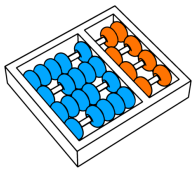
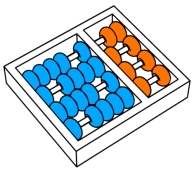


Jefersson Alex dos Santos

"Semi-Automatic Classification of Remote Sensing Images"

"*Classificação Semi-Automática de Imagens de Sensoriamento Remoto*"

CAMPINAS
2013



University of Campinas
Institute of Computing

Universidade Estadual de Campinas
Instituto de Computação

Jefersson Alex dos Santos

"Semi-Automatic Classification of Remote Sensing Images"

Supervisor: **Prof. Dr. Ricardo da Silva Torres**
Orientador(a):

Co-Supervisor: **Prof. Dr. Alexandre Xavier Falcão**
Co-orientador(a):

"Classificação Semi-Automática de Imagens de Sensoriamento Remoto"

PhD Thesis presented to the Post Graduate Program of the Institute of Computing of the University of Campinas to obtain a PhD degree in Computer Science.

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Computação da Universidade Estadual de Campinas para obtenção do título de Doutor em Ciência da Computação.

THIS VOLUME CORRESPONDS TO THE FINAL VERSION OF THE THESIS DEFENDED BY JEFERSSON ALEX DOS SANTOS, UNDER THE SUPERVISION OF PROF. DR. RICARDO DA SILVA TORRES.

Este exemplar corresponde à versão final da Tese defendida por Jefersson Alex dos Santos, sob orientação de Prof. Dr. Ricardo da Silva Torres.

Ricardo Torres

Supervisor's signature / *Assinatura do Orientador(a)*

CAMPINAS
2013

FICHA CATALOGRÁFICA ELABORADA POR
MARIA FABIANA BEZERRA MULLER - CRB8/6162
BIBLIOTECA DO INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E
COMPUTAÇÃO CIENTÍFICA - UNICAMP

Santos, Jefersson Alex dos, 1984-
Sa59c Classificação semi-automática de imagens de sensoriamento remoto / Jefersson Alex dos Santos. – Campinas, SP : [s.n.], 2013.

Orientador: Ricardo da Silva Torres.
Coorientador: Alexandre Xavier Falcão.
Tese (doutorado) – Universidade Estadual de Campinas, Instituto de Computação.

1. Reconhecimento de padrões. 2. Sensoriamento remoto. 3. Imagens de sensoriamento remoto. 4. Sistemas de computação interativos. 5. Processamento de imagens. I. Torres, Ricardo da Silva,1977-. II. Falcão, Alexandre Xavier,1966-. III. Universidade Estadual de Campinas. Instituto de Computação. IV. Título.

Informações para Biblioteca Digital

Título em inglês: Semi-automatic classification of remote sensing images
Palavras-chave em inglês:
Pattern recognition
Remote sensing
Images, Remote-sensing
Interactive computer systems
Image segmentation
Área de concentração: Ciência da Computação
Titulação: Doutor em Ciência da Computação
Banca examinadora:
Ricardo da Silva Torres [Orientador]
Sylvie Philipp Foliguet
Jocelyn Chanussot
William Robson Schwartz
Siome Klein Goldenstein
Data de defesa: 25-03-2013
Programa de Pós-Graduação: Ciência da Computação

TERMODEAPROVAÇÃO

Tese Defendida e Aprovada em 25 de Março de 2013, pela Banca examinadora composta pelos Professores Doutores:

Profª. Drª. Sylvie Philipp Foliguet
ENSEA / Université de Cergy-Pontoise

Prof. Dr. William Robson Schwartz
DCC / UFMG

Prof. Dr. Siome Klein Goldenstein
IC / UNICAMP

Prof. Dr. Franck Jocelyn Chanussot
GIPSA-Lab / GRENOBLE INP

Prof. Dr. Ricardo da Silva Torres
IC / UNICAMP

Institute of Computing /*Instituto de Computação*
University of Campinas /*Universidade Estadual de Campinas*

Semi-Automatic Classification of Remote Sensing Images

Jefersson Alex dos Santos[1]

March 25, 2013

Examiner Board/*Banca Examinadora*:

- Prof. Dr. Ricardo da Silva Torres (Supervisor/*Orientador*)
- Sylvie Philipp-Foliguet
  ENSEA, University of Cergy-Pontoise
- Jocelyn Chanussot
  ENSE3, Grenoble Institute of Technology
- William Robson Schwartz
  DCC, UFMG
- Siome Klein Goldenstein
  Institute of Computing, UNICAMP
- Silvio J. F. Guimarães (Instituto de Informática, PUC-Minas)
  Eduardo A. do Valle Junior (FEEC, UNICAMP)
  Neucimar J. Leite (Institute of Computing - UNICAMP)
  (Substitutes/*Suplentes*)

[1]Este projeto contou com apoio financeiro da FAPESP (processo 2008/58528-2) e Projeto CAPES-COFECUB 592/08 (BEX 5224101). This project was supported by FAPESP (PhD grant 2008/58528-2) and CAPES/COFECUB 592/08 (BEX 5224101).

To my dear wife Flávia

Resumo

Um grande esforço tem sido feito para desenvolver sistemas de classificação de imagens capazes de criar mapas temáticos de alta qualidade e estabelecer inventários precisos sobre o uso do solo. As peculiaridades das imagens de sensoriamento remoto (ISR), combinados com os desafios tradicionais de classificação de imagens, tornam a classificação de ISRs uma tarefa difícil. Grande parte dos desafios de pesquisa estão relacionados à escala de representação dos dados e, ao mesmo tempo, à dimensão e à representatividade do conjunto de treinamento utilizado.

O principal foco desse trabalho está nos problemas relacionados à representação dos dados e à extração de características. O objetivo é desenvolver soluções efetivas para classificação interativa de imagens de sensoriamento remoto. Esse objetivo foi alcançado a partir do desenvolvimento de quatro linhas de pesquisa.

A primeira linha de pesquisa está relacionada ao fato de embora descritores de imagens propostos na literatura obterem bons resultados em várias aplicações, muitos deles nunca foram usados para classificação de imagens de sensoriamento remoto. Nessa tese, foram testados doze descritores que codificam propriedades espectrais e sete descritores de textura. Também foi proposta uma metodologia baseada no classificador K-Vizinhos mais Próximos (K-nearest neighbors – KNN) para avaliação de descritores no contexto de classificação. Os descritores *Joint Auto-Correlogram* (JAC), *Color Bitmap*, *Invariant Steerable Pyramid Decomposition* (SID) e *Quantized Compound Change Histogram* (QCCH), apresentaram os melhores resultados experimentais na identificação de alvos de café e pastagem.

A segunda linha de pesquisa se refere ao problema de seleção de escalas de segmentação para classificação de imagens de sensoriamento baseada em objetos. Métodos propostos recentemente exploram características extraídas de objetos segmentados para melhorar a classificação de imagens de alta resolução. Entretanto, definir uma escala de segmentação adequada é uma tarefa desafiadora. Nessa tese, foram propostas duas abordagens de classificação multiescala baseadas no algoritmo *Adaboost*. A primeira abordagem, *Multiscale Classifier (MSC)*, constrói um classificador forte que combina características extraídas de múltiplas escalas de segmentação. A outra, *Hierarchical Multiscale Classifier (HMSC)*, explora a relação hierárquica das regiões segmentadas para melhorar a eficiência sem reduzir a qualidade da classificação

quando comparada à abordagem MSC. Os experimentos realizados mostram que é melhor usar múltiplas escalas do que utilizar apenas uma escala de segmentação. A correlação entre os descritores e as escalas de segmentação também é analisada e discutida.

A terceira linha de pesquisa trata da seleção de amostras de treinamento e do refinamento dos resultados da classificação utilizando segmentação multiescala. Para isso, foi proposto um método interativo para classificação multiescala de imagens de sensoriamento remoto. Esse método utiliza uma estratégia baseada em aprendizado ativo que permite o refinamento dos resultados de classificação pelo usuário ao longo de interações. Os resultados experimentais mostraram que a combinação de escalas produzem melhores resultados do que a utilização de escalas isoladas em um processo de realimentação de relevância. Além disso, o método interativo obtém bons resultados com poucas interações. O método proposto necessita apenas de uma pequena porção do conjunto de treinamento para construir classificadores tão fortes quanto os gerados por um método supervisionado utilizando todo o conjunto de treinamento disponível.

A quarta linha de pesquisa se refere à extração de características de uma hierarquia de regiões para classificação multiescala. Assim, foi proposta uma abordagem que explora as relações existentes entre as regiões da hierarquia. Essa abordagem, chamada *BoW-Propagation*, utiliza o modelo *bag-of-visual-word* para propagar características ao longo de múltiplas escalas. Essa ideia foi estendida para propagar descritores globais baseados em histogramas, a abordagem *H-Propagation*. As abordagens propostas aceleram o processo de extração e obtém bons resultados quando comparadas a descritores globais.

Résumé

Un effort considérable a été fait dans le développement des systèmes de classification des images avec l'objectif de créer des cartes de haute qualité et d'établir des inventaires précis sur l'utilisation de la couverture terrestre. Les particularités des images de télédétection combinées avec les défis traditionnels de classification font de classification de ces images une tâche difficile. La plupart des problèmes sont liés à la fois à l'échelle de représentation des données, et ainsi qu'à la taille et à la représentativité de l'ensemble d'apprentissage utilisé.

L'objectif de cette thèse est de développer des solutions efficaces pour la classification interactive des images de télédétection. Cet objectif a été réalisé en répondant à quatre questions de recherche.

La première question porte sur le fait que les descripteurs d'images proposées dans la littérature obtiennent de bons résultats dans diverses applications, mais beaucoup d'entre eux n'ont jamais été utilisés pour la classification des images de télédétection. Nous avons testé douze descripteurs qui codent les propriétés spectrales et la couleur, ainsi que sept descripteurs de texture. Nous avons également proposé une méthodologie basée sur le classificateur KNN (K plus proches voisins) pour l'évaluation des descripteurs dans le contexte de la classification. Les descripteurs *Joint Auto-Correlogram* (JAC), *Color Bitmap*, *Invariant Steerable Pyramid Decomposition* (SID) et *Quantized Compound Change Histogram* (QCCH), ont obtenu les meilleurs résultats dans les expériences de reconnaissance des plantations de café et de pâturages.

La deuxième question se rapporte au choix de l'échelle de segmentation pour la classification d'images basée sur objets. Certaines méthodes récemment proposées exploitent des caractéristiques extraites des objets segmentés pour améliorer classification des images haute résolution. Toutefois, le choix d'une bonne échelle de segmentation est une tâche difficile. Ainsi, nous avons proposé deux approches pour la classification multi-échelles fondées sur le les principes du *Boosting*, qui permet de combiner des classifieurs faibles pour former un classifieur fort. La première approche, *Multiscale Classifier (MSC)*, construit un classifieur fort qui combine des caractéristiques extraites de plusieurs échelles de segmentation. L'autre, *Hierarchical Multiscale Classifier (HMSC)*, exploite la topologie hiérarchique de régions segmentées afin d'améliorer l'efficacité des classifications sans perte de précision par rapport au MSC. Les

expériences montrent qu'il est préférable d'utiliser des plusieurs échelles plutôt qu'une seul échelle de segmentation. Nous avons également analysé et discuté la corrélation entre les descripteurs et des échelles de segmentation.

La troisième question concerne la sélection des exemples d'apprentissage et l'amélioration des résultats de classification basés sur la segmentation multiéchelle. Nous avons proposé une approche pour la classification interactive multi-échelles des images de télédétection. Il s'agit d'une stratégie d'apprentissage actif qui permet le raffinement des résultats de classification par l'utilisateur. Les résultats des expériences montrent que la combinaison des échelles produit de meilleurs résultats que les chaque échelle isolément dans un processus de retour de pertinence. Par ailleurs, la méthode interactive permet d'obtenir de bons résultats avec peu d'interactions de l'utilisateur. Il n'a besoin que d'une faible partie de l'ensemble d'apprentissage pour construire des classificateurs qui sont aussi forts que ceux générés par une méthode supervisée qui utilise l'ensemble d'apprentissage complet.

La quatrième question se réfère au problème de l'extraction des caractéristiques d'un hiérarchie des régions pour la classification multi-échelles. Nous avons proposé une stratégie qui exploite les relations existantes entre les régions dans une hiérarchie. Cette approche, appelée *BoW-Propagation*, exploite le modèle de *bag-of-visual-word* pour propager les caractéristiques entre les échelles de la hiérarchie. Nous avons également étendu cette idée pour propager des descripteurs globaux basés sur les histogrammes, l'approche *H-Propagation*. Ces approches accélèrent le processus d'extraction et donnent de bons résultats par rapport à l'extraction de descripteurs globaux.

Abstract

A huge effort has been made in the development of image classification systems with the objective of creating high-quality thematic maps and to establish precise inventories about land cover use. The peculiarities of Remote Sensing Images (RSIs) combined with the traditional image classification challenges make RSI classification a hard task. Many of the problems are related to the representation scale of the data, and to both the size and the representativeness of used training set.

In this work, we addressed four research issues in order to develop effective solutions for interactive classification of remote sensing images.

The first research issue concerns the fact that image descriptors proposed in the literature achieve good results in various applications, but many of them have never been used in remote sensing classification tasks. We have tested twelve descriptors that encode spectral/color properties and seven texture descriptors. We have also proposed a methodology based on the K-Nearest Neighbor (KNN) classifier for evaluation of descriptors in classification context. Experiments demonstrate that Joint Auto-Correlogram (JAC), Color Bitmap, Invariant Steerable Pyramid Decomposition (SID), and Quantized Compound Change Histogram (QCCH) yield the best results in coffee and pasture recognition tasks.

The second research issue refers to the problem of selecting the scale of segmentation for object-based remote sensing classification. Recently proposed methods exploit features extracted from segmented objects to improve high-resolution image classification. However, the definition of the scale of segmentation is a challenging task. We have proposed two multiscale classification approaches based on boosting of weak classifiers. The first approach, *Multiscale Classifier (MSC)*, builds a strong classifier that combines features extracted from multiple scales of segmentation. The other, *Hierarchical Multiscale Classifier (HMSC)*, exploits the hierarchical topology of segmented regions to improve training efficiency without accuracy loss when compared to the MSC. Experiments show that it is better to use multiple scales than use only one segmentation scale result. We have also analyzed and discussed about the correlation among the used descriptors and the scales of segmentation.

The third research issue concerns the selection of training examples and the refinement of classification results through multiscale segmentation. We have proposed an approach for

interactive multiscale classification of remote sensing images. It is an active learning strategy that allows the classification result refinement by the user along iterations. Experimental results show that the combination of scales produces better results than isolated scales in a relevance feedback process. Furthermore, the interactive method achieves good results with few user interactions. The proposed method needs only a small portion of the training set to build classifiers that are as strong as the ones generated by a supervised method that uses the whole available training set.

The fourth research issue refers to the problem of extracting features of a hierarchy of regions for multiscale classification. We have proposed a strategy that exploits the existing relationships among regions in a hierarchy. This approach, called *BoW-Propagation*, exploits the bag-of-visual-word model to propagate features along multiple scales. We also extend this idea to propagate histogram-based global descriptors, the *H-Propagation* method. The proposed methods speed up the feature extraction process and yield good results when compared with global low-level extraction approaches.

Acknowledgements

I wish to acknowledge all the people and institutions that contributed to the completion of this work. This thesis was only possible due to their support.

I thank my advisors Ricardo, Sylvie, Falcão, and Philippe. They gave me, each in their own way, notions that are essential for success in both scientific career and life. In a few words, I can only try to describe what I learned from them.

Ricardo has the power of motivation, I could say that "he gives wings to those who desire to fly". My sincere gratitude for giving me mine.

I would say that Sylvie is a facilitator. I am grateful to her valuable assistance and insistence, which were essential to the implementation of the Joint PhD Agreement (*Convencion de Cotutelle*).

Falcão is the "idea factory". He always has a lot of ideas that make people able to reason and guide the research. It is not recommended to talk to him without a notepad.

Philippe has taught me the importance of generalizing ideas. I thank him for making theoretical rigor meaningful.

I take to this opportunity to thank all the faculty members from the UNICAMP and ENSEA-UCP that have helped the development of this research somehow.

Thanks also go to friends and colleagues from Brazil and France for the priceless collaboration.

I owe particular thanks to FAPESP (PhD grant 2008/58528-2) and CAPES/COFECUB 592/08 (BEX 5224101) for the financial support.

Last but not least, I thank all my family, especially my wife and my mother.

My wife Flávia has always been by my side, in joy and in sorrow, especially during the four years required for the accomplishment of this work.

My mother, a constant source of strength, has showed me the value of knowledge, and encouraged me in this career since the beginning of this journey.

List of Abbreviations and Acronyms

ACC	Color Autocorrelogram
BIC	Border/Interior Pixel Classification
BPT	Binary Partition Tree
BoW	Bag of visual Words
CBC	Color-Based Clustering
$CBERS$	China-Brazil Earth Resources Satellite
$CCOM$	Color Co-Occurrence Matrix
CCV	Color Coherence Vector
$CGCH$	Cumulative Global Color Histogram
CM	Chromaticity Moments
CSD	Color Structure Descriptor
$CW\text{-}HSV$	Color Wavelet HSV
GCH	Global Color Histogram
$GEOBIA$	Geographic Object-Based Image Analysis
GIS	Geographic Information System
$GLCM$	Gray Level Co-Occurrence Matrix
GP	Genetic Programming
$HMSC$	Hierarchical Multiscale Classifier
HTD	Homogeneous Texture Descriptor
$IHMSC$	Interactive Hierarchical Multiscale Classifier
JAC	Joint Auto-Correlogram
KNN	K-Nearest Neighbor
LAS	Local Activity Spectrum
LBP	Local Binary Pattern
LCH	Local Color Histogram
LLC	Locality-constrained Linear Coding
MLC	Maximum Likelihood Classification
MSC	Multiscale Classifier
OA	Overall Accuracy

$QCCH$	Quantized Compound Change Histogram
RBF	Radial Basis Function
RSI	Remote Sensing Image
SAR	Synthetic Aperture Radar
SID	Invariant Steerable Pyramid Decomposition
$SIFT$	Scale-Invariant Feature Transform
$SPOT$	System for Earth Observation
SVM	Support Vector Machines

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Since the satellite imagery information became available to the civil community in the 1970s, a huge effort has been made on the creation of high quality thematic maps to establish precise inventories about land cover use [117]. However, the peculiarities of Remote Sensing Images (RSIs) combined with the traditional image classification challenges have turned RSI classification into a hard task.

The use of RSIs as a source of information in agribusiness, for example, is very common. In those applications, it is fundamental to know and monitor the land-use. However, identification and recognition of crop regions in remote sensing images are not trivial tasks. Classification of RSIs meets some specific issues in agriculture. This work is part of a Brazilian project involving a cooperative of coffee producers. It aims, among other applications, at finding the coffee plantations in remote sensing images. Concerning the identification of coffee areas, the difficulties come from the fact that coffee usually grows in mountainous regions (as in Brazil). First, this causes shadows and distortions in the spectral information, which make difficult the classification and the interpretation of shaded objects in the image because the spectral information is either reduced or totally lost [126]. Second, the growing of coffee is not a seasonal activity, and, therefore, in the same region, there may be coffee plantations of different ages, which also affect the observed spectral patterns. However, to be more general, we did not limit ourselves to this kind of images and we will present other applications, such as pasture and urban areas recognition.

The common approaches to implement RSI classification systems can be divided into two groups: pixel-based and object-based methods. Pixel-based methods have always been very popular for RSI classification [117]. They only use the value of the pixel in each band as a spectral signature to perform the classification. Indeed, concerning hyperspectral images, it is possible to associate a detailed spectral signature with each pixel, whose dimensions usually

correspond to large areas. In spite of that, high resolution data representation cannot rely only on pixels, because their image characteristics are not usually enough to capture the patterns of the classes (regions of interest).

To the extent that high-resolution images became accessible, new approaches of representation and feature extraction have been proposed in order to make better use of data [62]. Several methods using region-based analysis, also called GEOBIA (Geographic Object-Based Image Analysis), presented improvements in results when compared with traditional methods based on pixels [72]. The main problem of region-based approaches is their dependence on good algorithms of *segmentation*.

Traditionally, the objective of *image segmentation* is to partition the image into groups of pixels [39], called regions. The *scale of segmentation* refers to the size of the regions. It is usually defined by the input parameters of the used algorithm. Thereby, it is possible to create finer segmentations (with small regions) or coarser ones (with large regions) just by changing input parameters. Defining the most appropriate scale of segmentation is still a challenging research topic in remote sensing area [97, 2, 84, 25].

Recently, several multiscale segmentation methods have been proposed [42, 103, 37, 115, 59, 58, 8, 2, 54]. Some of them propose to represent the image according to a hierarchical structure [42, 37, 2, 54]. In these cases, the segmentation result is not a single image, but a hierarchy of regions. Figure 1.1 illustrates an example of hierarchy of segmented regions. The base of the hierarchy is composed of the smallest regions and the top is composed by the coarsest regions.

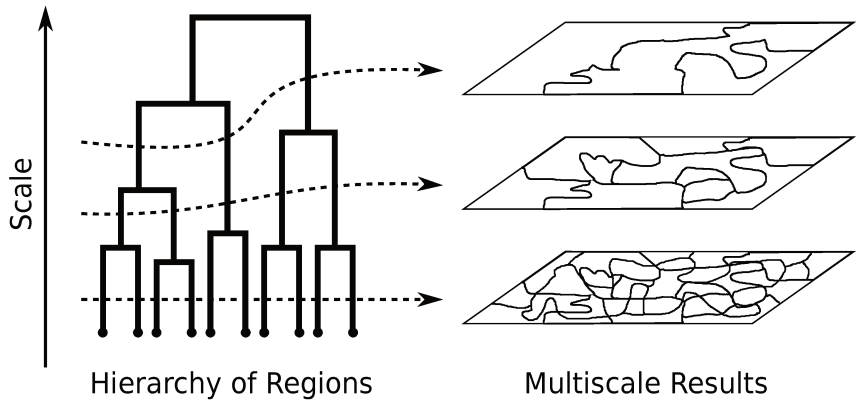


Figure 1.1: An example of a hierarchy of segmented regions.

Some approaches have been proposed to overcome the segmentation scale problem by selecting regions of different sizes during segmentation [104]. Other methods exploit regions from different scales [10, 8, 51, 114, 84, 25], or select a single scale representation from a hierarchy of segmented regions [97].

A particular case of region-based strategy is the plot-based approach [43, 86, 75], which

exploits cartographic information to define region boundaries. The main advantage is that the use of cartographic data enables a better delineation of user's interest objects than automatic segmentation techniques. Its main problem is the lack of available cartographic data. Thus, most of the papers related to the plot-based approach focus on urban applications.

Regarding the *training* process, there are many research challenges that concern the labeling of samples. The most important ones are related to the size and redundancy of the training set [101]. The size and quality of the training set have a direct impact on the execution time needed for training and on the final result of the classification. In addition, labeling of samples often requires visits to the study site, which can add extra costs to the analysis. The training set must, thus, be carefully chosen, avoiding redundancy patterns, but also ensuring a good representation of the considered classes. In order to assist users in selecting samples, several interactive methods have been proposed for dealing with remote sensing data [85, 100, 20, 24, 21, 99, 80, 91].

Typically, the classification process of RSIs uses supervised learning, which can be divided into three main steps: *data representation*, *feature extraction*, and *training*. *Data representation* indicates the objects for classification. *Feature extraction* provides a mathematical description for each object (by taking into account, for example, spectral characteristics, texture, shape). *Training* learns how to separate objects from distinct classes by building a classifier based on machine learning techniques (for instance, support vector machines [104], optimum-path forest [21], genetic programming [22], and neural networks [77]).

The final quality of the classification depends on the performance of each step as a whole. For example, the classification result relies on the accuracy of the employed learning techniques. Regarding the performance of learning algorithms, it is directly dependent on the quality of the extracted image features. Finally, features are extracted according to the model used for data representation.

1.2 Research challenges

The research challenges in remote sensing image classification can be arranged into three main axes as illustrated in Figure 1.2. These axes are based on the following aspects: data representation, target recognition, and user interaction.

The *data representation* axis concerns the kind of data which are considered as the samples in the classification process (e.g., pixels [89], blocks of pixels [22], regions [3], and hierarchy of regions [10]). In the following, we discuss some of the research challenges related to data representation:

- **Segmentation method:** there are several image segmentation strategies in the literature. The main challenge is to define the appropriate algorithms to segment the RSI into repre-

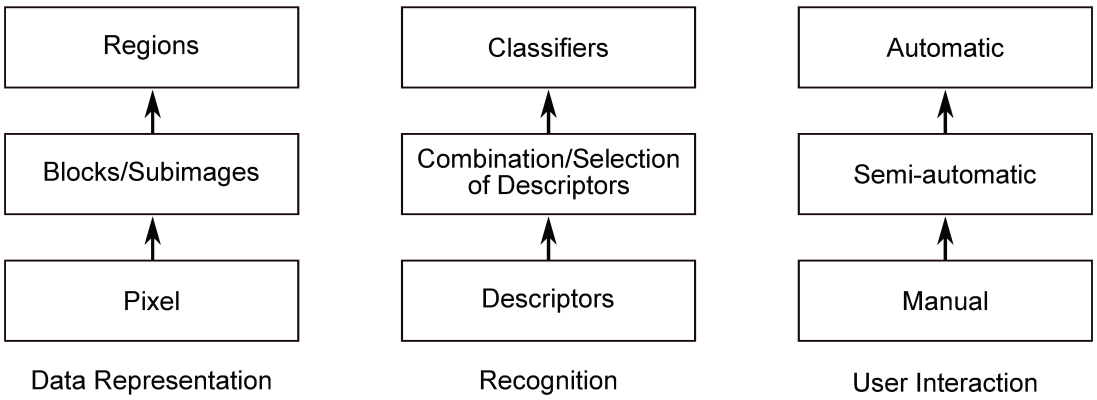


Figure 1.2: The remote sensing image classification research axes.

sentative regions, given a target application.

- **Selection of representation scale:** what is the best scale of representation? This question refers to: the window size in pixel-based classification; the number of pixels that defines the size of the blocks in a block-based classifier; or the size of the regions obtained by a segmentation algorithm.

- **Combination of different scale-dependent data:** Given a hierarchy of segmented regions, how to combine the different scales and/or to obtain a relevant hierarchy? Is it useful to combine distinct segmentation scales?

The *recognition* axis comprises the research challenges related to feature extraction and classification of samples. The feature extraction provides a mathematical description for each object (by taking into account for example, spectral characteristics, texture, or shape). The classification module is in charge of separating objects from distinct classes based on machine learning techniques.

- **Feature extraction:** are RGB-based color descriptors useful for representation of images composed of non-visible spectral bands? Which color and texture descriptors yield the best results for applications related to agriculture studies? How to extract features efficiently and effectively? Image descriptors developed for other applications may be useful to describe targets in remote sensing images. Moreover, the data representation adopted (a hierarchy of regions, for example) could require/allow more suitable feature extraction mechanisms.

- **Feature combination/selection:** how to automatically combine and select the best suitable descriptors for an application that exploits RSIs?

- **Fusion of classifiers:** given a set of classifiers, how to combine them to improve classification results? Good classifiers may not be correlated. The diversity measured in terms of the level of agreement of classifiers can be exploited to select and fuse them.

The *user interaction* axis refers to the challenges that are related to user interactivity over the classification process: manual, automatic, and semi-automatic. In a manual classification, the recognition is completely dependent on users' perceptions and decisions. This process typically consists of drawing the areas of interest in the RSI by using some software (e.g., Spring [11]). It often requires visits to the studied place to confirm obtained results. In automatic approaches, the user indicates the training set samples and some supervised method is used to classify the remaining samples given a learning process. The semi-automatic classification strategy does not only use supervised classification but also allows the user to refine the classification process along iterations.

- **Selection of training samples:** selecting representative training samples frequently requires to revisit the area under study. An effective strategy to select training samples for user annotation is important to avoid extra costs. In an interactive approach, the semantic information obtained from each user interaction needs to be associated with extracted features to improve the classification results. *Active learning* is a concept developed to address these issues. It is a machine learning strategy that allows the system to interactively query the user and, then, improve the training data.

- **Visualization/Annotation:** In a typical content-based image retrieval system with relevance feedback, a small set of images is shown to the user at each learning iteration. In a semi-automatic RSI application, it is desirable to show the entire image because the spatial relationship among the regions is informative for better user annotation. Since the image is large, another problem concerns the definition of strategies to call user attention to the selected regions that should be annotated.

The work developed in this thesis addresses some of those important research challenges.

1.3 Hypothesis, objectives, and contributions

In this thesis, we focus on the data representation and feature extraction problems with the objective of developing effective solutions for interactive classification of remote sensing images. This objective was accomplished based on the four validation hypotheses below:

1. Descriptors designed for general use in different applications are useful for classification of agricultural areas in RSIs.

2. Multiscale image segmentation may provide more useful information for RSI classification than simple image segmentation.

3. Active learning is an effective approach for interactive RSI multiscale classification as it enables the user to refine the classification results and it reduces the training data simultaneously.

4. The propagation of features from the finer scales to the coarser ones along the hierarchy of segmented regions may be more efficient and effective than the use of features extracted from each scale individually.

The first hypothesis concerns the use of successful image descriptors, which are developed for different purposes, in RSI classification tasks. It comes from the fact that image descriptors proposed in the literature achieve good results in various applications, but many of them have never been used in remote sensing classification tasks.

Our contribution concerning the first hypothesis comprises the evaluation of descriptors in the context of remote sensing image classification. We have tested twelve descriptors that encode spectral/color properties and seven texture descriptors for classification and retrieval tasks of coffee and pasture targets. To evaluate descriptors in classification tasks, we also propose a methodology based on the KNN classifier. Experiments demonstrate that Joint Auto-Correlogram (JAC) [118], Color Bitmap [63], Invariant Steerable Pyramid Decomposition (SID) [125] and Quantized Compound Change Histogram (QCCH) [44] yield the best results. These contributions were published in the *International Conference on Computer Vision Theory and Applications* (VISAPP) [28], in 2010.

The second hypothesis is related to the need for classification techniques of RSIs able to deal with images with high spatial resolution. Several recently proposed methods exploit features extracted from segmented objects. A common problem is the definition of the scale of segmentation. Moreover, by using a single segmentation scale, how could we insure the quality of this segmentation? We want to discover if the combination of multiple segmentation scales can achieve better results than using a single segmentation scale in isolation. Another important question is: how to perform multiscale classification without excessive computational costs? Finally, given classification results obtained with coarser segmentation, is it possible to refine the results by using finer segmentation scales?

The second contribution of this thesis, which refers to the second hypothesis, includes two boosted-based approaches for multiscale classification of remote sensing images. The first approach, Multiscale Classifier (MSC), builds a strong classifier that combines features extracted from multiple scales of segmentation. The other, Hierarchical Multiscale Classifier (HMSC), exploits the hierarchical topology of segmented regions to improve training efficiency without accuracy loss when compared to the MSC. We have shown that it is better to use multiple

scales than use only one segmentation scale result. We also analysed and discussed about the correlation among the used descriptors and the scales of segmentation. The MSC and HMSC approaches were published in the *IEEE Transactions on Geoscience and Remote Sensing* [25] (TGRS), in 2012. The correlation analysis was published in the *International Conference on Pattern Recognition* (ICPR) [23], in 2012.

The third hypothesis considers user interactions to aid both the refinement of classification results through multiscale segmentation and the selection of training examples. Some research questions are: how to select regions for the user feedback? How to take advantage of multiple scales without spending excessive time in training? Is it possible to achieve acceptable results with few user interactions?

The third contribution of this thesis is an approach for interactive multiscale classification of remote sensing images. We proposed an active learning strategy and adapted the HMSC to allow the classification result refinement by the user along interations with the system. The experimental results showed that the combination of scales produces better results than isolated scales in a relevance feedback process. Furthermore, the interactive method achieved good results with few user interactions. The proposed method needs only a small portion of the training set to build classifiers that are as strong as the ones generated by a supervised method that uses the whole training set. This contribution was reported in an article accepted for publication in the *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* (JSTARS) [26].

The fourth hypothesis refers to the problem of extracting features of a hierarchy of regions for multiscale classification. Our strategy relies on exploiting the existing relationship among the regions in a hierarchy. The challenge is how to use this information to speed up the feature extraction process without the quality of the generated representation.

Our contribution regarding the fourth hypothesis is an approach for efficient and effective feature extraction from a hierarchy of segmented regions. This approach, called BoW-Propagation, exploits the bag-of-visual words model to propagate features along multiple scales by taking into account the hierarchical relation among the regions of different scales. We also extended this idea to propagate histogram-based global descriptors, the H-Propagation. Experiments using the BoW-Propagation approach for feature extraction of arbitrary-shaped regions are presented in the *International Conference on Pattern Recognition* (ICPR) [29], in 2012. The H-Propagation was accepted for publication in the proceedings of the *IEEE International Geoscience and Remote Sensing Symposium* (IGARSS) [30], in 2012.

1.4 Organization of the text

This thesis is outlined according to the hypotheses. It is organized in eight chapters, including this introduction.

In Chapter 2, we review **the state-of-the-art on region-based and interactive classification for remote sensing images**. We also introduce **background concepts** related to the hierarchical segmentation method proposed by Guigues [42] and bag of visual words.

In Chapter 3, we present **the remote sensing image datasets** used in the experiments. We also explain how the experimental results are evaluated by using well-known **classification measures**.

In Chapter 4, we present the **evaluation of descriptors in the context of remote sensing image classification**.

In Chapter 5, we propose **two boosting-based approaches for multiscale classificaton of remote sensing images**. The first approach, **Multiscale Classifier (MSC)**, builds a strong classifier that combines features extracted from multiple scales of segmentation. The other, **Hierachical Multiscale Classifier (HMSC)**, exploits the hierarchical topology of segmented regions to improve training efficiency without accuracy loss when compared to the MSC. In this chapter, we also present a **correlation analysis among the used descriptors and the scales of segmentation**.

In Chapter 6, we propose **an approach for interactive multiscale classification of remote sensing images**. We proposed an active learning strategy and adapted the HMSC to allow the classification result refinement by the user along interations.

In Chapter 7, we present an **approach for efficient and effective feature extraction from a hierarchy of segmented regions**. The approach, **BoW-Propagation**, exploits the bag-of-visual words model to propagate features along multiple scales by taking into account the hierarchical relation among the regions of different scales. We also extended this idea to propagate histogram-based global descriptors, the **H-Propagation** approach.

Finally, in Chapter 8, **we present our conclusions and future perspectives**.

Chapter 2

Related Work and Background

In this chapter, we present related work, and the background concepts related to image representation and description necessary to understand the approaches we have proposed in this thesis. Section 2.1 presents related work. Section 2.2 presents the Guigues' segmentation algorithm, which we have used to obtain hierarchy of regions. Section 2.3 presents the low-level descriptors used along this thesis. Finally, Section 2.4 introduces basic concepts of the BoW approach.

2.1 Related Work

A study of published works between 1989 and 2003 examined the results and implications of RSI classification research [117]. According to this study, despite the high number of approaches in that period, there was not significative improvement in terms of classification results. Most of the proposed methods were pixel-based. These methods try to estimate the probability of each pixel to belong to the possible classes employing statistic measures based on spectral properties. The *Maximum Likelihood Classification* (MLC) [89] has remained as one of the most popular methods for RSI classification.

The improvements in sensor technologies increased the accessibility to high-resolution and hyperspectral imagery. As a result, new approaches were developed to make better use of the available data [62]. Two main research approaches to address those issues can be observed in the literature. The first one, which is more related to high-resolution images, focuses on data representation and feature extraction [57, 124, 48, 77, 114, 51, 104]. The other approach, more associated with pixel-based classification methods, is focused on issues related to the selection of samples for training and the inclusion of the user in the classification process [100, 99, 101, 85, 80, 20, 91].

In the next two subsections, we discuss each of these approaches. Concerning the first one, we highlight proposed methods for classification based on regions. Regarding the second ap-

proach, we point out proposed techniques related to interactive classification of remote sensing images.

2.1.1 Region-based Classification Methods

Initially, advances towards the classification of high-resolution data focused on the use of the neighborhood of the pixels in the analysis, which included texture descriptors [62].

More recently, many studies [57, 124, 48] have considered information encoded in regions (group of pixels) for RSI classification tasks. Gigandet et al. [38] proposed a classification algorithm for high resolution RSIs combining non-supervised and supervised classification strategies. In this method, regions were classified by using Mahalanobis distance and Support Vector Machines (SVM). Lee et al. [57] created a region-based classification method for high resolution images that exploited two approches: MLC with region means, and MLC with Gaussian Probability Density Function. Both works presented better results than pixel-based classifiers. Yu et al. [124] also proposed a method to classify RSI based on regions. The image segmentation and classification were performed by using evolution of fractal networks and non-parametric K-Nearest Neighbor (KNN), respectively. Another recent work in this research area has been developed by Katartzis et al. [48]. They proposed a region-based RSI classification method that uses Hierarchical Markov Models.

The growth of classification approaches based on regions has been analyzed in [4]. According to Blaschke et al., the goal of GEOBIA is to outline objects within images that are useful. It combines, at the same time, image processing and features of Geographic Information Systems (GIS) aiming to use spectral and contextual information seamlessly. The paper shows that the growth in the number of new approaches is accompanied by the increase of the accessibility to high-resolution images and, hence, the development of alternative techniques to the classification based on pixels. As pointed out by the authors, the growth in research involving GEOBIA was motivated in part by the use of commercial software eCognition [3]. The software has allowed research involving classification of regions, enabling the inclusion of data from different scales by using an approach supported on the KNN classifier.

These new trends have encouraged research studies that compare techniques based on pixels and/or regions [48, 126, 7, 72], and propose new segmentation techniques that support the classification of regions in RSIs [37, 115, 59, 8].

Likewise, new researches that take advantage of the use of multiple scales of data have been carried out [77, 114, 51, 102, 104, 107]. Both Ouma et al. [77] and Wang et al. [114] proposed approaches that use multiscale data for land cover change detection. In [77], Ouma et al. presented a technique for multi-scale segmentation with an unsupervised neural network for vegetation analysis. Wang et al. [114], on the other hand, proposed an approach for change detection in urban areas. The method relies on the fusion of features from different scales based

on a combination of means for each pixel in the used scales. The result is a new image which corresponds to the combination of the scales.

Like Wang et al. [114], Kim et al. [51] used the eCognition software to create the multi-scale segmentation. The objective, however, was to perform multi-class classification. In the segmentation process, the size of the regions is controlled by a scale parameter. For each scale, a different set of classes is defined according to a hierarchy between the classes of each scale. Thus, for each level, a different classification is performed. It includes structural knowledge and high semantic contents. The result of the coarsest scales is used for the classification of the most specific classes, restricting the regions that belong to the same subtree in the hierarchy.

Valero et al. [107] proposed a region-based hierarchical representation for hyperspectral images based on Binary Partition Tree (BPT). They show that the proposed Pruning BPT method can be suitable for classification. Furthermore, they mention that by using different prunings based on the same idea the method can be also used for filtering and segmentation purposes.

Tzotsos et al. [102, 104] used multiple scales for RSIs classification. In [102], they proposed a classification based on SVM with Gaussian Kernel that uses multi-scale segmentation. One single segmentation result is used for the extraction of objects by combining segments of various sizes. The size of the selected objects is controlled by a scale parameter as well. In [104], the authors proposed a method for the fusion of scales by nonlinear scale-space filtering. This technique avoids the use of parameters to control the creation of objects selected for classification.

2.1.2 Interactive Classification Methods

Several recent approaches handle the RSI classification problem by exploiting the user interactions [85, 100, 20, 24, 21]. The main purpose of these methods is to help the user to build a representative training set, improving classification results along iterations. According to Tuia et al. [101], in high-resolution imagery, the selection of training data can be easily done on the image. However, several neighboring pixels can be included in the selection, carrying the same spectral information. Consequently, the training set may be highly redundant. Furthermore, the labeling of training samples may require visits to the studied places, as those samples may be linked to geographical references. That adds extra costs to the classification process.

Most of the proposed methods are SVM-based [20, 80, 100, 99]. In these approaches, active learning plays a key role. It provides an interactive way to build training sets that correctly represent the boundaries of separation between classes, avoiding redundancies.

Pasolli et al. [80] proposed a classification method based on active learning for SVMs. The idea relies on classifying the samples as significant and non-significant, according to a concept of significance which is proprer to the theory of SVMs. Thus, a significance space is built, which is used to select samples to be displayed to the user. Demir et al. [20] investigated and tested

different active learning techniques in order to reduce the redundancy in the training set for pixel-based applications. Based on their analysis, it was proposed a new query function, called MCLU-CBD (Multiclass-Level Clustering with Uncertainty Based Diversity). This function uses the k-means clustering method in the kernel space and selects the most informative samples at each iteration according to the identification of the most uncertain sample of each cluster.

Tuia et al. have proposed strategies to perform active learning in remote sensing applications by using SVMs [100, 112, 99]. In [112], they proposed an active learning approach to minimize the redundancy of the sampled pixels and maximize the speed of convergence to an optimal classification accuracy. In [100], they presented two active learning algorithms for semi-automatic definition of training samples in RSI classification. They show that the training set can be 10% reduced by using the proposed method. In [99], they improve their method by applying sample clustering to the SVM margin samples.

Rajan et al. [85], on the other hand, proposed an approach based on active learning that can be applied to any classifier since this classifier is able to work with decision bounds. They apply the principle of selecting data points that most change the existing belief in class distributions.

Santos et al. [24, 21] have also recently proposed two interactive methods for classification of RSIs. In [24], they proposed an interactive framework based on relevance feedback, called GP_{SR}. That framework allows the classification of RSIs and the combination of distances from feature descriptors by using genetic programming (GP). In [21], they propose a new framework ($GOPF$) that integrates the Optimum-Path Forest classifier [78] and GP to perform interactive classification combining different types of features.

It is worth mentioning that none of the above cited methods are proposed to work on regions. The methods proposed in [85, 100, 80, 20] are based on feature extracted from pixels, some of them specifically focused on the classification of hyperspectral images [85, 80]. The methods proposed in [24, 21], in turn, use features extracted from regular blocks of pixels.

2.2 Hierarchical Segmentation

Lately, many multiscale segmentation methods have been proposed for remote sensing purposes [103, 37, 115, 59, 58, 8, 2, 54]. In this work, we use the scale-set representation introduced by Guigues et al. [42]. It builds a hierarchy of regions or a single suite of partitions. As the optimal partitioning of an image depends on the application, this method proposes to keep all partitions obtained at all scales, from the pixel level until the complete image.

Basically, we use the Guigues' approach because it is hierarchical (essential for our proposal) and it has a strong theoretical foundation. Anyway, the proposed approach for interactive multiscale classification is general and can exploit any other hierarchical region-based segmentation method.

Among other applications, this method has been successfully used in tasks of multiscale segmentation of remote sensing images by Trias-Sanz et al. [98]. They justify the use of Guigues's algorithm by the fact that it makes both the segmentation criterion and the scale parameter explicit. We concisely introduce the algorithm below.

Let image I be defined over a domain $\mathcal{D}$, a partition P is a division of $\mathcal{D}$ into separate regions. A partition P_2 is finer than a partition P_1 if each region R of P_2 is included in one and only one region of P_1. The scale-set representation consists in defining a set of partitions P_λ of $\mathcal{D}$, indexed by a scale parameter λ, such that if $\lambda_1 \leq \lambda_2$ then P_2 is finer than P_1. The transition between P_i and P_{i+1} is obtained by merging some adjacent regions of P_i into larger regions by optimizing a criterion. The criterion we use corresponds to Mumford-Shah energy [70], which approximates the color image by a piecewise constant function, while minimizing the edge lengths:

$$E(P) = \sum_{R_i \in P} E_D(R_i) + \lambda E_C(R_i) \tag{2.1}$$

where E_D is the distance with the piecewise constant model and E_C is the length of the contour.

The compromise between both constraints is defined by the parameter λ. For small values of λ, the image is over-segmented, the approximation of each region by a constant is perfect, but the total length of all edges is very large. On the contrary, when λ is large, the partition contains few regions (until only one), then the approximation of each region by a constant is poor, but the total length of all edges is very small. The set of partitions has a structure of a hierarchy H of regions: two elements of H, which are not disjoint, are nested. A partition P_λ is composed by the set of regions obtained from a cut in the hierarchy H at scale λ (see Figure 2.1). Guigues et al. showed that this algorithm can be performed with the worst case complexity in $O(N^2 logN)$, where N is the size of the initial over-segmentation.

The Guigues' algorithm is a merging process, which iteratively merges neighbouring regions by minimizing an energy criterion. It starts at pixel level, or after a watershed process, aiming to obtain regions more reliable to compute the energy. It stops when all regions are merged.

Figure 2.1 shows the segmentation structure obtained by Guigues' algorithm. The hierarchy of regions is drawn as a tree and the vertical axis is the scale axis (in logarithmic representation). A cut in scale λ retrieves a partition P_λ.

To automatically select partitions at different scales, Guigues et al. proposed the use of a dichotomous cutoff-based strategy, which consists of successively splitting the hierarchy of regions into two. Each division is a dichotomous cut and creates a partition at the defined scale.

Let Λ be the maximum scale in hierarchy H, i.e., the one in which the image I is represented by a single region, the cut-scale λ^c is defined by $\lambda^c = \Lambda/2^n$, where n is the order of each division in the hierarchy. Figure 2.2 presents some cuts extracted from the hierarchy illustrated

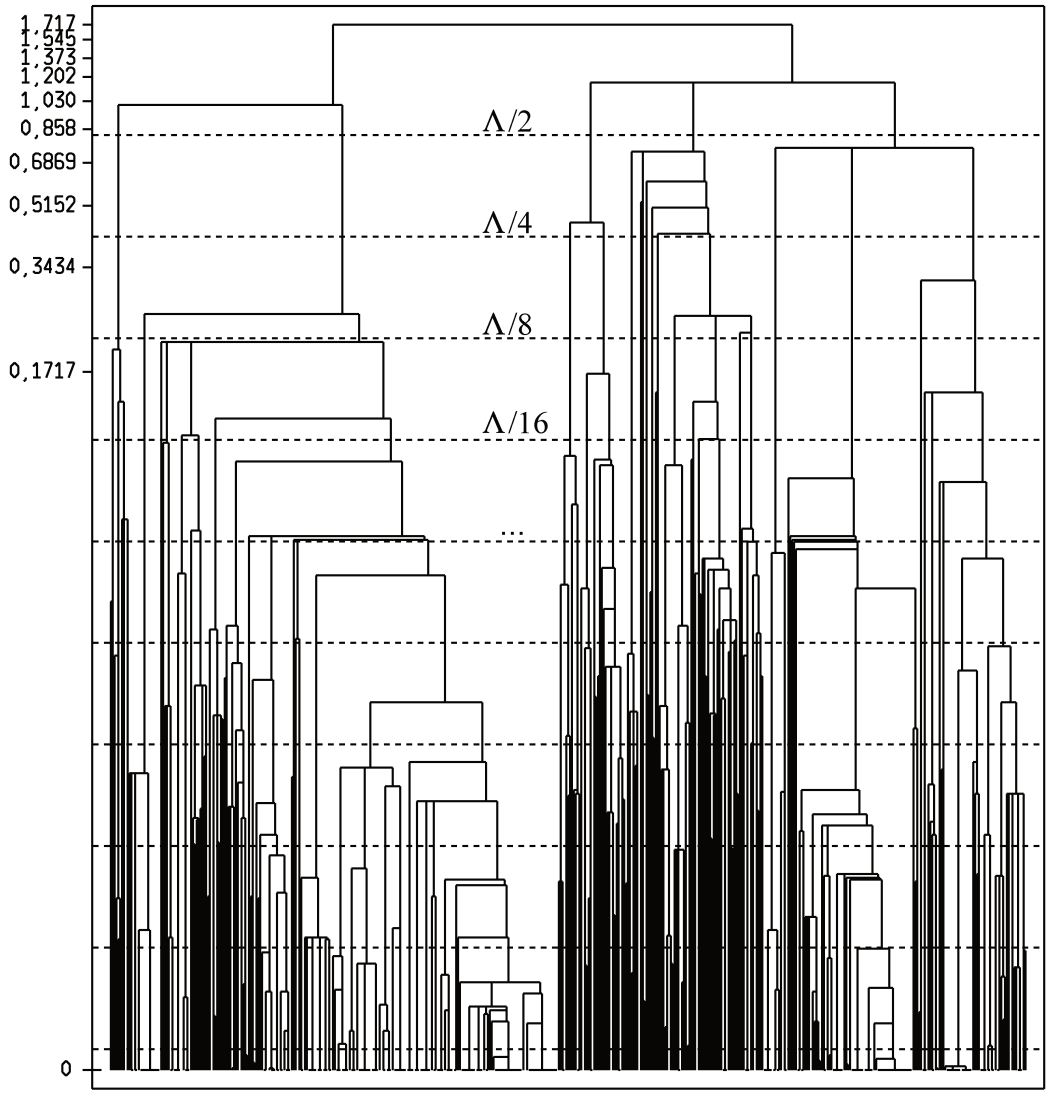


Figure 2.1: A scale-sets image representation. Horizontal axis: the regions. Vertical axis: the scales (logarithmic representation).

in Figure 2.1.

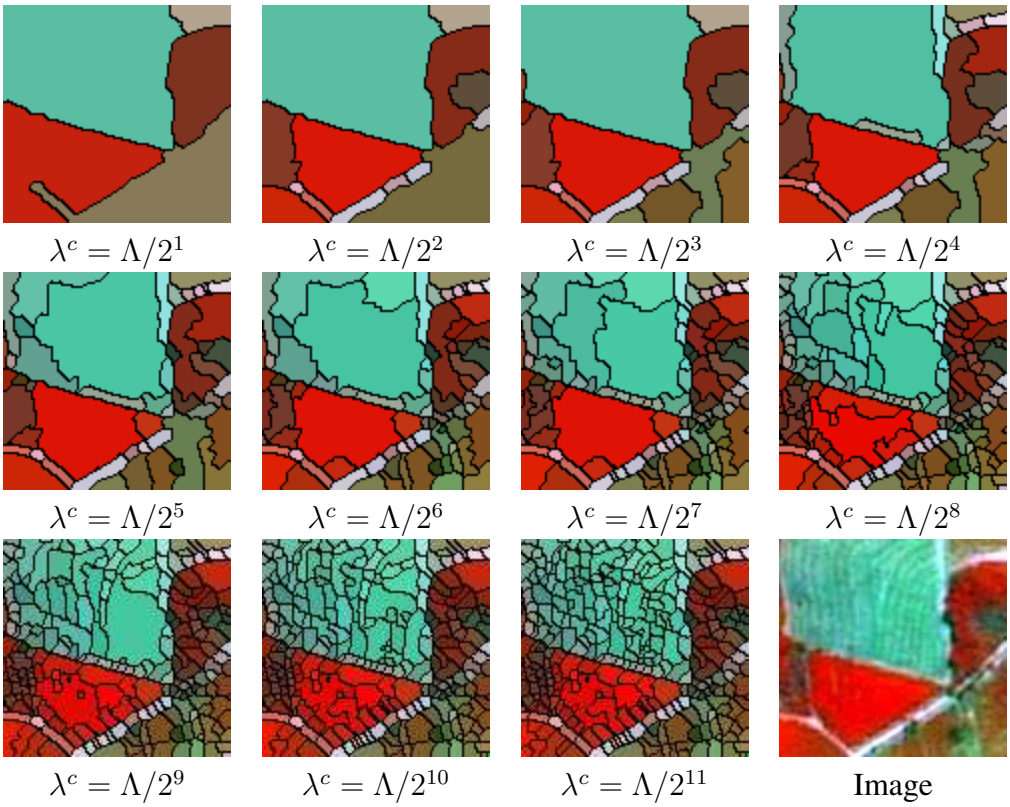


Figure 2.2: Some cuts of the scale-sets and the original image.

The highest scale of the hierarchy shown in Figure 2.1 is $\Lambda = 1.716$. Thus, the first cut is defined at the scale $\lambda^c = 0.858$, the second one, at the scale $\lambda^c = 0.429$, and so on.

2.3 Low-Level Descriptors

In this thesis, we have used nineteen low-level image descriptors. It encodes color/spectral properties (Section 2.3.1) and texture (Section 2.3.2).

2.3.1 Color Descriptors

This section presents the low-level color descriptors we have used in the experiments along this thesis.

Global Color Histogram (GCH) [94]

This is one of the most commonly used descriptors. It uses an extraction algorithm which quantizes the color space in a uniform way and it scans the image computing the number of pixels belonging to each color (bin). The size of the feature vector depends on the quantization used. In this work, the color space was split into 64 bins, thus, the feature vector has 64 values.

Color Coherence Vector (CCV) [81]

Like GCH, the CCV is recurrent in the literature. It uses an extraction algorithm that classifies the image pixels as "coherent" or "incoherent" pixels. This classification takes into consideration whether the pixel belongs or not to a region with similar colors, that is, coherent regions. Two color histograms are computed after quantization: one for coherent pixels and another for incoherent ones. Both histograms are merged to compose the feature vector. In our experiments, the color space was quantized into 64 bins.

Color Autocorrelogram (ACC) [45]

The role of this descriptor is to map the spatial information of colors by pixel correlations at different distances. It computes the probability of finding in the image two pixels with color C at distance d from each other. For each distance d, m probabilities are computed, where m represents the number of colors in the quantized space. The implemented version quantized the color space into 64 bins and considered 4 distance values (1, 3, 5, and 7).

Border/Interior Pixel Classification (BIC) [18]

This descriptor has presented good results in image retrieval and classification tasks (e.g., [28], [22], and [24]). The first step of the feature vector extraction process relies on the classification of image pixels into *border* or *interior* ones. When a pixel has the same spectral value in the quantized space as its four neighbors (the ones which are above, below, on the right, and on the left), it is classified as *interior*. Otherwise, the pixel is classified as *border*. Two histograms are computed after the classification: one for the interior pixels and another for the border ones. Both histograms are merged to compose the feature vector. The implemented version quantized the color space into 64 bins. We used the $dlog$ function distance in our experiments, as well as the $L1$ distance.

Cumulative Global Color Histogram (CGCH) [92]

This descriptor is very popular in the literature and is very similar to the GCH descriptor. The main difference in the extraction algorithm is that the value of each bin is cumulated in the next

bin. This makes the last bin have the sum of all the previous bins plus the actual bin. In our experiments, the color space was quantized into 64 bins and the L1 distance function was used.

Local Color Histogram (LCH) [94]

LCH is one of the most popular descriptors that is based on fixed-size regions to describe image properties. Its extraction algorithm splits the image into fixed-size regions and computes a color histogram for each region. After that, the histograms of each region are concatenated to compose one single histogram. The implemented version splitted the image into 16 regions (4x4 grid) and quantized the RGB color space into 64 bins. This generated feature vectors with 1024 values. The L1 distance function was used.

Joint Auto-Correlogram (JAC) [118]

This descriptor follows the same principle used by ACC. However, its extraction algorithm computes the autocorrelogram for more than one image property. The properties considered are: color, gradient magnitude, *rank*, and *texturedness*. Color is extracted in RGB color space and the other properties are extracted from the gray level image. The joint autocorrelogram indicates, for each distance considered, the probability of simultaneously occurring the four properties considered. The implemented version used the HSV color space quantized into 64 bins, 5 bins for the other three properties, a 5×5 pixel neighborhood and 4 distance values (1, 3, 5, and 7). The L1 distance function was used.

Color-Based Clustering (CBC) [17]

CBC is a method for feature extraction based on image segmentation . The method decomposes the image into disjoint connected components. Each region has a minimum size and a maximum color difference. A region is defined by its average color in the CIE Lab color space, by its horizontal and vertical center, and by its size in relation to the image size. The distance function is a combination of L2 distance and *Integrated Region Matching (IRM)* functions.

Color Bitmap [63]

This descriptor analyzes image color properties globally and locally. Its extraction algorithm computes the mean and the standard deviation of each of the R, G, and B channels independently. After that, the image is split into m blocks and the mean of each block is computed for each channel. If the block mean is greater than the image mean, the correspondent feature vector position receives 1; otherwise, it receives 0. The implemented version used 100 blocks. The distance was computed in two steps: L2 function for the mean and standard deviation values; and Hamming distance for the binary values.

Color Structure (CSD) [67]

This is one of the color descriptors used in the MPEG-7 standard. The CSD extraction algorithm uses the HMMD (hue, max, min, diff) color space and scans the image with a 8x8 pixels structuring element. A histogram $h(m)$ is incremented if the color m is inside the structuring element, where m varies from 0 to $M-1$ and M is the color space quantization. The implemented version quantized the space in 184 bins as suggested in [67] and used the L1 distance function.

Color Wavelet HSV (CW-HSV) [106]

This descriptor considers image color properties in the wavelet domain. Its extraction algorithm uses the HSV color space quantized into 64 bins and computes a global color histogram for the image. After that, the Haar wavelet coefficients are hierarchically computed. This is done recursively by dividing the histogram in the middle: if the sum of the values from the first half are greater than the sum of the values from the second half, the correspondent feature vector position receives 1; otherwise, 0. The process is repeated until the last possible level of division, what leads to 63 bits in the feature vector. The distance function is used is the Hamming distance.

Chromaticity Moments (CM) [79]

This descriptor characterizes the image by chromaticity values. Its extraction algorithm first converts the image to the CIE XYZ color space. The chromaticity values (x, y) are computed as $x = \frac{X}{X+Y+Z}$ and $y = \frac{Y}{X+Y+Z}$. After that, two features are computed: the *trace*, that indicates the presence or not of each (x, y) value, and the histogram of chromaticities. The trace and the histogram are used to define the chromaticity moments. In the implemented version, 6 moments were used, leading to 12 values in the feature vector. The distance function cumulates the modular differences between the corresponding moments.

2.3.2 Texture Descriptors

This section presents the low-level texture descriptors we have used in the experiments along this thesis..

Invariant Steerable Pyramid Decomposition (SID) [125]

In this descriptor, a set of filters sensitive to different scales and orientations is used. The image is first decomposed into two sub-bands using a high-pass and a low-pass filter. After that, the low-pass sub-band is decomposed recursively into K sub-bands by band-pass filters and

into one sub-band by a low-pass filter. Directional information about each scale is captured at each recursive iteration. The mean and standard deviation of each sub-band are used as feature values. To obtain the invariance to scale and orientation, circular shifts in the feature vector are applied. The implemented version uses 2 scales and 4 orientations, which leads to a feature vector with 16 values.

Unser [105]

This descriptor is based on co-occurrence matrices, still one of the most widely used descriptors to encode texture in remote sensing applications. Its extraction algorithm computes a histogram of sums H_{sum} and a histogram of differences H_{dif}. The histogram of sums is incremented considering the sum, while the histogram of differences is incremented by taking into account the difference between the values of two neighbor pixels. As well as gray level co-occurrence matrices, measures such as energy, contrast, and entropy can be extracted from the histograms. In our experiments, eight different measures were extracted from histograms and four angles are used (0°, 45°, 90°, and 135°). The final feature vector is composed of 32 values.

Quantized Compound Change Histogram (QCCH) [44]

This descriptor uses the relation between pixels and their neighbors to encode texture information. This descriptor generates a representation invariant to rotation and translation. Its extraction algorithm scans the image with a square window. For each position in the image, the average gray value of the window is computed. Four variation rates are then computed by taking into consideration the average gray values in four directions: horizontal, vertical, diagonal, and anti-diagonal directions. The average of these four variations is calculated for each window position. They are then grouped into 40 bins and a histogram of these values is computed.

Local Binary Pattern (LBP) [76]

LBP is a simple texture descriptor that is invariant to rotation and variations in the gray scale values. Its extraction algorithm defines a window with radio R and a quantity of neighbors P and scans the image counting the quantity of positive and negative variations between the gray values of the neighbor pixels and the central pixel of the window. For gray scale invariance, only the signal of the variation is considered, being 1 for positive and 0 for negative variation. After that, the number of 0/1 and 1/0 transitions are computed, what guarantees the rotation invariance. If the number of transitions is less than 2, the LBP value for that window position is equal to the quantity of 1 signals in the neighborhood. Otherwise, the LBP value is $P+1$. After all the image is scanned, a histogram of LBP values is computed. In our experiments, $R = 1$ and $P = 8$ values. The distance function used was the L1 distance.

Homogeneous Texture Descriptor (HTD) [119]

This descriptor is one of the texture descriptors from the MPEG-7 standard. Its extraction algorithm applies a set of filters sensitive to different scales and orientations. The output of each filter is an image from which the average and standard deviation values are computed. The commonest filters used are Gabor filters. In the implemented version, Gabor filters sensitive to 4 scales and 6 orientations were used, leading to a feature vector with 48 values. The distance function computes the difference between each correspondent average and standard deviation values.

Color Co-Occurrence Matrix (CCOM) [52]

This descriptor is a variation of Gray Level Co-Occurrence Matrix (one of the commonest approaches for texture analysis and classification of RSIs [124, 62, 51]). CCOM extracts the feature vector by first quantizing the color space and then scanning the image to compute the co-occurrence matrix $W(c_p, c_q, d)$. For each pair of image pixels p, q with distance d between themselves, $W(c_p, c_q, d)$ is incremented by one, where c_p is the color of pixel p in the quantized space, c_q is the color of pixel q in the quantized space, and d is the distance between them. The feature vector stores the positive values of the matrix that are below a superior threshold, leading to a variable size feature vector. The implemented version quantizes the RGB color space into 216 bins and uses d equal to 1. The distance function computes the differences between the corresponding W values.

Local Activity Spectrum (LAS) [96]

This descriptor captures texture spatial activity in four different directions separately: horizontal, vertical, diagonal, and anti-diagonal. The four activity measures are computed for a pixel (i, j) by considering the values of neighboring in the four directions. The values obtained are used to compute a histogram that is called *local activity spectrum*. Each component g_i is quantized independently. In our experiments, each component was non-uniformly quantized into 4 bins, leading to a histogram with 256 bins. Distance is computed by L1 function.

2.4 Bag of Visual Words

In this work, we use the notion of global and local descriptor that is normally employed in content-based image retrieval. Global descriptors [14] rely on describing an object (image or region, for example) by using all available pixels. Local descriptors [68], in turn, are extracted from predefined points of interest in the object. Hence, if an object has more than one point of interest in its interior, it can be described by more than one feature vector. A very effective

way to combine local features that describe an object is to group them through the visual-word concept [5, 109].

The representation of object features through visual words involves the construction of a visual dictionary, whose aim is to list all the words present in a given set of objects (an image database or a segmented image, for example).

To create a visual dictionary and, then, an image representation based on visual words, the *Bag of visual Words* (BoW), several steps need to be performed and many variations can be employed in each step. It can be grouped into four main steps: low-level feature extraction; dictionary construction (feature space quantization); coding; and pooling.

We breafly describe each step in the following sections. We also comment, in Section 2.4.5 state-of-the-art researches that use BoW in remote sensing applications.

2.4.1 Low-Level Feature Extraction

Initially, local low-level features are extracted from images. Interest-point detectors or simply a dense grid over the image are used to select images local patches. Literature presents better results for dense sampling in classification tasks [108]. Each local patch is described by an image descriptor, SIFT being the most popular one. Figure 2.3 illustrates a dense sampling strategy to extract features. For each point in the grid, low-level features are extracted considering an area around the point. In Figure 2.3 (a), the features are extracted from a circle area around the interest point. In Figure 2.3 (b), the features are extracted considering a rectangular area with the interest point in the center.

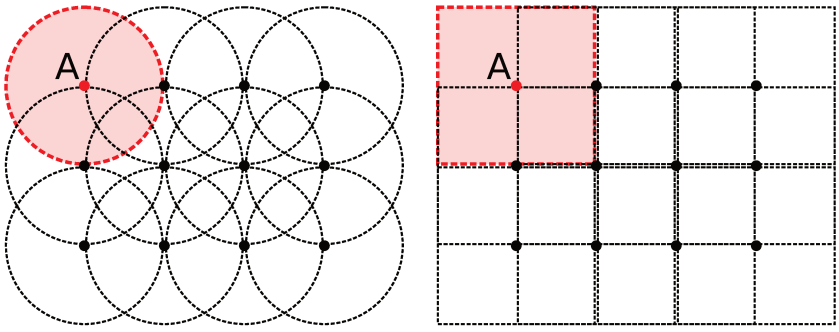


Figure 2.3: Dense sampling using (a) circles and (b) square windows. The highlighted area indicates the region from which the features corresponding to point A are extracted.

2.4.2 Feature Space Quantization

The feature space, obtained from low-level feature extraction, is quantized to create the visual words. Figure 2.4 illustrates the process of building a visual dictionary.

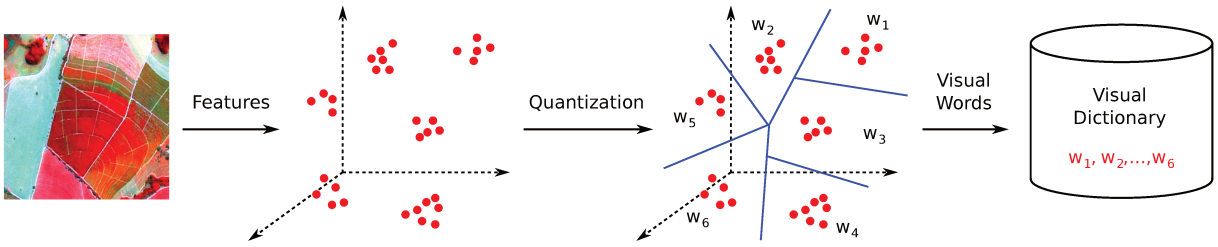


Figure 2.4: Construction of a visual dictionary to describe a remote sensing image. The features are extracted from groups of pixels (e.g., tiles or segmented regions), the feature space is quantized so that each cluster corresponds to a visual word w_i.

A common technique used for feature space quantization is the K-means algorithm [113]. Another way is to perform a simple random selection. We have used the random selection in this work since it is much faster than K-means. Moreover, according to Viitaniemi and Laaksonen [110], in high dimension feature space [47], random selection can generate dictionaries with similar quality to the ones obtained by using K-means.

2.4.3 Coding

Coding is the process of assigning the feature vectors of local patches to one or more visual words in the dictionary. Some coding strategies are: Sparse coding [56], LLC [113], Hard assignment [109], and Soft assignment [109].

Concerning *hard* and *soft* assignments, which are the most tradicional codding strategies, soft assignment are more robust to feature space quantization problems [109]. While *hard* assigns to a local patch the label of the nearest visual word in the feature space, *soft* considers all the visual words near to a local patch, proportionally to their distance. For a dictionary of k words, soft assignment of a local patch p_i can be formally given by Equation 2.2 [109]:

$$\alpha_{i,j} = \frac{K_\sigma(D(p_i, w_j))}{\sum_{l=1}^{k} K_\sigma(D(p_i, w_l))} \tag{2.2}$$

where j varies from 1 to k, $K_\sigma(x) = \frac{1}{\sqrt{2\pi}\times\sigma} \times exp(-\frac{1}{2}\frac{x^2}{\sigma^2})$, and $D(a, b)$ is the distance between vectors a and b. The assignment step results in one k-dimensional vector α_i for each point in the image.

2.4.4 Pooling

The pooling step is the process of summarizing the set of local descriptions into one single feature vector. *Average* and *max* pooling are popular strategies employed, with an advantage to the latter [5].

Average pooling can be formally defined as follows:

$$h_j = \frac{(\sum_{i=1}^{N} \alpha_{i,j})}{N} \tag{2.3}$$

Max pooling is given by the following equation:

$$h_j = \max_{i \in N} \alpha_{i,j} \tag{2.4}$$

In both equations, N is the number of points in the image and j varies from 1 to k.

2.4.5 BoWs and Remote Sensing Applications

The bag-of-visual-words (BoW) model has been used [116, 121, 93], evaluated [9], and adapted for remote sensing applications [46, 34, 122] in several recent works.

Weizman and Goldberger [116] proposed a solution based on visual words to detect urban regions. They apply a pixel-level variant of the visual words concept. The approach is composed of the following steps: build a visual dictionary, learn urban words from labeled images (urban and non-urban), and detect urban regions in a new image. Xu et al. [121] proposed a similar classification strategy based on bag of words. The main difference is that their approach builds the visual vocabulary in patch-level by using interest-points detectors and local descriptors. In [93], Sun et al. used visual dictionaries for target detection in high-resolution images. Another approach focused on high resolution images is described in [46]. Huaxin et al. [46] proposed a local descriptor which encodes color, texture, and shape properties. The extracted features are used to build a visual dictionary by using k-means clustering.

Chen et al. [9] evaluated 13 different local descriptors for high resolution image classification. In their experiments, the SIFT descriptor obtained the best results.

Feng et al. [34] proposed a BoW-based approach to synthetic aperture radar (SAR) image classification. The proposed method starts by extracting Gabor and GLCM features from segmented regions. The dictionary is built by using the clonal selection algorithm (CSA), which is a searching method. Yang et al. [122] also proposed an approach based on bag of words for synthetic aperture radar (SAR) image classification. Their approach relies on a hierarchical Markov model on quadtrees. For each tile in each level of the quadtree, a vector of local visual descriptors is extracted and quantized by using a level-specific dictionary.

Chapter 3

Experimental Protocol

This chapter describes the experimental protocol used to validate the methods proposed in this work. Section 3.1 describes the datasets used. Section 3.2 presents the measures used to evaluate the classification results obtained in the performed experiments.

3.1 Remote Sensing Image Datasets

We have used three different remote sensing image datasets to perform experiments in this work. We refer in this text to each dataset according to the target or region of interest: COFFEE, PASTURE, and URBAN areas. Table 3.1 presents a brief overview about each image. The datasets are described in details in the following sections.

3.1.1 COFFEE Dataset

This dataset is a composition of scenes taken by the SPOT sensor in 2005 over Monte Santo de Minas county, in the State of Minas Gerais, Brazil. This area is a traditional place of coffee cultivation, characterized by its mountainous terrain. In addition to common issues in the area

Table 3.1: Remote sensing images used in the experiments.

	PASTURE	**COFFEE**	**URBAN**
Terrain	plain	mountainous	plain
Satellite	CBERS	SPOT	QuickBird
Spatial res.	$20m$	$2.5m$	$0.6m$
Bands comp.	R-IR-G	IR-R-G	R-G-B
Acquisition date	08–20–2005	08–29–2005	2003
Location	Laranja Azeda Basin, MS	Monte Santo County, MG	Campinas,SP
Dimensions (px)	1310×1842	14017×13488	9079×9486

of pattern recognition in remote sensing images, these factors add further problems that must be taken into account. In mountainous areas, the spectral patterns tend to be affected by the topographical differences and interference generated by the shadows. This dataset provides an ideal environment for multi-scale analysis, since the variations in topography require the cultivation of coffee in different crop sizes. Another problem is that coffee is not an annual crop. This means that, in the same area, there may be plantations of different ages, as illustrated in Figure 3.1. In terms of classification, we have several completely different patterns representing the same class, while some of these patterns are very similar to those of other classes.

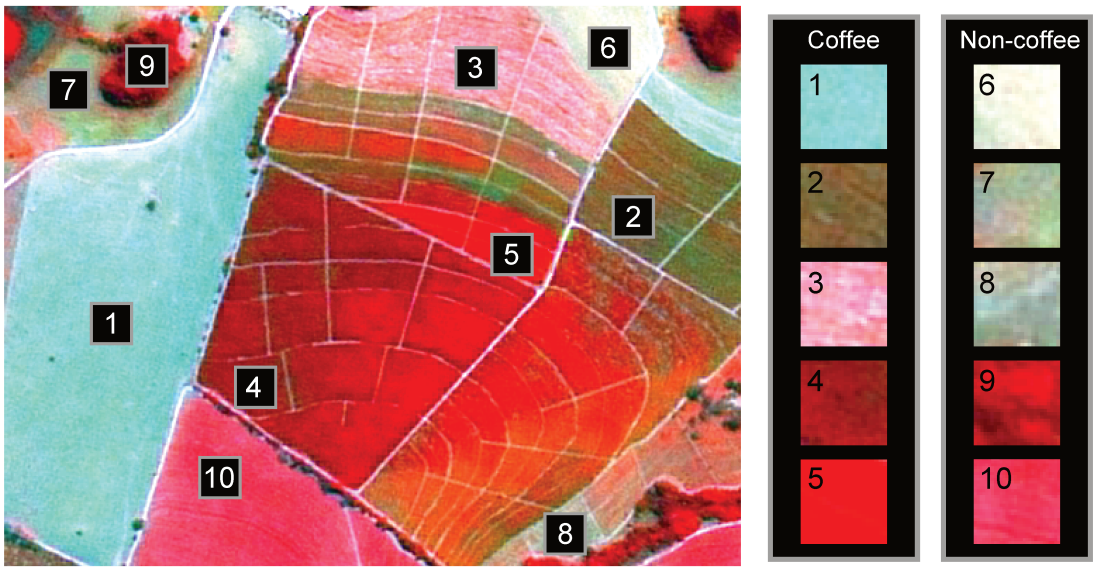
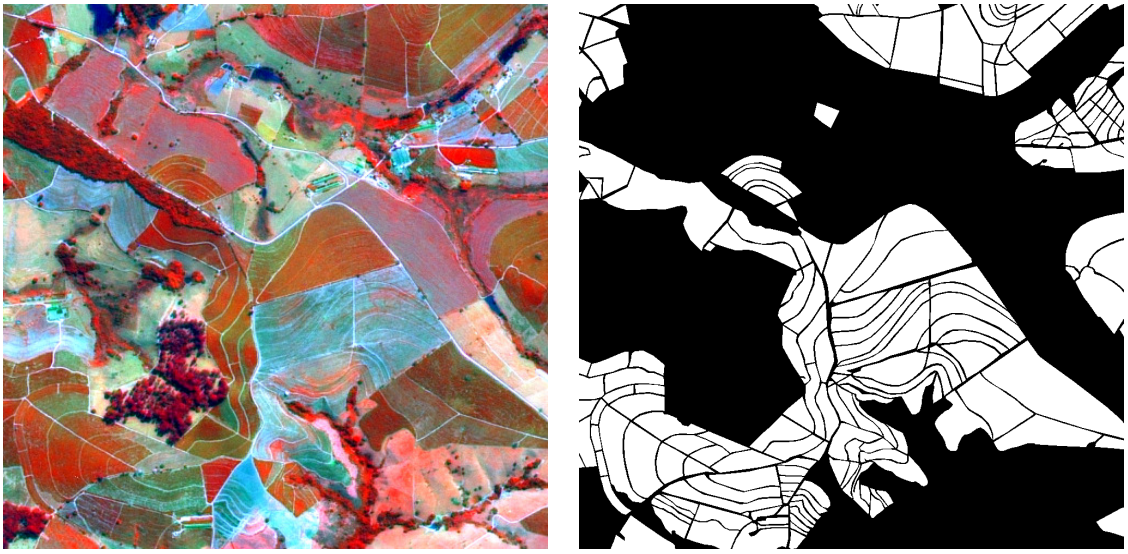


Figure 3.1: Example of coffee and non-coffee samples in the used RSI. Note the difference among the samples of coffee and their similarities with non-coffee samples [22].

We have used a complete mapping of the coffee areas in the dataset for training and assessing the quality of experimental results. The identification of coffee crops was done manually in the whole county by agricultural researchers. They used the original image as reference and visited the place to compose the final result. Figure 3.2 (a) illustrates a subimage of the COFFEE dataset. Figure 3.2 (b) illustrates all the coffee crops from Figure 3.2 (a).

We considered five different scales to extract features from λ_1 (the finest one) to λ_5 (the coarsest one). We selected the scales according to the principle of dichotomic cuts (see Section 2.2). Figure 3.3 illustrates the multi-scale segmentation for one of the subimages. At λ_5 scale, subimages contain between 200 and 400 regions while, at scale λ_1, they contain between $9,000$ and $12,000$ regions.



(a) (b)

Figure 3.2: COFFEE data with (a) a subimage from the original RSI and (b) the ground truth that indicates the regions corresponding to coffee crops.

3.1.2 PASTURE Dataset

The PASTURE image (Figure 3.5(a)) is a cutout of an RSI captured by CBERS satellite that corresponds to the Laranja Azeda Basin in the State of Mato Grosso do Sul, Brazil. This image is from a plain region, without major distortions in the terrain. Because of that, there are no many interferences in the spectral pattern and the classification is considered easy.

The PASTURE ground truth (Figure 3.5(b)) was created by agricultural specialists by using the Spring software [11]. First, the PASTURE image was segmented by applying a region growing algorithm [39]. After the segmentation, each object was classified by using the Bhattacharya algorithm with 90% certainty. The PASTURE ground truth image was revised by the agricultural researches after visiting the region.

3.1.3 URBAN Dataset

This dataset is a Quickbird scene taken in 2003 from Campinas region, Brazil. It is composed by three bands that correspond to the visible spectrum (red, green, and blue). We have empirically created the ground truth based on our knowledge about the region. We considered as urban the places which correspond to residential, commercial, or industrial regions. Highways, roads, native vegetation, crops, and rural buildings are considered non-urban areas. Figure 3.2 (a) illustrates the URBAN image. Figure 3.2 (b) indicates the urban areas in the URBAN image.

Figure 3.6 illustrates the multi-scale segmentation by using the Guigues' algorithm (Sec-

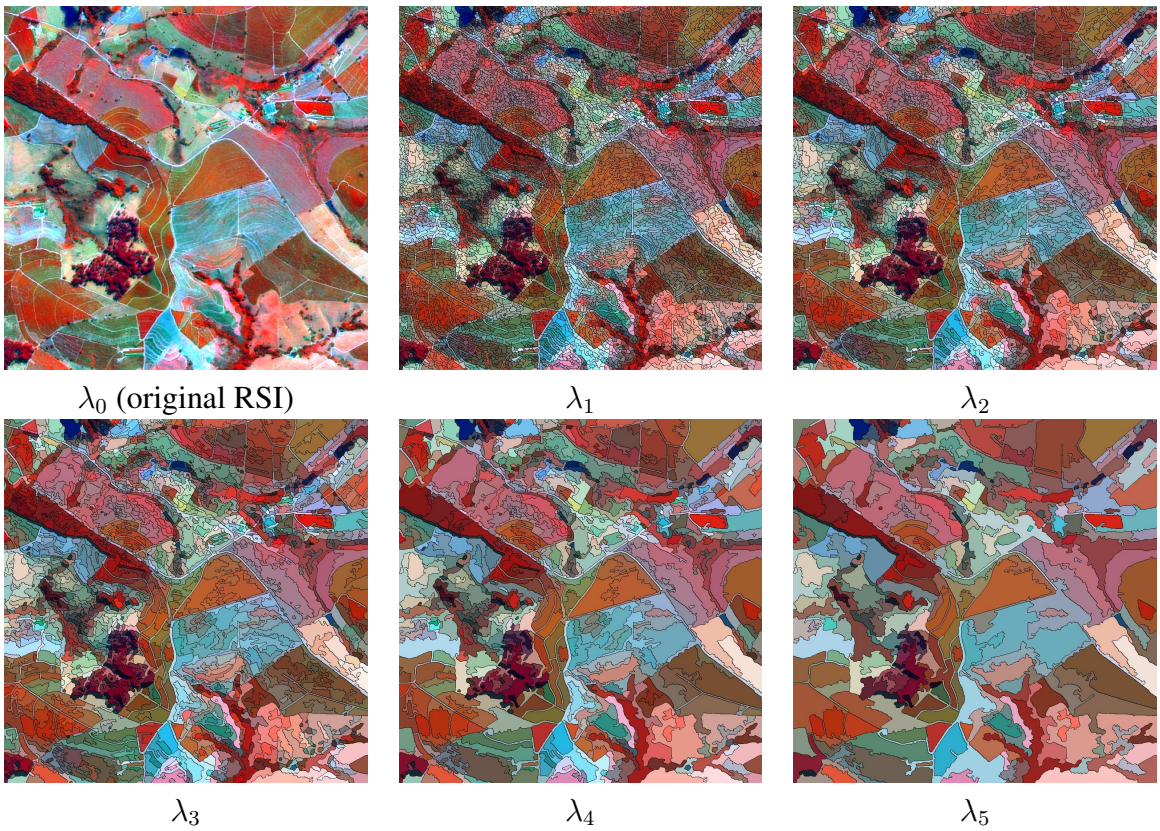


Figure 3.3: One of the tested subimages and the results of segmentation in each of the selected scales for the COFFEE dataset.

tion 2.2) for one of the subimages used in the experiments from URBAN dataset.

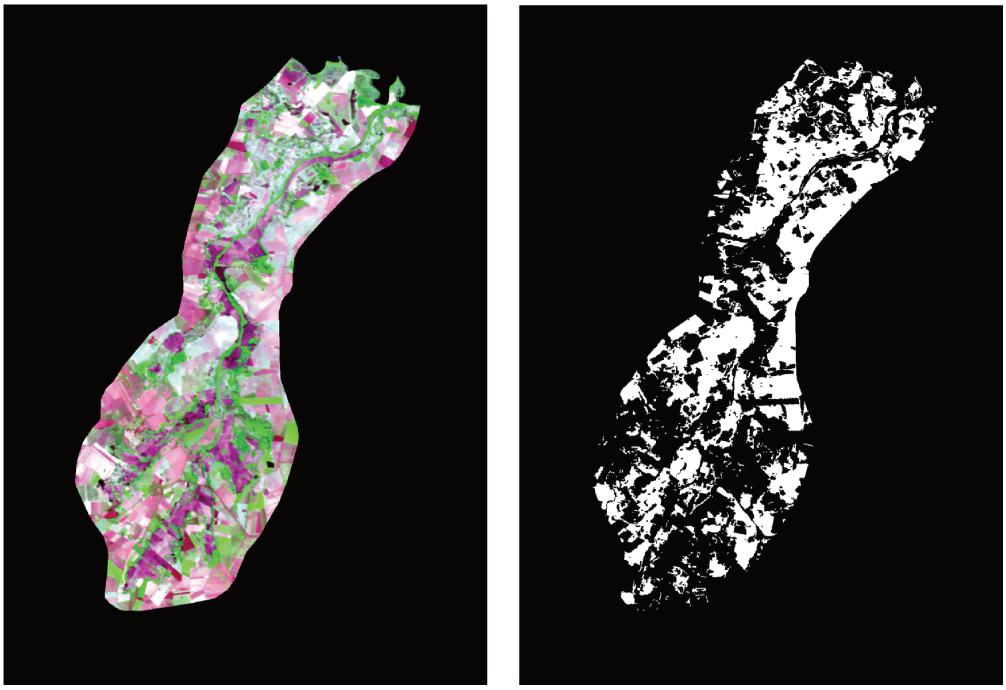
3.2 Measures

In our experiments, we have used evaluation measures in terms of values stored in confusion matrices [61]. Table 3.2 presents a confusion matrix for m classes constructed with both reference and the classified data for all pixels in the studied RSI.

The three evaluation measures used along this thesis are: overall accuracy, kappa index (κ), and tau index (τ). A comparison of measures can be found in [36]. In our experiments, we assess the results quality for region-based classification at pixel level.

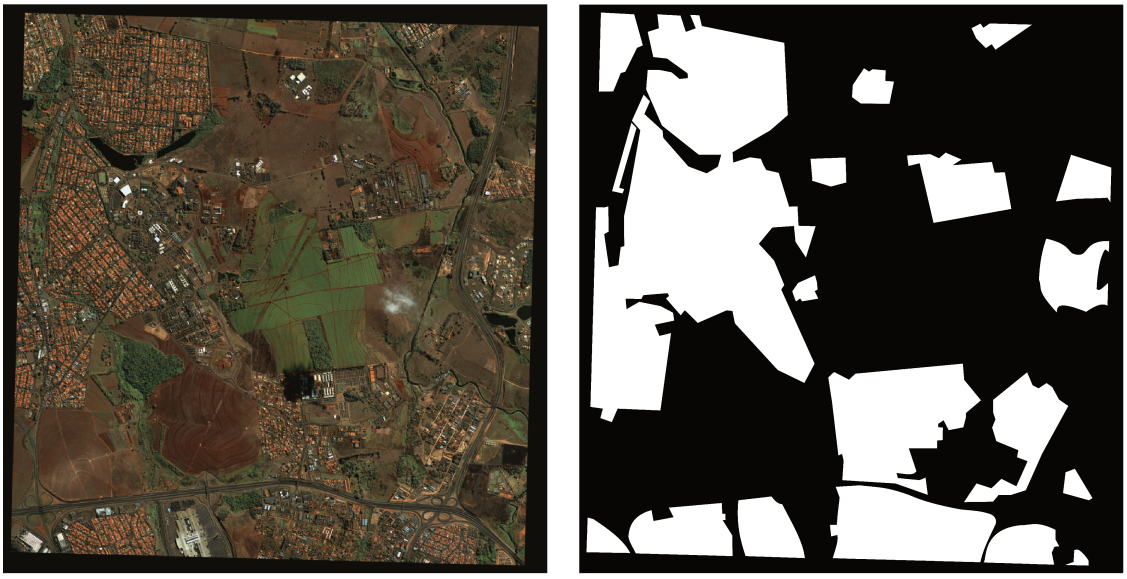
Overall accuracy [13] is the most popular accuracy measure. It is computed as follow:

$$OA = \frac{\sum_{i=1}^{m} x_{ii}}{N} \times 100 \tag{3.1}$$



(a) (b)

Figure 3.4: PASTURE data with (a) original RSI and (b) the ground truth that indicates the regions that correspond to pasture.



(a) (b)

Figure 3.5: URBAN data with (a) original RSI and (b) ground truth that indicates the regions that correspond to urban areas.

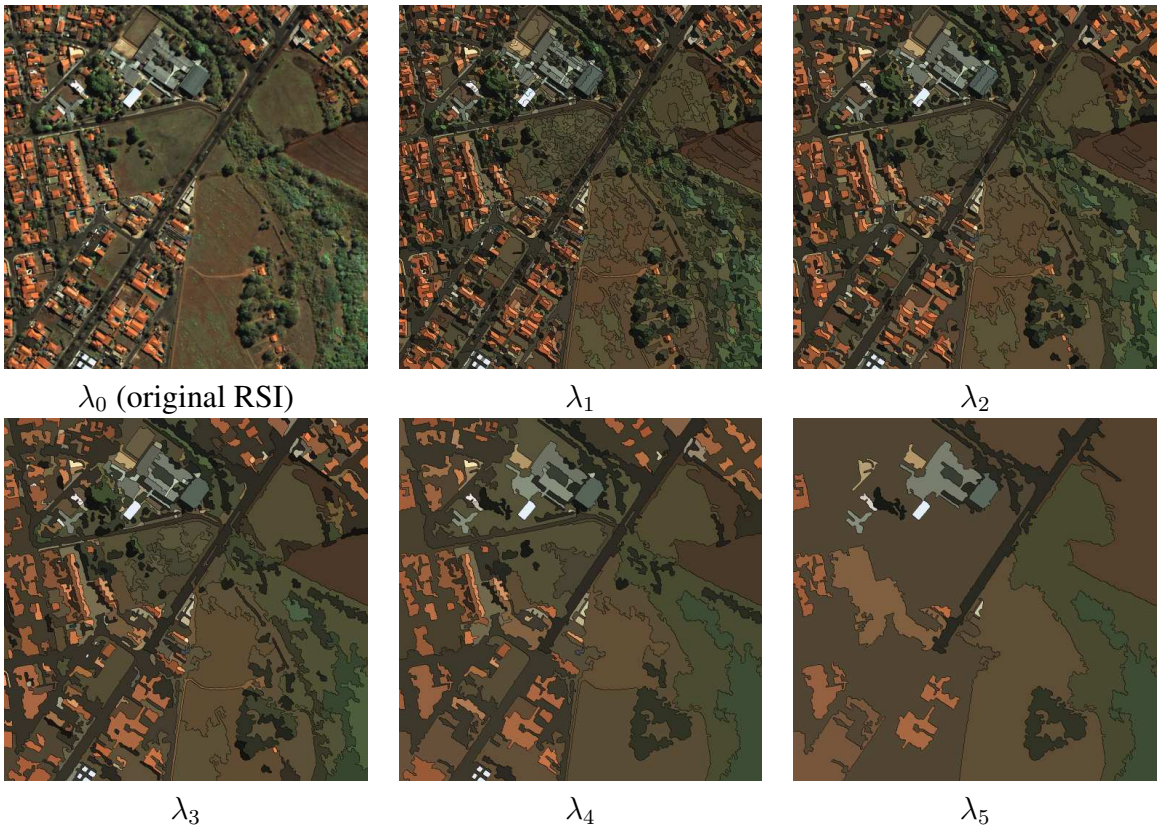


Figure 3.6: One of the tested subimages and the segmentation results in each of the selected scales for the URBAN dataset.

Table 3.2: Confusion matrix with x_{ij} representing the number of pixels in the classified (observed) image category i and the ground truth (reference) cover category j. Adapted from [61].

		Reference				
		1	2	...	m	Total
Observed	1	x_{11}	x_{12}	...	x_{1m}	x_{1+}
	2	x_{21}	x_{22}	...	x_{2m}	x_{2+}
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
	m	x_{m1}	x_{m2}	...	x_{mm}	x_{m+}
	Total	x_{+1}	x_{+2}	...	x_{+m}	

where m is the number of rows in the confusion matrix, x_{ii} is the number of pixels observations in main diagonal (row i and column i).

The Kappa index κ [12, 13] is a measure of aggreement between the reference data and the classifier result. It is computed by:

$$\kappa = \frac{N \sum_{i=1}^{m} x_{ii} - \sum_{i=1}^{m} (x_{i+} \times x_{+i})}{N^2 - \sum_{i=1}^{m} (x_{i+} \times x_{+i})} \tag{3.2}$$

where r is the number of rows in the confusion matrix, x_{ii} is the number of observations in row i and column i; x_{i+} and x_{+i} are the marginal totals of row i and column i, respectively; and N is the total number of observations.

In general, negative Kappa means that there is no agreement between classified data and reference data. Kappa value equals to 1.0 means "perfect agreement". Experiments in different areas show that Kappa could have various interpretations and these guidelines could be different depending on the application. Table 3.3 illustrates a possible interpretation, suggested in [55]:

Table 3.3: Possible interpretations for kappa values.

Kappa index	**Interpretation**
$\kappa = 1$	Perfect agreement
$0.8 < \kappa < 1.0$	Almost perfect agreement
$0.6 < \kappa \leq 0.8$	Substantial agreement
$0.4 < \kappa \leq 0.6$	Moderate agreement
$0.0 < \kappa \leq 0.4$	Poor agreement
$\kappa \leq 0$	No agreement

The Tau index [49, 65] indicates the percentage of extra pixels correctly classified when compared to the expected by using a random classifier. Like Kappa, the better the classification performance, the higher the Tau index. It is given by:

$$\tau = \frac{P_0 - P_r}{1 - P_r} \tag{3.3}$$

where

$$P_0 = \frac{1}{N} \sum_{i=1}^{m} x_{ii} \quad P_r = \frac{1}{N^2} \sum_{i=1}^{m} (x_{i+} \times x_{ii}) \tag{3.4}$$

Chapter 4

Evaluation of Descriptors for RSI Classification

This chapter presents an evaluation of image descriptors for RSI retrieval and classification. Seven descriptors that encode texture information (see Section 2.3.2) and twelve color descriptors (see Section 2.3.1) that can be used to encode spectral information were selected. We perform experiments to evaluate the effectiveness of these descriptors in retrieval sessions and classification tasks. The evaluation methodology is presented in Section 4.1. The experimental results are presented in Section 4.2.

4.1 Descriptor Evaluation Methodology

We performed experiments to evaluate and compare the descriptors considering their effectiveness performance. For this purpose, we designed two experiments: one for retrieval effectiveness evaluation and another for overall accuracy classification.

Two image databases were created to evaluate image descriptors based on the PASTURE and COFFEE datasets. One of them can be classified as "easy recognition" (PASTURE image) while the other as "hard recognition" (COFFEE image). Section 3.1 provides more detais regarding these images.

In the experiments, one image is represented by a tile of the original RSI. The size of the tile was fixed according to the common extension value of a *region of interest*. COFFEE crops are normally in small parcels on the same farm. We defined that $75 \times 75\ meters$ is a good value to the size of the tile. For PASTURE parcels, that are larger, the chosen value was $400 \times 400\ meters$. The dimension of partitions are fixed in the experiments. We used $30 \times 30\ pixels$ to tile the COFFEE image and $20 \times 20\ pixels$ for the PASTURE image. The number of partitions for the PASTURE and COFFEE images was 5980 and 6400, respectively.

To evaluate retrieval effectiveness, Precision × Recall curves were used. *Precision* quantifies the percentage of relevant images present in the retrieved results. *Recall* is a measure that represents the percentage of the relevant images that are retrieved. A Precision × Recall curve indicates the variation in Precision values as the rate of relevant images from the database (Recall) changes. Intuitively, the higher the curve, the better the effectiveness.

The Precision × Recall curves were computed based on the average values obtained for each query image in each database. We used 340 and 100 queries from the PASTURE and COFFEE image sets, respectively for all the color and texture descriptors presented in Sections 2.3.1 and 2.3.2, respectively. We have used the EVA tool to perform these experiments [82].

To compute the overall accuracy of each descriptor, we implemented a variation of the K-Nearest Neighboor (KNN) classifier. First of all, a set of tiles from the database was randomly selected to be used as training set. The set, corresponding to 10% of the database size, is composed of relevant and non-relevant samples in the same proportion in the full database. To classify one tile, each descriptor was used to compute the distance between the given tile and all the training set tiles. Based on the descriptor distances, the training set is ranked and the first K tiles are weighted inversely proportional to their position in the rank. Finally, the sum of the weights for each class (relevant or non-relevant) is computed. The largest sum indicates the class of the input tile. To test the classification effectiveness of the descriptors, 100 tiles were used for each dataset.

4.2 Experimental Results

Figures 4.1, 4.2, 4.3, and 4.4 show the Precision × Recall curves for color and texture descriptors in the databases used. From Figure 4.1, we can see that good descriptors considering retrieval effectiveness are Color Bitmap, and ACC. From Figure 4.2, it is possible to see that JAC presents the highest Precision values even for small values of Recall and for Recall equal to 1. Analyzing Figure 4.3, we notice that SID has the highest Precision values for all values of Recall among texture descriptors. Considering curves for the COFFEE database in Figure 4.4, it is possible to see that the descriptors present similar Precision values and these values are around 32% to 40% when Recall reaches 10%. In general, SID presents a small advantage.

After analyzing the curves for color and texture descriptors, we can say that color descriptors are slightly better than texture descriptors for the databases used. For example, in the PASTURE database, for Recall equal to 10%, the highest Precision value for color descriptors is around 62% (JAC) and for texture descriptors is near 47%. For Recall equal to 1, color descriptors achieve Precision of 25% (Color Bitmap) and texture descriptors achieve almost 23%. Concerning the COFFEE dataset, for Recall equal to 10%, the highest curve of a color descriptor reaches 61% (JAC) while the highest curve of a texture descriptor reaches almost 40% (SID). For Recall equal to 1, there is almost no difference in the Precision values.

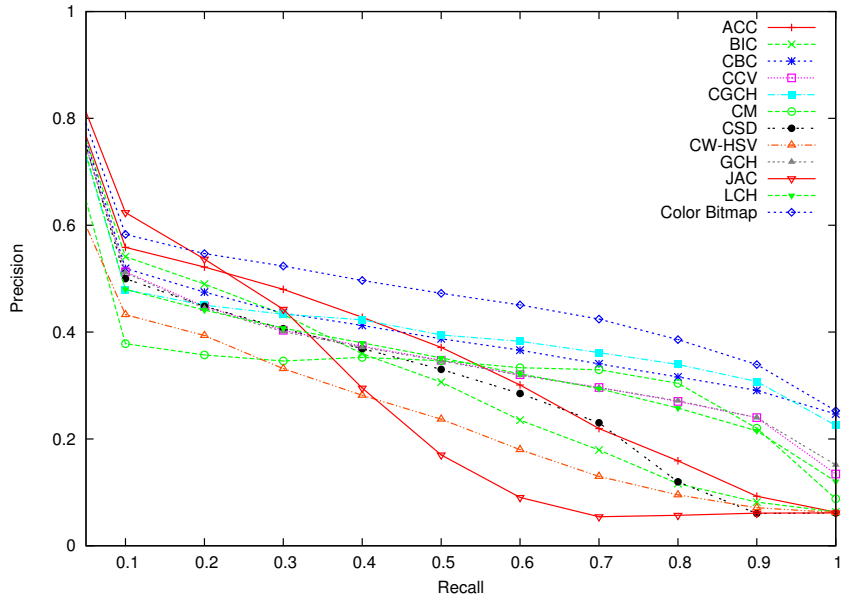


Figure 4.1: Precision $\times$ Recall curves for color descriptors, considering the PASTURE dataset.

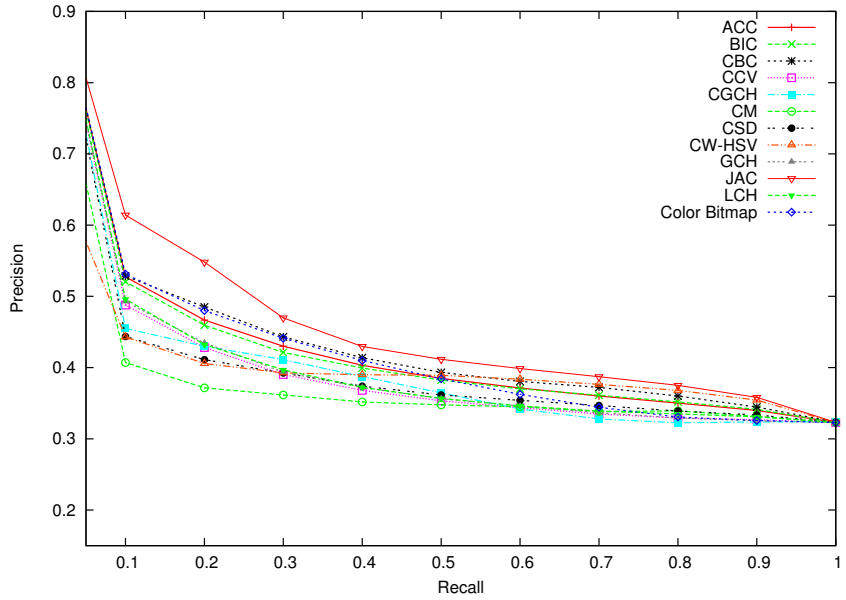


Figure 4.2: Precision $\times$ Recall curves for color descriptors, considering the PASTURE dataset.

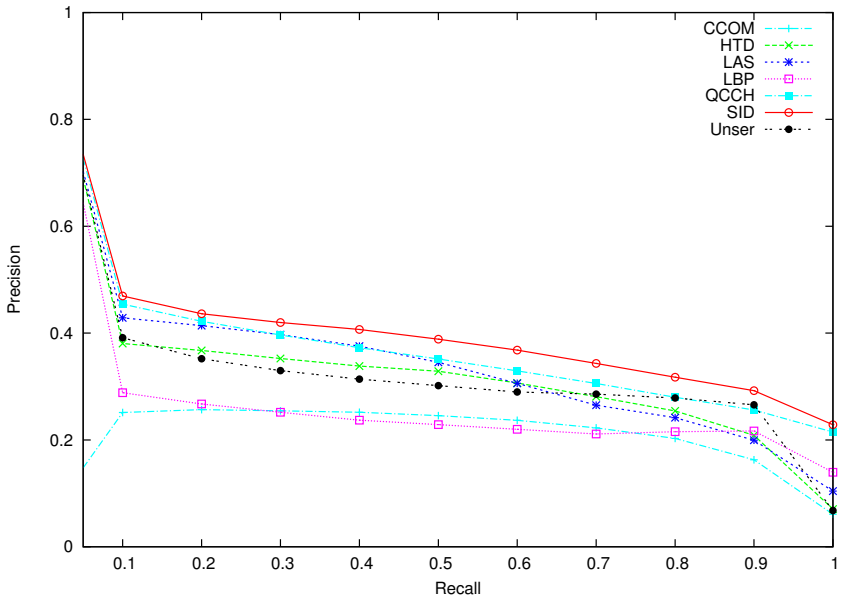


Figure 4.3: Precision $\times$ Recall curves for texture descriptors, considering the PASTURE dataset.

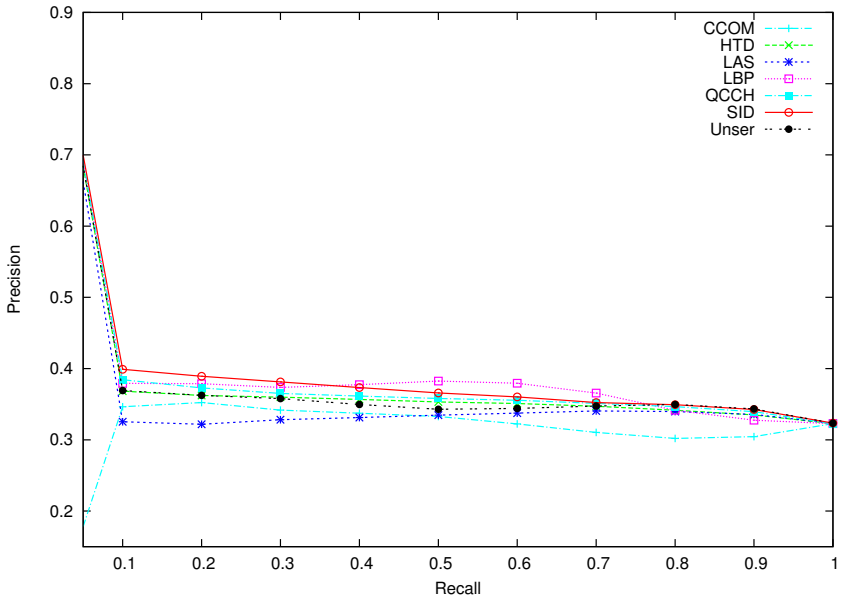


Figure 4.4: Precision $\times$ Recall curves for texture descriptors, considering the PASTURE dataset.

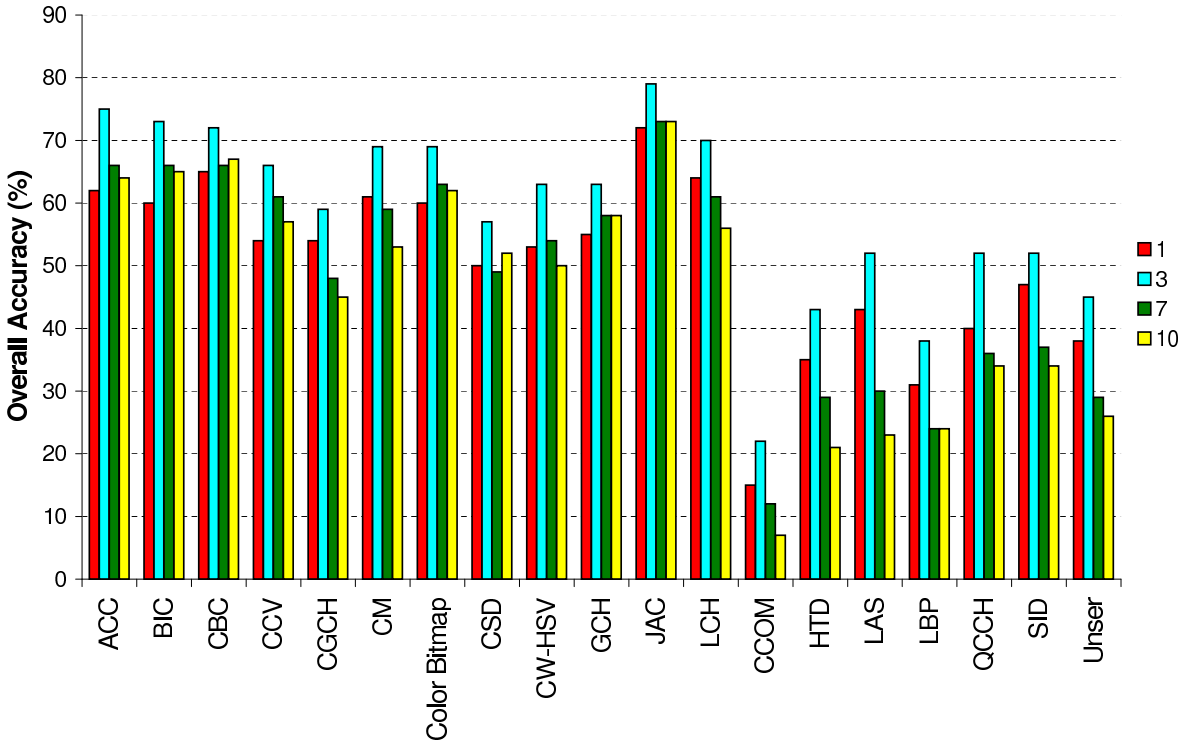


Figure 4.5: Overall accuracy classification of each descriptor for the COFFEE dataset, using KNN with k equal to 1, 3, 7 and 10.

According to the results for the COFFEE database presented in Figure 4.5, one can observe that some descriptors achieve high overall accuracy values. The color descriptors BIC, ACC, CBC, Color Bitmap, and JAC are the best ones reaching more than 60% of overall accuracy for any k. JAC produced the highest accuracy values, being the only one with values over 70% (72% for $k = 1$, 79% for $k = 3$, and 73% for $k = 7$ and $k = 10$). With regard to the texture descriptors, QCCH, SID, and LAS yield the highest accuracy values, 52% for k=3. For k values different than 3, the texture descriptors presented accuracy below 48%. The CCOM descriptor does not reach 25% of accuracy in any of the experiments in the COFFEE dataset.

According to the results for the PASTURE database (Figure 4.6), we can see that some descriptors yield good accuracy values. The color descriptors JAC, Color Bitmap, and CBC reach near or more than 60% of overall accuracy. The JAC descriptor is again the descriptor with the highest accuracy value, reaching 78% for k=3 and being over 65% for all k values. The texture descriptors yield lower accuracy values when compared with most of color descriptors. QCCH, SID, and Unser are the only texture descriptors to reach accuracy above 50%. For k=3, QCCH reaches 58% of accuracy; SID, 55%; and Unser, 53%. The CCOM descriptor yields the lowest accuracy values, being below 25% for all k values.

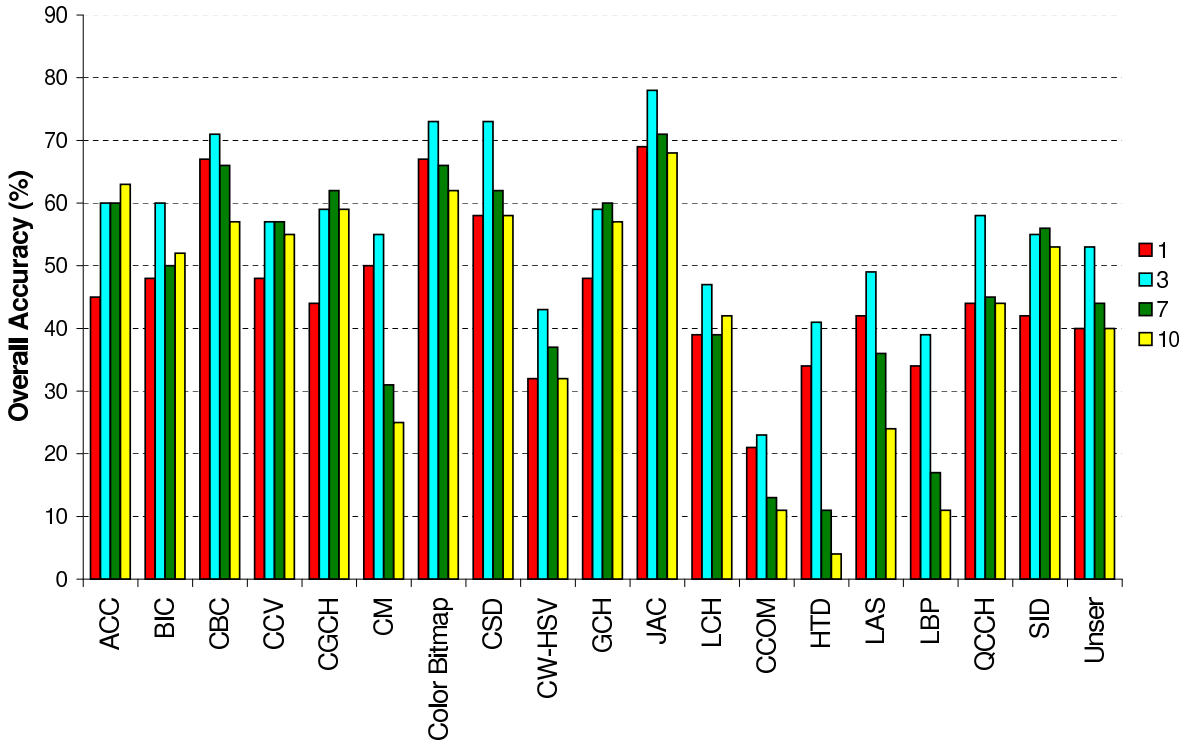


Figure 4.6: Overall accuracy classification of each descriptor for the PASTURE dataset, using KNN with k equal to 1, 3, 7 and 10.

4.3 Conclusions

We can point JAC as the best color descriptor. However, JAC generates large feature vectors and therefore, it is slower to be used in practical applications. If storage and time requirements are not critical, JAC is the best choice. Other descriptors with high effectiveness are CBC and Color Bitmap. CBC has complex extraction and distance function. Color Bitmap can be a good choice among the color descriptors, as it balance simple algorithms and relative good effectiveness. Among the texture descriptors, QCCH and SID yield the highest accuracy values, being SID computationally more complex than QCCH for feature extraction.

We take aforementioned analysis into account to select the descriptors employed in the other experiments described in this thesis. However, we also consider some other aspects like extraction time, size of the produced feature vector, and implementation simplicity. These aspects, which are essential for the multiscale approaches proposed in the next chapters, are extensively analysed and reported for all the tested descriptors in [83].

Hence, we have selected the following color descriptors: ACC, BIC, CCV, and GCH. Although JAC (Joint Auto-Correlogram) presents the best results in our experiments, we have

replaced it by ACC (Color Auto-Correlogram) because the above mentioned drawbacks of using JAC. BIC was selected because it presents reasonable accuracy for the COFEEE dataset, which is high resolution and the most used one in this thesis. BIC has presented good results in many other applications [83, 14] as well. Moreover, BIC is easy to implement. GCH and CCV are well-known descriptors and also easy to implement. Their extraction time and feature vector size are positive aspects for multiscale tasks.

Concerning texture, we have selected QCCH, SID, and Unser descriptors. SID and QCCH achieve the best results in the COFFEE dataset. The Unser descriptor exploits the coocurrence matrix indexes, which are widely used features in remote sensing applications.

Chapter 5

Multiscale Training and Classification based on Boosting of Weak Classifiers

5.1 Introduction

Regardless of the data representation model adopted in supervised classification of RSIs, both the training input and the result of the classifier can be expressed as sets of pixels. In spite of that, data representation cannot only rely on pixels, because their image characteristics are not usually enough to capture the patterns of the classes (regions of interest). In order to bridge that semantic gap, multiscale image segmentation can play an important role. As pointed out by Trias-Sanz et al. [98], most of the image segmentation methods use threshold parameters to create a partition of the image. These methods usually create a single-scale representation of the image: small thresholds give segmentation with small regions and many details, while large thresholds preserve only the most salient regions. The problem is that various structures can appear at different scales and this segmentation result can be difficult to obtain without prior knowledge about the data or by using only empirical parameters. It is difficult to define the optimal scale for segmentation. Some parts of an image may need a fine segmentation, since the plots are small, whereas, in other parts, a coarse segmentation is sufficient. For this reason, the main drawback of classification methods based on regions is that they depend on the segmentation method used. Bearing this in mind, many researchers have exploited multiple scales of data [77, 114, 51, 102, 104, 107].

Allied to the problem of finding the best scale of segmentation, there is the problem of selection/combination of extracted features. In addition to this, several studies show that the combination of features improve classification results [22, 24].

We propose a kind of boost-classifier adapted to multiscale segmentation, taking advantage of various region features computed at various levels of segmentation. To build multiscale classifiers, we propose two approaches for multiscale analysis of images: the Multiscale Classifier

(MSC) and the Hierarchical Multiscale Classifier (HMSC). The MSC is based on the Adaboost algorithm [87], which builds a strong classifier from a set of weak ones. The HMSC is also based on boosting weak classifiers, but it relies on a sequential strategy of training, according to the segmentation hierarchy of scales (from the coarsest to the finest). In the proposed work, we employ two types of weak learners: Support Vector Machine (SVM) and Radial Basis Function (RBF). The RBF approach is based on the distances provided by the used descriptors. We have also analyzed the correlation between the used descriptors at different scales.

Instead of choosing any particular scale, which is usually not enough to represent all regions of interest, we segment the image using Guigues algorithm (see Section 2.2). The choice of the most relevant regions and of the most discriminative features between relevant and non-relevant samples is done by the machine learning. Our method differs from the others in four main aspects. First, it does not rely on particular scale and, thus, it can capture the information from different parts and scales of the image. Then, it exploits the results of auxiliary scales to improve classification. Furthermore, it combines classification results from different scales rather than fusing features. Last, it assigns the same set of classes for all scales, producing a single final result, instead of producing a distinct classification result per scale.

The use of the proposed method only depends on the used descriptors. Thus, the proposed method can be used to classify any image/region, given that the descriptors are suitable for the target image/region. It is important to clarify that the method will better work for images with some noise and higher resolutions, in which representative features can be extracted from both small and large regions.

This chapter is divided into four sections. Section 5.2 introduces the proposed approaches for multiscale training and classification. Experimental results concerning the proposed aproaches are presented in Section 5.3. In Section 5.4, we present a correlation analysis among the descriptors and each scale of segmentation. Finally, in Section 5.5, we present our conclusions.

5.2 Multiscale Training and Classification

In the next sections, we describe the basic ideas of our approach, as well as the major processing steps for multiscale classification. In Section 5.2.1, we introduce the concepts and the general functioning of the proposed approach. In Sections 5.2.2 and 5.2.3, the two approaches that we propose for training classifiers using several scales are presented. Finally, in Section 5.2.4, we describe the weak classifiers used in the proposed method.

5.2.1 Classification Principles

The aim of RSI classification is to build a classification function $F(p)$ that returns a classification score ($+1$ for relevant, and -1 otherwise) for each pixel p of a RSI. Let us note that, even

if the classification returns a result at a pixel level, the decision may be based on regions of different scales containing the pixel.

In order to create such classification function $F(p)$, we first extract different features at different scales using multiscale segmentation. After this step, we use boosting to build a linear combination of weak classifiers, each of them related to a specific scale and feature type. The training is performed using a RSI image I where each pixel is labeled. Figure 5.1 illustrates the steps of the multiscale training approach.

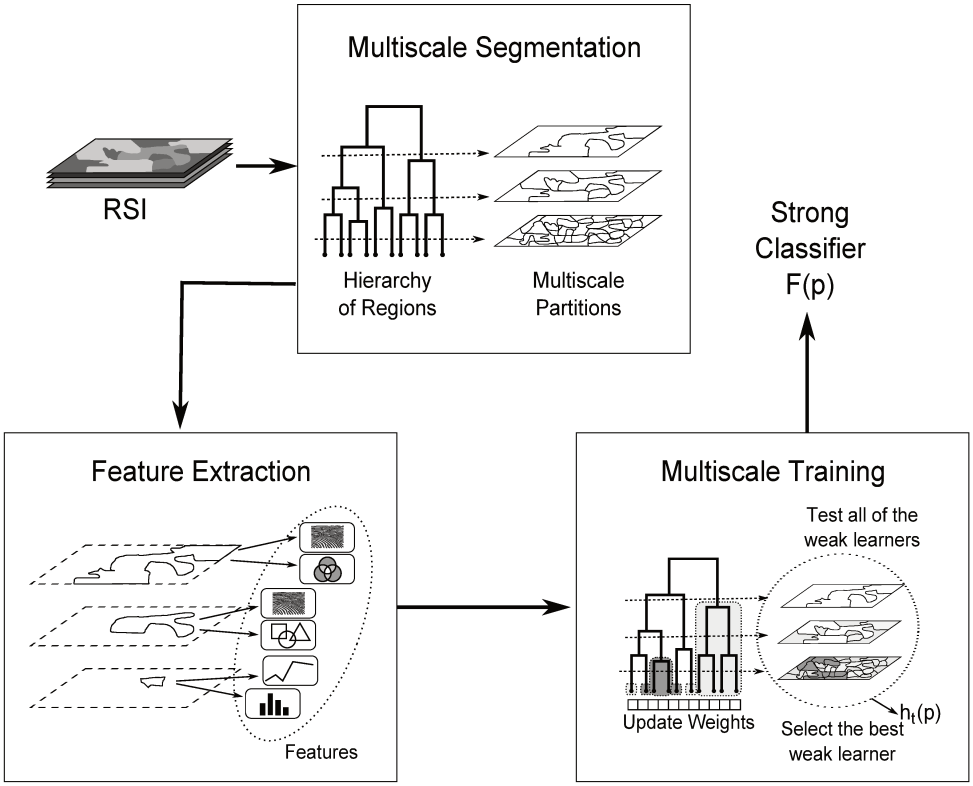


Figure 5.1: Steps of the multiscale training approach. At the begining, several partitions P_λ of hierarchy H at various scales λ are selected. Then, at each scale λ, a set of features is computed for each region $R \in P_\lambda$. Finally, a classifier $F(p)$ is built by using the Multiscale Training (Section 5.2.2) or the Hierarchical Multiscale Training (Section 5.2.3).

As explained in Section 2.2, the base of hierarchy H is composed of the set of pixels from the training image I, and it will be denoted P_0. We will use several partitions P_λ of hierarchy H at various scales λ. At each scale λ, a set of features is computed for each region of P_λ. These features can be different according to the level, and, thus, to the size of the regions. For example, a texture feature is not appropriate for too small regions and a histogram (such as color

histogram) is less accurate for large or very small regions.

5.2.2 Multiscale Training

The *Multiscale classifier* (MSC) aims at assigning a label ($+1$ for relevant class, and -1 otherwise) to each pixel p of P_0 taking advantage of various features computed on regions of various levels of the hierarchy. To build multiscale classifiers, we propose a learning strategy based on boosting of weak learners. This strategy is based on AdaBoost algorithm proposed by Schapire [87], which builds a linear combination $MSC(p)$ of T weak classifiers $h_t(p)$:

$$MSC(p) = \text{sign}\Big(\sum_{t=1}^{T} \alpha_t h_t(p)\Big) \tag{5.1}$$

The proposed algorithm repeatedly calls *weak learners* in a series of rounds[1] $t = 1, \dots T$. Each weak learner creates a weak classifier that decreases the expected classification error of the combination. The algorithm then selects the weak classifier that most decreases the error.

The strategy consists in keeping a set of weights over the training set. These weights can be interpreted as a measure of the difficulty level to classify each training sample. At the beginning, all pixels have the same weight, but in each round, the weights of the misclassified pixels are increased. Thus, in the next rounds the weak learners are forced to focus on hardest samples. We will note $W_t(p)$ the weight of pixel p in round t, and $D_{t,\lambda}(R)$ the misclassification rate of region R in round t at scale λ given by the mean of the weights of its pixels:

$$D_{t,\lambda}(R) = \Big(\frac{1}{|R|}\sum_{p\in R} W_t(p)\Big) \tag{5.2}$$

Algorithm 1 presents the proposed Multiscale Training process. Let $Y_\lambda(R)$, the set of labels of regions R at scale λ, be the input dataset. We divide this set into training ($Y_\lambda^t(R)$) and validation sets ($Y_\lambda^v(R)$). In a serie of rounds $t = 1, \dots T$, for all scales λ, the weight of each region $D_{t,\lambda}(R)$ is computed (line 3). This piece of information is employed to select the regions to be used for training the weak learners, building a subset of labeled regions $\hat{Y}_{t,\lambda}$ (line 6). The subset $\hat{Y}_{t,\lambda}$ is used to train the weak learners with each feature $\mathcal{F}$ at scale λ (line 9). Each weak learner produces a weak classifier $h_{t,(\mathcal{F},\lambda)}$ (line 10). The algorithm then selects the weak classifier h_t that most decreases the error Err_{h_t} on the validation set Y_λ^v (line 12). The level of error of h_t is used to compute the coefficient α_t, which indicates the degree of importance of h_t in the final classifier (line 13). The selected weak classifier h_t and the coefficient α_t are used to update weights $W_{(t+1)}(p)$ which can be used in the next round (line 14).

The classification error of classifier h is:

[1] Despite the term "iterations" be more common, we use the term "rounds" that is typically applied to refer to the main loop present in boosting-based methods.

Algorithm 1 Multiscale Training

Input:

$Y_\lambda(R)$ = labels of regions R at scale λ ($Y_\lambda = Y_\lambda^t \cup Y_\lambda^t$, where Y_λ^t is the training set and Y_λ^v is the validation set)

Initialize:

For all pixels p, $W_1(p) \leftarrow \frac{1}{|Y_0|}$, where $|Y_0|$ is the number of pixels in the image level

For $t \leftarrow 1$ to T do
  For all scales λ do
    For all $R \in P_\lambda$ do
      Compute $D_{t,\lambda}(R)$
    End for
    Build $\hat{Y}_{t,\lambda} \subset Y_\lambda^t$ (a training subset based on $D_{t,\lambda}(R)$)
  End for
  For each pair feature/scale $(\mathcal{F}, \lambda)$ do
    Train *weak learners* using features $(\mathcal{F}, \lambda)$ and training set $\hat{Y}_{t,\lambda}$.
    Evaluate resulting classifier $h_{t,(\mathcal{F},\lambda)}$ on the validation set Y_λ^v by computing $Err(h_{t,(\mathcal{F},\lambda)}, W_{t,\lambda}))$ (Equation 6.3)
  End for
  Select *weak classifier* h_t, the one with minimum error $Err^* = \text{argmin}_{h_{t,(\mathcal{F},\lambda)}} Err(h_{t,(\mathcal{F},\lambda)}, W_{t,\lambda})$
  Compute $\alpha_t \leftarrow \frac{1}{2} ln\left(\frac{1+r_t}{1-r_t}\right)$ with $r_t \leftarrow \sum_p cY_0(p)h_t(p)$
  Update $W_{t+1}(p) \leftarrow \dfrac{W_t(p)\exp\left(-\alpha_t Y_0(p)h_t(p)\right)}{\sum_p W_t(p)\exp\left(-\alpha_t Y_0(p)h_t(p)\right)}$
End for

Output: Multiscale Classifier $MSC(p)$ (Equation 5.1)

$$Err(h, W) = \sum_{p|h(p)Y_0^v(p)<0} W(p) \tag{5.3}$$

where Y_0^v is the validation set (the label of each pixel in the image).

The training is performed on the training set labels Y_λ^t, which is the learning at a single scale λ. The weak learners (linear SVM, for example) use the subset $\hat{Y}_{t,\lambda}$ for training and produce a weak classifier $h_{t,(\mathcal{F},\lambda)}$. The training/validation set labels Y_0 are the labels of pixels of image I, and training/validation sets labels Y_λ with $\lambda > 0$ are defined according to the proportions of pixels belonging to one of the two classes (for example, at least 80% of one region).

The idea of building the subset $\hat{Y}$ is to force the classifiers to train with the most difficult samples. The weak learner should allow the most difficult samples to be differentiated from the other ones according to their weights. Thus, the strategy of creating $\hat{Y}$ is directly dependent on the configuration of the weak classifier and may contain all regions, since the classifier considers the weights of the samples.

5.2.3 Hierarchical Training

The Multiscale Training presented in Section 5.2.2 creates a classifier based on the linear combination of weak classifiers. In this case, both the selection of scales and features, and the weights of each weak classifier are obtained by a strategy based on AdaBoost. Although this approach provides the selection of the most appropriate scales to the training set, it does not ensure the representation of all scales in the final result. In addition, the cost of training with each scale is proportional to the number of regions it contains. However, the coarse scales are not always selected, which means that training time can be reduced if we avoid this analysis.

In order to overcome these problems, we propose a *hierarchical multiscale classification* scheme. The proposed strategy is presented in Figure 5.2. It consists of individually selecting the weak classifiers for each scale, starting from the coarsest one to the finest one. Thereby, each scale provides a different stage of training. At the end of each stage, only the most difficult samples are selected, limiting the training set used in the next stage. In each stage, the process is similar to the one described in Algorithm 1. However, the weak learners are trained with only the features related to the current scale. For each scale, the weak learner produces a set H_λ of weak classifiers.

The *hierarchical multiscale classifier* ($HMSC$) is a combination of the set of weak classifiers $\mathcal{S}_\lambda(p)$ selected for each scale λ:

$$HMSC(p) = \text{sign}\Big(\sum_{\lambda_i} \mathcal{S}_{\lambda_i}(p)\Big) = \text{sign}\Big(\sum_{\lambda_i}\sum_{t=1}^{T} \alpha_{t,\lambda_i} h_{t,\lambda_i}(p)\Big) \tag{5.4}$$

where T is the number of rounds for each boosting step.

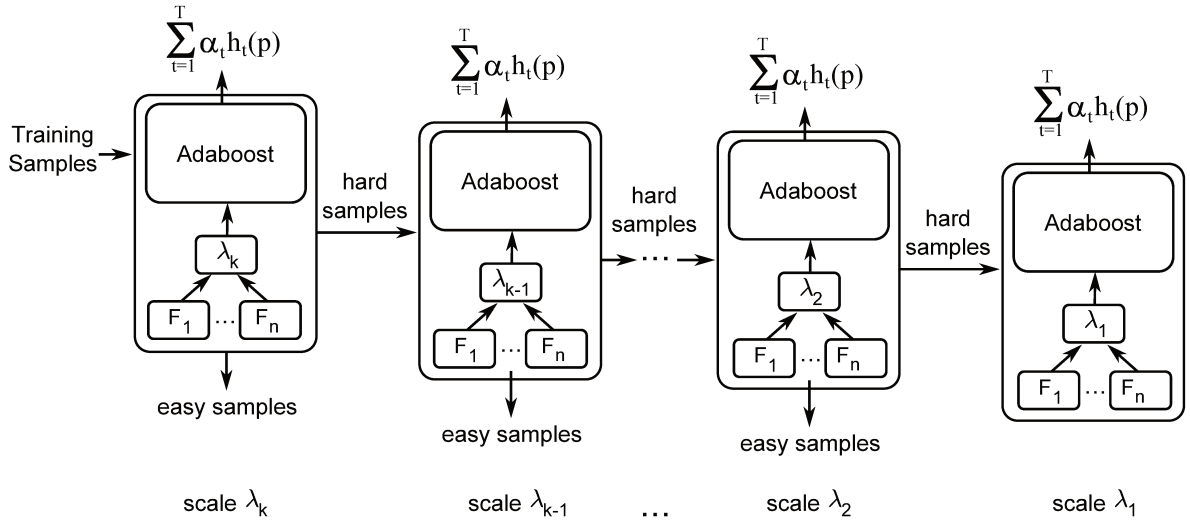


Figure 5.2: The hierarchical multiscale training strategy.

At the end of each stage, we withdraw the easiest samples. Let W_i be the weights of the pixels after training with scale λ_i, we denote $D_i(R_{i+1})$ the weight of region $R_{i+1} \in P_{\lambda_{i+1}}$, which is given by:

$$D_i(R_{i+1}) = \left(\frac{1}{|R|}\sum_{p \in R} W_i(p)\right) \tag{5.5}$$

The set of regions $\breve{Y}_{i+1}$ used in the training stage with scale λ_{i+1} is composed by the regions $R_{i+1} \in P_{\lambda_{i+1}}$ with mean $D_i(R_{i+1}) > \frac{1}{2|Y_0|}$. This means that the regions that ended a training stage with distribution equal to half the initialization value $\frac{1}{|Y_0|}$, are discarded for the next stage.

It is important to note that despite the strategy described above is based on a hierarchical procedure resembling a cascade of classifiers, the purpose is completely different. While a cascade of rejections, as used by Viola and Jones [111], aims to create efficient classifiers (fast detectors), the main focus of the HMSC is the reduction of the training time.

5.2.4 Weak Classifiers

We adopted two types of weak learners: Support Vector Machines (SVM) and Radial Basis Function (RBF).

SVM-based weak learner

This SVM trainer is based on a specific feature type $\mathcal{F}$ and a specific scale λ. Given the training subset labels $\hat{Y}_\lambda$, the strategy is to find the best linear hyperplane of separation between RSI regions according to their classes (relevant and non-relevant regions), trying to maximize the data separation margin. These samples are called support vectors and are found during the training. Once the support vectors and the decision coeffients ($\alpha_i, i = 1, \ldots, N$) are found, the SVM weak classifier can be defined as:

$$SVM_{(\mathcal{F},\lambda)}(R) = \text{sign}\Big(\sum_{i}^{N} y_i \alpha_i (f_R \cdot f_i) + b\Big) \tag{5.6}$$

where b is a parameter found during the training. The support vectors are the f_i such that $\alpha_i > 0$, y_i is the support vector class and f_R is the feature vector of the region.

The training subset $\hat{Y}_{t,\lambda}$ is composed of n labels from Y_λ with values of $D_{t,\lambda}(R)$ larger or equal to $\frac{1}{|Y_0|}$. This strategy means that only regions considered as the most difficult ones are used for the training. For the first round of the boosting, the regions which compose the subset $\hat{Y}_{0,\lambda}$ are randomly selected.

The weakness of the linear SVM classifier is due to our strategy of creating subsets instead of providing all regions of a partition for training. It decreases the power of the produced classifier. Moreover, in our experiments the dimension of the feature space is smaller than the number of samples, which theoretically guarantees the weakness of linear classifiers.

RBF-based weak learner

The RBF approach is based on the distances provided by the used descriptors. It consists in selecting a target region that best separates the other regions between both classes for a specific image descriptor $\hat{D}$ and a specific scale λ. The distances are normalized with the sigmoid function.

The RBF-based weak learner tests all training regions (i.e, $\hat{Y}_\lambda = Y_\lambda$) as targets in the classification task. The exception are the regions that have already been used as targets.

$$RBF_{(R_t,\hat{D},\lambda)}(R) = \begin{cases} y, & \text{if } d(R_t, R) \leq l \\ -y, & \text{otherwise} \end{cases} \tag{5.7}$$

where $d(R_t, R)$ is the distance between target region R_t and region R using descriptor $\hat{D}$ and l is a threshold value.

5.3 Multiscale Classification Experiments

In this section, we present the experiments that we performed to validate our method. We have carried out experiments in order to address the following research questions:

- Is the set of used descriptors effective for object-based RSI classification task?
- Is the multiscale classification results effective in RSI tasks?
- Are the proposed weak learners effective in the RSI classification problem?
- Can the hierarchical strategy for multiscale classification improve the results?
- Are the proposed methods effective in the RSI classification problem when compared with a baseline?

In Section 5.3.1, we describe the basic configuration of our experiments. In Section 5.3.2, we compare the used descriptors through the proposed MSC exploiting a single-scale segmentation. In Section 5.3.3, we compare the combination of multiple scales approach against individual scales combining descriptors through the MSC approach. In Section 5.3.4, we compare the proposed weak classifiers Linear SVM and RBF. In Section 5.3.5, we present the results for the HMSC approach and the comparison with MSC. Finally, in Section 5.3.6, we compare the proposed approaches against a baseline based on the SVM classifier.

5.3.1 Setup

We extracted different features from the COFFEE dataset (see Section 3.1.2) by using four color and three texture descriptors. The color descriptors are: Global Color Histogram (GCH), Color Coherence Vector (CCV), Color Autocorrelogram (ACC), and Border/Interior Pixel Classification (BIC). The texture descriptors are: Invariant Steerable Pyramid Decomposition (SID), Unser, and Quantized Compound Change Histogram (QCCH). These descriptors were preselected based on previous results, as reported in Section 4.3.

To facilitate the experimental protocol, we divided the dataset into a grid of 3×3, generating 9 subimages with dimensions equal to 1000×1000 pixels. In the experiments, we used 9 different sets of 1 million pixels each, to be used for training and classification (testing stage). The results of the experiments described in the following sections are obtained from all combinations of the 9 subimages used (3 for training, 3 for validation, and 3 for classification).

To analyze the results, we computed the overall accuracy and Kappa index for the classified images (for more details, see Section 3.2).

The experiments were carried out on a 2.40GHz Quad Core Xeon with 32 GB RAM.

5.3.2 Comparison of Descriptors

The result of classification is directly related to the quality of the features extracted from the image. In this sense, the objective of this experiment is to compare descriptors in region-based classification tasks. To do so, we used the MSC approach with linear Support-Vector Machines in an intermediate scale of segmentation (λ_2). Table 5.1 presents the overall accuracy and Kappa results for each descriptor.

Table 5.1: Classification results for the used descriptors at λ_2 scale.

	Descriptor	**Overall Acc. (%)**	**Kappa (κ)**
Color	ACC	78.60 $\pm$1.88	0.7238 $\pm$0.029
	BIC	**79.92 $\pm$2.04**	**0.7447 $\pm$0.033**
	CCV	77.38 $\pm$2.72	0.7011 $\pm$0.046
	GCH	77.64 $\pm$2.71	0.7056 $\pm$0.045
Texture	$QCCH$	**69.94 $\pm$4.21**	**0.5503 $\pm$0.086**
	$UNSER$	68.72 $\pm$3.67	0.5255 $\pm$0.078
	SID	68.63 $\pm$3.76	0.5215 $\pm$0.078

BIC yields the best results among all the descriptors. BIC takes into account the spatial distribution of colors, which in a way encodes both color and texture. QCCH achieves a small highlight among the texture ones. The results present a small difference between GCH and CCV. In fact, we observed that their classification results are correlated.

The great difference between the color and texture descriptors classification rates was expected. This fact is consistent with those results obtained in [28] and [98]. Anyway, we believe that the combination of texture and color descriptors can improve the results.

5.3.3 Multiscale versus Individual Scale

In this section, we compare the classification results obtained by using individual scales against the combination of scales by using the MSC approach presented in Section 5.2.2 with 10 rounds. In this experiments, we used all descriptors referenced in Section 4.3. Table 5.2 presents the classification results. Table 5.3 presents the time spent for training and classification.

According to the results, one can observe that the combination of scales ($\bigcup_{i=1}^{5} \lambda_i$) is slightly better than the best individual scale (λ_4). We can conclude that the proposed method MSC not only found the best scale but also could improve the result by adding other less significant scales.

Concerning time, the combination is longer to train when compared to scale λ_2, but not longer than scale λ_1 alone. The same effect can be observed for the classification time.

Table 5.2: Classification results using individual scales and the combination.

Scale	Overall Acc. (%)	Kappa (κ)
λ_1	79.07 $\pm$1.60	0.7298 $\pm$0.028
λ_2	79.90 $\pm$2.04	0.7441 $\pm$0.033
λ_3	80.43 $\pm$2.11	0.7519 $\pm$0.033
λ_4	81.04 $\pm$1.70	0.7625 $\pm$0.026
λ_5	80.31 $\pm$1.23	0.7494 $\pm$0.020
$\bigcup_{i=1}^{5} \lambda_i$	**82.28 $\pm$1.60**	**0.7800 $\pm$0.025**

Table 5.3: Time spent on classification using individual scales and the combination.

Scale	Training Time (s)	Classification Time (s)
λ_1	44454.54	103.98
λ_2	9163.32	36.99
λ_3	1272.69	14.59
λ_4	349.27	8.56
λ_5	84.85	6.25
$\bigcup_{i=1}^{5} \lambda_i$	24939.34	38.52

5.3.4 Comparison of Weak Classifiers (Linear SVM $\times$ RBF)

In this section, we compare the weak learners presented in Section 5.2.4. We performed experiments with 10 rounds for SVM-based and 50 rounds for RBF-based weak learners. This is the amount of rounds which normally stabilizes the results using each of the weak learners. In other words, after 10 rounds for SVM and 50 rounds for RBF, the selected weak learner typically gets very small weights and does not interfere in the final classification. Table 5.4 presents the classification results. Table 5.5 presents training/classification times.

Table 5.4: Classification results comparing the MSC approach using RBF and SVM-based weak learners.

Weak Learners	Overall Acc. (%)	Kappa (κ)
RBF	77.78 $\pm$3.68	0.6957 $\pm$0.082
$Linear\ SVM$	**82.28 $\pm$1.60**	**0.7800 $\pm$0.025**

We can observe that MSC with SVM-based weak learners produces better results than with RBF-based. Moreover, the RBF-based weak learner spends more time in both training and classification stages. However, it is necessary to point out that, in these experiments, the distances between regions using the descriptors are computed during the classification stage. If distances

Table 5.5: Time spent on classification using the MSC approach with RBF and SVM-based weak learners.

Weak Learners	Training Time (s)	Classification Time (s)
RBF	31030.987	327.01
$Linear\ SVM$	24939.34	38.52

are previously computed, RBF-based weak learners are an alternative since they can be easily implemented.

5.3.5 Hierarchical Multiscale Classification

In this section, we present the results of the proposed Hierarchical Multiscale Classification approach. Table 5.6 presents the overall accuracy and Kappa index for HMSC and MSC approach. Time is presented in Table 5.7. We used 10 rounds for MSC and 50 rounds for HMSC (10 rounds for each scale). To maintain the detection time of the classifier HMSC equivalent to the MSC, the weak learners with very low weights are excluded from the final classifier: the threshold on the weights is 0.01. This reduces the final classifier to a combination between 10 and 15 weak learners.

Table 5.6: Classification results comparing the HMSC against MSC.

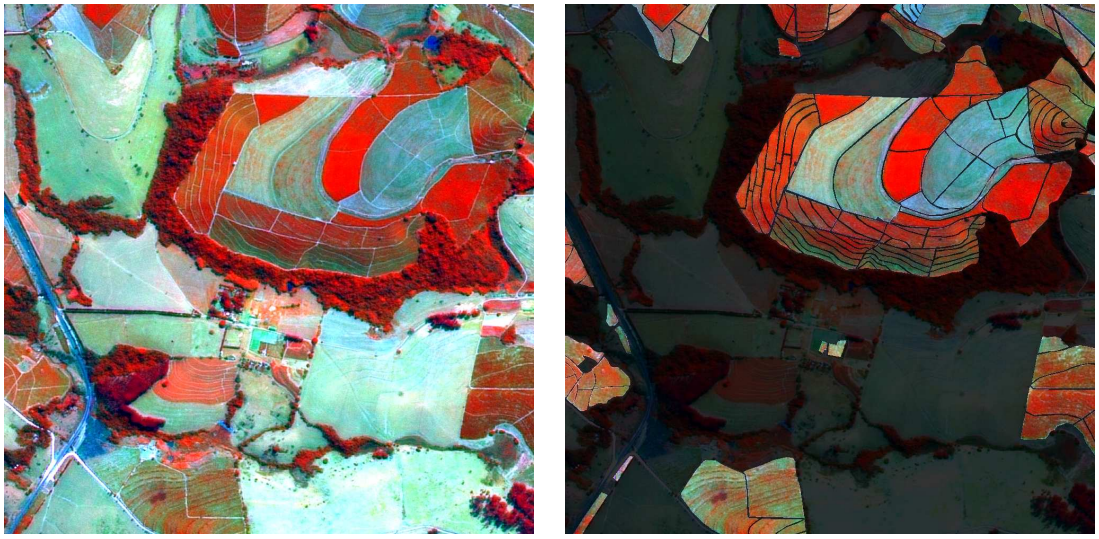
Method	Overall Acc. (%)	Kappa (κ)
$HMSC$	**82.69 ±1.68**	**0.7875 ±0.024**
MSC	82.28 ±1.60	0.7800 ±0.025

Table 5.7: Time spent on classification for MSC and HMSC.

Method	Training Time (s)	Classification Time (s)
$HMSC$	13637.62	39.06
MSC	24939.34	38.52

Both methods produce similar values of accuracy. The most important point concerns the training time. As the hierarchical approach does not use all regions of all scales, training time is considerably reduced (almost half time) because the training focuses only on the most difficult regions.

Figure 5.3 (a) shows a subimage used in these experiments and Figure 5.3 (b) illustrates the same image with coffee crops, which are the regions of interest in focus. Figures 5.4 (a) and (b) illustrate an example of results obtained with both methods HMSC and MSC.



(a) (b)

Figure 5.3: The image used for classification in Figure 5.4 (a) and the same image with coffee crops highlighted (b).

Although producing almost the same accuracy rates, the main difference in these examples is that HMSC produces less false positives than MSC (HMSC produces also more false negatives). We assume that the HMSC is more efficient to recognize coffee crops.

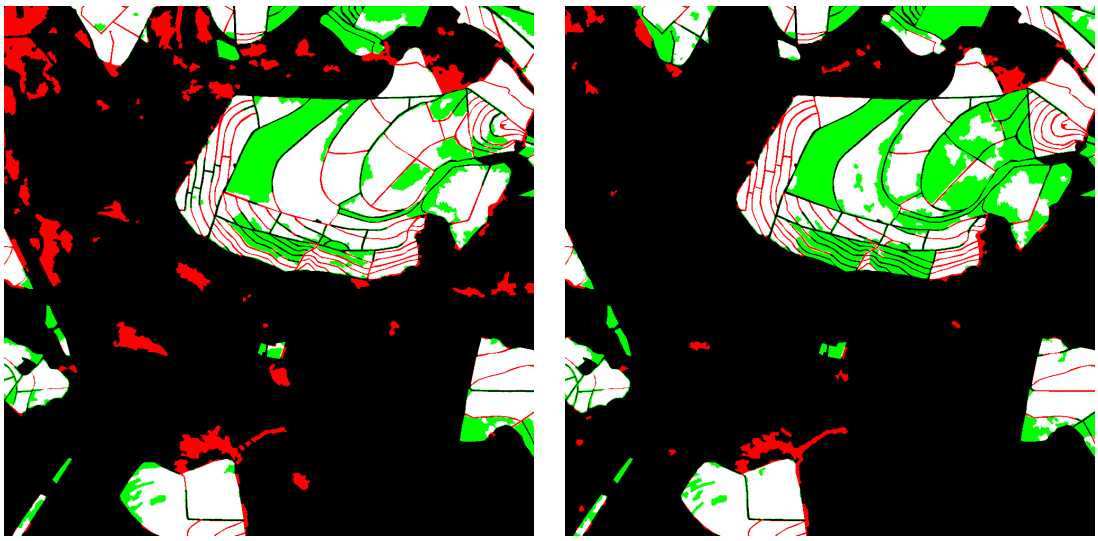
Table 5.8: Accuracy analysis of classification for the example presented in Figure 5.4 (TP = true positive, TN = true negative, FP = false positive, FN = false negative).

Method	**TP**	**TN**	**TP+TN**	**FP**	**FN**	**FP+FN**
MSC	194,378	670,493	864,871	64,228	70,901	135.129
$HMSC$	167,293	705,196	**872,489**	29,525	97,986	**127.511**

We observed that most of the classification errors are related to the confusion caused by recently planted coffee crops. These regions usually appear in light blue in the composition of colors displayed (see Figure 5.3).

5.3.6 Comparison with a baseline

Although they are very used in image classification [69], SVMs are so far less used in remote sensing community than other classifiers (e.g., decision trees and variants of neural networks). However, in recent years there has been a significant increase in SVM-based works that achieves very good results in remote sensing problems. Tzotsos et al. [102] have proposed and evaluated SVMs for object-oriented classification. They proposed an approach that uses SVMs with a



(a) (b)

Figure 5.4: A result obtained with the proposed methods: MSC (a) and HMSC (b). Pixels correctly classified are shown in white (true positive) and black (true negative) while the errors are displayed in red (false positive) and green (false negative).

Gaussian kernel to classify the regions obtained by a multiscale segmentation process. This approach outperforms the results of the eCognition software [3]. Therefore, we used SVM with Gaussian kernel applied to an intermediate segmentation scale obtained by the Guigues' method as baseline with BIC descriptor. As the baseline was not designed to use the validation set, we performed these experiments with two settings: 3 subimages for training and 3 for classification; 6 subimages for training and 3 for classification. Table 5.9 displays the results.

Table 5.9: Classification results comparing the MSC, HMSC and the baselines. $SVM + Gaussian\ Kernel$ (3,3) is the baseline trained with 3 subimages. $SVM + Gaussian\ Kernel$ (6,3) is the same baseline trained with 6 subimages.

Method	**Overall Acc. (%)**	**Kappa (κ)**
$SVM + Gaussian\ Kernel$ (3,3)	77.47 $\pm$2.64	0.7054 $\pm$0.044
$SVM + Gaussian\ Kernel$ (6,3)	80.09 $\pm$1.58	0.7478 $\pm$0.025
$MSC\ (linear\ SVM\ learner)$	82.28 $\pm$1.60	0.7800 $\pm$0.025
$HMSC\ (linear\ SVM\ learner)$	**82.69 $\pm$1.68**	**0.7875 $\pm$0.024**

As it can be noticed, both MSC and HMSC overcome the results of the baseline. This shows that the combination of descriptors and scales using the strategies proposed in this work can be a powerful tool for classification of remote sensing images.

5.4 Multiscale Correlation Analysis

In Section 5.3, we show that the combination of features at different scales improves the classification results, but these results still lack more explanation about how to select the best scales and descriptors. In this context, the objective of this section is to address such questions.

We have carried out experiments by using support vector machines (SVMs) with no kernels for each descriptor at scale λ_i. In the experiments with the MSC, we used "weakened" SVMs as weak learners. More details about the implementation of SVMs as weak learners can be found in Section 5.2.4. The protocol is the same as described in Section 5.3.1.

In Section 5.4.1, we present the correlation analysis of classifiers at different scales. In Section 5.4.2, we propose an approach to select classifiers on each scale based on the accuracy and correlation of them.

5.4.1 Correlation Analysis

The first study is concerned with the analysis of the accuracy of classifiers at different segmentation scales. The second study is the correlation analysis of each pair of classifiers. In these experiments, a classifier is defined for a descriptor and a segmentation scale. We use Cor [53] to assess the correlation of two classifiers c_i and c_j:

$$COR(c_i, c_j) = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}} \tag{5.8}$$

where a is the percentage of pixels that both classifiers c_i and c_j classified correctly in the training set, b and c are the percentage of pixels that c_j hit and c_i missed and vice versa, and d is the percentage of pixels that both classifiers missed.

Classifier Accuracy for Different Segmentation Scales

Figure 5.5 and Figure 5.6 show the overall accuracy and the tau index for each SVM classifier implemented using each descriptor/scale. We observe a large difference between the accuracy results (Figure 5.5) with color and texture descriptors for almost all scales. Among the color descriptor accuracies, we have no significant difference, although BIC presents the highest values at all scales. Among the texture ones, they present almost the same accuracies at all scales except for QCCH that presents its best results at the coarser scales.

Regarding the tau indexes (Figure 5.6), which is more discriminative than overall accuracy, we observe that BIC achieves the best results for all scales. GCH also yields the best result at the coarser scale λ_5.

Among the texture descriptors, all of them are almost random at the finest scales (λ_1 and λ_2). QCCH presents the best results at the intermediate scale λ_3. The texture descriptors present their

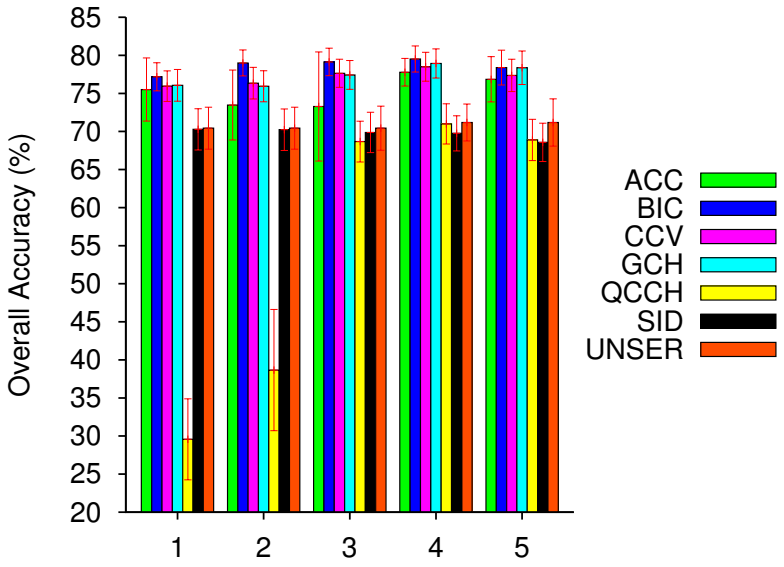


Figure 5.5: Overall accuracy for each descriptor at segmentation scales $\lambda_1, \ldots, \lambda_5$.

best results at the coarsest scales λ_4 and λ_5. At the coarsest scales, QCCH and Unser present better results than SID.

The main conclusion of this experiment is that color descriptors are very important at all scales while texture features can contribute only at the coarsest ones.

Classifier Correlation for Different Segmentation Scales

In this section, we analyze the correlation of each pair of classifiers at the segmentation scales.

Figure 5.7 shows the correlation scores considering the different descriptors and scales. We have observed that the correlation among the descriptors presents minor differences depending on the training set. We report in this section the commonest patterns observed in the experiments. Note that the correlation among the finest scales is large (scales λ_1 and λ_2), while the correlation among the coarsest scales (λ_4 and λ_5) is small. As expected, the overall correlation between scales with regions of different sizes is low. This suggests that the use of different scales improves the classification of RSI according to what have been reported in the literature.

Region **A** is related to the anti-correlation among QCCH-based classifiers at low scale and classifiers created using other descriptors. Region **B** refers to the low correlation of ACC-based classifiers at intermediary scales with other ones. That suggests that ACC-based classifiers are good candidates to be combined. Region **C** refers to the high correlation observed among the classifiers created with texture descriptors, mainly when fine scales (small regions) are considered. Finally, the region labeled with **D** refers to the high correlation score observed for

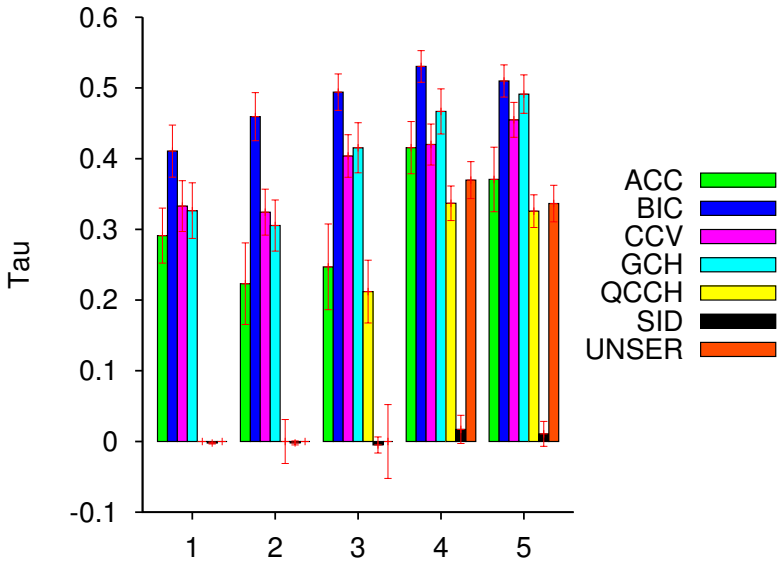


Figure 5.6: Tau index for each descriptor at segmentation scales $\lambda_1, \ldots, \lambda_5$.

CCV and GCH descriptors. Classifiers based on those descriptors are *not* good candidates to be combined.

Figure 5.8 presents the correlation coefficient (see Equation 5.8) of each pair of descriptors at the segmentation scales $\lambda_1, \ldots, \lambda_5$. Note that the smaller the segmentation scale, the higher the correlation between the descriptors. The finest scales are composed by more homogeneous and smaller regions. In such scenario, global descriptors as those used in our experiments have less visual patterns to encode. This may be one of the reasons why region-based methods have presented better results than traditional pixel-based classification in the literature when high-resolution RSIs are considered. One exception occurs with ACC. For this descriptor, its correlation with other descriptors decreases until the intermediate scale (scale λ_3). From that scale on, the observed correlation increases. We can also observe that CCV and GCH are very correlated at all scales. QCCH is not well correlated with other descriptors at scale λ_1. That is expected given its poor accuracy performance at that scale (see Figure 5.5).

In face of the results above, most promising combination would involve the classifiers implemented with color descriptors, at all scales. Some examples are ACC and BIC at λ_4, and BIC and GCH at λ_5. With regard to texture descriptors, one should consider only the created classifiers considering scales with large regions.

Finally, with this experiment we can conclude that combining descriptors improves the classification results, but some descriptors contribute more than others and that depends on the scale. Furthermore, we assume that low correlated classifiers are good candidates to be combined as

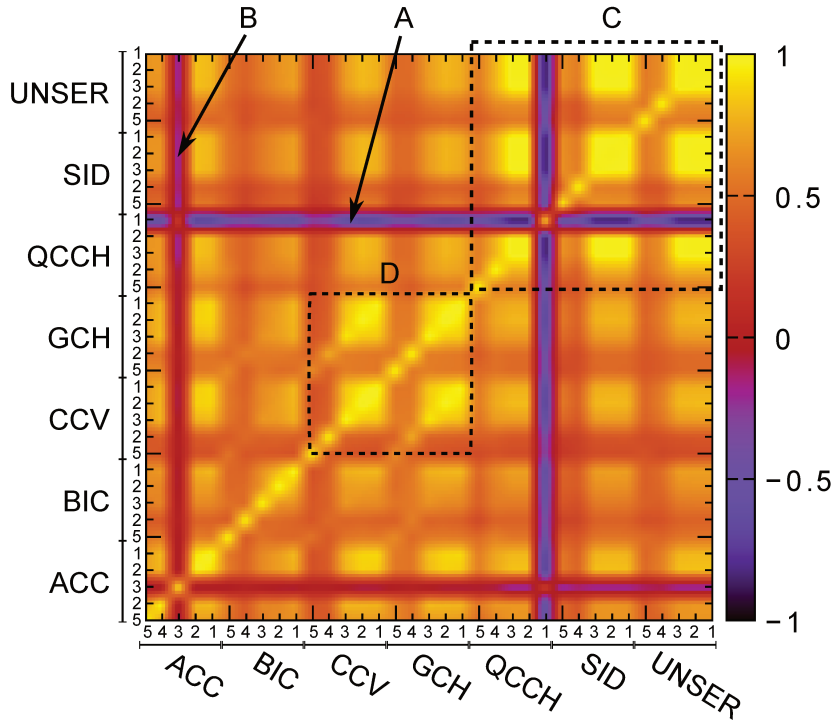


Figure 5.7: Complete correlation coefficients for each descriptors at the segmentation scales $\lambda_1, \ldots, \lambda_5$.

proposed in [6].

5.4.2 Selection of Descriptors

As we observed that not all scales and descriptors have the same contributions and that some classifiers might be very correlated to others, we need to devise a method to select the most promising combination pair (descriptor, classifier).

The simplest idea is to select the most accurate classifiers/descriptors for combination. However, by using only the overall accuracy as the majority of works in the literature, we can have a wrong notion about the results, mainly in binary classification problems. Therefore, we design a simple strategy using two other variables to select classifiers. The first is the tau index, which can be interpreted as a measure of difference to the classification randomly obtained. We used tau because it is more discriminative than the overall accuracy. The other one is the correlation between pairs of classifiers. Correlation gives a notion of diversity that can be used to select classifiers specialized in different kinds of features or subclasses and captures the most appopriate ones to be combined.

Consider a plane where *x* and *y axes* represent the tau index and the correlation of a pair of

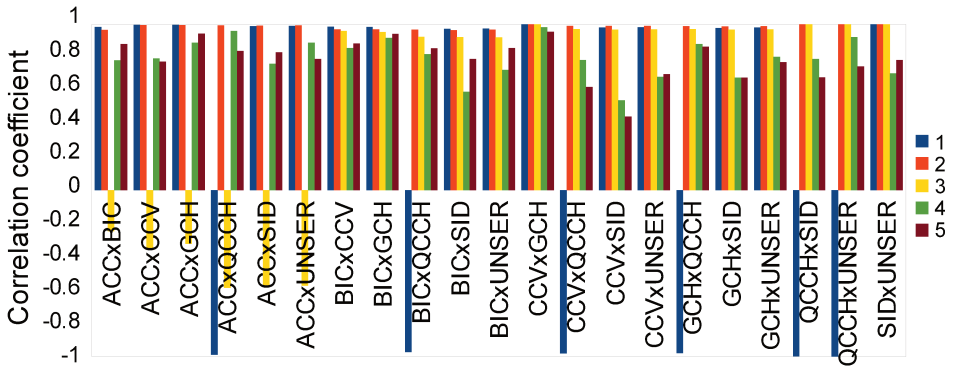


Figure 5.8: Correlation of pairs of classifiers for different segmentation scales.

classifiers, respectively. Let $\mathcal{C}$ be the set of pairs of classifiers in a given scale. The position $P_{(c_i,c_j)}$ of a pair of classifiers $c_i \in \mathcal{C}$ and $c_j \in \mathcal{C}$ on this plane is defined by the ordered pair $P_{(c_i,c_j)} = (Cor(c_i, c_j), \frac{\tau_{c_i}+\tau_{c_j}}{2})$, where $Cor(c_i, c_j)$ is the correlation of classifiers c_i and c_j, given by Equation 5.8, and τ_{c_i} and τ_{c_j} are the classification effectiveness measured using the tau index for classifiers c_i and c_j, respectively. Both the correlation and the tau index are computed on the validation set. Figure 5.9 shows the distribution of pairs of classifiers considering the λ_5 scale for one of the validation sets. Similar distributions are computed for all scales.

An ideal pair of classifiers should have low correlation and high tau index. Let $\mathcal{P}$ be the position of the ideal pair of classifiers. In our approach, $\mathcal{P} = (1.0, 0.0)$. The set $\mathcal{R}$ of selected pairs of classifiers for a given scale is defined by the K-nearest neighbours of $\mathcal{P}$:

$$K - NN(\mathcal{P}) = \{\mathcal{R} \subseteq \mathcal{C}, |\mathcal{R}| = K \wedge \forall x \in \mathcal{R}, y \in \mathcal{C} - \mathcal{R} : \rho(\mathcal{P}, x) \leqslant \rho(\mathcal{P}, y)\}$$

where ρ is the distance between two points. In our case, we use the Euclidean distance. We use this strategy with $K = 1$ to select the nearest pair of classifiers to the ideal position for each scale. Since we consider five scales, 10 classifiers are selected for combination.

We perform experiments using the MSC approach to assess the effectiveness of our selection strategy. However, any other method could be used without loss of generalization. The objective is to show that the effectiveness of MSC is the same, when it uses the small set of relevant classifiers selected by our approach. The MSC, which is based on boosting of weak classifiers, defines a weight for each selected classifier along T rounds. The "strong" final classifier is a linear combination of these weak classifiers, as detailed in Section 5.2.

Table 5.10 presents the average overall accuracy (O. A.), Kappa, and Tau measures of the MSC considering the 10 selected classifiers by our strategy (MSC_{10}) and using all available classifiers (MSC_{35}, five classifiers per scale). One can see that the accuracies are almost the

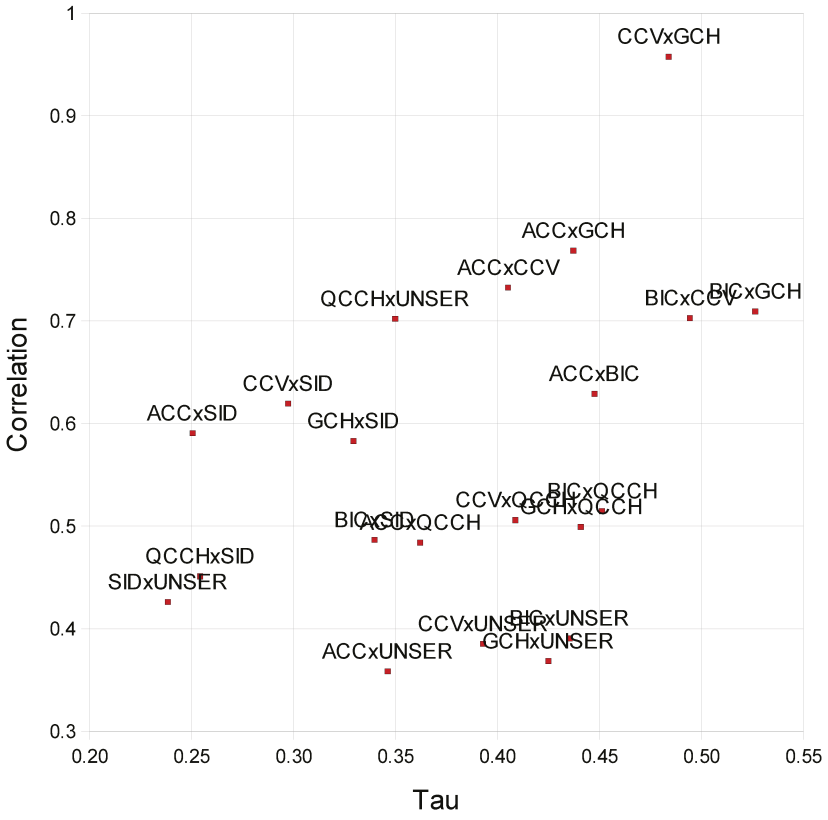


Figure 5.9: Distribution of pairs of classifiers considering the λ_5 scale for one of the validation sets.

same. The time spent for training MSC, however, are very different. MSC_{10} takes around $9h$, while MSC_{35} takes $16h$. Table 5.11 shows the weight computed by MSC for weak classifiers selected across the training rounds, considering all classifiers and those 10 found by our selection strategy. As it can be observed, the set of weak classifiers and their weights are almost the same for both configurations.

Table 5.10: Classification results using 10 and 35 classifiers.

Method	**O.A. (%)**	**Kappa** (κ)	**Tau** (τ)
MSC_{10}	82.01 ± 1.11	0.7775 ± 0.02	0.6203 ± 0.02
MSC_{35}	82.28 ± 0.99	0.7800 ± 0.02	0.6321 ± 0.01

Table 5.11: Weak classifiers chosen by the MSC for each round t considering 10 automatically selected classifiers and all 35 classifiers.

	MSC_{10}		MSC_{35}	
	Classifier	Weight	Classifier	Weight
0	BIC,λ_3	0.73	BIC,λ_3	0.73
1	BIC,λ_5	0.21	BIC,λ_5	0.21
2	Unser,λ_4	0.10	Unser,λ_4	0.10
3	Unser,λ_5	0.02	GCH,λ_4	0.10
4	BIC,λ_5	0.16	BIC,λ_5	0.16
5	ACC,λ_2	0.25	GCH,λ_5	0.18
6	Unser,λ_5	0.08	ACC,λ_3	0.20
7	ACC,λ_1	0.07	CCV,λ_2	0.15
8	BIC,λ_1	0.21	ACC,λ_5	0.14
9	BIC,λ_5	0.12	GCH,λ_5	0.08

5.5 Conclusions

The proposed approaches for multiscale image analysis are the Multiscale Classifier (MSC) and the Hierarchical Multiscale Classifier (HMSC). The MSC is a boosting-based classifier that builds a strong classifier from a set of weak ones. The HMSC is also based on boosting of weak classifiers, but it adopts a sequential strategy of training, according to the hierarchy of scales (from the coarsest to the finest). The experimental results indicate that the BIC descriptor is presently the most powerful descriptor to detect regions of coffee. The MSC results show that the combination of scales increases the power of the final classifier. The HMSC results, in turn, demonstrate that it is possible to speed up the training time and keep the quality of the final classifier.

In this chapter, we also have performed experiments to analyse the correlation among descriptors and the segmentation scales. Coarser scales offer great power of description while the finer ones can improve the classification by detailing the segmentation. Another branch of studies confirmed that the use of different descriptors is important. However, the descriptors do not contribute equally at all scales.

Chapter 6

Interactive Classification of RSIs based on Active Learning

6.1 Introduction

In this chapter, we present the proposed method for interactive classification of remote sensing images considering multiscale segmentation. Our aim is to improve the selection of training samples using the features from the most appropriate scales of representation. Figure 6.1 gives an overview of the architecture used in our approach for interactive classification. This kind of architecture is very common in information retrieval systems with relevance feedback [31, 88, 27, 21, 35]. The framework is composed of three main processing modules: segmentation, feature extraction, and classification. Segmentation and feature extraction are offline steps. When an image is inserted into the system, the segmentation is performed, building a hierarchical representation of regions. Feature vectors from these regions are then computed and stored.

The interactive classification starts with the user's annotation. He/she selects a small set of relevant and non-relevant pixels. Using these pixels as training set, the method builds a classifier to label the remaining pixels. Although the training set is at the pixel level, the training is performed by using features extracted from the segmented regions for each considered scale. At the end of the classification step, the method selects regions for possible feedback. When the result of the classification is displayed, the user feeds the system by labeling the region with the correct class. These steps are repeated until the user finishes the process. The final classification is a multiscale result combining all scales of segmentation.

For the training stage, we propose a kind of boost-classifier adapted to the segmentation, which takes advantage of various region features. In each iteration, this method builds a strong classifier from a set of weak ones. The weak classifiers are SVMs (Support Vector Machines) with a linear kernel, each trained for one feature descriptor of one scale of segmentation. We use

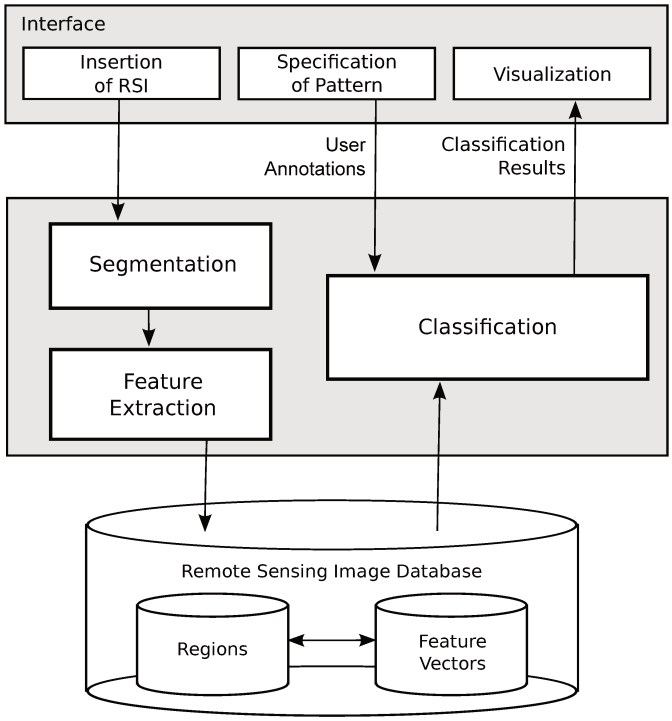


Figure 6.1: Architecture of the interactive classification system.

a boosting-based active learning strategy to select regions for user's relevance feedback. The idea is to select the closest regions to the border separating both target classes. Those regions are theoretically the most uncertain regions.

Experimental results show that the combination of scales produces better results than isolated scales in a relevance feedback process. The interactive method achieves good results with few iterations. Furthermore, by using only 5% of the pixels of the training set, our approach can build classifiers which are as strong as the ones generated by a supervised method using the whole training set (see HMSC approach in Chapter 5).

This chapter is outlined as follows. Section 6.2 introduces our method for multiscale training and classification. Experimental results are presented in Section 6.3. In Section 6.4, we present our conclusions.

6.2 The Proposed Interactive Classification Method

At the end of the off-line process, the image is represented by a set of several nested partitions. Each region of each scale is described by a feature vector. The classification step is performed on-line, through an interactive process of region classification.

Given a set of labeled pixels as training set Y_0 and features extracted from regions of various scales, our method aims at producing a classifier to label the remaining pixels. Moreover, the method uses an active learning strategy based on user interaction to increase the training set and, hence, the classification results. Algorithm 2, which presents the proposed interactive classification process, will be further explained in the next section.

Algorithm 2 The interactive classification process.

1 *Annotation of the initial training set* Y_0 = *label of pixels* (see Section 6.2.3)
2 Build a classifier $F_0(p)$ using multiscale training (see Section 6.2.1)
3 Classify image I by using $F_0(p)$
4 $i \leftarrow 1$
5 **while** user is not satisfied **do**
6 Select uncertain regions Q_i in the classified image (see Section 6.2.2)
7 *Annotation of the selected regions* Q_i (see Section 6.2.3)
8 Update the training set $Y_i \leftarrow Y_{i-1} \bigcup Q_i$
9 Build a classifier $F_i(p)$ using multiscale training (see Section 6.2.1)
10 Classify image I by using $F_i(p)$
11 $i \leftarrow i + 1$
12 **end while**

Algorithm 2 starts the process with the definition of the training set Y_0 annotated by the user (line 1). We consider that, in a real scenario, the samples indicated by the user may not be always representative. The training set is used to build a multiscale classifier $F_0(p)$ (line 2). This approach is based on the boosting of weak classifiers (see Chapter 5). The multiscale classifier $F_0(p)$ is used to classify the whole image I (line 3). The feedback process starts using this initial classification result. In the loop, the user can stop the classification process or continue the classification refinement process (line 5). For selection of regions displayed for user annotation, also known as active learning, we exploit the notion of separating border in AdaBoost, which is originally proposed in [120]. In the refinement iterations, the following steps are performed: selection of the most uncertain regions in each scale λ (line 6); annotation of the selected regions by the user (line 7); update of the training set by adding the new labeled regions to Y_i (line 8); multiscale training by using the new set Y_i (line 9); reclassification of the whole image I by using $F_i(p)$ (line 10).

The proposed approach is designed to assist specialists, who are our final users. Our approach expects the user to have reasonable knowledge about the region and the targets of interest. A way to create a stopping criterion is to define some validation points (it can be pixels) in the image as usually done by experts to assess the quality of supervised classification in practical situations. A validation point is a well-known place in the scene which is not used to train and can be used to evaluate classification results. When the method achieves acceptable

accuracy in the validation points, the user can stop the interactive process. Another option is to previously determine a number of iterations. It is important to clarify that, besides seeing the regions selected for annotation, the user can also check the classification results.

We explain in details each step of the process in the following sections. In Section 6.2.1, we present the multiscale classification based on boosting. The active learning process is explained in Section 6.2.2. In Section 6.2.3, we present how user annotation is carried out.

6.2.1 Multiscale Training/Classification

We adapted *hierarchical multiscale classifier* (HMSC), presented in Section 5.2.3 to perform multiscale training between each user interaction. The main difference is that this version of the HMSC does not consider the use of a validation set, since the training data is very small.

In each stage/scale, the proposed method repeatedly calls *weak learners* in a series of rounds $t = 1, \dots T$. Each weak learner creates a weak classifier that decreases the expected classification error of the combination. The algorithm then selects the weak classifier that most decreases the error.

For each scale λ, the weak learner produces a set $\mathcal{S}_\lambda$ of weak classifiers $\{h_{t,\lambda}\}$. The *multiscale classifier* (F) is a combination of the set of weak classifiers $\mathcal{S}_\lambda(p)$ selected for each scale λ:

$$F(p) = \text{sign}\Big(\sum_{\lambda_i} \mathcal{S}_{\lambda_i}(p)\Big) = \text{sign}\Big(\sum_{\lambda_i}\sum_{t}^{T} \alpha_{t,\lambda_i} h_{t,\lambda_i}(p)\Big) \tag{6.1}$$

The strategy of building a multiscale classifier consists in keeping a set of weights over the training set. These weights can be interpreted as a measure of the level of difficulty to classify each training sample. At the beginning, the pixels have the same weight, then in each round, the weights of misclassified pixels are increased. Thus, in the next rounds the weak learners focus on difficult samples. We will note $W_t(p)$ the weight of pixel p in round t, and $D_{t,\lambda}(R)$ the misclassification rate of region R in round t at scale λ given by the mean of the weights of its pixels:

$$D_{t,\lambda}(R) = \Big(\frac{1}{|R|}\sum_{p\in R} W_t(p)\Big) \tag{6.2}$$

Algorithm 3 presents the boosted-based training used in each stage described in Figure 5.2. Let $Y_\lambda(R)$, the set of labels of regions R at scale λ, be the training set. In a series of rounds $t = 1, \dots T$, for scale λ, the weight of each region $D_{t,\lambda}(R)$ is computed (line 3). This piece of information is used to select the regions to be used for training the weak learners, building a subset of labeled regions $\hat{Y}_{t,\lambda}$ (line 5). The subset $\hat{Y}_{t,\lambda}$ is used to train the weak learners with each feature $\mathcal{F}$ at scale λ (line 6). Each weak learner produces a weak classifier $h_{t,(\mathcal{F},\lambda)}$ (line 8).

The algorithm then selects the weak classifier h_t that decreases the error $Err(h, W)$ the most (line 10). The level of error of h_t is used to compute the coefficient α_t, which indicates the degree of importance of h_t in the final classifier (line 11). The selected weak classifier h_t and the coefficient α_t are used to update the weights of the pixels $W_{(t+1)}(p)$ which can be applied in the next round (line 12).

Algorithm 3 The boosted-based training.

Given:

Training labels $Y_\lambda(R) =$ labels of some regions R at scale λ

Initialize:

For all pixels p, $W_1(p) \leftarrow \frac{1}{|Y_0|}$, where $|Y_0|$ is the number of pixels in the image level

1 For $t \leftarrow 1$ to T do
2 For all $R \in P_\lambda$ do
3 Compute $D_{t,\lambda}(R)$
4 End for
5 Build $\hat{Y}_{t,\lambda}$ (a training subset based on $D_{t,\lambda}(R)$)
6 For each feature type $\mathcal{F}$ do
7 Train *weak learners* using features $(\mathcal{F}, \lambda)$ and training set $\hat{Y}_{t,\lambda}$.
8 Evaluate resulting classifier $h_{t,(\mathcal{F},\lambda)}$: compute $Err(h_{t,(\mathcal{F},\lambda)}, W))$ (Equation 6.3)
9 End for
10 Select the *weak classifier* h_t whose $Err = \text{argmin}_{h_{t,(\mathcal{F},\lambda)}} Err(h_{t,(\mathcal{F},\lambda)}, W_{t,\lambda})$
11 Compute $\alpha_t \leftarrow \frac{1}{2} ln\left(\frac{1+r_t}{1-r_t}\right)$ with $r_t \leftarrow \sum_p cY_0(p)h_t(p)$
12 Update $W_{t+1}(p) \leftarrow \dfrac{W_t(p)\exp\left(-\alpha_t Y_0(p)h_t(p)\right)}{\sum_p W_t(p)\exp\left(-\alpha_t Y_0(p)h_t(p)\right)}$
13 End for

Output: Classifier $\mathcal{S}_\lambda(p)$

The classification error of classifier h is:

$$Err(h, W) = \sum_{p|h(p)Y_0(p)<0} W(p) \tag{6.3}$$

The training is performed on the training set labels Y_λ corresponding to the same scale λ. The weak learners (linear SVM, for example) use the subset $\hat{Y}_{t,\lambda}$ for training and produce a weak classifier $h_{t,(\mathcal{F},\lambda)}$. The training set labels Y_0 are the labels of pixels of image I, and training sets labels Y_λ with $\lambda > 0$ are defined according to the rate of pixels belonging to one of the two classes (for example, at least 80% of one region).

The idea of buiding the subset $\hat{Y}$ is to force the classifiers to train with the most difficult samples. The weak learner should allow the most difficult samples to be differentiated from the other ones according to their weight. Thus, the strategy of creating $\hat{Y}$ is directly dependent on the configuration of the weak classifier and may contain all regions, since the classifier considers the weights of the samples.

At the end of each stage, we withdraw the easiest samples. Let W_i be the weights of the pixels after training with scale λ_i. We denote $D_i(R_{i+1})$ the weight of region $R_{i+1} \in P_{\lambda_{i+1}}$, which is given by:

$$D_i(R_{i+1}) = \left(\frac{1}{|R|} \sum_{p \in R} W_i(p) \right) \tag{6.4}$$

where $W_i(p)$ is the weight of pixel $p \in R$ concerning scale λ_i.

The set of regions $\breve{Y}_{i+1}$ to be used in the training stage with scale λ_{i+1} is composed by the regions $R_{i+1} \in P_{\lambda_{i+1}}$ with mean $D_i(R_{i+1}) > \frac{1}{2|Y_0|}$. This means that the regions that ended a training stage with distribution equal to half the initialization value $\frac{1}{|Y_0|}$ are discarded from one stage to another in the hierarchical training (see Figure 5.2).

6.2.2 Active Learning

Active learning is a machine learning approach which aims at obtaining high classification accuracy using very few training samples [40]. It attempts to overcome the training sample selection by asking queries in the form of unlabeled instances to be labeled by the user. The main challenge is to find the most "informative" samples, i.e., once added to the training set, the ones which lead the system to build the best classification function.

Active learning is widely used in the literature, even in remote sensing community, in applications based on SVM [100]. These approaches exploit the notion of minimum marginal hyperplane in SVMs, to select representative samples. The general strategy consists in selecting the unlabeled samples that are closer to the separation margin.

Nevertheless, many approaches have been proposed to perform active learning in boosting-based methods [120, 50, 64, 123]. We adopted the active learning strategy (active AdaBoost) proposed by Lee et al. [120]. They proposed a geometrical representation of AdaBoost output. In this representation, each sample is a point in a version space. Each point is based on the label provided by each weak learner. Therefore, each weak classifier corresponds to a dimension in this space.

Let $\mathcal{S}$ be the output of Algorithm 3. $\mathcal{S} = 0$ can be interpreted as a separating hyperplane in the version space. The strategy proposed by Lee et al. consists in maximizing the distance of the samples to the separating hyperplane by selecting the most uncertain samples in each feedback

iteraction. We adapt this idea to our problem, by computing for each scale λ, the closest sample (corresponding for a region) to the hyperplane.

Let $\mathcal{S}_\lambda(p)$ be the output of training at scale λ, the distance of pixel p to separating hyperplane $g(p)$ is:

$$g(p) = |\sum_{\lambda_i} \mathcal{S}_{\lambda_i}(p)| = |\sum_{\lambda_i} \sum_{t}^{T} \alpha_{t,\lambda_i} h_{t,\lambda_i}(p)| \tag{6.5}$$

The distance of region $R \in P_\lambda$ to the separating hyperplane $g(R)$ is given by:

$$g(R) = \Big(\frac{1}{|R|} \sum_{p \in R} g(p)\Big) \tag{6.6}$$

Thus, the region corresponding to the minimal distance to the separating hyperplane g_λ^- for scale λ is defined as:

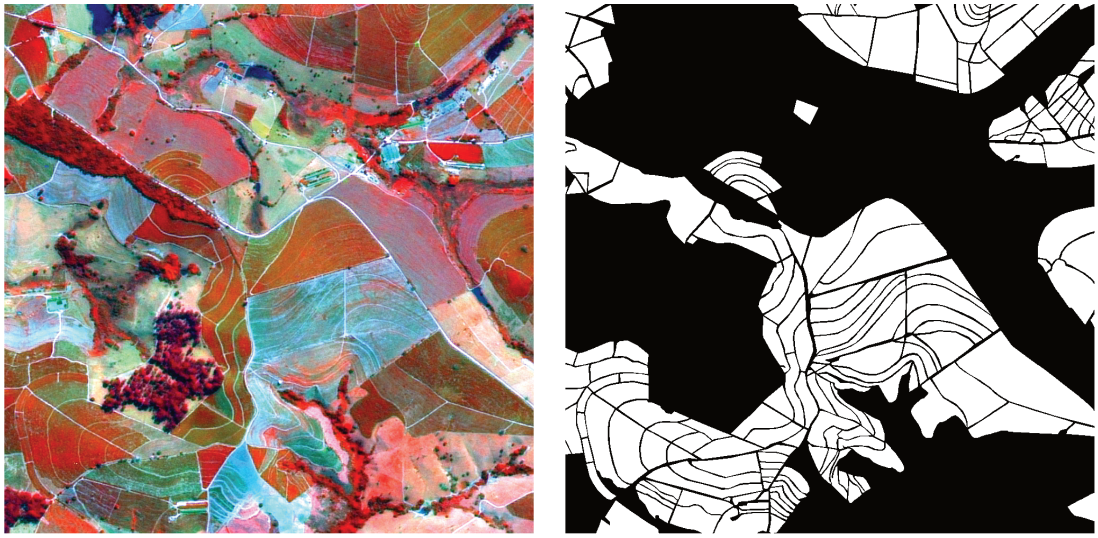
$$g_\lambda^- = \underset{R \in P_\lambda}{\operatorname{argmin}}\, g(R) \tag{6.7}$$

Equation 6.6 gives a measure of the degree of doubt to classify an unlabeled region. Figure 6.2 shows an example of classification with different classification levels. In this figure, the white regions represent the class of interest (coffee), while the black represents non-interest regions. The redder the region, the closer to the decision function, i.e., the more interesting for user feedback.

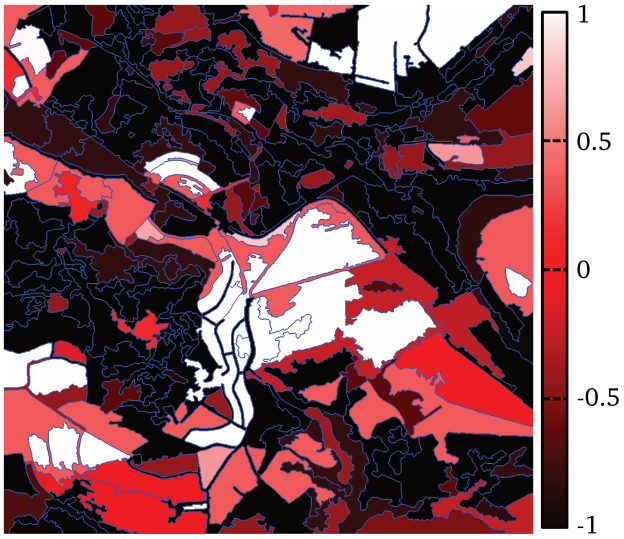
6.2.3 User Interaction

Our system is strongly interactive. This means that the user is in control of the classification by introducing new examples and counter-examples to the supervised classifier at each feedback step. The classification is performed on regions of various scales, but the final result is a classification of pixels.

At first, the user has to indicate a few areas of each class. Let us remind that we have two classes, one is the class of interest (i.e., coffee) and the other one is the rest of the image (non-coffee). There are different alternatives to label pixels, from which the system is going to obtain the first region samples. The most naïve way is to label pixels as belonging to one class or to the other. It is surely a laborious and time-consuming strategy to get enough region samples to start the classification. However, this strategy can be used at the end of the classification process to refine the final classification. Another commonly used approach is to draw rectangles or polygons on the image, whose class is known for sure (examples and counter-examples). Another tool often provided to users is a *brush*, with which users can identify the target classes by painting regions on a RSI.



(a) (b)



(c)

Figure 6.2: Example of classification with different degrees of doubt: (a) original imagem, (b) ground truth and, (c) regions with different classification levels. In (b) and (c), white and black regions are coffee and non-coffee crops, respectively. The redder the region, the closer to the decision function.

For all cases, the system has to translate the sets of pixels labeled by the user into a set of regions. This can be achieved by a majority vote scheme: if a region is covered by a certain percentage (for example more than 80% for the coarse scale) of pixels indicated by the user as belonging to one class, the region is used as example of this class.

Surely when the image is segmented, it is faster to directly annotate regions as examples or counter-examples for the current query [41]. In the simulation of interaction we present in the experiment section, we cannot use the regions, since our system works with several scales of segmentation. Therefore, we use rectangles drawn from inside regions whose label is known for sure. During the feedback iterations, intermediate results of classification are displayed to the user. The method selects a region at each scale. The number of regions may be lower than the number of scales if there is an intersection between the selected regions at two or more scales. In these cases, the coarsest region is selected. In our approach, the user annotates requested regions by scratching/brushing the pixels of each class as illustrated in Figure 6.3.

Figure 6.3 (a) illustrates the regions selected to be annotated. The user annotates the pixel classes by scratching/brushing the regions. Figure 6.3 (b) shows an example of annotation. In this example, positive samples are in green and negative samples are in red. The labels are then propagated to the other pixels of the selected regions as in Figure 6.3 (c). The remaining region pixels receive the same label of the nearest pixel annotated by the user.

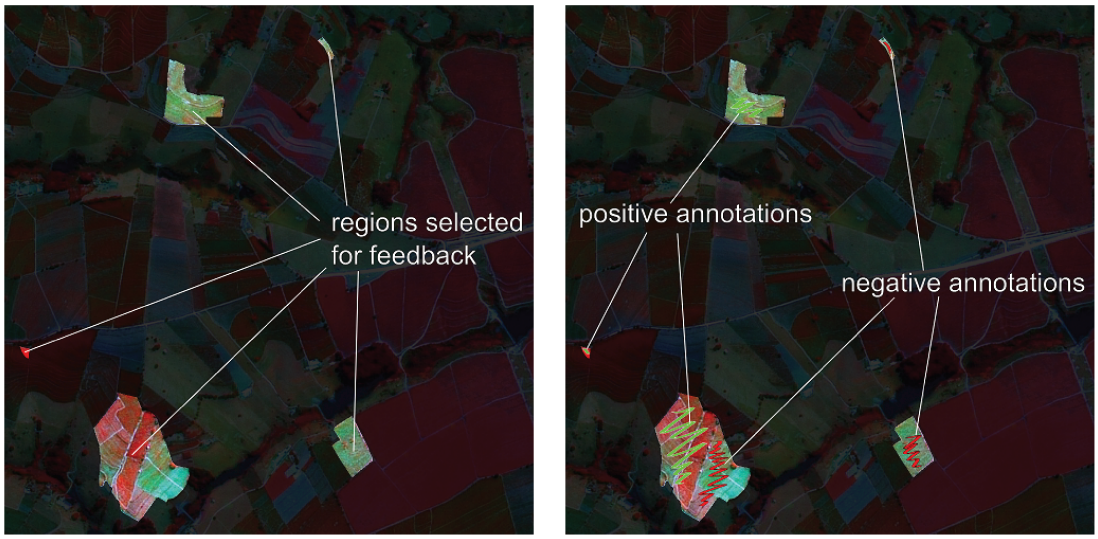
6.3 Experiments

In this section, we present the experiments performed to validate our method. They were carried out to address the following research questions:

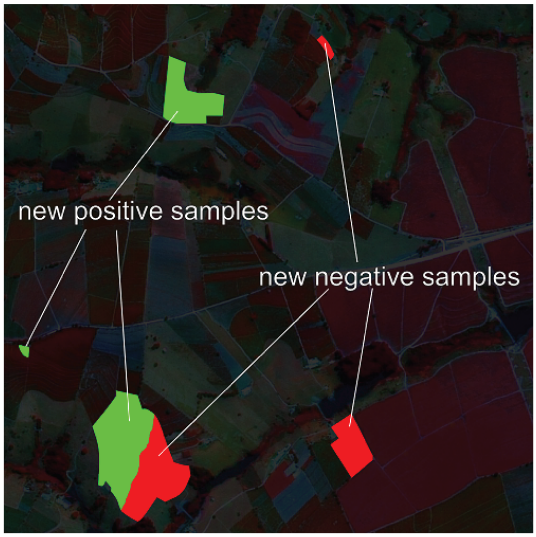
- Is the proposed multiscale approach for interactive classification effective in RSI classification tasks (Section 6.3.2)?
- Is the interactive method more effective than supervised classifiers built on a large training set (Section 6.3.3)?

We used a similar protocol as described in Section 5.3. The results of the experiments described in Section 6.3.2 were obtained considering all combinations of the five images used, training with three of them and testing in the same three images. The results of the experiments described in Section 6.3.3 were obtained considering all combinations of the 5 used subimages (3 for training and 2 for testing). In the experiments, we also used 5 subimages from the COFFEE and URBAN datasets.

We considered five different scales to extract features from λ_1 (the finest scale) to λ_5 (the coarsest one). We selected the scales according to the principle of dichotomic cuts (see Section 2.2). For the COFFEE dataset, at λ_5 scale, subimages contain between 200 and 400 regions



(a) (b)



(c)

Figure 6.3: Example of the process of regions annotation for user feedback: (a) regions selected for annotation; (b) user annotations, and (c) annotations converted into labeled pixels.

Table 6.1: Accuracy analysis of classification for the example presented in Figure 6.5 (TP = true positive, TN = true negative, FP = false positive, FN = false negative).

Feedback Step	TP	TN	FP	FN
9	117,322	688,431	51,737	142,510
10	119,225	722,912	17,256	140,607
40	167,255	712,611	27,557	92,577

while, at scale λ_1, they contain between $9,000$ and $12,000$ regions. For the URBAN dataset, at λ_5 scale, subimages contain between 40 and 100 regions while, at scale λ_1, they contain between $4,000$ and $5,000$ regions.

In the experiments, the ground truth for unlabeled regions are used to simulate the user annotations. A similar strategy was adopted in [24, 21], as well as in content-based image retrieval methods based on relevance feedback [35]. The initial annotation was simulated by randomly selecting a small set of contiguous pixels from the training set. In the remaining steps, we used all pixels in the selected regions as user annotations, which is the process described in Section 6.2.3.

6.3.1 Interactive Classification Example

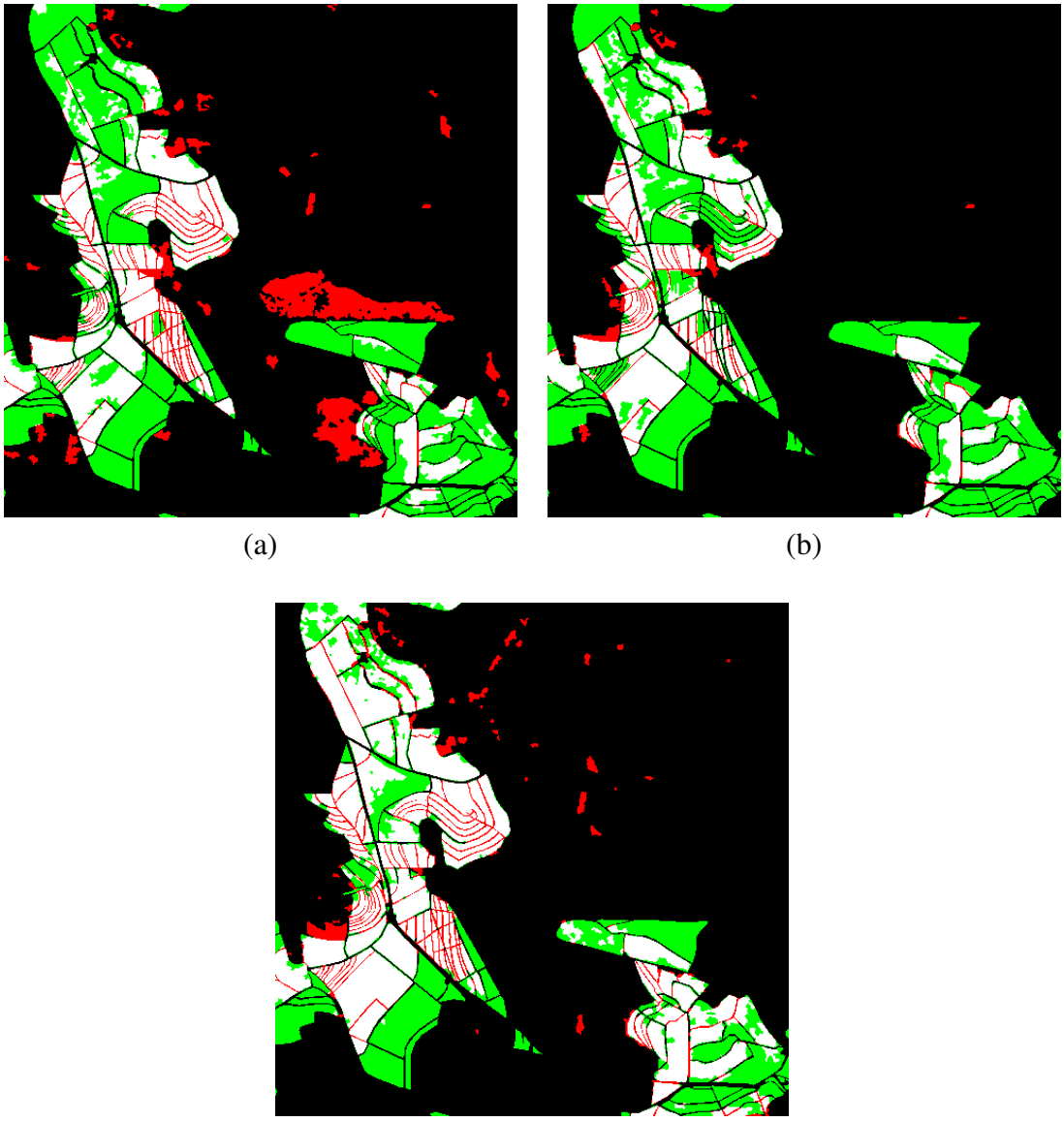
In this section, we present an example of a result of the proposed method for interactive classification. Figure 6.4 presents the results for one of the tested images from the COFFEE dataset compared to the original image and the ground truth. This image is composed of several regions of coffee, pasture, native forest, and some lakes.

As the method begins with a very small training set, the "Initial Result" is visually different from the ground truth. One reason is that the training set may not have been large enough to correctly classify regions. With the gradual increase in the training set, the results improve until the fourth iteration (OA=83.55% κ=0.8031). Between the fifth and the ninth feedback steps, we can note many variations in the results due to confusion between: 1) "new coffee" crops and pasture; and 2) "mature coffee" and native forest. The result is improved and becomes more stable from the tenth feedback step on. Although the improvements are smaller, they continue along the iterations, as it can be seen from the results of feedback steps 20, 30, 40, and so on.

To better illustrate the results, Figure 6.5 presents an error analysis (false positive and false negative samples) for the result in feedback steps 9, 10, and 40. Table 6.1 presents the accuracy values.

From the feedback steps 9 to 10, one can notice a great reduction in the number of false positives (red pixels). Most of the removed pixels correspond to areas of natural vegetation. This indicates that the confusion between natural vegetation and mature coffee is reduced. Com-

Figure 6.4: Example of the results from the initial classification to the feedback step 10, 20, 30, and 40 compared to the original image and the ground truth. Coffee and non-coffee regions are represented in white and in black respectively. OA=Overall Accuracy; κ=Kappa index.



(a) (b)

(c)

Figure 6.5: A result obtained with the proposed method in feedback steps 9 (a), 10 (b), and 40 (c) for the experiment presented in Figure 6.4. Pixels correctly classified are shown in white (true positive) and black (true negative) while the errors are displayed in red (false positive) and green (false negative).

paring Figure 6.5 (b) and Figure 6.5 (c), we note that the classification seems to go through a refining process. Visually, it is possible to see small difference between the results in feedback steps 10 and 40. However, in Table 6.1, we can observe that the number of pixels corresponding to coffee regions significantly increased (from 119,225 to 167,255).

As far as time is concerned, experiments with the COFFEE dataset showed that the proposed method takes around $50s$ for each training step using one scale and the combination of the seven weak classifiers. The proposed method needs less than one hour to perform 10 steps using five scales. Considering that 10 feedback steps is a good number to get a satisfactory result of classification, one hour is not much if compared with the time usually spent to perform manual mapping of large areas [101]. Furthermore, the steps to evaluate each descriptor in the method is easily parallelizable and, hence, the training time in each interaction can be reduced in a real scenario.

Regarding the URBAN dataset, the IHMSC needs $12s$ to train on each scale since it has less regions.

6.3.2 Multiscale versus Individual Scale

In this section, we compare the classification results obtained by using individual scales against the combination of scales by using the IHMSC approach presented in Section 6.2.1 with 20 rounds for each scale. We used IHMSC to perform the individual scales experiment with 100 rounds. In these experiments, we tested all descriptors referenced in Section 4.3. The initial training set is a rectangle composed by $10,000$ pixels with both classes.

Figure 6.6 presents the Kappa $\times$ Feedback Steps curves for the COFFEE dataset. Figure 6.7 shows the Overall Accuracy $\times$ Feedback Steps curves for the COFFEE dataset.

According to the results for the COFFEE dataset, one can observe that the combination of scales presents better results than individual ones. We can note that intermediate scales (λ_4, λ_3) use more iterations to converge, but achieve better results after many feedback steps. Concerning the coarser scale (λ_5), it quickly obtains good results, but there is no improvement after 14 feedback steps. In this scenario, regions of interest for training in the coarse scales are more quickly exhausted. We conclude that the HMSC method yields reasonable results with few feedback steps. It is even able to improve them later as it allows the refinement of the training set along iterations.

Figures 6.8 and 6.9 present the Kappa $\times$ Feedback Steps curves and Overall Accuracy $\times$ Feedback Steps curves, respectively, for the URBAN dataset.

We can observe that, for the URBAN dataset, multiscale training achieves results that are better the ones for individual scales, except for the two first feedback steps in which the training set is too small. Coarse scales produce better results than finer ones. Which means that the features extracted from finer scales can not properly represent the urban areas. However, it is

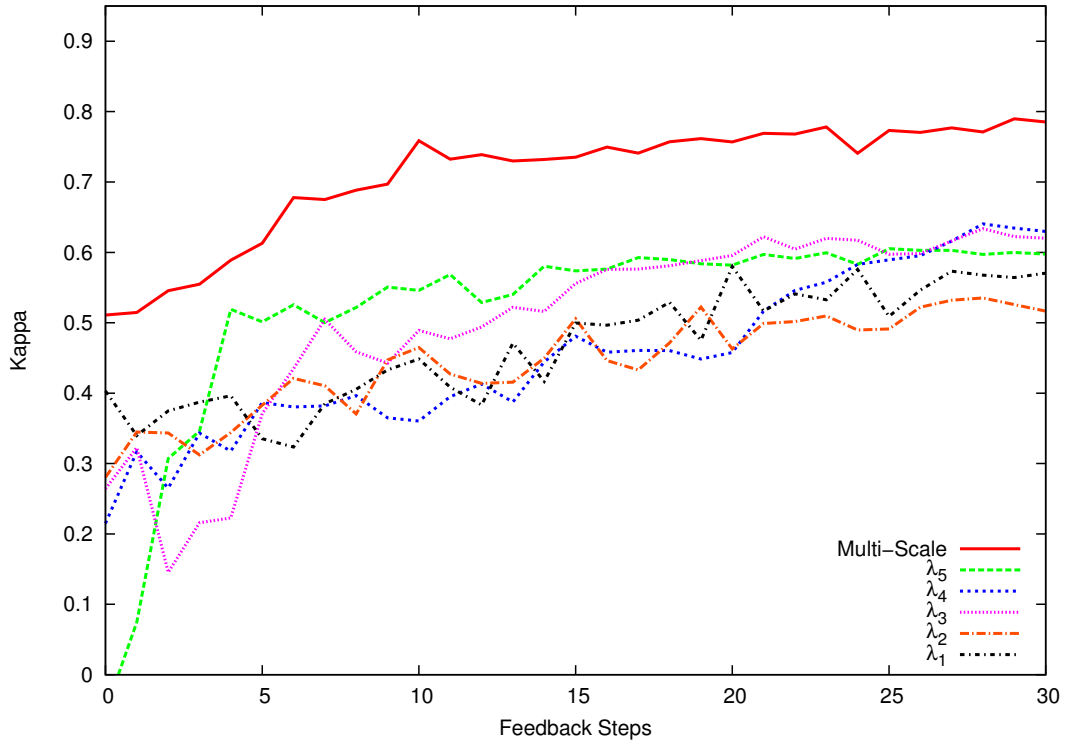


Figure 6.6: Kappa index for each iteration of feedback for the COFFEE dataset considering five scales and the multiscale classification approach.

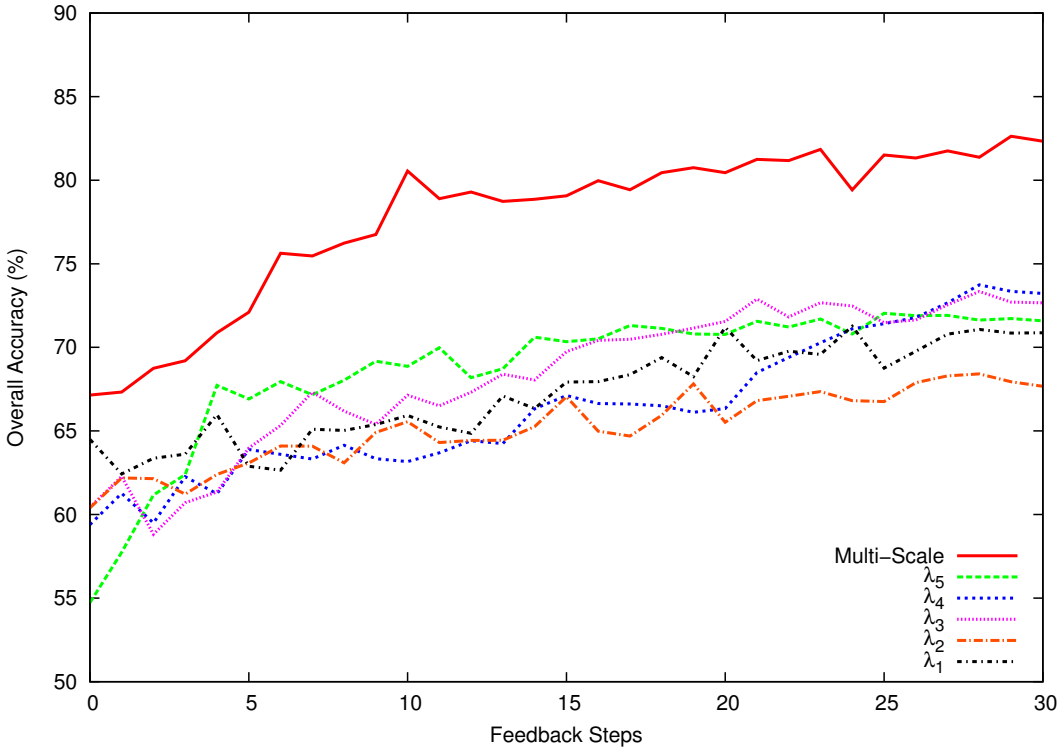


Figure 6.7: Overall accuracy for each iteration of feedback for the COFFEE dataset, considering five scales and the multiscale classification approach.

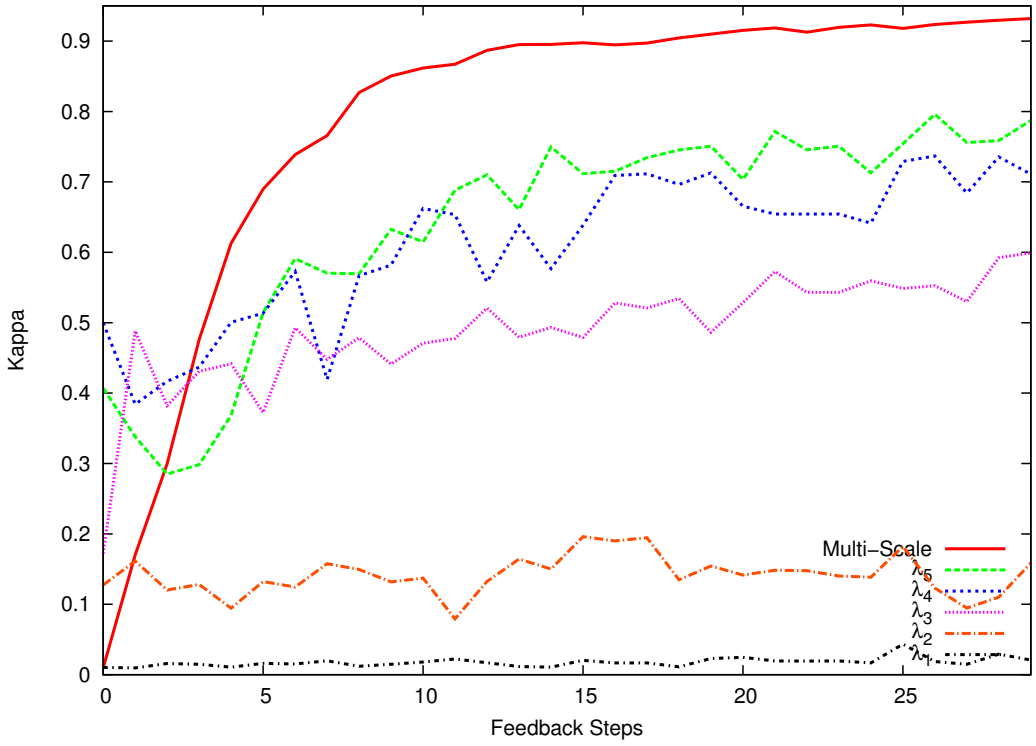


Figure 6.8: Kappa index for each iteration of feedback for the URBAN dataset, considering five scales and the multiscale classification approach.

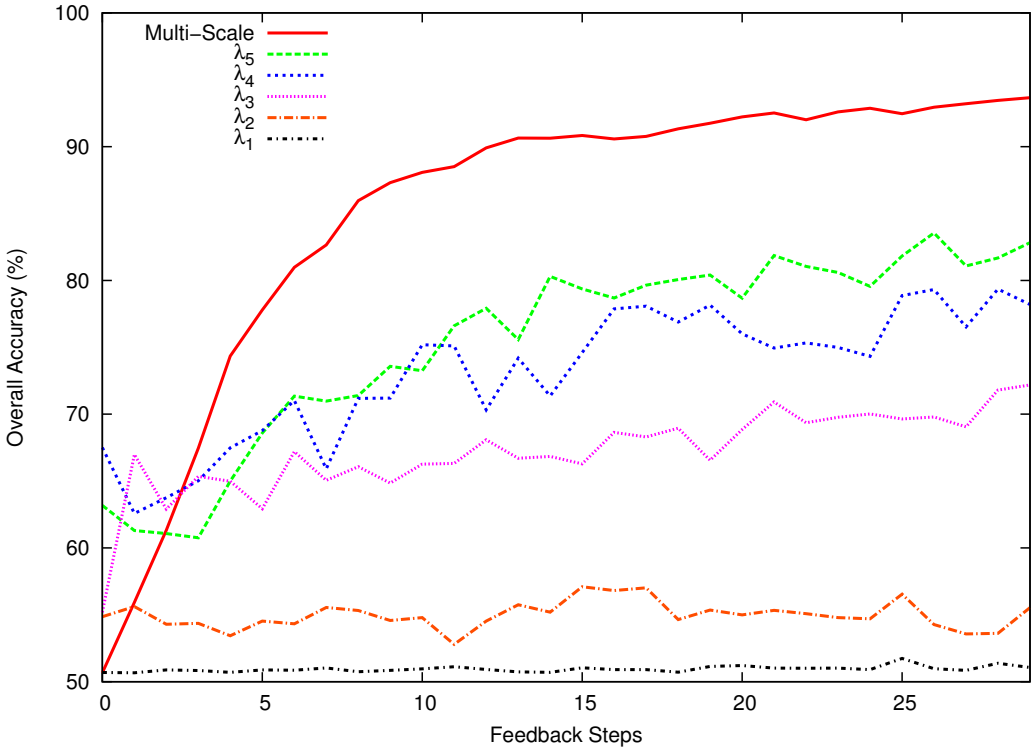


Figure 6.9: Overall accuracy for each iteration of feedback for URBAN dataset, considering five scales and the multiscale classification approach.

not difficult to understand this phenomenon. Urban areas are complex targets since they are composed by smaller objects with specific characteristics. If we use a fine scale, small objects (e.g., trees) can be present in both urban and non-urban areas. This fact makes the classification task more difficult.

6.3.3 Interactive versus Supervised Classification Strategy

In this section, we present experiments that compare the proposed method for interactive classification with a traditional supervised approach that uses the whole available training set. For this reason, the experiments of this section (including the interactive method) were performed using all combinations of the five images from our dataset: three images as training set and two images for testing. The difference is that the supervised method uses all the available training images to learn while the same information is used to simulate the user annotations in the interactive approach.

It is important to note that in a real situation the user would typically annotate and classify regions present in the same image scenes like in the experiments reported in Section 6.3.2.

We used two supervised methods as baselines. The first method is the HMSC with no user interactions using 100% of the pixels available for training (3,000,000 of pixels in this experiments). The other method is based on Tzotsos et al. [102]. They proposed a method that uses SVMs with RBF kernels to classify the regions obtained from a multiscale segmentation process. That approach outperforms the results obtained by using the software eCognition [3]. Therefore, we used SVM + RBF kernels applied to an intermediate segmentation scale defined by the Guigues method as baseline. The BIC descriptor was used in this baseline.

Figure 6.10 presents the classification results for the baselines and the Kappa $\times$ Feedback Steps curves using the COFFEE dataset. This figure also includes the histogram of the percentage of the training set used in each feedback step by the proposed interactive method. Figure 6.11 presents the same classification results using the COFFEE dataset, but using the Overall Accuracy. The interactive HMSC training set starts with two rectangles composed of $5,000$ pixels for each class (coffee and non-coffee). It corresponds to 0.33% of the training set.

According to the results of Figures 6.10 and 6.11, HMSC has Kappa equal to 0.77 and overall accuracy equal to 82%. SVM has Kappa equal to 0.71 and overall accuracy equal to 77%. The interactive method starts with Kappa index equal to 0.15 and overall accuracy equal to 57%. After 20 iterations, the results converge to Kappa index equal to 0.76 and overall accuracy equal to 81%.

One can note that the interactive $HMSC$ obtains similar results to the SVM baseline after 5 feedback steps. After 20 feedback steps, interactive HMSC obtains results close to the supervised HSMC. Therefore, these experiments show that by using about 1% of the pixels in the training set, we can obtain results close to SVM. By using a little bit more than 5% of the

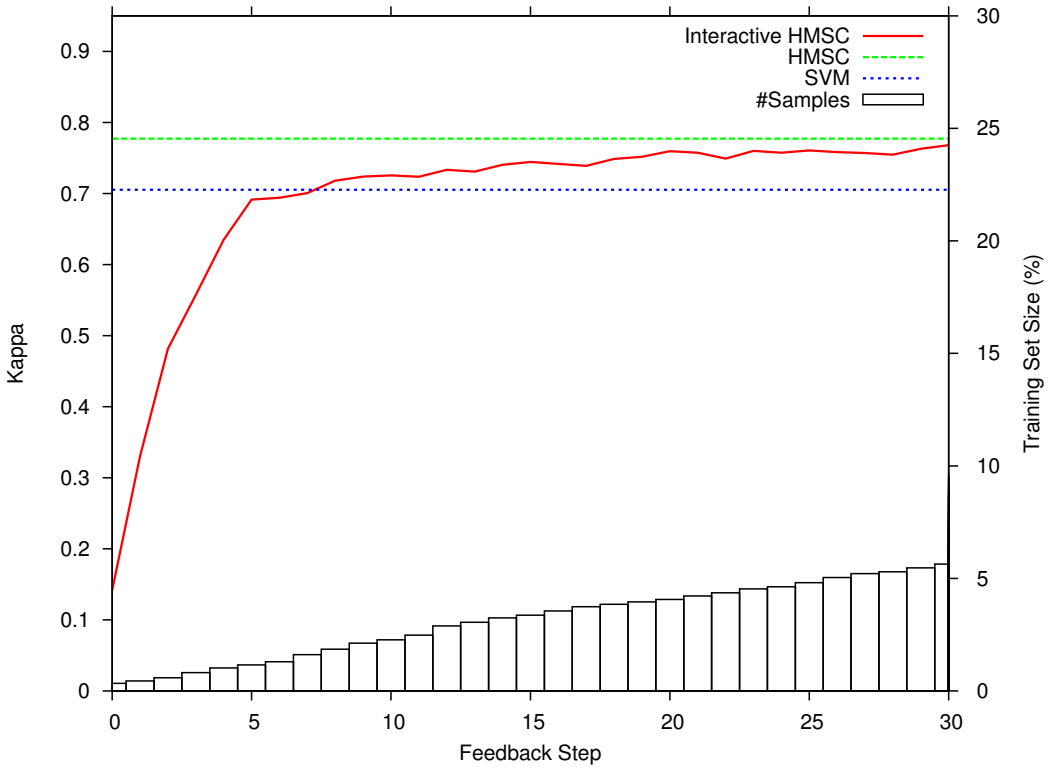


Figure 6.10: Kappa index for the HMSC and SVM and Kappa $\times$ Feedback Steps curves for interactive HMSC using the COFFEE dataset. The histogram represents the percentage of the training set used in the interactive method.

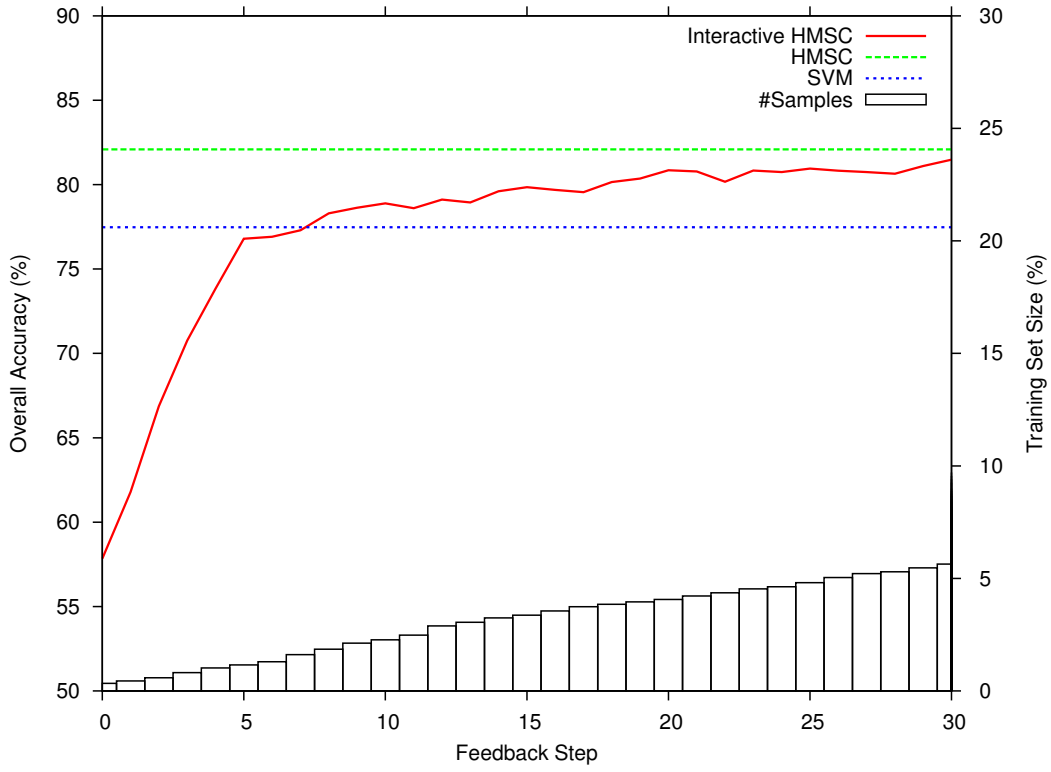


Figure 6.11: Overal Accuracy results for the HMSC and SVM and Overal Accuracy $\times$ Feedback Steps curves for interactive HMSC using the COFFEE dataset. The histogram represents the percentage of the training set used in the interactive method.

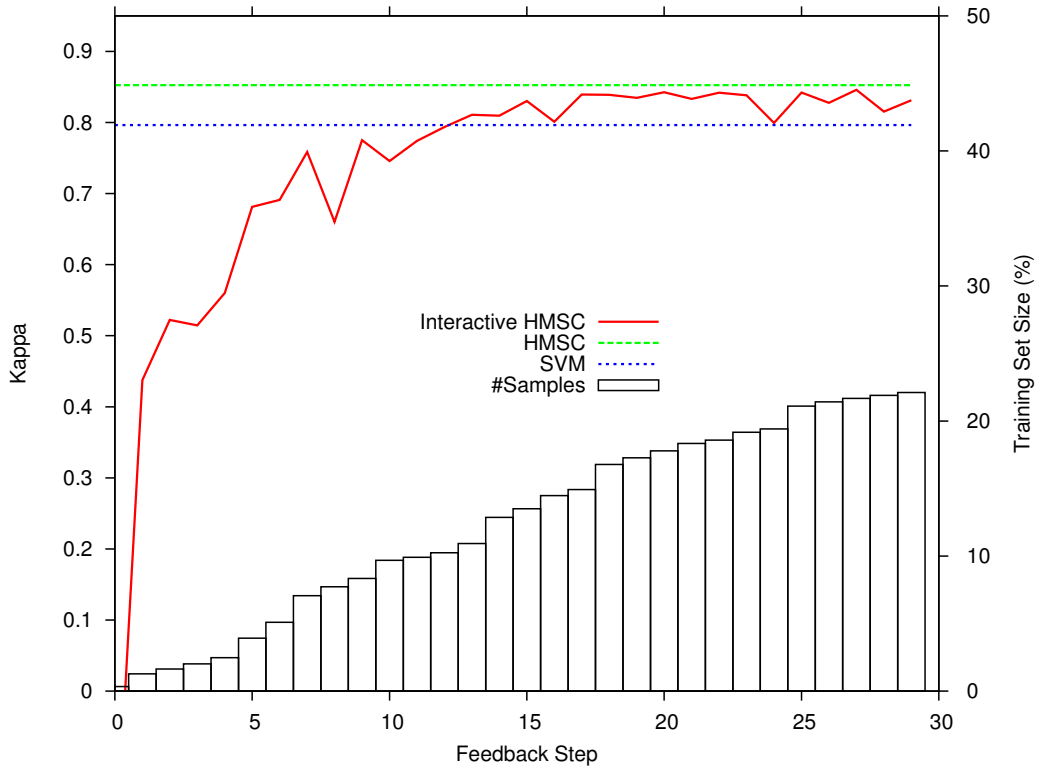


Figure 6.12: Kappa index for the HMSC and SVM and Kappa $\times$ Feedback Steps curves for interactive HMSC using the URBAN dataset. The histogram represents the percentage of the training set used in the interactive method.

training set, the interactive method can achieve the same results as the HMSC trained with the whole set.

Figure 6.12 presents the classification results for the baselines and the Kappa $\times$ Feedback Steps curves using the URBAN dataset. Figure 6.13 presents the same classification results using the URBAN dataset, considering the Overall Accuracy.

In general, the conclusions for the URBAN dataset are similar to the results obtained for the COFFEE dataset. With few iterations, the IHMSC achieves classification results as good as the SVM baseline. With some more feedback steps, by using 15% of the training set, the interactive approach achieved almost the same accuracy (88%) obtained by the HMSC supervised approach.

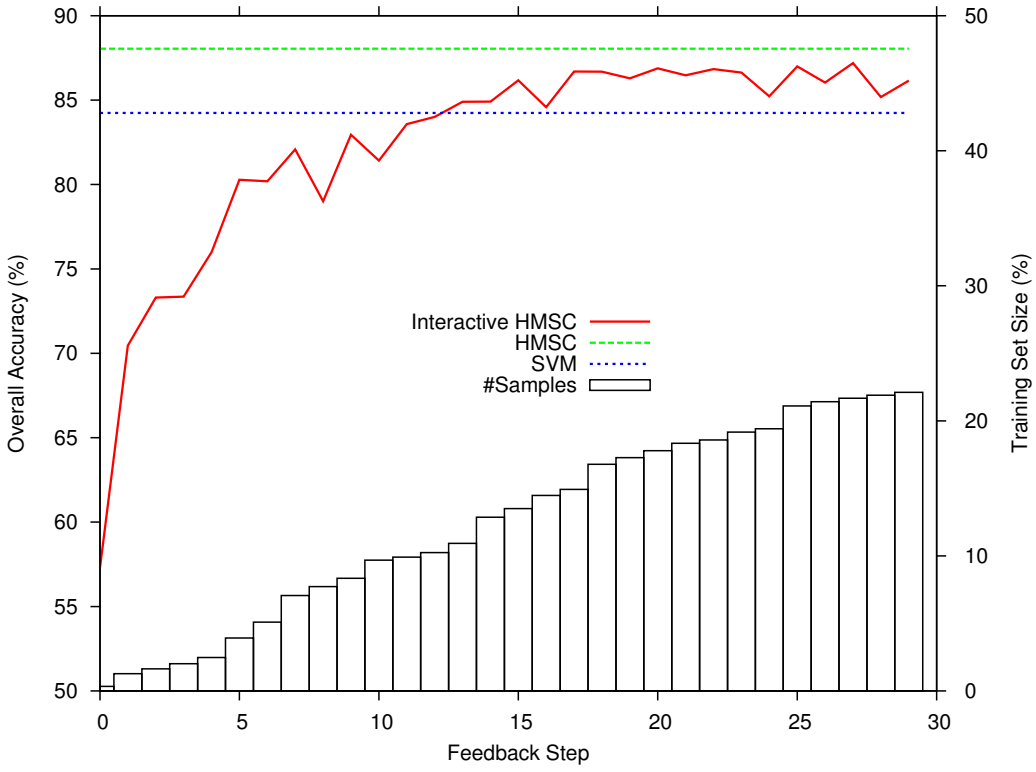


Figure 6.13: Overal Accuracy results for the HMSC and SVM and Overal Accuracy $\times$ Feedback Steps curves for interactive HMSC using the URBAN dataset. The histogram represents the percentage of the training set used in the interactive method.

6.4 Conclusions

We have shown in this chapter that interactive classification based on active learning can be a good alternative to the selection of a suitable training set for high resolution remote sensing analysis. We proposed a method for interactive classification of remote sensing images considering multiscale segmentation: the interactive HMSC (Hierarchical Multiscale Classifier). The objective is to improve the selection of training samples by using the features from the most appropriate scales of representation.

The experiments showed that the combination of scales produce better results than isolated scales in a relevance feedback process. The interactive HMSC achieves more than 80% of accuracy with 10 iterations in both used datasets, overcoming the baseline based on SVM. By using a little bit more than 5% of the training set for the COFFEE dataset and 10% for the URBAN dataset, the interactive method can achieve the same results as the supervised HMSC trained with the whole set.

Chapter 7

Hierarchical Feature Propagation

7.1 Introduction

A suitable segmentation scale relies on the semantics and its association with the studied targets. Figure 7.1 illustrates an example that simulates an image obtained from a forest region. In a fine scale, the segmented objects would allow the analysis based on features extracted from *leaves*. In an intermediary scale level, we could identify different kinds of *trees*. In coarse scales, the segmented objects may represent groups of trees or even complete *forests*.

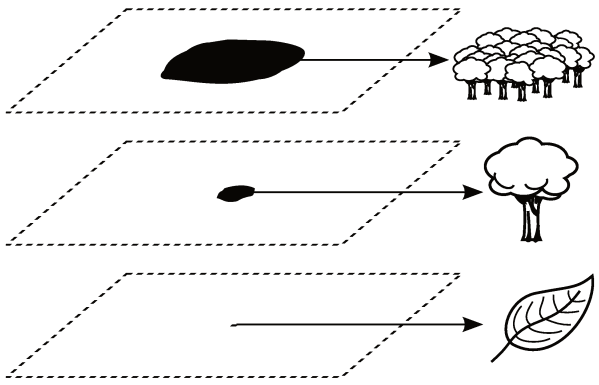


Figure 7.1: An example of different target objects at different scales.

To address the problem of scale selection, several approaches based on multiscale analysis for RSI applications have been proposed [77, 8, 25, 71, 97, 2, 95]. In these approaches, the feature extraction at various segmentation scales is an essential step. However, depending on the strategy, the extraction can be a very costly process. If we apply the same feature extraction algorithm for all regions of different segmentation scales, for example, the pixels in the image would need to be accessed at least once for each scale.

In this chapter, we propose an approach based on the Bag-of-visual-Word (BoW) model [90] to extract features from hierarchy of segmented regions [42]. Our approach is based on processing only the image pixels in the base of the hierarchy (the finest regions scale). The features are quickly propagated to the upper scales by exploiting the hierarchical association among regions at different scales. The strategy starts by creating a visual dictionary based on low-level features extracted from the pixel level (the base of the hierarchy). The low-level feature space is quantized, creating the visual words, and each region in the base of the hierarchy is described according to that dictionary. The features are then propagated to the other scales. At the end, all regions in the hierarchy are represented by a bag of visual words.

The use of visual dictionaries is very effective for visual recognition [90, 108, 5, 109]. It offers a powerful alternative to the description of objects based only on global [83] or on local descriptors [68]. The main drawback of global descriptors – e.g., color histograms (GCH) – is the lack of precision in the representation, which captures few details about the object of interest. Local descriptors, in turn, normally create a large number of features per image or object, which makes it costly to assess the similarities among objects.

In this scenario, representations based on visual dictionaries provide, at the same time, a more precise representation than global descriptions and a more general and simple representation than pure local descriptions. The increase in precision is the result of employing local descriptors and the increase in generality is the result of vector-quantizing the space of local descriptions. Furthermore, the bag-of-visual-word model solves the problem of multiple feature vectors as only one vector is used to describe each object.

Considering a hierarchical topology of regions, there is a natural logical relationship in the visual properties among regions from different scales. Using the example presented in Figure 7.1, the visual properties of a *leaf* are not only present in the *tree* but also in the entire *forest*. Hence, it is logical to have visual properties from *leaves* present in the feature vectors that describe *trees* and *forests*. By employing a bag-of-visual-word representation, the propagation of such features to other levels of the hierarchy becomes straightforward. The pooling strategies used to pool the local features and generate the bag-of-visual-word representation can be successively applied for each level of the hierarchy. Therefore, the low-level feature extraction needs to be performed only at the finest scale of the hierarchy.

The problem of using a simple scale for object-based classification is the dependence on the quality of the segmentation result. If the segmentation is not appropriate to the objects of study, the final result of classification may be harmed. The multiscale interactive approach, presented in Chapter 6, solves this problem, but the refinement of the classification result depends on the hierarchy created by the Guigues' algorithm (see Section 2.2).

A good solution is an interactive system that allows both the improvement and the modification of the hierarchy according the user interactions. Regions may arise or be extinct from the top scales of the hierarchy in each interactive step. It would require feature extraction in

runtime. That would be intractable if we use many low-level global descriptors. However, the propagation approaches we have proposed in this chapter solve this problem. When the hierarchy is changed, the strategy is to recompute the feature vectors of new regions, starting from the basis to the top of the hierarchy.

The rest of this chapter is organized as follows. Section 7.2 details the approaches for hierarchical feature propagation. Section 7.3 present the experimental results. The conclusions and final remarks of this chapter are given in Section 7.4.

7.2 The Hierarchical Feature Propagation

In this section, we present two approaches for hierarchical feature propagation. The first, called *BoW-propagation*, is based on the Bag-of-Word concept. The other, *H-propagation*, is an adaptation of the *BoW-propagation* to propagate low-level features based on histograms from fine scales to the coarsest ones.

7.2.1 BoW-propagation

This approach exploits the bag-of-word concept to iteratively propagate the features along the hierarchy from the finest regions to the coarsest ones. Figure 7.2 illustrates each step of the proposed approach in an example using three scales.

We used the term *interest points* to indicate the points that are used to extract low-level features at the pixel level. We have chosen dense sampling to ensure the representation of homogeneous regions in the dictionary. By using interest-points detectors, the representation of homogeneous regions is not always possible since it tends to select only points in the most salient regions.

Let P_{λ_x} and P_{λ_y} be partitions obtained from the hierarchy H at the scales λ_x e λ_y, respectively. We consider that $P_{\lambda_x} > P_{\lambda_b}$, i.e, P_{λ_x} is coarser than P_{λ_y}. Let $R \in P_{\lambda_x}$ be a region from the partition P_{λ_x}. We call *subregion* of R the region $\hat{R} \in P_{\lambda_y}$ such that $\hat{R} \subseteq R$.

The set $\Gamma(R)$, which is composed of the subregions of R in the partition P_{λ_y}, is given by:

$$\Gamma(R) = \{\forall \hat{R} \in P_{\lambda_y} | p \in R \cap p \in \hat{R}\} \tag{7.1}$$

where p is a pixel. The set of subregions of R in a finer scale are all the regions $\hat{R}$ that have all pixels inside $\hat{R}$ and inside R.

The principle of *BoW-propagation* is to compute the feature histogram h_R, which describes region R, by combining the histograms of subregions $\Gamma(R)$:

$$h_R = f\{h_{\hat{R}_c} \mid \hat{R}_c \in \Gamma(R)\} \tag{7.2}$$

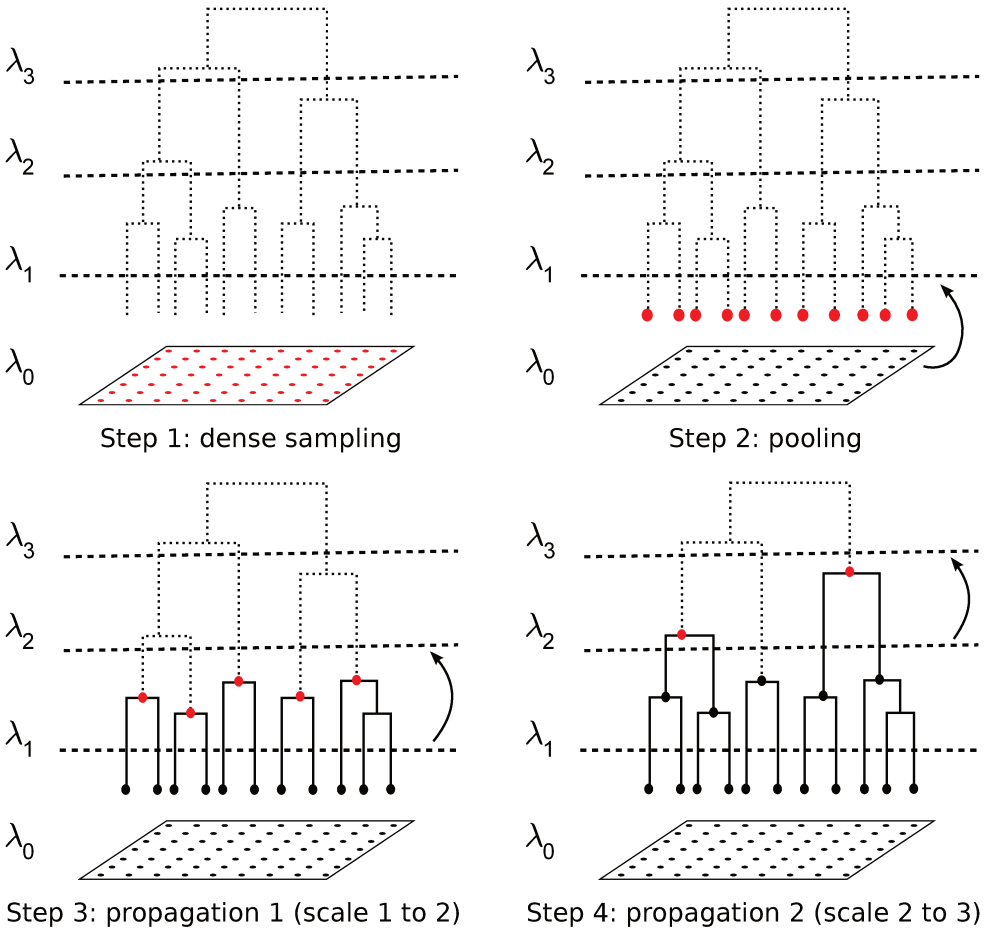


Figure 7.2: The BoW-propagation main steps. The process starts with the dense sampling in the pixel level (scale λ_0). Low-level features are extracted from each interest point. Then, in the second step, a feature histogram is created for each region $R \in P_{\lambda_1}$ by pooling the features from the internal interest points. In the third step, the features are propagated from scale λ_1 to scale λ_2. In the fourth step, the features are propagated from scale λ_2 to the coarsest considered scale (λ_3). To obtain the feature histograms of a given scale, the propagation is performed by considering the histograms of the previous scale.

where f is a combination function.

Algorithm 4 presents the proposed feature extraction and propagation approach. The first step is to extract low-level features from the interest points obtained from a dense sampling schema (line 1). Then, the feature space is quantized, creating a visual dictionary D_k, where k is the dictionary size (line 2). The low-level features are assigned to the visual words (line 3). After this step, each interest point is described by a BoW, which is represented by a histogram. The "first propagation" consists in computing the BoWs h_R of each region $R \in P_{\lambda_1}$ based on the interest points (lines 4 to 6). The "main propagation loop" is responsible for propagating the features to other scales (lines 7 to 10). For all regions R from a partition P_{λ_x}, the BoW h_R is computed based on the $\Gamma(R)$ BoWs, which is described by Equation 7.2 (line 9).

Algorithm 4 BoW-Propagation

```
1  Extract low-level features from the interest points
2  Create the visual dictionary D_k
3  Assign the low-level features to visual words
4  For all R ∈ P_λ1 do
5      Compute the BoW h_R based on the interest points inside R
6  End for
7  For i ← 2 to n do
8      For all R ∈ P_λi do
9          Compute the BoWs h_R based on the Γ(R) BoWs (Equation 7.2)
10     End for
11 End for
```

In the first propagation (lines 4–6), the BoW h_R is obtained by pooling the features from each point inside the region R. The dense sampling scheme shown in Figure 7.3 (a) highlights in red the points considered for pooling. Figure 7.3 (b) shows only the internal points selected and their influence zones. In this example, although we used a circular extraction area for each point, any topology can be used. It is important to clarify that the influence zones outside the region have a very few impact in the final BoW since the radius of the circumference is very small. Anyway, the external influence zone can also be exploited depending on the application.

Figure 7.4 illustrates a schema to represent a segmented region by using dense sampling through a bag of words. The low-level features extracted from the internal points are assigned to visual words and combined by a pooling function.

In the loop defined in lines 7–10, the BoW h_R is computed by combining the BoWs of the subregions $\Gamma(R)$, which is given by Equation 7.2. The combination function f has the same properties of the pooling function. The idea consists in using the same operator either in the pooling or in the combination steps.

Figure 7.5 illustrates an example by using the combination function f to compute the BoW

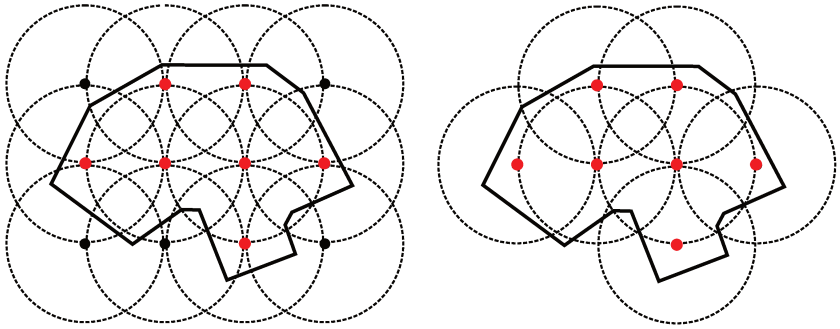


Figure 7.3: Selecting points to describe a region (defined by the bold line). The feature vector that describes the region is obtained by combining the histograms of the points within the defined region. The internal points are indicated in red.

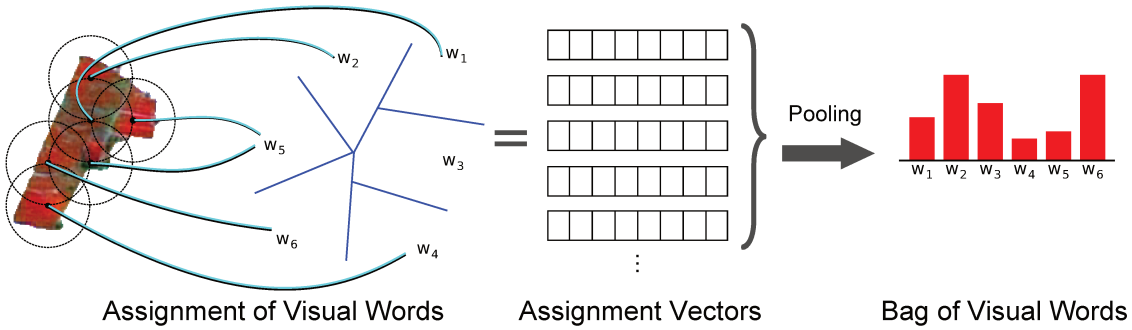


Figure 7.4: Schema to represent a segmented region based on a visual dictionary with dense sampling feature extraction.

h_r of a region r. The region $r \in P_{\lambda_2}$ is composed of the set of subregions $\Gamma(r) = \{a, b, c\}$ at the scale λ_1. Figure 7.5 (a) illustrates, in gray, the region r and its subregions $\Gamma(r)$ in the hierarchy of regions. In Figure 7.5 (b), the BoW h_r is computed based on the function f: $h_r = f(h_a, h_b, h_c)$. Figure 7.6 illustrates the computation of h_r by using the *max* operator as combination function.

The resulting BoW h_r, if we consider the use of *max* pooling, relies on the maximum values of each bin of the BoWs h_a, h_b, and h_c. Considering that each BoW value represents the degree of existence of each visual word in a region, the propagation using the max operator means that the region r is described by the visual words that are in the subregions from the finest scales of the hierarchy.

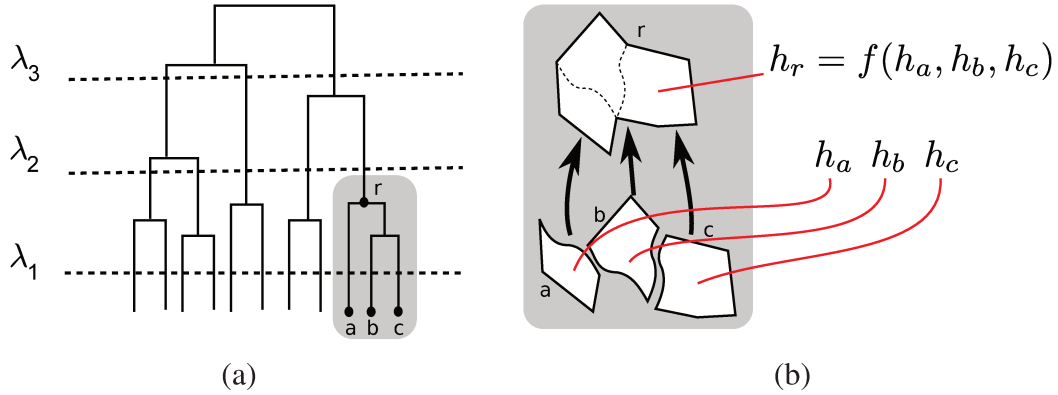


Figure 7.5: Computing the bag h_r of region r by combining the features h_a, h_b, and h_c from the subregions a, b, and c.

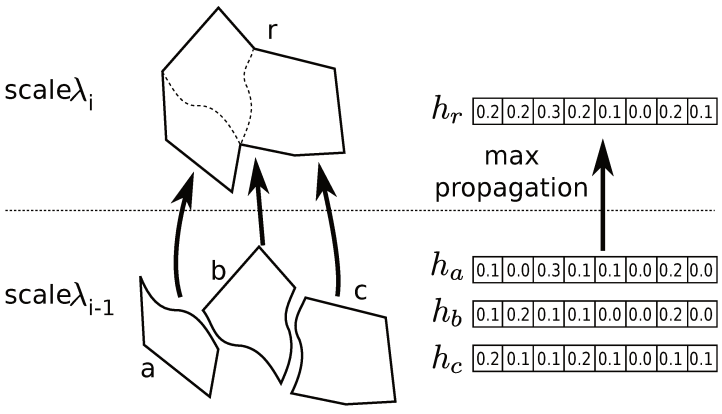


Figure 7.6: Feature propagation example using a *max* pooling operation.

7.2.2 H-propagation

The histogram propagation (H-propagation) consists in estimating the feature histogram representation of a region R, given the low-level histograms extracted from the R subregions $\Gamma(R)$.

Algorithm 5 presents the proposed H-propagation. It works simililarly to the proposed Algorithm 4 for the BoW-propagation.

Algorithm 5 H-Propagation

```
1 Extract low-level feature histograms from the regions in the finest scale λ_1
2 For i ← 2 to n do
3     For all R ∈ P_{λ_i} do
4         Compute the histogram h_R based on the Γ(R) histograms
5     End for
6 End for
```

H-propagation does not quantize the low-level feature space to create a visual dictionary. Another difference, when compared with BoW-propagation, is that H-propagation propagates histogram bins instead of the probabilities of visual words. BoW-propagation is suitable for propagating low-level local features. H-propagation, in turn, is designed only for global descriptors based on histogram representations.

An important issue is the definition of the propagation function f in the case of low-level histograms. Contrarily the propagation of visual words, we use the average function instead of the max function. It is expected that with the average propagation, the quality of the histograms be the same as that performed by the extraction directly from the pixels at all scales of the hierarchy.

7.3 Experiments

In this section, we present the experiments that we performed to validate the proposed approach. We have carried out experiments in order to address the following research questions:

- Are the propagation approaches as effective as the extraction using global descriptors?
- Is the BoW-propagation suitable for both texture and color feature extraction?
- Is it useful to quantize global color descriptors like BIC in a BoW-based model?
- Is it possible to achieve the same accuracy results of global descriptors by propagating features with the H-Propagation approach?

We have used two datasets in our experiments: COFFEE and URBAN. We used linear SVMs to evaluate the classification results.

We designed the experimental protocol to address those questions in the context of texture and color descriptors. In Section 7.3.1, we present the experimental results concerning texture features. In Section 7.3.2, we present the results comparing different strategies to encode color features from a hierarchy of segmented regions.

7.3.1 Texture Description Analysis

SIFT BoW-Propagation: Study of Parameters

In this section, we present an study of parameters for the BoW-Propagation strategy by using the SIFT descriptor in a intermediary scale of segmentation for the COFFEE dataset. Results are shown in Table 7.1.

Table 7.1: Classification results for BoW representation parameters with SIFT descriptor (S=Sampling; DS=Dictionary Size; F=Propagation Function).

S	DS	F	O.A. (%)	Kappa (κ)	Tau (τ)
6	10^2	avg	73.69 ± 2.77	0.25 ± 0.04	0.38 ± 0.04
		max	72.71 ± 2.73	0.22 ± 0.04	0.38 ± 0.03
	10^3	avg	71.24 ± 3.46	0.24 ± 0.06	0.42 ± 0.03
		max	70.80 ± 3.19	0.25 ± 0.05	0.44 ± 0.03
	10^4	avg	73.48 ± 3.00	0.19 ± 0.04	0.30 ± 0.03
		max	73.40 ± 3.48	0.32 ± 0.06	0.48 ± 0.04
4	10^2	avg	72.93 ± 2.82	0.22 ± 0.04	0.35 ± 0.04
		max	73.22 ± 2.53	0.21 ± 0.04	0.34 ± 0.04
	10^3	avg	71.32 ± 2.96	0.24 ± 0.05	0.41 ± 0.03
		max	71.68 ± 2.91	0.29 ± 0.05	0.46 ± 0.03
	10^4	avg	$\mathbf{73.74} \pm 2.73$	0.21 ± 0.04	0.32 ± 0.03
		max	72.66 ± 3.74	$\mathbf{0.33} \pm 0.06$	$\mathbf{0.49} \pm 0.04$

We have used a very dense sampling in the experiments, by overlapping circles of radius **4** and **6** pixels [108], as in the remote sensing images the use of some interest regions can be very small. The difference in classification is very small between the two sampling scales, however we have noticed that the number of regions represented in the finest regions scale is larger for the circles of radius 4. This happens because in COFFEE dataset there are very small regions.

The SIFT features extracted from each region in the dense sampled images were used to generate the visual dictionary. We have tested dictionaries of 10^2, 10^3, and 10^4 visual words. If very few differences among feature vectors need to be encoded, a large visual dictionary is

recommended. However, if some small differences in local textures must be ignored, smaller dictionaries can be useful. We have used soft assignment in these experiments ($\sigma = 60$). The results in Table 7.1 show that larger dictionaries are more representative, specially considering Kappa and Tau measures.

We have also evaluated the impact of different pooling/propagation functions. *Average* pooling tends to smooth the final feature vector, because assignments are divided by the number of points in the image. If we have many points in the image strongly assigned to some visual words, this information is going to be kept in the final feature vector. However, if only a few points have large visual words associations, they can become very small in the image feature vector. This effect is good to remove noise, but it can also eliminate rare visual words, which could be important for the image description. Average pooling tends to work badly with very soft assignments and large dictionaries, due to the fact that points may have a low degree of membership to many visual words, and computing their average is going to generate a too soft vector. We can see this phenomenon in the low values of Kappa and Tau measures for the dictionary of 10^4 words in Table 7.1.

Max pooling captures the strongest assignment of each visual word in the image. Therefore, if only one point has a high degree of membership to a visual word, this information will be hold in the image feature vector. Max pooling tends to present better performance for larger dictionaries with softer assignments. In our experiments, max pooling presents better performances with the largest dictionaries.

BoW Propagation vs BoW Padding

A strategy used to extract texture from segmented regions is based on their bounding boxes. It consists in filling the outside area between the region and its box with a pre-defined value to reduce the interference of external pixels in the extracted texture pattern. This process is known as padding [60] and the commonest approach is to assign zero to the external pixels (ZR-Padding).

We perform experiments to investigate the impact of the segmentation in the feature extraction. Table 7.2 presents the results comparing BoW with ZR-Padding and BoW with Propagation for the COFFEE dataset. Table 7.3 presents the results comparing BoW with ZR-Padding and BoW with Propagation for the URBAN dataset.

Table 7.2: Classification results comparing BoW-ZR-Padding and BoW-Propagation for the COFFEE dataset.

Method	**O.A. (%)**	**Kappa (κ)**	**Tau (τ)**
ZR-Padding	64.39 ± 1.78	0.00 ± 0.02	0.27 ± 0.02
Propagation	$\mathbf{72.66} \pm 3.74$	$\mathbf{0.33} \pm 0.06$	$\mathbf{0.49} \pm 0.04$

Table 7.3: Classification results comparing BoW-ZR-Padding and BoW-Propagation for the URBAN dataset.

Method	**O.A. (%)**	**Kappa (κ)**	**Tau (τ)**
ZR-Padding	48.00 ± 4.18	-0.01 ± 0.04	0.28 ± 0.03
Propagation	$\mathbf{63.55} \pm 2.56$	$\mathbf{0.24} \pm 0.02$	$\mathbf{0.44} \pm 0.01$

As we can observe, the BoW-Propagation strategy yields better results than the ZR-Padding. We can say that in these experiments, the padding strategy caused a loss of 8.37% in the accuracy of the BoW descriptor for the COFFEE dataset. Concering the URBAN dataset, this loss was of 15.55%. Regarding Kappa index, ZR-Padding produces results with no agreement when compared with the ground truth.

SIFT BoW Propagation vs Global Descriptors

Tables 7.4 and 7.5 present the classification results for the BoW-Propagation with SIFT and three successful global texture descriptors (see Chapter 4) for the COFFEE and URBAN datasets, respectively.

Table 7.4: Classification results comparing SIFT BoW-Propagation with the best tested Global descriptors for the COFFEE dataset.

Method	**O.A. (%)**	**Kappa (κ)**	**Tau (τ)**
BoW	$\mathbf{72.66} \pm 3.74$	$\mathbf{0.33} \pm 0.06$	$\mathbf{0.49} \pm 0.04$
QCCH	70.36 ± 2.71	0.14 ± 0.03	0.31 ± 0.02
SID	69.35 ± 2.52	0.01 ± 0.02	0.13 ± 0.03
Unser	69.77 ± 3.11	0.16 ± 0.04	0.34 ± 0.03

Considering the COFFEE dataset, the BoW propagation yields slightly better overall accuracy than global descriptors. The difference is more perceptible regarding the Kappa and Tau indexes. The BoW descriptor achieves 0.3289 of agreement while the best global descriptor (Unser) achieves Kappa index equals to 0.1636. Observing Tau index, BoW yields results almost 50% better than a random classification, while Unser produces classification 34% better than the random.

For the URBAN dataset, the Unser descriptor presents the best results, with Tau index equal to 0.55. BoW propagation yields the second best results, which is more perceptible by observing Tau index (it achieves 0.44).

Table 7.5: Classification results comparing SIFT BoW-Propagation with the best tested Global descriptors for the URBAN dataset.

Method	**O.A. (%)**	**Kappa (κ)**	**Tau (τ)**
BoW	63.55 ± 2.56	0.24 ± 0.02	0.44 ± 0.01
QCCH	50.21 ± 5.15	0.02 ± 0.01	0.06 ± 0.03
SID	63.45 ± 1.46	0.17 ± 0.01	0.39 ± 0.02
Unser	$\mathbf{74.88} \pm 2.92$	$\mathbf{0.44} \pm 0.03$	$\mathbf{0.55} \pm 0.02$

Table 7.6: Classification results for BIC descriptor using BoW-Propagation, Histogram Propagation and, global feature extraction for the COFFEE dataset at segmentation scale λ_3.

Method	**O.A. (%)**	**Kappa (κ)**	**Tau (τ)**
BoW-Propagation	73.41 ± 2.76	0.25 ± 0.03	0.36 ± 0.02
H-Propagation	**79.97** ± 1.76	**0.46** ± 0.02	**0.54** ± 0.02
Global Descriptor	**80.07** ± 1.81	**0.47** ± 0.02	**0.54** ± 0.02

7.3.2 Color/Spectral Description Analysis

In this section, we have tested the proposed approaches for color feature propagation. We have selected BIC descriptor since it produced the best results in the previous results of this thesis that considered segmented regions. We compare the propagation approaches against BIC low-level feature extraction.

BIC BoW-Propagation was computed by using: *max* pooling function, dictionary size of 10^3 words, and soft assignment ($\sigma = 0.1$). We have extracted low-level features from a dense sampling by overlapping squares with 4×4 pixels, as shown in Figure 2.3 (a). BIC H-Propagation, in turn, was computed by using the *avg* pooling function.

Table 7.6 presents the classification by using BIC descritptor with BoW-Propagation, Histogram Propagation, and low-level extraction (Global Descriptor) for the COFFEE dataset.

Concerning the results for the COFFEE dataset, H-Propagation and the Global Descriptor present the same overall accuracy (around 80%). The same can be observed for $kappa$ and tau indexes. BoW-Propagation yields results slightly worse than the other two approaches for the three computed measures.

Table 7.7 shows classification results for the URBAN dataset by using BIC descritptor with BoW-Propagation, Histogram Propagation, and Global Descriptor.

Regarding the URBAN dataset, H-Propagation and Global Descriptor obtained the same overall accuracy, Kappa, and Tau ($\sim 70\%$, 0.31, and 0.47, respectively). The BoW-Propagation approach yields slightly worse results than the other methods concerning overall accuray and Kappa index. The Tau index was the same (0.47).

Table 7.7: Classification results for BIC descriptor using BoW-Propagation, Histogram Propagation and, global feature extraction for the URBAN dataset at segmentation scale λ_3.

Method	**O.A. (%)**	**Kappa** (κ)	**Tau** (τ)
BoW-Propagation	67.03 ± 2.65	0.26 ± 0.03	**0.47** ± 0.02
H-Propagation	**69.86** ± 4.76	**0.31** ± 0.05	**0.47** ± 0.04
Global Descriptor	**69.63** ± 3.33	**0.31** ± 0.04	**0.47** ± 0.03

7.4 Conclusions

The proposed propagation approaches revealed be suitable for saving time on feature extraction from a hierarchy of segmented regions.

Concerning texture, BoW-propagation with SIFT was very promising for encoding features. On the COFFEE dataset, it obtained the best results compared with three global texture descriptors. For the URBAN dataset, the BoW-Propagation with SIFT yields the second best result, lower than the results using Unser descriptor.

Regarding color features, BOW-Propagation seems to be promising, but it requires the setup parameters are better studied. However, H-Propagation shows that it is possible to compute low-level features based only on the hierarchy basis. The features can be propagated without losses in terms of representation quality.

Chapter 8

Conclusions and Future Work

This thesis addresses remote sensing image classification challenges. Many of them are related to the representation scale of the data, and to both the size and the representativeness of used training set.

In this thesis, we have presented contributions in four main research topics that concerns those remote sensing image classification challenges.

In Chapter 4, we presented a comparative study of image descriptors for the classification and recognition of RSI regions. Twelve color descriptors and seven texture descriptors were compared considering effectiveness issues. The effectiveness was measured by precision-recall curves and overall accuracy. JAC and Color Bitmap presented the best results among the color descriptors evaluated, while SID was the best texture descriptor. We also proposed a methodology to evaluate image descriptors in classification problems by using the KNN classifier. It is worth mentioning that there is no work in the literature that applies more descriptors than this study for remote sensing image classification.

The main contributions presented in Chapter 5 are two multiscale classification approaches. The proposed approaches for multiscale image analysis are the Multi-Scale Classifier (MSC) and the Hierarchical Multi-Scale Classifier (HMSC). The MSC is a boosting-based classifier that builds a strong classifier from a set of weak ones. The HMSC is also based on boosting of weak classifiers, but it adopts a sequential strategy of training, according to the hierarchy of scales (from the coarsest to the finest). In this work, we adopted two configurations of weak learners: SVM and RBF. The SVM approach, which yields best results, is based on the SVM classifier with linear kernel. The other one is based on the distances provided by Radial Basis Function. The MSC results show that the combination of scales increase the power of the final classifier. We have also discussed about the correlation among descriptors and the segmentation scales. Experiments show that coarsest scales offer great power of description while the finest ones can improve the classification by detailing the segmentation.

The MSC and HMSC approaches differ from the other studies found in the literature in sev-

eral aspects. First of all, if we consider that there is an ideal scale to represent the objects, we consider the cases in which it is not known and, hence, it can not be defined by empirical parameters. Moreover, even if the optimal scale is known, we can not assure that the use of auxiliary scales does not improve the classification accuracy. Another aspect is that our approach does not propose the fusion of features, but the combination of the classification results at different scales. Finally, our proposal uses different scales to classify the image by assigning the same set of classes at all scales, producing a single final result, i.e, a single model for all classification problems. Our work also differs from others that use a set of classes for each scale and consider semantic information to produce a classification result for each scale.

An interactive approach for interactive multiscale classification of remote sensing images is presented Chapter 6. The strategy, interactive HMSC, improves the selection of training samples by using the features from the most appropriate scales of representation. During a feedback step, for each considered scale, the method selects the regions that are the closest to the separating border. It is also the first interactive method proposed in the literature that consider multiple scales instead of pixel-based information. The experiments showed that the combination of scales produces better results than isolated scales in a relevance feedback process. The interactive HMSC achieves more than 80% of accuracy with 10 iterations in both used datasets, overcoming the baseline based on SVM. By using a little bit more than 5% of the training set for the COFFEE dataset and 10% for the URBAN dataset, the interactive method can achieve the same results as the supervised HMSC trained with the whole set.

Chapter 7 deals with the problem of extracting features from a hierarchy of segmented regions. We have proposed the *BoW-Propagation*, which is a strategy based on the bag-of-visual-word model to propagate features from the finest scales to the coarsest ones in the hierarchy. We have also adapted this strategy to propagate histogram-based low-level features along the hierarchy of segmented regions. This new approach is called *H-Propagation*. These approaches are suitable for saving time on feature extraction from a hierarchy of segmented regions. To the best of our knowledge, these are the first approaches that deal with the propagation of features in a hierarchy of regions. Moreover, experiments using BoW-propagation with SIFT was very promising for encoding texture features. For color features, BOW-Propagation seems to be promising, but it requires many setup parameters. Experiments using H-Propagation show that it is possible to quickly compute low-level features and have a high-quality representation at the same time.

8.1 Future Work

The contributions presented in this thesis focus primarily on solving problems associated with spatial resolution. However, the proposed solutions make us reflect on the treatment of many other kind of datasets that also contain large amount of data and of high dimensionality. There-

fore, in addition to dealing with multiscale classification, future work includes processing of hyperspectral images, multitemporal data, and combination of data from different sensors. The approaches proposed in this thesis can be useful for solving problems with such data. The challenge here is how to extend these approaches for those kinds of data.

Concerning feature extraction, the management of large amount of data from hyperspectral, multitemporal and multi sensors also requires new approaches. Thus, other possible research venues are:

- *Spatio-temporal feature extraction*. This is a topic of great interest not only for the remote sensing community [84], but also in research areas such as Phenology [1]. Some challenges are: how to extract representative features? How to deal with the high dimensionality of the data?

- *Feature extraction from hyperspectral images for object-based classification*. Color descriptors used in this thesis are designed to extract features in three channels. In our experiments, we have selected the most informative bands from our datasets according to the interest targets. However, extracting features from all bands may improve classification results. But, how to adapt those descriptors? How to deal with both spectral and spatial aspects? How to avoid the curse of dimensionality?

- *Combination of features from multiple sensors*. It may involve selection of spectral bands from each sensor. A challenge consists in to adjust and to maintain the georeference among different spatial resolutions.

From the point of view of user interactivity, possible extensions include:

- *Active learning techniques for multiscale classification*. In this thesis, we have selected one region per segmentation scale to require user annotation in each interaction. We question what are the best strategy to make use of the user indications. The use of clustering techniques may be a good option.

- *Visualization and annotation of regions by the user*. The way the user can interact with a multiscale classification system should still be better exploited. We intend to implement an interface as proposed in this thesis and test it with real users. Other ways of annotation should be tested as well (e.g., by means of polygons, rectangle corners).

- *Interactive multiscale classification and segmentation*. The classification method proposed in this thesis considers the use of a hierarchy of coherent regions. In other words, the method depends on the quality of the segmentation. We wonder if the results may be improved by changing the hierarchy structure along the interactions. This would allow not only the multiscale interactive classification, but also interactive multiscale segmentation.

Appendix A

List of Publications

This thesis has generated publications directly and indirectly related to its content.

List of journal papers:

1. *Interactive Multiscale Classification of High-Resolution Remote Sensing* [26], J. A. dos Santos, P-H. Gosselin and S. Philipp-Foliguet, R. da S. Torres, and A. X. Falcão, in the IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, to appear.

2. *A Nature-inspired Framework for Hyperspectral Band Selection* [73], R. Nakamura, L. Fonseca, J. A. dos Santos, R. da S. Torres, X. Yang and J. P. Papa, in the IEEE Transactions on Geoscience and Remote Sensing, to appear.

3. *Multiscale Classification of Remote Sensing Images* [25], J. A. dos Santos, P-H. Gosselin and S. Philipp-Foliguet, R. da S. Torres, and A. X. Falcão, in the IEEE Transactions on Geoscience and Remote Sensing, 2012.

4. *Incorporating Multiple Distance Spaces in Optimum-Path Forest Classification to Improve Feedback-based Learning* [16], A. T. da Silva, J. A. dos Santos, A. X. Falcão, R. da S. Torres, and L. P. Magalhães, in Computer Vision and Image Understanding, 2012.

5. *A Relevance Feedback Method based on Genetic Programming for Classification of Remote Sensing Images* [24], J. A. dos Santos, C. D. Ferreira, R. da S.Torres, M. A. Gonçalves, and R. A. C. Lamparelli, in Information Sciences, 2011.

6. *Relevance feedback based on genetic programming for image retrieval* [35], C. D. Ferreira, J. A. dos Santos, R. da S. Torres, M. A. Gonçalves, R. C. Rezende, and W. Fan in Pattern Recognition Letters, 2011.

List of conference papers:

1. *Remote Sensing Image Representation based on Hierarchical Histogram Propagation* [30], J. A. dos Santos, O. A. B. Penatti, R. da S. Torres, P-H. Gosselin, S. Philipp-Foliguet, and A. X. Falcão, in International Geoscience and Remote Sensing Symposium (IGARSS), 2013, to appear.

2. *Shape-based Time Series Analysis for Remote Phenology Studies* [15], R .da S. Torres, M. Hasegawa, S. Tabbone, J. Almeida, J. A. dos Santos, B. Alberton, P. Morellato, in International Geoscience and Remote Sensing Symposium (IGARSS), 2013, to appear.

3. *Improving Texture Description in Remote Sensing Image Multi-Scale Classification Tasks By Using Visual Words* [29], J. A. dos Santos, O. A. B. Penatti, R. da S. Torres, P-H. Gosselin, S. Philipp-Foliguet, and A. X. Falcão, in International Conference on Pattern Recognition (ICPR), 2012.

4. *Descriptor Correlation Analysis for Remote Sensing Image Multi-Scale Classification*, J. A. dos Santos, F. A. Faria, R .da S. Torres, A. Rocha, P-H. Gosselin, S. Philipp-Foliguet, and A. X. Falcão, in International Conference on Pattern Recognition (ICPR), 2012.

5. *Remote phenology: Applying machine learning to detect phenological patterns in a cerrado savanna* [1], J. Almeida, J. A. dos Santos, B. Alberton, R. da S. Torres, and L. P. C. Morellato, in International Conference on eScience (eScience), 2012.

6. *Automatic Fusion of Region-Based Classifiers For Coffee Crop Recognition* [33], F. A. Faria, J. A. dos Santos, R. da S. Torres, A. R. Rocha, A. X. Falcão, in International Geoscience and Remote Sensing Symposium (IGARSS), 2012.

7. *Hyperspectral Band Selection Through Optimum-Path Forest And Evolutionary-Based Algorithms* [74], R. Nakamura, J. P. Papa, L. Fonseca, J. A. dos Santos, R. da S. Torres, in International Geoscience and Remote Sensing Symposium (IGARSS), 2012.

8. *Automatic Classifier Fusion for Produce Recognition* [32], F. Faria, J. A. dos Santos, R. da S. Torres, and A. Rocha, in Conference on Graphics, Patterns and Images (SIBGRAPI), 2012.

9. *Sinimbu - Multimodal queries to support biodiversity studies.* [19], G. Fedel, C. M. B. Medeiros, J. A. dos Santos, in ICCSA 2012 - Computational Science and Its Applications, p. 620–634, Salvador, Brazil, 2012.

10. *Interactive Classification of Remote Sensing Images by Using Optimum-Path Forest and Genetic Programming* [21], J. A. dos Santos, A. T. da Silva, R. da S. Torres, A. X. Falcão, L. P. Magalhães, R. A. C. Lamparelli, in Computer Vision, Image Analysis and Processing (CAIP), in 2011.

11. *Evaluating the Potential of Texture and Color Descriptors for Remote Sensing Image Retrieval and Classification* [28], J. A. dos Santos, O. A. B Penatti, and R. da S. Torres, in International Conference on Computer Vision Theory and Applications (VISAPP), 2010.

12. *Annotating data to support decision-making: a case study* [66], C. G. N. Macario, J. A. dos Santos, C. M. B. Medeiros, R. da S. Torres, in Workshop on Geographic Information Retrieval (GIR), 2010.

13. *A Genetic Programming Approach for Coffee Crop Recognition* [22], J. A. dos Santos, F. A. Faria, R. T. Calumby, R. da S. Torres, and R. A. C. Lamparelli, in International Geoscience and Remote Sensing Symposium (IGARSS), 2010.

Bibliography

[1] J. Almeida, J. A. dos Santos, B. Alberton, R. da S. Torres, and L. P. C. Morellato. Remote phenology: Applying machine learning to detect phenological patterns in a cerrado savanna. In *8th IEEE International Conference on eScience (eScience 2012)*, Chicago, USA, October 2012.

[2] A. Alonso-González, S. Valero, J. Chanussot, C. López-Martínez, and P. Salembier. Processing multidimensional sar and hyperspectral images with binary partition tree. *Proceedings of the IEEE*, PP(99):1 –25, 2012.

[3] U. C. Benz, P. Hofmann, G. Willhauck, I. Lingenfelder, and Markus Heynen. Multi-resolution, object-oriented fuzzy analysis of remote sensing data for gis-ready information. *ISPRS Journal of Photogrammetry and Remote Sensing*, 58(3-4):239 – 258, 2004.

[4] T. Blaschke. Object based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(1):2 – 16, 2010.

[5] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. *Conference on Computer Vision and Pattern Recognition*, pages 2559–2566, 2010.

[6] G. Brown and L. Kuncheva. "Good" and "Bad" Diversity in Majority Vote Ensembles. In *Multiple Class. Systems*, volume 5997, chapter 13, pages 124–133. 2010.

[7] I. L. Castillejo-González, F. López-Granados, A. García-Ferrer, J. M. Peña-Barragán, M. Jurado-Expósito, M. S. de la Orden, and M. González-Audicana. Object- and pixel-based analysis for mapping crops and their agro-environmental associated measures using quickbird imagery. *Computers and Electronics in Agriculture*, 68(2):207 – 215, 2009.

[8] J. Chen, D. Pan, and Z. Mao. Image-object detectable in multiscale analysis on high-resolution remotely sensed imagery. *International Journal of Remote Sensing*, 30(14):3585–3602, 2009.

[9] L. Chen, W. Yang, K. Xu, and T. Xu. Evaluation of local features for scene classification using vhr satellite images. In *Joint Urban Remote Sensing Event*, pages 385 –388, 2011.

[10] Y. Chen, W. Su, J. Li, and Z. Sun. Hierarchical object oriented classification using very high resolution imagery and lidar data over urban areas. *Advances in Space Research*, 43(7):1101 – 1110, 2009.

[11] G. Câmara, R.Cartaxo, M. Souza, U. M. Freitas, J. Garrido, and F. M. Ii. Spring: Integrating remote sensing and GIS by object oriented data modeling. *Computers and Graphics*, 20(3):395–403, 1996.

[12] J. Cohen. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37, 1960.

[13] R.G. Congalton. A review of assessing the accuracy of classifications of remotely sensed data. *Remote Sensing of Environment*, 37(1):35–46, 1991.

[14] R. da S. Torres and A. X. Falcão. Content-Based Image Retrieval: Theory and Applications. *Revista de Informática Teórica e Aplicada*, 13(2):161–185, 2006.

[15] R .da S. Torres, M. Hasegawa, S. Tabbone, J. Almeida, J. A. dos Santos, B. Alberton, and P. Morellato. Shape-based time series analysis for remote phenology studies. In *Geoscience and Remote Sensing Symposium, IEEE International*, Melbourne, Australia, 2013. to appear.

[16] A. T. da Silva, J. A. dos Santos, A. X. Falcão, R. da S. Torres, and L. P. Magalhães. Incorporating multiple distance spaces in optimum-path forest classification to improve feedback-based learning. *Computer Vision and Image Understanding*, 116(4):510–523, 2012.

[17] R. de O. Stehling, M. A. Nascimento, and A. X. Falcão. An adaptive and efficient clustering-based approach for content-based image retrieval in image databases. In *Proceedings of the International Database Engineering & Applications Symposium*, pages 356–365, Washington, DC, USA, 2001.

[18] R. de O. Stehling, M. A. Nascimento, and A. X. Falcão. A compact and efficient image retrieval approach based on border/interior pixel classification. In *CIKM*, pages 102–109, New York, NY, USA, 2002.

[19] Gabriel de S. Fedel, Claudia Bauzer Medeiros, and Jefersson Alex dos Santos. Sinimbu - multimodal queries to support biodiversity studies. In *ICCSA (1)*, pages 620–634, 2012.

[20] B. Demir, C. Persello, and L. Bruzzone. Batch-mode active-learning methods for the interactive classification of remote sensing images. *Geoscience and Remote Sensing, IEEE Transactions on*, 49(3):1014 –1031, march 2011.

[21] J. A. dos Santos, A. T da Silva, R. da S. Torres, A. X. Falcão, L. P. Magalhães, and R. A. C. Lamparelli. Interactive classification of remote sensing images by using optimum-path forest and genetic programming. In *International Conference on Computer Analysis of Imagens and Patterns*, pages 300–307, 2011.

[22] J. A. dos Santos, F. A. Faria, R. T. Calumby, R. da S. Torres, and R. A. C. Lamparelli. A genetic programming approach for coffee crop recognition. In *Geoscience and Remote Sensing Symposium, IEEE International*, pages 3418–3421, Honolulu, USA, July 2010.

[23] J. A. dos Santos, F. A. Faria, R. da S. Torres, A. Rocha, P-H. Gosselin, S. Philipp-Foliguet, and A. X. Falcão. Descriptor correlation analysis for remote sensing image multi-scale classification. In *International Conference on Pattern Recognition*, Tsukuba, Japan, November 2012.

[24] J. A. dos Santos, C. D. Ferreira, R. da S.Torres, M. A. Gonçalves, and R. A. C. Lamparelli. A relevance feedback method based on genetic programming for classification of remote sensing images. *Information Sciences*, 181(13):2671–2684, 2011.

[25] J. A. dos Santos, P.H. Gosselin, S. Philipp-Foliguet, R. da S. Torres, and A. X. Falcão. Multiscale classification of remote sensing images. *Geoscience and Remote Sensing, IEEE Transactions on*, 50(10):3764–3775, 2012.

[26] J. A. dos Santos, P.H. Gosselin, S. Philipp-Foliguet, R. da S. Torres, and A. X. Falcão. Interactive multiscale classification of high-resolution remote sensing images. *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of*, 2013. To appear.

[27] J. A. dos Santos, R. A. C. Lampareli, and R. da S. Torres;. Using relevance feedback for classifying remote sensing images. In *XIV Brazilian Remote Sensing Symposium*, pages 7909–7916, Natal, RN, Brazil, Abril 2009.

[28] J. A. dos Santos, O. A. B. Penatti, and R. da S. Torres. Evaluating the potential of texture and color descriptors for remote sensing image retrieval and classification. In *The International Conference on Computer Vision Theory and Applications*, pages 203–208, Angers, France, May 2010.

[29] J. A. dos Santos, O. A. B. Penatti, R. da S. Torres, P-H. Gosselin, S. Philipp-Foliguet, and A. X. Falcão. Improving texture description in remote sensing image multi-scale classification tasks by using visual words. In *International Conference on Pattern Recognition*, Tsukuba, Japan, November 2012.

[30] J. A. dos Santos, O. A. B. Penatti, R. da S. Torres, P-H. Gosselin, S. Philipp-Foliguet, and A. X. Falcão. Remote sensing image representation based on hierarchical histogram propagation. In *Geoscience and Remote Sensing Symposium, IEEE International*, Melbourne, Australia, 2013. to appear.

[31] S. S. Durbha and R. L. King. Semantics-enabled framework for knowledge discovery from earth observation data archives. *Geoscience and Remote Sensing, IEEE Transactions on*, 43(11):2563 – 2572, nov. 2005.

[32] F. Faria, J. A. dos Santos, R. da S. Torres, and A. Rocha. Automatic classifier fusion for produce recognition. In *SIBGRAPI 2012*, Ouro Preto-MG, Brazil, August 2012.

[33] F. Faria, J. A. dos Santos, R. da S. Torres, A. Rocha, and A. X. Falcão. Automatic fusion of region-based classifiers for coffee crop recognition. In *Geoscience and Remote Sensing Symposium, IEEE International*, Munique, Germany, July 2012.

[34] J. Feng, L. C. Jiao, X. Zhang, and D. Yang. Bag-of-visual-words based on clonal selection algorithm for sar image classification. *Geoscience and Remote Sensing Letters*, 8(4):691 –695, July 2011.

[35] C. D. Ferreira, J. A. dos Santos, R. da S. Torres, M. A. Gonçalves, R. C. Rezende, and W. Fan. Relevance feedback based on genetic programming for image retrieval. *Pattern Recognition Letters*, 32(1):27–37, 2011.

[36] C. Ferri, J. Hernández-Orallo, and R. Modroiu. An experimental comparison of performance measures for classification. *Pattern Recognition Letters*, 30(1):27–38, 2009.

[37] R. Gaetano, G. Scarpa, and G. Poggi. Hierarchical texture-based segmentation of multiresolution remote-sensing images. *Geoscience and Remote Sensing, IEEE Transactions on*, 47(7):2129 –2141, july 2009.

[38] X. Gigandet, M.B. Cuadra, A. Pointet, L. Cammoun, R. Caloz, and J.-Ph. Thiran. Region-based satellite image classification: method and validation. In *International Conference on Image Processing*, volume 3, pages III–832–5, September 2005.

[39] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1992.

[40] P-H. Gosselin and M. Cord. Active learning methods for interactive image retrieval. *Image Processing, IEEE Transactions on*, 17(7):1200 –1211, july 2008.

[41] P-H. Gosselin, M. Cord, and S. Philipp-Foliguet. Kernels on bags of fuzzy regions for fast object retrieval. In *International Conference on Image Processing*, volume 1, pages I –177 –I –180, 16 2007-oct. 19 2007.

[42] L. Guigues, J. Cocquerez, and H. Le Men. Scale-sets image analysis. *International Journal of Computer Vision*, 68:289–317, 2006.

[43] T. Hermosilla, L.A. Ruiz, J. A. Recio, and M. Cambra-López. Assessing contextual descriptive features for plot-based classification of urban areas. *Landscape and Urban Planning*, 106(1):124 – 137, 2012.

[44] C. Huang and Q. Liu. An orientation independent texture descriptor for image retrieval. *International Conference on Communications, Circuits and Systems*, pages 772–776, July 2007.

[45] J. Huang, S. R. Kumar, M. Mitra, W. Zhu, and R. Zabih. Image indexing using color correlograms. In *Conference on Computer Vision and Pattern Recognition*, page 762, Washington, DC, USA, 1997.

[46] Z. Huaxin, B. Xiao, and Z. Huijie. A novel approach for satellite image classification using local self-similarity. In *Geoscience and Remote Sensing Symposium, IEEE International*, pages 2888 –2891, 2011.

[47] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *International Conference on Computer Vision*, pages 604–610, Washington, DC, USA, 2005.

[48] A. Katartzis, I. Vanhamel, and H. Sahli. A hierarchical markovian model for multiscale region-based classification of vector-valued images. *Geoscience and Remote Sensing, IEEE Transactions on*, 43(3):548–558, March 2005.

[49] M. G. Kendall. A New Measure of Rank Correlation. *Biometrika*, 30(1/2):81–93, 1938.

[50] H-J. Kim and J-U. Kim. Combining active learning and boosting for naïve bayes text classifiers. In *Advances in Web-Age Information Management*, volume 3129 of *Lecture Notes in Computer Science*, pages 519–527, 2004.

[51] M. Kim, T. A. Warner, M. Madden, and D. S. Atkinson. Multi-scale geobia with very high spatial resolution digital aerial imagery: scale, texture and image objects. *International Journal of Remote Sensing*, 32(10):2825–2850, 2011.

[52] V. Kovalev and S. Volmer. Color co-occurence descriptors for querying-by-example. *MultiMedia Modeling*, 0:32–38, 1998.

[53] L. I. Kuncheva and C. J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51(2):181–207, 2003.

[54] C. Kurtz, N. Passat, A. Puissant, and P. Gançarski. Hierarchical segmentation of multiresolution remote sensing images. In *Mathematical Morphology and Its Applications to Image and Signal Processing*, volume 6671 of *Lecture Notes in Computer Science*, pages 343–354. Springer Berlin / Heidelberg, 2011.

[55] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33:159–174, March 1977.

[56] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y. Ng. Efficient sparse coding algorithms. In *Advances in Neural Information Processing Systems*, pages 801–808, 2007.

[57] J.Y. Lee and T. A. Warner. Image classification with a region based approach in high spatial resolution imagery. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 181–187, Istanbul, Turkey, July 2004.

[58] H. Li, H. Gu, Y. Han, and J. Yang. An efficient multiscale srmmhr (statistical region merging and minimum heterogeneity rule) segmentation method for high-resolution remote sensing imagery. *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of*, 2(2):67 –73, june 2009.

[59] N. Li, H. Huo, and T. Fang. A novel texture-preceded segmentation algorithm for high-resolution imagery. *Geoscience and Remote Sensing, IEEE Transactions on*, 48(7):2818 – 2828, 2010.

[60] Zhengrong Li et al. Evaluation of spectral and texture features for object-based vegetation species classification using support vector machines. In *ISPRS Technical VII Symposium*, pages 122–127, 2010.

[61] C. Liu, P. Frazier, and L. Kumar. Comparative assessment of the measures of thematic classification accuracy. *Remote Sensing of Environment*, 107(4):606–616, 2007.

[62] D. Lu and Q. Weng. A survey of image classification methods and techniques for improving classification performance. *International Journal of Remote Sensing*, 28(5):823–870, 2007.

[63] T. Lu and C. Chang. Color image retrieval technique based on color features and image bitmap. *Information Processing and Management*, 43(2):461–472, 2007.

[64] Y. Lu, Q. Tian, and T. Huan. Interactive boosting for image classification. In *Multimedia Content Analysis and Mining*, volume 4577 of *Lecture Notes in Computer Science*, pages 315–324, 2007.

[65] Z. Ma and R. L. Redmond. Tau coefficients for accuracy assessment of classification of remote sensing data. *Photogrammetric Engineering and Remote Sensing*, 61(4):439–453, 1995.

[66] C. G. N. Macário, J. A. dos Santos, C. M. B. Medeiros, and R. da S. Torres. Annotating data to support decision-making: a case study. In *6th Workshop on Geographic Information Retrieval (GIR'10)*, pages 1–7, Zurich, Switzerland, February 2010.

[67] B. S. Manjunath, J.-R. Ohm, V. V. Vasudevan, and A. Yamada. Color and texture descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):703–715, June 2001.

[68] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[69] G. Mountrakis, J. Im, and C. Ogole. Support vector machines in remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(3):247 – 259, 2011.

[70] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.

[71] J. Munoz-Mari, D. Tuia, and G. Camps-Valls. Semisupervised classification of remote sensing images with active queries. *Geoscience and Remote Sensing, IEEE Transactions on*, 50(10):3751 –3763, oct. 2012.

[72] S. W. Myint, P. Gober, A. Brazel, S. Grossman-Clarke, and Q. Weng. Per-pixel vs. object-based classification of urban land cover extraction using high spatial resolution imagery. *Remote Sensing of Environment*, 115(5):1145–1161, 2011.

[73] R. Nakamura, L. Fonseca, J. A. dos Santos, Yang X.-S. Torres, R.da S., and J. Papa. A nature-inspired framework for hyperspectral band selection. *Geoscience and Remote Sensing, IEEE Transactions on*, 2013. To appear.

[74] R. Nakamura, J. Papa, L. Fonseca, J.A. dos Santos, and R.da S. Torres. Hyperspectral band selection through optimum-path forest and evolutionary-based algorithms. In *Geoscience and Remote Sensing Symposium, IEEE International*, pages 3066–3069, 2012.

[75] T. Novack, H.J.H. Kux, R.Q. Feitosa, and G.A. Costa. Per block urban land use interpretation using optical vhr data and the knowledge-based system interimage. In *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVIII-4/C7*, Ghent, Belgium, 2010.

[76] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[77] Y. O. Ouma, S. S. Josaphat, and R. Tateishi. Multiscale remote sensing data segmentation and post-segmentation change detection based on logical modeling: Theoretical exposition and experimental results for forestland cover change analysis. *Computers and Geosciences*, 34(7):715 – 737, 2008.

[78] J. P. Papa, A. X. Falcão, and C. T. N. Suzuki. Supervised pattern classification based on optimum-path forest. *International Journal of Imaging Systems and Technology*, 19(2):120–131, 2009.

[79] G. Paschos, I. Radev, and N. Prabakar. Image content-based retrieval using chromaticity moments. *Transactions on Knowledge and Data Engineering*, 15(5):1069–1072, 2003.

[80] E. Pasolli, F. Melgani, and Y. Bazi. Support vector machine active learning through significance space construction. *Geoscience and Remote Sensing Letters*, 8(3):431 –435, may 2011.

[81] G. Pass, R. Zabih, and J. Miller. Comparing images using color coherence vectors. In *ACM Multimedia*, pages 65–73, 1996.

[82] O. A. B. Penatti and R. da S. Torres. Eva: an evaluation tool for comparing descriptors in content-based image retrieval tasks. In *International Conference on Multimedia Information Retrieval*, pages 413–416, 2010.

[83] O. A. B. Penatti, E. Valle, and R. da S. Torres. Comparative study of global color and texture descriptors for web image retrieval. *Journal of Visual Communication and Image Representation*, 23(2):359–380, 2012.

[84] F. Petitjean, C. Kurtz, N. Passat, and P. Gançarski. Spatio-temporal reasoning for the classification of satellite image time series. *Pattern Recognition Letters*, 33(13):1805 – 1815, 2012.

[85] S. Rajan, J. Ghosh, and M.M. Crawford. An active learning approach to hyperspectral data classification. *Geoscience and Remote Sensing, IEEE Transactions on*, 46(4):1231 –1242, april 2008.

[86] O. Rozenstein and A. Karnieli. Comparison of methods for land-use classification incorporating remote sensing and gis inputs. *Applied Geography*, 31(2):533 – 544, 2011.

[87] R. E. Schapire. A brief introduction to boosting. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, IJCAI '99, pages 1401–1406, 1999.

[88] V. P. Shah, N. H. Younan, S. S. Durbha, and R. L. King. Feature identification via a combined ICA-wavelet method for image information mining. *Geoscience and Remote Sensing Letters*, 7(1):18 –22, jan. 2010.

[89] R. Showengerdt. *Techniques for Image Processing and Classification in Remote Sensing*. Academic Press, New York, 1983.

[90] J. Sivic and A. Zisserman. Video google: a text retrieval approach to object matching in videos. In *International Conference on Computer Vision*, pages 1470–1477 vol.2, 2003.

[91] D. G. Stavrakoudis, J. B. Theocharis, and G. C. Zalidis. A boosted genetic fuzzy classifier for land cover classification of remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(4):529 – 544, 2011.

[92] M. A. Stricker and M. Orengo. Similarity of color images. In W. Niblack and R. C. Jain, editors, *Proc. SPIE Storage and Retrieval for Image and Video Databases III*, volume 2420, pages 381–392, March 1995.

[93] H. Sun, X. Sun, H. Wang, Y. Li, and X. Li. Automatic target detection in high-resolution remote sensing images using spatial sparse coding bag-of-words model. *Geoscience and Remote Sensing Letters*, 9(1):109 –113, 2012.

[94] M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[95] A. H. Syed, E. Saber, and D. Messinger. Encoding of topological information in multiscale remotely sensed data: Applications to segmentation and object-based image analysis. In *International Conference on Geographic Object-based Image Analysis*, pages 102–107, Rio de Janeiro, Brazil, May 2012.

[96] B. Tao and B. W. Dickinson. Texture recognition and image retrieval using gradient indexing. *Journal of Visual Communication and Image Representation*, 11(3):327 – 342, 2000.

[97] Y. Tarabalka, J.C. Tilton, J.A. Benediktsson, and J. Chanussot. A marker-based approach for the automated selection of a single segmentation from a hierarchical set of image segmentations. *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of*, 5(1):262 –272, feb. 2012.

[98] R. Trias-Sanz, G. Stamon, and J. Louchet. Using colour, texture, and hierarchial segmentation for high-resolution remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, 63(2):156 – 168, 2008.

[99] D. Tuia, E. Pasolli, and W.J. Emery. Using active learning to adapt remote sensing image classifiers. *Remote Sensing of Environment*, 115(9):2232 – 2242, 2011.

[100] D. Tuia, F. Ratle, F. Pacifici, M. F. Kanevski, and W. J. Emery. Active learning methods for remote sensing image classification. *Geoscience and Remote Sensing, IEEE Transactions on*, 48(6):2767, june 2010.

[101] D. Tuia, M. Volpi, L. Copa, M. Kanevski, and J. Munoz-Mari. A survey of active learning algorithms for supervised remote sensing image classification. *Selected Topics in Signal Processing, IEEE Journal of*, 5(3):606 –617, june 2011.

[102] A. Tzotsos and D. Argialas. Support vector machine classification for object-based image analysis. In *Object-Based Image Analysis*, Lecture Notes in Geoinformation and Cartography, pages 663–677. Springer Berlin Heidelberg, 2008.

[103] A. Tzotsos, C. Iosifidis, and D. Argialas. A hybrid texture-based and region-based multiscale image segmentation algorithm. In *Object-Based Image Analysis*, Lecture Notes in Geoinformation and Cartography, pages 221–236. Springer Berlin Heidelberg, 2008.

[104] A. Tzotsos, K. Karantzalos, and D. Argialas. Object-based image analysis through nonlinear scale-space filtering. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(1):2–16, 2011.

[105] M. Unser. Sum and difference histograms for texture classification. *Transactions on Pattern Analysis and Machine Intelligence*, 8(1):118–125, 1986.

[106] A. Utenpattanant, O. Chitsobhuk, and A. Khawne. Color descriptor for image retrieval in wavelet domain. *Eighth International Conference on Advanced Communication Technology*, 1:818–821, 20-22 February 2006.

[107] S. Valero, P. Salembier, and J. Chanussot. New hyperspectral data representation using binary partition tree. In *Geoscience and Remote Sensing Symposium, IEEE International*, pages 80–83, Honolulu, USA, july 2010.

[108] K. van de Sande et al. Evaluating color descriptors for object and scene recognition. *Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, 2010.

[109] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J-M Geusebroek. Visual word ambiguity. *Transactions on Pattern Analysis and Machine Intelligence*, 32:1271–1283, 2010.

[110] V. Viitaniemi et al. Experiments on selection of codebooks for local image feature histograms. In *International Conference on Visual Information Systems: Web-Based Visual Information Search and Management*, pages 126–137, 2008.

[111] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Conference on Computer Vision and Pattern Recognition*, volume 1, page 511, Los Alamitos, CA, USA, 2001.

[112] M. Volpi, D. Tuia, and M. Kanevski. Advanced active sampling for remote sensing image classification. In *Geoscience and Remote Sensing Symposium, IEEE International*, pages 1414 –1417, july 2010.

[113] J. Wang, J. Yang, K. Yu, F. Lu, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *Conference on Computer Vision and Pattern Recognition*, 2010.

[114] W-J. Wang, Z-M. Zhao, and H-Q. Zhu. Object-oriented change detection method based on multi-scale and multi-feature fusion. In *Urban Remote Sensing Event, 2009 Joint*, pages 1 –5, may 2009.

[115] Z. Wang, J. R. Jensen, and J. Im. An automatic region-based image segmentation algorithm for remote sensing applications. *Environment Modeling and Software*, 25:1149–1165, October 2010.

[116] L. Weizman and J. Goldberger. Urban-area segmentation using visual words. *Geoscience and Remote Sensing Letters*, 6(3):388 –392, July 2009.

[117] G. G. Wilkinson. Results and implications of a study of fifteen years of satellite image classification experiments. *Geoscience and Remote Sensing, IEEE Transactions on*, 43:433–440, March 2005.

[118] A. Williams and P. Yoon. Content-based image retrieval using joint correlograms. *Multimedia Tools and Applications*, 34(2):239–248, 2007.

[119] P. Wu, B. S. Manjunath, S. Newsam, and H. D. Shin. A texture descriptor for browsing and similarity retrieval. *Signal Processing: Image Communication*, 16(1-2):33 – 43, 2000.

[120] L. Wang X. Li and E. Sung. Improving adaboost for classification on small training sample sets with active learning. In *Proceedings of Asian Conference on Computer Vision*, ACCV 2004, 2004.

[121] S. Xu, T. Fang, De. Li, and S. Wang. Object classification of aerial images with bag-of-visual words. *Geoscience and Remote Sensing Letters*, 7(2):366 –370, Abril 2010.

[122] W. Yang, De. Dai, B. Triggs, and G-S. Xia. Sar-based terrain classification using weakly supervised hierarchical markov aspect models. *Image Processing, IEEE Transactions on*, 21(9):4232–4243, 2012.

[123] J. Yu, Y. Lu, Y. Xu, N. Sebe, and Q. Tian. Integrating relevance feedback in boosting for content-based image retrieval. In *ICASSP (1)'07*, pages 965–968, 2007.

[124] Q. Yu, P. Gong, N. Clinton, G. Biging, M. Kelly, and D. Schirokauer. Object-based detailed vegetation classification with airborne high spatial resolution remote sensing imagery. *Photogrammetric Engineering and Remote Sensing*, 72(7):799–811, 2006.

[125] J. A. M. Zegarra, N. J. Leite, and R. da S. Torres. Rotation-invariant and scale-invariant steerable pyramid decomposition for texture image retrieval. In *XX Brazilian Symposium on Computer Graphics and Image Processing*, pages 121–128, 2007.

[126] W. Zhou, G. Huang, A. Troy, and M. L. Cadenasso. Object-based land cover classification of shaded areas in high spatial resolution imagery of urban areas: A comparison study. *Remote Sensing of Environment*, 113(8):1769 – 1777, 2009.