accessing the database.

To access services from the external world, different protocols are used, depending on the action to be performed: when the Application User wants to use IIM techniques, a GUI is provided which is accessible via Internet through the HTTP protocol; on the other side, catalogues exposed in the SSE portal are connected to this infrastructure via SOAP;

also clients to invoke KEO functionalities exchange SOAP messages; and finally, web services, implemented for uploading new data in KEO repository are reachable through the FTP protocol. The communication between the application server and the database servers (catalogues and OGC repository) is reached through Java Database Connectivity (JDBC) connection.

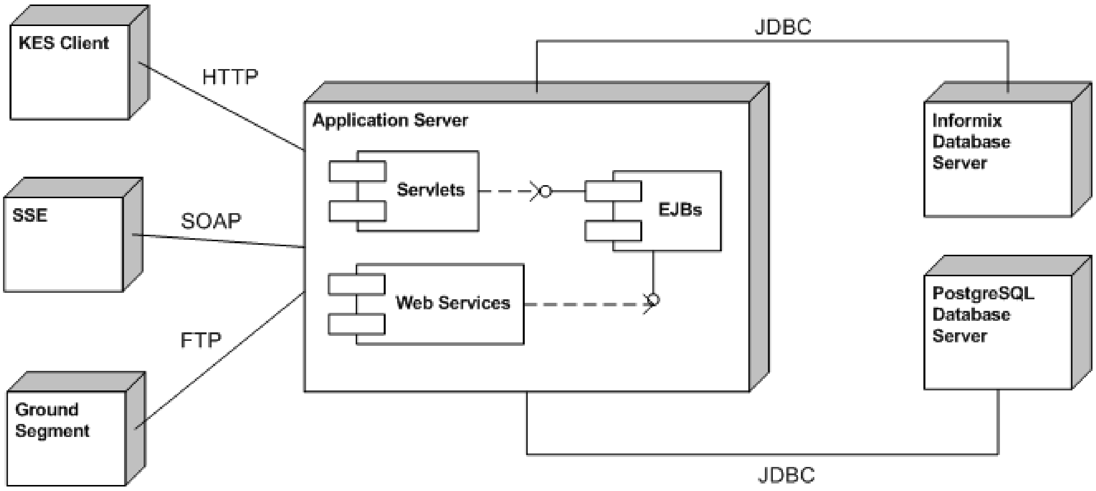


Figure 9.5: UML physical diagram of KEO system.

Once the global KEO architecture is explained, we concentrate on the functionalities to be implemented and exposed to the external world. As discussed, KEO is a component-based system, that aggregates a certain number of software modules (cots, libraries, loosely-coupled services, etc.) in order to get the maximum benefit from their mutual interaction. The system is based on a service oriented architecture, hence it is intrinsically able to orchestrate service workflows with a variable level of complexity. From this point of view, the system will be perceived as an open container, hosting predefined modular components, but also able to enlarge the set of provided functionalities by registering and managing new modules, coming from the external world. It will also be capable to let them participate in broader workflows. Hence, KEO will create, edit and manage processing workflows and/or components, working on data from external interfaces.

The selected tool to expose any working workflow as a web service is the Oracle BPEL Process Manager. This authoring tool allows to easily build processing lines to be setup into an operational environment. KEO will also take into account the possibility to be manned by a human being, hence it will provide graphical interfaces as simple as possible to be used by end-users for a certain number of tasks.

Fig. 9.6 shows the KEO architecture composed of web services, graphical interfaces and Oracle BPEL tools. These modules are described later on.

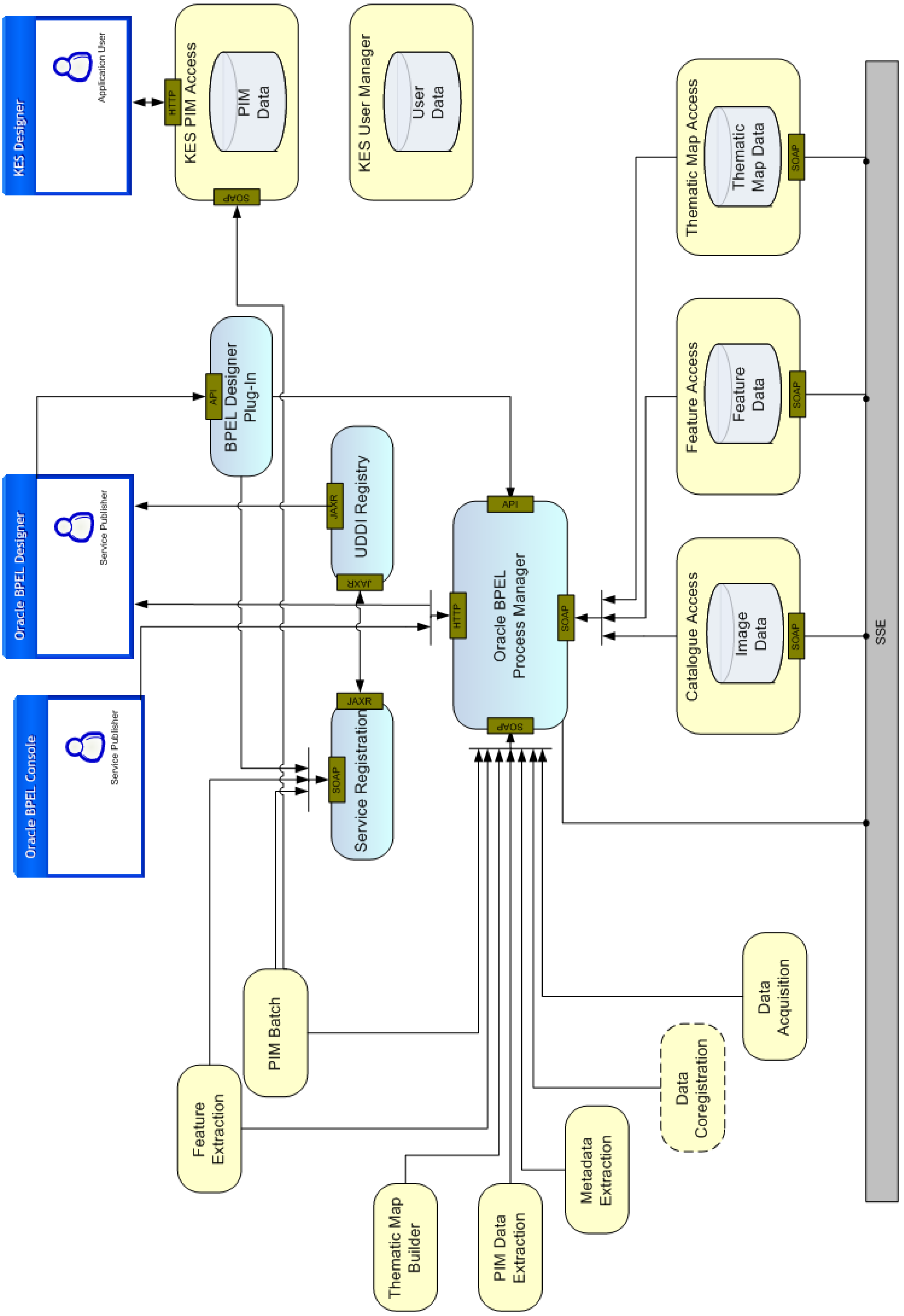


Figure 9.6: KEO architecture.

9.3.1 KEO Subsystems

In this section, we give a full description of the modules, building the KEO architecture. Fig. 9.7 shows the complete architecture of KEO, grouping modules by their functionalities. In yellow we mark those subsystems, whose design is the scope of this dissertation. These subsystems are: the Ingestion Subsystem, the Workflow Management Subsystem, the IIM Training Subsystem and the Catalogues Subsystem. However, the design of the SSE Interface Subsystem and the Knowledge Base Subsystem, displayed in green, have not been studied in this thesis.

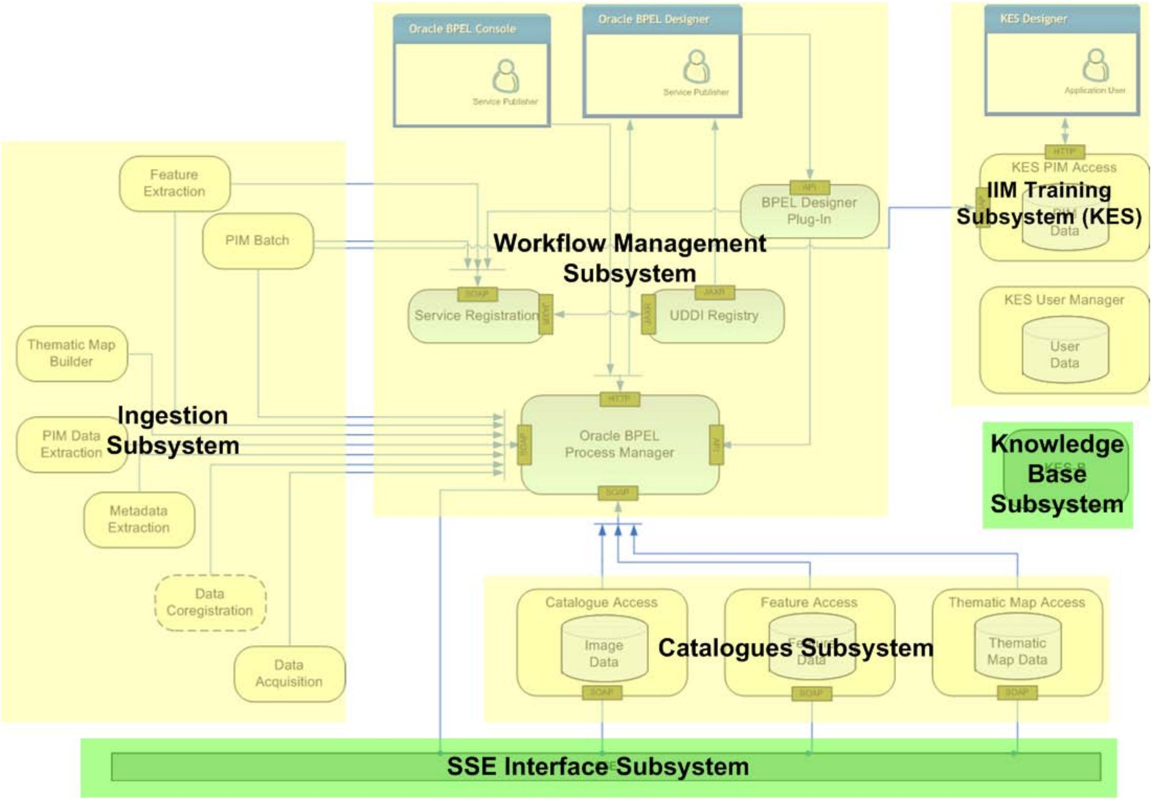


Figure 9.7: KEO subsystems. In yellow we reference those subsystems, whose design is the scope of this dissertation. These subsystems are: the Ingestion Subsystem, the Workflow Management Subsystem, the IIM Training Subsystem and the Catalogues Subsystem. The design of the SSE Interface Subsystem and the Knowledge Base Subsystem, displayed in green, have not been studied in this thesis.

1. **Ingestion Subsystem**: This subsystem consists of those modules that deal with the ingestion chain: images coming from a Ground Segment facility or a third-party user machine are uploaded via FTP protocol, then, primitive features and

Probabilistic Information Mining (PIM) data are extracted, and finally, they are archived in clustered form. The following modules belong to this subsystem:

  - Data Acquisition Service: Service in charge of uploading data from external actors via FTP.
  - Data Coregistration Service: Service that enables the coregistration of data files.
  - Metadata Extraction Service: It extracts metadata from images and stores them in an image catalogue archive.
  - PIM Data Extraction Service: PIM is the process of linking user specified semantic terms interactively defined during training of the system, to feature labels. Thus, this web service is in charge of extracting primitive features from data.
  - PIM Batch Service: This service enables the spatial localization of the extracted primitive features, building GIS maps.
  - Feature Extraction Service: This service extracts non primitive features from data.
  - Thematic Map Builder Service: Service that produces thematic maps by merging together different GIS layers.

2. **Catalogues Subsystem**: In this subsystem, we find the modules that provide access to the catalogues. These catalogues are exposed as web services, and therefore, accessible by the BPEL workflow engine to be part of a workflow and also accessible through the SSE environment.

   - Catalogue Access Service: It provides access to original images and related metadata.
   - Feature Access Service: This service provides access to extracted features.
   - Thematic Map Access Service: Service that provides access to thematic maps.

3. **IIM Training Subsystem (KES)**: This subsystem corresponds to the KES system, enabling the user to train the system searching for desiderable features by performing a probabilistic search and building classification maps containing one or more features.

   - KES PIM Access Service: This service provides an interface to access the PIM data catalogue.
   - KES User Manager Service: It manages user registration and knowledge sharing between users belonging to the same domain.

- KES Designer: Graphical User Interface (GUI) that allows the user to interact with the KES system. The kind of actor that can play with the system is an Application User. He can be an expert in sensor characteristics or data interpretation or an unexperienced user in these fields.

4. **Knowledge Base Subsystem (KES-B)**: This subsystem corresponds to the KES-B system and is in charge of exploiting the resources and identifying the required information by using users vocabulary, domain knowledge and preferences. This subsystem will be not described in detail, because it was not designed in the scope of this thesis.

5. **Workflow Management Subsystem**: This subsystem creates and deploys flows of Web services and orchestrates the execution of them by a workflow engine. The subsystem itself builds a web service, providing the capability of being enlarged with new modules. The provided GUIs are used by a Service Publisher actor, who is able to create and deploy new workflows. The main modules of this subsystem are:

   - Oracle BPEL Process Manager: This module orchestrates the web services, involved in a workflow, being exposed itself as a web service and activated on demand.
   - Service Registration: This software component is in charge of registering new web services, in order to be accessible through Oracle BPEL Process Manager.
   - UDDI Registry: Registry of all web services available for the Oracle BPEL Process Manager to be used. It provides the service interface specification in form of a WSDL file.
   - BPEL Designer Plug-in: This additional plug-in deploys a Web service into an application server and registers it in the UDDI server.
   - Oracle BPEL Designer: GUI for designing, creating and deploying workflows of web services.
   - Oracle BPEL Console: This module allows monitoring and debugging deployed workflows at runtime.

6. **SSE Interface Subsystem**: The whole KEO architecture is implemented as SSE compliant to be accessed through the operational SSE portal. This subsystem provides dedicated tools to create and publish the designed workflows as a web service in the SSE environment, and therefore, reachable for the community.

9.3.2 Use Cases and User Scenarios

In the previous section, the KEO subsystems have been described, explaining the functionalities, the user roles and the provided GUI available for each subsystem. This section tries to explain visually and in a more practical way the architecture, showing user interactions and flow of actions to be executed when ingesting new data, performing an interactive learning for definition of feature labels, or creating a workflow to be exposed as a web service.

9.3.2.1 Data Ingestion

In chapter 3, KIM and KES systems have been presented, revealing that neither KIM nor KES provides any automatic way of ingesting new data, and therefore, it should be manually executed. KEO introduces a new workflow for data ingestion in such a way that any time, new data is uploaded via FTP, they are automatically ingested. This workflow consists of a composition of several web services, performing feature and PIM data extraction, GIS maps production and thematic maps composition. These products are also exposed externally as web services. Fig. 9.8 displays the flow of operations when new data arrive at KEO:

1. The Process Manager starts the workflow operations, waiting for a notification event from the Data Availability Trigger Service (monitoring the presence of new data file in the KEO FTP area).
2. Optionally passed to the Data Coregistration Service, and
3. to the Metadata Extraction Service. It will extract metadata, depending on the format of its input data and save them in the KEO Temporary Storage Area (KTSA).
4. This metadata will be stored into the OGC-compliant Image Catalogue by the Catalogue Access Service.
5. Now all services that also require the extracted metadata for their processing will be started in parallel. Specifically:
   - the PIM Data Extraction Service, that extracts the PIM Data and save the association links between PIM data and metadata, back to the Catalogue Access Service,
     5.1. then the PIM Data are moved into the KES PIM Server
     5.2. and all PIM Batch Services deployed in KEO will be started in parallel. They will extract features that will be saved in the KTSA as GIS layers;
   - the user provided Feature Extraction Services, that will extract features that will be saved in the KTSA as GIS layers too.
6. At this point, the Process Manager will ask the Feature Access Service to move extracted features from the KTSA into the OGC-compliant Feature Server.

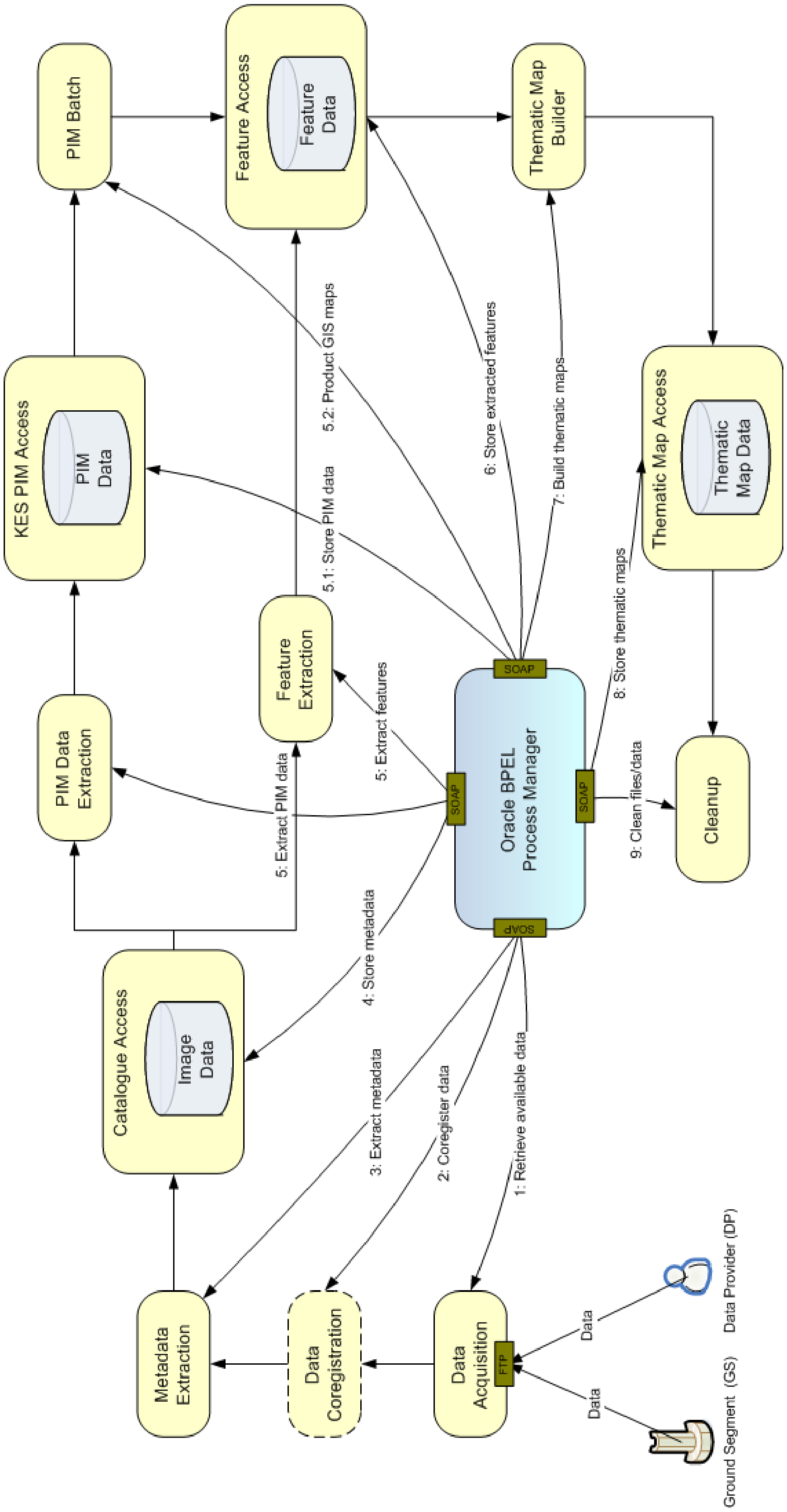


Figure 9.8: Ingestion of data from a Ground Segment or a Data Provider.

7. When required, the Thematic Map Building Service will be started in order to build thematic maps from data coming from the Feature Access Service,

8. and previously saved maps will be moved from the KTSA to the Thematic Map Access Service.

9. At the end, any useless/temporary data/files will be removed.

9.3.2.2 Creation of Feature Labels

With this use case, a practical view of user interaction and feature labels definition, described in chapter 8, is given. This process has been present in KIM and KES systems, and therefore, KEO also performs this functionality, wrapping the whole business logic and exposing it as a web service. This use case can also describe the behavior of the multiple classifier, explained in chapter 8.

The user goes through an iterative process by interacting with the system with several elaboration cycles, until he is satisfied with its definition. The whole interaction is performed visually by selecting an image pixel with a positive or negative sample representing the searched feature. Once the feature label has been created, it can be stored on a remote server to be used afterwards to perform refinements or queries by content-based parameters. The actor involved in this scenario is the Application User, who owns a GUI called KES Designer to execute the IIM techniques. Fig. 9.9 shows the flow of operations:

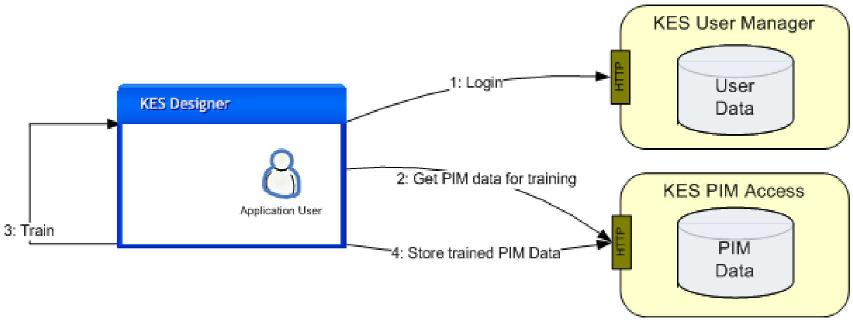


Figure 9.9: Interactive learning and feature label definition process.

1. The Application User logs into the system in order to let it know about its profile, privileges, etc.

2. Then, he searches for the image collection, he wants to work with, and selects the PIM data

3. to train the system on the new Feature Label being defined.

4. Once the training is complete, the Feature Label is stored as PIM data and published.

9.3.2.3 Creation of a Service Workflow

This use case explains, how the Service Publisher can create new workflows of cooperating web services. To this aim, it has to use the Oracle BPM designer that provides an intuitive way of designing new workflows from available web services, registered in the UDDI registry. Once the new service (the workflow) is created and deployed, it can be monitored through the provided interface Oracle BPEL Console. Fig. 9.10 shows the flow of actions to create a new service workflow.

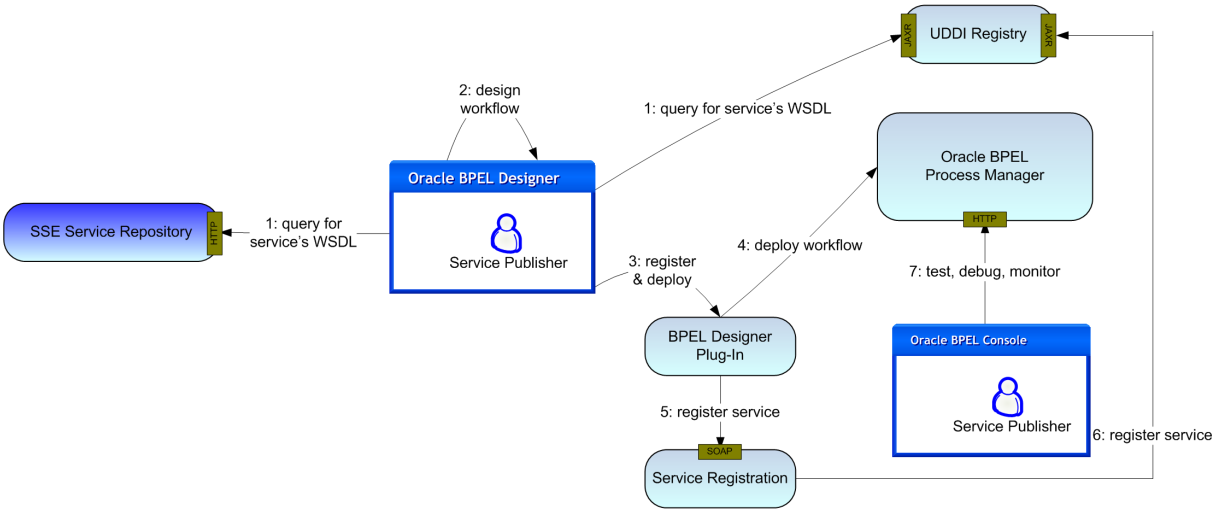


Figure 9.10: Workflow creation process.

1. The Service Publisher accesses the Oracle BPEL Designer to define the workflow with the appropriate invocation of web services. To this aim, he needs the service interface specification, provided in the form of the service WSDL file, either by the UDDI registry (KEO internal services) or by the SSE Service Repository. From the WSDL file, the Oracle BPEL Designer learns about service interfaces and invoques them in the workflow BPEL file.

2. Throughout the design process, the Oracle BPEL Designer can perform a syntactic check of the BPEL file being produced.

3. The workflow is then assembled into a web service by the Oracle tool and through an additional Plug-In the web service is

4. first deployed to an Application Server where the production part of the Oracle BPM manager is located,
5. then registered through the Service Registration module
6. in the UDDI server to allow the location of the new web service (the workflow).
7. At this point the workflow is operational and can be launched, monitored and/or debugged through the Oracle BPEL console, which is a web application used by the Service Publisher to manage deployed business processes.

9.4 Conclusions

This chapter describes the system perspective of this dissertation, trying to design the optimal software architecture for KEO. The concept design is based on a modular and user oriented architecture, and the communication is focused on standard exchange of messages.

Tab. 9.1 depicts the link between signal processing and system perspectives. In here, we observe that original data, extracted features and thematic maps are stored in the database, building each one a separate catalogue. The generated classfiles that reference the clusters with the user defined semantic labels are transformed in database indexes. All the signal processing methods are implemented as web services, which provides accessibility from outside. The graphical user interfaces are developed in Java and installed on the client through Java Web Start (JWS). The communication between modules is performed via SOAP, between systems via the FTP protocol and the communication client-server is achieved through HTTP requests. Finally, web services and interfaces are orchestrated by the Oracle BPEL Process Manager, that care about the correct data flow between modules.

Information Hierarchy **(Signal Processing Perspective)**	**Technology** **(System Perspective)**
Data (Image, Features, Thematic Maps)	Database tables
Classfiles	Database indexes
Image processing methods	Web services
Interfaces	Java Web Start
Communication	SOAP, FTP, HTTP
Data flow	Oracle BPEL Process Manager

Table 9.1: Link between Signal Processing and System perspectives of this dissertation

10

Application Domains

Since the system intends to respond to existing and new requirements of a very broad range of applications, the interest of using an IIM system reflects a broad diversity of tasks: access to information in large EO archives, image interpretation, target detection, object recognition, knowledge discovery, etc. It can be applied in fields like meteorological data, analysis or surveillance applications, depending the application domain, or the sensor resolution. Tab. 10.1 depicts examples of domains where the different sensors can be applied. Aerospace agencies, satellite centers, universities, industry or data providers are common users of this kind of systems.

Application Domain	**Data**				
	MERIS	Landsat	SPOT	Ikonos	TerraSAR
Climate	x				
Atmosphere effects	x				
Environment	x	x	x		
Vegetation	x	x	x		x
Agriculture		x	x	x	x
Urbanization		x	x	x	x
Hydrology		x	x	x	x
Disaster/Conflicts			x	x	x
Humanitarian help			x	x	x
Security			x	x	x

Table 10.1: Application domain of MERIS, Landsat, Spot, Ikonos and TerraSAR sensors.

In this last chapter, we give two examples of application domains of an IIM system, trying to motivate the reader to use it for further EO applications and to discover the wide variety of fields where it can be applied. Sec. 10.1 presents a real application for target detection by multi temporal analysis addressing how the unsupervised extracted features are fused by providing a supervised change detection approach, and Sec. 10.2 details an evaluation procedure of the system for feature classification.

10.1 Multi Temporal Analysis for Target Detection

In chapter 5, several feature extraction algorithms for change detection applications have been implemented: Temporal Spectral Angular Distance (TSAD) and Normalized Difference Vegetation Index (NDVI) methods for the detection of small targets from multi temporal images and the Color Normalization Gaussian (CNG), suitable for extracting strong shadows.

In this section, an unsupervised synergy of the described methods is proposed, which will be applied later in a supervised change detection approach. Both alternatives, the unsupervised and supervised, are analyzed and evaluated for a real application.

10.1.1 Unsupervised Change Detection

Fig. 10.1 illustrates the operational flow for the unsupervised change detection approach. After extracting primitive features (TSAD, NDVI and CN) from original images, the noise is eliminated applying a quasi-automatic threshold. Because the goal is to obtain only one map, containing the targets, an operation between the processed multi-temporal image, like difference or union must be performed, obtaining the CNG, TSAD and NDVI map. TSAD and NDVI are able to detect valuable changes and CNG method extract strong shadows, therefore, we propose a synergy between TSAD and CNG on one side, and between NDVI and CNG on the other side. With this approach, the accuracy of detection is improved gaining a quasi-complete description of the image signal changes.

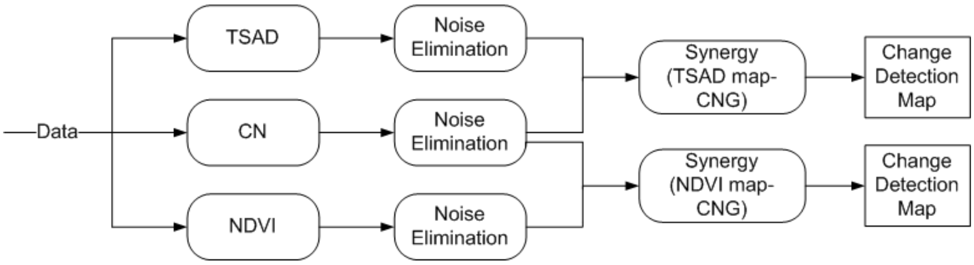


Figure 10.1: Procedure of the unsupervised change detection approach.

NOISE ELIMINATION

The results obtained through TSAD and NDVI approaches contain some noise, considering noise as false alarms, that can be easily eliminated by applying a certain threshold. However, this threshold discards information, both targets and false alarms. Thus, one of the main problems related to unsupervised change detection based on combination of spectral bands lies in the lack of automatic methods for discrimination between changed and unchanged areas. The accuracy of the final target map will depend on the selection of this decision threshold.

Generally, in remote sensing applications, this choice is provided by trial-and-error heuristic strategies. Bruzzone and Prieto [12][14] propose two automatic techniques, based on Bayes theory for the selection of an appropriate threshold. One of them assumes that pixels in the difference image are independent of each other, and the other one considers the spatial-contextual information based on Markov Random Fields (MRF) included in the neighborhood of each pixel. Both techniques require the knowledge of the statistical distributions of the changed and unchanged pixels. They assumed a Gaussian distribution because, in several change detection application, it has been proven to be reasonable when images are acquired by passive sensors. However, when different typologies of land cover must be identified or when active sensors are used, the Gaussian model may not be appropriate.

We propose a quasy-automatic mixture of concepts for selecting the threshold: for the false alarm map (output of *CNG*), we consider the statistical distribution of its pixels, and for a target map (output of *TSAD* and *NDVI*), we analyze the scaled pixel values. Consequently, the selection of the threshold will be different, depending on the method:

- Temporal Spectral Angular Distance

  After applying the TSAD algorithm, large angle values represent pixels belonging to a target, and small ones correspond to non targets. Analyzing the angle values, the mean of them is considered as a good threshold.

  Once the threshold is applied to both images, the difference between them is computed to achieve a single image. The resulting map, in the following the TSAD map, consists of a binary mask denoting the presence of a target by a pixel value of 1 and the absence of target with 0 value.

- Normalized Difference Vegetation Index

  After scaling the NDVI images, the pixel value range is between 0 and 200, the value of 100 represents the limit between vegetation and non vegetative areas. Thus, this value can be considered as a good threshold for the computation of the target map.

  In order to enhance the spectral differences and to reduce the effects of topography, we compute the difference between both data. The resulting map, in the following the NDVI map, consists of a binary mask denoting the presence of target by a pixel value of 1 and the absence of target with 0 value.

- Color Normalization Gaussian

  In chapter 5, we described, how the CNG algorithm eliminates the noise from the CN results. Applying this approach to image data, strong shadows come visible enabling their isolation.

As we need a mask, containing all shadows on the multi temporal image, the union of the bitemporal CNG data is computed. This algorithm provides a procedure to build a false alarm mask, valuable for the assessment of methods. In this mask, the false alarm presence is denoted by 1 and 0 represents absence of it.

SYNERGY OF METHODS

Each of the previous methods can extract only part of change information, limiting the performance of target detection. To enhance the probability of detection and to reduce the probability of false alarms, we propose a synergy of the algorithms. The basic idea is to clean the TSAD and NDVI maps of false alarms, detected by CNG algorithm.

However, the spatial and temporal clutter of data reduces the probability of target detection. For this reason, a reasonable neighborhood must be considered, in such a way that when a pixel belonging to a false alarm is found, the method will consider not only that pixel as a false alarm, but also a previously fixed neighborhood, removing them from the TSAD and NDVI maps. The data must be carefully analyzed for the selection of the neighborhood size, called order. In the Dorsten case, pixels belonging to false alarms are far from those belonging to target ones. On the contrary, in Borculo and Marseille data, targets are very close to false alarms. In fact, while cleaning maps of false alarms, targets can be also removed, rendering the order selection a critical decision. Therefore, the considered neighborhood will be of one pixel, where false alarm and target are nearby (Borculo and Marseille scenes), and five pixels, when they are far away (case of Dorsten site). In general, to avoid deleting targets, while removing false alarms, a neighborhood of one pixel can be considered, but this reduces the accuracy of the method.

Fig. 10.2 shows the synergy of methods applied to Dorsten scene, and the ground truth of two targets detected by applying the synergy technique.

10.1.2 Supervised Change Detection

In the literature, several supervised techniques for detecting changes in EO images have been proposed. Most of them require the availability of a multi temporal ground truth to derive a training dataset, containing information about the spectral signatures of the changes. However, a suitable ground truth is not often available or it is very expensive to manage.

After applying the synergy of methods described in Sec. 10.1.1, we obtain a map of targets with still some remaining false alarms, but containing the knowledge of the changed areas location. This result can be used as primitive features for an interactive learning process considering the detected targets as a training dataset. This supervised method will improve the detection of targets not requiring the availability of ground truth.

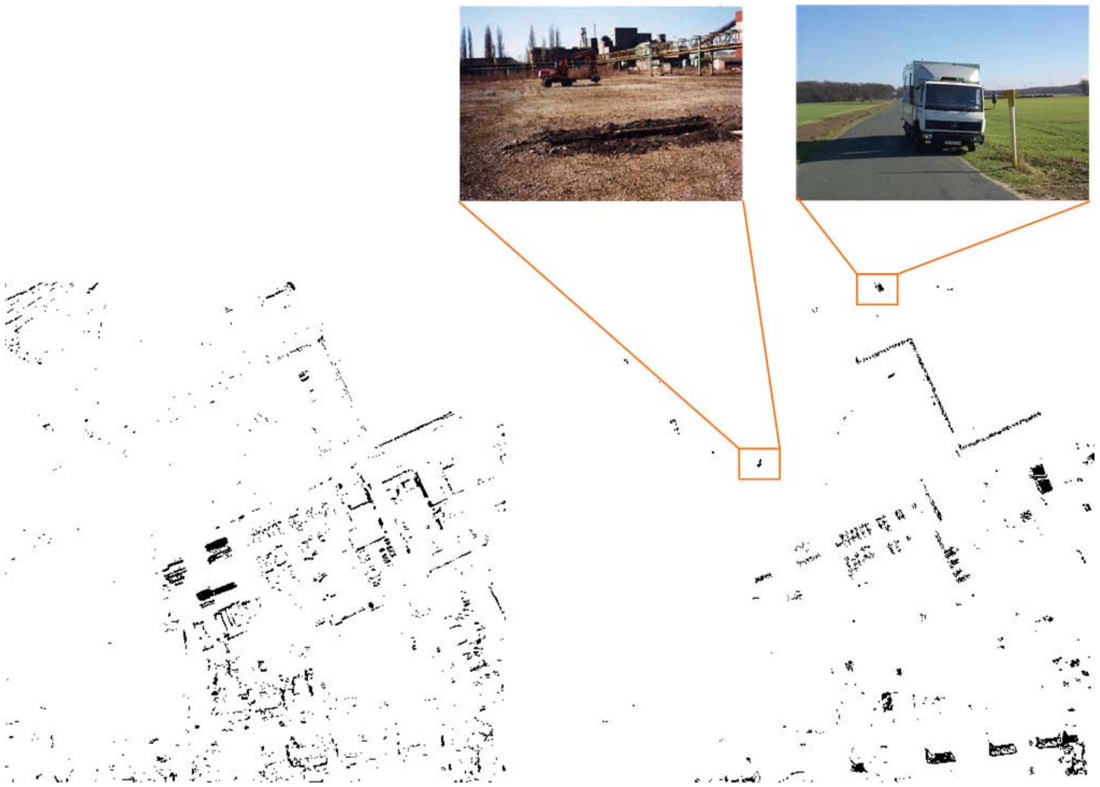


Figure 10.2: Image on the left side shows the synergy of TSAD map and CNG applied to Dorsten site; Image on the right side displays the synergy of NDVI map and CNG over Dorsten scene. The top row images show the ground truth of two detected targets.

Summarizing, we apply the following process to a real change detection application, using the IIM system:

1. Extraction of primitive features: As primitive features we consider the unsupervised change detection results, texture parameters and Principal Component Analysis (PCA), computed through a PCA process.

2. Clustering of features: The extracted primitive features are classified through the unsupervised k-means method.

3. Model selection: The relevant information extracted from the images and represented by clusters is fused by selecting a combination of models. Available models are: spectral signature at time 1 and time 2, PCA at time 1 and time 2, texture extraction at both times, texture PCA at time 1 and time 2, TSAD, NDVI, CNG and synergy of methods.

4. Interactive learning: Once models are selected and a random image has been chosen, the user provides positive and negative examples. The system returns a feedback to the user showing a target detection map.

Fig. 10.3 shows the results of different interactive learning iterations selecting different combination of models.

10.1.3 Approach Assessment

Once the target detection map is obtained, we need to evaluate our results, to see how accurate our method is. In order to quantify detected targets and false alarms, a mask for each dataset is needed.

The complexity of generating a mask for targets resides on the inspection of the data. The image analyst must intensively examine the data, looking for changes between both images or, at least, positioning the targets. In addition to this problem, we are further faced with the decision of grouping non relevant pixels in order to measure the number of detected false alarms.

In this section, we present a quantitative assessment method to evaluate the accuracy of the proposed approach for target detection.

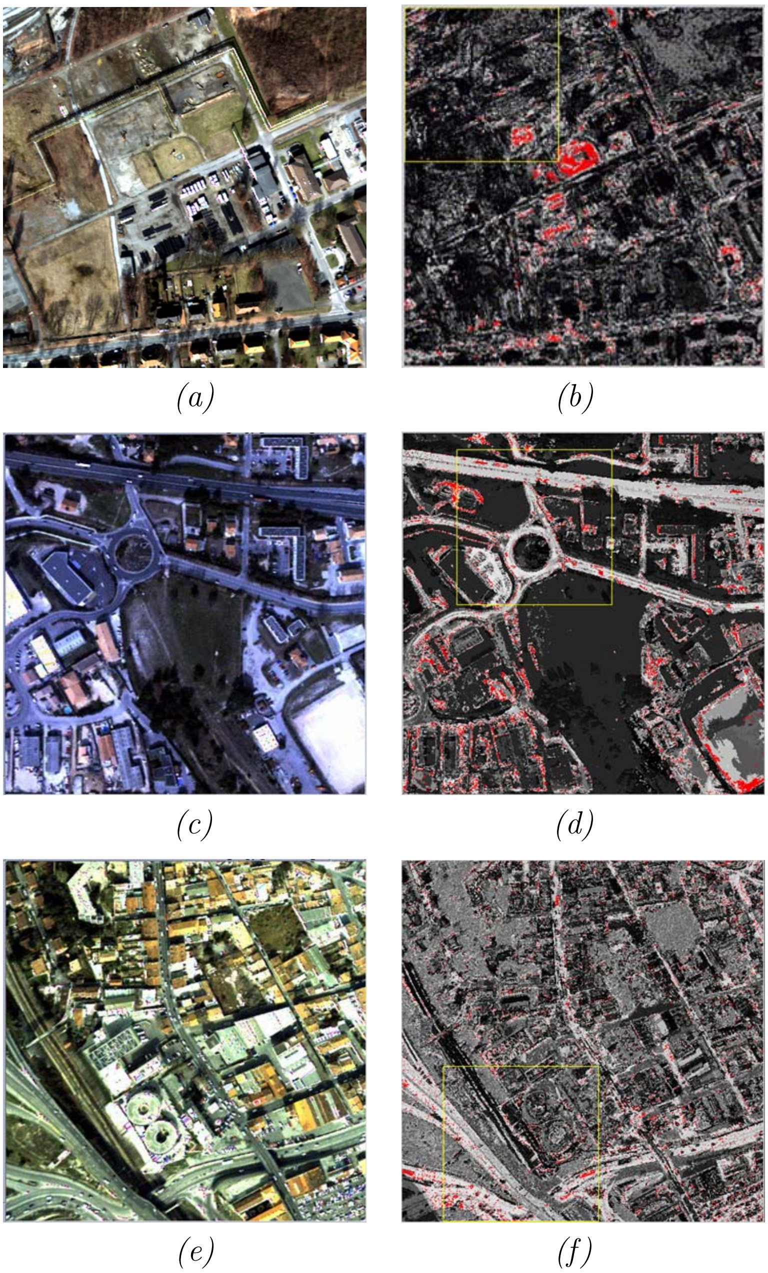
BUILDING MASKS

For a complete evaluation of the change detection algorithms, two masks are designed. If we were able to find an automatic way of building a target mask, the change detection problem would be solved. Consequently, the mask for targets is obtained by firstly, a visual inspection of the data, and secondly, by a manual annotation of the observed changes. Due to the difficulties of getting the appropriate pixel and because targets are significantly bigger than one pixel, a neighborhood is considered. Thus, the synthetic target is formed by a grid of 3x3 pixels.

On the other side, the Color Normalization (CNG) method is able to identify the strong false alarms contained in the data, as for instance, shadows on the roof of buildings or thick vegetation. Therefore, the resulting map of this approach can be considered as a good false alarm mask. Fig. 10.4 shows the resulting target and false alarm masks for Dorsten, Marseille and Borculo sites, that will be used in the evaluation procedure.

EVALUATION METHOD AND RESULTS

The evaluation protocol consists of a pixel based comparison between masks and results of the algorithms, measuring the number of detected targets and false alarms, acquiring a significative rate of goodness of the methods.



(a) *(b)*

(c) *(d)*

(e) *(f)*

Figure 10.3: Supervised change detection results. *(a)* Dorsten original site and *(b)* target detection map, obtained through an interactive learning process combining spectral PCA t_1, TSAD, synergy (TSAD map-CNG) and texture t_2; *(c)* Marseille original subset and *(d)* target detection map, obtained through an interactive learning process combining spectral PCA t_1, spectral PCA t_2, NDVI and synergy (NDVI map-CNG); *(e)* Marseille original subset and *(f)* target detection map, obtained through an interactive learning process, combining spectral PCA t_1, spectral PCA t_2, TSAD and synergy (TSAD map-CNG).

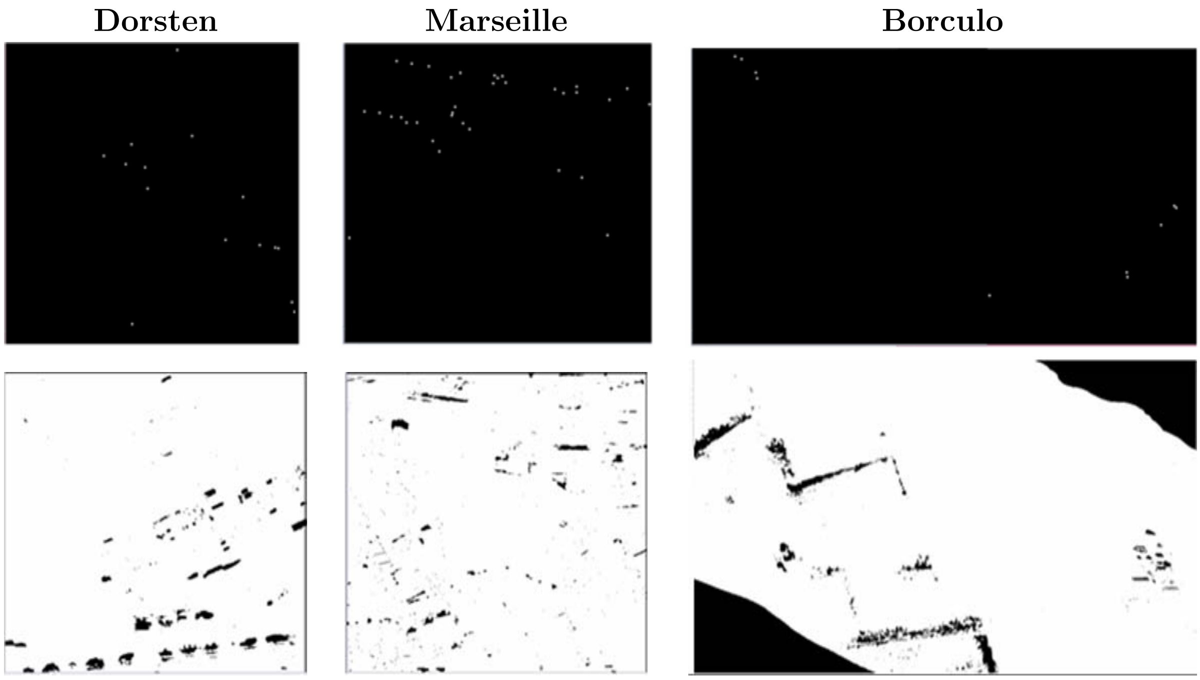


Figure 10.4: Evaluation masks. Images in upper row, from left to right: target mask of Dorsten, Marseille and Borculo sites. Bottom row, images from left to right: false alarm mask of Dorsten, Marseille and Borculo sites.

Basically, our evaluation approach is based on the Receiver Operating Characteristic (ROC) criterion that has been widely used in evaluating the performance of radar and sonar systems. To apply this criterion to geographical change detection problem, Kosugi [83] defined the probability of detection (PD) and the probability of false alarm (PF) as

$$PD = \frac{S_d}{S_c} \tag{10.1}$$

$$PF = \frac{S_f}{S_i} \tag{10.2}$$

where S_c is the number of pixels belonging to real targets, S_d the number of successfully detected targets, S_i the number of pixels covering unchanged areas, and S_f the number of pixels belonging to false alarms. In our assessment procedure, we apply a light variation of this criterion. We consider the number of detected targets and false alarms instead the number of pixels belonging to them. Thus, our probability of target detection (PTG) and probability of detection of false alarm (PFA) are defined as

$$PTG = \frac{number\_detected\_targets}{total\_number\_targets} \tag{10.3}$$

$$PFA = \frac{number\_detected\_FA}{total\_number\_FA} \tag{10.4}$$

In order to calculate the false alarm rate, we introduce the neighborhood order concept, which is the order of grouping neighbor pixels by counting false alarms. Therefore, the greater the order, the less the number of false alarms. Tab. 10.2 shows the number of false alarms (FA), contained in Dorsten and Borculo scenes, considering different orders of neighborhood. Analyzing this table, we observe that from order three to order seven the number of false alarms decreases drastically, meanwhile after order seven the differences are insignificant. With this results, we consider the order seven as a good measure for the evaluation.

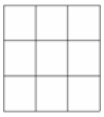
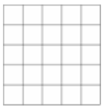
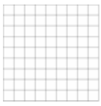
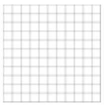
Neighboring Pixel	Order	Dorsten (FA)	Borculo (FA)
	3	135	258
	5	104	165
	7	87	98
	9	81	69
	11	68	60

Table 10.2: Neighborhood order for grouping false alarm pixels.

Tab. 10.3 shows the assessment results. Analyzing the Dorsten results, the TSAD algorithm can detect 80% of the total number of targets; however, the NDVI algorithm presents only a 66,7% of success. For the case of detection of false alarms, NDVI identifies less than the TSAD. The synergy of methods behaves as expected, reducing the false

	Dorsten		**Borculo**		**Marseille**	
Algorithms	PTG	PFA	PTG	PFA	PTG	PFA
TSAD	80%	35%	78%	40%	73,5%	60%
NDVI	66,7%	11%	50%	5%	100%	90%
Synergy (TSAD-CNG)	66,7%	0%	57%	0%	50%	0,41%
Synergy (NDVI-CNG)	66,7%	0%	50%	0%	11,8%	0%

Table 10.3: Evaluation of the unsupervised target detection approach applied over Dorsten, Borculo and Marseille scenes.

alarms rate despite of loosing some targets. Comparing the results for the Dorsten image case, NDVI and the combination of NDVI and CNG methods achieves the best target detection accuracy. The evaluation results in the Borculo site present a similar behavior, due to the vegetative nature of the scene. Consequently, a synergy with vegetation indexes provides the most accurate results. In the case of the Marseille image, the targets consist of cars and lorries, and no vegetation covers the surface. Observing the results of the application of the NDVI method, the target detection reaches the maximum, but also the detection of false alarms. This concludes that the NDVI algorithm or any synergy with it, is not applicable due to the lack of vegetation in the scene. In this case, the synergy with TSAD is considered as the best solution, despite the low probability of detection. The reasons of these unexpected results can be twofold: either the coregistration process has not been properly performed, and some spatio-temporal clutter still resides in the scene, or targets are very small and are placed nearby false alarms. In this case, during the synergy process, targets are considered as part of the false alarm neighbor and removed.

	Dorsten		**Borculo**		**Marseille**	
Models	PTG	PFA	PTG	PFA	PTG	PFA
Spectral PCA t_1 Spectral PCA t_2 TSAD Synergy (TSAD-CNG)	80%	14%	57%	7%	88,2%	13%
Spectral PCA t_1 Spectral PCA t_2 NDVI Synergy (NDVI-CNG)	80%	8%	70%	10%	94,1%	14%

Table 10.4: Evaluation of the supervised target detection approach applied over Dorsten, Borculo and Marseille scenes.

To assess the results of the supervised change detection approach, an evaluation protocol similar to the one used for the unsupervised case is applied. Tab. 10.4 shows the assessment results of the approach. We have to remark the drastic decrease of the number of detected false alarm respect to the unsupervised procedure due to the influence of negative examples provided over those areas during the interactive learning process. In this case, the accuracy of the results depends on the expertise of the user, and on the knowledge of the target location.

10.2 Validation of KIM Classification by User-defined Labels

A valuable characteristic of an IIM system is the ability to automatically classify semantic features for real application domains. This section aims at measuring the quality of semantic labels, defined by the user through a supervised classification, using KIM system.

10.2.1 Meris Data Product

We select a dataset of nine images, acquired by the MERIS sensor. For the evaluation, two types of MERIS products are provided: Meris product level 1b, a product that will be ingested in our system for supervised classification and Meris product level 2, which is a provided classification product of Level 1b. To achieve this second product, physical parameters from the area have been studied (for processing details, visit [31]). In the following, we will call the Meris product level 2, Meris II. Meris II contains several flags, giving information about image content. We focus our attention on the following:

- l2_flags: this flag is only present in those images, containing more than two classes. Pixels are labeled with: 'cloud', 'water', 'land', 'ice' or 'glint', as example.
- cloud_type: this flag exists only in images, containing two classes, one of them being labeled as 'cloud' and the other as 'no cloud', as example.

Because the Meris II product has been processed, based on physical properties and later annotated manually, we consider it as a good reference for the assessment of the quality of the user-defined labels. We select nine images, distributed over the world (see Fig. 10.5) covering different types of areas and containing cloud, water, land or ice features.

10.2.2 Approach Assessment

The evaluation process is depicted in Fig. 10.6. It starts with a selection of Meris product level 1b, called Meris I, containing all kind of features. The Meris I product is classified in two different ways:

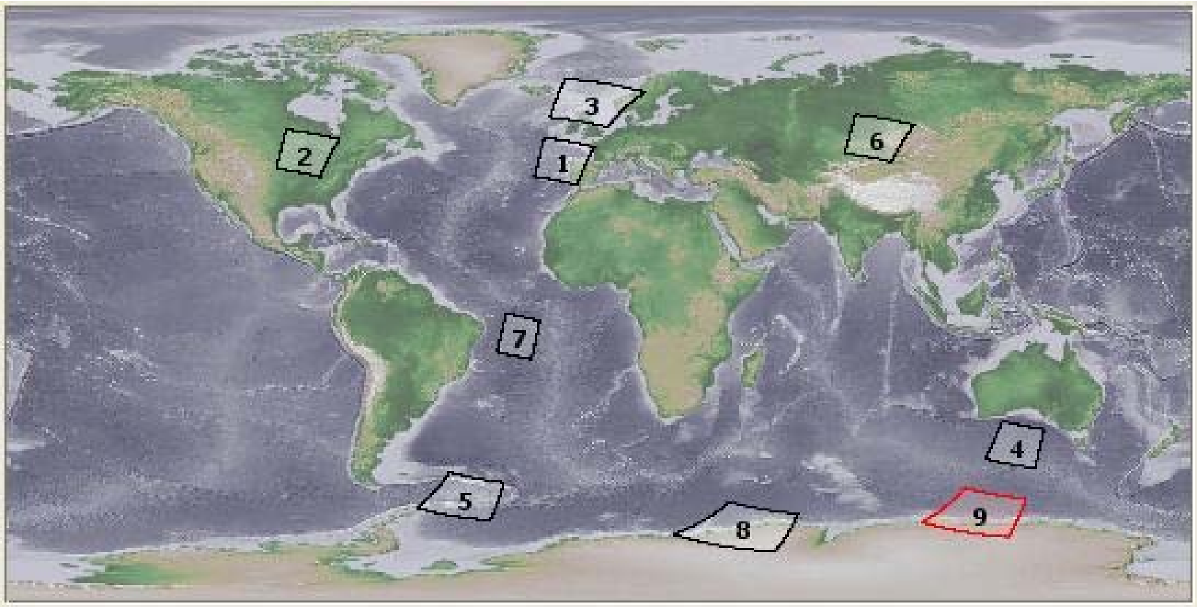


Figure 10.5: Location of Meris data products selected for the evaluation.

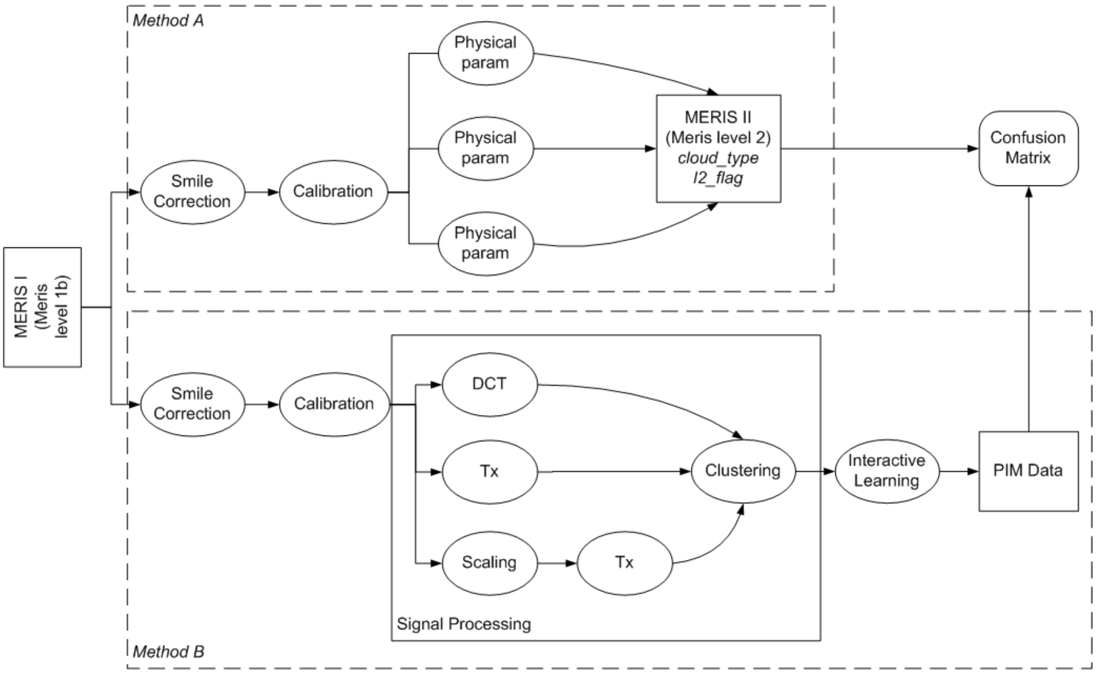


Figure 10.6: Evaluation process of Meris image classification.

1. Classification Method A: Based on physical parameters

   After performing two corrections, the *smile correction*, consisting of the normalization of the wavelengths within one spectral band to one reference wavelength, and the *radiometric conversion*, where radiances are converted to reflectances, physical parameters from the area are studied.

2. Classification Method B: Using KIM

   After performing the same correction as for classification method A, *simile correction* and calibration, primitive features are extracted: Discrete Cosine Transform (DCT), texture parameters (Tx) and texture at different scales. These extracted features are compressed by applying the dyadic k-means exploiting the spectral angular distance. Finally, a supervised learning process is performed, obtaining the classification maps.

Finally, we compare the two classifications by computing the confusion matrix across them.

10.2.3 Evaluation Results

From the evaluation of a dataset of nine Meris products, only two of them are presented here, one containing the l2_flag, and a second one considering the cloud_type.

CASE STUDY A: l2_flag

Fig. 10.7 displays on the left, the RGB of Meris level 1b product, covering the Michigan - Wisconsin area in the United States (area marked with number 2 in Fig. 10.5). The image on the right corresponds to the Meris II (l2_flag) classification product of the same scene. In this classification, we can clearly distinguish three features: water colored in red, cloud in green and land displayed in yellow.

Now, we perform the interactive learning in KIM for evaluating the classification capabilities of KIM. Fig. 10.8 shows three classifications corresponding to 'water', 'cloud' and 'land' features, respectively.

We compute the stochastic confusion matrix to assess the classification, obtained by KIM against Meris II product. Tab. 10.5 shows the confusion matrix between 'water', 'cloud' and 'land' features.

The results are interpreted as follows:

- cloud: 86,2% classified by KIM as 'cloud' corresponds to 'cloud' in Meris II product. The 13,8% remaining considered in KIM as 'cloud' is classified as 'land' in level 2 product.

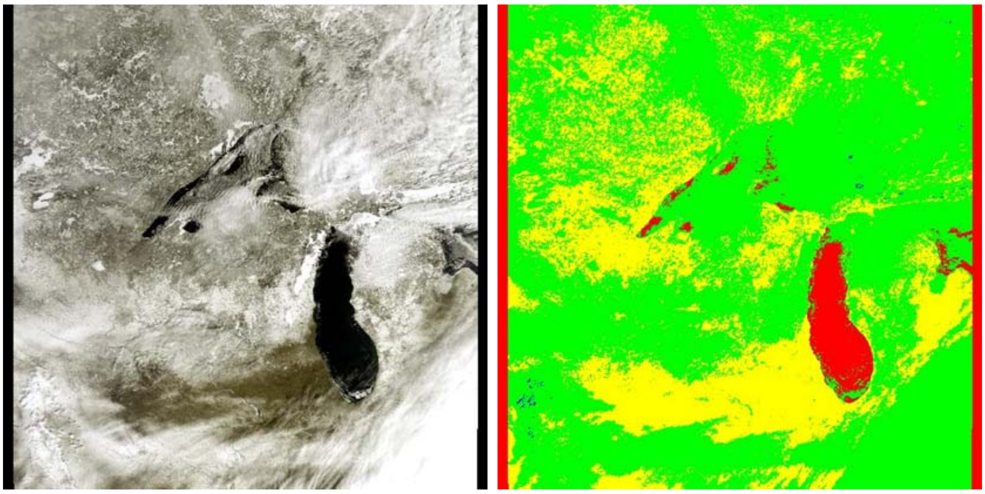


Figure 10.7: Classification method A. Images from left to right: RGB Meris level 1b product and Meris II classification product of the same scene. Classified features: 'water' (red), 'cloud' (green) and 'land' (yellow).

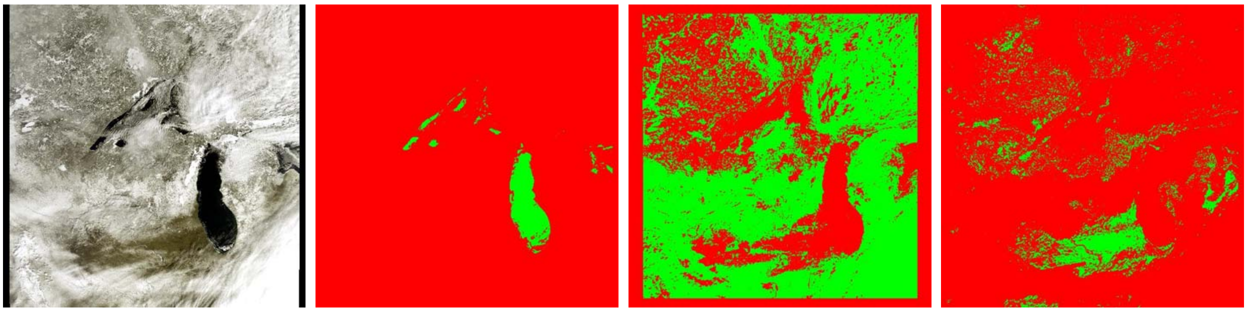


Figure 10.8: Classification method B. Images from left to right: Meris level 1b product and classification obtained by KIM of 'water', 'cloud' and 'land' features.

KIM labels	l2_flag classification		
	cloud	water	land
cloud	**86,2%**	0%	13,8%
water	7,4%	**92,6%**	0%
land	17,4%	0%	**82,6%**

Table 10.5: Case study A: Confusion matrix between Meris II product and KIM classification for 'cloud', 'water' and 'land' features.

- water: 92,6% classified by KIM as ’water’ corresponds to ’water’ in Meris II product. The 7,4% remaining considered in KIM as ’water’ is classified as ’cloud’ in level 2 product.

- land: 82,6% classified by KIM as ’land’ corresponds to ’land’ in Meris II product. The 17,4% remaining considered in KIM as ’land’ is classified as ’cloud’ in level 2 product.

CASE STUDY B: cloud_type

The second case study considers the cloud_type flag of the Meris II for measuring the accuracy of our classification method.

Fig. 10.9 shows the selected Meris level 1b product, covering a zone in the Indian Ocean in the south of Australia and the corresponding Meris II cloud_type. The scene is marked with number 4 in Fig. 10.5.

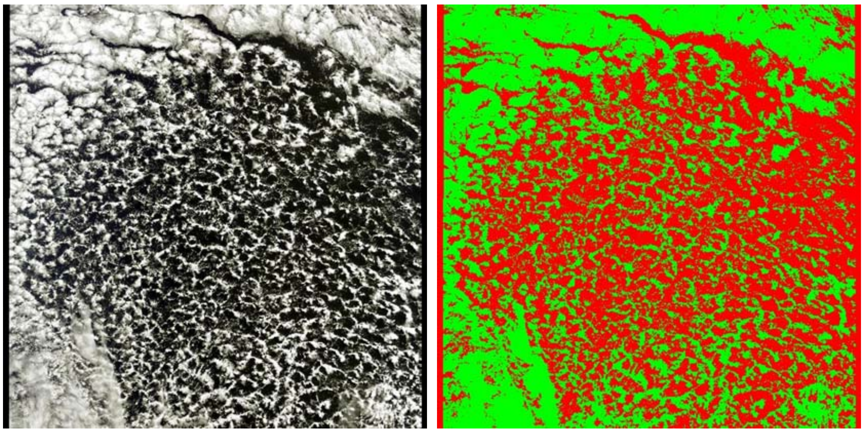


Figure 10.9: Classification method A. Images from left to right: RGB Meris level 1b product and Meris II classification product of the same scene. Classified features: ’cloud’ (green) and ’no cloud’ (red).

Performing interactive learning over this scene using KIM, the classification for ’cloud’ and ’water’ (classified as ’no cloud’ in Meris II) is very similar to Meris II (see Fig. 10.10).

Tab. 10.6 presents the confusion matrix between both classifications with the following meaning:

- cloud: 98% classified by KIM as ’cloud’ corresponds to ’cloud’ in Meris II product. The 2% remaining considered in KIM as ’cloud’ is classified as ’no cloud’ in level 2 product.

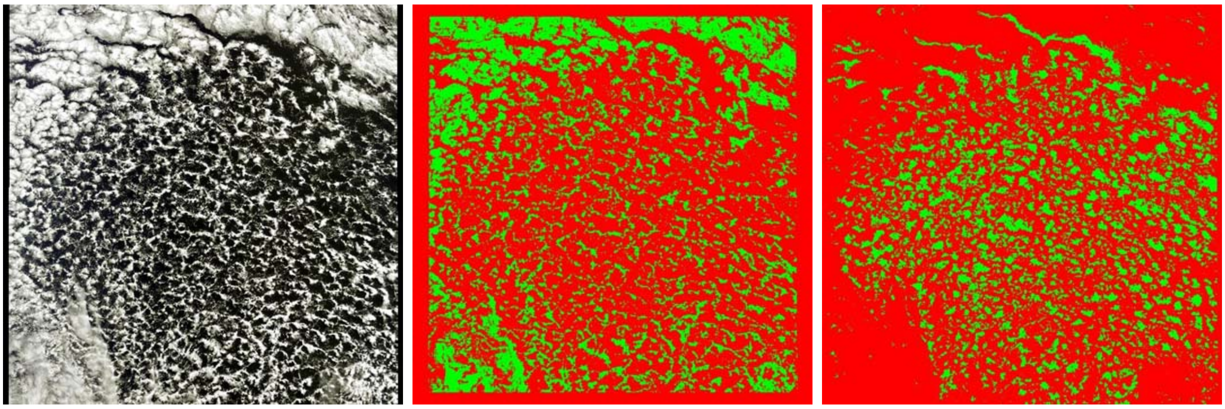


Figure 10.10: Classification method B. Images from left to right: Meris level 1b product and classification obtained by KIM of 'cloud' and 'water' features.

- water: 100% classified by KIM as 'water' corresponds to 'no cloud' in Meris II product. In this case, the classification done by KIM agrees 100% to Meris II product.

KIM labels	cloud_type classification	
	cloud	no cloud(water)
cloud	**98%**	2%
water (no cloud)	0%	**100%**

Table 10.6: Case study B: confusion matrix between Meris II product and KIM classification for 'cloud' and 'no cloud' (water) features.

CASE STUDY C: cloud_type

In the case studies A and B, the classification, obtained through interactive learning, using KIM is very similar to the Meris II product, obtained through a completely different procedure. The features, found in those cases are 'water', 'cloud' and 'land'. In this case study, we try to classify 'ice', using both methods.

For the experiment, a scene from the Antartic continent is chosen. Fig. 10.11 shows the Meris level 1b and level 2 l2_flag and cloud_type products. Fig. 10.12 displays the classification done by KIM of this scene. The Meris II l2_flag product distinguishes between two types of cloud, classifying real ice as cloud (visualized in green). Applying interactive learning, however ice is separated from cloud. Through a visual inspection of the original Meris products, the coast line can be identified, and distinguished from ice areas and clouds. Although ice is not fully detected (see right bottom corner of last

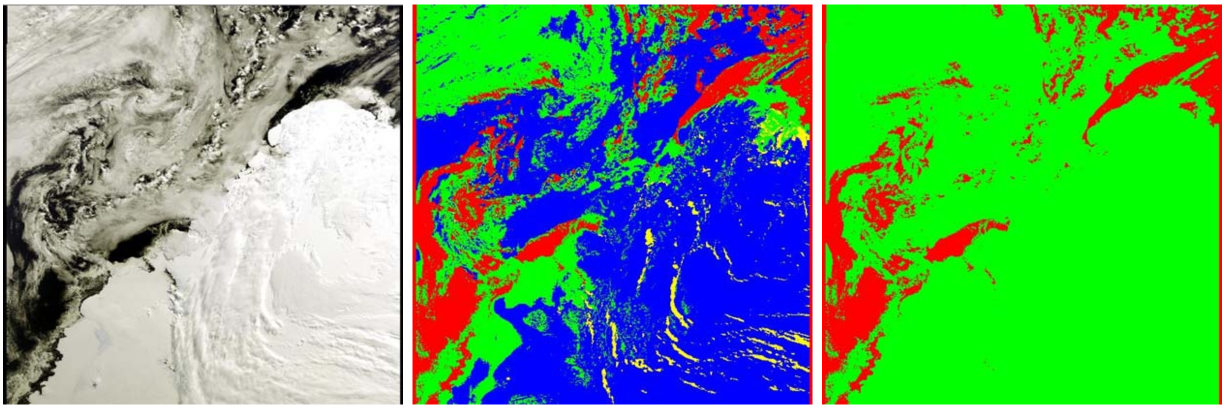


Figure 10.11: Classification method A. Images from left to right: RGB Meris level 1b product; Meris II l2_flag classifying 'cloud' type 1 (green), 'cloud' type 2 (blue) and 'water' (red); Meris II cloud_type classifying features as 'cloud' (green) and 'no cloud' (red).

image in Fig. 10.12) by KIM classification, it has not been misclassified as 'cloud', being considered as a better classification as the one provided by Meris II product.

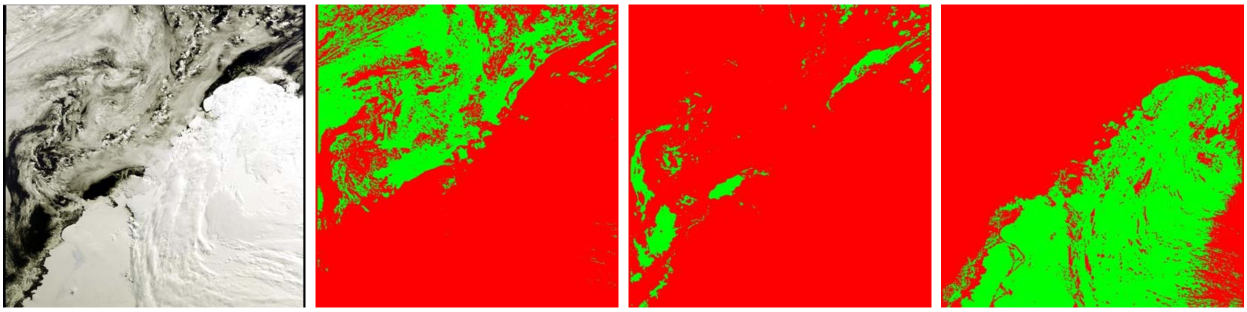


Figure 10.12: Classification method B. Images from left to right: Meris product level 1b, 'cloud', 'water' and 'ice' features.

Computing the confusion matrix, a 97,1% classified as 'water' by KIM coincides with the water feature in Meris II. The 2,9% remaining, is considered as 'cloud' by Meris II. For cloud and ice features, the confusion matrix between both products has no meaning, due to the lack of ice classification on Meris II product.

After the evaluation of KIM for classification of features, we extract the following conclusions:

- cloud and water features: the classification, given by training the system is more than 90% similar to Meris II and, therefore, considered as a good performance.

- land features: classification of land is not so similar, because it is affected by two facts:

    1. land can be covered by clouds: in this case, KIM can consider the scene as cloud instead of land.
    2. Land is a very general concept: land contains vegetation, mountains, flat areas, etc. Thus, if the area is covered by strong vegetation, the system will not be able to detect mountains.

- ice features: The Meris II product classifies ice as 'cloud', meanwhile KIM can separate snow or ice from cloud in two different classes.

With these results, we can conclude that KIM produces high quality classification products, useful in many application fields like environment, atmosphere effects or vegetation ones.

10.3 Conclusions

In this chapter, we have seen two examples of application domains of our IIM system, KIM:

- Multi temporal analysis: the first example focuses on the analysis of multi temporal data for target detection. With this example, we provide an evaluation method for the proposed target detection technique, and we demonstrate the ability of KIM for detecting small targets through supervised learning. We conclude, that the synergy between detection of targets and false alarms provides a good probability of detection, and a posterior use of KIM improves this probability.
- Classification analysis: with the second validation process, we have tested the classification power of KIM. Comparing the results of the KIM classification with the ones obtained through an extensive process of physical parameters analysis, we conclude that KIM is able to classify features with approximately the same accuracy. Besides, it can discriminate between features as ice and cloud, which are difficult to separate.

11

Conclusions

Exploring and explaining image data from huge Earth Observation archives has been an study object during the last decades. IIM and CBIR fields have been born to provide solutions for querying image databases by its content. To achieve this aim, advanced theories in image processing, feature extraction, data compression and semantic knowledge discovery are needed. On the other side, operational tools, based on high technology and human-centered concepts, where those theories will be integrated, are required.

To achieve these goals, a hierarchical approach with three levels of information abstraction is proposed: a data-driven, user-driven and knowledge-driven level. In the data-driven processing, the acquisition, pre-processing and archiving of the data is performed. Starting from the original image, features for target detection are extracted. In this dissertation, new algorithms for target detection by change analysis of multi temporal high resolution hyperspectral images have been investigated and implemented, being able to cope with situations of varying illumination conditions and strong background clutter. After extracting the features, they are grouped together through a totally unsupervised classification technique. A novel method to estimate the quality of the parameters for spectral and texture algorithms, using Information and Rate Distortion theories has been presented. At this level of information, a signal content index is created, using the cluster, image description and the type of stochastic model assumed. The user-driven processing is characterized by interactive learning, where the user is able to generate a supervised classification, taking into account the clusters and model types from the previous level. The user can also summarize acquired knowledge by defining different semantic labels and, applying Bayesian inference, retrieve images with similar content from the archive using probabilistic search. The development of the novel multiple classifier enables semantic learning over more than one feature by using a Dirichlet model.

Finally, the previous methods, based on the Bayesian hierarchy are assembled and integrated in an operative system. From the software architecture point of view, the system will be web service based, interoperable, standard compliant and reliable. The proposed design entails a modular architecture, orchestrated by a workflow engine, which decides which modules are executed when and under what circumstances. Modules communicate using the SOAP protocol, which permits to execute a web service, located in the external world. Therefore, other feature extraction methods or data providers can be easily plugged in the IIM system.

11.1 The Value of the Contributions

In this dissertation, we have analyzed the theory and application of several image processing methods and the IIM system, KIM. Based on this system, we have contributed in several disciplines, going from feature extraction to software design architecture, through parameter estimation theory, Bayesian hierarchy information representation, Information and Rate Distortion theory and machine learning. In addition to the individual contributions on each mining phase, this dissertation achieves a valuable content in system integration. The selection of best suitable software technologies and modular design separating the access to data, from the business logic, and the image processing time consuming algorithms from interactive real time interfaces. An optimal system architecture is the key point for an operable and useful IIM system.

Summarizing the value of the contributions of this thesis, we have to enumerate:

- Feature extraction: we have developed an automatic target detection algorithm for high resolution images. The multi temporal analysis has been performed under illumination dependent conditions and presence of clutter and misregistration on bitemporal images. The methodology is able to detect small targets through pixel based band combinations and synergy between target maps and strong shadows detection masks.
- Quality of parameters: after performing a clustering approach, the quality of some parameters can be measured, applying Information and Rate Distortion theories. In detail, we have analyzed, how the size of the analyzing window for a textural information extraction algorithm modifies the accuracy of the method, and therefore, with our approach, we try to estimate the optimal analyzing window size. Of course, the optimal size will depend on the spatial diversity of the image content.
- Machine learning: in the interactive learning phase of an IIM system, we have developed a novel multiple classifier. We have seen, how the Dirichlet prior for learning class assignation is well adapted to map generation. The multiple classifier helps in diverse content based image retrieval and in thematic map generation.
- Software architecture: an optimal software architecture is mandatory for an operable IIM system based on real time scenarios. The use of SOAP architecture helps in linking the signal processing Bayesian hierarchy model with heterogeneous technologies.

A

Data Characterization

1. Daedalus

   - Airborne Sensor
   - Number of spectral bands: 11
   - Spatial resolution: up to 0.8m
   - Spectral range: 0.4 - 14 μm
   - Bandwidth: 0.02 - 5 μm

Num. Band	Wavelenght
1	0.420 - 0.450
2	0.450 - 0.520
3	0.520 - 0.600
4	0.605 - 0.625
5	0.630 - 0.690
6	0.695 - 0.750
7	0.760 - 0.900
8	0.910 - 1.050
9	1.550 - 1.750
10	2.080 - 2.350
11	8.500 - 13.000

Table A.1: Spectral channels of the Daedalus sensor

2. Envisat

   - Satellite Sensor MERIS
   - Number of spectral bands: 15

- Spatial resolution: 300m
- Spectral range: 0.4 - 0.9 μm
- Bandwidth: 2.5 - 20 μm
- Type of product: Reduced Resolution Geophysical Product

Num. Band	Wavelenght(μm)	Bandwidth (μm)
1	0.4125	10
2	0.4425	10
3	0.49	10
4	0.51	10
5	0.56	10
6	0.62	10
7	0.665	10
8	0.68125	7.5
9	0.705	10
10	0.75375	7.5
11	0.76	2.5
12	0.775	15
13	0.865	20
14	0.89	10
15	0.9	10

Table A.2: Spectral channels of the Meris sensor

3. Landsat TM

- Satellite Sensor Thematic Mapper (TM)
- Number of spectral bands: 7
- Spatial resolution: 30m and 120m (band 6)
- Spectral range: 0.45 - 12.50 μm
- Type of product: geo-reference

4. SPOT 5

- Satellite Sensor HRVIR
- Number of spectral bands: 5
- Spatial resolution: 20m
- Spectral range: 0.5 - 1.75 μm

Num. Band	Wavelenght(μm)	Description
1	0.45 - 0.52	Blue
2	0.52 - 0.60	Green
3	0.63 - 0.69	Red
4	0.76 - 0.90	Near Infrared (NIR)
5	1.55 - 1.75	Shortwave Infrared
6	10.40 - 12.50	Thermal Infrared
7	2.08 - 2.35	Shortwave Infrared

Table A.3: Spectral channels of LandsatTM

Num. Band	Wavelenght(μm)	Description
XI1	0.50 - 0.59	Green
XI2	0.61 - 0.68	Red
XI3	0.79 - 0.89	Near Infrared (NIR)
XI4	1.53 - 1.75	Shortwave Infrared

Table A.4: Spectral channels of SPOT 5

- Type of product: geo-reference

5. Ikonos

- Satellite Sensor
- Number of spectral bands: 4
- Spatial resolution: 4m multispectral (MS), 1m pan-sharpened (PS) and 0.8m panchromatic (PAN)
- Spectral range: 0.445 - 0.9 μm

Num. Band	MS and PS Wavelenght(μm)	PAN (μm)	Description
1	0.445 - 0.516	0.45 - 0.90	Blue
2	0.506 - 0.595		Green
3	0.632 - 0.698		Red
4	0.757 - 0.853		Near Infrared (NIR)

Table A.5: Spectral channels of Ikonos

B

Theorems

B.1 Rate Distortion Theorem.

Lemma (Convexity of $R(D)$). The rate distortion function $R(D)$ is a non-increasing convex function of D.

Proof of the lemma. $R(D)$ is the minimum of the mutual information over increasingly larger sets as D increases. Thus $R(D)$ is non-increasing in D.
To prove that $R(D)$ is convex, consider two rate distortion pairs (R_1, D_1) and (R_2, D_2) which lie on the rate-distortion curve. Let the joint distribution that achieves these pairs be $p_1(x, \hat{x}) = p(x)p_1(\hat{x}|x)$ and $p_2(x, \hat{x}) = p(x)p_2(\hat{x}|x)$. Consider the distribution $p_\lambda = \lambda p_1 + (1-\lambda)p_2$. Since the distortion is a linear function of the distribution, we have $D(p_\lambda) = \lambda D_1 + (1-\lambda)D_2$. Mutual information, on the other hand, is a convex function of the conditional distribution and hence

$$I_{p_\lambda}(X;\hat{X}) \leq \lambda I_{p_1}(X;\hat{X}) + (1-\lambda)I_{p_2}(X;\hat{X}). \tag{B.1}$$

Hence by the definition of the rate distortion function,

$$\begin{aligned} R(D_\lambda) &\leq I_{p_\lambda}(X;\hat{X}) && \text{(B.2)}\\ &\leq \lambda I_{p_1}(X;\hat{X}) + (1-\lambda)I_{p_2}(X;\hat{X}) && \text{(B.3)}\\ &= \lambda R(D_1) + (1-\lambda)R(D_2), && \text{(B.4)} \end{aligned}$$

which proves that $R(D)$ is a convex function of D.

Proof of the Rate Distortion Theorem. In order to prove the converse of the theorem, we must show, for any source X drawn i.i.d. $\sim p(x)$ with distortion measure $d(x, \hat{x})$, and any $(2^{nR}, n)$ rate distortion code with distortion $\leq D$, that the rate R of the code satisfies $R \geq R(D)$.
Consider any $(2^{nR}, n)$ rate distortion code defined by functions f_n and g_n. Let $\hat{X}^n =$

$\hat{X}^n(X^n) = g_n(f_n(X^n))$ be the reproduced sequence corresponding to X^n. Then we have the following chain of inequalities:

$$
\begin{aligned}
nR &\overset{(a)}{\geq} H(\hat{X}^n) && \text{(B.5)}\\
&\overset{(b)}{\geq} H(\hat{X}^n) - H(\hat{X}^n|X^n) && \text{(B.6)}\\
&\overset{(c)}{=} I(\hat{X}^n; X^n) && \text{(B.7)}\\
&= H(X^n) - H(X^n|\hat{X}^n) && \text{(B.8)}\\
&\overset{(d)}{=} \sum_{i=1}^{n} H(X_i) - H(X^n|\hat{X}^n) && \text{(B.9)}\\
&\overset{(e)}{=} \sum_{i=1}^{n} H(X_i) - \sum_{i=1}^{n} H(X_i|\hat{X}^n, X_{i-1}, \ldots, X_1) && \text{(B.10)}\\
&\overset{(f)}{\geq} \sum_{i=1}^{n} H(X_i) - \sum_{i=1}^{n} H(X_i|\hat{X}_i) && \text{(B.11)}\\
&= \sum_{i=1}^{n} I(X_i; \hat{X}_i) && \text{(B.12)}\\
&\overset{(g)}{\geq} \sum_{i=1}^{n} R\Big(E\big[d(X_i, \hat{X}_i)\big]\Big) && \text{(B.13)}\\
&= n \sum_{i=1}^{n} \frac{1}{n} R\Big(E\big[d(X_i, \hat{X}_i)\big]\Big) && \text{(B.14)}\\
&\overset{(h)}{\geq} nR\bigg(\frac{1}{n} \sum_{i=1}^{n} E\big[d(X_i, \hat{X}_i)\big]\bigg) && \text{(B.15)}\\
&\overset{(i)}{=} nR\Big(E\big[d(X^n, \hat{X}^n)\big]\Big) && \text{(B.16)}\\
&= nR(D) && \text{(B.17)}
\end{aligned}
$$

where

(a) follows from the fact that there are at most $2^{nR}\hat{X}^n$ in the range of the encoding function,

(b) from the fact that $\hat{X}^n$ is a function of X^n and thus $H(\hat{X}^n|X^n) = 0$,

(c) from the definition of mutual information,

(d) from the fact that X_i are independent,

(e) from the chain rule for entropy,

(f) from the fact that conditioning reduces entropy,

(g) from the definition of the rate distortion function,

(h) from the convexity of the rate distortion function (Lemma) and Jensen's inequality, and

(i) from the definition of distortion for blocks of length n.

This shows that the rate R of any rate distortion code exceeds the rate distortion function $R(D)$, evaluated at the distortion level $D = E[d(X^n, \hat{X^n})]$.

B.2 de Bruijn's identity

Theorem (de Bruijn's identity: Entropy and Fisher information). Let $H(\cdot)$ denote the differential entropy and $J(\cdot)$ the Fisher's Information. Let X be any random variable with a finite variance with a density $f(x)$, and Z an independent normally distributed random variable with zero mean and unit variance. Then

$$\frac{\partial}{\partial t}H(X+\sqrt{t}Z)=\frac{1}{2}J(X+\sqrt{t}Z) \tag{B.18}$$

In particular, if the limit exist as $t \to 0$

$$\lim_{t\to 0}\frac{\partial}{\partial t}H(X+\sqrt{t}Z)=\frac{1}{2}J(X) \tag{B.19}$$

This theorem is really important because it shows the relationship between the differential entropy and the Fisher's information.

Proof. Let $Y_t = X + \sqrt{t}Z$. Then the density of Y_t is

$$g_t(y)=\int_{-\infty}^{+\infty} f(x)\frac{1}{\sqrt{2\pi t}}e^{-\frac{(y-x)^2}{2t}}dx. \tag{B.20}$$

Then

$$\frac{\partial}{\partial t}g_t(y)=\int_{-\infty}^{+\infty} f(x)\frac{\partial}{\partial t}\left[\frac{1}{\sqrt{2\pi t}}e^{-\frac{(y-x)^2}{2t}}\right]dx \tag{B.21}$$

$$=\int_{-\infty}^{+\infty} f(x)\frac{1}{\sqrt{2\pi t}}\left[-\frac{1}{2t}+\frac{(y-x)^2}{2t^2}\right]e^{-\frac{(y-x)^2}{2t}}dx \tag{B.22}$$

$$\frac{\partial}{\partial y}g_t(y)=\int_{-\infty}^{+\infty} f(x)\frac{1}{\sqrt{2\pi t}}\frac{\partial}{\partial y}\left[e^{-\frac{(y-x)^2}{2t}}\right]dx \tag{B.23}$$

$$=\int_{-\infty}^{+\infty} f(x)\frac{1}{\sqrt{2\pi t}}\left[-\frac{y-x}{t}e^{-\frac{(y-x)^2}{2t}}\right]dx \tag{B.24}$$

$$\frac{\partial^2}{\partial y^2}g_t(y)=\int_{-\infty}^{+\infty} f(x)\frac{1}{\sqrt{2\pi t}}\frac{\partial}{\partial y}\left[-\frac{y-x}{t}e^{-\frac{(y-x)^2}{2t}}\right]dx \tag{B.25}$$

$$=\int_{-\infty}^{+\infty} f(x)\frac{1}{\sqrt{2\pi t}}\left[-\frac{1}{t}+\frac{(y-x)^2}{t^2}\right]e^{-\frac{(y-x)^2}{2t}}dx \tag{B.26}$$

Thus

$$\frac{\partial}{\partial t}g_t(y)=\frac{1}{2}\frac{\partial^2}{\partial y^2}g_t(y). \tag{B.27}$$

We will use this relationship to calculate the derivative of the entropy of Y_t, where the entropy is given by

$$H(Y_t)=-\int_{-\infty}^{+\infty} g_t(y)\ln g_t(y)dy. \tag{B.28}$$

Differentiating, we obtain

$$\frac{\partial}{\partial t}H(Y_t) = -\int_{-\infty}^{+\infty}\frac{\partial}{\partial t}g_t(y)dy - \int_{-\infty}^{+\infty}\frac{\partial}{\partial t}g_t(y)\ln g_t(y)dy \tag{B.29}$$

$$= -\frac{\partial}{\partial t}\int_{-\infty}^{+\infty} g_t(y)dy - \frac{1}{2}\int_{-\infty}^{+\infty}\frac{\partial^2}{\partial y^2}g_t(y)\ln g_t(y)dy. \tag{B.30}$$

The first term is zero since $\int g_t(y)dy = 1$. The second term can be integrated by parts to obtain

$$\frac{\partial}{\partial t}H(Y_t) = -\frac{1}{2}\left[\frac{\partial g_t(y)}{\partial y}\ln g_t(y)\right]_{-\infty}^{+\infty} + \frac{1}{2}\int_{-\infty}^{+\infty}\left[\frac{\partial}{\partial y}g_t(y)\right]^2\frac{1}{g_t(y)}dy. \tag{B.31}$$

The second term in (B.31) is $\frac{1}{2}J(Y_t)$. So the proof will be complete if we show that the first term in (B.31) is zero. We can rewrite the first term as

$$\frac{\partial g_t(y)}{\partial y}\ln g_t(y) = \left[\frac{\frac{\partial g_t(y)}{\partial y}}{\sqrt{g_t(y)}}\right][2\sqrt{g_t(y)}\ln\sqrt{g_t(y)}]. \tag{B.32}$$

The square of the first factor integrates to the Fisher information, and hence must be bounded as $y \to \pm\infty$. The second factor goes to zero since $x\ln x \to 0$ as $x \to 0$ and $g_t(y) \to 0$ as $y \to \pm\infty$. Hence the first term in (B.31) goes to 0 at both limits and the theorem is proved. The exchanging of integration and differentiation in (B.21), (B.23), (B.25), (B.29), can be strictly justify by the application of the bounded convergence and mean value theorems.

C

Acronyms

API	Application Programming Interfaces
BPEL	Business Process Execution Language (for Web services)
CBIR	Content-Based Image Retrieval
CPU	Central Processing Unit
CRLB	Cramér-Rao Lower Bound
CVA	Change Vector Analysis
DBMS	Database Management System
DFD	Deutsches Fernerkundungsdatenzentrum (German Remote Sensing Data Center)
DIMS	Data Information Management System (German Aerospace Center)
DLR	Deutsches Zentrum für Luft- und Raumfahrt (German Aerospace Center)
EJB	Enterprise JavaBean
EO	Earth Observation
ER	Entity Relationship
ESA	European Space Agency
ETHZ	Eidgenössische Technische Hochschule in Zürich (Swiss Federal Institute of Technology in Zurich)
FTP	File Transfer Protocol
GIS	Geographical Information System
GMRF	Gauss-Markov Random Field
GRF	Gibbs-Markov Random Field
GUI	Graphical User Interface
HTTP	HyperText Transfer Protocol
IIM	Image Information Mining

IMF	Institut für Methodik der Fernerkundung (Remote Sensing Technology Institute)
J2EE	Java 2 Enterprise Edition
JDBC	Java Database Connectivity
JRE	Java Runtime Environment
JWS	Java Web Start
KEO	Knowledge-centered Earth Observation
KES	Knowledge Enabled Services
KIM	Knowledge-driven Information Mining
KL	Kullback Leibler
LDAP	Lightweight Directory Access Protocol
MAP	Maximum A-Posteriori
MIT	Massachusetts Institute of Technology
ML	Maximum Likelihood
MMSE	Minimum Mean Square Error
MRF	Markov Random Field
ODBC	Open Database Connectivity
OGC	Open Geospatial Consortium
OODB	Object Oriented Database
PCA	Principal Component Analysis
PIM	Probabilistic Information Mining
RD	Rate Distortion
RMI	Remote Method Invocation
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SSE	Service Support Environment
SVM	Support Vector Machine
UDDI	Universal Description, Discovery and Integration
UML	Unified Modeling Language
URI	Uniform Resource Identifier
W3C	World Wide Web Consortium
WCS	Web Coverage Service
WFS	Web Feature Service
WMS	Web Map Service
WSDL	Web Service Definition Language
XML	eXtensible Mark-up Language

Bibliography

[1] Alexandria digital library. *http://www.alexandria.ucsb.edu/adl/*.

[2] Alexandria digital library gazetteer server client. *http://webclient.alexandria.ucsb.edu/client/gaz/adl/index.jsp*.

[3] http://earthobservatory.nasa.gov/library/measuringvegetation/.

[4] Ngda globetrotter. *http://clients.alexandria.ucsb.edu/globetrotter/*.

[5] Spectral signatures and multi-spectral image interpretation. note 8.

[6] K. Conradsen A. A. Nielsen and J. J. Simpson. Multivariate alteration detection (mad) and maf postprocessing in multispectral, bitemporal image data: New approaches to change detection studies. *http://dial.ucsd.edu/projects/mafmad/mafmad.html*.

[7] E. Martínez Izquierdo A. Arquero Hidalgo, C. Gonzalo Martín. *Teledetección: Una Aproximación Desde la Superficie al Satélite*. Fundación General de la Universidad Politécnica de Madrid, 2003.

[8] R. W. Picard A. Pentland and S. Sclaroff. Photobook: Content-based manipulation of image databases. *SPIE Storage and Retrieval Image and Video Databases II*, No. 2185, February 1994.

[9] S. Santini A. Gupta A. W. M. Smeulders, M. Worring and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22, No. 12:1349–1380, 2000.

[10] S. Aksoy. *A probabilistic similarity framework for content-based image retrieval*. PhD thesis, University of Washington, 2001.

[11] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[12] L. Buzzone and D. Fernández Prieto. Automatic analysis of the difference image for unsupervised change detection. *IEEE Trans. on Geoscience and Remote Sensing*, 38, May 2000.

[13] L. Buzzone and D. Fernández Prieto. Unsupervised retraining of a maximum likelihood classifier for the analysis of multitemporal remote sensing images. *IEEE Trans. on Geoscience and Remote Sensing*, 39:456–460, February 2001.

[14] L. Buzzone and D. Fernández Prieto. An adaptative semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images. *IEEE Trans. on Image Processing*, 11, April 2002.

[15] F. J. Seinstra C. G. M. Snoek, J-M. Geusebroek and A. W. M. Smeulders. The semantic pathfinder: Using an authoring metaphor for generic multimedia indexing. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28, No. 10, October 2006.

[16] N. Maillot C. Hudelot and M. Thonnat. Symbol grounding for semantic image interpretation: From image data to semantics. *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05)*, 2005.

[17] G. J. Scott A. S. Barb C. H. Davis C. R. Shyu, M. Klaric and k. Palaniappan. Geoiris: Geospatial information retrieval and indexing system - content mining, semantics modeling and complex queries. *IEEE Trans. on Geoscience and Remote Sensing*, 45:839–852, April 2007.

[18] C. C. Chang C. W. Hsu and C. J. Lin. A practical guide to support vector classification. *www.csie.ntu.edu.tw/ cjlin/papers/guide/guide.pdf*, 2003.

[19] George C. Canavos. *Probabilidad y Estadística - Aplicaciones y Métodos.* Mc Graw Hill, 1988.

[20] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24, No. 5, May 2002.

[21] T. M. Cover and J. A. Thomas. Elements of information theory. *Wiley Series in Telecomunication.*

[22] I. J. Cox, T. V. Papathomas M. L. Miller, T. P. Minka, and P. N. Yianilos. The bayesiabn image retrieval system pichunter: Theory, implementation, and psychophysical experiments. *IEEE Trans. on Image Processing*, 9 No.1:20–37, 2000.

[23] H. Daschiel. *Advanced Methods for Image Information Mining System: Evaluation and Enhancement of User Relevance.* PhD thesis, Fakultät IV - Elektrotechnik und Informatik der Technischen Universität Berlin, July 2004.

[24] H. Daschiel and M. Datcu. Classification, semantic grouping and category learning of image content. *IEEE Trans. on Image Processing*, 2003.

[25] M. Datcu. Scene understanding from sar images. *Proceedings of the Int. Geoscience and Remote Sensing Symposium (IGARSS)*, 1:310–314, May 1996.

[26] M. Datcu and I. Gómez. High resolution sar and optical image fusion from mine risk map generation. *Proceedings of the Workshop PSIP 2005 on Physics in Signal and Image Processing*, Toulouse, France, February 2005.

[27] M. Datcu and k. Seidel. Human-centered concepts for exploration and understanding of earth observation images. *IEEE Trans. on Geoscience and Remote Sensing*, 43, No. 3:601–609, March 2005.

[28] M. Datcu, K. Seidel, and M. Walessa. Spatial information retrieval from remote sensing images. part A: information theoretical perspective. *IEEE Trans. on Geoscience and Remote Sensing*, 36:1431–1445, September 1998.

[29] M. Datcu and K. Seidel. New concepts for remote sensing information dissemination: query by image content and information mining. *Proceedings of IEEE Int. Geoscience and Remote Sensing Symposium (IGARSS)*, 3:1335–1337, 1999.

[30] DLR. Eoweb - earth observation data service. *http://eoweb.dlr.de:8080/servlets/template/welcome/entryPage.vm.*

[31] ESA. Level 2 products and algorithms. *http://envisat.esa.int/dataproducts/meris/ CNTR2-7.htm#eph.meris.prodalg.levc.*

[32] T. Finin F. Perich, A. Joshi and Y. Yesha. On data management in pervasive computing environments. *IEEE Trans. on Knowledge and Data Engineering*, 16, No. 5, May 2004.

[33] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. *Califorina Institute of Technology*, USA.

[34] ESA-EUSC 2006 Conference: Image Information Mining for Security and Intelligence. http://earth.esa.int/rtd/events/esa-eusc_2006/. *EUSC, Madrid (Spain)*, November 27-29 2006.

[35] S. D. Hordley G. D. Finlayson and M. S. Drew. Removing shadows from images. *School of Information Systems, University of East Anglia, UK, School of Computer Science, Simon Fraser University, Canada*, http://www.cs.sfu.ca/ mark/ftp/Eccv02/shadowlessblurb.pdf.

[36] W. Chang G. Sheikholeslami and A. Zhang. Semquery: Semantic clustering and querying on heterogeneous features for visual data. *IEEE Trans. on Knowledge and Data Engineering*, 14, No.5, Sept/Oct 2002.

[37] I. Gómez and M. Datcu. Mining image temporal changes. *Proceedings of the Workshop ESA-EUSC 2004 on Theory and Applications of Knowledge-Driven Image Information Mining with Focus on Earth Observation*, Madrid, Spain, March 2004.

[38] I. Gómez and M. Datcu. Target detection by change analysis of multitemporal high resolution hyperspectral images. *Proceedings of the Workshop PSIP 2005 on Physics in Signal and Image Processing*, Toulouse, France, February 2005.

[39] I. Gómez and M. Datcu. Concepts elaboration and system architectures for mining very large image archives. *Proceedings on Image Information Mining for Security and Intelligence*, Madrid, Spain, November 2006.

[40] I. Gómez and M. Datcu. A bayesian multi-class image content retrieval. *Proceedings of the Int. Geoscience and Remote Sensing Symposium (IGARSS)*, Barcelona, Spain, July 2007.

[41] I. M. Gomez and M. Datcu. Study comparison of feature extraction algorithms for feature map generation. *Proceedings of the Workshop ESA-EUSC 2005 Image*

Information Mining - Theory and Applications to Earth Observation, Frascati, Italy, October 2005.

[42] D. Maravall Gómez-Allende. *Reconocimiento de Formas y Visión Artificial.* RA-MA, 1993.

[43] G. G. Hazel. Object-level change detection in spectral imagery. *IEEE Trans. on Geoscience and Remote Sensing*, 39, March 2001.

[44] D. Heckerman. A tutorial on learning with bayesian networks. *Technical Report MSR-TR-95-06, Microsoft Research, Advanced Technology Division*, Available on the WWW, 1996.

[45] AutoClass Project Homepage. http://ic.arc.nasa.gov/ic/projects/bayes-group/autoclass. 1998.

[46] H. Greenspan J. Goldberger and S. Gordon. Unsupervised image clustering using the information bottleneck method. *Pattern Recognition, 24th DAGM Symposium*, Zurich, Switzerland, September 2002.

[47] W. Pedrycz K. Cios and R. Swiniarski. Data mining methods for knowledge discovery. *Kluwer Academic Publishers.*

[48] C. Tusk K. Kopersky, G. Marchisio and S. Aksoy. Interactive models for semantic labeling of satellite images. *Proceedings of Earth Observing Systems VII*, 4814:423–434, 2002.

[49] S. Aksoy K. Kopersky, G. Marchisio and C. Tusk. Visimine: Interactive mining in image databases. *Proceedings of the Int. Geoscience and Remote Sensing Symposium (IGARSS)*, Toronto, Canada, 2002.

[50] S. A. Khayam. The discrete cosine transform (dct): Theory and application. *Department of Electrical and Computer Engineering*, Michigan State University, 2003.

[51] F. Kurz and M. Datcu. On the problematic of data integrity for communication of large eo repositories and infomration mining systems. *Proceedings of the Workshop ESA-EUSC 2005 Image Information Mining - Theory and Applications to Earth Observation*, Frascati, Italy, October 2005.

[52] A. Nayate J. Zheng L. Gao, M. Dahlin and A. Iyengar. Improving availability and performance with application-specific data replication. *IEEE Trans. on Knowledge and Data Engineering*, 17, No. 1, January 2005.

[53] F. J. Martín-Pliego López L. Ruíz-Maya Pérez. *Fundamentos de Inferencia Estadística.* AC, 2005.

[54] J. Li and R. M. Narayanan. Integrated spectral and spatial information mining in remote sensing imagery. *IEEE Trans. on Geoscience and Remote Sensing*, 42, No. 3, March 2004.

[55] Y. Li and T. Bretschneider. Remote sensing image retrieval using a context-sensitive bayesian network with relevance feedback. *Proceedings of the Int. Geoscience and Remote Sensing Symposium (IGARSS)*, 5:2461–2464, 2006.

[56] Y. Li and T. Bretschneider. Remote sensing image retrieval using a context-sensitive bayesian network with relevance feedback. *0-7803-9510-7/06 IEEE*, 2006.

[57] A. Pelizzari M. Quartulli A. Galoppo A. Colapicchioni M. Pastori K. Seidel P. G. Marchetti M. Datcu, H. Daschiel and S. D'Elia. Information mining in remote sensing images archives - part a: system concepts. *IEEE Trans. on Geoscience and Remote Sensing*, 41(12):2923–2936, 2003.

[58] K. Seidel M. Datcu and M. Walessa. Spatial information retrieval from remote sensing images: Part i. information theoretical perspective. *IEEE Trans. on Geoscience and Remote Sensing*, 36, No. 12, 1998.

[59] S. DElia M. Datcu, K. Seidel and P. G. Marchetti. Knowledge-driven information-mining in remote sensing image archives. *ESA Bulletin*, 110:26–35, 2002.

[60] W. Niblac J. Ashley Q. Huang B. Dom M. Gorkani J. Hafner D. Lee D. Petkovic D. Steele M. Flickner, H. Sawhney and P. Yanker. Query by image and video content: The qbic system. *IBM Almaden Research Center*, http://wwwqbic.almaden.ibm.com/, 1995.

[61] K. Seidel M. Schröder, H. Rehrauer and M. Datcu. Spatial information retrieval from remote sensing images: Part ii. gibbs markov random fields. *IEEE Trans. on Geoscience and Remote Sensing*, 36, No. 12, 1998.

[62] K. Seidel M. Schröder, H. Rehrauer and M. Datcu. Interactive learning and probabilistic retrieval in remote sensing image archives. *IEEE Trans. on Geoscience and Remote Sensing*, 38:2288–2298, 2000.

[63] D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms.* Cambridge University Press, 2003.

[64] B. S. Manjunath and W. Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18, No.8:837–842, 1996.

[65] ESA-EUSC 2008 Conference: Image Information Mining: pursuing automation of geospatial intelligence for environment and security. http://earth.esa.int/rtd/events/esa-eusc_2008/index.html. *ESRIN, Frascati (Italy)*, March 4-6 2008.

[66] O. Al-Kofahi R. J. Radke, S. Andra and B. Roysam. Image change detection algorithms: A systematic survey. *Department of Electrical, Computer, and Systemas Engineering Rensselaer Polytechnic Institute*, EDICS categories: 2-MODL, 2-ANAL, 2-SEQP, August 2004.

[67] J-Shanmugam R. M. Haralick and I. Dinstein. Texture features for image classification. *IEEE Trans. on Systems, Man, and Cybernetics*, 3:610–621, 1973.

[68] S. Ram and J. Park. Semantic conflict resolution ontology (scrol): An ontology for detecting and resolving data and schema-level semantic conflicts. *IEEE Trans. on Knowledge and Data Engineering*, 16, No. 2, February 2004.

[69] T. Randen. Brodatz textures. *http://www.ux.his.no/ tranden/brodatz.html.*

[70] IMEDIA Project research team. Ikona system. *http://www-rocq.inria.fr/cgi-bin/imedia/cbir-gen.cgi*, INRIA Rocquencourt, France, 2001.

[71] C. Tusk G. Marchisio S. Aksoy, K. Kopersky and J. C. Tilton. Learning bayesian classifiers for scene classification with a visual grammar. *IEEE Trans. on Geoscience and Remote Sensing*, 43, No. 3:581–589, march 2005.

[72] A. Quintanilla Rodenas S. Castao Fernández. *Teledetección: Avances y Aplicaciones.* Diputación de Albacete, 1999.

[73] V. P. Shah S. S. Durbha, R. L. King and N. H. Younan. Image information mining for coastal disaster management. *IEEE International Geoscience and Remote Sensing Symposium*, Barcelona, Spain, July 2007.

[74] E. Chuvieco Salinero. *Teledetección Ambiental.* Ariel, 2007.

[75] M. Schroder-Brzosniowsky. *Stochastic Modeling of Image Content in Remote Sensing Image Archives.* PhD thesis, Swiss Federal Institute of Technology, January 2000.

[76] Matteo Soccorsi. Rate distortion based analysis of image paramenters estimation for information mining. Master's thesis, Universit*à* Degli Studi di Roma Tor Vergata, October 2005.

[77] L. K. Soh and C. Tsatsoulis. Texture analysis of sar sea ice imagery using gray level co-occurrence matrices. *IEEE Trans. on Geoscience and Remote Sensing*, 37, No. 2, March 1999.

[78] A. Spătaru. *Fondements de la théorie de la transmission de l'information.* Presses polytechniques romandes, 1987.

[79] A. K. Ray T. Acharya. *Image Processing, Principles and Applications.* Wiley, 2005.

[80] ESA-EUSC 2004 Conference: Theory and with focus on Earth Observation Applications of Knowledge driven Image Information Mining. http://earth.esa.int/rtd/events/esa-eusc_2004/. *EUSC, Madrid (Spain)*, March 17-18 2004.

[81] ESA-EUSC 2005 Conference: Image Information Mining Theory and Applications to Earth Observation. http://earth.esa.int/rtd/events/esa-eusc_2005/. *ESRIN, Frascati (Italy)*, October 5-7 2005.

[82] A. B. Watson. Image compression using the discrete cosine transform. *Mathematica Journal*, 4, No.1:81–88, 1994.

[83] M. Fukunishi W. Lu T. Doihara Y. Kosugi, M. Sakamoto and S. Kakumoto. Urban change detection related to earthquakes using an adaptive nonlinear mapping of high-resolution images. *IEEE Trans. on Geoscience and Remote Sensing Letters*, 1, July 2004.

[84] A. C. She Y: Rui and T. S. Huang. A modified fourier descriptor for shape matching in mars. *Image Databases and Multimedia Search, Series on Software Engineering and Knowledge Engineering*, Ed. S. K. Chang, 1998.

[85] S. K. Mostéfaoui Z. Maamar and H. Yahyaoui. Toward an agent-based and context-oriented approach for web services composition. *IEEE Trans. on Knowledge and Data Engineering*, 17, No. 5, May 2005.

[86] K. Zhang and T. J. Sejnowski. Accuracy and learning in neural populations. *M.A.L.Nicolelis (Ed.) Progress in Brain Research Elsevier Science*, 130, 2001.

[87] S. Zhong and J. Ghosh. A unified framework for model-based clustering. *Machine Learning Research*, 4:1001–1037, 2003.

Acknowledgments

I want to special thank to my supervisor at DLR Prof. Dr.-Ing. Mihai Datcu. First, he offered me the opportunity to work at DLR introducing me in remote sensing and image information mining fields. Then, he offered and encouraged me to write a PhD, although I was not leaving in Germany. I am very fortunate to have worked together with such a competent researcher.

I am also very grateful to my second supervisor Prof. Dr.-Ing Otmar Loffeld, for accepting me as part of the IPP research team, and for giving me the opportunity to present this dissertation at Siegen University. Also many thanks to Dr.-Ing. Stefan Knedlik, for helping me with the administrative matters at the university.

From DLR, I would like to thank to Prof. Dr.-Ing. Manfred Schröder, who employed me in 2003 at IMF Institute, beginning my career in remote sensing activities. I also want to thank to all my colleges at image analysis group, specially to my two roommates, firstly Dr.-Ing Herbert Daschiel and later Dr.-Ing. Franz Kurz. Together, we spent very friendly time and large conversations in German. Vielen Dank für eure Geduld mit meinem Deutsch. Besides, I want to thank to my favorite Italian, Matteo Soccorsi, for his friendliness and contribution to Rate Distortion theory of this dissertation. Cyrille Maire, Houda Chaabouni, Marco Quartulli and Mariana Ciucu provided also a perfect environment to work.

Very special thanks to my boyfriend Kike for his patience, his help, thanks for teaching me technical basis, thanks for proofreading this dissertation, for adapting your holidays to my conferences, thanks for encouraging me every day to finish this PhD, thanks for believing in me, for being you, for being there. Gracias.

Quiero agradecer muy especialmente a mis padres y a mi hermana todo el esfuerzo, confianza y apoyo que siempre me han brindado. Sin ellos no habría llegado hasta aquí.