

# Contents

<b>Contents</b>	<b>i</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Analysis of forested areas . . . . .	2
1.2 Remote sensing for forested areas . . . . .	5
1.3 Context of the thesis . . . . .	7
1.4 Objectives . . . . .	7
1.5 Strategy . . . . .	12
1.6 Structure of the thesis . . . . .	13
<b>2 State of the art</b>	<b>15</b>
2.1 Stand segmentation . . . . .	18
2.1.1 Stand segmentation using VHR optical images . . . . .	18
2.1.2 Stand segmentation using lidar . . . . .	19
2.1.3 Stand segmentation using VHR optical images and lidar	21
2.1.4 Challenges of stand segmentation . . . . .	22
2.2 Segmentation . . . . .	24
2.2.1 "Traditional" segmentation methods . . . . .	24
2.2.2 Superpixels methods . . . . .	27
2.2.3 Segmentation of point cloud . . . . .	27
2.3 Classification . . . . .	28
2.3.1 Supervised classification . . . . .	28
2.3.2 Random Forest . . . . .	29
2.4 Dimension reduction and feature selection . . . . .	31
2.4.1 Dimension reduction: feature extraction . . . . .	31
2.4.2 Feature selection . . . . .	32
Existing methods . . . . .	32
Optimize the selection . . . . .	34
2.5 Smoothing methods . . . . .	35

2.5.1	Local methods . . . . .	35
2.5.2	Global methods . . . . .	35
<b>3</b>	<b>Method</b>	<b>39</b>
3.1	General flowchart . . . . .	40
3.2	Over-segmentation . . . . .	40
3.2.1	Segmentation of lidar data . . . . .	40
3.2.2	Segmentation of images . . . . .	42
3.3	Feature extraction . . . . .	45
3.3.1	Point-based lidar features. . . . .	46
3.3.2	Pixel-based multispectral features. . . . .	47
3.3.3	Pixel-based lidar features. . . . .	48
3.3.4	Object-based feature map. . . . .	48
3.4	Classification . . . . .	49
3.4.1	Training set design . . . . .	49
3.4.2	Feature selection . . . . .	51
3.5	Smoothing . . . . .	51
3.5.1	Local methods . . . . .	52
Filtering . . . . .	52	
Probabilistic relaxation . . . . .	53	
3.5.2	Global smoothing . . . . .	54
Unary term . . . . .	55	
Prior . . . . .	55	
Energy minimization . . . . .	57	
<b>4</b>	<b>Results</b>	<b>61</b>
4.1	Data . . . . .	62
4.2	Segmentation methods . . . . .	64
4.2.1	Retrieve stands borders . . . . .	66
4.2.2	Add semantic information . . . . .	69
4.3	Results of the method . . . . .	71
4.3.1	Over-segmentation . . . . .	72
4.3.2	Feature selection . . . . .	72
4.3.3	Classification . . . . .	76
4.3.4	Regularization . . . . .	84
Local methods . . . . .	84	

Global methods . . . . .	86
4.3.5 Importance of fusion . . . . .	89
4.4 Co-products of the method . . . . .	91
4.4.1 Semi-automatic update . . . . .	91
4.4.2 Inventory . . . . .	93
<b>5 Conclusions and perspectives</b>	<b>97</b>
<b>A Color code</b>	<b>99</b>
<b>B Publication</b>	<b>103</b>
B.1 Journal articles . . . . .	104
B.2 Peer-reviewed conference papers . . . . .	104
<b>C Results - Confusion matrices and accuracy metrics</b>	<b>107</b>
C.1 Accuracy metrics . . . . .	108
C.1.1 Metrics at the local level . . . . .	109
C.1.2 Metrics at the global level . . . . .	111
C.2 Segmentation methods . . . . .	112
C.2.1 VHR optical images . . . . .	113
C.2.2 nDSM . . . . .	114
C.3 Results of the method . . . . .	115
C.3.1 Over-segmentation . . . . .	115
C.3.2 Classification . . . . .	121
C.3.3 Regularization . . . . .	125
C.3.4 Local methods . . . . .	125
Global methods . . . . .	127
<b>Bibliography</b>	<b>133</b>







**1**

# Introduction

---

1.1	Analysis of forested areas . . . . .	2
1.2	Remote sensing for forested areas . . . . .	5
1.3	Context of the thesis . . . . .	7
1.4	Objectives . . . . .	7
1.5	Strategy . . . . .	12
1.6	Structure of the thesis . . . . .	13

---

## 1.1 Analysis of forested areas

Forests are a core component of planet's life. They are defined as large area dominated by trees. Hundreds of other definitions of forest may be used all over the world, incorporating factors such as tree density, tree height, land use, legal standing and ecological function (Schuck et al., 2002; Achard, 2009).

Forest are commonly defined as land with tree crown cover (or equivalent stocking level) of more than 10 percent and area of more than 0.5 hectares (ha). The trees should be able to reach a minimum height of 5 meters at maturity in situ. They may consist either of closed forest formations where trees of various storeys and undergrowth cover a high proportion of the ground; or open forest formations with a continuous vegetation cover in which tree crown cover exceeds 10 percent. Young natural stands and all plantations established for forestry purposes which have yet to reach a crown density of 10 percent or tree height of 5 m are included under forest, as are areas normally forming part of the forest area which are temporarily unstocked as a result of human intervention or natural causes but which are expected to revert to forest.

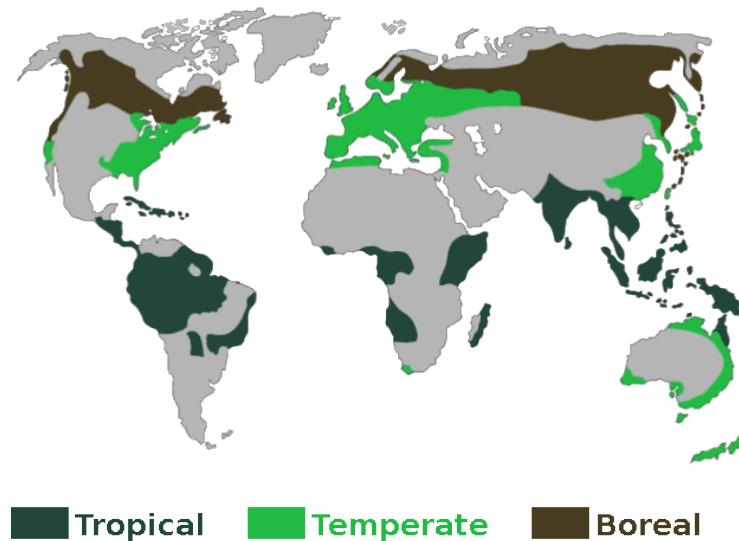


FIGURE 1.1: Forest repartition and categorization in the world.

Forests are the dominant terrestrial ecosystem of Earth, and are distributed across the globe (Pan et al., 2013). They cover about four billion hectares, or approximately 30% of the world's land area (see Figure 1.1). Forests at different latitudes and elevations form distinctly different eco-zones: boreal forests near the poles, tropical forests near the Equator and temperate forests at mid-latitudes (see Figure 1.1). Higher elevation areas tend to support forests similar to those at higher latitudes, and the amount of precipitation also affects forest composition. Since these ecozones are very different, the study at a fine level (e.g. species composition) of forested areas must be restricted to a single ecozone at a time.

Human society and forests influence each other in both positive and negative ways (Vogt et al., 2006). Human activities, including harvesting forest resources, can negatively affect forest ecosystems. Forests have 3 main contributions to human: ecosystem services, tourist attraction and harvesting.

#### **Ecosystem services.**

Forests provide ecosystem services. Indeed, Forests account for 75% of the gross primary productivity of the Earth's biosphere, and contain 80% of the Earth's plant biomass (Pan et al., 2013). They also hold about 90% of terrestrial biodiversity (Brooks et al., 2006; Wasiq et al., 2004). Forests are also beneficial for the environment; they capt and store the CO<sub>2</sub> (Fahey et al., 2010) (see Figure 1.2). About 45% of the total global carbon is held by forests. They also filter dust and microbial pollution of the air (Smith, 2012). Finally, they also play an important role in hydrological regulation and water purification (Lemprière et al., 2008) (see Figure 1.2).

#### **Tourist attraction.**

Forests serve as tourist attractions. In France, there are hundred of long distance footpaths (~ 60000km) through forests. Other activities such as rock climbing, mountain bike or adventure parks are mostly practiced in forests.

#### **Harvesting.**

Wood from trees displays many uses. It has been widely used for fuel (Sterrett, 1994). In this case, hardwood is preferred over softwood because it creates less smoke and burns longer. Wood is still an important construction material (Ramage et al., 2017): Elm was used for the construction of wood boats. In Europe, oak is still the preferred variety for

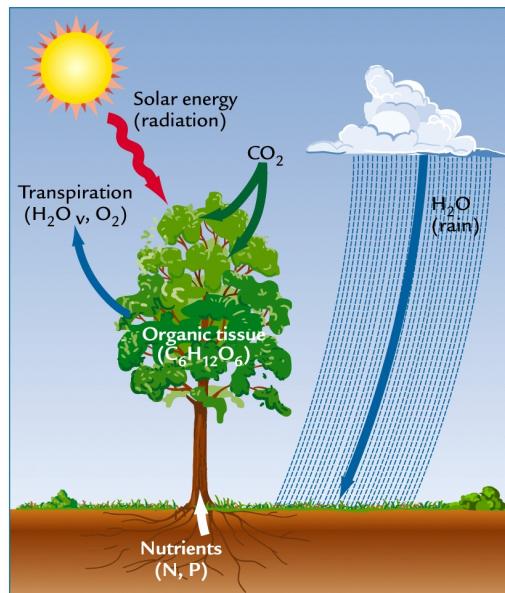


FIGURE 1.2: Carbon cycle: a process of  $CO_2$  storage, water and air purification.

all wood constructions, including beams, walls, doors, and floors. A wider variety of woods is also used such as poplar, small-knotted pine, and Douglas fir. Wood is also needed in the paper industry since wood fibers are an important component of most papers. Eventually, wood is also extensively used for furniture or for making tools or music instruments.

The evolution of forests need to be monitored in order to exploit efficiently the forest resources in a sustainable way (Paris Agreement). For example, France is a great wood importer ( $\sim 25$  millions of  $m^3$  per year), while the french forest is the third in Europe in term of volume. It is therefore needed to better manage and exploit the french wood stocks.

In order to evaluate the forest resources, a precise mapping of forests is needed. Forests are complex structures (Pommerening, 2002), for which information is needed for management, exploitation and more generally for public and private policies. Such information can be the tree species or the tree maturity of the forest. There are two ways to extract such information from forest; **field inventory** or **remote sensing**. The field inventories are very expensive to set up and are also not adapted for a national mapping, but is more adapted for statistics. Remote sensing is a more relevant way in

order to obtain such information since it allows to extract them at a large scale.

In order to meet these needs, two synergistic products could be produced: a statistical inventory or a forest mapping.

## 1.2 Remote sensing for forested areas

The analysis of forested areas from a remote sensing point of view can be performed at three different levels: pixel, object (mainly trees) or stand. In statistical national forest inventory (NFI), an automated and accurate tree segmentation is needed in order to extract tree level features (basal area, dominant tree height, etc., (Means et al., 2000; Kangas et al., 2006)). However, the tree level is not the only reliable level of analysis for forest studies at a national scale but could be employed for a local study. When a joint mapping and statistical reasoning is required (e.g., land-cover (LC) mapping and forest inventory (Tomppo et al., 2008)), forest stands remain the prevailing scale of analysis (Means et al., 2000; White et al., 2016). A stand can be defined in many different ways in terms of homogeneity: tree species, age, height, maturity, and its definition varies according to the countries.

From a remote sensing point of view, the delineation of the stands is a segmentation problem. Forest stands are preferred, since they allow to extract reliable and statistically meaningful features and to provide an input for multi-source statistical inventory. For land-cover mapping, this is highly helpful for forest database updating (Kim et al., 2009), whether the labels of interest are *vegetated areas* (e.g., *deciduous/evergreen/mixed/non-forested*), or, even more precisely, the tree species. To obtain such information, most of the time in national forestry inventory institutes, for reliability purposes, each area is manually interpreted by human operators with very high resolution (VHR) geospatial images focusing on the infra-red channel (Kangas et al., 2006). This work is extremely time consuming and subjective (Wulder et al., 2008b). Furthermore, in many countries, the wide variety of tree species (e.g., >20) significantly complicates the problem.

This is all the more true than photo-interpretation may not always be sufficient and even in case of few species (3-5), automatic classification techniques are not effective enough. The design of an automatic procedure based on remote sensing data would fasten and ease such process. Additionally, the standard manual delineation procedure only takes into account the species, and few characteristics (alternatively height, age, stem density or crown closure). Instead, an automatic method could offer more flexibility being not limited to a visual analysis and using characteristics extracted from complementary data sources and not only CIR ortho-images.

The use of remote sensing data for the automatic analysis of forests has been growing in the last 15 years, especially with the synergistic use of airborne laser scanning (ALS) and optical VHR imagery (multispectral imagery and hyperspectral imagery) (Torabzadeh et al., 2014; White et al., 2016). Several countries have already integrated such sources in their operational pipeline for forest management and characterization. They appear to be both well adapted and complementary inputs for stand segmentation (Dalponte et al., 2012; Dalponte et al., 2015; Lee et al., 2016). Furthermore, they can be employed for forest management (Tokola, 2015; Wulder et al., 2008a; Patenaude et al., 2005). ALS provides a joint direct access to the vertical distribution of the trees and to the ground underneath (Holmgren, 2004). Hyperspectral and multispectral optical images are particularly relevant for tree species classification: spectral and textural information from VHR images can allow a fine discrimination of many species. Multispectral images are often preferred due to their higher availability, and higher spatial resolution. Multispectral images can be acquired from airplanes or satellites. Spaceborne sensors allow to capture large areas with a high temporal rate but suffer from a lower spatial resolution, even if the gap decreases every year (see Table 1.1). For a better spatial resolution, airborne multispectral images are preferred since they allow to extract texture features that are very relevant for tree species classification (Franklin et al., 2000). The airborne linear lidar has been widely used for remote sensing tasks (Lim et al., 2003; Shan et al., 2008; Vosselman et al., 2010). Lidar has been successfully employed for many forest applications (Ferraz et al., 2016a). The new Geiger mode lidar (Ullrich et al., 2016) and

single photon lidar (Viterbini et al., 1987) is also very promising, allowing a significantly higher point density with different angles at a higher altitude and could have an important impact of especially on the studies of forested areas (Jakubowski et al., 2013; Strunk et al., 2012).

Synthetic Aperture Radar (SAR) is widely employed for the evaluation of biomass, especially in forested environment (Le Toan et al., 1992; Beaudoin et al., 1994). With its ability to penetrate the vegetation, SAR in P-band (0.3-1 GHz) allows to estimate efficiently the aboveground biomass. Thus, SAR can be employed in order to extract relevant information of forests but not for their mapping.

### 1.3 Context of the thesis

In France, the study of forests is two fold. They need to be mapped and inventoried. The forest inventory allows to obtain an estimation of the wood stock and the forestation rate at a national scale (see Figures 1.3 & 1.4). Statistics such as volume per hectare, deciduous volume or conifer volume can then be derived. The inventory is performed through field inventory and extrapolated using the forest mapping. Thus, the mapping of forest is very important in order to derive accurate statistics.

Forest mapping is traditionally provided through a national forest LC database (see Figure 1.5). It is manually interpreted by human operators using VHR colored infra-red (CIR) ortho-images. It assigns a vegetation type to each mapped beach of more than 5000 m<sup>2</sup>. The nomenclature is composed of 32 classes based on hierarchical criteria such as pure stands of the main tree species of the French forest. The forest LC should be updated in a 10 years cycle.

### 1.4 Objectives

Currently, the forest land cover (LC) is obtained through remote sensing (namely photo-interpretation). A method should be developed to update it automatically using remote sensing data processing. Since the forest LC is

	<b>SPOT 6,7</b>	<b>Ikonos</b>	<b>Quickbird</b>	<b>Pléiades</b>	<b>RapidEye</b>	<b>Sentinel 2</b>	<b>Landsat 7</b>	<b>Worldview 3</b>
Swath	60 km	11 km	16.5 km	20 km	77 km	290 km	185 km	13.1
Revisit time	2 d	2 d	1-3.5 d	1 d	5.5 d	5 d	16 d	<1 d
Resolution	6 m	4 m	2.44-2.88 m	2.8 m	6.5 m	10 m	30 m	1.24 m
Number of bands	4	4	4	4	5	13	8	28
Main spectral bands (nm)								
Blue	455-520	450-530	450-520	430-550	440-510	460-520	450-520	450-510
Green	530-600	520-610	520-600	500-620	520-590	545-575	520-600	510-580
Red	620-690	640-720	630-690	590-710	630-685	650-680	630-690	630-690
NIR	760-890	760-880	760-900	740-940	760-850	785-900	760-900	770-895

TABLE 1.1: Principal multispectral spatial optical sensors.

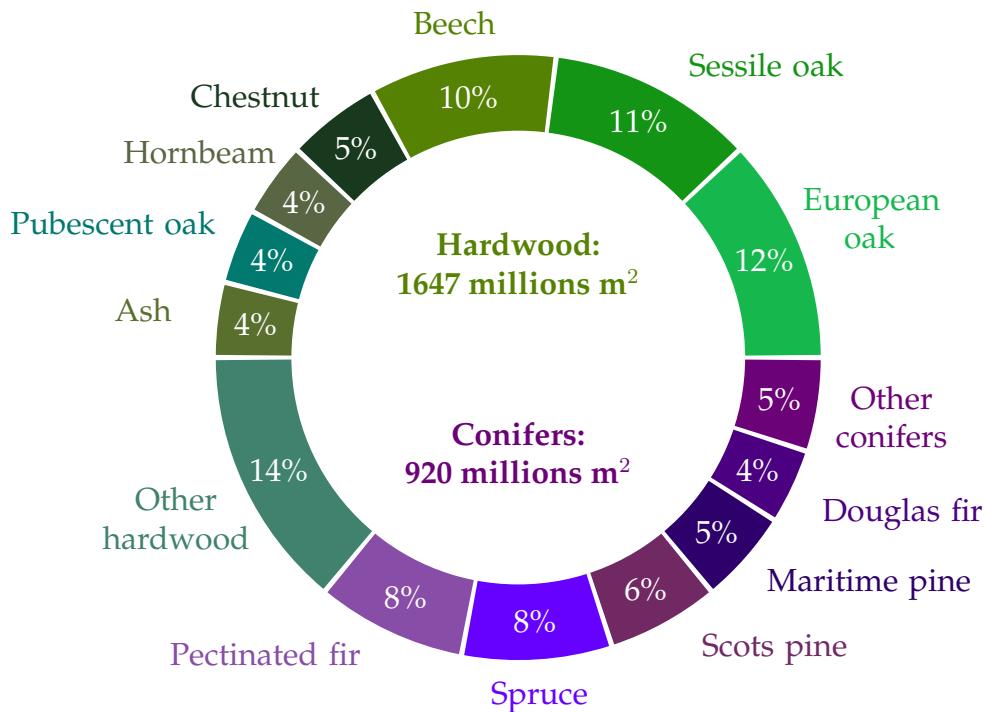


FIGURE 1.3: Distribution wood volume per species.

available, it can be used as an input for subsequent classification (Gressin et al., 2013b). However, the learning process should be carried out carefully (Gressin et al., 2014b). Indeed, some areas might have changed (e.g. forest cuts). Furthermore, the database is designed generalized (Smith et al., 1977). Indeed, forests are not perfectly homogeneous in term of species and there can be many gaps in the canopy, leading to a noisy classification. Thus, such classification would then not be sufficient in order to retrieve homogeneous patches similar to the forest LC. In order to retrieve homogeneous patches, the classification could be regularized using smoothing methods (Schindler, 2012). Furthermore, an automatic method considering more data sources than only CIR ortho-images would allow to enrich the LC, i.e. retrieve homogeneous tree species stands also homogeneous in terms of height (Gressin et al., 2014a).

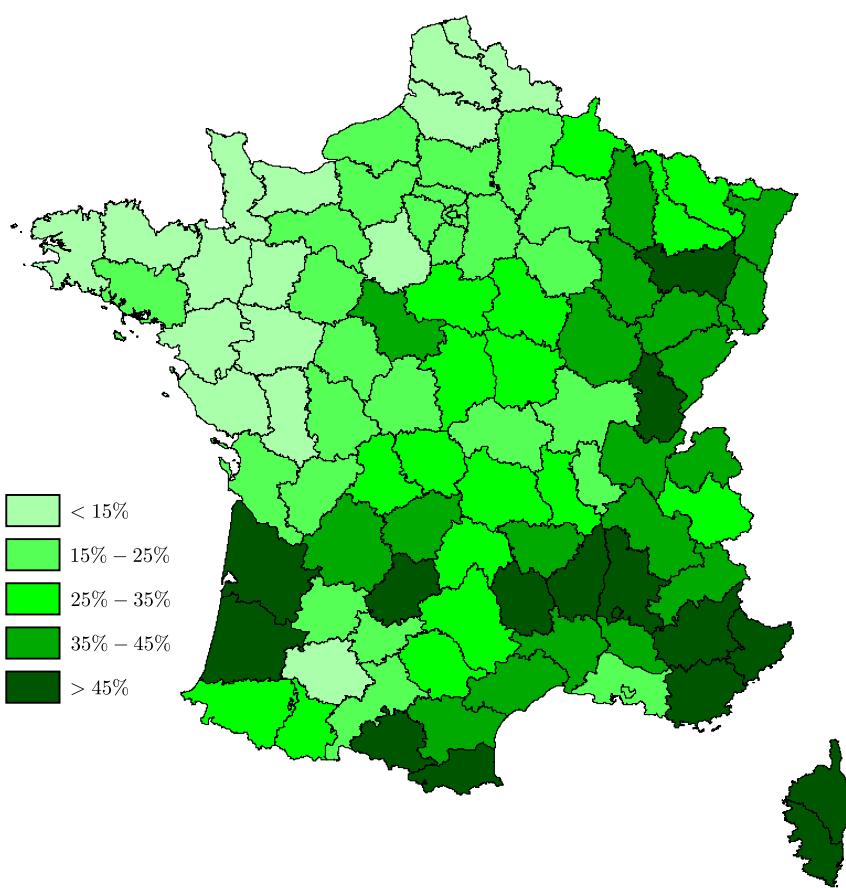


FIGURE 1.4: Forestation rate in France.

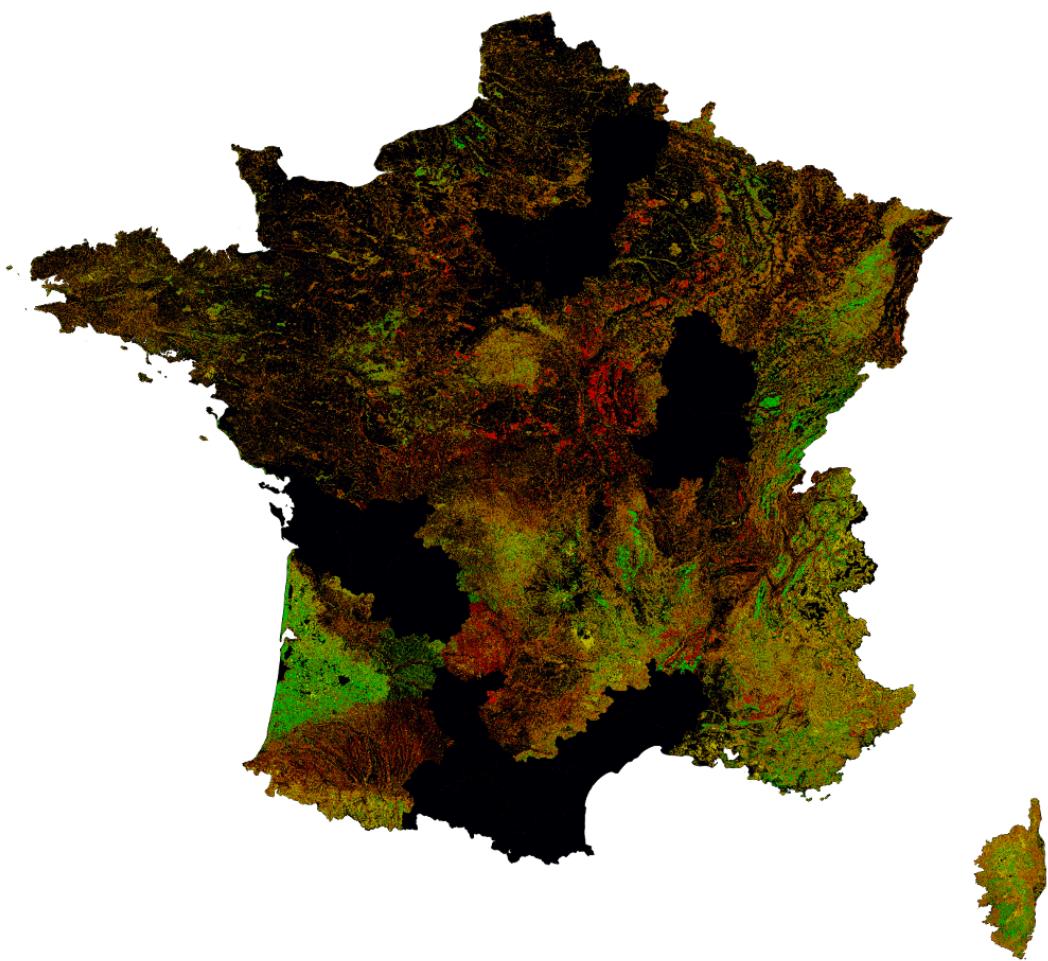


FIGURE 1.5: The French forest LC. Each color is associated to a single species (~20 species in total), black corresponds to non-labeled zones (not operated or non forested).

## 1.5 Strategy

Two remote sensing modalities are available for the mapping of forested areas at IGN; VHR optical images and lidar cloud points. Both are acquired at the national level for various needs of IGN public policies.

### VHR optical images

The VHR images are a part of a national database. In this thesis, the images used have a spatial resolution of 50 cm. The ortho-images employed have 4 bands (red, green, blue near infra-red) captured by the IGN digital cameras (Souchon et al., 2012). Such VHR optical multispectral images are available over full France every 3 years and are one of the component of RGE (a public service mission of the IGN, that aims at describing the national land cover in a precise, complete and homogeneous way).

### Airborne Laser Scanning

IGN also process lots of flights over forested areas with a laser scanning device. The point density for all echoes ranges from 2 to 4 points/m<sup>2</sup>. Forested areas and areas subject to flooding are mainly flown. About 40000 km<sup>2</sup> are acquired each year for Digital Terrain Model generation as main purpose.

The registration between airborne lidar point clouds and VHR multispectral images was performed by IGN itself using ground control points. This is a standard procedure in the French mapping agency since IGN operates both sensors and has also a strong expertise in data georeferencing (this is in fact the national institute responsible for that in France for both airborne and spaceborne sensors).

The combination of these two data is very relevant for the study of forest, indeed, optical images provide the major information about the tree species, while lidar give information about the vertical structure of the forest. Furthermore, the lidar allows to extract consistent object such as trees, that could be used in the stand segmentation process, even if

delineated coarsely.

In order to extract more information from these two modalities, the fusion could be performed at different levels. 3 levels could be defined:

- Low level: It corresponds to the fusion of the observations, in this case, only the reflectance from the optical images and the height of the lidar points.
- Medium level: It corresponds to the fusion of features, derived from both sources, and merged together. It also corresponds to the cooperative understanding of the data; a feature is derived on a modality and applied on the other (e.g. segmentation of the point cloud applied to images).
- High level: It corresponds to decision fusion. One or many classifications have been performed and the final decision is an optimal combination of the classifications and the input data.

## 1.6 Structure of the thesis

- State of the art: Chapter [2](#)
- Method: Chapter [3](#)
- Results: Chapter [4](#)
- Conclusion and perspectives: Chapter [5](#)



**2**

# State of the art

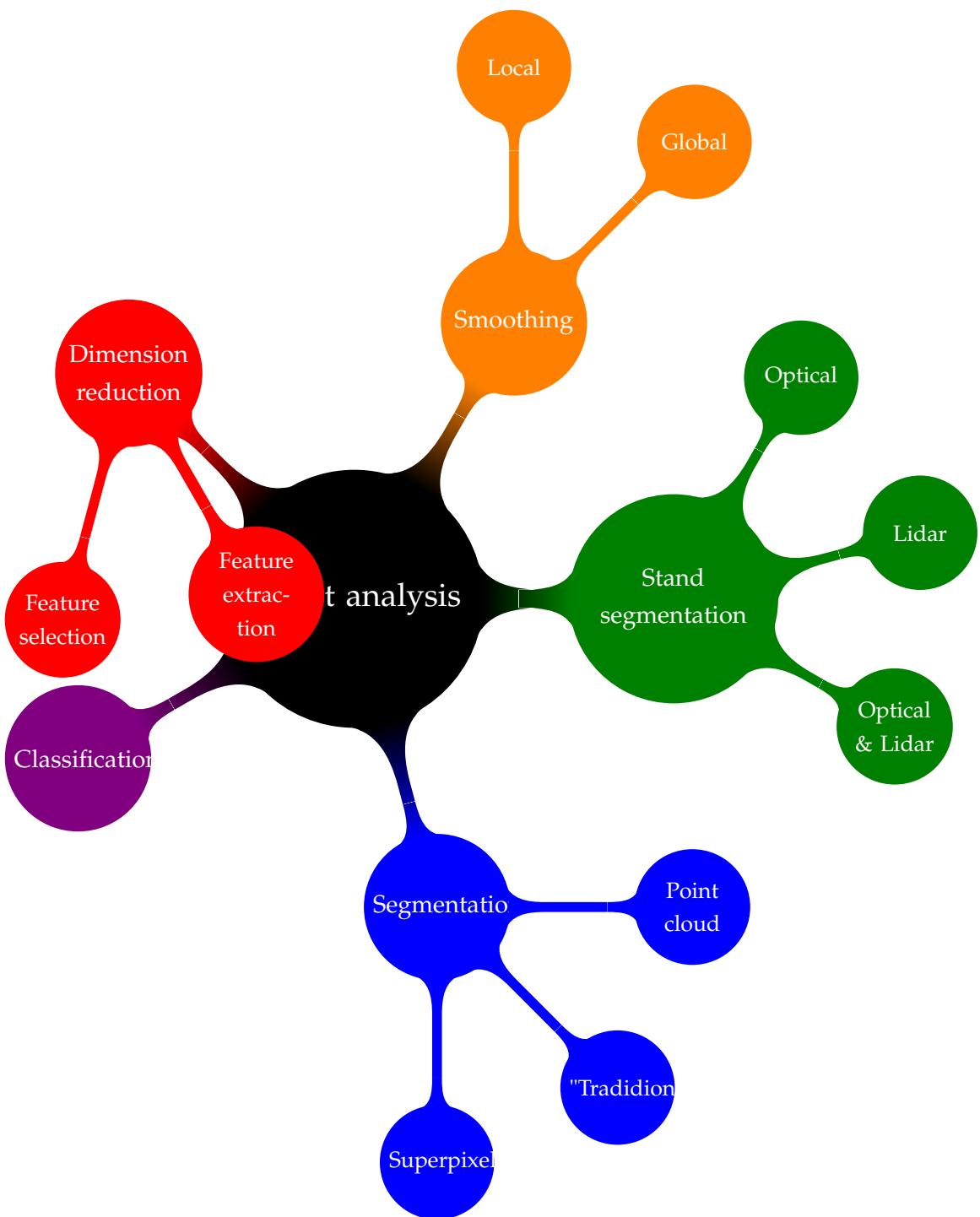
---

2.1	Stand segmentation . . . . .	18
2.1.1	Stand segmentation using VHR optical images . . . . .	18
2.1.2	Stand segmentation using lidar . . . . .	19
2.1.3	Stand segmentation using VHR optical images and lidar	21
2.1.4	Challenges of stand segmentation . . . . .	22
2.2	Segmentation . . . . .	24
2.2.1	"Traditional" segmentation methods . . . . .	24
2.2.2	Superpixels methods . . . . .	27
2.2.3	Segmentation of point cloud . . . . .	27
2.3	Classification . . . . .	28
2.3.1	Supervised classification . . . . .	28
2.3.2	Random Forest . . . . .	29
2.4	Dimension reduction and feature selection . . . . .	31
2.4.1	Dimension reduction: feature extraction . . . . .	31
2.4.2	Feature selection . . . . .	32
	Existing methods . . . . .	32
	Optimize the selection . . . . .	34
2.5	Smoothing methods . . . . .	35
2.5.1	Local methods . . . . .	35
2.5.2	Global methods . . . . .	35

---

Forest are complex areas, the mapping of such environment needs the use of different image processing methods. The extraction of "homogeneous" forest stands is at the interplay between different kinds of image processing methods.

Several methods have been proposed for the forest stand segmentation (section 2.1). They employ different image processing algorithms. Segmentation (section 2.2) algorithms can be employed for a fine or coarse delineation of the principal components of the forests. Classification is also very useful to discriminate the different elements of the forest and detect tree species (section 2.3). Furthermore, with the growing number of feature that could be derived, feature selection algorithms are mandatory in order to improve the results while decreasing the computational load and times (section 2.4). Eventually, smoothing methods could be employed in order to obtain a minimum of an energy. Such energy minimization processes are used for a refinement of raw results (section 2.5).



## 2.1 Stand segmentation

A forest stand is defined as a contiguous group of trees that are uniform in species composition, structure, age and/or height, spatial arrangement, site quality or condition to distinguish it from adjacent other groups of trees.

One should note that the literature remains heavily focused on individual tree extraction and tree species classification (Dalponte et al., 2014; Véga et al., 2014; Kandare et al., 2014), developing site-specific workflows with similar advantages, drawbacks and classification performance. More authors have focused on forest delineation (Eysn et al., 2012; Wang et al., 2012; Radoux et al., 2007), that most of the time do not convey information about the tree species and their spatial distribution. Even if some methods have proposed forest stand delineation, they remain very specific to the study area and provide a binary mask as final output. Consequently, no operational framework embedding the automatic analysis of remote sensing data has been yet proposed in the literature for forest stand segmentation at large scale (Dechesne et al., 2017).

Hence, in the large amount of literature in the field, only few papers focus on the issue of stand segmentation or delineation. They can be categorized with regard to the type of data processed.

### 2.1.1 Stand segmentation using VHR optical images

A stand delineation technique using VHR airborne hyperspectral imagery is proposed in (Leckie et al., 2003). The trees are extracted using a valley following approach and classified into 7 tree species (5 coniferous, 1 deciduous, and 1 non-specified) with a maximum likelihood classifier. A semi-automatic iterative clustering procedure is then introduced to generate the forest polygons.

A hierarchical and multi-scale approach for the identification of stands is adopted in (Hernando et al., 2012). The data inputs were the 4 bands of an airborne 0.5 m orthoimage (Red, Green, Blue, and Near Infra-Red) allowing to derive the Normalized Difference Vegetation Index (NDVI). The stand

mapping solution is based on the Object-Based Image Analysis concept. It is composed of two main phases in a cyclic process: first, segmentation, then classification. The first level consists in over-segmenting the area of interest and performing fine-grained land cover classification. The second level aims to transfer the vegetation type provided by a land cover geodatabase in the stand polygons, already retrieved from another segmentation procedure. The multi-scale analysis appears to have a significant benefit on the stand labeling but it is highly heuristic and requires a correct definition of the stand while we consider it is an interleaved problem.

Following the work of (Wulder et al., 2008b) with IKONOS images, Quickbird-2 panchromatic images are used in (Mora et al., 2010) to automatically delineate forest stands. A standard image segmentation technique is used and the novelty mainly lies on the fact that its initial parameters are optimized with respect to NFI protocols. They show that meaningful stand heights can be derived, which are a critical input for various modeled inventory attributes.

## 2.1.2 Stand segmentation using lidar

A seminal stand mapping method using low density airborne lidar data is proposed in (Koch et al., 2009). It is composed of several steps of feature extraction, creation and raster-based classification. Forest stands are created by grouping neighboring cells within each class. Then, only the stands with a pre-defined minimum size are accepted. Neighboring small areas of different forest types that do not reach the minimum size are merged together to an existing forest stand. The approach offers the advantage of detecting 15 forest types (deciduous/coniferous and maturity) that match very well with the ground truth but to the detriment of simplicity: the flowchart has to be highly reconsidered to fit to other stand specifications. Additionally, the tree species discrimination is not addressed.

The forest stand delineation proposed in (Sullivan et al., 2009) also uses low density airborne lidar still coupling an object-oriented image

segmentation and a supervised classification procedure. Three features are computed and rasterized. The segmentation is performed using a region growing approach. Spatially adjacent pixels are grouped into homogeneous discrete image objects or regions. Then, a supervised discrimination of the segmented image is performed using a Battacharya classifier, in order to determine the maturity of the stands. The tree species are ignored and the procedure requires a careful inspection of the raw data both for feature generation and model training.

The method proposed in (Eysn et al., 2012) aims to generate a forest mask (*forested area* label only) using low density airborne lidar. A Canopy Height Model (CHM) with a spatial resolution of 1 m is derived. The positions and heights of single trees are determined from the CHM using a local maximum filter, based on a moving window approach. Only detected positions with a CHM height superior to 3 m are considered. The crown radii are estimated using an empirical function. The three neighboring trees are connected using a Delaunay triangulation applied to the previously-detected tree position. The crown cover is then calculated using the crown areas of three neighboring trees and the area of their convex hull for each tree triple. The forest mask is derived from the canopy cover values. While this is not a genuine stand delineation method, this approach could be easily extended to a multi-class problem and enlightens the necessity of individual tree extraction even with limited point densities as a basis for the stand-level analysis.

A forest stand delineation also based on airborne lidar data is proposed in (Wu et al., 2014). Three features are first directly extracted from the point cloud. A coarse forest stand delineation is then performed on the feature image using the unsupervised Mean-Shift algorithm, in order to obtain under-segmented raw forest stands. A forest mask is then applied to the segmented image in order to retrieve forest and non-forest raw stands. It may create some small isolated areas, iteratively merged to their most similar neighbor until their size is larger than a user-defined threshold in order to product big raw forest stands. They are then refined into finer level using a seeded region growing based on superpixels. The idea is to select

several different superpixels in a raw forest stand and merge them. This method provides a coarse-to-fine segmentation with relatively large stands. The process was only applied on a small area of a forest in Finland, thus, general conclusions can not be drawn.

### 2.1.3 Stand segmentation using VHR optical images and lidar

The analysis of the lidar and multispectral data is performed at three levels in (Tiede et al., 2004), following a given hierarchical nomenclature of classes in forested environments. The first level represents small objects (single tree scale, individual trees or small groups of trees) that can be differentiated by spectral and structural characteristics using a rule-based classification. The second level corresponds to the stand level. It is built using the same classification process which summarizes forest development phases by referencing to small scale sub-objects at level 1. The third level is generated by merging objects of the same classified forest-development into larger spatial units. The multi-scale analysis offers the advantage of alleviating the standard issue of individual tree crown detection and proposing development stage labels. Nevertheless, the pipeline is highly heuristic, under-exploits lidar data and significant confusions between classes are reported.

The automatic segmentation process of forests in (Diedershagen et al., 2004) is also supplied with Lidar and VHR multispectral images. The idea is to divide the forests into higher and lower sections with lidar. An unsupervised classification process is applied to the two images. The final stand delineation is achieved by segmenting the classification results with pre-defined thresholds. The segmentation results are improved using morphological operators such as opening and closing, which fill the gaps and holes at a specified extent. This method is efficient if the canopy structure is homogeneous and requires a strong knowledge on the area of interest. Since it is based on height information only, it cannot differentiate

two stands of similar height but different species.

In (Leppänen et al., 2008) a stand segmentation technique for a forest composed of *Scots Pine*, *Norway Spruce* and *Hardwood* is defined. A hierarchical segmentation on the Crown Height Model followed by a restricted iterative region growing approach is performed on images composed of rasterized lidar data and Colored Infra-Red images. The process was only applied on a limited area of Finland and prevents from drawing strong conclusions. However, the quantitative analysis carried out by the authors shows that lidar data can help to define statistically meaningful stands (here the criterion was the timber volume) and that multispectral images are inevitable inputs for tree species discrimination.

#### 2.1.4 Challenges of stand segmentation

Table 2.1 summarizes the presented methods of forest stands segmentation. Firstly, it appears that the fusion of the remote sensing modalities (optical images and lidar) improve the results for the problematic of forest stand delineation. However, the stands are not defined the same way in the different proposed methods, preventing from drawing general conclusion.

Regarding the existing state of the art on the forest stand segmentation, it appears that such task is very complex to implement. Indeed, a simple segmentation (without semantic information) is not sufficient since it does not allow to retrieve consistent stands. A classification is mandatory in order to obtain the tree species. However, it is very difficult to discriminate species, since some have a very close looking (e.g. deciduous oak and beech), and the intra-class variability might be important (depending on age, maturity). Eventually, the desired stands are not totally pure, a certain level of generalization is desired in order to have a consistent mapping at large scale. Thus, a regularization process can be employed for such purpose. It also appears that the type of data employed has an impact on the results.

- The VHR optical images permits to obtain information about the tree species, especially when using textural features (Franklin et al., 2000).

Reference	Data processed	Country	Segmentation criteria
Leckie et al., 2003	Hyperspectral images	Canada	Tree species (7)
Hernando et al., 2012	Multispectral images	Spain	Vegetation type
Mora et al., 2010	Panchromatic images	Canada	Height
Koch et al., 2009	Lidar	Germany	Forest types (15)
Sullivan et al., 2009	Lidar	USA	Tree maturity
Eysn et al., 2012	Lidar	Austria	Forest mask
Wu et al., 2014	Lidar	Finland	Tree size, tree density
Tiede et al., 2004	Lidar and multispectral images	Germany	Development phase
Diedershagen et al., 2004	Lidar and multispectral images	Germany	Canopy structure, height
Leppänen et al., 2008	Lidar and multispectral images	Finland	Tree species (3)

TABLE 2.1: Existing methods for forest stand segmentation, see text for more details.

- The lidar data provides information about the vertical structure of the forest that can also be useful for the discrimination of tree species (Brandtberg, 2007). It also brings information about the height that allows to separate forest stands of different ages. Most of the time, lidar is deeply under exploited since it is used only as a simple DSM.

## 2.2 Segmentation

The direct segmentation of optical image and/or lidar point clouds is not sufficient in order to retrieve forest stands. Indeed, such segmentation methods can not take into account the information needed to define the stand. However, with adapted parameters, segmentations algorithms might be useful to obtain relevant segmentation of the data (Dechesne et al., 2017). They can be divided in two categories:

- The "traditional" segmentation methods; in these methods, a specific attention must be paid to the choice of the parameters in order to obtain relevant results. Such segmentation can be applied on an image or a point cloud. Specific methods have also been developed for the segmentation of lidar point cloud (Nguyen et al., 2013).
- The superpixels segmentation methods: they natively produce an over-segmentation of the image. The parameters control the size and the shape of the resulting segments.

### 2.2.1 "Traditional" segmentation methods

The segmentation of an image can be performed using a large variety of techniques (Wilson et al., 1988; Nitzberg et al., 1993; Pal et al., 1993; Zhang, 2006).

The easiest way to segment an image is the thresholding of a gray level histogram of the image (Taxt et al., 1989). When the image is noisy or the background is uneven and illumination is poor, such thresholding is not sufficient. Thus, adaptive thresholding methods have been developed

(Yanowitz et al., 1989).

The watershed transformation is also a simple segmentation method that considers the gradient magnitude of an image as a topographic surface. Pixels having the highest gradient magnitude intensities correspond to watershed lines, which represent the region boundaries. Water placed on any pixel enclosed by a common watershed line flows downhill to a common local intensity minimum. Pixels draining to a common minimum form a catch basin, which represents a segment.

The segmentation can be considered as an unsupervised classification problem. Algorithms dealing with such problems adopt iterative process. The most popular algorithm is the k-means algorithm. Segmentation methods using the spatial interaction models like Markov Random Field (CRF) (Hansen et al., 1982) or Gibbs Random Field (GRF) (Derin et al., 1987). Neural networks are also interesting for image segmentation (Ghosh et al., 1991) as they take into account the contextual information.

The segmentation of an image can also be obtained by the detection of the edges of the image (Peli et al., 1982). The idea is to extract points of significant changes in depth values. Edges are local features and are determined based on local information.

Eventually, hierarchical segmentation algorithms can be employed. They analyze simultaneously the image at several different scales of analysis. Their output is not a single partition, but a hierarchy of regions or data structure that captures different partitions for different scales of analysis (Trias-Sanz, 2006; Guigues et al., 2006; Baatz et al., 2004). These methods allow to control the complexity of the segmentation, which was not the case for the previous methods. The algorithm is a bottom-up approach that starts with an initial over-segmentation (e.g. segmenting almost each pixel on a different own region) and uses this level as a base for the construction of subsequent significant levels. The segmentation process is guided by an energy  $E$  of the form:

$$E = D + \mu C \quad (2.1)$$

where,  $D$  is a fit-to-data measure (how well the segmentation fits to the

original image, better fits give lower values of  $D$ );  $C$  is a measure of segmentation complexity (less complex solutions give lower values of  $C$ ); and  $\mu$  is a dimensional parameter, the scale parameter. The parameter  $\mu$  balances between a perfect fit to the original data ( $\mu = 0$ ), consisting of one segmentation region for each pixel in the original image, and the simplest segmentation, consisting of a single region containing the whole image (Guigues et al., 2006) (see Figure 2.1). The level of segmentation can be adjusted gradually from the finest to the coarsest depending of the image complexity.

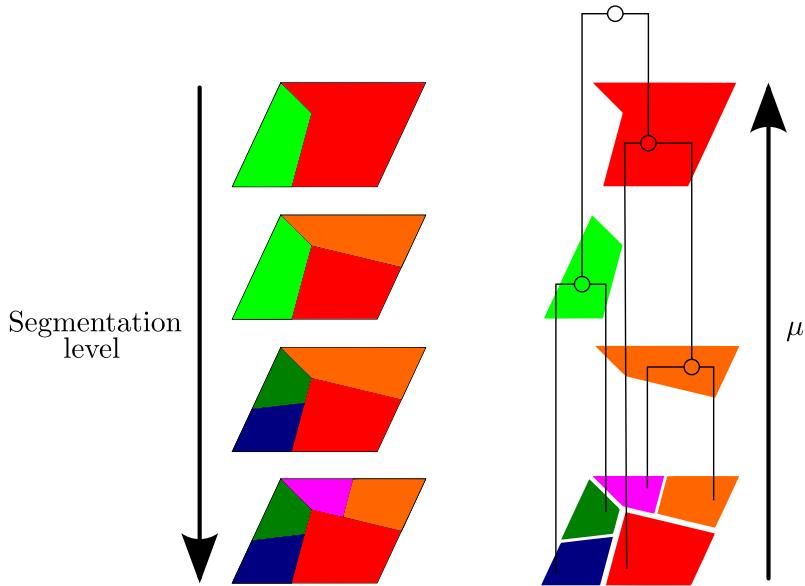


FIGURE 2.1: Graphical depiction of concepts related to hierarchical segmentation. The diagram on the left shows partitions of an image at four different scales  $\mu$ . The partition at the top has the highest  $\mu$  and is therefore the coarsest, the partition at the bottom is the finest.

Top-down approaches can also be employed for image segmentation. In Landrieu et al., 2016 working-set/greedy algorithms to efficiently solve problems penalized respectively by the total variation on a general weighted graph are proposed. The algorithms exploit this structure by recursively splitting the level-sets of a piecewise-constant candidate solution using graph cuts.

### 2.2.2 Superpixels methods

Several superpixels algorithms have been developed (Achanta et al., 2012). They group pixels into perceptually meaningful atomic regions. Many traditional segmentation algorithms have been employed with more or less success to generate superpixels (Shi et al., 2000; Felzenszwalb et al., 2004; Comaniciu et al., 2002; Vedaldi et al., 2008; Vincent et al., 1991). These algorithms produce satisfactory results, however, they may be relatively slow and the number, size and shape of the superpixels might not be specified.

Superpixels algorithms have then been developed. One can control the number of superpixel, their size and their shape. Moore et al., 2008 creates superpixels based on a grid. Optimal path are found using graph cut methods. Veksler et al., 2010 proposes a generation of superpixels based on a global optimization. They are obtained by stitching together overlapping image patches such that each pixel belongs to only one of the overlapping regions. Levinshtein et al., 2009 generate superpixels by a dilatation of a set of seed locations using level-set geometric flow. Resulting superpixels are constrained to have uniform size, compactness, and boundary adherence. Finally, Achanta et al., 2012 proposes a generation of superpixels based on the k-means algorithms. A weighted distance that combines color and spatial proximity is introduced in order to control the size and the compactness of the superpixels.

### 2.2.3 Segmentation of point cloud

The segmentation of point cloud has been highly assessed (Nguyen et al., 2013). The aim is to extract meaningful objects. Such extraction has two principal objectives:

- Objects are detected so as to ease or strengthen subsequent classification task. A precise extraction is not mandatory since the labels would be refined after.
- Objects are precisely delineated in order to derive features from these objects. A high spatial resolution is therefore expected.

In forested areas, the only reliable objects to extract are trees. The first way to extract trees from lidar data is to rasterize the point cloud and use image-based segmentation techniques to obtain trees. Several methods have been developed for single tree delineation (Dalponte et al., 2014; Véga et al., 2014; Kandare et al., 2014).

## 2.3 Classification

A classification is a process that aims to categorize observation. The idea is to assign an observation to one or more classes. This can be done manually or automatically. The classification can be unsupervised, the classes need to be learned and the observation assigned. Such classification is similar to segmentation (see section 2.2). The classification can be supervised, the target classes are known and observations with labels are available.

### 2.3.1 Supervised classification

A great number of supervised classification algorithms have been developed and used for remote sensing issues (Landgrebe, 2005; Lu et al., 2007; Mather et al., 2016). There are two kind of algorithms: the parametric (or generative) and the non-parametric (or discriminative) methods.

The parametric method assume that each class follow a specific distribution (mainly gaussian). The parameters of the distribution are estimated using the learning set. This is the case for the maximum likelihood (Strahler, 1980), maximum a posteriori (Fauvel et al., 2015) or in Trias-Sanz et al., 2005.

The non parametric methods do not make any assumption on the classes distribution. In this category of algorithms, the most popular are the Support Vector Machines (SVM) (Boser et al., 1992; Scholkopf et al., 2001) and the Random Forest (RF) (Breiman, 2001). The artificial neural networks are also efficient algorithms (Hepner et al., 1990; Atkinson et al., 1997). However, despite their great performance in terms of accuracy, they have several drawbacks: firstly, the training process is time consuming and good GPU cards or specific architectures are required in order to reach decent training times (Dean et al., 2012; Moritz et al., 2015). Secondly, it requires an important amount of training data in order to correctly optimize the large number

of parameters (e.g., hundred of millions). Simpler methods exist, such as the k-nearest neighbor (Indyk et al., 1998) or the decision trees (Breiman et al., 1984). The non parametric methods are more efficient for the discrimination of complex classes (Paola et al., 1995; Foody, 2002), and are considered as a basis for land cover classification (Camps-Valls et al., 2009).

We chose to use the RF, which besides their widespread use, since they also offer the possibility of obtaining the probability of belonging of a pixel to a class. This posterior probabilities can be then integrated into a smoothing process. They also report good results, similar to SVM (see Chapter 4). The RF are described in section 2.3.2.

### 2.3.2 Random Forest

The RF have been introduced by Breiman, 2001 and are defined by the aggregation of predictors (decision trees). Here, we refer to the RF with random inputs proposed in Breiman, 2001.

The idea is to create an ensemble of samples  $\mathcal{S}_n^{\Theta_1}, \dots, \mathcal{S}_n^{\Theta_k}$  from an initial training set. A Classification and Regression Tree (CART) (Breiman et al., 1984) is built on each sample  $\mathcal{S}_n^{\Theta_i}$ . Each tree is built using a random pool of  $m$  features among the  $M$  available features. The final classification is obtained by majority vote; each tree votes for a class and the class reaching the most votes wins (see Figure 2.2). This algorithm has two parameters: the number of trees  $k$  and the number of features  $m$  used to build a tree. The first parameter is arbitrary fixed to a high value. The second is generally fixed to the square root of the total number of feature (Gislason et al., 2006).

RF have shown better classification performances than traditional Boosting methods (Breiman, 2001) or SVM (Pal, 2005). They are also able to handle big dataset with large number of feature. Furthermore, a measure of feature importance have been introduced in Breiman, 2001. It allows to qualify the relevance of the feature in the classification process (Strobl et al., 2007).

The importance of a feature  $X_j$ ,  $j \in \{1, \dots, q\}$  (with  $q$  the number of feature) is defined as follow. Let  $\mathcal{S}_n^{\Theta_i}$  be a ensemble of sample and  $OOB_i$  all the observations that does not belong to  $\mathcal{S}_n^{\Theta_i}$ .  $err_{OOB_i}$ , the error on  $OOB_i$  using  $\mathcal{S}_n^{\Theta_i}$ , is then computed. A random permutation on the value of the  $j^{\text{th}}$  feature of  $OOB_i$  is performed in order to obtain  $\widetilde{OOB}_i^j$ .  $\widetilde{err}_{OOB_i^j}$  is then computed.

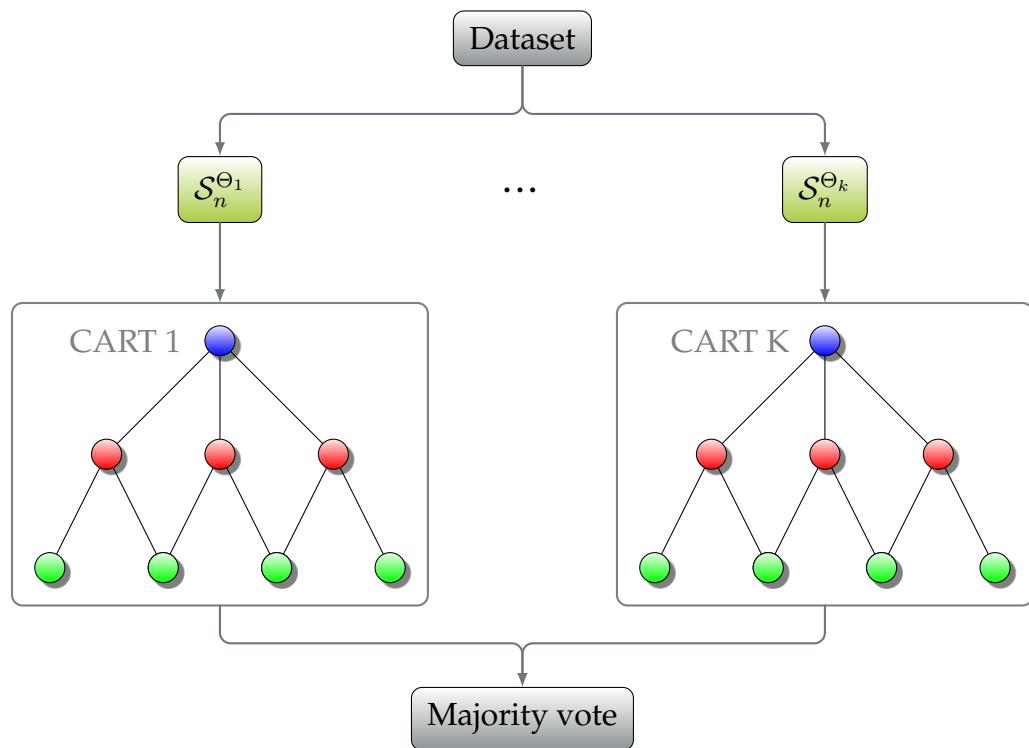


FIGURE 2.2: General diagram of the operation of the Random Forest

The importance of the feature  $j$ ,  $FI(\mathbf{X}_j)$  is the mean of the difference of the errors (see Equation 2.2).

$$FI(\mathbf{X}_j) = \frac{1}{k} \sum_{i=1}^k (\widetilde{\text{err}OOB_i^j} - \text{err}OOB_i) \quad (2.2)$$

where  $k$  is the number of CART.

## 2.4 Dimension reduction and feature selection

It is possible to derive a lot of features from the original data. All the features are used for the classification. The feature selection methods try to overcome the curse of high dimensionality (Bellman, 2015; Hughes, 1968). Indeed, the increasing number of features available tends to decrease the accuracy of the classifiers. Furthermore, the computation times increase with the number of features. Thus, reducing the feature dimension is beneficial for the classification task.

Two kind of approaches exist: first the ones based on the extraction of new features summarizing the information by the transformation of the data, generally using a projection in a space of lower dimensionality. Secondly, feature selection approaches that aim to search for an optimal subset of the features.

### 2.4.1 Dimension reduction: feature extraction

The most popular dimension reduction method is the Principal Component Analysis (PCA). It is an unsupervised method that aim to maximize the variance between data (Jolliffe, 2011). However, it has been demonstrated that PCA is not optimal for the purpose of classification (Cheriyadat et al., 2003). Other methods have been developed based on the PCA: the Independent Component Analysis (ICA) (Jutten et al., 1991) maximizes the statistical independence between data, and the Maximum Autocorrelation Factor (MAF) (Larsen, 2002) maximizes the spatial auto-correlation. When training samples are available, supervised methods exist, such as the linear discriminant analysis (LDA) that tries to maximize both the intra-class homogeneity and the inter-class variance (Fisher, 1936; Lebart et al., 1997).

### 2.4.2 Feature selection

Feature selection aims to search for an optimal subset of features without modifying them. To obtain such subset, one can explore the subsets of features or define a criteria to evaluate the subsets. Furthermore, the selection can be supervised or unsupervised. The first aims to discriminate the better the classes while the second are looking for an optimal subset that contains the most informative and less redundant features. Many exploration methods for feature selection have been proposed in the literature. The naive exhaustive exploration of all the subsets can be envisaged when the number of features is not important.

#### Existing methods

The feature selection methods can be separated into 3 categories: filters, wrapper and embedded. Within the filter methods, one can distinguish the supervised and unsupervised case depending on whether the notion of classes is taken into account or not.

##### • Filters

The filters methods use a feature selection criteria independent from the classifier. They consider the features according to their capacity to bring together elements of the same class and separate the different elements (John, 1997) Thus, these methods compute an individual importance score for each feature, classify the features according to this score and keep only the best. Such scores can be computed using training sample or not. Such methods are independent from a classifier and are used as preliminary step to classification. When training samples are available, separability measures (e.g., Fisher (Fisher, 1936), Bhattacharrya or Jeffires-Matusia) allow to determine whether a feature or a subset of feature is well adapted to discriminate the classes (Bruzzone et al., 2000; Herold et al., 2003; De Backer et al., 2005; Serpico et al., 2007). Statistical measures derived from information theory such as the divergence, the entropy or the mutual information have been proposed in the unsupervised case (Martínez-UsÓMartinez-Uso et al., 2007; Le Moan et al., 2011) or supervised case (Battiti, 1994; Guo et al., 2008; Estévez et al., 2009; Sotoca et al., 2010; Cang et al., 2012). To summarize, criteria for

filter selection methods are numerous and cover different approaches. The supervised ones, which sort features according to an individual importance score and retain only the  $n$  best remain limited since they do not take into account the dependencies between the selected features. Approaches that directly associate relevance scores with feature sets are more interesting. A distinction is made between supervised and unsupervised approaches. The unsupervised criteria are interesting, but present a risk of selecting attributes that would not all also be useful for classification.

### • **Wrapper**

The wrapper methods weight the features according to their pertinence for the prediction (Kohavi et al., 1997). This weighting is related to the performance of a classifier. Estévez et al., 2009; Li et al., 2011; YANG et al., 2007; Zhuo et al., 2008 propose approaches with SVM classifiers. Zhang et al., 2007; Fauvel et al., 2015 use maximum likelihood classifiers. The RF is also employed in Díaz-Uriarte et al., 2006. Data are separated into two subset. The first is used for the training, while the second for the evaluation. The use of a classifier is a big advantage as it fits more to the envisaged problem but can lead to overfitting. However, the use of a classifier significantly increases the computation times. Furthermore, worse results could be obtained when using a feature subset with an other classifier.

### • **Embedded**

Eventually, the embedded methods also involve a classifier and select the features during the training process (Tang et al., 2014). They have two advantages: since they use the data as training, they are robust. Furthermore, the feature selection and the classification are performed together, thus, they are faster than the wrapper methods. Many methods have been proposed. The RF allow to assess the feature importance (Breiman, 2001) and is also natively embedded since the irrelevant features will not be used in the classification process. Other methods are based on the SVM classifiers, the SVM-RFE (Recursive Feature Elimination) (Tuia et al., 2009) recursively removes the less pertinent features according to a weight estimated with a SVM.

## Optimize the selection

The set of possible solutions is generally too large to be visited entirely. Thus, using heuristic rules allows to find a solution close enough to the optimal solution while visiting only a reasonable number of configurations. These optimization methods can generally be distinguished in sequential or incremental methods and stochastic methods.

### • Sequential approaches

The first idea is to add features step by step (forward approaches), also called Sequential Forward Selection (SFS) (Marill et al., 1963). It could also be methods that start from the entire feature set and remove feature step by step (backward approaches), also called Sequential Backward Selection (SBS) (Whitney, 1971). A generalization of these methods have been proposed in Kittler, 1978. Finally, the forward and backward methods could be combined in order to improve the process. The Sequential Floating Forward Selection (SFFS) and the Sequential Floating Backward Selection (SFBS) (Pudil et al., 1994) propose such improvement.

### • Stochastic approaches

Stochastic algorithms will involve hazard in their exploration of the space of solutions. The random initialization and search for a solution can therefore propose different solutions of equivalent quality from a single dataset. The generation of the subset can be totally random (Liu et al., 1997). Genetic algorithms propose a ponderation of the subsets according to their importance (Goldberg, 1989). They allow a faster convergence to a more stable solution. The Particle Swarm Optimization (PSO) algorithm (YANG et al., 2007) is also a fast and select relevant features. For finding an approximate optimal subset of features, simulated annealing (De Backer et al., 2005; Chang et al., 2011).

## 2.5 Smoothing methods

Pixel-wise classification is not sufficient for both accurate and smooth land-cover mapping with VHR remote sensing data. This is particularly true in forested areas: the large intra-class and low inter-class variabilities of classes result in noisy label maps at pixel or tree levels. This is why various regularization solutions can be adopted from the literature (from simple smoothing to probabilistic graphical models).

According to Schindler, 2012, both local and global methods can provide a regularization framework, with their own advantages and drawbacks.

### 2.5.1 Local methods

In local methods, the neighborhood of each element is analyzed by a filtering technique. The labels of the neighboring pixels (or the posterior class probabilities) are combined so as to derive a new label for the central pixel. Majority voting, Gaussian and bilateral filtering can be employed if it is not targeted to smooth class edges. The majority vote can also be used when a segmentation is available: the majority class is assigned to the segment.

The probabilistic relaxation is an other local smoothing method that aims at homogenizing probabilities of a pixel according to its neighboring pixels. The relaxation is an iterative algorithm in which the probability at each pixel is updated at each iteration in order to have it closer to the probabilities of its neighbors (Gong et al., 1989). It reports good accuracies with decent computing time and offers an alternative to edge aware/gradient-based techniques that may not be adapted in semantically unstructured environments.

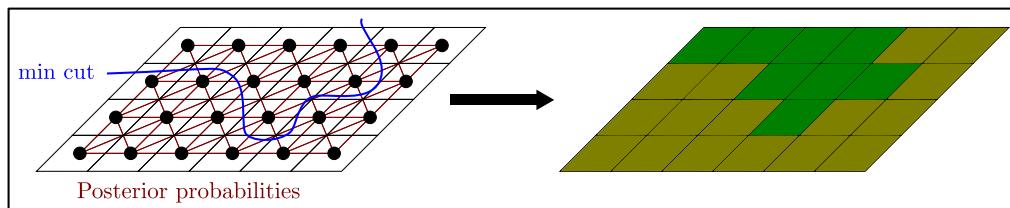
### 2.5.2 Global methods

Global methods consider the full area of interest at the same time. They are based on Markov Random Fields (MRF, see Figure 2.3), the labels at different locations are not considered to be independent. The optimal configuration of labels is retrieved when finding the Maximum A Posteriori over the entire field (Moser et al., 2013). The problem is therefore considered as the minimization procedure of an energy  $E$  over the full image  $I$ . Despite

a simple neighborhood encoding (pairwise relations are often preferred), the optimization procedure propagates over large distances. Depending on the formulation of the energy, the global minimum may be reachable. However, a large range of optimization techniques allow to reach local minima close to the real solution, in particular for random fields with pairwise terms (Kolmogorov et al., 2004). For genuine structured predictions, in the family of graphical probabilistic models, Conditional Random Fields (CRF, see Figure 2.3) have been massively adopted during the last decade. Interactions between neighboring objects, and subsequently the local context can be modeled and learned. In particular, Discriminative Random Fields (DRF, (Kumar et al., 2006)) are CRF defined over 2D regular grids, and both unary/association and binary/interaction potentials are based on labeling procedure outputs. Many techniques extending this concept or focusing on the learning or inference steps have been proposed in the literature (Kohli et al., 2009; Ladický et al., 2012). A very recent trend even consists in jointly considering CRF and deep-learning techniques for the labeling task (Kirillov et al., 2015).

In standard LC classification tasks, global methods are known to provide significantly more accurate results (Schindler, 2012) since contextual knowledge is integrated. This is all the more true for VHR remote sensing data, especially in case of a large number of classes (e.g., 10, (Albert et al., 2016)), but presents two disadvantages. For large datasets, their learning and inference steps are expensive to compute. Furthermore, parameters should often be carefully chosen for optimal performance, and authors that managed to alleviate the latter problem still report a significant computation cost (Lucchi et al., 2011).

### Markovian Random Field



### Conditional Random Field

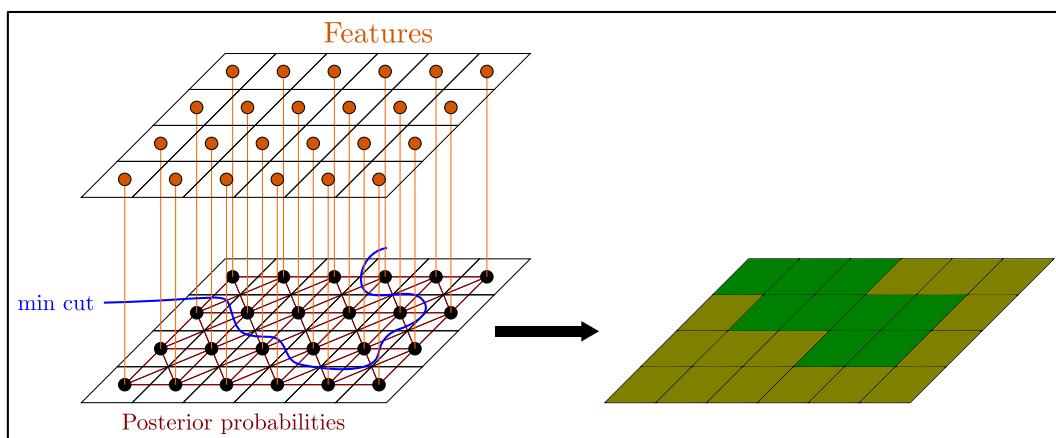


FIGURE 2.3: 8-connected MRF and CRF. The MRF only take into account the posterior probabilities to compute the graph, while CRF also include contextual information (the features).



## 3

# Method

3.1	General flowchart . . . . .	40
3.2	Over-segmentation . . . . .	40
3.2.1	Segmentation of lidar data . . . . .	40
3.2.2	Segmentation of images . . . . .	42
3.3	Feature extraction . . . . .	45
3.3.1	Point-based lidar features. . . . .	46
3.3.2	Pixel-based multispectral features. . . . .	47
3.3.3	Pixel-based lidar features. . . . .	48
3.3.4	Object-based feature map. . . . .	48
3.4	Classification . . . . .	49
3.4.1	Training set design . . . . .	49
3.4.2	Feature selection . . . . .	51
3.5	Smoothing . . . . .	51
3.5.1	Local methods . . . . .	52
Filtering . . . . .	52	
Probabilistic relaxation . . . . .	53	
3.5.2	Global smoothing . . . . .	54
Unary term . . . . .	55	
Prior . . . . .	55	
Energy minimization . . . . .	57	

### 3.1 General flowchart

With respect to the methods mentioned above, it appears that there are no forest stand segmentation method, based on tree species, that can satisfactorily handle a large number of classes ( $>5$ ). The proposed framework is a fully automatic and modular method for species-based forest stand segmentation. The method is composed of four main steps. An over-segmentation is firstly performed in order to retrieve relevant objects. Features are then computed at the pixel level for the optical images and at the point level for the lidar data. The object extracted from the over-segmentation and the computed features allows to derive features at the object level. A classification of the vegetation type (mainly tree species) is then performed at the object level, since it significantly improve the discrimination results. Here, the training set is automatically derived from an existing forest LC database. Specific attention is paid to the extraction of the most relevant training pixels, which is highly challenging with outdated and generalized vector databases. Because of the high number of features, a feature selection is also carried out in order to increase the classification accuracy, reduce the computational load and time, but also to assess the complementarity of the features (multispectral optical images/lidar point cloud). Finally, a regularization is performed in order to retrieve homogeneous forest stands. The flowchart of the method is presented in Figure 3.1.

### 3.2 Over-segmentation

The over-segmentation aims to extract small object that are consistent according to the input data. They are detected so as to ease and strengthen subsequent classification task. A precise extraction is not mandatory since the labels would be refined after.

#### 3.2.1 Segmentation of lidar data

Two approaches could be envisaged: the direct segmentation of the point cloud or the segmentation of a rasterized lidar feature using image-based segmentation algorithms.

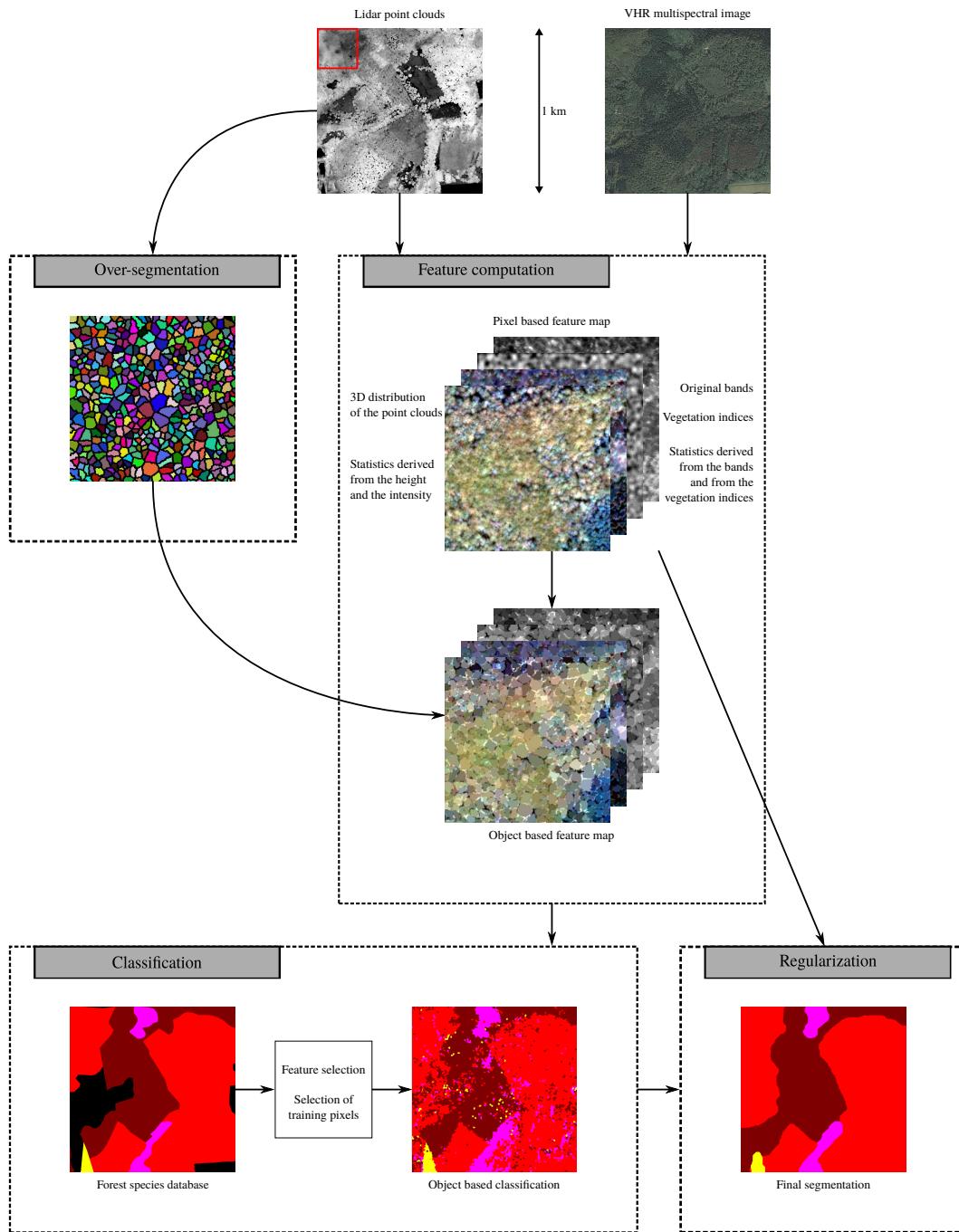


FIGURE 3.1: Flowchart of the proposed method.

The tree extraction from the point cloud is a complex task that has been widely discussed (Dalponte et al., 2014; Véga et al., 2014; Kandare et al., 2014). However, a precise tree extraction is not needed here, since the extracted trees are only needed to improve the classification task. A coarse method is therefore adopted: the tree tops are first extracted from the lidar point clouds using a local maximum filter. A point is considered as a tree top when it has the highest height value within a 5 meter radius. Only the points above 3 meters are retained as it is a common threshold of the literature (Eysn et al., 2012), and appears to be highly discriminative in non-urban areas. Points belonging to a tree are obtained through two criteria. (i) If the height of a point within a 5 m radius is greater or equal than 80% the height of the closest tree top, it is aggregated to the tree top. (ii) If the distance in the  $(x, y)$  plane between an unlabeled point and the closest tree point is smaller than 3 m they are also aggregated. This delineation method allows to discard low vegetation, but buildings might be extracted and considered as trees.

The image-based segmentation are also very efficient for the over-segmentation of lidar data. They are mainly applied on the normalized digital surface model (height). Thus a method using a single band is needed. The watershed algorithm (Vincent et al., 1991) with specific parameters allow to obtain quickly a consistent over-segmentation of the image. A hierarchical segmentation (Guigues et al., 2006) is more adapted since only one parameter that control the segmentation level needs to be provided.

### **3.2.2 Segmentation of images**

Several algorithms have been developed for the over-segmentation of optical RGB (Red-Green-Blue) images. The most common are the superpixels methods (Achanta et al., 2012). Superpixels can be generated using segmentation algorithms (Shi et al., 2000; Felzenszwalb et al., 2004; Comaniciu et al., 2002; Vedaldi et al., 2008; Vincent et al., 1991). A special attention must be paid to the parameters in order to obtain relevant superpixels. These methods produces superpixels that might not be homogeneous in terms of

size and shape but delineate precisely objects. Superpixels algorithms have then been developed, they allow to control the number of superpixel, their size and their shape (Moore et al., 2008; Veksler et al., 2010; Levinshtein et al., 2009; Achanta et al., 2012). For the over-segmentation of the optical images, we will use both "traditional" and superpixels methods.

Five methods have been employed for the segmentation of images (VHR optical images or rasterized lidar features):

- The watershed algorithm (Vincent et al., 1991),
- A segmentation algorithm based on graph-cut (called here PFF) (Felzenszwalb et al., 2004),
- A multilevel hierarchical segmentation algorithms (Guigues et al., 2006),
- A segmentation algorithm based on the mean shift algorithm (called quickshift) (Vedaldi et al., 2008),
- A superpixel algorithm (called SLIC) (Achanta et al., 2012).

### **Watershed.**

Watershed segmentation "classifies" pixels into regions using gradient descent on image features and analysis of weak points along region boundaries. It has been proposed after the observation natural observation of water raining onto a landscape topology and flowing with gravity to collect in low basins. The size of those basins will grow with increasing amounts of precipitation until they spill into one another, causing small basins to merge together into larger basins. Regions (catchment basins) are formed by using local geometric structure to associate points in the image domain with local extrema in some feature measurement such as curvature or gradient magnitude. This technique is less sensitive to user-defined thresholds than classic region-growing methods, and may be better suited for fusing different types of features from different data sets. The watersheds technique is also more flexible in that it does not produce a single image segmentation, but rather a hierarchy of segmentations from which a single region or set of regions can be extracted a-priori.

**PFF.**

This algorithm define a predicate for measuring the evidence for a boundary between two regions using a graph-based representation of the image. An efficient segmentation algorithm based on this predicate is employed, and show that although this algorithm makes greedy decisions it produces a segmentation that satisfy global properties. The algorithm runs in time nearly linear in the number of graph edges and is also fast in practice. An important characteristic of the method is its ability to preserve detail in low-variability image regions while ignoring detail in high-variability regions.

**Hierarchical segmentation.**

This segmentation method introduces a multi-scale theory of piecewise image modeling, called the scale-sets theory, and which can be regarded as a region-oriented scale-space theory. A general formulation of the partitioning problem which involves minimizing a two-term-based energy, of the form  $D + \mu C$ , where  $D$  is a goodness-of-fit term and  $C$  is a regularization term. Such energies arise from basic principles of approximate modeling and relate them to operational rate/distortion problems involved in lossy compression problems. An important subset of these energies constitutes a class of multi-scale energies in that the minimal cut of a hierarchy gets coarser and coarser as parameter  $\mu$  increases. This allows to devise a procedure to find the complete scale-sets representation of this family of minimal cuts. Considering then the construction of the hierarchy from which the minimal cuts are extracted, ending up with an exact and parameter-free algorithm to build scale-sets image descriptions whose sections constitute a monotone sequence of upward global minima of a multi-scale energy, which is called the "scale climbing" algorithm. This algorithm can be viewed as a continuation method along the scale dimension or as a minimum pursuit along the operational rate/distortion curve. Furthermore, the solution verifies a linear scale invariance property which allows to completely postpone the tuning of the scale parameter to a subsequent stage.

**Quickshift**

Quickshift (Vedaldi et al., 2008) is a kernelized version of a mode seeking

algorithm similar in concept to mean shift (Comaniciu et al., 2002; Fukunaga et al., 1975) or medoid shift (Sheikh et al., 2007). Given  $N$  data points  $x_1, \dots, x_N$ , it computes a Parzen density estimate around each point using, for example, an isotropic Gaussian window:

$$P(x) = \frac{1}{2\pi\sigma^2 N} \sum_{i=1}^N e^{-\frac{\|x-x_i\|^2}{2\sigma^2}} \quad (3.1)$$

Once the density estimate  $P(x)$  has been computed, quick shift connects each point to the nearest point in the feature space which has a higher density estimate. Each connection has a distance  $d_x$  associated with it, and the set of connections for all pixels forms a tree, where the root of the tree is the point with the highest density estimate. To obtain a segmentation from a tree of links formed by quick shift, a threshold  $\tau$  is chosen and break all links in the tree with  $d_x > \tau$ . The pixels which are a member of each resulting disconnected tree form each segment.

### SLIC superpixels.

The SLIC superpixel algorithm (Achanta et al., 2012) clusters pixels in the combined five-dimensional color (CIELAB color space) and image plane (xy) space to efficiently generate compact, nearly uniform superpixels. It is basically based on the k-mean algorithm. The number of desired cluster corresponds to the number of desired superpixels. The employed distance is based on a weighted sum of a color based distance and a plane space distance. This method produces superpixels achieving a good segmentation quality measured by boundary recall and under-segmentation error. The benefits of superpixel approaches have already been shown, as they increase performance over pixel-based methods.

## 3.3 Feature extraction

The extraction of feature is an important step in order to obtain an accurate classification. The feature can be handcrafted. Such feature have been extensively employed for remote sensing application. The features could also be learned for a specific classification task using convolutional neural

networks (Demuth et al., 2014). Here, the proposed features have been derived manually, most are standards of the literature.

### 3.3.1 Point-based lidar features.

Lidar-derived features require a consistent neighborhood for their computation. For each lidar point, 3 cylindrical neighborhoods, aligned with the vertical axis, are used (1 m, 3 m and 5 m radii, infinite height). A cylinder appears to be the most relevant environment in forested areas so as to take into account the variance of altitudes of the lidar points. Three radius values are considered so as to handle the various sizes of the trees and assuming a feature selection step will prune the initial set of attributes. Two vegetation density features,  $\mathcal{D}_1$  and  $\mathcal{D}_2$ , are computed: the first one based on the number of local maxima within the neighborhoods, and the second one related to the number of non-ground points within the neighborhoods (ground points were previously determined by a filtering step).  $\mathcal{D}_1$  and  $\mathcal{D}_2$  are calculated as follows:

$$\mathcal{D}_1 = \sum_{r_1 \in \{1,3,5\}} \sum_{r_2 \in \{1,3,5\}} Nt_{r_1,r_2}, \quad (3.2)$$

$$\mathcal{D}_2 = \frac{1}{3} \sum_{r \in \{1,3,5\}} \frac{Ns_r}{Ntot_r}, \quad (3.3)$$

where  $Nt_{r_1,r_2}$  is the number of local maxima retrieved from a  $r_1$  maximum filter within the cylindrical neighborhood of radius  $r_2$ .  $Ns_r$  is the number of points classified as ground points within the cylindrical neighborhood of radius  $r$  and  $Ntot_r$  is the total number of points within the cylindrical neighborhood of radius  $r$ . Additionally, the scatter  $\mathcal{S}$  and the planarity  $\mathcal{P}$  features are computed following Weinmann et al. (2015):

$$\mathcal{S} = \frac{1}{3} \sum_{r \in \{1,3,5\}} \frac{\lambda_{3,r}}{\lambda_{1,r}}, \quad (3.4)$$

$$\mathcal{P} = \frac{1}{3} \sum_{r \in \{1,3,5\}} 2 \times (\lambda_{2,r} - \lambda_{3,r}), \quad (3.5)$$

where  $\lambda_{1,r} \geq \lambda_{2,r} \geq \lambda_{3,r}$  are the eigenvalues of the covariance matrix within the cylindrical neighborhood of radius  $r$ . They are retrieved with a

standard Principal Component Analysis.

Statistical features, known to be relevant for vegetation type (mainly tree species) classification (Dalponte et al., 2014; Torabzadeh et al., 2015), are also derived. For each lidar point, the same 3 cylindrical neighborhoods are used. Two basic information from the lidar data, namely height and intensity, are used to derive statistical features. A statistical feature  $f_d$ , derived from an original feature  $f_o$ , (height or intensity) is computed as follows:

$$f_d = \frac{1}{3} \sum_{r \in \{1,3,5\}} f_s(\mathbf{p}_{r,f_o}), \quad (3.6)$$

where  $f_s$  is a statistical function (minimum; maximum; mean; median; standard deviation; median absolute deviation from median (medADmed); mean absolute deviation from median (meanADmed); skewness; kurtosis; 10<sup>th</sup>, 20<sup>th</sup>, 30<sup>th</sup>, 40<sup>th</sup>, 50<sup>th</sup>, 60<sup>th</sup>, 70<sup>th</sup>, 80<sup>th</sup>, 90<sup>th</sup> and 95<sup>th</sup> percentiles), and  $\mathbf{p}_{r,f_o}$  a vector containing the sorted values of the original feature  $f_o$  within the cylindrical neighborhoods of radius  $r$ . All the statistical functions are used for the height. Only the mean is used for the intensity: it is hard to know how well the sensor is calibrated and a suitable correction of intensity values within tree canopies has not yet been proposed.

24 features are extracted during this step; 2 related to vegetation density, 2 related to the 3D local distribution of the point cloud (planarity and scatter), and 20 statistical features.

### 3.3.2 Pixel-based multispectral features.

The original 4 spectral bands of the image are kept and considered as multispectral features. The Normalized Difference Vegetation Index (NDVI), (Tucker, 1979), the Difference Vegetation Index (DVI), (Bacour et al., 2006) and the Ratio Vegetation Index (RVI) (Jordan, 1969) are computed as they are relevant vegetation indices. Many other vegetation indices have been proposed (Bannari et al., 1995). Indeed, they can provide more information about the species than the original bands alone (Zargar et al., 2011). As the point-based lidar features, statistical features are also derived from each band and each vegetation index according to Equation 3.6

(3 circular neighborhoods of 1 m, 3 m and 5 m radii). Other statistical functions are used (minimum; maximum; mean; median; standard deviation; mean absolute deviation from median (meanADmed); mean absolute deviation from mean (meanADmean); median absolute deviation from median (medADmed); median absolute deviation from mean (medADmean)). Finally, the pixel-based multispectral feature set is composed of 70 attributes.

### 3.3.3 Pixel-based lidar features.

The lidar features are rasterized at the same resolution of the multispectral image using a pit-free method proposed in (Khosravipour et al., 2014). This rasterization method is interesting because it produces smooth images that will lead to better results for classification and regularization (Li et al., 2013). Such data fusion process at the feature level is valid since both datasets have approximately the same spatial resolution. The Canopy Height Model (CHM) is also computed using this method, at the same spatial resolution using an existing 1 m Digital Terrain Model provided with the filtered point cloud (Ferraz et al., 2016b). The CHM is very important as it allows to derive the height above the ground and is known as a very discriminative feature for classification (Mallet et al., 2011; Weinmann, 2016).

### 3.3.4 Object-based feature map.

The pixel-based multispectral and lidar maps are merged so as to obtain a pixel-based feature map. Then, an object-based feature map is created using the over-segmentation and the pixel-based feature map. The value  $v_t$  of a pixel belonging to an object  $t$  in the object-based feature map is computed as follows:

$$v_t = \frac{1}{N_t} \sum_{p \in t} v_p, \quad (3.7)$$

where  $N_t$  is the number of pixels in object  $t$ , and  $v_p$  is the value of the pixel  $p$ . If a pixel does not belong to a tree, it keeps the value of the pixel-based feature map. Here, only the mean value of the pixels within the tree is envisaged but one can also consider other statistical values (minimum, maximum, percentiles etc.).

Other morphological features could also be directly derived from the lidar

cloud point at the object-level. For instance, an alpha-shape can be performed on the individual trees (Vauhkonen et al., 2010) and a penetration feature can be derived as it can help to classify vegetation type (mainly tree species). However, low point densities (1-5 points/m<sup>2</sup>) compatible with large-scale lidar surveys are not sufficient in order to derive a significant penetration indicator.

## 3.4 Classification

The classification is performed using a supervised classifier, in order to discriminate the vegetation type (mainly tree species) provided by an existing forest land-cover database. The classifier used in this study is the Random Forest (RF), implemented in OpenCV (Bradski et al., 2008), as it has been shown relevant in the literature (Belgiu et al., 2016) and in a previous study compared to SVM (Dechesne et al., 2016), since it provide similar results while being faster. Indeed, the SVM classifier very efficient (Vapnik, 2013), however, the training of such classifier is time consuming, especially when the number of training sample increases (which is the case when the learning is based on a database). Furthermore, when using different type of feature (here spectral and lidar features) a special attention should be paid to the employed kernel. The RF classifier is therefore preferred because is natively handle features of different type and works better when the number of samples increases. The outputs of the classification are (i) label map and (ii) probability map (posterior class probabilities for each pixel/object). This probability map is the main input for the subsequent regularization step.

In order to reduce the computation times, a feature selection is carried out to identify an "optimal" feature subset. Additionally, a strategy is proposed in order to select the most suitable training pixels for an existing land-cover forest maps, subsequently improving the classification accuracy.

### 3.4.1 Training set design

Using an existing forest land-cover (LC) database for training a model is not straightforward (Gressin et al., 2013a; Radoux et al., 2014; Maas et al.,

2016). First, locally it can suffer from a lack of information (not all the classes of interest are present). Secondly, this database may also be semantically and, more frequently, geometrically incorrect: changes may have happened (forest cut or grow) and the geodatabase may have been generalized, resulting in sharp polygon vertices that do not exactly correspond to the class borders. Thirdly, in many forest LC databases, polygons of a given vegetation type (mainly tree species) may contain other vegetation type (mainly tree species) in a small proportion.

In order to correct the potential errors of the LC database, a k-means clustering has been therefore performed on each of the labels in the training area. We assume that erroneous pixels are present in a small proportion and that therefore the main cluster corresponds to the class of interest. Let  $p_{i-c,t}$  be the  $i^{\text{th}}$  pixel of the vegetation type (mainly tree species)  $t$  in the cluster  $c$  of the k-means. The pixels  $P_t$  used to train the model for the vegetation type (mainly tree species)  $t$  correspond to the set:

$$P_t = \{p_{i-c,t} \mid c = \underset{c \in [1, k]}{\operatorname{argmax}} \operatorname{Card}(\cup_i p_{i-c,t})\}. \quad (3.8)$$

That is to say, only samples belonging to the main k-mean cluster among training pixels for one class are kept in the training dataset.

In practice,  $k = 3$ : the main cluster corresponds to the label of interest whereas the two other ones correspond to the ground and minority vegetation type (mainly tree species) within the polygons. 1000 samples per class are then randomly selected in order to design the final training set. Such selection is a bit exclusive and does not allow a lot of variability of the selected training pixels. A more reasonable way to select training pixels is to keep the pixels such as:

$$P_t = \{p_{i-c,t} \mid \frac{\operatorname{Card}(\cup_i p_{i-c,t})}{\operatorname{Card}(\cup_{\forall i, \forall c_k} p_{i-c_k,t})} \geq u_p\}. \quad (3.9)$$

where  $u_p$  is a proportion defined by the user. Such selection is equivalent to select the pixels that are in a cluster which size represent a significant proportion of the total pixels labeled as  $t$  in the Forest LC. in this case, the

number of clusters for the k-mean can be increased. In the experiments,  $u_p$  was set to 0.3 (i.e. 30%) and  $k$  to 4.

### 3.4.2 Feature selection

Due to the high number of features involved, an automatic Feature Selection (FS) has been integrated. This selection is composed of two steps: the choice of the number of features to select and the feature selection itself. Indeed, the choice of the number of features is very important because it enables to greatly decrease the computation times.

The Sequential Forward Floating Search (SFFS) (Pudil et al., 1994) algorithm is used for both steps. The SFFS algorithm has two main advantages: (i) it can be used with many classification scores (in this study, the Kappa coefficient), (ii) it enables to access to the evolution of the classification score/accuracy according to the number of selected features. The accuracy of the classification is assessed through the Kappa coefficient of the RF classifier. The SFFS algorithm selects  $p$  features by maximizing FS score criterion (the Kappa coefficient). In order to retrieve the optimal number of features, the SFFS algorithm is performed  $n$  times on different training sets with  $p$  equal to the total number of features (95). The classification accuracy is conserved for each selection of  $s$  features ( $s \in [1, p]$ ) and averaged over the  $n$  iterations. The number of optimal features  $n_{\text{opt}}$  corresponds to the size of the selection of  $s$  features having the maximal mean accuracy.

The feature selection is then carried out for each area of interest (one selection for each area) with  $p = n_{\text{opt}}$ . The selected features are used for both the classification and the energy minimization framework. The feature selection could be carried out only once samples from multiple areas in order to retrieve the relevant features and thus only compute them.

## 3.5 Smoothing

After the classification, we assume that a label map is provided for the areas of interest, and is accompanied with a class membership probability

map, which provides, for each pixel of the image, the posterior class membership for all classes of interest. These are the necessary inputs for all methods described below.

Here, both local and global methods are tested. For local techniques, majority voting and probabilistic relaxation are selected. For global methods, various energy formulations based on a feature-sensitive Potts model are proposed.

### 3.5.1 Local methods

#### Filtering

An easy way to smooth a probability map is to filter it. All the pixels in a  $r \times r$  pixels moving window  $\mathcal{W}$  are combined in order to generate an output label of the central pixel. The most popular filter is the majority filter. Firstly, the class probabilities are converted into labels, assuming that the label of pixel  $\mathbf{x}$  is the label of the most probable class.

$$C(\mathbf{x}) = [c_i | P(\mathbf{x}, c_i) \geq P(\mathbf{x}, c_j) \forall j], \quad (3.10)$$

with  $i, j \in [1, n_c]$ , where  $n_c$  is the number of classes. From this label image, the final smoothed result is obtained by taking the majority vote in a local neighborhood.

$$C_{smooth}(\mathbf{x}) = \arg \max_i \left[ \sum_{\mathbf{u} \in \mathcal{W}} [C(\mathbf{u}) = c_i] \right]. \quad (3.11)$$

The majority filter does not take into account the original class posterior likelihoods. For example, if in a  $5 \times 5$  neighborhood, 13 pixels have a probability of 51% for class *Douglas fir*, and the 12 other pixels have a 99% probability for class *beech*, the voting will nevertheless prefer *Douglas fir*. There are variants which give pixels closer to the center more voting power, but typically yield similar results. Other filters have been developed such as the Gaussian Filter, the Bilateral Filter (Paris et al., 2006; Paris et al., 2009) and the Edge-Aware Filter (Chen et al., 2007).

### Probabilistic relaxation

The probabilistic relaxation aims at homogenizing probabilities of a pixel according to its neighboring pixels. The relaxation is an iterative algorithm in which the probability at each pixel is updated at each iteration in order to have it closer to the probabilities of its neighbors (Gong et al., 1989). It was adopted for simplicity reasons. First, good accuracies are reported with decent computing time, which is beneficial over large scales. Secondly, it offers an alternative to edge aware/gradient-based techniques that may not be adapted in semantically unstructured environments like forests. The probability  $P_k^t(\mathbf{u})$  of class  $k$  at a pixel  $\mathbf{u}$  at the iteration  $t$  is defined by  $\delta P_k^t(\mathbf{u})$  which depends on:

- The distance  $d_{\mathbf{u}, \mathbf{v}}$  between the pixel  $\mathbf{u}$  and its neighbors  $\mathbf{v}$  (the pixels that are distant of less than  $r$  pixels from  $\mathbf{u}$ ).
- A co-occurrence matrix  $T_{k,l}$  defining a priori correlation between the probabilities of neighboring pixels. The local co-occurrence matrix has been tuned arbitrarily, but can also be estimated using training pixels (Volpi et al., 2015). The matrix is expressed as follow:

$$T_{k,l} = \begin{bmatrix} 0.8 & p & \cdots & p \\ p & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & p \\ p & \cdots & p & 0.8 \end{bmatrix}, \text{ with } p = \frac{0.2}{n_c - 1}.$$

The update factor is then defined as:

$$\delta P_k^t(\mathbf{u}) = \sum_{\mathbf{v} \in \mathcal{N}_{\mathbf{u}}} d_{\mathbf{u}, \mathbf{v}} \sum_{l=1}^{n_c} T_{k,l}(\mathbf{u}, \mathbf{v}) \times P_l^t(\mathbf{v}). \quad (3.12)$$

In order to keep the probabilities normalized, the update is performed in two steps using the unnormalized probability  $Q_k^{t+1}(\mathbf{u})$  of class  $k$  at a pixel  $\mathbf{u}$  at the iteration  $t + 1$ :

$$Q_k^{t+1}(\mathbf{u}) = P_k^t(\mathbf{u}) \times (1 + \delta P_k^t(\mathbf{u})), \quad (3.13)$$

$$P_k^{t+1}(\mathbf{u}) = \frac{Q_k^{t+1}(\mathbf{u})}{\sum_{l=1}^{n_c} Q_l^{t+1}(\mathbf{u})}. \quad (3.14)$$

### 3.5.2 Global smoothing

The global smoothing method uses only a small number of pairwise cliques between neighboring pixuseels (4-neighbors or 8-neighbors) to describe the smoothness. Over the entire resulting first order random fields, the maximization of the posterior probability leads to a smoothed results. This can be done by finding the minimum of the negative log-likelihood,  $\arg \min_C E(I, C, A)$  with

$$E(I, C, A) = \sum_{\mathbf{u} \in I} E_{\text{data}}(\mathbf{u}, P(\mathbf{u})) + \gamma \sum_{\mathbf{u} \in I, \mathbf{v} \in \mathcal{N}_{\mathbf{u}}} E_{\text{pairwise}}(\mathbf{u}, \mathbf{v}, C(\mathbf{u}), C(\mathbf{v}), A(\mathbf{u}), A(\mathbf{v})), \quad (3.15)$$

where  $P(\mathbf{u}) = [P(\mathbf{u}, c_i) | P(\mathbf{u}, c_i) \geq P(\mathbf{u}, c_j) \forall j]$ ,  $A(\mathbf{u})$  are the values of the features at pixel  $\mathbf{u}$  (such as height, reflectance...) and  $\mathcal{N}_{\mathbf{u}}$  is the 8-connected neighborhood of the pixel  $\mathbf{u}$  (only the 8-connected neighborhood has been investigated). When  $\gamma = 0$ , the pairwise term has no effect in the energy formulation; the most probable class is attributed to the pixel, leading to the same result as the classification output. When  $\gamma \neq 0$ , the resulting label map becomes more homogeneous, and the borders of the segments/stands are smoother. However, if  $\gamma$  is too high, the small areas are bound to be merged into larger areas, removing a part of the useful information provided by the classification step. The automatic tuning of the parameter  $\gamma$  has been addressed in Moser et al., 2013 but is not used here.

In spite of having only connections between neighbors, the optimization propagates information over larger distances. The problem is NP-hard, but strong approximate optimization algorithms exist (Boykov et al., 2001; Kolmogorov, 2006; Felzenszwalb et al., 2006).

Here, two formulations of  $E_{\text{data}}$  (unary term) and four formulations of  $E_{\text{pairwise}}$  (prior) are investigated.

### Unary term

A widely used formulation for the unary term is the log-inverse formulation using the natural logarithm. It corresponds to the information content in information theory and is formulated as follow:

$$E_{\text{data}} = -\log(P(\mathbf{u})). \quad (3.16)$$

It highly penalizes the low-probability classes but increase the complexity with potential infinite values.

An other simple formulation for the unary term is the linear formulation,

$$E_{\text{data}} = 1 - P(\mathbf{u}). \quad (3.17)$$

It penalizes less than the log-inverse formulation but has the advantage of having values lying in  $[0, 1]$ .

### Prior

In this work, the prior has a value depending on the class of neighboring pixels. In the four formulations, two neighboring pixels pay no penalty if they are assigned to the same class. Two basic and popular priors, the *Potts model* and the *contrast-sensitive Potts model* (called here *z-Potts model*), are investigated. In the *Potts model*, two neighboring pixels pay the same penalty if they are assigned to different labels, the prior for the *Potts model* is:

$$\begin{aligned} E_{\text{pairwise}}(C(\mathbf{u}) = C(\mathbf{v})) &= 0, \\ E_{\text{pairwise}}(C(\mathbf{u}) \neq C(\mathbf{v})) &= 1. \end{aligned} \quad (3.18)$$

In the *z-Potts model*, the penalty for a change of label depends on the gradient of height between two neighboring pixels. The *z-Potts model* is a standard *contrast-sensitive Potts model* applied to the height obtained from the point clouds. Here, since we are dealing with forest stands that are likely to exhibit distinct heights, the gradient of the height map (given with the 3D lidar point cloud) is computed for each of the four directions separately. The

maximum  $M_g$  over the whole image in the four directions is used to compute the final pairwise energy. A linear function has been used: the penalty is highest when the gradient is 0, and decreases until the gradient reaches its maximum value. The prior of the *z-Potts model* is therefore:

$$\begin{aligned} E_{\text{pairwise}}(C(\mathbf{u}) = C(\mathbf{v})) &= 0, \\ E_{\text{pairwise}}(C(\mathbf{u}) \neq C(\mathbf{v})) &= 1 - \frac{g_{\mathbf{u} \rightarrow \mathbf{v}}}{M_g}, \end{aligned} \quad (3.19)$$

where  $g_{\mathbf{u} \rightarrow \mathbf{v}}$  is the gradient between pixel  $\mathbf{u}$  and pixel  $\mathbf{v}$ , i.e., the absolute value of the height difference of the two pixels.

An other pairwise energy investigated is a global feature sensitive energy (called here *Exponential-features model*). The pairwise energy is computed with respect to a pool of  $n$  features. When the features have close values, the penalty is high and decreases when the features tends to be very different. The pairwise energy in this case is expressed as follows:

$$\begin{aligned} E_{\text{pairwise}}(C(\mathbf{u}) = C(\mathbf{v})) &= 0, \\ E_{\text{pairwise}}(C(\mathbf{u}) \neq C(\mathbf{v})) &= \frac{1}{n} \sum_{i=1}^n \exp(-|A_i(\mathbf{u}) - A_i(\mathbf{v})|), \end{aligned} \quad (3.20)$$

where  $A_i(\mathbf{u})$  is the value of the  $i^{\text{th}}$  feature of the pixel  $\mathbf{u}$ . To compute such energy, the features need to be first normalized (i.e., zero mean, unit standard deviation) in order ensure that they all have the same dynamic.

The last formulation investigated is also a global feature sensitive energy (called here *Distance-features model*). The pairwise energy is still computed with respect to a pool of  $n$  features. In this case, the energy is computed according to the distance between the two neighboring pixels in the feature space, the penalty is high when the pixels are close in the feature space and decrease when they get distant. The pairwise energy in this case is expressed as follow:

$$\begin{aligned} E_{\text{pairwise}}(C(\mathbf{u}) = C(\mathbf{v})) &= 0, \\ E_{\text{pairwise}}(C(\mathbf{u}) \neq C(\mathbf{v})) &= 1 - \|A(\mathbf{u}); A(\mathbf{v})\|_{n,2}, \end{aligned} \quad (3.21)$$

with

$$\|A(\mathbf{u}); A(\mathbf{v})\|_{n,2} = \frac{1}{\sqrt{n}} \sqrt{\sum_{i=1}^n (A_i(\mathbf{u}) - A_i(\mathbf{v}))^2}. \quad (3.22)$$

To compute such energy, the features need to be first normalized (i.e., zero mean, unit standard deviation) in order ensure that they all have the same dynamic. They are then rescaled between 0 and 1 to ensure that  $\|A(\mathbf{u}); A(\mathbf{v})\|_{n,2}$  lies in  $[0; 1] \forall (\mathbf{u}, \mathbf{v})$ .

In Dechesne et al., 2017, a high number of features was extracted from available lidar and optical images ( $\sim 100$ ) but can be selected. They can also be weighted according to their importance, computed through the Random Forest classification process. Since the most important features (20) are almost all equally weighted, it does not bring additional discriminative information for the global feature sensitive energy.

### Energy minimization

The energy minimization is performed using graph-cut methods. The graph-cut algorithm employed is the quadratic pseudo-boolean optimization (QPBO). The QPBO is a popular and efficient graph-cut method as it efficiently solves energy minimization problems (such as the proposed ones) by constructing a graph and computing the min-cut (Kolmogorov et al., 2007).  $\alpha$ -expansion moves are used, as they are an efficient way to deal with the multi-class problems (Kolmogorov et al., 2004). The QPBO is an interesting method since it allows to solve any Markov Random Field optimization problems. Thus the energy formulation can be modified in order to add more constraint to the problem. Such investigation have been envisaged. Two constraints have been tested.

### Size constraint.

Firstly, the size of a segment can be taken into account in the energy formulation. This is done by setting the pairwise term to an important value  $v_s$  (ideally  $v_s = \infty$ ) when a pixel belong to a segment defined as too small by the user (in forested area, a minimum stand size is 0.5 ha). Thus the

Equation 3.18 becomes:

$$\begin{aligned} E_{\text{pairwise}}(C(\mathbf{u}) = C(\mathbf{v})) &= f(\mathbf{u}, \mathbf{v}), \\ E_{\text{pairwise}}(C(\mathbf{u}) \neq C(\mathbf{v})) &= 1 + f(\mathbf{u}, \mathbf{v}), \end{aligned} \quad (3.23)$$

with

$$\begin{aligned} f(\mathbf{u}, \mathbf{v}) &= 0 \text{ if } \mathbf{u} \text{ and } \mathbf{v} \text{ are not in a small segment,} \\ f(\mathbf{u}, \mathbf{v}) &= v_s \text{ if } \mathbf{u} \text{ or } \mathbf{v} \text{ are in a small segment.} \end{aligned} \quad (3.24)$$

Such constraint can be applied to the other proposed priors.

### Border constraint.

The second constraint that can easily added is a related to the borders. Indeed, it is possible to set the energy to a specific value in order to ensure that a border will be created. However, an a priori border need to be defined. Let  $b(\mathbf{u}, \mathbf{v})$  be a binary function that define if a border between a pixel  $\mathbf{u}$  and  $\mathbf{v}$  wants to be set. If  $b(\mathbf{u}, \mathbf{v}) = 0$ , no borders want to be set, thus Equation 3.18 remains the same. Otherwise, if  $b(\mathbf{u}, \mathbf{v}) = 1$ , Equation 3.18 becomes:

$$\begin{aligned} E_{\text{pairwise}}(C(\mathbf{u}) = C(\mathbf{v})) &= v_b, \\ E_{\text{pairwise}}(C(\mathbf{u}) \neq C(\mathbf{v})) &= 0, \end{aligned} \quad (3.25)$$

with  $v_b$  an important value (ideally  $v_b = \infty$ ).

Adding constraints increases the computational load and time but might be interesting in order to refine even more the results. However, adding such constraints also leads to some issues. Firstly, even if the minimum size of a forest stand is clearly defined in the specifications of the forest LC database, forcing segment to have a minimum size could suppress some information (such as pure islets). Such generalization could be obtained by increasing the  $\gamma$  parameter. Secondly, adding borders means that pre-determined relevant borders could be retrieved, however, in practice, such borders can not be straightforward extracted. Thus, adding these constraint have been considered in a limited way.





4

# Results

---

4.1	Data	62
4.2	Segmentation methods	64
4.2.1	Retrieve stands borders	66
4.2.2	Add semantic information	69
4.3	Results of the method	71
4.3.1	Over-segmentation	72
4.3.2	Feature selection	72
4.3.3	Classification	76
4.3.4	Regularization	84
4.3.4	Local methods	84
4.3.4	Global methods	86
4.3.5	Importance of fusion	89
4.4	Co-products of the method	91
4.4.1	Semi-automatic update	91
4.4.2	Inventory	93

---

## 4.1 Data

### VHR optical images

The VHR images are a part of a national database. In this thesis, the images used have a spatial resolution of 50 cm. Two type of ortho-images are available, a color image (3 bands; red: 600-720 nm, green: 490-610 nm and blue: 430-550 nm) and and IRC image (3 bands; near infra-red: 750-950 nm, red and green) captured by the IGN digital cameras (Souchon et al., 2012). It is then possible to obtain four band ortho-images by the combination of the two ortho-images type.

### Airborne Laser Scanning

IGN also process lots of flights over forested areas with a laser scanning device. The airborne lidar data were collected using an Optech 3100EA device. The footprint was 0.8 m in order to increase the probability to reach the ground. The point density for all echoes ranges from 2 to 4 points/m<sup>2</sup>. Data were acquired under leaf-on conditions and fit with the standards used in many countries for large-scale operational forest mapping purposes.

A prerequisite for data fusion is the most accurate alignment of the two data (Torabzadeh et al., 2014). A frequently used technique is to geo-rectify images using ground controls points (GCPs). A geometric transformation is established between the coordinates of GCPs and their corresponding pixels in the image. It is then applied to each pixel, so that coordinate differences on those points are reduced to the lowest possible level. This method can be easily applied and is relatively fast in terms of computation time. However the use of GCPs can still cause that the unknowns in the trajectory of the platforms produce some remarkable residual errors. Automatic methods for data registration have also been developed (Habib et al., 2005; Mastin et al., 2009).

The registration between airborne lidar point clouds and VHR multi-spectral images was performed by IGN itself using ground control points. This is a standard procedure in the French mapping agency since IGN

operates both sensors and has also a strong expertise in data georeferencing (this is in fact the national institute responsible for that in France for both airborne and spaceborne sensors).

### National Forest Land Cover database

The IGN's forestry reference database is a reference tool for professionals in the wood industry and for environmental and spatial planning stakeholders.

The forest LC database is a reference vector database for forest and semi-natural environments. Developed by photo-interpretation of VHR IRC optical images, The forest LC database is realized by departmental authorities in the metropolitan territory.

#### • Forest LC database, version 1

The version 1 of the forest LC database, was developed by photo-interpretation of aerial images in infrared colors. Its minimum mapped surface area is 2.25 ha. The version 1 of the forest LC database presents the soil cover (by description of the structure and the dominant composition of wooded or natural formations), based on a departmental nomenclature ranging from fifteen to sixty positions according to diversity Forestry of the mapped department. Constituted, until 2006, at the departmental level, it is available throughout the metropolitan territory. For more than half of the departments, several versions of the version 1 of the forest LC database are available.

#### • Forest LC database, version 2

The forest LC database version 2 has been developed since 2007 by photo-interpretation of VHR IRC optical images. It assigns to each mapped range of more than 5000m<sup>2</sup> a type of vegetation formation. Its main characteristics are the following:

- A national nomenclature of 32 posts based on a hierarchical breakdown of the criteria, distinguishing, for example, pure stands from the main forest tree species in the French forest (see Figure 4.1).

- A type of vegetation formation assigned to each mapped range greater than or equal to 50 ares ( $5000 m^2$ ).
- A layer geometrically compatible with the other vegetation layers produced by the IGN.

Produced by department in metropolitan France, the version 2 of the forest LC database in 75 (out of 95) departments.

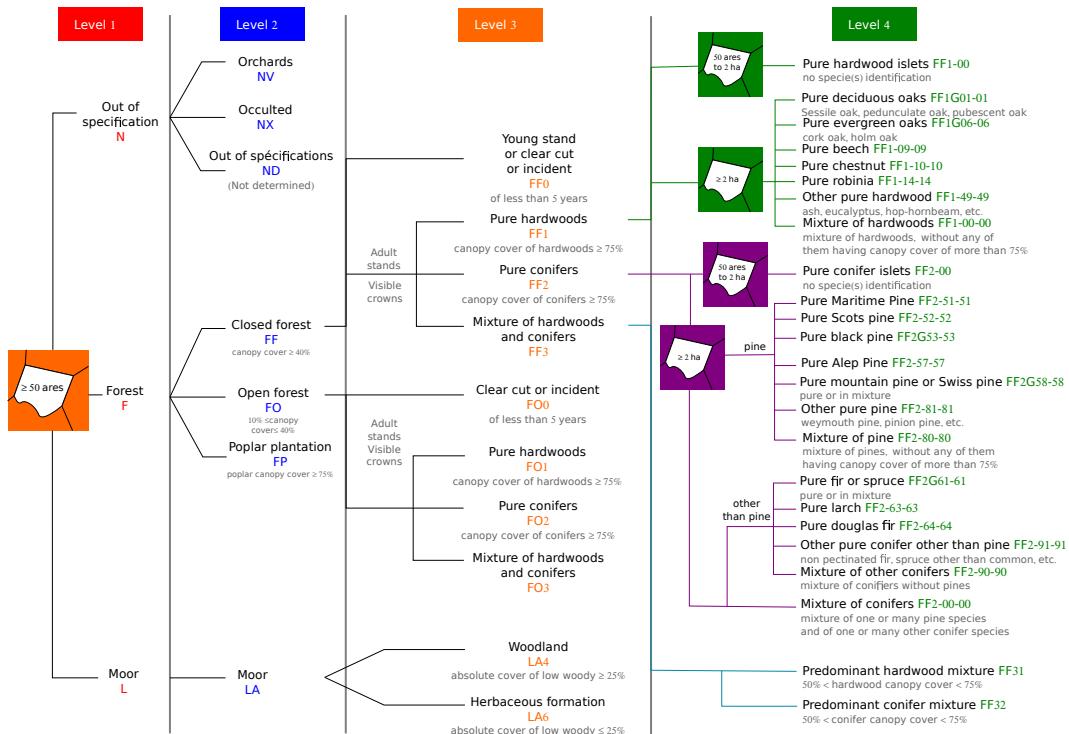


FIGURE 4.1: Organizational chart of the version 2 of the forest LC database.

## 4.2 Segmentation methods

A naive method to retrieve forest stands is to segment the input data. Such segmentation algorithms do not take into account the species information. Two algorithms were employed in order to obtain relevant stands only through the segmentation of the data. The first segmentation algorithm is the one proposed in Guigues et al., 2006. It is a hierarchical segmentation

algorithm that allows to control the level of segmentation through a unique scale parameter  $\mu$ .

The second segmentation algorithm (called here PFF) employed is presented in Felzenszwalb et al., 2004. It is a method for image segmentation based on pairwise region comparison considering the minimum weight edge between two regions in measuring the difference between them. 3 simple parameters need to be tuned in order to obtain relevant segmentation.  $\sigma$  is the standard deviation of the gaussian filter employed to smooth the image as a pre-processing (the authors recommend  $\sigma = 0.8$ ).  $k$  is a second parameter that set a scale of observation (a larger  $k$  will lead to larger segments). Finally, the parameter  $m$  permits to define the minimum size of a segment.

Such segmentation could allow to retrieve the stands borders easily. Furthermore, once the segmentation is performed, one can add semantic information using classification results.

These experiments have been performed only on one area presented in Figure 4.2. It is a  $1 \text{ km}^2$  area, the spatial resolution of the VHR optical image is  $0.5 \text{ m}$ , and the nDSM has been rasterized at the same resolution. From these data one can see that a stand is composed of zones that are not homogeneous in term of reflectance and/or height. Furthermore, one can also note that the variability between two stands in terms of reflectance and/or height might not be important.

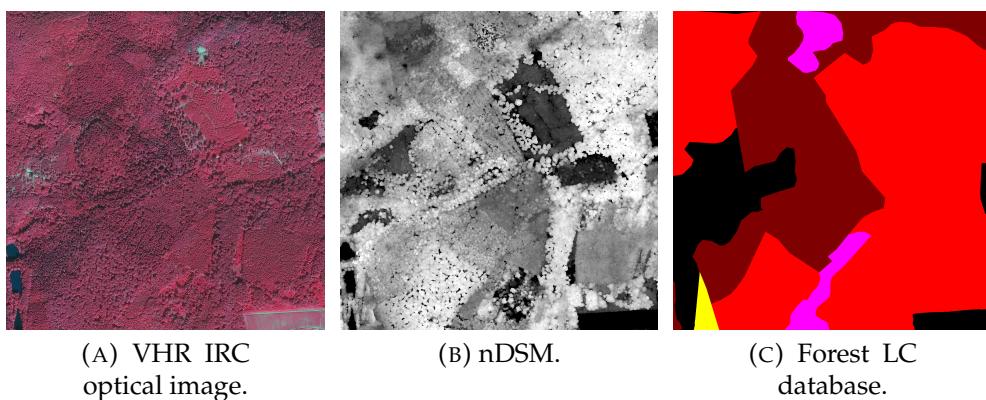


FIGURE 4.2: VHR IRC optical image, rasterized nDSM and forest LC of the proposed area for the direct segmentation tests.

### 4.2.1 Retrieve stands borders

Two strategies are employed to apply the segmentation in order to retrieve the forest stands borders from the forest LC database:

- The segmentation is applied to the VHR optical images, thus the resulting segment will correspond to "stands" that are homogeneous in terms of spectral reflectance. Since the optical images are employed by photo-interpreters in order to derive the forest LC, such segmentation should produce results similar to the forest LC.
- The segmentation is also applied to the rasterized normalized digital surface model (nDSM) (canopy height without ground relief). Such segmentation would produce "stands" that are homogeneous in term of height.

The results of the segmentation of the VHR optical image using the two segmentation algorithm is presented in Figure 4.3. In both cases, most of the borders found are not consistent with the forest LC database. Visually, the hierarchical segmentation seems to be more relevant than the PFF segmentation. However, the hierarchical segmentation produces small segments due to high variation of illumination in the image, while the PFF segments are all relatively large.

The results of the segmentation of rasterized nDSM using the two segmentation algorithm is presented in Figure 4.4. Just like the segmentation of the VHR optical image, most of the borders found are not consistent with the forest LC database. Here, the PFF segmentation seems to perform better than the hierarchical segmentation visually.

Since the segmentation on the VHR optical image and the nDSM does not allow to retrieve the borders from the forest LC database, different values of the parameter  $\mu$  for the hierarchical segmentation on the VHR optical images (see Figure 4.5). It appears that decreasing  $\mu$  does not allow to obtain the borders of the forest LC. It only leads to and over-segmentation of the image. Such over-segmentation is employed in the proposed method but is not employed as a relevant segmentation for stand delineation but as an input for object based classification.

The two proposed segmentation algorithms are very efficient for image segmentation tasks (Guigues et al., 2006; Felzenszwalb et al., 2004) but are

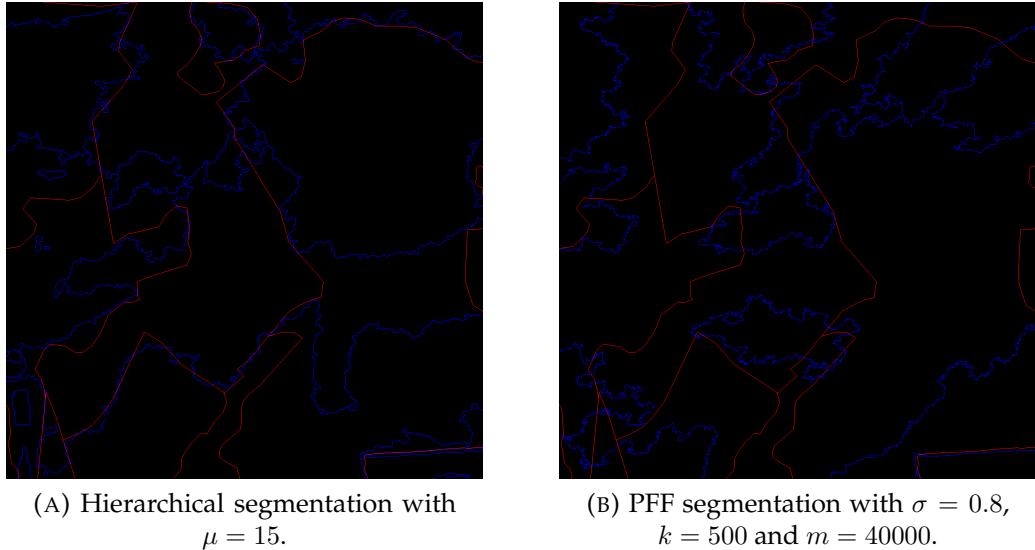


FIGURE 4.3: Result of the segmentation of the VHR optical image for the two segmentation algorithms. Blue lines correspond to the borders of the segments, red lines correspond to the borders of the forest LC.

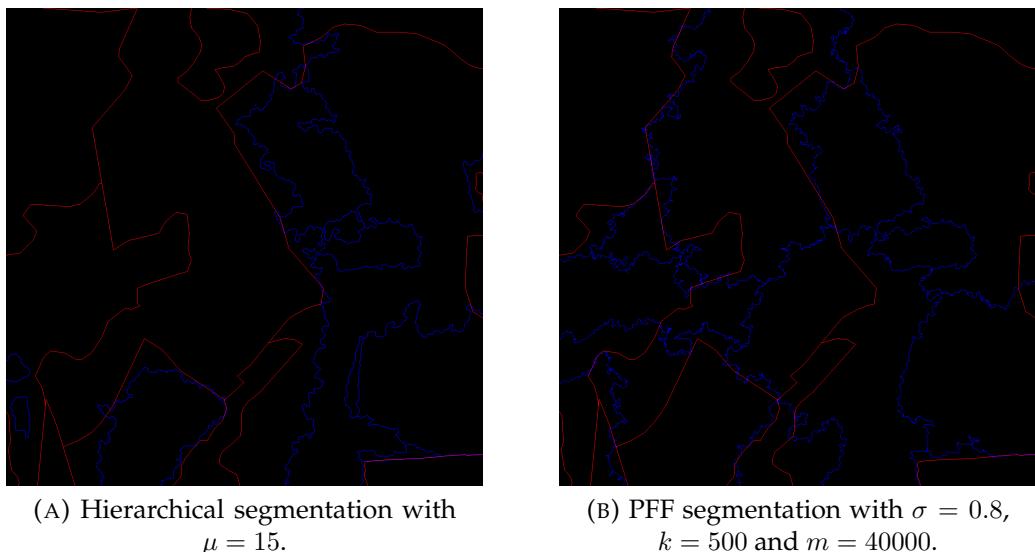


FIGURE 4.4: Result of the segmentation of the nDSM for the two segmentation algorithms. Blue lines correspond to the borders of the segments, red lines correspond to the borders of the forest LC.

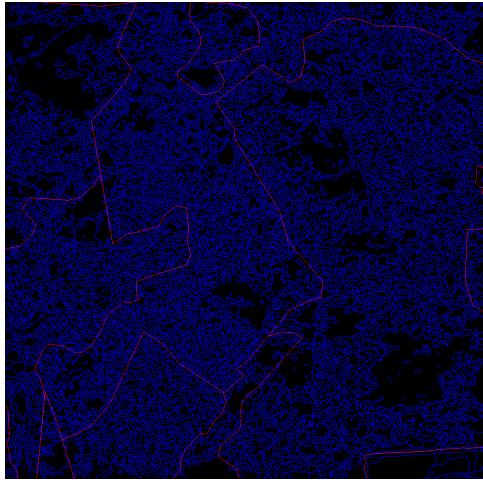
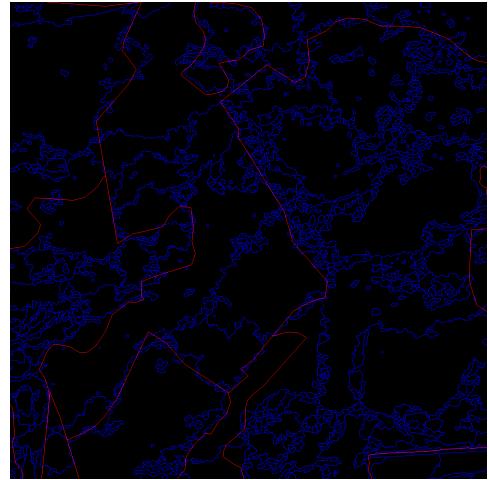
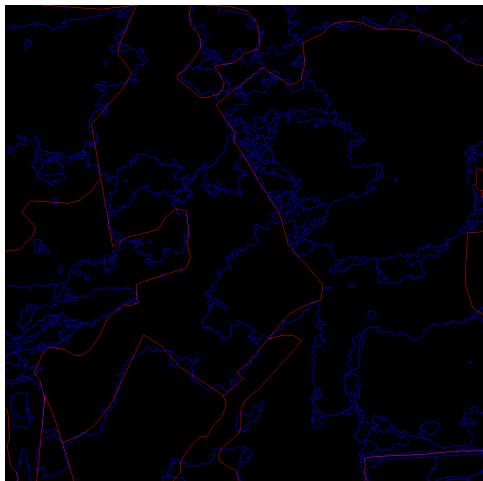
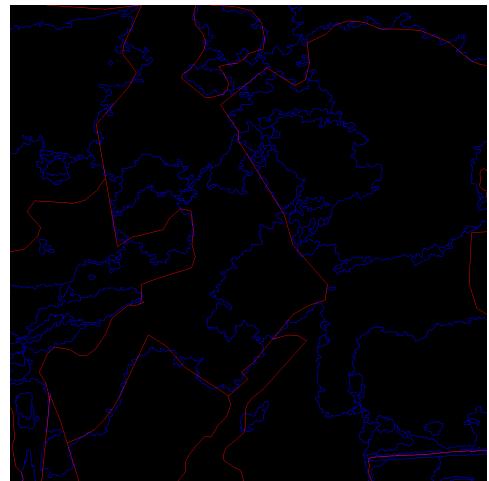
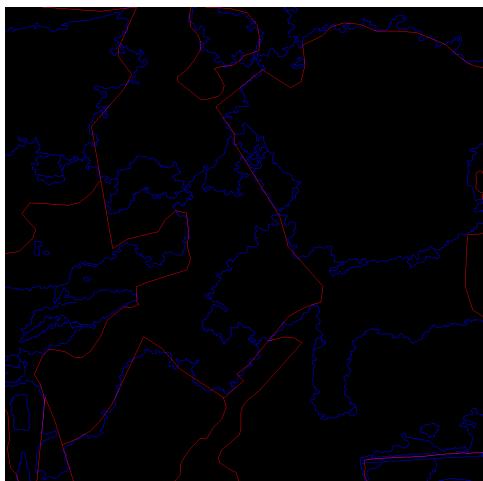
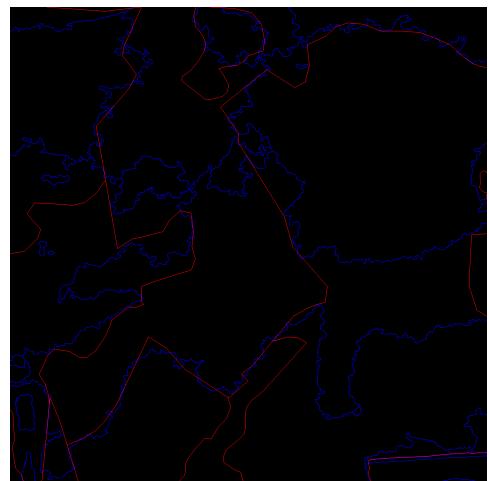
(A) Hierarchical segmentation with  $\mu = 3$ .(B) Hierarchical segmentation with  $\mu = 6$ .(C) Hierarchical segmentation with  $\mu = 8$ .(D) Hierarchical segmentation with  $\mu = 10$ .(E) Hierarchical segmentation with  $\mu = 12$ .(F) Hierarchical segmentation with  $\mu = 15$ .

FIGURE 4.5: Result of the segmentation of the VHR optical image using different values of  $\mu$  for the hierarchical segmentation. Blue lines correspond to the borders of the segments, red lines correspond to the borders of the forest LC.

not adapted to retrieve forest stands borders. However, they can produce interesting over-segmentation since they are able to retrieve some relevant borders.

### 4.2.2 Add semantic information

The classification proposed in the method give information about the species at the object level. Since the segments extracted below are large than the small objects, a majority vote can be applied for each segment. The obtained label map could be compared with the forest LC. The result of the classification for the area of interest is presented in Figure 4.6, the confusion matrix and other accuracy metrics for this classification are presented in Table C.3.

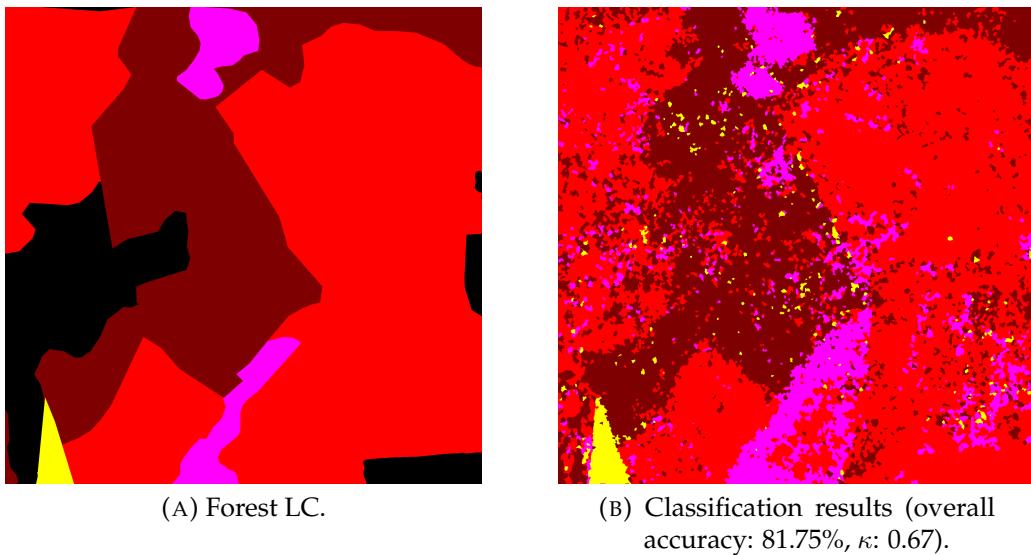


FIGURE 4.6: Forest LC and classification results.

The results of the majority vote for the segmentation of the VHR optical images is presented in Figure 4.7, the confusion matrices and other metrics are presented in Tables C.4 & C.5.

From a visual inspection, it appears that adding semantic information to the obtained segments does not allow to retrieve a relevant mapping. Indeed, some classes are not represented. For the hierarchical segmentation, the overall accuracy increases when adding semantic information through

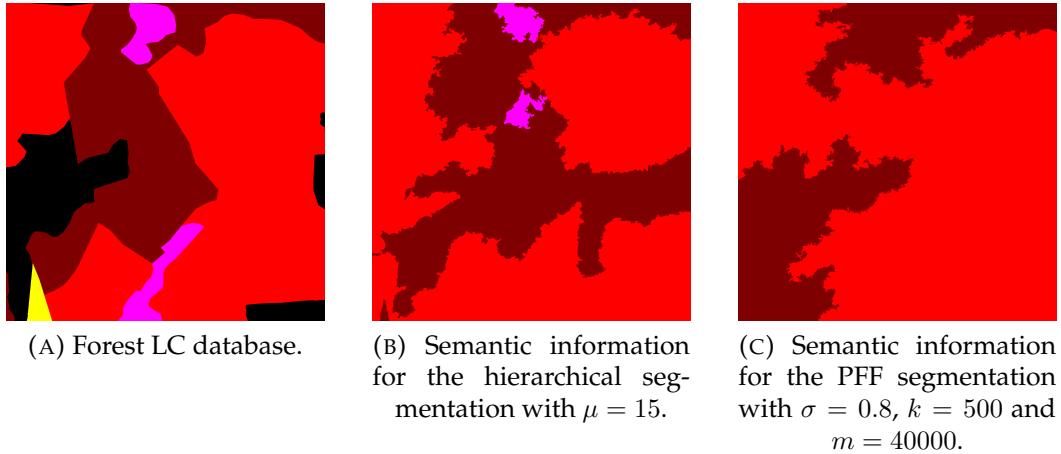


FIGURE 4.7: Results of the majority vote on the segmentation of the VHR optical images.

majority vote (81.94%, versus 81.75% for the classification). The  $\kappa$  (0.64) is also revealing a good agreement. This shows the limitation of the two metrics: the result appears to be good when inspecting them, while it is not. The Intersection over Union is more relevant in order to underline the irrelevance of the majority vote on a segmentation (41.1%). The F-score (50.68%) also concurs to this conclusion.

The results of the majority vote for the segmentation of the nDSM is presented in Figure 4.8, the confusion matrices and other metrics are presented in Tables C.6 & C.7.

The same results are observed, a majority vote applied to a segmentation equivalent to stands (in term of size) does not allow to retrieve a relevant mapping of forested areas.

From this section, two main conclusions can be drawn: firstly, the direct segmentation of the data does not allow to retrieve relevant forest stands in terms of species. Even with an addition of semantic information from a classification, the results are not sufficient for a mapping of the forest. Secondly, some metrics are not relevant in order to evaluate the results. Indeed, the overall accuracy or the  $\kappa$  are not sufficient, other metrics are needed for the correct evaluation of the results.

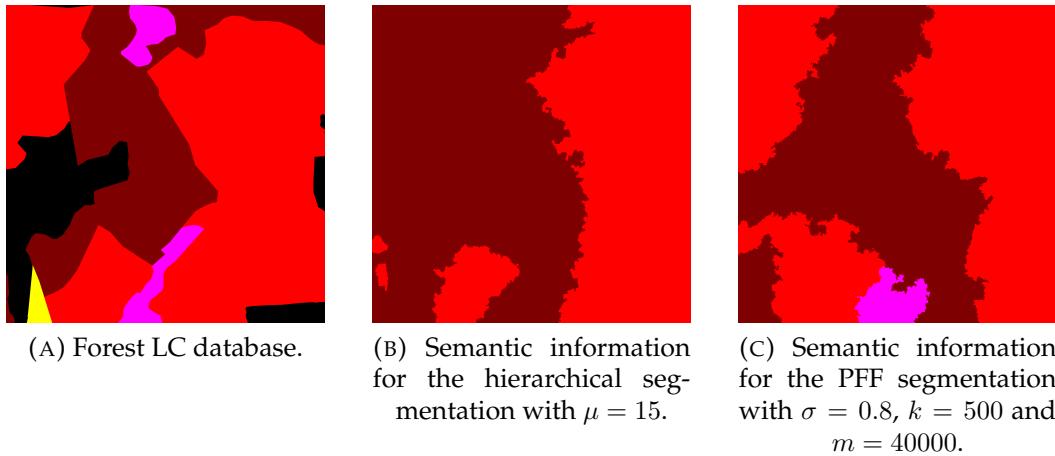


FIGURE 4.8: Results of the majority vote on the segmentation of the VHR optical images.

## 4.3 Results of the method

The proposed method is composed of different steps (defined above). Each step will be evaluated at 2 levels; the direct output will be first considered and the impact on the final segmentation will also be investigated. In order to simplify the reading of captions, when no specific information is provided the method is employed using the following parameters:

- Feature selection:
  - Selection of training samples,
  - Selection of 20 features,
  - Object-based features (lidar and spectral) obtained from the PFF over-segmentation,
  - Pixel-based features (lidar and spectral) for the regularization.
- Classification performed with:
  - Selection of training samples,
  - Object-based features (lidar and spectral) obtained from the feature selection.
- Regularization using global method with:

- $\gamma = 10$ ,
- Linear data formulation,
- *Exponential-features model* with pixel-based features obtained from the feature selection.

### 4.3.1 Over-segmentation

The results of the over-segmentation of proposed segmentation algorithms are presented in Figure 4.9.

In all the proposed methods, the resulting segments are relevant since they all represent small homogeneous objects. The objects are mostly not of the same size and shape, except for the SLIC superpixels (the aim of the methods is to obtain such uniform segments). The PFF algorithm produces objects with rough borders that are following precisely the borders observed on the images. From a visual point of view, it appears that there is no segmentation method that performs better than the others. Furthermore, it is impossible to evaluate the segmentation since no ground truth is available. Thus, the different segmentation methods are compared through the results they produce after the object-based classification (which indicates how the objects are relevant for classification) and after the regularization (how the objects impacts the final results). These results are presented in Figure 4.10

### 4.3.2 Feature selection

The feature selection have been carried out for 4 main reasons.

- It allows to determine how many features are needed for an optimal classification.
- It shows the complementarity of the data (optical images and lidar).
- It permits to understand which features are interesting for tree species classification.
- It reduces the computational loads and times.

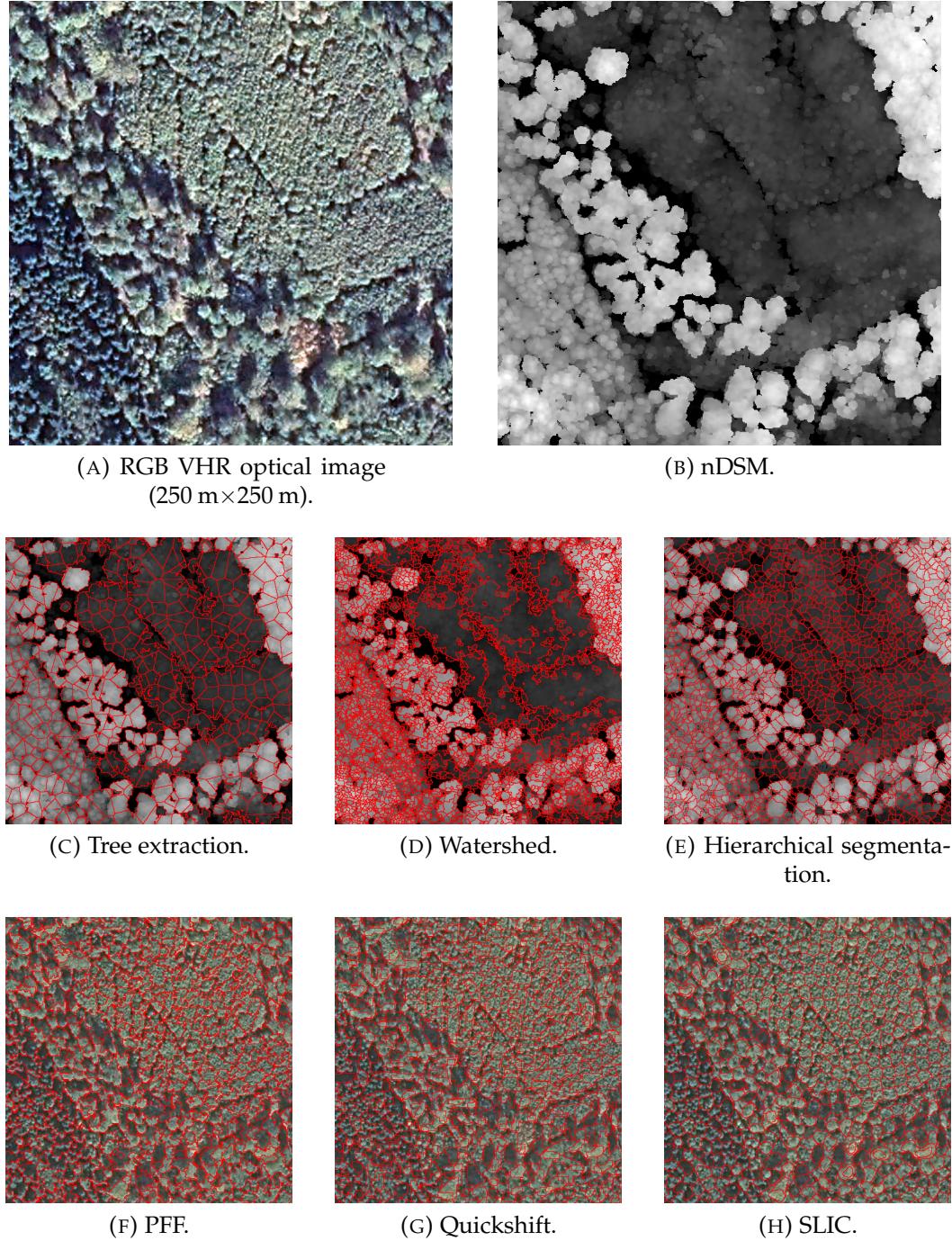


FIGURE 4.9: Over segmentation results, red corresponds to the borders found by the segmentation algorithm.

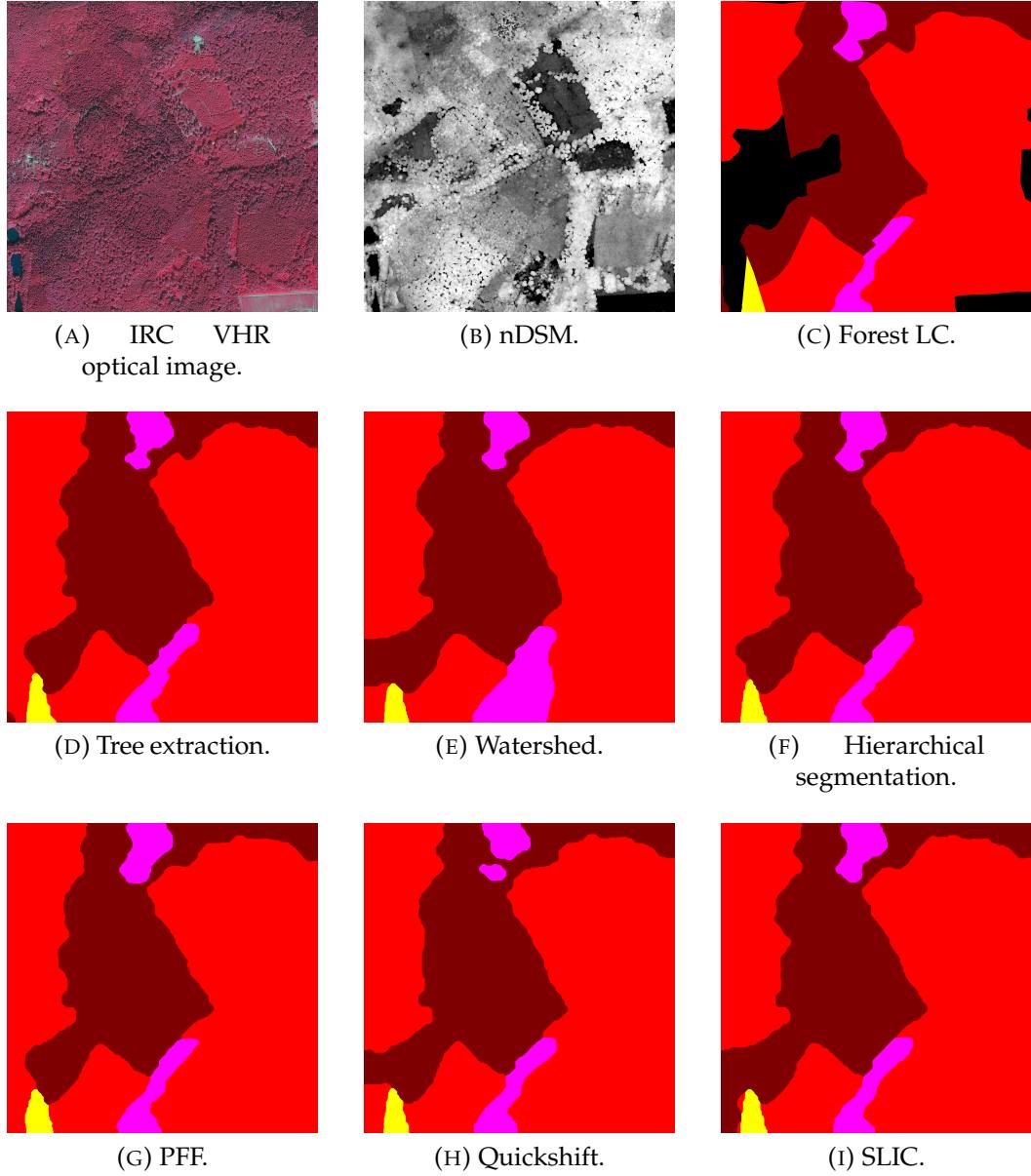


FIGURE 4.10: Over segmentation results.

### How many features for the classification?

The idea here is to obtain the number feature for an optimal classification. The idea is to run the feature selection  $N$  times using all the feature. Let  $a_{n,k}$  be an accuracy metric for the classification of the  $n^{\text{th}}$  iteration using the  $k$  ( $k \in [1, K]$ ) best features. The optimal number of feature  $k_{opt}$  is defined as follow:

$$\forall k \in [1, K], \quad \sum_{n=1}^N a_{n,k} \leq \sum_{n=1}^N a_{n,k_{opt}} \quad (4.1)$$

Here,  $K = 95$  and  $N = 50$ . This experiment have been carried out on different areas, and the optimal number of feature found is  $k_{opt} = 20$ . Thus, in all the following experiments, the classification is performed using only 20 features.

### Complementarity of the data.

Once the optimal number of features was determined, the feature selection was performed 40 times over all the test areas in order to retrieve the most relevant features. The retained attributes are presented in Figures 4.11, 4.12 and 4.13. On average, 61% of the selected features are derived from the spectral information and 39% from the lidar information for a single selection (i.e., in a random selection of 20 features, 12 are derived from VHR optical images and 8 from lidar). This shows the complementarity of both remote sensing data.

### Features for tree species classification.

For the spectral information, the features derived from the original band set are more relevant than the vegetation indices: the near-infrared derived features represent 18% of the spectral selected features, 16% for the red and the green, 15% for the blue and the DVI, only 11% for the NDVI and 10% for the RVI.

The most relevant statistical feature for the spectral information is the minimum (17% of the spectral selection). The maximum (12%), the median (11%), the mean (11%) and the standard deviation (10%) are also relevant. The other statistics are selected at less than 9% each. For more details see Figure 4.12 and 4.13.

For the lidar information, the most relevant feature is surprisingly the intensity, selected in each of the 40 selections (12% of the lidar selection, 5% of the total selection). The standard deviation (8% of the lidar selection), the maximum (7%) and the densities (5% and 6%) are also relevant. The other lidar derived features count for less than 4% each. For more details, see Figure 4.11.

### **Reduce computational loads and times.**

It is obvious that processing a reduced number of features (20 instead of 95) reduces the computational loads and times. Furthermore, if an optimal subset of feature is found, only the concerned features need to be computed. Thus, the feature computation step would be reduced to a minimum step by computing only the relevant features.

### **4.3.3 Classification**

The classification is composed of two steps, the selection of training samples and the classification itself. The classification is mainly performed using standard RF classifier (Bradski et al., 2008). Tests have also been conducted using a SVM classifier with RBF kernel (Vapnik, 2013). The classification can be impacted by two factors:

- The selection or non-selection of training pixels.
- The object-based or pixel-based analysis.

#### **Pixel-based classification versus object-based classification.**

The results of pixel-based and object-based classification is presented in Figure 4.14. The corresponding confusion matrices and accuracy metrics are presented in Tables C.20 and C.21.

The pixel-based classification (Figure 4.14c) appears more noisy than the object-based classification (Figure 4.14d). However, the pixel-based classification already provide good discrimination results. Conversely, even if the objects are roughly extracted (no specific attention is paid to the relevance of the extracted objects), the object-based classification produces more spatially consistent labels. Such results impacts the final output (see Figure 4.15 and Tables C.22 and C.23).

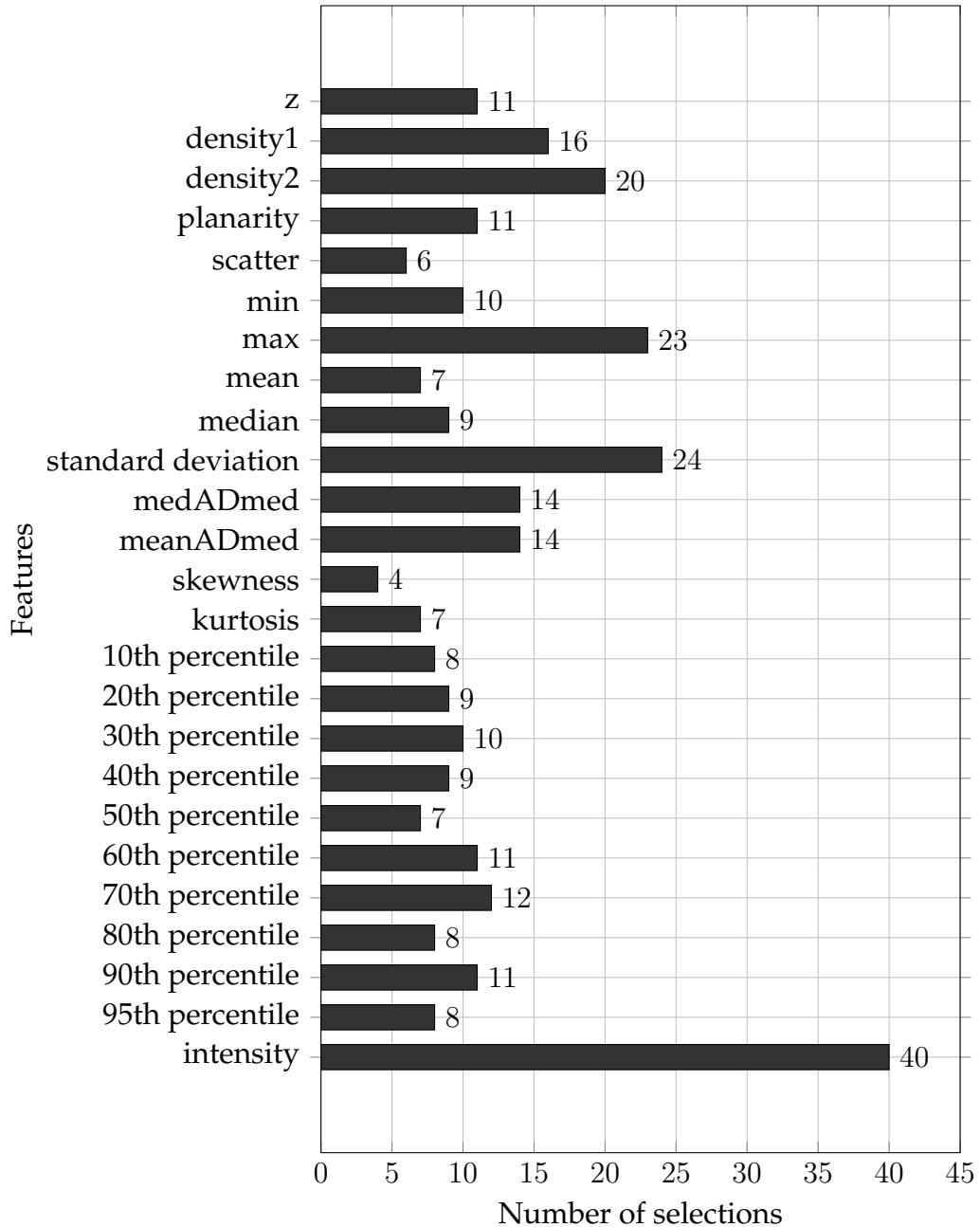


FIGURE 4.11: Result of the feature selection for the lidar features over 40 trials of 20 optimal features.

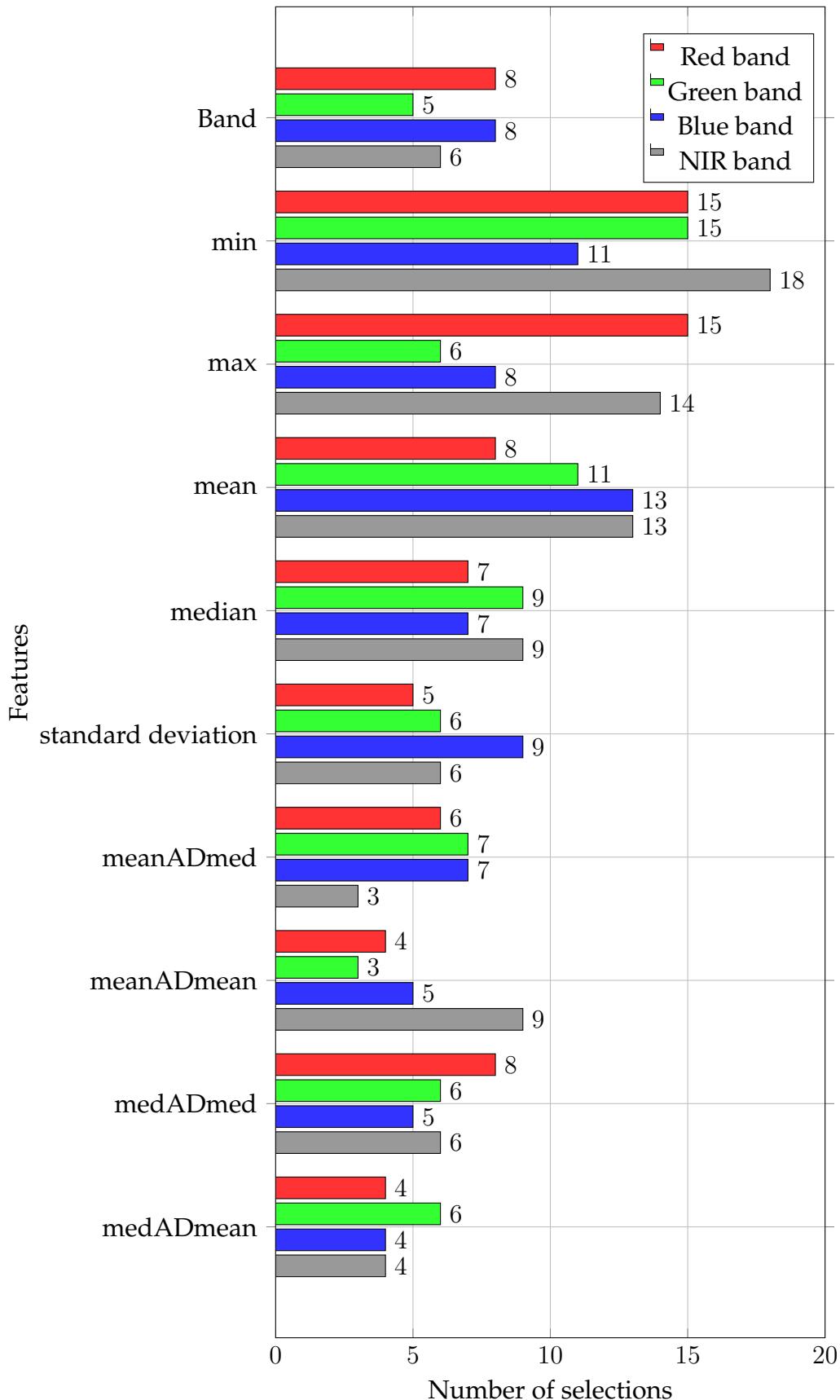


FIGURE 4.12: Result of the feature selection for the spectral features over 40 trials of 20 optimal features.

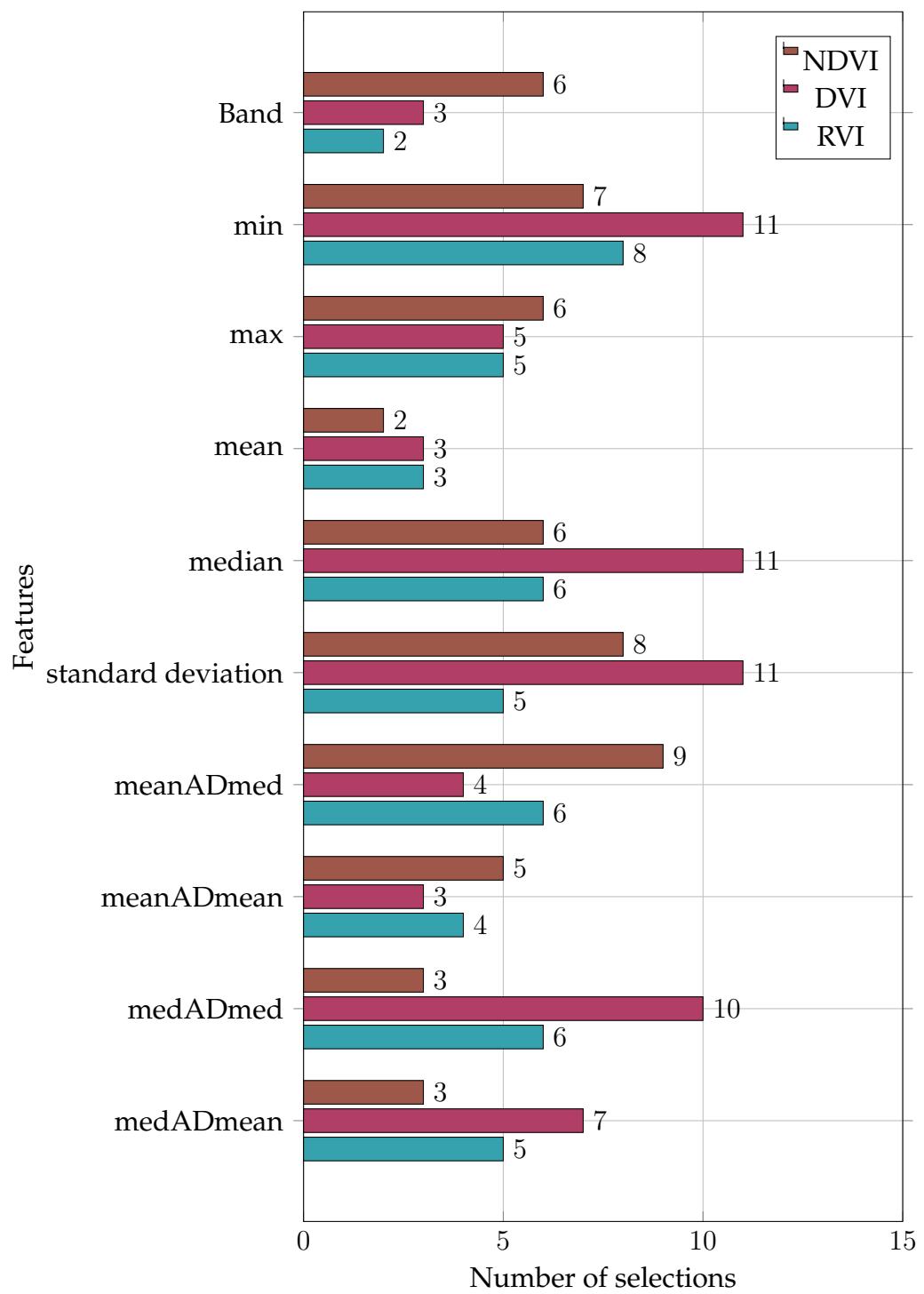


FIGURE 4.13: Result of the feature selection for the vegetation indices features over 40 trials of 20 optimal features.

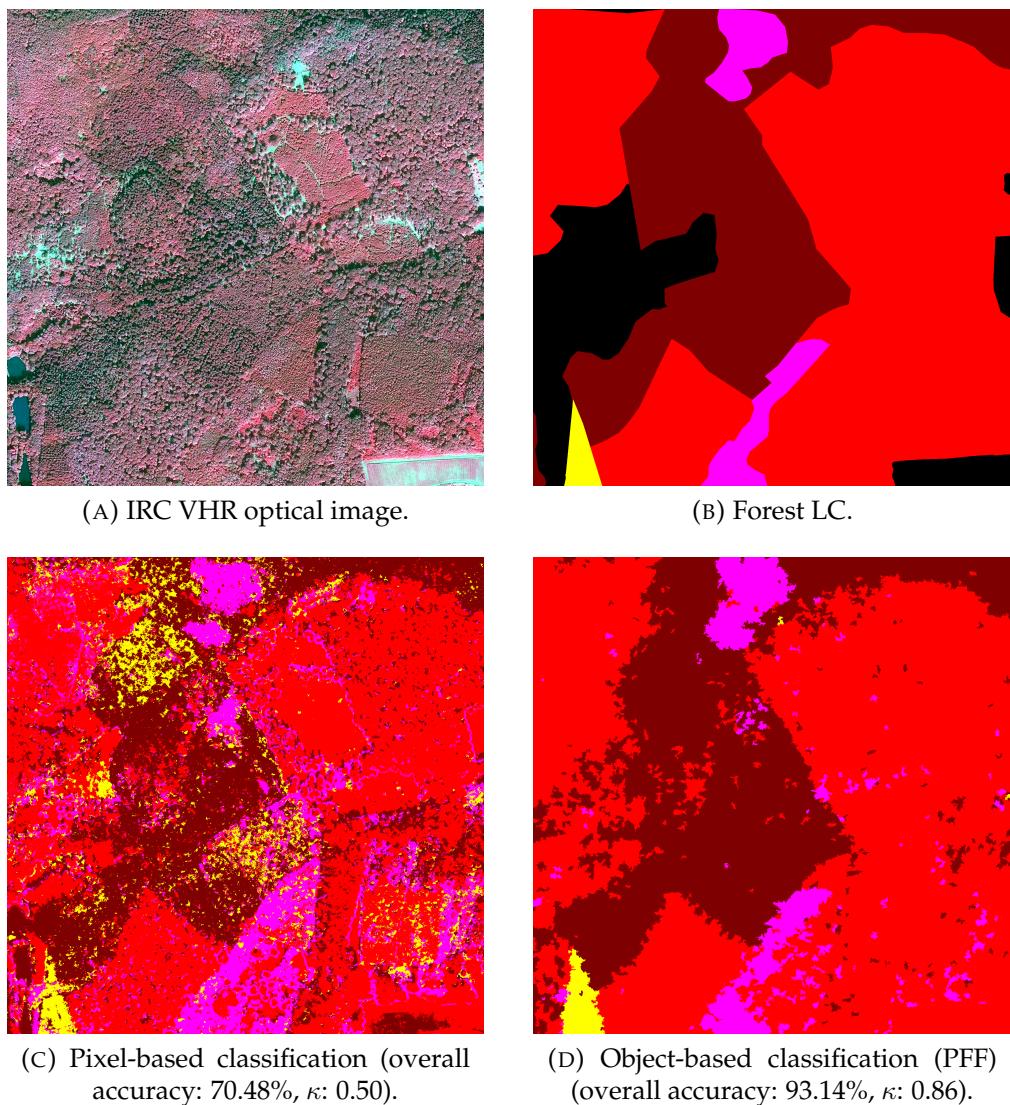


FIGURE 4.14: Classification results; pixel-based versus object-based.

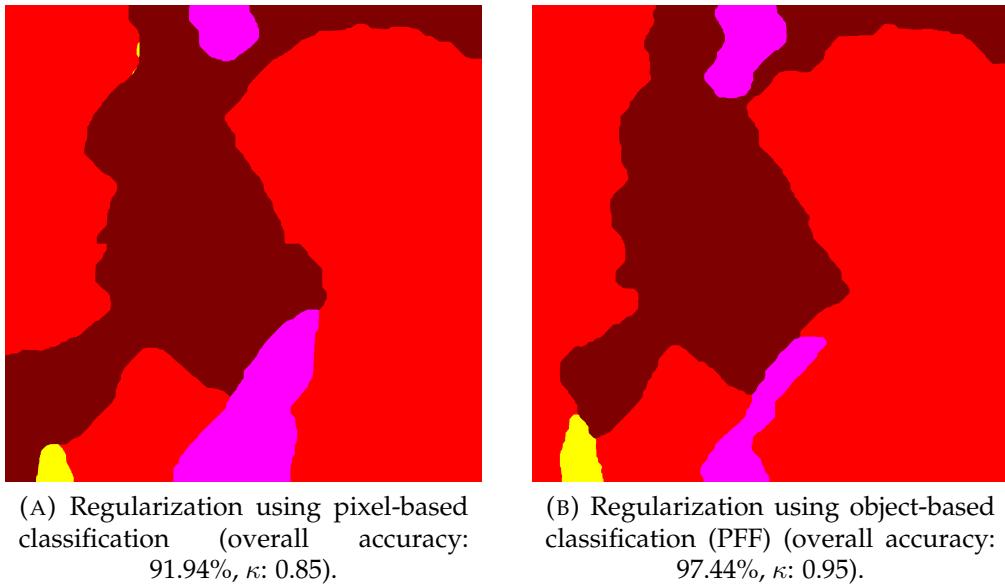


FIGURE 4.15: Regularization results; pixel-based classification versus object-based classification.

### Training set design.

The selection of the training pixels is an important step for the classification. It is a two sided problem:

- If the selected pixels are randomly taken from the polygons of the forest LC, some might not correspond to the target class, leading to confusion in the final classification.
- If the pixels are selected using a too discriminative criteria, the variability of the target class will not be taken into account, also leading to confusion in the final classification.

The Figure 4.16 shows the pixels that have been selected by the k-mean algorithm for the training. Most of the pixel from the forest LC are retained. However, the pixels that are excluded from the training are visually relevant. Indeed, they mostly correspond to:

- Shadows from the optical images,
- gaps in the canopy (retrieved from the lidar data),

- pixels that are visually different from the other pixels of the considered class (i.e. other species).

The selection of training pixels is beneficial since it allows to remove the obviously irrelevant pixels from the training set while maintaining a certain variability within classes.

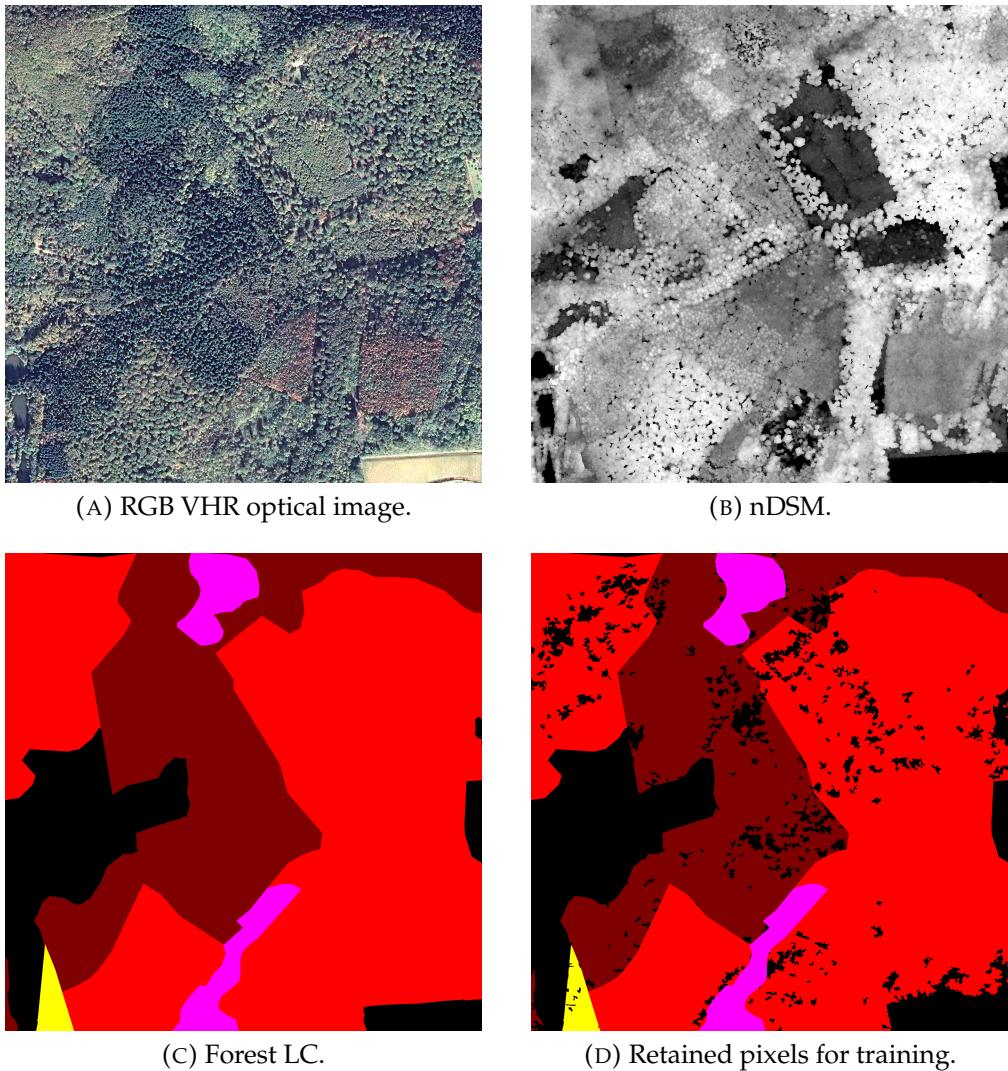
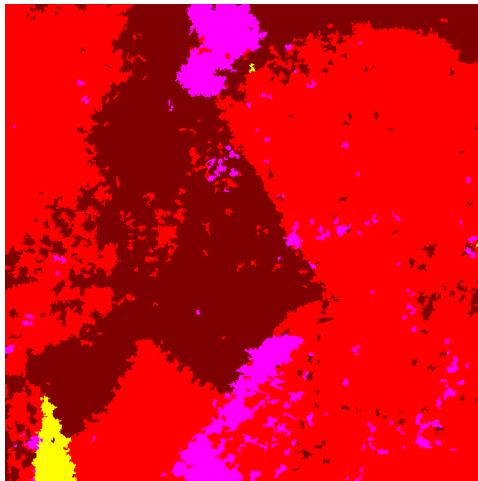
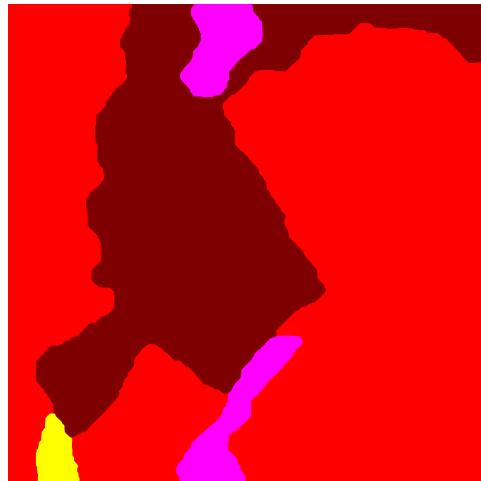


FIGURE 4.16: Selection of the training pixels.

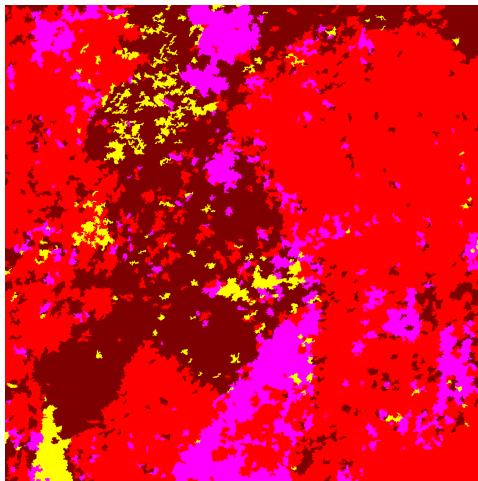
The results of the classification and regularization using different training set is presented in Figure 4.17. The confusion matrices and accuracy metrics when the training pixels are not selected are presented in Tables C.24 and C.25.



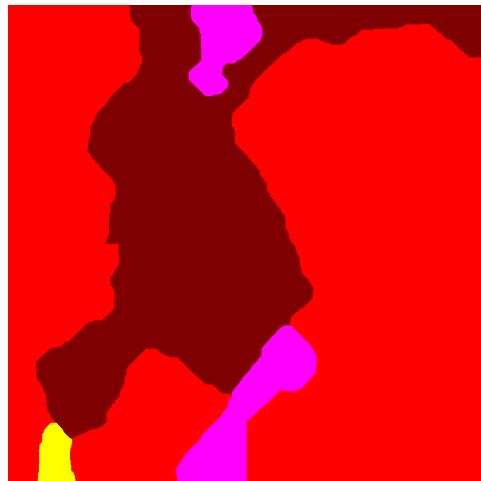
(A) Classification with selection of training pixels (overall accuracy: 93.14%,  $\kappa$ : 0.86).



(B) Regularization with selection of training pixels (overall accuracy: 97.44%,  $\kappa$ : 0.95).



(C) Classification without selection of training pixels (overall accuracy: 78.98%,  $\kappa$ : 0.62).



(D) Regularization without selection of training pixels (overall accuracy: 94.67%,  $\kappa$ : 0.89).

FIGURE 4.17: Selection of the training pixels.

The selection of training pixels greatly increases the classification results. Indeed, without any selection, the classification is very noisy (even when working at the object level) and many confusion are reported. The regularization attenuates the errors, but the result remains worse than the one obtained with the selection of training pixels.

#### **4.3.4 Regularization**

The smoothing of the classification can be performed using local or global methods.

##### **Local methods**

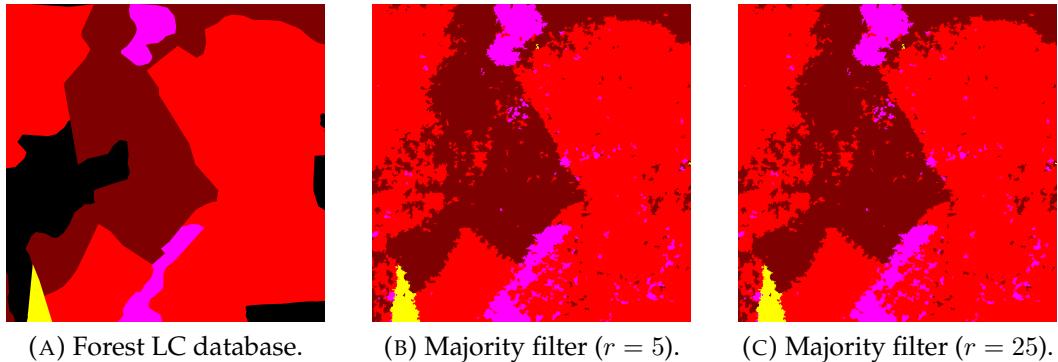
Two methods have been investigated for the local smoothing of the classification. They are very easy to implement and have low computation times. Firstly, the use of a majority filter have been employed. Since it does not take into account the probabilities, a probabilistic relaxation have then also been tested. For both method, the main problem is to defined a relevant neighborhood size. If it is too small, the results will remain noisy, and if it is too important, the results will be over generalized and the computational times will explode.

##### **Majority filter.**

The results for the majority filter is presented in Figure 4.18. The filtering method performs the worse with a gain of less than 1% compared to the classification, even with large filters. Furthermore, the larger the filter is, the longer are the computation times.

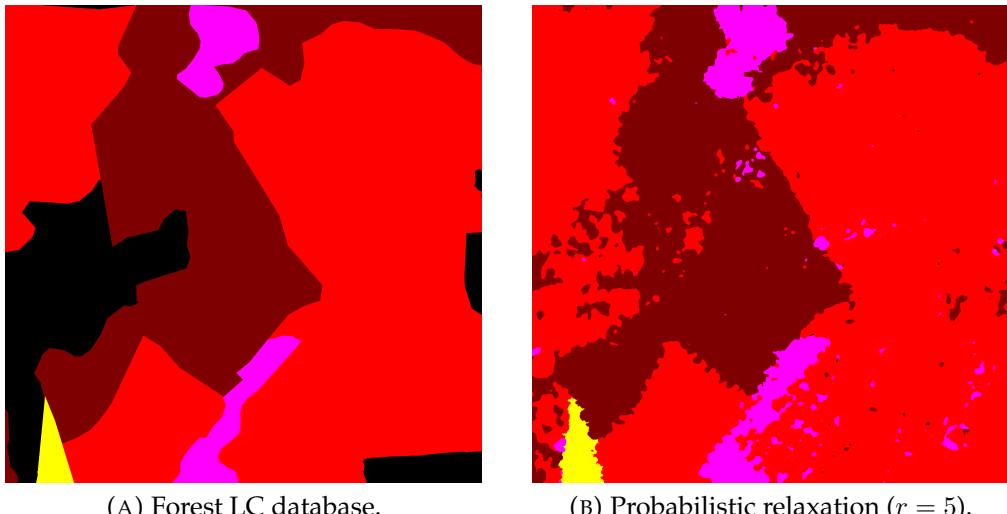
##### **Probabilistic relaxation.**

The results for the probabilistic relaxation is presented in Figure 4.19. The probabilistic relaxation has also poor results (+5% than the classification) and has also important computation times, since the iterative process runs until the convergence has been reached.



(A) Forest LC database. (B) Majority filter ( $r = 5$ ). (C) Majority filter ( $r = 25$ ).

FIGURE 4.18: Results of the majority filters.



(A) Forest LC database. (B) Probabilistic relaxation ( $r = 5$ ).

FIGURE 4.19: Results of the probabilistic relaxation.

## Global methods

The regularization is the final step that smooth the former classification. In the formulation of the energy, 3 aspect are taken into account. The two most important are the integration of the information from the classification (unary term) and how the feature are integrated into the smoothing process (prior). The last aspect is the integration of other constraints, allowed by the use of the QPBO algorithm.

The global methods produce significantly better results than the local methods.

The effect of the parameter  $\gamma$  is presented in Figure 4.20. When  $\gamma$  is low, the borders are rough and small regions might appear (Figure 4.20c). Increasing  $\gamma$  smooth the borders, however, a too high value have a negative impact on the results, reducing the size of meaningful segments (Figure 4.20e) or even removing them (Figure 4.20f). The tuning of the parameter  $\gamma$  is an important issue, since different values of  $\gamma$  might be acceptable depending on the level of detail expected for the segmentation. In forest inventory, having small regions of pure species is interesting for the understanding of the behavior of the forest. For generalization purposes (such as forest LC), the segments must have a decent size and may exhibit variability.

### **Unary term.**

The unary term have the major impact on the regularization results (see Figure 4.21). Indeed, the log-inverse formulation highly penalize pixels with low probability, thus, small areas with high probabilities will be kept, even with an important  $\gamma$ . Contrarily, the linear formulation have a stronger smoothing effect.

Both are interesting for the analysis of forest. The log-inverse allows to obtain small areas that keep an important class confidence. Such areas are useful for forest inventory since they give information about large forested areas with small islets of pure species. Conversely, the linear formulation produces smooth segments that are more relevant for a national mapping.

### **Prior.**

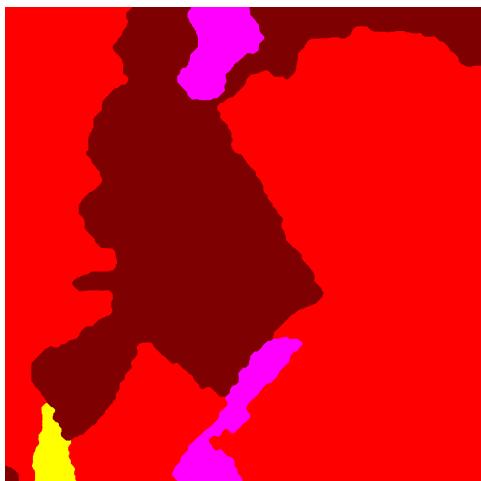
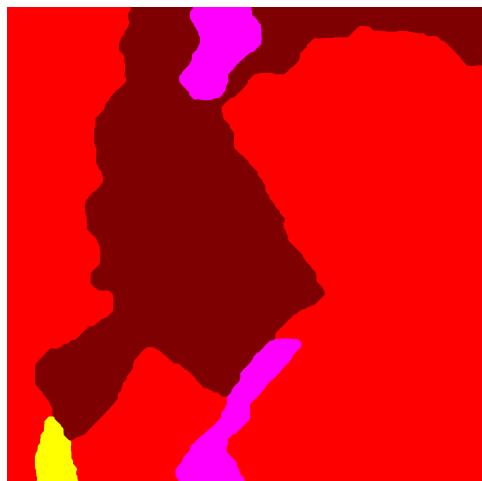
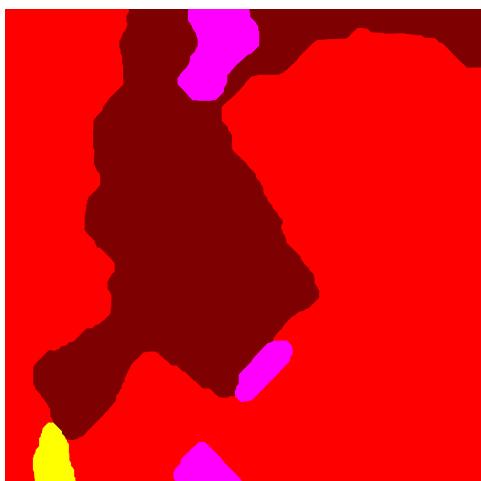
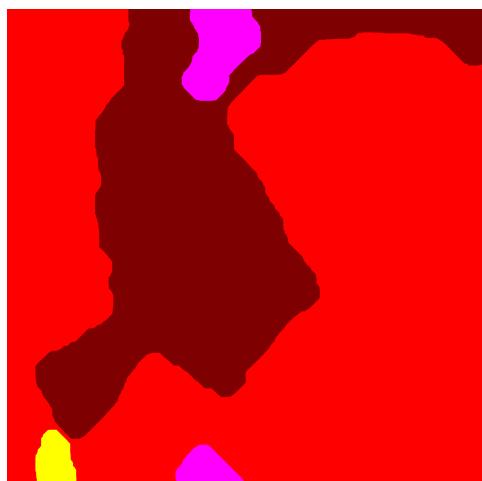
The prior have a weak impact on the regularization results (see Figure 4.22).



(A) IRC VHR optical image.



(B) Forest LC.

(C) Regularization ( $\gamma = 5$ ).(D) Regularization ( $\gamma = 10$ ).(E) Regularization ( $\gamma = 15$ ).(F) Regularization ( $\gamma = 20$ ).FIGURE 4.20: Regularization results, effect of the parameter  $\gamma$ .

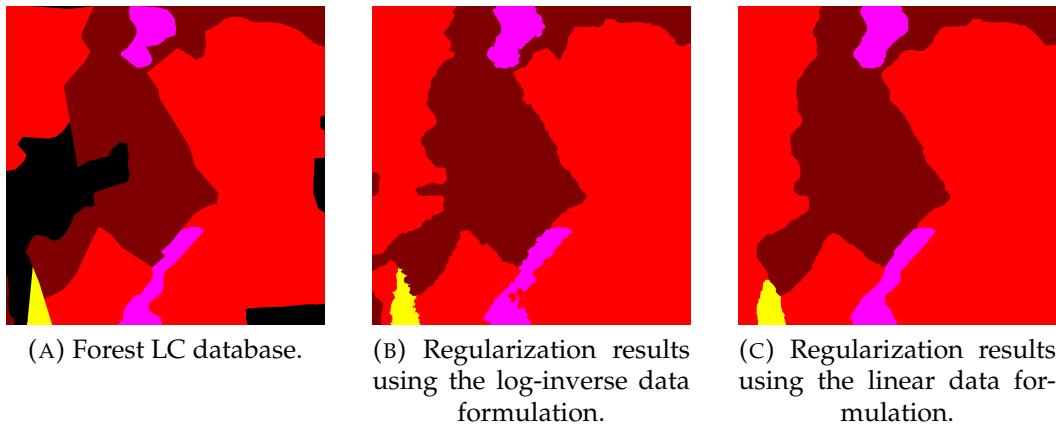


FIGURE 4.21: Impact of the formulation of the unary term.

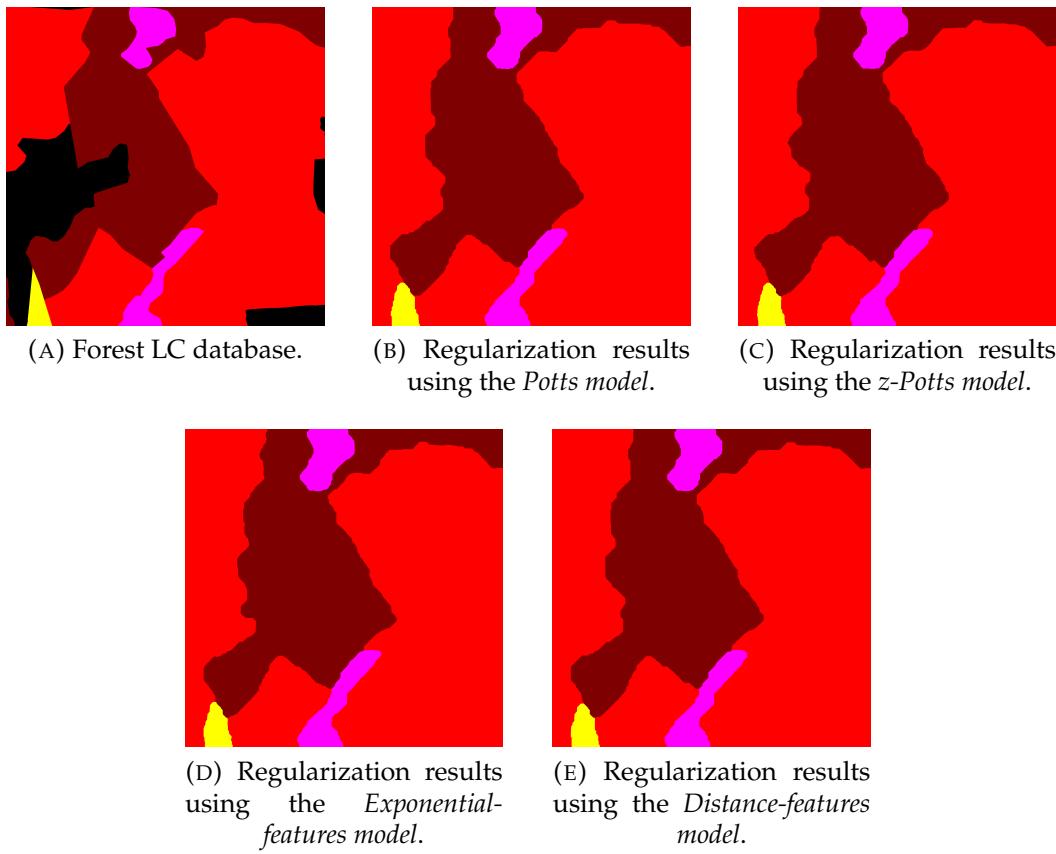


FIGURE 4.22: Impact of the formulation of the data term.

The *z-Potts model* tends to have slightly worse results than the other methods. The *Potts model* and the *Distance-features model* have very close results regardless of the unary term. The *Exponential-features model* have the greatest results with the linear unary, but have poor results with the log-inverse unary.

### Adding constraints.

The addition of constraints could be easily carried out with the QPBO algorithm. We can add strong borders that we want to retrieve in the final segmentation. Unfortunately, such borders can not be easily extracted (see Section 4.2.1). It is then not relevant to add such constraints since they will not lead to accurate results.

A second constraint can be employed in order to ensure a minimal size of the final segments. However, it leads to three issues

1. The minimal size of a segment should be defined. Even if such minimal size is defined by the definition of forest stands (0.5ha), a segment with a size of 0.5ha+1 pixels will not be considered as a small segment.
2. If such constraint wants to be employed, the QPBO algorithm have to run until all the segments have reached the minimum size. Such iteration might be long
3. The final result will be similar to a majority vote. For a small segment, selecting the label of the neighbor segment that have the most neighbor pixels with the small segment will nearly lead to the same results of the regularization using minimal size constraint.

### 4.3.5 Importance of fusion

In this framework, the fusion is performed at multiple levels;

- Choice of data for the over-segmentation,
- Features employed for the classification,
- Features employed for the regularization.

The impact of the choice of the data for the over-segmentation have already been investigated in Section 4.3.1. When employing an over-segmentation from VHR optical images (e.g. PFF, Quickshift or SLIC) or from a lidar data (e.g. hierarchical segmentation on nDSM or tree extraction) produces similar results.

Experiments have been conducted in order to determine how the choice of the data impact both the classification and regularization step. In order to evaluate the impact of the data choice on the classification and regularization, 5 scenarii have been investigated:

- A full lidar scenario:
  - Hierarchical segmentation on the nDSM,
  - Object-based classification (with selection of training samples) using the 25 lidar features,
  - Regularization using the 25 lidar features.
- A full spectral scenario:
  - PFF segmentation of the VHR optical image,
  - Object-based classification (with selection of training samples) using a selection of 20 spectral features,
  - Regularization using a selection of 20 spectral features.
- 3 scenarii:
  - PFF segmentation of the VHR optical image,
  - Object-based classification (with selection of training samples) using a selection of 20 features (lidar and spectral),
  - 1. Regularization using the 25 lidar features,
  - 2. Regularization using a selection of 20 spectral features,
  - 3. Regularization using a selection of 20 features (lidar and spectral).

The overall accuracies for these 5 scenarii are presented in Table 4.1.

Scenario	Overall accuracy (%)		
	Classification	Regularization	Gain
Full lidar	74.8	92.2	17.4
Full spectral	79.1	95.2	16.1
Regularization lidar		96.0	8.4
Regularization spectral	87.6	96.1	8.5
Regularization lidar + spectral		96.2	8.6

TABLE 4.1: Classification and stand segmentation accuracies for the 5 scenarii investigated.

The impact of the choice of the data on the classification is obvious; the classification using a single data source performs worse than when using both. Furthermore, the spectral information tends to be more relevant than the lidar information. Such results are coherent with the results of the feature selection since the spectral feature are selected for 61% and for 39% for the lidar.

The impact of the choice of the data on the regularization is less significant. We can first note that the use of a single modality can greatly improve the results from a poor classification (+17.4% for lidar and +16.1% for spectral). However, when the classification is already significant, the improvement is more or less the same. The spectral information is a bit more beneficial than the lidar information for the regularization step. Fusing the two data sources or using only the spectral information in the regularization step does not significantly change the final results.

## 4.4 Co-products of the method

The outputs of the method can be employed for further investigation in order to extract relevant information about forest. Firstly, the forest stands obtained can be employed for the semi-automatic update of the forest LC. Secondly, the information extracted from the original data can be employed for forest inventory.

### 4.4.1 Semi-automatic update

The semi-automatic update of the forest LC can be performed by the joint analysis of the final stand segmentation and the forest LC employed.

A change map can be derived and changes can be prioritized according to their size and shape. Here 3 criteria are employed for an area of change :

- The number of pixels,
- The rectangular bounding box (see Figure 4.23b),
- The circular bounding box (see Figure 4.23c).

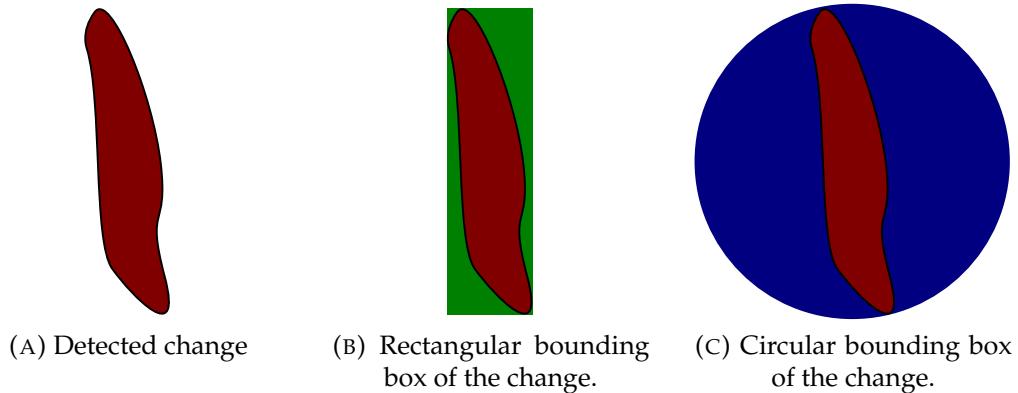


FIGURE 4.23: Bounding boxes of the change detection.

The changes are classified using thresholds based on the size (in pixels) of the change  $s$ , the ratio between the size of the change and the size of the rectangular bounding box  $r_1$  and the ratio between the size of the change and the size of the circular bounding box  $r_2$ . A change is defined as major when the condition (4.2) or (4.3) is verified.

$$s \geq 100 \text{ and } r_1 \geq 0.3, \quad (4.2)$$

$$s \geq 100 \text{ and } r_2 \geq 0.2. \quad (4.3)$$

An "updated" forest LC can be then created, the area of major change are labeled as *no data* (see Figure 4.24). A difference map is also produced with two kinds of differences:

- The minor differences; they correspond to areas where the borders of the Forest LC does not exactly fit with the borders obtained in the method. This type of error is common since the border from the Forest LC are mostly strait lines while the methods tends to follow the natural borders of the forest because of the regularization.

- The major differences; they correspond to large patches that differ from the Forest LC. Two cases can be differentiated:
  - The method can have produced wrong results,
  - The forest has changed or has been exploited (cut or plantation).



(A) Forest LC.



(B) "Updated" Forest LC.

FIGURE 4.24: Automatic update of the Forest LC.

In case of major change, the decision can be taken to keep the original forest LC or to employ the result of the method. An operator can also focus of the change and correct it manually (see Figure 4.25)

#### 4.4.2 Inventory

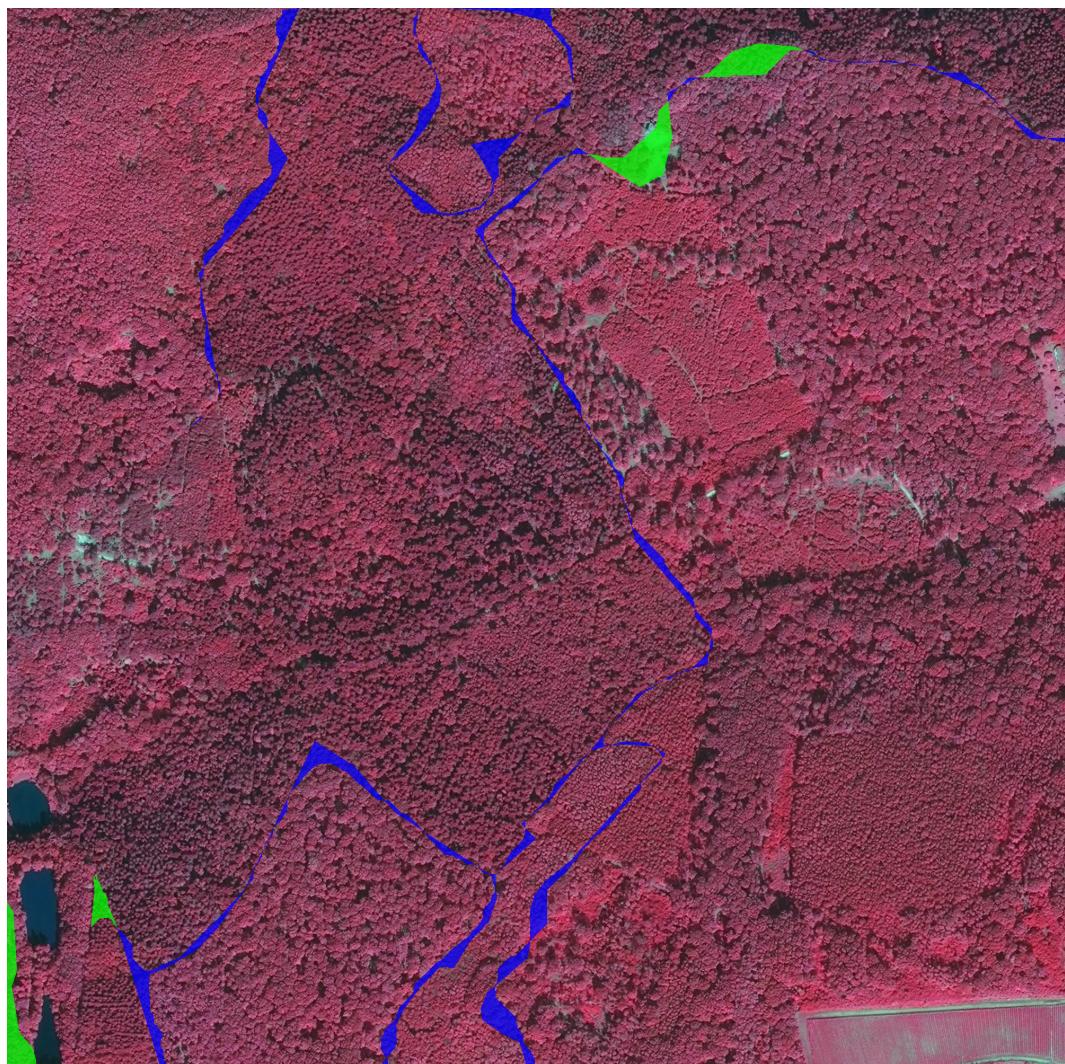


FIGURE 4.25: Detected changes of the Forest LC, green corresponds to major changes, blue corresponds to minor changes.







5

## Conclusions and perspectives







**Color code**

Label	Color	Vegetation type
1	●	<i>Chênes décidus</i>
2	●	<i>Chênes sempervirents</i>
3	●	<i>Hêtre</i>
4	●	<i>Châtaignier</i>
5	●	<i>Robinier</i>
6	●	<i>Autre feuillu pur</i>
7	●	<i>Pin maritime</i>
8	●	<i>Pin sylvestre</i>
9	●	<i>Pin laricio ou pin noir</i>
10	●	<i>Pin d'Alep</i>
11	●	<i>Pin à crochet ou pin cembro</i>
12	●	<i>Autre pin pur</i>
13	●	<i>Sapin ou épicéa</i>
14	●	<i>Mélèze</i>
15	●	<i>Douglas</i>
16	●	<i>Autre conifère pur autre que pin</i>
17	●	<i>Lande ligneuse</i>
18	●	<i>Formation herbacée</i>
19	●	<i>Peupleraie</i>

TABLE A.1: Vegetation type color code





**B**

## Publication

---

B.1	Journal articles . . . . .	104
B.2	Peer-reviewed conference papers . . . . .	104

---

## B.1 Journal articles

C. Dechesne, C. Mallet, A. Le Bris, V. Gouet-Brunet. *Semantic segmentation of forest stands of pure species combining airborne lidar data and very high resolution multispectral imagery.* ISPRS Journal of Photogrammetry and Remote Sensing, 126 (2017), pp.129–145, 2017. 

M. Fauvel, C. Dechesne, A. Zullo, F. Ferraty. *Fast forward feature selection of hyperspectral images for classification with gaussian mixture models.* IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 8(6), pp. 2824-2831, 2015. 

## B.2 Peer-reviewed conference papers

C. Dechesne, C. Mallet, A. Le Bris, V. Gouet-Brunet. *How to combine LIDAR and very high resolution multispectral images for forest stand segmentation?* Proc. of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, USA, July 2017. 

C. Dechesne, C. Mallet, A. Le Bris, V. Gouet-Brunet. *Semantic segmentation of forest stands of pure species as a global optimisation problem.* ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2017. 

C. Dechesne, C. Mallet, A. Le Bris, V. Gouet-Brunet. *Segmentation séquentielle de données de télédétection multimodale : application aux peuplements forestiers.* ORASIS, Colleville-sur-Mer, France, Juin 2017. 

C. Dechesne, C. Mallet, A. Le Bris, V. Gouet-Brunet, A. Hervieu. *Forest stand segmentation using airborne Lidar data and very high resolution multispectral imagery.* International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 41 (B3), pp 207-214 , ISPRS Congress, Prague, Juillet 2016. 







## Results - Confusion matrices and accuracy metrics

C.1	Accuracy metrics . . . . .	108
C.1.1	Metrics at the local level . . . . .	109
C.1.2	Metrics at the global level . . . . .	111
C.2	Segmentation methods . . . . .	112
C.2.1	VHR optical images . . . . .	113
C.2.2	nDSM . . . . .	114
C.3	Results of the method . . . . .	115
C.3.1	Over-segmentation . . . . .	115
C.3.2	Classification . . . . .	121
C.3.3	Regularization . . . . .	125
C.3.4	Local methods . . . . .	125
	Global methods . . . . .	127

## C.1 Accuracy metrics

In this section, the different accuracy metrics presented in the following tables are detailed. Most are standard metrics for classification evaluation.

They are based on the confusion matrix. The metrics can be computed at the global level, or for each class. In this section, the metrics are defined using the  $r$ -class confusion matrix presented in Table C.1. In order to compute the metrics at the local level a confusion matrix can be derived. In this section such confusion matrix is presented in Table C.2.

Confusion matrix					
Classes	1	2	...	$r$	Total
1	$n_{11}$	$n_{12}$	...	$n_{1r}$	$n_{1.}$
2	$n_{21}$	$n_{22}$	...	$n_{2r}$	$n_{2.}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$r$	$n_{r1}$	$n_{r2}$	...	$n_{rr}$	$n_{r.}$
Total	$n_{.1}$	$n_{.2}$	...	$n_{.r}$	$n$

TABLE C.1: Confusion Matrix.

Confusion matrix					
	$i$	$\neq i$	Total		
$i$	TP	FP	Pp		
$\neq i$	FN	TN	Np		
Total	P	N	n		

TABLE C.2: Confusion Matrix at for the local level of the class  $i$ .

The relation between the elements of Table C.1 and Table C.2 are the following:

$$\begin{aligned}
 TP &= n_{ii} & FP &= n_{.i} - n_{ii} & Pp &= n_{.i} \\
 FN &= n_{.i} - n_{ii} & TN &= n - n_{.i} - n_{ii} + n_{ii} & Np &= n - n_{.i} \\
 P &= n_{.i} & N &= n - n_{.i}
 \end{aligned} \tag{C.1}$$

### C.1.1 Metrics at the local level

#### Precision

For the class  $i \in [1, r]$ , the precision (or producer's accuracy)  $p_i$  is defined as follow:

$$p_i = \frac{n_{ii}}{n_{\cdot i}} = \frac{\text{TP}}{\text{Pp}}. \quad (\text{C.2})$$

It is the accuracy from the point of view of the map maker (the producer). This is how often are real features on the ground correctly shown on the classified map or the probability that a certain land cover of an area on the ground is classified as such. The Producer's Accuracy is complement of the Omission Error, Producer's Accuracy = 100%-Omission Error. It is also the number of reference sites classified accurately divided by the total number of reference sites for that class

#### Recall

For the class  $i \in [1, r]$ , the recall (or user's accuracy)  $r_i$  is defined as follow:

$$r_i = \frac{n_{ii}}{n_{\cdot i}} = \frac{\text{TP}}{\text{P}}. \quad (\text{C.3})$$

It is the accuracy from the point of view of a map user, not the map maker. The User's accuracy essentially tells use how often the class on the map will actually be present on the ground. This is referred to as reliability. The User's Accuracy is complement of the Commission Error, User's Accuracy = 100%-Commission Error. When a class is not represented in the classification map, the recall can not be computed.

#### Intersection over Union

The Intersection over Union (or Jaccard index) (Jaccard, 1912) measures similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets. For a class  $i$ , the Intersection over Union ( $IoU_i$ ) is defined as follow:

$$IoU_i = \frac{n_{ii}}{n_{\cdot i} + n_{\cdot i} - n_{ii}} = \frac{\text{TP}}{\text{Pp} + \text{P} - \text{TP}} \quad (\text{C.4})$$

### F-score

It considers both the precision  $p$  and the recall  $r$  of the test to compute the score. The F-score can be interpreted as a weighted average of the precision and recall, where an F-score reaches its best value at 1 and worst at 0. The F-score ( $F_1$ ) of the class  $i$  is defined as follow

$$F_{1,i} = 2 \frac{p_i r_i}{p_i + r_i} \quad (\text{C.5})$$

### Accuracy

The accuracy ( $A_i$ ) (or relative observed agreement among raters) of the class  $i$  is computed as follow:

$$A_i = \frac{\text{TP} + \text{TN}}{n} \quad (\text{C.6})$$

### Kappa coefficient

The Kappa coefficient (Cohen, 1960) ( $\kappa_i$ ) is generated from a statistical test to evaluate the accuracy of a classification. Kappa essentially evaluate how well the classification performed as compared to just randomly assigning values (i.e. did the classification do better than random.) The Kappa Coefficient can range from -1 to 1. A value of 0 indicated that the classification is no better than a random classification. A negative number indicates the classification is significantly worse than random. A value close to 1 indicates that the classification is significantly better than random. The Kappa coefficient is computed as follow:

$$\kappa_i = \frac{P_{0,i} - P_{e,i}}{1 - P_{e,i}}, \quad (\text{C.7})$$

where  $P_0$  is the relative observed agreement among raters, and  $P_e$  is the hypothetical probability of chance agreement. They are defined as follow:

$$P_{0,i} = \frac{\text{TP} + \text{TN}}{n} \quad (\text{C.8})$$

$$P_{e,i} = \frac{P \times P_p + N \times N_p}{n^2} \quad (\text{C.9})$$

### C.1.2 Metrics at the global level

#### Intersection over Union

The overall Intersection over Union score ( $IoU$ ) is defined as the mean of the local Intersection over Union scores:

$$IoU = \frac{1}{r} \sum_{i=1}^r IoU_i \quad (\text{C.10})$$

#### F-score

The overall F-score ( $F_1$ ) is defined as the mean of the local F-scores. If a  $F_{1,i}$  can not be computed, it is considered as zero.

$$F_1 = \frac{1}{r} \sum_{i=1}^r F_{1,i} \quad (\text{C.11})$$

#### Overall Accuracy

The overall accuracy ( $OA$ ) is defined as follow:

$$OA = \frac{1}{n} \sum_{i=1}^r n_{ii} \quad (\text{C.12})$$

#### Kappa coefficient

The Kappa coefficient is computed as follow:

$$\kappa = \frac{P_0 - P_e}{1 - P_e}, \quad (\text{C.13})$$

where  $P_0$  is the relative observed agreement among raters, and  $P_e$  is the hypothetical probability of chance agreement. They are defined as follow:

$$P_0 = \frac{1}{n} \sum_{i=1}^r n_{ii} \quad (\text{C.14})$$

$$P_e = \frac{1}{n^2} \sum_{i=1}^r n_{i\cdot} n_{\cdot i} \quad (\text{C.15})$$

## C.2 Segmentation methods

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	1783941	13837	204918	219306	80.29
2 (4)	63	29545	62	285	98.63
3 (5)	6482	726	145449	9054	89.94
4 (13)	113114	19865	52105	907266	83.06
Recall	93.71	46.18	36.13	79.87	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	76.18	45.89	34.73	68.68	56.37
F-score	86.48	62.91	51.56	81.43	70.59
Accuracy	84.09	99.01	92.2	88.2	81.75
P0	0.84	0.99	0.92	0.88	0.82
Pe	0.51	0.97	0.85	0.57	0.45
$\kappa$	0.6744	0.6247	0.4814	0.7279	0.6679

TABLE C.3: Confusion Matrix and accuracy metrics of the classification (object-based hierarchical).

### C.2.1 VHR optical images

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	1869715	0	6022	346265	84.15
2 (4)	8142	0	0	21813	0
3 (5)	82695	0	44484	34532	27.51
4 (13)	108307	0	25382	958661	87.76
Recall	90.37	-	58.62	70.42	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	77.22	0	23.03	64.13	41.1
F-score	87.15	-	37.44	78.14	50.68
Accuracy	84.27	99.15	95.76	84.7	81.94
P0	0.84	0.99	0.96	0.85	0.82
Pe	0.52	0.99	0.93	0.54	0.5
$\kappa$	0.6695	0	0.3555	0.6659	0.6417

TABLE C.4: Confusion Matrix and accuracy metrics when adding semantic information to a direct hierarchical segmentation ( $\mu = 15$ ).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2021033	0	0	200969	90.96
2 (4)	0	0	0	29955	0
3 (5)	83682	0	0	78029	0
4 (13)	589638	0	0	502712	46.02
Recall	75.01	-	-	61.94	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	69.8	0	0	35.87	26.42
F-score	82.22	-	-	52.81	33.76
Accuracy	75.06	99.15	95.39	74.37	71.98
P0	0.75	0.99	0.95	0.74	0.72
Pe	0.57	0.99	0.95	0.6	0.56
$\kappa$	0.4176	0	0	0.3573	0.3644

TABLE C.5: Confusion Matrix and accuracy metrics when adding semantic information to a direct PFF segmentation ( $\sigma = 0.8$ ,  $k = 500$  and  $m = 40000$ ).

### C.2.2 nDSM

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	1526001	0	0	696001	68.68
2 (4)	0	0	0	29955	0
3 (5)	109	0	0	161602	0
4 (13)	105832	0	0	986518	90.31
Recall	93.51	-	-	52.64	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	65.55	0	0	49.83	28.84
F-score	79.19	-	-	66.51	36.43
Accuracy	77.13	99.15	95.39	71.67	71.66
P0	0.77	0.99	0.95	0.72	0.72
Pe	0.49	0.99	0.95	0.49	0.46
$\kappa$	0.5508	0	0	0.4477	0.4737

TABLE C.6: Confusion Matrix and accuracy metrics when adding semantic information to a direct hierarchical segmentation ( $\mu = 15$ ).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	1788281	0	59707	374014	80.48
2 (4)	5687	0	0	24268	0
3 (5)	2510	0	44950	114251	27.8
4 (13)	244694	0	2	847654	77.6
Recall	87.61	-	42.95	62.32	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	72.26	0	20.3	52.82	36.34
F-score	83.89	-	33.75	69.12	46.69
Accuracy	80.42	99.15	94.97	78.4	76.47
P0	0.8	0.99	0.95	0.78	0.76
Pe	0.52	0.99	0.93	0.54	0.49
$\kappa$	0.5903	0	0.3126	0.5282	0.5374

TABLE C.7: Confusion Matrix and accuracy metrics when adding semantic information to a direct PFF segmentation ( $\sigma = 0.8$ ,  $k = 500$  and  $m = 40000$ ).

## C.3 Results of the method

### C.3.1 Over-segmentation

Trees.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2020114	2556	113280	86058	90.91
2 (4)	335	28093	354	1173	93.78
3 (5)	2569	0	143267	15875	88.59
4 (13)	36915	2886	11342	1041207	95.32
Recall	98.07	83.77	53.41	90.99	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	89.31	79.37	49.97	87.1	76.44
F-score	94.36	88.5	66.64	93.1	85.65
Accuracy	93.11	99.79	95.91	95.6	92.2
P0	0.93	1	0.96	0.96	0.92
Pe	0.52	0.98	0.88	0.57	0.48
$\kappa$	0.8553	0.8839	0.6461	0.8988	0.8507

TABLE C.8: Confusion Matrix and accuracy metrics of the classification (Tree extraction).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2168270	2468	22793	28477	97.58
2 (4)	684	29235	0	36	97.6
3 (5)	86	0	151836	9789	93.89
4 (13)	13407	2457	5586	1070900	98.04
Recall	99.35	85.58	84.25	96.55	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.96	83.82	79.88	94.72	88.84
F-score	98.46	91.2	88.81	97.29	93.94
Accuracy	98.06	99.84	98.91	98.3	97.55
P0	0.98	1	0.99	0.98	0.98
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9585	0.9111	0.8824	0.9604	0.9515

TABLE C.9: Confusion Matrix and accuracy metrics of the regularization (Tree extraction).

**Watershed.**

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	1803817	11010	197844	209337	81.18
2 (4)	41	27438	175	2301	91.6
3 (5)	8018	1115	140922	11656	87.14
4 (13)	72314	15007	39069	965960	88.43
Recall	95.73	50.28	37.28	81.22	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	78.35	48.06	35.34	73.42	58.79
F-score	87.86	64.92	52.22	84.67	72.42
Accuracy	85.78	99.15	92.64	90.03	83.8
P0	0.86	0.99	0.93	0.9	0.84
Pe	0.51	0.98	0.86	0.56	0.45
$\kappa$	0.7098	0.6453	0.4892	0.773	0.7048

TABLE C.10: Confusion Matrix and accuracy metrics of the classification (Watershed).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2078028	977	110274	32729	93.52
2 (4)	317	26414	0	3224	88.18
3 (5)	443	0	153631	7637	95
4 (13)	42882	1009	3336	1045123	95.68
Recall	97.94	93.01	57.49	96	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	91.72	82.7	55.8	92.01	80.56
F-score	95.68	90.53	71.63	95.84	88.42
Accuracy	94.65	99.84	96.53	97.41	94.21
P0	0.95	1	0.97	0.97	0.94
Pe	0.53	0.98	0.88	0.57	0.48
$\kappa$	0.8866	0.9045	0.699	0.9396	0.8879

TABLE C.11: Confusion Matrix and accuracy metrics of the regularization (Watershed over-segmentation).

### Hierarchical segmentation.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	1906809	2069	136358	176772	85.81
2 (4)	317	26775	141	2722	89.38
3 (5)	2540	0	154347	4824	95.45
4 (13)	57939	4404	31601	998406	91.4
Recall	96.91	80.53	47.87	84.42	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	83.53	73.5	46.8	78.2	70.51
F-score	91.03	84.73	63.76	87.77	81.82
Accuracy	89.28	99.72	95	92.06	88.03
P0	0.89	1	0.95	0.92	0.88
Pe	0.52	0.98	0.87	0.56	0.47
$\kappa$	0.7783	0.8459	0.6139	0.8191	0.7762

TABLE C.12: Confusion Matrix and accuracy metrics of the classification (Hierarchical).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2164661	1350	27069	28928	97.42
2 (4)	1019	27568	0	1368	92.03
3 (5)	373	0	159813	1525	98.83
4 (13)	35484	612	5944	1050310	96.15
Recall	98.32	93.36	82.88	97.06	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	95.83	86.37	82.07	93.43	89.43
F-score	97.87	92.69	90.15	96.6	94.33
Accuracy	97.31	99.88	99	97.89	97.04
P0	0.97	1	0.99	0.98	0.97
Pe	0.53	0.98	0.9	0.57	0.5
$\kappa$	0.9423	0.9263	0.8963	0.9508	0.9412

TABLE C.13: Confusion Matrix and accuracy metrics of the regularization (Hierarchical over-segmentation).

## PFF.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2072223	2635	66560	80590	93.26
2 (4)	200	29345	119	291	97.96
3 (5)	12398	0	146668	2645	90.7
4 (13)	51976	2751	20296	1017327	93.13
Recall	96.98	84.49	62.77	92.41	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	90.63	83.03	58.98	86.52	79.79
F-score	95.08	90.73	74.2	92.77	88.19
Accuracy	93.89	99.83	97.09	95.48	93.14
P0	0.94	1	0.97	0.95	0.93
Pe	0.53	0.98	0.89	0.57	0.49
$\kappa$	0.8701	0.9064	0.7271	0.8948	0.8662

TABLE C.14: Confusion Matrix and accuracy metrics of the classification (PFF).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2174574	3039	14711	29684	97.87
2 (4)	1643	28105	0	207	93.82
3 (5)	2407	0	158293	1011	97.89
4 (13)	23327	1550	12265	1055208	96.6
Recall	98.76	85.96	85.44	97.15	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.67	81.36	83.89	93.94	88.97
F-score	98.31	89.72	91.24	96.88	94.04
Accuracy	97.87	99.82	99.13	98.06	97.44
P0	0.98	1	0.99	0.98	0.97
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9542	0.8963	0.9079	0.9547	0.949

TABLE C.15: Confusion Matrix and accuracy metrics of the regularization (PFF over-segmentation).

**Quickshift.**

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2010751	1970	85980	123307	90.49
2 (4)	441	28932	420	162	96.58
3 (5)	8280	0	137651	15780	85.12
4 (13)	57765	4237	16873	1013475	92.78
Recall	96.8	82.34	57.13	87.92	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	87.86	80.01	51.95	82.29	75.53
F-score	93.54	88.89	68.38	90.28	85.27
Accuracy	92.08	99.79	96.37	93.78	91.01
P0	0.92	1	0.96	0.94	0.91
Pe	0.52	0.98	0.89	0.56	0.48
$\kappa$	0.8333	0.8879	0.6653	0.8571	0.8267

TABLE C.16: Confusion Matrix and accuracy metrics of the classification (Quickshift).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2172018	1463	19015	29512	97.75
2 (4)	536	28364	0	1055	94.69
3 (5)	427	0	147858	13426	91.43
4 (13)	23755	1499	4103	1062993	97.31
Recall	98.87	90.54	86.48	96.03	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.67	86.17	80	93.55	89.1
F-score	98.31	92.57	88.89	96.66	94.11
Accuracy	97.87	99.87	98.95	97.91	97.3
P0	0.98	1	0.99	0.98	0.97
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9543	0.925	0.8833	0.9514	0.9462

TABLE C.17: Confusion Matrix and accuracy metrics of the regularization (Quickshift over-segmentation).

### SLIC.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	1999280	2337	61150	159241	89.98
2 (4)	398	26354	653	2550	87.98
3 (5)	4645	0	152316	4750	94.19
4 (13)	44996	2776	18514	1026064	93.93
Recall	97.56	83.75	65.47	86.04	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	87.99	75.15	62.93	81.51	76.9
F-score	93.61	85.81	77.25	89.81	86.62
Accuracy	92.22	99.75	97.44	93.36	91.39
P0	0.92	1	0.97	0.93	0.91
Pe	0.52	0.98	0.89	0.56	0.48
$\kappa$	0.837	0.8569	0.7594	0.849	0.8345

TABLE C.18: Confusion Matrix and accuracy metrics of the classification (SLIC).

---

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2179198	2066	15072	25672	98.07
2 (4)	33	28139	0	1783	93.94
3 (5)	911	0	158212	2588	97.84
4 (13)	16802	1696	6974	1066878	97.67
Recall	99.19	88.21	87.77	97.26	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	97.3	83.46	86.1	95.05	90.48
F-score	98.63	90.98	92.53	97.46	94.9
Accuracy	98.27	99.84	99.27	98.42	97.9
P0	0.98	1	0.99	0.98	0.98
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9629	0.909	0.9215	0.9631	0.9583

TABLE C.19: Confusion Matrix and accuracy metrics of the regularization (SLIC over-segmentation).

### C.3.2 Classification

**Pixel-based classification versus object-based classification.**

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	1573764	74079	323206	250959	70.83
2 (4)	568	23026	1642	4719	76.87
3 (5)	18420	7765	119886	15640	74.14
4 (13)	124530	134448	79061	754311	69.05
Recall	91.64	9.622	22.89	73.55	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	66.53	9.351	21.2	55.31	38.1
F-score	79.9	17.1	34.98	71.23	50.8
Accuracy	77.42	93.63	87.29	82.62	70.48
P0	0.77	0.94	0.87	0.83	0.7
Pe	0.5	0.92	0.82	0.58	0.41
$\kappa$	0.5508	0.1582	0.3005	0.588	0.5004

TABLE C.20: Confusion Matrix and accuracy metrics of the pixel-based classification.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2072223	2635	66560	80590	93.26
2 (4)	200	29345	119	291	97.96
3 (5)	12398	0	146668	2645	90.7
4 (13)	51976	2751	20296	1017327	93.13
Recall	96.98	84.49	62.77	92.41	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	90.63	83.03	58.98	86.52	79.79
F-score	95.08	90.73	74.2	92.77	88.19
Accuracy	93.89	99.83	97.09	95.48	93.14
P0	0.94	1	0.97	0.95	0.93
Pe	0.53	0.98	0.89	0.57	0.49
$\kappa$	0.8701	0.9064	0.7271	0.8948	0.8662

TABLE C.21: Confusion Matrix and accuracy metrics of the object-based classification (PFF).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2032550	112	144846	44500	91.47
2 (4)	469	18581	0	10905	62.03
3 (5)	839	0	130259	30613	80.55
4 (13)	47199	689	2430	1042032	95.39
Recall	97.67	95.87	46.93	92.37	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	89.52	60.41	42.16	88.43	70.13
F-score	94.47	75.32	59.31	93.86	80.74
Accuracy	93.21	99.65	94.9	96.11	91.94
P0	0.93	1	0.95	0.96	0.92
Pe	0.53	0.99	0.88	0.57	0.48
$\kappa$	0.8571	0.7516	0.5679	0.9102	0.845

TABLE C.22: Confusion Matrix and accuracy metrics of the regularization after a pixel-based classification.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2174574	3039	14711	29684	97.87
2 (4)	1643	28105	0	207	93.82
3 (5)	2407	0	158293	1011	97.89
4 (13)	23327	1550	12265	1055208	96.6
Recall	98.76	85.96	85.44	97.15	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.67	81.36	83.89	93.94	88.97
F-score	98.31	89.72	91.24	96.88	94.04
Accuracy	97.87	99.82	99.13	98.06	97.44
P0	0.98	1	0.99	0.98	0.97
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9542	0.8963	0.9079	0.9547	0.949

TABLE C.23: Confusion Matrix and accuracy metrics of the regularization after an object-based classification (PFF).

**Training set design.**

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	1776715	14410	268660	162223	79.96
2 (4)	167	26119	1203	2466	87.19
3 (5)	10505	2383	135761	13062	83.95
4 (13)	119353	84627	57826	830544	76.03
Recall	93.18	20.48	29.29	82.37	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	75.54	19.88	27.74	65.39	47.14
F-score	86.07	33.17	43.43	79.08	60.44
Accuracy	83.59	97	89.91	87.46	78.98
P0	0.84	0.97	0.9	0.87	0.79
Pe	0.51	0.96	0.83	0.58	0.44
$\kappa$	0.6639	0.3223	0.3928	0.7015	0.6242

TABLE C.24: Confusion Matrix and accuracy metrics of the classification when not selecting training samples.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2118007	1016	57835	45150	95.32
2 (4)	1913	25818	0	2224	86.19
3 (5)	517	0	152774	8420	94.47
4 (13)	63771	1016	4869	1022694	93.62
Recall	96.97	92.7	70.9	94.83	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	92.56	80.71	68.08	89.07	82.61
F-score	96.14	89.33	81.01	94.22	90.17
Accuracy	95.15	99.82	97.96	96.42	94.67
P0	0.95	1	0.98	0.96	0.95
Pe	0.53	0.98	0.9	0.57	0.49
$\kappa$	0.8961	0.8924	0.7995	0.9163	0.8948

TABLE C.25: Confusion Matrix and accuracy metrics of the regularization when not selecting training samples.

### C.3.3 Regularization

### C.3.4 Local methods

**Majority filter.**

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2083695	2660	61515	74138	93.78
2 (4)	129	29478	86	262	98.41
3 (5)	12020	0	147361	2330	91.13
4 (13)	47586	2529	19812	1022423	93.6
Recall	97.21	85.03	64.41	93.02	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	91.32	83.88	60.61	87.46	80.82
F-score	95.46	91.23	75.48	93.31	88.87
Accuracy	94.35	99.84	97.27	95.82	93.64
P0	0.94	1	0.97	0.96	0.94
Pe	0.53	0.98	0.89	0.57	0.49
$\kappa$	0.8799	0.9115	0.7407	0.9027	0.8757

TABLE C.26: Confusion Matrix and accuracy metrics after the majority filtering ( $r = 5$ ).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2083643	2660	61569	74136	93.77
2 (4)	129	29478	86	262	98.41
3 (5)	12003	0	147401	2307	91.15
4 (13)	47040	2526	19802	1022982	93.65
Recall	97.24	85.04	64.41	93.02	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	91.34	83.88	60.62	87.51	80.84
F-score	95.47	91.24	75.48	93.34	88.88
Accuracy	94.37	99.84	97.27	95.83	93.65
P0	0.94	1	0.97	0.96	0.94
Pe	0.53	0.98	0.89	0.57	0.49
$\kappa$	0.8802	0.9116	0.7408	0.9031	0.876

TABLE C.27: Confusion Matrix and accuracy metrics after the majority filtering ( $r = 25$ ).

### Probabilistic relaxation.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2122380	2802	43233	53593	95.52
2 (4)	11	29880	0	64	99.75
3 (5)	9862	0	150745	1104	93.22
4 (13)	32943	2299	17229	1039879	95.2
Recall	98.02	85.42	71.37	95	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	93.71	85.24	67.85	90.65	84.36
F-score	96.75	92.03	80.85	95.1	91.18
Accuracy	95.94	99.85	97.96	96.94	95.35
P0	0.96	1	0.98	0.97	0.95
Pe	0.53	0.98	0.9	0.57	0.49
$\kappa$	0.9133	0.9196	0.7979	0.9287	0.9085

TABLE C.28: Confusion Matrix and accuracy metrics after the probabilistic relaxation ( $r = 5$ ).

## Global methods

### Impact of the parameter $\gamma$ .

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2170873	3034	16065	32036	97.7
2 (4)	153	29715	0	87	99.2
3 (5)	6914	0	154199	598	95.35
4 (13)	17452	2019	12837	1060042	97.04
Recall	98.88	85.47	84.22	97.01	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.63	84.88	80.9	94.22	89.16
F-score	98.29	91.82	89.44	97.02	94.14
Accuracy	97.84	99.85	98.96	98.15	97.4
P0	0.98	1	0.99	0.98	0.97
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9537	0.9175	0.889	0.9568	0.9483

TABLE C.29: Confusion Matrix and accuracy metrics of the regularization ( $\gamma = 5$ ).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2174574	3039	14711	29684	97.87
2 (4)	1643	28105	0	207	93.82
3 (5)	2407	0	158293	1011	97.89
4 (13)	23327	1550	12265	1055208	96.6
Recall	98.76	85.96	85.44	97.15	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.67	81.36	83.89	93.94	88.97
F-score	98.31	89.72	91.24	96.88	94.04
Accuracy	97.87	99.82	99.13	98.06	97.44
P0	0.98	1	0.99	0.98	0.97
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9542	0.8963	0.9079	0.9547	0.949

TABLE C.30: Confusion Matrix and accuracy metrics of the regularization ( $\gamma = 10$ ).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2184507	2975	5458	29068	98.31
2 (4)	2514	27213	0	228	90.85
3 (5)	22584	0	137459	1668	85
4 (13)	25778	1754	10980	1053838	96.47
Recall	97.72	85.2	89.32	97.15	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.11	78.46	77.16	93.82	86.39
F-score	98.02	87.93	87.11	96.81	92.47
Accuracy	97.48	99.79	98.84	98.02	97.06
P0	0.97	1	0.99	0.98	0.97
Pe	0.54	0.98	0.91	0.57	0.5
$\kappa$	0.9456	0.8782	0.865	0.9537	0.9409

TABLE C.31: Confusion Matrix and accuracy metrics of the regularization ( $\gamma = 15$ ).

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2181685	2552	3839	33932	98.19
2 (4)	4080	25647	0	228	85.62
3 (5)	51623	0	106115	3973	65.62
4 (13)	27573	899	9543	1054335	96.52
Recall	96.32	88.14	88.8	96.51	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	94.64	76.77	60.6	93.26	81.32
F-score	97.25	86.86	75.47	96.51	89.02
Accuracy	96.47	99.78	98.03	97.83	96.06
P0	0.96	1	0.98	0.98	0.96
Pe	0.54	0.98	0.92	0.57	0.51
$\kappa$	0.9235	0.8675	0.7447	0.9494	0.9198

TABLE C.32: Confusion Matrix and accuracy metrics of the regularization ( $\gamma = 20$ ).

**Unary term.**

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2168246	2951	15896	34915	97.58
2 (4)	26	29861	0	68	99.69
3 (5)	6376	0	154601	734	95.6
4 (13)	14023	2380	12605	1063342	97.34
Recall	99.07	84.85	84.43	96.75	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.69	84.63	81.28	94.26	89.21
F-score	98.32	91.67	89.67	97.05	94.18
Accuracy	97.88	99.85	98.98	98.15	97.43
P0	0.98	1	0.99	0.98	0.97
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9547	0.916	0.8914	0.957	0.9491

TABLE C.33: Confusion Matrix and accuracy metrics of the regularization using the log-inverse data formulation.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2174574	3039	14711	29684	97.87
2 (4)	1643	28105	0	207	93.82
3 (5)	2407	0	158293	1011	97.89
4 (13)	23327	1550	12265	1055208	96.6
Recall	98.76	85.96	85.44	97.15	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.67	81.36	83.89	93.94	88.97
F-score	98.31	89.72	91.24	96.88	94.04
Accuracy	97.87	99.82	99.13	98.06	97.44
P0	0.98	1	0.99	0.98	0.97
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9542	0.8963	0.9079	0.9547	0.949

TABLE C.34: Confusion Matrix and accuracy metrics of the regularization using the linear data formulation.

**Prior.**

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2172746	3032	14419	31811	97.78
2 (4)	1700	27558	0	697	92
3 (5)	2673	0	157955	1083	97.68
4 (13)	24348	1737	11912	1054353	96.52
Recall	98.7	85.25	85.71	96.91	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.54	79.36	84	93.64	88.38
F-score	98.24	88.49	91.3	96.72	93.69
Accuracy	97.78	99.8	99.14	97.96	97.34
P0	0.98	1	0.99	0.98	0.97
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9522	0.8839	0.9085	0.9523	0.947

TABLE C.35: Confusion Matrix and accuracy metrics of the regularization using the *Potts model*.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2174110	3124	13860	30914	97.84
2 (4)	2033	27472	0	450	91.71
3 (5)	2416	0	158306	989	97.89
4 (13)	25814	1653	11583	1053300	96.43
Recall	98.63	85.19	86.15	97.02	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.53	79.1	84.59	93.65	88.47
F-score	98.23	88.33	91.65	96.72	93.73
Accuracy	97.77	99.79	99.18	97.96	97.35
P0	0.98	1	0.99	0.98	0.97
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9521	0.8822	0.9122	0.9524	0.9473

TABLE C.36: Confusion Matrix and accuracy metrics of the regularization using the *z-Potts model*.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2174574	3039	14711	29684	97.87
2 (4)	1643	28105	0	207	93.82
3 (5)	2407	0	158293	1011	97.89
4 (13)	23327	1550	12265	1055208	96.6
Recall	98.76	85.96	85.44	97.15	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.67	81.36	83.89	93.94	88.97
F-score	98.31	89.72	91.24	96.88	94.04
Accuracy	97.87	99.82	99.13	98.06	97.44
P0	0.98	1	0.99	0.98	0.97
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.9542	0.8963	0.9079	0.9547	0.949

TABLE C.37: Confusion Matrix and accuracy metrics of the regularization using the *Exponential-features model*.

Confusion matrix					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Precision
1 (1)	2171863	2951	14706	32488	97.74
2 (4)	2144	27578	0	233	92.06
3 (5)	2633	0	157920	1158	97.66
4 (13)	23487	1544	11976	1055343	96.61
Recall	98.72	85.99	85.55	96.89	

Accuracy metrics					
Classes	1 (1)	2 (4)	3 (5)	4 (13)	Overall
IU	96.52	80.05	83.82	93.71	88.52
F-score	98.23	88.92	91.2	96.75	93.77
Accuracy	97.76	99.8	99.13	97.98	97.34
P0	0.98	1	0.99	0.98	0.97
Pe	0.53	0.98	0.91	0.57	0.5
$\kappa$	0.952	0.8882	0.9075	0.9528	0.9471

TABLE C.38: Confusion Matrix and accuracy metrics of the regularization using the *Distance-features model*.



# Bibliography

- Achanta, Radhakrishna, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk (2012). "SLIC superpixels compared to state-of-the-art superpixel methods". In: *IEEE transactions on pattern analysis and machine intelligence* 34.11, pp. 2274–2282.
- Achard, Frédéric (2009). *Vital forest graphics*. UNEP/Earthprint.
- Albert, L., F. Rottensteiner, and C. Heipke (2016). "Contextual land use classification: How detailed can the class structure be?" In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLI-B4, pp. 11–18.
- Atkinson, PM, MEJ Cutler, and H Lewis (1997). "Mapping sub-pixel proportional land cover with AVHRR imagery". In: *International Journal of Remote Sensing* 18.4, pp. 917–935.
- Baatz, Martin, Gerd Binnig, Peter Eschenbacher, Andreas Melchinger, and Michael Sogtrop (2004). *Method of iterative segmentation of a digital picture*. US Patent 6,832,002.
- Bacour, Cédric, François-Marie Bréon, and Fabienne Maignan (2006). "Normalization of the directional effects in NOAA–AVHRR reflectance measurements for an improved monitoring of vegetation cycles". In: *Remote Sensing of Environment* 102.3, pp. 402–413.
- Bannari, A, D Morin, F Bonn, and AR Huete (1995). "A review of vegetation indices". In: *Remote sensing reviews* 13.1-2, pp. 95–120.
- Battiti, Roberto (1994). "Using mutual information for selecting features in supervised neural net learning". In: *IEEE Transactions on neural networks* 5.4, pp. 537–550.

- Beaudoin, A, T Le Toan, S Goze, E Nezry, A Lopes, E Mougin, CC Hsu, HC Han, JA Kong, and RT Shin (1994). "Retrieval of forest biomass from SAR data". In: *International Journal of Remote Sensing* 15.14, pp. 2777–2796.
- Belgiu, Mariana and Lucian Drăguț (2016). "Random Forest in remote sensing: A review of applications and future directions". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 114, pp. 24–31.
- Bellman, Richard E (2015). *Adaptive control processes: a guided tour*. Princeton university press.
- Boser, Bernhard E, Isabelle M Guyon, and Vladimir N Vapnik (1992). "A training algorithm for optimal margin classifiers". In: *Proceedings of the fifth annual workshop on Computational learning theory*. ACM, pp. 144–152.
- Boykov, Yuri, Olga Veksler, and Ramin Zabih (2001). "Fast approximate energy minimization via graph cuts". In: *IEEE Transactions on pattern analysis and machine intelligence* 23.11, pp. 1222–1239.
- Bradski, Gary and Adrian Kaehler (2008). *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc."
- Brandtberg, Tomas (2007). "Classifying individual tree species under leaf-off and leaf-on conditions using airborne lidar". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 61.5, pp. 325–340.
- Breiman, Leo (2001). "Random forests". In: *Machine learning* 45.1, pp. 5–32.
- Breiman, Leo, Jerome Friedman, Charles J Stone, and Richard A Olshen (1984). *Classification and regression trees*. CRC press.
- Brooks, Thomas M, Russell A Mittermeier, Gustavo AB da Fonseca, Justin Gerlach, Michael Hoffmann, John F Lamoreux, Cristina Goettsch Mittermeier, John D Pilgrim, and Ana SL Rodrigues (2006). "Global biodiversity conservation priorities". In: *science* 313.5783, pp. 58–61.
- Bruzzone, Lorenzo and Sebastiano B Serpico (2000). "A technique for feature selection in multiclass problems". In: *International Journal of Remote Sensing* 21.3, pp. 549–563.

- Camps-Valls, Gustau and Lorenzo Bruzzone (2009). *Kernel methods for remote sensing data analysis*. John Wiley & Sons.
- Cang, Shuang and Hongnian Yu (2012). "Mutual information based input feature selection for classification problems". In: *Decision Support Systems* 54.1, pp. 691–698.
- Chang, Yang-Lang, Kun-Shan Chen, Bormin Huang, Wen-Yen Chang, Jon Atli Benediktsson, and Lena Chang (2011). "A parallel simulated annealing approach to band selection for high-dimensional remote sensing images". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 4.3, pp. 579–590.
- Chen, Jiawen, Sylvain Paris, and Frédo Durand (2007). "Real-time edge-aware image processing with the bilateral grid". In: *ACM Transactions on Graphics (TOG)* 26.3, p. 103.
- Cheriyadat, Anil and L Mann Bruce (2003). "Why principal component analysis is not an appropriate feature extraction method for hyperspectral data". In: *Geoscience and Remote Sensing Symposium, 2003. IGARSS'03. Proceedings. 2003 IEEE International*. Vol. 6. IEEE, pp. 3420–3422.
- Cohen, Jacob (1960). "A coefficient of agreement for nominal scales". In: *Educational and psychological measurement* 20.1, pp. 37–46.
- Comaniciu, Dorin and Peter Meer (2002). "Mean shift: A robust approach toward feature space analysis". In: *IEEE Transactions on pattern analysis and machine intelligence* 24.5, pp. 603–619.
- Dalponte, Michele, Lorenzo Bruzzone, and Damiano Gianelle (2012). "Tree species classification in the Southern Alps based on the fusion of very high geometrical resolution multispectral/hyperspectral images and LiDAR data". In: *Remote Sensing of Environment* 123, pp. 258–270.
- Dalponte, Michele, Hans Ole Ørka, Liviu Theodor Ene, Terje Gobakken, and Erik Næsset (2014). "Tree crown delineation and tree species classification in boreal forests using hyperspectral and ALS data". In: *Remote Sensing of Environment* 140, pp. 306–317.

- Dalponte, Michele, Francesco Reyes, Kaja Kandare, and Damiano Ganelle (2015). "Delineation of Individual Tree Crowns from ALS and Hyper-spectral data: a comparison among four methods". In: *European Journal of Remote Sensing* 48, pp. 365–382.
- De Backer, Steve, Pieter Kempeneers, Walter Debruyn, and Paul Scheunders (2005). "A band selection technique for spectral classification". In: *IEEE geoscience and remote sensing letters* 2.3, pp. 319–323.
- Dean, Jeffrey, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Andrew Senior, Paul Tucker, Ke Yang, Quoc V Le, et al. (2012). "Large scale distributed deep networks". In: *Advances in neural information processing systems*, pp. 1223–1231.
- Dechesne, C, C Mallet, A Le Bris, V Gouet-Brunet, and A Hervieu (2016). "Forest stand segmentation using airborne lidar data and very high resolution multispectral imagery". In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 41-B3, pp. 207–214.
- Dechesne, Clément, Clément Mallet, Arnaud Le Bris, and Valérie Gouet-Brunet (2017). "Semantic segmentation of forest stands of pure species combining airborne lidar data and very high resolution multispectral imagery". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 126, pp. 129–145.
- Demuth, Howard B, Mark H Beale, Orlando De Jess, and Martin T Hagan (2014). *Neural network design*. Martin Hagan.
- Derin, Haluk and Howard Elliott (1987). "Modeling and segmentation of noisy and textured images using Gibbs random fields". In: *IEEE Transactions on pattern analysis and machine intelligence* 1, pp. 39–55.
- Díaz-Uriarte, Ramón and Sara Alvarez De Andres (2006). "Gene selection and classification of microarray data using random forest". In: *BMC bioinformatics* 7.1, p. 3.
- Diedershagen, Oliver, Barbara Koch, and Holger Weinacker (2004). "Automatic segmentation and characterisation of forest stand parameters using airborne lidar data, multispectral and fogis data". In: *International*

- Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 36(8/W2), pp. 208–212.
- Estévez, Pablo A, Michel Tesmer, Claudio A Perez, and Jacek M Zurada (2009). “Normalized mutual information feature selection”. In: *IEEE Transactions on Neural Networks* 20.2, pp. 189–201.
- Eysn, Lothar, Markus Hollaus, Klemens Schadauer, and Norbert Pfeifer (2012). “Forest delineation based on airborne LIDAR data”. In: *Remote Sensing* 4.3, pp. 762–783.
- Fahey, Timothy J, Peter B Woodbury, John J Battles, Christine L Goodale, Steven P Hamburg, Scott V Ollinger, and Christopher W Woodall (2010). “Forest carbon storage: ecology, management, and policy”. In: *Frontiers in Ecology and the Environment* 8.5, pp. 245–252.
- Fauvel, Mathieu, Clément Dechesne, Anthony Zullo, and Frederic Ferraty (2015). “Fast forward feature selection of hyperspectral images for classification with Gaussian mixture models”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 8.6, pp. 2824–2831.
- Felzenszwalb, Pedro F and Daniel P Huttenlocher (2004). “Efficient graph-based image segmentation”. In: *International journal of computer vision* 59.2, pp. 167–181.
- (2006). “Efficient belief propagation for early vision”. In: *International journal of computer vision* 70.1, pp. 41–54.
- Ferraz, António, Sassan Saatchi, Clément Mallet, and Victoria Meyer (2016a). “Lidar detection of individual tree size in tropical forests”. In: *Remote Sensing of Environment* 183, pp. 318–333.
- Ferraz, António, Clément Mallet, and Nesrine Chehata (2016b). “Large-scale road detection in forested mountainous areas using airborne topographic lidar data”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 112, pp. 23 –36.
- Fisher, Ronald A (1936). “The use of multiple measurements in taxonomic problems”. In: *Annals of human genetics* 7.2, pp. 179–188.

- Foody, Giles M (2002). "Status of land cover classification accuracy assessment". In: *Remote sensing of environment* 80.1, pp. 185–201.
- Franklin, SE, RJ Hall, LM Moskal, AJ Maudie, and MB Lavigne (2000). "Incorporating texture into classification of forest species composition from airborne multispectral images". In: *International journal of remote sensing* 21.1, pp. 61–79.
- Fukunaga, Keinosuke and Larry Hostetler (1975). "The estimation of the gradient of a density function, with applications in pattern recognition". In: *IEEE Transactions on information theory* 21.1, pp. 32–40.
- Ghosh, Ashish, Nikhil R Pal, and Sankar K Pal (1991). "Image segmentation using a neural network". In: *Biological Cybernetics* 66.2, pp. 151–158.
- Gislason, Pall Oskar, Jon Atli Benediktsson, and Johannes R Sveinsson (2006). "Random forests for land cover classification". In: *Pattern Recognition Letters* 27.4, pp. 294–300.
- Goldberg, David E (1989). "Genetic algorithms in search, optimization, and machine learning, 1989". In: *Reading: Addison-Wesley*.
- Gong, P. and P.J. Howarth (1989). "Performance analyses of probabilistic relaxation methods for land-cover classification". In: *Remote Sensing of Environment* 30.1, pp. 33–42.
- Gressin, A., C. Mallet, N. Vincent, and N. Paparoditis (2013a). "Updating land cover databases using a single very high resolution satellite image". In: *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences II-3/W2*, pp. 13–18.
- Gressin, Adrien, Clément Mallet, Nicole Vincent, and Nicolas Paparoditis (2013b). "Updating land cover databases using a single very high resolution satellite image". In: *The ISPRS Workshop on Image Sequence Analysis*. Vol. 2. 3, pp. 13–18.
- Gressin, Adrien, Nicole Vincent, Clément Mallet, and Nicolas Paparoditis (2014a). "A unified framework for land-cover database update and enrichment using satellite imagery". In: *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, pp. 5057–5061.

- Gressin, Adrien, Clément Mallet, Nicole Vincent, and Nicolas Paparoditis (2014b). "Updating the new French national land cover database". In: *Geoscience and Remote Sensing Symposium (IGARSS), 2014 IEEE International*. IEEE, pp. 3534–3537.
- Guigues, L., J.-P. Cocquerez, and H. Le Men (2006). "Scale-sets image analysis". In: *IJCV* 68.3, pp. 289–317.
- Guo, Baofeng, Robert I Damper, Steve R Gunn, and James DB Nelson (2008). "A fast separability-based feature-selection method for high-dimensional remotely sensed image classification". In: *Pattern Recognition* 41.5, pp. 1653–1662.
- Habib, Ayman, Mwafag Ghanma, Michel Morgan, and Rami Al-Ruzouq (2005). "Photogrammetric and LiDAR data registration using linear features". In: *Photogrammetric Engineering & Remote Sensing* 71.6, pp. 699–707.
- Hansen, FR and Howard Elliott (1982). "Image segmentation using simple Markov field models". In: *Computer Graphics and Image Processing* 20.2, pp. 101–132.
- Hepner, Georgef, Thomas Logan, Niles Ritter, and Nevin Bryant (1990). "Artificial neural network classification using a minimal training set- Comparison to conventional supervised classification". In: *Photogrammetric Engineering and Remote Sensing* 56.4, pp. 469–473.
- Hernando, A, Dirk Tiede, Florian Albrecht, and Stefan Lang (2012). "Spatial and thematic assessment of object-based forest stand delineation using an OFA-matrix". In: *International Journal of Applied Earth Observation and Geoinformation* 19, pp. 214–225.
- Herold, Martin, Margaret E Gardner, and Dar A Roberts (2003). "Spectral resolution requirements for mapping urban areas". In: *IEEE Transactions on Geoscience and remote sensing* 41.9, pp. 1907–1919.
- Holmgren, Johan (2004). "Prediction of tree height, basal area and stem volume in forest stands using airborne laser scanning". In: *Scandinavian Journal of Forest Research* 19.6, pp. 543–553.

- Hughes, Gordon (1968). "On the mean accuracy of statistical pattern recognizers". In: *IEEE transactions on information theory* 14.1, pp. 55–63.
- Indyk, Piotr and Rajeev Motwani (1998). "Approximate nearest neighbors: towards removing the curse of dimensionality". In: *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. ACM, pp. 604–613.
- Jaccard, Paul (1912). "The distribution of the flora in the alpine zone." In: *New phytologist* 11.2, pp. 37–50.
- Jakubowski, Marek K, Qinghua Guo, and Maggi Kelly (2013). "Tradeoffs between lidar pulse density and forest measurement accuracy". In: *Remote Sensing of Environment* 130, pp. 245–253.
- John, George H (1997). "Enhancements to the data mining process". PhD thesis. stanford university Ph. D. thesis.
- Jolliffe, Ian (2011). "Principal component analysis". In: *International encyclopedia of statistical science*. Springer, pp. 1094–1096.
- Jordan, Carl F (1969). "Derivation of leaf-area index from quality of light on the forest floor". In: *Ecology* 50.4, pp. 663–666.
- Jutten, Christian and Jeanny Herault (1991). "Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture". In: *Signal processing* 24.1, pp. 1–10.
- Kandare, Kaja, Michele Dalponte, Damiano Ganelle, and Jonathan Cheung-Wai Chan (2014). "A new procedure for identifying single trees in under-story layer using discrete LiDAR data". In: *Geoscience and Remote Sensing Symposium (IGARSS), 2014 IEEE International*. IEEE, pp. 1357–1360.
- Kangas, Annika and Matti Maltamo (2006). *Forest inventory: methodology and applications*. Vol. 10. Springer Science & Business Media.
- Khosravipour, Anahita, Andrew K Skidmore, Martin Isenburg, Tiejun Wang, and Yousif A Hussin (2014). "Generating pit-free canopy height models from airborne lidar". In: *Photogrammetric Engineering & Remote Sensing* 80.9, pp. 863–872.

- Kim, M., M. Madden, and T.A. Warner (2009). "Forest Type Mapping using Object-specific Texture Measures from Multispectral Ikonos Imagery: Segmentation Quality and Image Classification Issues". In: *Photogrammetric Engineering & Remote Sensing* 75.7, pp. 819–829.
- Kirillov, Alexander, Dmitrij Schlesinger, Walter Forkel, Anatoly Zelenin, Shuai Zheng, Philip Torr, and Carsten Rother (2015). "A Generic CNN-CRF Model for Semantic Segmentation". In: *arxiv:1511.05067*.
- Kittler, Josef (1978). "Feature set search algorithms". In: *Pattern recognition and signal processing*.
- Koch, B, Ch Straub, M Dees, Y Wang, and H Weinacker (2009). "Airborne laser data for stand delineation and information extraction". In: *International Journal of Remote Sensing* 30.4, pp. 935–963.
- Kohavi, Ron and George H John (1997). "Wrappers for feature subset selection". In: *Artificial intelligence* 97.1-2, pp. 273–324.
- Kohli, Pushmeet, L'. Ladický, and P. Torr (2009). "Robust Higher Order Potentials for Enforcing Label Consistency". English. In: *International Journal of Computer Vision* 82.3, pp. 302–324. ISSN: 0920-5691.
- Kolmogorov, Vladimir (2006). "Convergent tree-reweighted message passing for energy minimization". In: *IEEE transactions on pattern analysis and machine intelligence* 28.10, pp. 1568–1583.
- Kolmogorov, Vladimir and Ramin Zabih (2004). "What energy functions can be minimized via graph cuts?" In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26.2, pp. 147–159.
- Kolmogorov, Vladimir and Carsten Rother (2007). "Minimizing non-submodular functions with graph cuts-a review". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.7, pp. 1274–1279.
- Kumar, S. and M. Hebert (2006). "Discriminative random fields." In: *International Journal of Computer Vision* 68.2, pp. 179–201.
- Ladický, L'ubor, Chris Russell, Pushmeet Kohli, and Philip Torr (2012). "Inference Methods for CRFs with Co-occurrence Statistics". In: *International Journal of Computer Vision* 103.2, pp. 213–225.

- Landgrebe, David A (2005). *Signal theory methods in multispectral remote sensing*. Vol. 29. John Wiley & Sons.
- Landrieu, Loic and Guillaume Obozinski (2016). "Cut Pursuit: fast algorithms to learn piecewise constant functions". In: *Artificial Intelligence and Statistics*, pp. 1384–1393.
- Larsen, Rasmus (2002). "Decomposition using maximum autocorrelation factors". In: *Journal of Chemometrics* 16.8-10, pp. 427–435.
- Le Moan, Steven, Alamin Mansouri, Yvon Voisin, and Jon Yngve Hardeberg (2011). "A constrained band selection method based on information measures for spectral image color visualization". In: *IEEE Transactions on Geoscience and Remote Sensing* 49.12, pp. 5104–5115.
- Le Toan, Thuy, André Beaudoin, J Riom, and Dominique Guyon (1992). "Relating forest biomass to SAR data". In: *IEEE Transactions on Geoscience and Remote Sensing* 30.2, pp. 403–411.
- Lebart, L, A Morineau, and Mr Piron (1997). "Multidimensional exploratory statistics". In: *París: Ed. Dunod*.
- Leckie, Donald G, François A Gougeon, Nicholas Walsworth, and Dennis Paradine (2003). "Stand delineation and composition estimation using semi-automated individual tree crown analysis". In: *Remote Sensing of Environment* 85.3, pp. 355–369.
- Lee, J., X. Cai, J. Lellmann, M. Dalponte, Y. Malhi, N. Butt, M. Morecroft, C. B. Schönlieb, and D. A. Coomes (2016). "Individual Tree Species Classification From Airborne Multisensor Imagery Using Robust PCA". In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 9.6, pp. 2554–2567. ISSN: 1939-1404.
- Lemprière, TC, PY Bernier, AL Carroll, MD Flannigan, RP Gilsenan, DW McKenney, EH Hogg, JH Pedlar, D Blain, et al. (2008). "The importance of forest sector adaptation to climate change". In: *Nat. Resour. Can., Can. For. Serv., North. For. Cent., Edmonton, AB. Inf. Rep. NOR-X-416E*.
- Leppänen, VJ, T Tokola, M Maltamo, L Mehtätalo, T Pusa, and J Mustonen (2008). "Automatic delineation of forest stands from LIDAR data".

- In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 38(4/C1), pp. 5–8.
- Levinshtein, Alex, Adrian Stere, Kiriakos N Kutulakos, David J Fleet, Sven J Dickinson, and Kaleem Siddiqi (2009). “Turbopixels: Fast superpixels using geometric flows”. In: *IEEE transactions on pattern analysis and machine intelligence* 31.12, pp. 2290–2297.
- Li, Jili, Baoxin Hu, and Thomas L. Noland (2013). “Classification of tree species based on structural features derived from high density LiDAR data”. In: *Agricultural and Forest Meteorology* 171–172, pp. 104 –114.
- Li, Shijin, Hao Wu, Dingsheng Wan, and Jiali Zhu (2011). “An effective feature selection method for hyperspectral image classification based on genetic algorithm and support vector machine”. In: *Knowledge-Based Systems* 24.1, pp. 40–48.
- Lim, Kevin, Paul Treitz, Michael Wulder, Benoît St-Onge, and Martin Flood (2003). “LiDAR remote sensing of forest structure”. In: *Progress in physical geography* 27.1, pp. 88–106.
- Liu, Huan and Rudy Setiono (1997). “Feature selection and classification-a probabilistic wrapper approach”. In: *Proceedings of 9th International Conference on Industrial and Engineering Applications of AI and ES*, pp. 419–424.
- Lu, Dengsheng and Qihao Weng (2007). “A survey of image classification methods and techniques for improving classification performance”. In: *International journal of Remote sensing* 28.5, pp. 823–870.
- Lucchi, A., Y. Li, X. Boix, K. Smith, and P. Fua (2011). “Are Spatial and Global Constraints Really Necessary for Segmentation?” In: *Proc. of ICCV*, pp. 9–16.
- Maas, A., F. Rottensteiner, and C. Heipke (2016). “Using label noise robust logistic regression for automated updating of topographic geospatial databases”. In: *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* III-7, pp. 133–140.
- Mallet, Clément, Frédéric Bretar, Michel Roux, Uwe Soergel, and Christian Heipke (2011). “Relevance assessment of full-waveform lidar data for

- urban area classification". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 66.6, S71–S84.
- Marill, Thomas and D Green (1963). "On the effectiveness of receptors in recognition systems". In: *IEEE transactions on Information Theory* 9.1, pp. 11–17.
- Martínez-UsÓMartinez-Uso, Adolfo, Filiberto Pla, José Martínez Sotoca, and Pedro García-Sevilla (2007). "Clustering-based hyperspectral band selection using information measures". In: *IEEE Transactions on Geoscience and Remote Sensing* 45.12, pp. 4158–4171.
- Mastin, Andrew, Jeremy Kepner, and John Fisher (2009). "Automatic registration of LIDAR and optical images of urban scenes". In: *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, pp. 2639–2646.
- Mather, Paul and Brandt Tso (2016). *Classification methods for remotely sensed data*. CRC press.
- Means, Joseph E, Steven A Acker, Brandon J Fitt, Michael Renslow, Lisa Emerson, and Chad J Hendrix (2000). "Predicting forest stand characteristics with airborne scanning lidar". In: *Photogrammetric Engineering & Remote Sensing* 66.11, pp. 1367–1372.
- Moore, Alastair P, Simon JD Prince, Jonathan Warrell, Umar Mohammed, and Graham Jones (2008). "Superpixel lattices". In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, pp. 1–8.
- Mora, Brice, Michael A. Wulder, and Joanne C. White (2010). "Segment-constrained regression tree estimation of forest stand height from very high spatial resolution panchromatic imagery over a boreal environment". In: *Remote Sensing of Environment* 114.11, pp. 2474 –2484.
- Moritz, Philipp, Robert Nishihara, Ion Stoica, and Michael I Jordan (2015). "Sparknet: Training deep networks in spark". In: *arXiv preprint arXiv:1511.06051*.

- Moser, G., S.B. Serpico, and J.A. Benediktsson (2013). "Land-Cover Mapping by Markov Modeling of Spatial Contextual Information in Very-High-Resolution Remote Sensing Images". In: *Proceedings of the IEEE* 101.3, pp. 631–651.
- Nguyen, Anh and Bac Le (2013). "3D point cloud segmentation: A survey". In: *Robotics, Automation and Mechatronics (RAM), 2013 6th IEEE Conference on*. IEEE, pp. 225–230.
- Nitzberg, Mark, David Mumford, and Takahiro Shiota (1993). *Filtering, segmentation and depth*.
- Pal, M (2005). "Random forest classifier for remote sensing classification". In: *International Journal of Remote Sensing* 26.1, pp. 217–222.
- Pal, Nikhil R and Sankar K Pal (1993). "A review on image segmentation techniques". In: *Pattern recognition* 26.9, pp. 1277–1294.
- Pan, Yude, Richard A Birdsey, Oliver L Phillips, and Robert B Jackson (2013). "The structure, distribution, and biomass of the world's forests". In: *Annual Review of Ecology, Evolution, and Systematics* 44, pp. 593–622.
- Paola, JD and RA Schowengerdt (1995). "A review and analysis of backpropagation neural networks for classification of remotely-sensed multi-spectral imagery". In: *International Journal of remote sensing* 16.16, pp. 3033–3058.
- Paris, Sylvain and Frédo Durand (2006). "A fast approximation of the bilateral filter using a signal processing approach". In: *Computer Vision–ECCV 2006*, pp. 568–580.
- Paris, Sylvain, Pierre Kornprobst, Jack Tumblin, Frédo Durand, et al. (2009). "Bilateral filtering: Theory and applications". In: *Foundations and Trends® in Computer Graphics and Vision* 4.1, pp. 1–73.
- Patenaude, Genevieve, Ronald Milne, and Terence P Dawson (2005). "Synthesis of remote sensing approaches for forest carbon estimation: reporting to the Kyoto Protocol". In: *Environmental Science & Policy* 8.2, pp. 161–178.

- Peli, Tamar and David Malah (1982). "A study of edge detection algorithms". In: *Computer graphics and image processing* 20.1, pp. 1–21.
- Pommerening, Arne (2002). "Approaches to quantifying forest structures". In: *Forestry: An International Journal of Forest Research* 75.3, pp. 305–324.
- Pudil, Pavel, Jana Novovičová, and Josef Kittler (1994). "Floating search methods in feature selection". In: *Pattern Recognition Letters* 15.11, pp. 1119–1125.
- Radoux, Julien and Pierre Defourny (2007). "A quantitative assessment of boundaries in automated forest stand delineation using very high resolution imagery". In: *Remote Sensing of Environment* 110.4, pp. 468–475.
- Radoux, Julien, Céline Lamarche, Eric Van Bogaert, Sophie Bontemps, Carsten Brockmann, and Pierre Defourny (2014). "Automated Training Sample Extraction for Global Land Cover Mapping". In: *Remote Sensing* 6.5, p. 3965. ISSN: 2072-4292. DOI: 10.3390/rs6053965.
- Ramage, Michael H, Henry Burridge, Marta Busse-Wicher, George Fereday, Thomas Reynolds, Darshil U Shah, Guanglu Wu, Li Yu, Patrick Fleming, Danielle Densley-Tingley, et al. (2017). "The wood from the trees: The use of timber in construction". In: *Renewable and Sustainable Energy Reviews* 68, pp. 333–359.
- Schindler, Konrad (2012). "An overview and comparison of smooth labeling methods for land-cover classification". In: *IEEE Transactions on Geoscience and Remote Sensing* 50.11, pp. 4534–4545.
- Scholkopf, Bernhard and Alexander J Smola (2001). *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press.
- Schuck, Andreas, Risto Päivinen, Tuomo Hytönen, and Brita Pajari (2002). *Compilation of forestry terms and definitions*. European Forest Institute.
- Serpico, Sebastiano B and Gabriele Moser (2007). "Extraction of spectral channels from hyperspectral images for classification purposes". In: *IEEE transactions on geoscience and remote sensing* 45.2, pp. 484–495.
- Shan, Jie and Charles K Toth (2008). *Topographic laser ranging and scanning: principles and processing*. CRC press.

- Sheikh, Yaser Ajmal, Erum Arif Khan, and Takeo Kanade (2007). "Mode-seeking by medoidshifts". In: *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, pp. 1–8.
- Shi, Jianbo and Jitendra Malik (2000). "Normalized cuts and image segmentation". In: *IEEE Transactions on pattern analysis and machine intelligence* 22.8, pp. 888–905.
- Smith, John Miles and Diane CP Smith (1977). "Database abstractions: aggregation and generalization". In: *ACM Transactions on Database Systems (TODS)* 2.2, pp. 105–133.
- Smith, William H (2012). *Air pollution and forests: interactions between air contaminants and forest ecosystems*. Springer Science & Business Media.
- Sotoca, José Martínez and Filiberto Pla (2010). "Supervised feature selection by clustering using conditional mutual information-based distances". In: *Pattern Recognition* 43.6, pp. 2068–2081.
- Souchon, Jean-Philippe, Christian Thom, Christophe Meynard, and Olivier Martin (2012). "A large format camera system for national mapping purposes". In: *Revue Française de Photogrammétrie et de Télédétection* 200, pp. 48–53.
- Sterrett, Frances S (1994). *Alternative fuels and the environment*. CRC Press.
- Strahler, Alan H (1980). "The use of prior probabilities in maximum likelihood classification of remotely sensed data". In: *Remote sensing of Environment* 10.2, pp. 135–163.
- Strobl, Carolin, Anne-Laure Boulesteix, Achim Zeileis, and Torsten Hothorn (2007). "Bias in random forest variable importance measures: Illustrations, sources and a solution". In: *BMC bioinformatics* 8.1, p. 25.
- Strunk, Jacob, Hailemariam Temesgen, Hans-Erik Andersen, James P Flewelling, and Lisa Madsen (2012). "Effects of lidar pulse density and sample size on a model-assisted approach to estimate forest inventory variables". In: *Canadian Journal of Remote Sensing* 38.05, pp. 644–654.
- Sullivan, Alicia A, Robert J McGaughey, Hans-Erik Andersen, and Peter Schiess (2009). "Object-oriented classification of forest structure from

- light detection and ranging data for stand mapping". In: *Western Journal of Applied Forestry* 24.4, pp. 198–204.
- Tang, Jiliang, Salem Alelyani, and Huan Liu (2014). "Feature selection for classification: A review". In: *Data Classification: Algorithms and Applications*, p. 37.
- Taxt, Torfinn, Patrick J Flynn, and Anil K Jain (1989). "Segmentation of document images". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11.12, pp. 1322–1329.
- Tiede, Dirk, Thomas Blaschke, and Marco Heurich (2004). "Object-based semi automatic mapping of forest stands with Laser scanner and Multi-spectral data". In: *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 36(8/W2), pp. 328–333.
- Tokola, Timo (2015). "Remote Sensing Concepts and Their Applicability in REDD+ Monitoring". In: *Current Forestry Reports* 1.4, pp. 252–260.
- Tomppo, Erkki, Håkan Olsson, Göran Ståhl, Mats Nilsson, Olle Hagner, and Matti Katila (2008). "Combining national forest inventory field plots and remote sensing data for forest databases". In: *Remote Sensing of Environment* 112.5, pp. 1982–1999.
- Torabzadeh, H, R Leiterer, ME Schaepman, and F Morsdorf (2015). "Optimal structural and spectral features for tree species classification using combined airborne laser scanning and hyperspectral data". In: *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*. IEEE. Milano, Italy, pp. 5399–5402.
- Torabzadeh, Hossein, Felix Morsdorf, and Michael E Schaepman (2014). "Fusion of imaging spectroscopy and airborne laser scanning data for characterization of forest ecosystems–A review". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 97, pp. 25–35.
- Trias-Sanz, Roger (2006). "Semi-automatic rural land cover classification from high-resolution remote sensing images". PhD thesis. PhD thesis, Université Paris 5.

- Trias-Sanz, Roger and Didier Boldo (2005). "A high-reliability, high-resolution method for land cover classification into forest and non-forest". In: *Image Analysis*, pp. 417–441.
- Tucker, Compton J (1979). "Red and photographic infrared linear combinations for monitoring vegetation". In: *Remote sensing of Environment* 8.2, pp. 127–150.
- Tuia, Devis, Fabio Pacifici, Mikhail Kanevski, and William J Emery (2009). "Classification of very high spatial resolution imagery using mathematical morphology and support vector machines". In: *IEEE Transactions on Geoscience and Remote Sensing* 47.11, pp. 3866–3879.
- Ullrich, A and M Pfennigbauer (2016). "Linear LIDAR versus Geiger-mode LIDAR: impact on data properties and data quality". In: *SPIE Defense+ Security*. International Society for Optics and Photonics, pp. 983204–983204.
- Vapnik, Vladimir (2013). *The nature of statistical learning theory*. Springer science & business media.
- Vauhkonen, Jari, Ilkka Korpela, Matti Maltamo, and Timo Tokola (2010). "Imputation of single-tree attributes using airborne laser scanning-based height, intensity, and alpha shape metrics". In: *Remote Sensing of Environment* 114.6, pp. 1263–1276.
- Vedaldi, Andrea and Stefano Soatto (2008). "Quick shift and kernel methods for mode seeking". In: *Computer vision–ECCV 2008*, pp. 705–718.
- Véga, Cédric, A Hamrouni, S El Mokhtari, J Morel, J Bock, J-P Renaud, M Bouvier, and Sylvie Durrieu (2014). "PTrees: A point-based approach to forest tree extraction from lidar data". In: *International Journal of Applied Earth Observation and Geoinformation* 33, pp. 98–108.
- Veksler, Olga, Yuri Boykov, and Paria Mehrani (2010). "Superpixels and supervoxels in an energy optimization framework". In: *Computer Vision–ECCV 2010*, pp. 211–224.

- Vincent, Luc and Pierre Soille (1991). "Watersheds in digital spaces: an efficient algorithm based on immersion simulations". In: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 6, pp. 583–598.
- Viterbini, Maurizio, Alberto Adriani, and Guido Di Donfrancesco (1987). "Single photon detection and timing system for a lidar experiment". In: *Review of scientific instruments* 58.10, pp. 1833–1839.
- Vogt, Kristina A, JM Honea, DJ Vogt, T Patel-Weynand, RL Edmonds, R Sigurdardottir, DG Briggs, MG Andreu, et al. (2006). "Global societies and forest legacies creating today's forest landscapes." In: *Forests and society: sustainability and life cycles of forests in human landscapes*, pp. 30–59.
- Volpi, M. and V. Ferrari (2015). "Semantic segmentation of urban scenes by learning local class interactions". In: *Proc. of CVPR Workshops*, pp. 1–9.
- Vosselman, George and Hans-Gerd Maas (2010). *Airborne and terrestrial laser scanning*. CRC Press.
- Wang, Zuyuan, Ruedi Boesch, and Christian Ginzler (2012). "Forest delineation of aerial images with Gabor wavelets". In: *International journal of remote sensing* 33.7, pp. 2196–2213.
- Wasiq, Mahwash and Masood Ahmad (2004). *Sustaining forests: A development strategy*. The World Bank.
- Weinmann, Martin (2016). "Reconstruction and Analysis of 3D Scenes". In: *Springer International Publishing, Cham. doi 10*, pp. 978–3.
- Weinmann, Martin, Boris Jutzi, Stefan Hinz, and Clément Mallet (2015). "Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 105, pp. 286 –304.
- White, Joanne C, Nicholas C Coops, Michael A Wulder, Mikko Vastaranta, Thomas Hilker, and Piotr Tompalski (2016). "Remote sensing technologies for enhancing forest inventories: A review". In: *Canadian Journal of Remote Sensing* 42.5, pp. 619–641.
- Whitney, A Wayne (1971). "A direct method of nonparametric measurement selection". In: *IEEE Transactions on Computers* 100.9, pp. 1100–1103.

- Wilson, Roland and Michael Spann (1988). *Image segmentation and uncertainty*. John Wiley & Sons, Inc.
- Wu, Zhengzhe, Ville Heikkinen, Markku Hauta-Kasari, Jussi Parkkinen, and Timo Tokola (2014). "ALS data based forest stand delineation with a coarse-to-fine segmentation approach". In: *IEEE Congress on Image and Signal Processing (CISP)*. Dalian, China, pp. 547–552.
- Wulder, Michael A, Christopher W Bater, Nicholas C Coops, Thomas Hilker, and Joanne C White (2008a). "The role of LiDAR in sustainable forest management". In: *The Forestry Chronicle* 84.6, pp. 807–826.
- Wulder, Michael A, Joanne C White, Geoffrey J Hay, and Guillermo Castilla (2008b). "Towards automated segmentation of forest inventory polygons on high spatial resolution satellite imagery". In: *The Forestry Chronicle* 84.2, pp. 221–230.
- YANG, Hua-chao, Shu-bi ZHANG, Ka-zhong DENG, and Pei-jun DU (2007). "Research into a feature selection method for hyperspectral imagery using PSO and SVM". In: *Journal of China university of mining and technology* 17.4, pp. 473–478.
- Yanowitz, Shimon D and Alfred M Bruckstein (1989). "A new method for image segmentation". In: *Computer Vision, Graphics, and Image Processing* 46.1, pp. 82–95.
- Zargar, Amin, Rehan Sadiq, Bahman Naser, and Faisal I Khan (2011). "A review of drought indices". In: *Environmental Reviews* 19, pp. 333–349.
- Zhang, Liangpei, Yanfei Zhong, Bo Huang, Jianya Gong, and Pingxiang Li (2007). "Dimensionality reduction based on clonal selection for hyperspectral imagery". In: *IEEE Transactions on Geoscience and Remote Sensing* 45.12, pp. 4172–4186.
- Zhang, Yu-Jin (2006). *Advances in image and video segmentation*. IGI Global.
- Zhuo, Li, Jing Zheng, Xia Li, Fang Wang, Bin Ai, and Junping Qian (2008). "A genetic algorithm based wrapper feature selection method for classification of hyperspectral images using support vector machine". In: *Proc.*

*Geoinformat. Joint Conf. GIS Built Environ. Classif. Remote Sens. Images Int. Soc. Opt. Photonics*, 71471J–71471J.