Land Cover CCI

Algorithm Theoretical Basis Document

Part III: Classification

Year 2 - 1.2

Document Ref:	CCI-LC-ATBDv2
Deliverable Ref:	D2.2
Version:	1.2
Creation Date:	2014-07-31
Last Modified:	2017-01-13

This page is intentionally blank.

Document Signature Table					
	NAME	FUNCTION	COMPANY	SIGNATURE	DATE
PREPARED	Bontemps S.		UCLouvain		
PREPARED	Radoux J.		UCLouvain		
PREPARED	De Maet T.		UCLouvain		
PREPARED	Lamarche C.		UCLouvain		
PREPARED	Moreau I.		UCLouvain		
PREPARED	Vittek M.		UCLouvain		
VERIFIED	Defourny P.	Science leader	UCLouvain		
APPROVED					

Document Change Record

VERSION	DATE	DESCRIPTION	APPROVED
0.1	2015-10-22	First draft version, based on Year 1 document (version 1.2)	
1.0	2016-05-02	First version of the document, representative of Year 2 classification processing chain	
1.1	2016-06-15	Updated to answer CCI-LC_Ph2_M4-Deliverables_RIDs	
1.2	2017-01-13	Second version of the document, representative of Year 2 classification processing chain	

From version 1.1 to version 1.2

SECTION	COMMENTS
4.2, 4.3, 4.4	Those sections were updated to be in line with the classification chain of the land cover products of Phase II, year 2
5	The quality control section is completed with the process applied to check the land cover products of Phase II, year 2

From version 1.0 to version 1.1

RID	SECTION	COMMENTS
FR-01	3.2.1	Precisions about the reprojection to the Plate-Carrée coordinate system were brought to the entire section and Figure 3-3
FR-02	3	In section "3 GENERATION OF THE BASELINE GLOBAL LC MAP", not all tables represent

		look-up tables (LUTs). Tables that represent LUTs are explicitly defined as such (e.g. p30: *"In order to reclassify the original maps, the CCI legend should be defined. It is stored in the LUT1 which is presented in Table 3-10)".* It is recognized that Tables and LUTs can be confusing. Therefore, specific annotations for LUTs will be defined for the next version of the ATBD.
FR-03	3.4.3	Pseudo-code representation. TYPO was corrected to "CLASSIF2_MY: raster where each..."
FR-04	4.1	The sub-classes were taken into account. Table 4-1 was updated.

From version 0.1 to version 1.0

RID	SECTION	COMMENTS
	3.3.1	Introduction of the algorithm that converts the reference LC database at 20 m to 300 m
	4.1	Completion of the decision rules for all types of change
	4.2	Update of the change delineation method
	0	Modifications in the Baseline update section to reflect the production of annual land cover maps

Document Diffusion List

ORGANISATION	NAME	QUANTITY
ESA	O. Arino, F. Ramoino	

SYMBOLS AND ACRONYMS

ATBD	: Algorithm Theoretical Basis Document
AVHRR	: Advanced Very High Resolution Radiometer
BRDF	: Bidirectional Reflectance Distribution Function
CCI	: Climate Change Initiative
CCI-LC	: CCI Land Cover
CMC	: Climate Modelling Community
DARD	: Data Access Requirement Document
Envisat	: Environmental Satellite
EO	: Earth Observation
ESA	: European Space Agency
FR	: Full Resolution
IPCC	: Intergovernmental Panel on Climate Change
LC	: Land Cover
LCCS	: Land Cover Classification System
LUT	: Look-Up Table
MC	: Mean Compositing
MERIS	: Medium Resolution Imaging Spectrometer
ML	: Maximum Likelihood
PDF	: Probability Density Function
PROBA-V	: Project for On-Board Autonomy - Vegetation
PSD	: Product Specification Document
PVASR	: Product Validation and Algorithm Selection Report
RR	: Reduced Resolution
SPOT	: Satellite Pour l'Observation de la Terre
SPOT-VGT	: SPOT-Vegetation
SR	: Surface Reflectance
SSE	: Sum of Squared Error
UR	: User Requirement
URD	: User Requirement Document

REFERENCE DOCUMENTS

Applicable documents

ID	TITLE	ISSUE	DATE
AD.1	Statement of Work for ESA Climate Change Initiative Phase II - CCI-PRGM-EOPS-SW-12-0012	1.2	07.06.2013
AD.2	ESA Climate Change Initiative Phase 2 - Land Cover ECV Technical baseline for the project (update of the technical proposal with clarification and negotiation items)	1.0	13.03.2014
AD.3	CCI-LC URD Phase II. Land Cover Climate Change Initiative - User Requirements Document	1.0	13.07.2015
AD.4	CCI-LC PSD Phase II. Land Cover Climate Change Initiative - Product Specification Document	1.2	13.04.2015
AD.5	CCI-LC ATBD Phase II. Land Cover Climate Change Initiative - Algorithm Specification Document - Part I: Overview	1.2	10.09.2015
AD.6	CCI-LC ATBD Phase II. Land Cover Climate Change Initiative - Algorithm Specification Document - Part II: Pre-processing	1.2	11.09.2015
AD.7	CCI-LC DARD Phase II. Land Cover Climate Change Initiative - Data Access Requirement Document	1.2	13.04.2015
AD.8	CCI-LC Memo. Global land cover map for 1990s from AVHRR – reasons for delay and planning	1.0	14.09.2015
AD.9	CCI-LC quality control database for LC maps v2.0	1.0	13.01.2017

Reference documents

ID	TITLE	ISSUE	DATE
RD.1	CCI-LC URD Phase I. Land Cover Climate Change Initiative - User Requirements Document	2.2	23.02.2011
RD.2	CCI-LC PSD Phase I. Land Cover Climate Change Initiative - Product Specification Document	1.11	03.07.2014
RD.3	CCI-LC PVP Phase I. Land Cover Climate Change Initiative - Product Validation Plan	1.3	04.07.2011
RD.4	CCI-LC ATBD Phase I. Land Cover Climate Change Initiative - Algorithm Theoretical Basis Document	2.3	28.11.2013

ID	TITLE	ISSUE	DATE
RD.5	CCI-LC PVASR Phase I. Land Cover Climate Change Initiative - Product Validation and Algorithms Selection Report	2.2	03.07.2012
RD.6	CCI-LC PUG Phase I. Land Cover Climate Change Initiative - Product User Guide	2.4	02.09.2014
RD.7	Vancutsem C., Peckel, J-F., Boagert, P., Defourny, P., 2004, "Assessment of the mean compositing strategy for SPOT VEGETATION time serie", International Journal of Remote Sensing		
RD.8	Langner, A., Miettinen, J. and Siegert, F., 2007, „Land cover change 2002–2005 in Borneo and the role of fire derived from MODIS imagery", Global Change Biology, 13, 2329–2340.		
RD.9	Bontemps, S., Langner, A. and Defourny, P., 2012, "Monitoring forest changes in Borneo on a yearly basis by an object-based change detection algorithm using SPOT-VEGETATION time series", 33, 15, 4673-4699		
RD.10	Cihlar, J., Ly, H., Li, Z., Chen, J., Pokrant, H. and Huang, F., 1997, "Multitemporal, multichannel AVHRR data sets for land biosphere studies-artefacts and corrections", Remote Sensing of Environment, 60, 35–57.		
RD.11	GlobCover 2005 Project - Validation Report (Report editors: Bicheron P. and Leroy, M, End-User Meeting 3	2.1	November 2008
RD.12	Lillesand, T. M. and Kiefer, R. W., 2000, "Remote sensing and image interpretation", New York: John Wiley & Sons ,Inc.	4th edition	
RD.13	Bontemps, S., Herold, M., Kooistra, L., van Groenestijn, A., Hartley, A., Arino, O., Moreau, I., Defourny, P. 2012, "Revisiting land cover observation to address the needs of the climate modeling community", Biogeosciences, 9, 2145-2157		
RD.14	Loveland, T. R., Reed, B. C., Brown, J. F., Ohlen, D. O., Zhu, Z., Yang, L., and Merchant, J. W., 2000, "Development of a global land cover characteristics database and IGBP DISCover from 1 km AVHRR data", International Journal of Remote Sensing, 21, 1303–1365		
RD.15	Bartholome, E. and Belward, A., 2005, "GLC2000: a new approach to ´ global land cover mapping from Earth Observation data", International Journal of Remote Sensing, 26, 1959–1977		
RD.16	Defourny, P., Bicheron, P., Brockman, C., Bontemps, S., van Bogaert, E., Vancutsem, C., Pekel, J. F., Huc, M., Henry, C., Ranera, F., Achard, F., Di Gregorio, A., Herold, M., Leroy, M., and Arino, O., 2009, "The first 300 m global land cover map for 2005 using Envisat MERIS time series: a product of the GlobCover system". In: "Proceedings of the 33rd International Symposium on Remote Sensing of Environment", Stresa, Italy, 4–8 May 2009, TS-5-1 (Ref 791)		

ID	TITLE	ISSUE	DATE
RD.17	Friedl, M. A., Sulla-Menashe, D., Tan, B., Schneider, A., Ramankutty, N., Sibley, A., and Huang, X., 2010, "MODIS Collection 5 global land cover: Algorithm refinements and characterization of new datasets", Remote Sensing of Environment, 114, 168–182		
RD.18	Bontemps, S., Defourny, P., van Bogaert, E., Kalogirou, V., and Arino, O.: GlobCover 2009: Products description and validation report, ESA GlobCover project, 53 pp., 2011		
RD.19	IPCC 2006, 2006 IPCC Guidelines for National Greenhouse Gas Inventories, Prepared by the National Greenhouse Gas Inventories Programme, Eggleston H.S., Buendia L., Miwa K., Ngara T. and Tanabe K. (eds). Published: IGES, Japan.		
RD.20	Arino, O., Bicheron, P. Achard, F. Latham, J. Witt, R. and Weber, J.-L., "GlobCover the most detailed portrait of Earth", ESA Bulletin, 136, pp 25-31, November 2008: Available at the following link: http://www.esa.int/esapub/bulletin/bulletin136/bul136d_arino.pdf		
RD.21	Mayaux, P., Lambin, E.F. 1995, Estimation of tropical forest area from coarse spatial resolution data: "A two-step correction function for proportional errors due to spatial aggregation", Remote Sensing of Environment, 53 (1), pp. 1-15.		
RD.22	Jeanjean, H., Achard, F. 1997, "A new approach for tropical forest area monitoring using multiple spatial resolution satellite sensor imagery". International Journal of Remote Sensing, Vol. 18, Iss. 11.		
RD.23	Pesaresi M., Ehrlich D., Ferri S., Florczyk A.J., Freire S., Halkia S., Julea A.M., Kemper T., Soille P. and V. Syrris. Operating procedure for the production of the Global Human Settlement Layer from Landsat data of the epochs 1975, 1990, 2000, and 2014. Publications Office of the European Union, EUR 27741 EN, 2016. doi: 10.2788/253582.		
RD.24	Esch, Thomas, Wieke Heldens, Andreas Hirner, Manfred Keil, Mattia Marconcini, Achim Roth, Julian Zeidler, Stefan Dech, and Emanuele Strano. 2017. "Breaking New Ground in Mapping Human Settlements from Space – The Global Urban Footprint." ISPRS Journal of Photogrammetry and Remote Sensing 134: 30–42. https://doi.org/https://doi.org/10.1016/j.isprsjprs.2017.10.012.		

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

1 INTRODUCTION

1.1 Scope

This document is part of the Algorithm Theoretical Basis Document (ATBD) of the Phase II Land Cover (LC) project belonging to the European Space Agency (ESA) Climate Change Initiative (CCI). It is dedicated to the classification chain that generates the global LC maps.

This second version focuses on the activities of year 2. The product evolutions of this second year are manifold. First, annual LC maps instead of 5-year epoch maps are released. Second, the land cover classes considered for the change have been extended to 8 instead of forests only. Third, the change detected at 1 km is now delineated in more details at 300 m. Finally, the time frame covered by the maps is extended in past to the 1990s with the Advanced Very High Resolution Radiometer (AVHRR) and to current years with PROBA-V.

The last version (v3) of this document will be released at the end of the 3rd year. The ATBD-v3 will be the final version of the document.

1.2 Structure of the document

This document gives a detailed description of the classification chain. After this introduction, the document is organized in 4 main sections:

- Section 2 gives a general overview of the classification chain implemented in Phase I and that still will be developed in Phase II, as well as the main input and output data of the 2nd year of this Phase II;
- Section 3 details all the steps of the classification chain that generates the baseline LC map;
- Section 4 presents the processing steps that allow deriving the annual land cover maps from the baseline LC map.
- Section 5 describes the quality control procedure applied on the global annual LC maps before their release.

2 CLASSIFICATION IN PHASE II

2.1 General overview

The classification chain is organized into 2 main processes: (i) the generation of a baseline global LC map using the entire archive of the Envisat Medium Resolution Imaging Spectrometer (MERIS) data and (ii) the generation of global annual LC maps from this baseline product. An overall overview is provided in Figure 2-1.

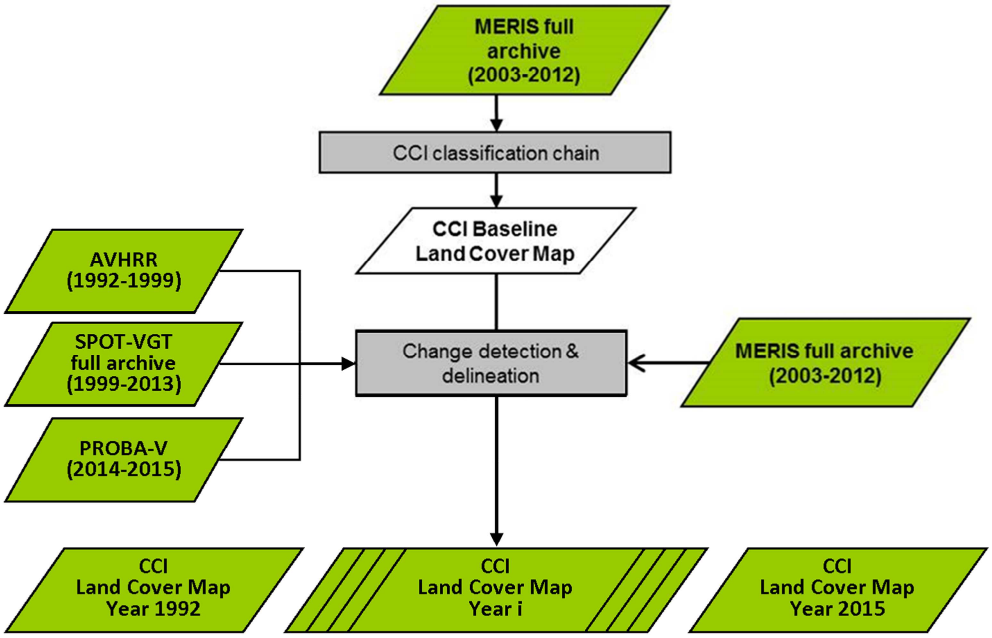

Figure 2-1: Schematic representation of the CCI-LC classification chain made of 2 main processes to generate global annual LC maps using the entire archives of Envisat MERIS, AVHRR time series between 1992 to 1999, SPOT-VGT time series between 1999 and 2013 and PROBA-V data for 2014 and 2015.

The two processes, classification and change detection, will be detailed in sections 3 and 4, respectively.

2.2 Input and output data

The global annual LC maps covering the 1990s until the current years rely on EO datasets coming from four different sensors: MERIS, SPOT-Vegetation (SPOT-VGT), PROBA-V and AVHRR 2.

The classification module ingests the Surface Reflectance (SR) time series that are generated in the project [AD.6]. The global annual LC maps are derived from a 10-year baseline LC map which is generated thanks to the entire MERIS Full and Reduced Resolution (FR and RR, respectively) archive from 2003 to 2012. This 10-year baseline LC map is then updated using (i) SPOT-VGT time series from 1999 to 2013, (ii) PROBA-V time series from 2014 and 2015 and (iii) AVHRR time series from 1992 to 1999.

Table 2-1 lists the satellite datasets that are used in order to generate the global LC maps.

Table 2-1: Satellite data sources that are planned to be used to generate the global LC maps.

GLOBAL LC DATABASE	REFERENCE PERIOD	SATELLITE DATA SOURCE
Baseline 10-year global LC map	2003-2012	• MERIS FR/RR global SR composites between 2003 and 2012
Global annual LC database	1992-1999	• Baseline 10-year global LC map • AVHRR global SR composites between 1992 and 1998 for back-dating the baseline
	1999-2013	• Baseline 10-year global LC map • SPOT-VGT global SR composites between 1999 and 2013 for up and back-dating the baseline • MERIS FR global SR composites between 2003 and 2012 to delineate the identified changes at 300m spatial resolution • PROBA-V global SR composites at 300 m for year 2013 to delineate the identified changes at 300m spatial resolution
	2014-2015	• Baseline 10-year global LC map • PROBA-V global SR composites at 1 km for years 2014 and 2015 for up-dating the baseline • PROBA-V time series at 300 m for 2014 and 2015 to delineate the identified changes at 300m spatial resolution

3 GENERATION OF THE BASELINE GLOBAL LC MAP

3.1 Classification logical model

3.1.1 General overview

The classification process transforms the L3 seasonal surface reflectance composites produced by the pre-processing step [AD.6] into meaningful global LC products.

As already introduced (section 2.1), the classification chain is organized into 2 main processes (Figure 2-1): (i) the generation of a baseline global LC map using the entire archive of the Envisat MERIS data and (ii) the generation of global annual LC from this baseline product. This section focuses on the first step, namely the generation of the baseline LC map.

The baseline map is obtained through a classification process organized in 4 major processing steps (Figure 3-1), which rely on data prepared during preliminary steps.

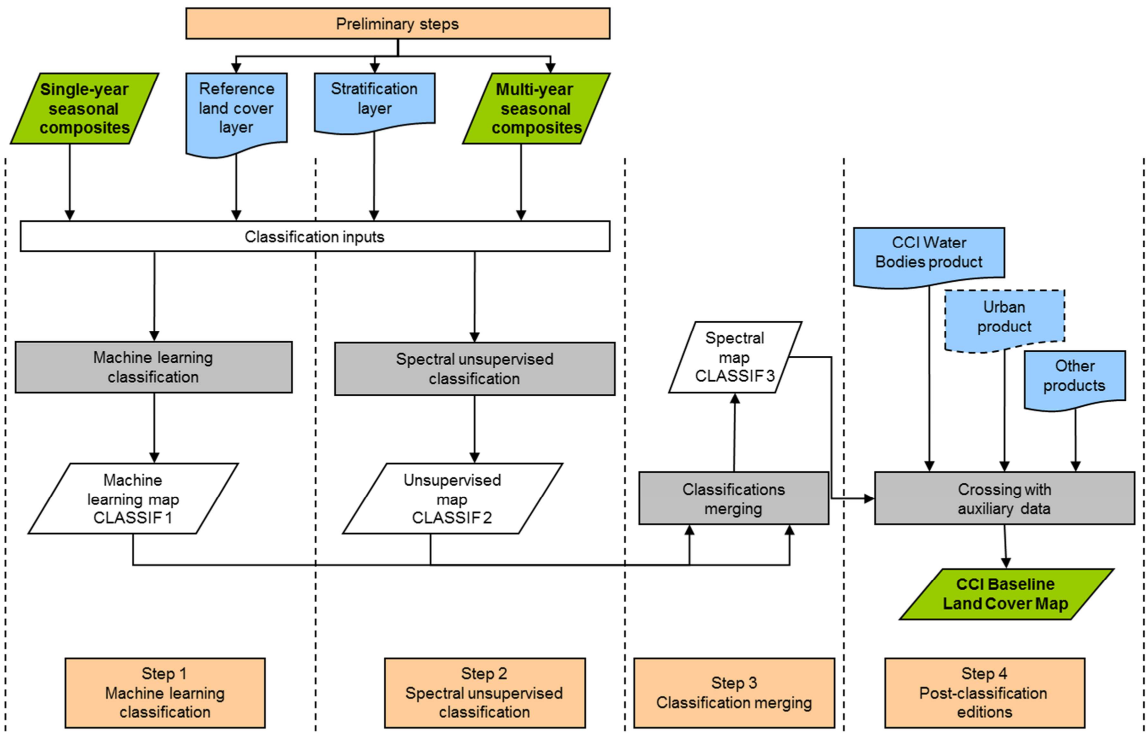

Figure 3-1: Schematic representation of the classification process developed to generate a baseline global LC map over the period 2003-2012 using the entire archives of Envisat MERIS data

Before the four classification steps, three preliminary steps are performed:

- the **preparation of the reference LC database**: a key auxiliary dataset in the classification chain is a global reference LC database. This database consists of a set of existing global, regional and local land cover maps which are merged together after preparatory transformations and according to specific merging rules;
- the **preparation of the stratification layer**: before the classification process, the world is stratified in equal-reasoning areas from an ecological and a remote sensing point of view. The stratification objectives are twofold: (1) reducing the land surface reflectance variability in the dataset in order to improve the classification efficiency and (2) allowing a regional tuning of the classification parameters to take into account the regional characteristics (vegetation seasonality, cloud coverage, etc.). In other terms, the classification chain runs independently for each equal-reasoning area with specific parameters. The great but much controlled flexibility of this strategy allows defining a classification process valid at global scale while tackling both the regional heterogeneity of the land cover characteristics;
- the **generation of multi-year seasonal composites**: the classification is done according to a multi-year approach. The rationale behind this approach is explained in the next section (3.1.2), as well as the main principles that underlie it. One consequence of this approach is that multi-year seasonal composites (i.e. seasonal composites that aggregate data from different years) need to be generated before the classification in order to be used as input by the processing chain.

In the first and second steps of the classification chain (Figure 3-1), machine learning and unsupervised classification algorithms are run using the spectral properties of seasonal composites as input, resulting in two different maps “CLASSIF1” and “CLASSIF2”. In the third step, these two maps are merged to produce a spectral map “CLASSIF3”. This baseline map is finalized in the fourth step through post-classification editions, the main one consisting in adding the CCI Water Bodies product.

The different algorithmic steps are presented in detail below in sections 3.2 to 3.6. For each step, the algorithm rationale is presented, input and output are identified, algorithms parameters are given and the pseudo-code is provided. In general, the parameters of the classification chain that are related to equal-reasoning areas are stored in Look-Up Tables (LUT), which are included throughout the document.

3.1.2 Multi-year approach

One requirement expressed by the Climate Modelling Community (CMC) in the User Requirements (UR) collection activity [AD.3, RD.1] is the need to have successive global LC maps stable over time. In the second year of this Phase II, the need for global annual land cover maps was explicitly expressed by the CMC.

Since the early nineties, several global land cover products have been delivered, all based on “single-year” and “single-sensor” approaches [RD.14, RD.15, RD.16, RD.17 and RD.18]. More recently, the accumulation of global long-term time series of EO data has allowed the delivery of several global maps derived from the same sensor. This is the case for the ESA GlobCover[1] and MODIS[2] products. This capacity to produce successive maps based on data acquired by a single sensor is certainly a major advance, but it also raised new issues. In the suite of MODIS products, [RD.17] reports significant year-to-year variations in land cover labels not associated with land cover change. Similarly, the comparison between the GlobCover 2005 and 2009 maps highlights discrepancies between products [RD.18].

This phenomenon was investigated through a series of tests in Phase 1 during the round-robin exercises. They demonstrated the interest of using several years of Earth Observation (EO) dataset to generate LC maps [RD.5, RD.13]. The CCI-LC project took therefore the twofold decision to: (i) base the classification chain on several years of EO data while (ii) delivering global annual LC maps. In Phase 1, three epochs were generated which were the 1998-2002, 2003-2007 and 2008-2012. In Phase 2, LC maps are delivered on an annual basis and the length of the time series is extended back to the 1990s and up to 2015 [AD.4, RD.2].

Two different strategies for handling multi-year dataset in the classification chain have been found efficient to increase the products accuracy and stability (Figure 3-2).

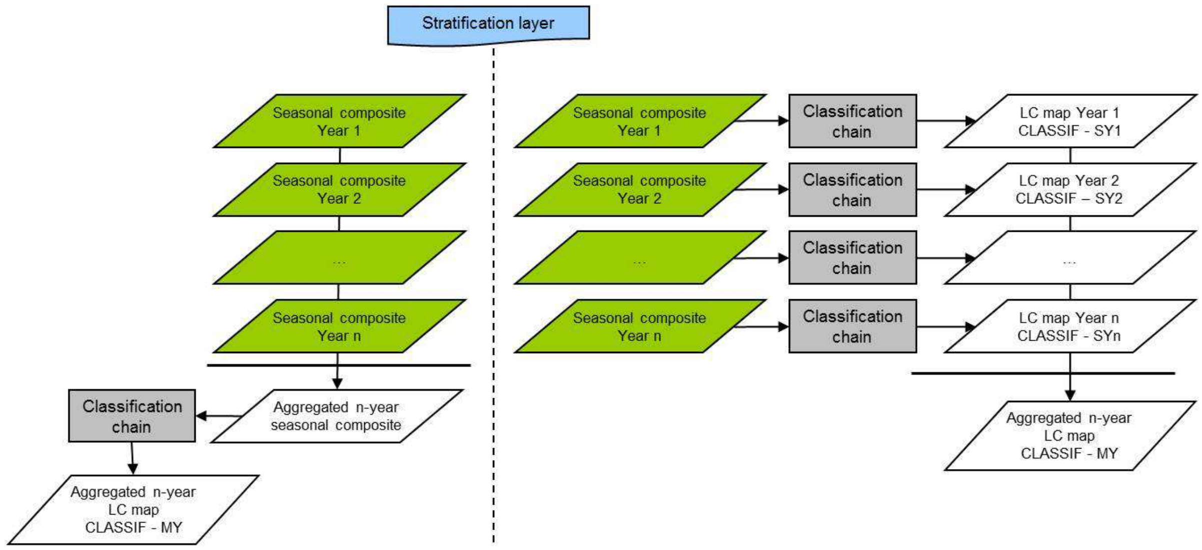

Figure 3-2: Logical flow for the multi-year approach in the classification chain

The first strategy (hereafter referred to as "MY_S1" for "Multi-Year Strategy 1") makes use of the multi-year dataset to increase the quality of the composites to classify. Seasonal composites from several years are combined into multi-year seasonal composites which then serve as input to the classification chain. In this case, the classification chain is run only once to directly produce a multi-year LC map ("CLASSIF_MY").

[1] http://due.esrin.esa.int/page_globcover.php
[2] http://modis.gsfc.nasa.gov/data/dataprod/dataproducts.php?MOD_NUMBER=12

In the second strategy (hereafter referred to as "MY_S2" for "Multi-Year Strategy 2"), the multiple years are combined at the end of the classification chain, at the level of the LC maps. First, the classification algorithms are run once for each year of interest, resulting in multiple "single-year" LC maps ("CLASSIF_SY"). Second, the multiple single-year products are combined in a multi-year LC map ("CLASSIF_MY") according to specific aggregation rules, based on the majority voting principle.

According to the stratum, either strategy 1 or 2 is applied (Table 3-12). Indeed, each equal-reasoning area represents specific climatic conditions, seasonal behaviours and remote sensing conditions and this influences the performance of both strategies. The main advantage of strategy 1 is to increase the quality of the composites to classify. It is therefore applied in strata where the data coverage and quality is poorer due to rather long snow and cloud periods. Conversely, in regions where the data coverage and quality is not problematic, the strategy 2 is applied since it allows better accounting for the inter-annual variability.

3.1.3 Multi-sensor approach

The development of a multi-sensor approach is required by (i) the lack of acquisition in the MERIS FR acquisition mode and (ii) the temporal coverage of the MERIS sensor limited to the 10 year-period from 2003 to 2012.

3.1.3.1 *Increasing MERIS FR spatial coverage*

Using images acquired in the MERIS RR mode is considered as the most convenient approach to deal with a possible lack of MERIS FR acquisitions. Indeed, MERIS FR and RR time series are acquired by the same sensor on the same satellite, with the difference that FR data are less regularly downlinked. In any cases, it must obey the following consideration: priority is given to the best available EO information and therefore, the use of MERIS RR is not systematic. In other terms, where and when the coverage of the MERIS FR dataset is high enough to allow producing consistent land cover maps, MERIS RR is not used.

The merging between MERIS FR and RR dataset is needed where there are gaps in the MERIS FR seasonal composites (i.e. where the number of MERIS FR available observation is null). The merging is operated at the baseline LC map level. It means that the classification chain is run for the MERIS FR and RR time series independently, resulting in a "FR_Baseline" and a "RR_Baseline".

If there are gaps in the "FR_Baseline", they are filled in with the "RR_Baseline". This operation takes place in the fourth "Post-classification editions" step of the classification chain (detailed in Section 3.6).

3.1.3.2 *Increasing MERIS temporal coverage*

In order to generate LC maps from the 1990s up to current years, AVHRR, SPOT-VGT and PROBA-V time series need to be used in addition to MERIS data which are only available between 2003 and 2012. These three sensors are not included in the classification chain that generates the baseline LC map, but they are used for detecting changes and back-date the baseline LC map. More information is given in Section 4.

3.2 Detailed processing scheme of the preliminary steps

3.2.1 Preparation of the reference land cover database

A key auxiliary dataset in the classification chain, for the spectral algorithms, is the global reference LC database. The reference database consists of a set of existing global, regional and local LC maps which have been selected as the most accurate ones available for a given region, with the highest spatial resolution and with a CCI-compatible legend.

Each dataset is re-projected to the Plate-Carrée coordinate system, translated into the CCI-LC legend and resampled to a standard spatial resolution, thus generating a set of GeoTiff intermediate products which are merged according to pre-defined rules to build the final reference LC database. The workflow is illustrated in Figure 3-3.

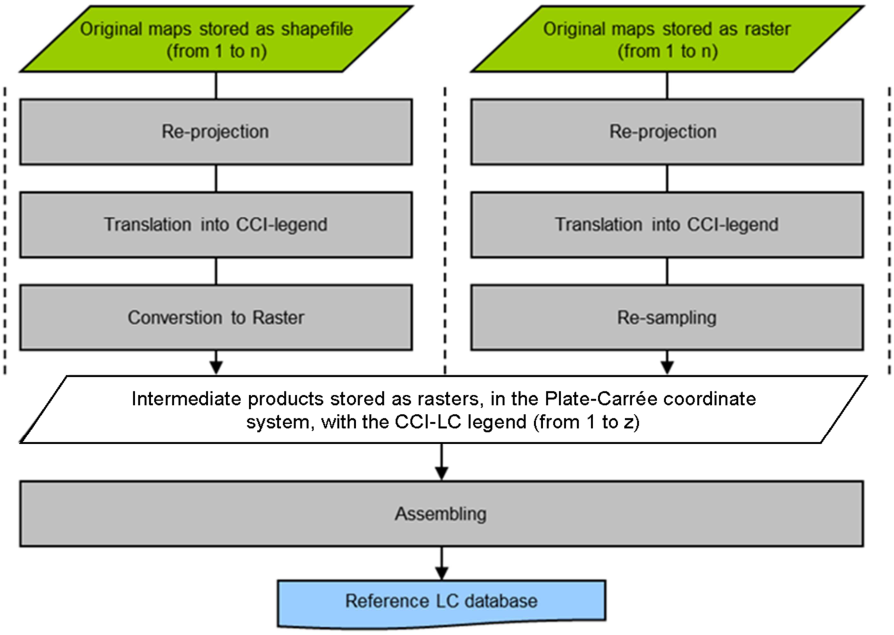

Figure 3-3: Logical flow for generation of the reference land cover database

These operations are performed with the ArcGIS software, using different toolboxes that vary depending on the original map stored as a shapefile or as a raster. In the case of an original shapefile, the toolboxes are "Project", "Add field", Calculate field" and "Polygon to Raster conversion". In the case of an original raster, the toolboxes are "Project_Raster", "Reclassify" and "Resample". The last step to assemble the products is run using a C++ code specifically developed.

The “Project” toolbox projects spatial data from one coordinate system to another. It uses as parameters the output dataset to which the results will be written, the output coordinate system and the transformation method used between two geographic systems or datum.

The “Add field” and “Calculate field” toolboxes allow first adding a field to the attribute table of the shapefile and then calculating new value (which will be the CCI class) to the new field based on mathematical expression or rules. The “Add field” toolbox uses as parameters the name of the field that will be added and its type (text, float, double, short, long, date, blob, raster). The “Calculate field” toolbox uses as parameters the field name that will be updated through the calculation and the expression used to attribute the new value.

The “Polygons_To_Raster conversion” toolbox converts polygons features to a raster dataset. It uses as parameters the field used to assign values to the output raster, the output raster name and format and the output cell size.

The “Project_Raster” toolbox transforms a raster dataset from one projection to another. It uses as parameters the output raster name and format, the output coordinate system and the transformation method used between two geographic systems or datum.

The “Reclassify” toolbox changes the values in a raster. It uses as parameters the field denoting the values that will be reclassified, the conversion table specifying how to reclassify values of the input raster and the information denoting whether missing values in the conversion table retain their value or get mapped to NoData.

The “Resample” toolbox alters the raster dataset by changing the cell size through a specific resampling method. It uses as parameters the output raster name and format, the cell size for the new raster dataset and the resampling algorithm to be used.

The C++ code executes single combinations of maps, based on a priori rules that define how combining the maps (which sequence and which classes).

It should be noted that not all toolboxes are run necessarily. If the original (shapefile or raster) maps are already in projected in the Plate-Carrée coordinate system, the first step is omitted. Similarly, if the original raster map has already the standard spatial resolution, the last step is not run.

- ***Algorithm assumptions and limitations***

None

- ***Input and output data***

The set of existing maps used to build the reference is documented in the Data Access Requirement Document (DARD) [AD.7]; maps are just listed in Table 3-1. Input and output data associated with all ArcGis Toolboxes and with the C++ code are presented in Table 3-2 to Table 3-8.

It should be noted that the older GlobCover 2005 product was preferred to the more recent GlobCover 2009 one. This decision is motivated by the general higher quality of the 2005 map. As written in the Products description and validation report [RD.18], the 2009 product was generated using 12 months of data (vs 18 months for GlobCover 2005) and it resulted in a lower classification quality in several regions of the world.

Table 3-1: List of maps used to build the reference LC database

NAME OF LC MAP	EXTENT	SPATIAL RESOLUTION
GLC2000 – global map	Global	1000 m
GLC2000 – South America map	South America	1000 m
GLC2000 – Europe	Europe	1000 m
GLC2000 – Africa	Africa	1000 m
GLC2000 – Greenland	Greenland	1000 m
GLC2000 – Asia	Asia	1000 m
GLC2000 – New Zealand	New Zealand	1000 m
GLC2000 – Fidji	Fidji	1000 m
GLC2000 – South Asia	South Asia	1000 m
GLC2000 – Southeast Asia	Southeast Asia	1000 m
GlobCover 2005	Global	300 m
Canada LC map	Canada	250 m
Canada ACI map	Canada	30 m
Meso-America LC map	Central America	500 m
North America Atlas	Canada, United States, Mexico	500 m
United States National LC Database	United States (including Alaska)	30 m
Alaska North Slope	Alaska	20 m
Corine Land Cover 2000	Europe	20 m
Corine Land Cover 2006	Europe	20 m
Corine Land Cover 2012	Europe	20 m
Congo Vegetation Types	Congo Basin	300 m
Burkina Faso LC database	Burkina Faso	20 m
GLCN LC maps	Senegal, Buthan, Nepal	30m
Africover 2000 LC maps	Burundi, Egypt, Eritree, Kenya, RDC, Rwanda, Somalia, Sudan, Tanzania, Uganda	30 m
SERVIR 2010 LC maps	Rwanda, Botswana, Malawi, Namibia, Zambia, Uganda, Tanzania, Ethiopia	30 m
South Africa SAGE	South Africa, Tanzania, Zimbabwe, Zambia, Botswana, Namibia, Mozambique, Malawi, Swaziland, Lesotho	30 m
JRC Crop Mask (rainfed, irrigated)	Africa	500 m
Russian Forest	Russia	300 m
Ukraine SRI LC map	Ukraine	30 m
Central Asia DLR LC map	Central Asia	500 m

Name of LC map	Extent	Spatial resolution
Indian LC map	India	30 m
Japan JAXA LC map	Japan	30 m
China LC map	China	1000 m
GLC30 LC map	Global	30 m
Cambodia LC map	Cambodia	30 m
Southeast Asia CRISP LC map	Southeast Asia	500 m
Australia LC map	Australia	250 m
Chatham islands LC map	Chatham islands	30 m
Mangrove atlas	Global	30 m
Global cropland extent	Global	250 m
MODIS urban extent	Global	500 m
JRC Global Human Settlement Layer	Global	38 m
CCI water body product	Global	150 m
Canada Forested Area map	Canada	30 m
Global Forest map	Global	30 m
Madagascar map	Madagascar	30 m
Brazil Forest map	Brazil	30 m
Afghanistan map	Afghanistan	30 m
Pakistan map	Pakistan	30 m
Uruguay map	Uruguay	30 m
New Zealand map	New Zealand	30 m
Venezuela Vegetation map	Venezuela	500 m

Table 3-2: Input and output data for the “Project” toolbox

Data	Description	Intent (IN, OUT, INOUT)	Physical unit	Range
OriginalShp_<n>	Original land cover map as a shapefile	IN	None	[0 … 255]
OriginalShp_<n>_WGS84	Original land cover map projected in the Plate-Carrée projection as a shapefile	OUT	None	[0 … 255]

Table 3-3: Input and output data for the "Add field" and "Calculate field" toolboxes

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
OriginalShp_<n>_WGS84	Original land cover map projected in the Plate-Carrée projection as a shapefile	IN	None	[0 ... 255]
OriginalShp_<n>_WGS84_Reclassified	Original land cover map projected in the Plate-Carrée projection as a shapefile	OUT	None	[0 ... 255]

Table 3-4: Input and output data for the "Polygons_To_Raster conversion" toolbox

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
OriginalShp_<n>_WGS84_Reclassified	Original land cover map projected in the Plate-Carrée projection as a shapefile	IN	None	[0 ... 255]
OriginalMap_<n>_WGS84 _Reclassified_Resample	Original land cover maps translated into the CCI LCCS legend, projected in the Plate-Carrée projection and resampled to a standard spatial resolution, as a GeoTiff raster file	IN	None	[0 ... 255]

Table 3-5: Input and output data for the "Project_Raster" toolbox

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
OriginalMap_<n>	Original land cover map as a raster file	IN	None	[0 ... 255]
OriginalMap_<n>_WGS84	Original land cover map projected in the Plate-Carrée projection as a GeoTiff raster file	OUT	None	[0 ... 255]

Table 3-6: Input and output data for the "Reclassify" toolbox

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
OriginalMap_<n>_WGS84	Original land cover map projected in the Plate-Carrée	IN	None	[0 ... 255]

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
	projection as a GeoTiff raster file			
OriginalMap_<n>	Original land cover map as a raster file	IN	None	[0 ... 255]
OriginalMap_<n>_WGS84 _Reclassified	Original land cover map translated into the CCI LCCS legend and projected in the Plate-Carrée projection as a GeoTiff raster file	OUT	None	[0 ... 255]

Table 3-7: Input and output data for the "Resample" toolbox

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
OriginalMap_<n>_WGS84 _Reclassified	Original land cover map translated into the CCI LCCS legend and projected in the Plate-Carrée projection as a GeoTiff raster file	OUT	None	[0 ... 255]
OriginalMap_<n>_WGS84 _Reclassified_Resample	Original land cover map translated into the CCI LCCS legend, projected in the Plate-Carrée projection and resampled to a standard spatial resolution, as a GeoTiff raster file	OUT	None	[0 ... 255]

Table 3-8: Input and output data for the C++ code that assembles all existing maps

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
OriginalMap_<n>_WGS84 _Reclassified_Resample	Original land cover maps translated into the CCI LCCS legend, projected in the Plate-Carrée projection and resampled to a standard spatial resolution, as a GeoTiff raster file	IN	None	[0 ... 255]
Reference_CCI_<data>	Reference LC database as a GeoTiff raster file	OUT	None	[0 ... 255]

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
Reference_CCI_<data>_Source	GeoTiff raster file which identified, on a pixel basis, the original LC map from which the reference label is derived	OUT	None	[0 ... 255]

- ***Parameters***

Parameters associated with the ArcGIS toolboxes are listed in Table 3-9. There is no parameter associated with the C++ code.

Table 3-9: Parameters needed for running the "Project", "Add field", "Calculate field", "Polygons_to_Raster", "Project raster", "Reclassify" and "Resample" toolboxes

PARAMETERS	DESCRIPTION	INTENT (IN, OUT, INOUT)	FORMAT	RANGE
Project toolbox				
Out_coor_system	Output coordinate System	IN	String	/
Geographic_transform	Transformation method used between two geographic systems or datums (optional when the input and output coordinate systems have the same datum)	IN	String	/
Out_shp	Output shapefile name	IN	String	/
Add field toolbox				
Field_name	Name of the field that will be added	IN	String	/
Field_type	Type of the field that will be added	IN	text, float, double, short, long, date, blob, raster, guid	/
Calculate field toolbox				
Field_name	Name of the field that will be updated through the calculation	IN	String	/
Expression	Expression used to attribute the new value	IN	VB, Python	/
Polygons_To_Raster toolbox				
Conversion_Field	Field used to assign values to the output raster	IN	String	/
Out_raster	Output raster name and format	IN	String	/

PARAMETERS	DESCRIPTION	INTENT (IN, OUT, INOUT)	FORMAT	RANGE
Cell_size	Cell size for the new raster dataset	IN	Short	/
Project_Raster toolbox				
Out_coor_system	Output coordinate System	IN	String	/
Geographic_transform	Transformation method used between two geographic systems or datums (optional when the input and output coordinate systems have the same datum)	IN	String	/
Out_raster	Output raster name and format	IN	String	/
Reclassify toolbox				
Reclass_Field	Field containing the raster values to modify	IN	String	/
Remap	Table associating old raster values to new values	IN	String	/
Missing_Values	Information about how dealing with missing values in the reclassification table	IN	Boolean	[DATA, NODATA]
Out_raster	Output raster name and format	IN	String	/
Resample toolbox				
Cell_size	Cell size for the new raster dataset	IN	Short	/
Resampling_Type	Resampling algorithm to be used	IN	String	
Out_raster	Output raster name and format	IN	String	/

In order to reclassify the original maps, the CCI legend should be defined. It is stored in the LUT1 which is presented in Table 3-10. The LUT1 includes the parameters describing the CCI LC legend and contains the following fields:

- **NB_LAB**, which lists the numbers (IDs) ranging between 0 and 255 and corresponding to each LCCS land cover class;
- **LABEL**, which gives the names of each LCCS land cover class;
- **R, G and B**, which provide the red, green and blue components (from 0 to 255) of the color codes associated with each LCCS land cover class.

Table 3-10: Parameters describing the CCI LC legend (contained in LUT1)

NB_LAB	LABEL	R	G	B
0	No data	0	0	0
10	Cropland, rainfed	255	255	100
11	Cropland, rainfed, herbaceous cover	255	255	100
12	Cropland, rainfed, shrub and tree cover	255	255	0
20	Cropland, irrigated or post-flooding	170	240	240
30	Mosaic cropland (>50%) / natural vegetation (tree, shrub, herbaceous cover) (<50%)	220	240	100
40	Mosaic natural vegetation (tree, shrub, herbaceous cover) (>50%) / cropland (< 50%)	200	200	100
50	Tree cover, broadleaved, evergreen, closed to open (>15%)	0	100	0
60	Tree cover, broadleaved, deciduous, closed to open (> 15%)	0	160	0
61	Tree cover, broadleaved, deciduous, closed (>40%)	0	160	0
62	Tree cover, broadleaved, deciduous, open (15-40%)	170	200	0
70	Tree cover, needleleaved, evergreen, closed to open (> 15%)	0	60	0
71	Tree cover, needleleaved, evergreen, closed (>40%)	0	60	0
72	Tree cover, needleleaved, evergreen, open (15-40%)	0	80	0
80	Tree cover, needleleaved, deciduous, closed to open (> 15%)	40	80	0
81	Tree cover, needleleaved, deciduous, closed (>40%)	40	80	0
82	Tree cover, needleleaved, deciduous, open (15-40%)	40	100	0
90	Tree cover, mixed leaf type (broadleaved and needleleaved)	120	130	0
100	Mosaic tree and shrub (>50%) / herbaceous cover (< 50%)	140	160	0
110	Mosaic herbaceous cover (>50%) / tree and shrub (<50%)	190	150	0
120	Shrubland	150	100	0
121	Shrubland evergreen	150	100	0
122	Shrubland deciduous	150	100	0
130	Grassland	255	180	0
140	Lichens and mosses	255	210	120
150	Sparse vegetation (tree, shrub, herbaceous cover)	255	235	175
151	Sparse tree	255	235	175
152	Sparse shrub	255	235	175
153	Sparse herbaceous cover	255	235	175

NB_LAB	LABEL	R	G	B
160	Tree cover, flooded, fresh or brakish water	0	120	90
170	Tree cover, flooded, saline water	0	150	120
180	Shrub or herbaceous cover, flooded, fresh-saline or brakish water	0	220	130
190	Urban	195	20	0
200	Bare areas	255	245	215
201	Consolidated bare areas	220	220	220
202	Unconsolidated bare areas	255	245	215
210	Water	0	70	200
220	Snow and ice	255	255	100

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

The reader is referred to the following webpage to access to the pseudo-codes of the different ArcGIS toolboxes:

- Project: http://resources.arcgis.com/en/help/main/10.1/index.html#//00170000007m000000
- Add field: http://resources.arcgis.com/en/help/main/10.1/index.html#/Add_Field/001700000047000000/
- Calculate field: http://resources.arcgis.com/en/help/main/10.1/index.html#/Calculate_Field/00170000004m000000/
- Polygons_to_raster: http://resources.arcgis.com/en/help/main/10.1/index.html#/Polygon_to_Raster/001200000030000000/
- Project_raster: http://resources.arcgis.com/en/help/main/10.1/index.html#/Project_Raster/00170000007q000000/
- Reclassify: http://resources.arcgis.com/en/help/main/10.1/index.html#/Reclassify/009z000000sr000000/
- Resample: http://resources.arcgis.com/en/help/main/10.1/index.html#/Resample/00170000009t000000/

The pseudo-code of the last assembling step is provided below, as Algorithm 3-1.

```
algorithm Assembling_Reference is

     input: all original LC maps translated into the CCI LC legend, projected in
     the Plate-Carrée projection and resampled to a standard spatial resolution,
     as GeoTiff raster files:

     output:
     Reference LC database as a GeoTiff raster file:
     Reference_CCI_<date>
     Geotiff raster file which identified, on a pixel basis, the original LC map
     from which the reference label is derived:
     Reference_CCI_<date>_SOURCE

     for each original input dataset:
        Attribute an ID to each original LC map
        <ID>_WGS84_LCCS_20m  =  resampled,  reprojected  and  LCCS  compatible  legend
        version of original LC map identified as <ID>
```

```
for each stratum:
     for each LC class:
        classify each original LC map with a weight according to the up-to-
        date,  per-class  quallilty  and  the  highest  spatial  resolution
        characteristics of the map
        per pixel, do:
             <ID>_WGS84_LCCS_20m_maxw = LC map with the highest weight
             Reference_CCI_<date> = <ID>_WGS84_LCCS_20m_maxw
             Reference_CCI_<date>_SOURCE = ID
     apply corrections if necessary
```

Algorithm 3-1. Assembling existing LC maps into a reference LC database

3.2.2 Preparation of the stratification layer

Using an a priori stratification in the land cover mapping processing chain allows increasing the classification algorithms performance but, at the same time, can also induce local artefacts in the final map [RD.5]. It has therefore to be used with caution. The location of strata limits has to be decided in order to separate regions characterized by different climatic conditions, seasonal behaviours and remote sensing conditions. The stratification layer developed in GlobCover, in which 22 strata were delineated manually.

The LUT2 describing the 22 strata included in the stratification layer is presented hereafter in Table 3-11. It contains the following fields:

- **NB_ST**, which lists the numbers (IDs) ranging between 1 and 22 and corresponding to each equal-reasoning area;
- **Name_ST**, which gives the names of each equal-reasoning area.

Table 3-11: Parameters describing the stratification layer (contained in LUT2)

NB_ST	NAME_ST
1	Polar areas
2	Canada
3	US
4	Central-America
5	Amazon
6	South-America
7	North-west Eurasia
8	North-east Eurasia
9	Eurasia
10	England
11	Mediterranea

NB_ST	NAME_ST
12	Africa desert
13	North-Africa
14	Central Africa
15	South-Africa
16	Madagascar
17	Asia desert
18	South-east Asia
19	Japan
20	Indonesia
21	Australia
22	New-Zealand

The classification chain will be run independently for each equal-reasoning area with specific parameters. A key parameter is the selection of appropriate period or seasons to classify. Indeed, the period or seasons that will optimize the discrimination between the different classes will differ if we are in tropical, temperate, or boreal areas. The compositing periods, stored internally as LUT 3, are specific for each stratum defined by NB_ST and Name_ST and are characterized by a date of start (Startdate_Season) and a date of end (Enddate_Season):

- **NB_ST**, which lists the numbers (IDs) ranging between 1 and 22 and corresponding to each equal-reasoning area;
- **Name_ST**, which gives the names of each equal-reasoning area;
- **Startdate_Season(i)**, specifying the exact date which marks the start of the seasonal composite i (i ranging from 1 to 3 according to the stratum);
- **Enddate_Season(i)**, specifying the exact date which marks the end of the seasonal composite i (i ranging from 1 to 3 according to the stratum).

3.2.3 Generation of multi-year seasonal composites

3.2.3.1 *General Mean Compositing approach*

7-day composites and seasonal composites are generated using the Mean Compositing (MC) algorithm [RD.7].

The compositing process aims at minimizing the cloud contamination and reducing variations in reflectance values due to image acquisition with varying geometries [RD.10]. The MC algorithm averages quality controlled reflectance values over the compositing period. It has proved to significantly reduce the Bidirectional Reflectance Distribution Function (BRDF) and atmospheric artefacts while resulting in spatially homogeneous cloud-free composites with good radiometric consistency even in humid tropical regions [RD.7, RD.8 and RD.9]. In addition, MC algorithm does not require any model adjustment or additional parameterizations, hence contributing to the generalization and automation of the process.

The MC algorithm uses as input a set of images (daily images for the 7-day composites and 7-day composites for the seasonal composites) which are made of reflectance values in several spectral bands and of quality flags [AD.6]. In both cases, the aggregation rule relies on the status associated with each pixel. On one hand, the aggregation is operated on pixels associated with a same status. On the other hand, rules exist to define the priorities between statuses. Priority is given to the “land” status and then, come the “snow”, “water” and “cloud” statuses in this sequence. The average is weighted by the number of observations associated with the selected status.

Based on these two principles, reflectance values from the input data are averaged by pixel and by band to generate the output composite, as indicated in Figure 3-4.

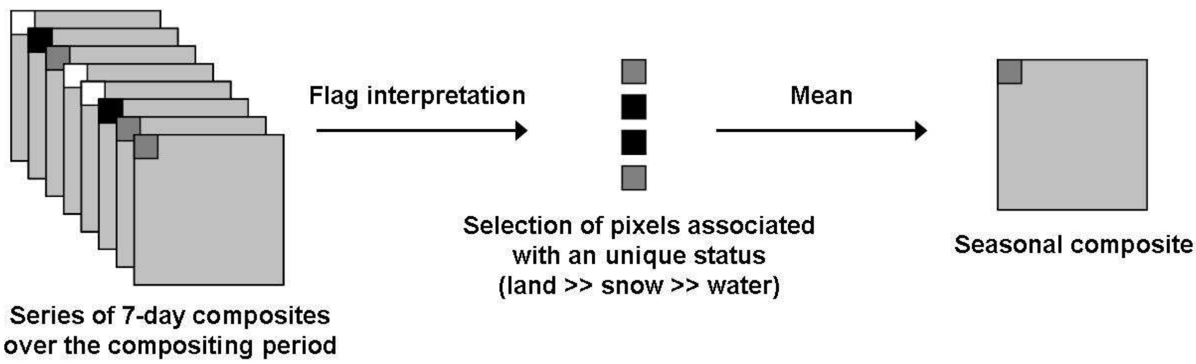

Figure 3-4: Mean compositing workflow to generate seasonal composites starting from 7-day composites

The pixel status associated with the surface reflectance in the output composite is recorded in a dedicated layer, as well as the contributing number of observations over the aggregation period.

3.2.3.2 *Multi-year seasonal composites*

As explained in the previous section 3.1.2, two strategies were developed to handle multi-year datasets: the first one classifies multi-year seasonal composites (“MY_S1”) and the second one relies on single-year seasonal composites (“MY_S2”). The decision to go for one or the other solution is made by stratum. This decision is documented in the LUT4 (Table 3-13); it contains the following fields:

- **NB_ST**, which lists the numbers (IDs) ranging between 1 and 22 and corresponding to each equal-reasoning area;
- **Name_ST**, which gives the names of each equal-reasoning area;
- **MY_Strategy**, specifying the multi-year strategy applicable for each stratum, MY_S1 and MY_S2 standing for the use of multi-year or single-year seasonal composites respectively.

Table 3-12: Definition of the multi-year strategy for the spectral classification (contained in LUT4)

NB_ST	NAME_ST	MY_STRATEGY
1	Polar areas	MY_S1
2	Canada	MY_S1
3	US	MY_S2
4	Central-America	MY_S1
5	Amazon	MY_S1
6	South-America	MY_S2
7	North-west Eurasia	MY_S1
8	North-east Eurasia	MY_S1
9	Eurasia	MY_S2
10	England	MY_S2
11	Mediterranea	MY_S2
12	Africa desert	MY_S2
13	North-Africa	MY_S2
14	Central Africa	MY_S1
15	South-Africa	MY_S2
16	Madagascar	MY_S1
17	Asia desert	MY_S2
18	South-east Asia	MY_S2
19	Japan	MY_S1
20	Indonesia	MY_S1
21	Australia	MY_S2
22	New-Zeland	MY_S2

For strata concerned by the MY_S1, multi-year seasonal composites are generated according to the same MC approach than the one underlying the generation of 7-day and single-year seasonal composites: single-year seasonal composites are aggregated into multi-year seasonal composites through an average operation weighted by the number of valid observations.

The aggregation is operated on a pixel basis and with pixels associated with a same status. Priority is given to the “land” status and then, come the “snow”, “water” and “cloud” statuses in this sequence. Pixel status was recorded in a specific flag band during the single-year seasonal composite generation, as well as the number of valid observations associated with each single-year seasonal composites.

- ***Algorithm assumptions and limitations***

None: the same processing chain is applicable to any type of satellite data, for any length of compositing period and for any temporal extent (single-year or multi-year).

- ***Input and output data***

Input and output data associated with this process are described in Table 3-13.

Table 3-13: Input and output data of the multi-year seasonal composites generation (multi-year approach)

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
L3_<sensor>_StartDate_EndDate_SR_<n> n = 1,..,15 (MERIS) sensor = FR for MERIS FR and RR for MERIS RR	Global raster of surface reflectance in each bands of the single-year seasonal composites (year information being included in the Date fields) GeoTiff format	IN	None	[0 ... 1]
L3_<sensor>_StartDate_EndDate_Status sensor = FR for MERIS FR and RR for MERIS RR	Global raster of pixel status associated with the surface reflectance in the single-year seasonal composites GeoTiff format	IN	None	[1 ... 7]
L3_<sensor>_StartDate_EndDate_NOBS sensor = FR for MERIS FR andRR for MERIS RR	Global raster counting the contributing observations in the single-year seasonal composites GeoTiff format	IN	None	[0 ... 500]
LUT4	Look-Up-Table indicating which strata require the generation of multi-year seasonal composites	IN	/	/
LUT3	Look-Up-Table indicating the seasonal compositing periods associated with each stratum in the classification chain	IN	/	/
L3_<sensor>_MY_StartYear_EndYear _StartMonthDay_EndMonthDay_SR_<n> n = 1,..,15 (MERIS)	Global raster of surface reflectance in each bands of the multi-year seasonal composites	OUT	None	[0 ... 1]

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
sensor = FR for MERIS FR and RR for MERIS RR	GeoTiff format			
L3_<sensor>_MY_StartYear_EndYear _StartMonthDay_EndMonthDay _Status sensor = FR for MERIS FR and RR for MERIS RR	Global raster of pixel status associated with the surface reflectance in the multi-year seasonal composites GeoTiff format	OUT	None	[1 ... 7]
L3_<sensor>_MY_StartYear_EndYear _StartMonthDay_EndMonthDay _NOBS sensor = FR for MERIS FR and RR for MERIS RR	Global raster counting the contributing observations in the multi-year seasonal composites GeoTiff format	OUT	None	[0 ... 500]

- ***Parameters***

No parameters are associated with this step.

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
algorithm MultiYear_Seasonal_Compositing is

     input: for each seasonal composite, averaged seasonal surface reflectance
     and NDVI values, associated final status and the number of observations
     which contributed to the seasonal average:
     L3_<sensor>_StartDate_EndDate_SR_<n> (where n = 1 … 15 for MERIS and sensor
     = FR for MERIS FR, RR for MERIS RR)
     L3_<sensor>_StartDate_EndDate_NDVI (where n = 1 … 15 for MERIS and sensor
     = FR for MERIS FR, RR for MERIS RR)
     L3_<sensor>_StartDate_EndDate_Status (where sensor = FR for MERIS FR, RR for
     MERIS RR)
     L3_<sensor>_StartDate_EndDate_NOBS (where sensor = FR for MERIS FR, RR for
     MERIS RR)
     Stratification layer and LUT 2
     LUT 3
     LUT 4


     output: for each stratum concerned by this step, associated multi-year
     seasonal composites made of surface reflectance and NDVI values, associated
     final status and the number of observations which contributed to the multi-
     year seasonal average:
     L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_EndMonthDay_SR_<n>
     L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_EndMonthDay_NDVI
     L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_EndMonthDay_Status
     L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_EndMonthDay_NOBS

     variables:
```

```
     L3_MY_Status = "no data"
     L3_MY_LandCount = 0
     L3_MY_SnowCount = 0
     L3_MY_WaterCound = 0
     L3_MY_CloudCount = 0
     L3_MY_CloudShadowCount = 0
     L3_MY_InvalidCount = 0
     L3_MY_NDVI_Land = 0
     L3_MY_NDVI_Snow = 0
     L3_MY_SR_<b>_Land = 0
     L3_MY_SR_<b>_Snow = 0
for each stratum NB_ST (1 to 22 – LUT 2) do:
     Read the field "MY_Strategy" of the LUT 4
     if MY_Strategy = MY_S1 do:
          Read the seasonal compositing period associated with the stratum NB_ST
          in the LUT 3
          Open the corresponding seasonal composites:
          L3_<sensor>_StartDate_EndDate_SR_<n>
          L3_<sensor>_StartDate_EndDate_NDVI
          L3_<sensor>_StartDate_EndDate_Status
          L3_<sensor>_StartDate_EndDate_NOBS
          for each seasonal compositing period S, do:
               for each pixel p do:
                    for each year Y (ranging 2002 and 2012), do:
                    /*read the status of the pixel p and updated the related
                    variables/*
                    if (L3_<sensor>_StartDate_EndDate_Status(p) = "LAND") do:
                         L3_MY_Status(p) = "LAND"
                         L3_MY_LandCount(p) = L3_MY_LandCount(p) +
                         L3_<sensor>_StartDate_EndDate_NOBS(p)
                         L3_MY_NDVI_Land(p) = L3_MY_NDVI_Land(p) +
                         (L3_<sensor>_StartDate_EndDate_NDVI(p) *
                         L3_<sensor>_StartDate_EndDate_NOBS(p))
                         for each band n do:
                              L3_MY_SR_Land_<n>(p) = L3_MY_SR_Land_<n>(p) +
                              (L3_<sensor>_StartDate_EndDate_SR_<n>(p) *
                              L3_<sensor>_StartDate_EndDate_NOBS(p))
                         end for n
                    elseif (L3_<sensor>_StartDate_EndDate_Status(p) = "SNOW") &
                    (L3_MY_Status(p) >< "LAND")) do:
                         L3_MY_Status(p) = "SNOW"
                         L3_MY_SnowCount(p) = L3_MY_SnowCount(p) +
                         L3_<sensor>_StartDate_EndDate_NOBS(p)
                         L3_MY_NDVI_Snow(p) = L3_MY_NDVI_Snow(p) +
                         (L3_<sensor>_StartDate_EndDate_NDVI(p) *
                         L3_<sensor>_StartDate_EndDate_NOBS(p))
                         for each band n do:
                              L3_MY_SR_Snow_<n>(p) = L3_MY_SR_Snow_<n>(p) +
                              (L3_<sensor>_StartDate_EndDate_SR_<n>(p) *
```

```
            L3_<sensor>_StartDate_EndDate_NOBS(p))
        end for n
    elseif ((L3_<sensor>_StartDate_EndDate_Status(p)= "WATER")
    & (L3_MY_Status(p) >< "LAND") & (L3_MY_Status(p) ><
    "SNOW")) do:
        L3_MY_Status(p) = "WATER"
        L3_MY_WaterCount(p) = L3_MY_WaterCount(p) +
        L3_<sensor>_StartDate_EndDate_NOBS(p)
    elseif ((L3_<sensor>_StartDate_EndDate_Status(p) = "CLOUD")
    & (L3_MY_Status(p) >< "LAND") & (L3_MY_Status(p) >< "SNOW")
    & (L3_MY_Status(p) >< "WATER")) do:
        L3_MY_Status(p) = "CLOUD"
        L3_MY_CloudCount(p) = L3_MY_CloudCount(p) +
        L3_<sensor>_StartDate_EndDate_NOBS(p)
    elseif ((L3_<sensor>_StartDate_EndDate_Status(p) = "CLOUD
    SHADOW") & (L3_MY_Status(p)>< "LAND") & (L3_MY_Status(p)
    >< "SNOW") & (L3_MY_Status(p)>< "WATER") & (L3_MY_Status(p)
    >< "CLOUD")) do:
        L3_MY_Status(p) = "CLOUD SHADOW"
        L3_MY_CloudShadowCount(p) = L3_MY_CloudShadowCount(p)
        + L3_<sensor>_StartDate_EndDate_NOBS(p)
    elseif ((L3_<sensor>_StartDate_EndDate_Status(p) =
    "INVALID") & (L3_MY_Status(p)>< "LAND") & (L3_MY_Status(p)
    >< "SNOW") & (L3_MY_Status(p)>< "WATER") &
    L3_MY_Status(p)>< "CLOUD") & (L3_MY_Status(p) >< "CLOUD
    SHADOW")) do:
        L3_MY_Status(p) = "INVALID"
        L3_MY_InvalidCount(p) = L3_MY_InvalidCount(p) +
        L3_<sensor>_StartDate_EndDate_NOBS(p)
    end if
    end for Y
    /* Averaging value based on status rules and writing
    results*/
    if (L3_MY_Status(p) = "LAND")
        L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
        _EndMonthDay_NOBS(p) = L3_MY_LandCount(p)
        L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
        _EndMonthDay_NDVI(p) = L3_MY_NDVI_Land (p) /
        L3_MY_LandCount(p)
        for each band n do:
            L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_
            EndMonthDay_SR_<n>(p) = L3_MY_SR_Land_<n>(p) /
            L3_MY_LandCount(p)
        end for n
    elseif (L3_MY_Status(p)= "SNOW")
        L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
        _EndMonthDay_NOBS(p) = L3_MY_SnowCount(p)
```

```
            L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
            _EndMonthDay_NDVI(p) = L3_MY_NDVI_Snow (p) /
            L3_MY_SnowCount(p)

             for each band n do:

                 L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_
                 EndMonthDay_SR_<n>(p) = L3_MY_SR_Snow_<n>(p) /
                 L3_MY_SnowCount(p)

             end for n

        elseif (L3_MY_Status(p)= "WATER")

           L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
           _EndMonthDay_NOBS(p) = L3_MY_WaterCount(p)

           L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
           _EndMonthDay_NDVI(p) = NaN

           for each band n do:

                L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_
                EndMonthDay_SR_<n>(p) = NaN
           end for n

      elseif (L3_MY_Status(p)= "CLOUD")

           L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
           _EndMonthDay_NOBS(p) = L3_MY_CloudCount(p)

           L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
           _EndMonthDay_NDVI(p) = NaN

           for each band n do:

                L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_
                EndMonthDay_SR_<n>(p) = NaN
           end for n

      elseif (L3_MY_Status(p) = "CLOUD SHADOW")

           L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
           _EndMonthDay_NOBS(p) = L3_MY_CloudShadowCount(p)

           L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
           _EndMonthDay_NDVI(p) = NaN

           for each band n do:

                L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_
                EndMonthDay_SR_<n>(p) = NaN
           end for n

     elseif (L3_MY_Status(p) = "INVALID")

           L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
           _EndMonthDay_NOBS(p) = L3_MY_InvalidCount(p)

           L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
           _EndMonthDay_NDVI(p) = NaN

           for each band n do:

                L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_
                EndMonthDay_SR_<n>(p) = NaN
           end for n

      end if

      L3_<sensor>_MY_StartYear_EndYear_StartMonthDay
      _EndMonthDay_Status(p) = L3_MY_Status(p)
```

```
                end for p
            end for S
    end if
end for NB_ST
```

Algorithm 3-2. Generation of multi-year multispectral seasonal composites

3.3 Detailed processing scheme of the machine learning spectral classification (step 1)

The machine learning classification is part of the supervised algorithms group. Any supervised classification procedure relies on the following steps:

- Gathering training data (i.e. spectral signatures for each land cover class of interest) as representative of the "real-world" as possible;
- Determining the type of the classifier to use – the classifier corresponding to the function which (i) analyzes the training data and (ii) predicts the output class of any input pixel by generalizing the training data to "unseen" situations;
- Defining the classification parameters to optimize the algorithm's performance.

With regard to the training dataset collection, two refinements are brought in this project with respect to a classical approach. First, training data are defined locally in order to take into account that a spectral signature of a given land cover class is not necessarily reliable over large extents. Indeed, it must be recognized that when working at large scales, each land cover label is probably associated with several spectral signatures (e.g. several spectral signatures for the crops to render maize or wheat for instance, etc). In order to face this problem, a locally-adjusted training dataset gathering (and thus classification approach) is developed. Each ***equal-reasoning area*** to classify is split into smaller moving windows which will stand for the ***classification areas***. The size of the classification areas varies according to the stratum. The regions inside which training dataset are gathered are called ***search areas*** and correspond to 240*240 km^2 areas centered on each classification area. This decoupling between classification and search areas allows accounting for the imperfection of the auxiliary reference dataset from which training samples are collected (e.g. for a false absence of a given land cover class over a specific region). If one land cover class is not represented in significant proportions within the classification area, it will still be included in the training data thanks to the larger extent of the search area. Furthermore, the overlap of the search areas contributes to the seamless change of the training data to avoid artefacts at the boundaries between two classification areas. Illustrations and relationships between equal-reasoning, classification and search areas are presented in Figure 3-5.

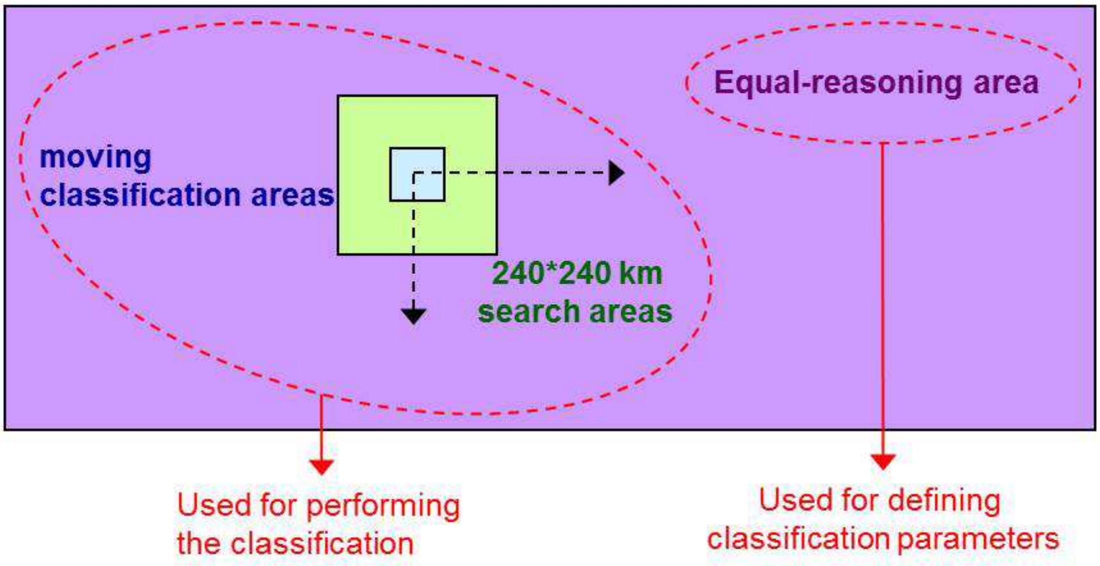

Figure 3-5: Illustration of the tree spatial units (equal-reasoning, classification and search areas) of the developed locally-adjusted supervised classification procedure

It should be clearly stated that these classification and search areas are completely distinct from the equal-reasoning areas defined in the stratification phase. Equal-reasoning areas allow adjusting the classification parameters while classification and search areas allow a local classification. The preparation of such training data is achieved in a ***first preliminary step*** of the supervised spectral classification procedure (see section 3.3.1).

The second refinement with regard to the training dataset collection consists in completing the classical training data (i.e. the representation of each land cover class through spectral signatures) with a priori information about the occurrence probability of each land cover class. This a priori information will be extracted from an auxiliary reference dataset at the spatial scale of the search areas. Indeed, while the local classification strategy allows accounting for local specificities, it also makes the algorithm more sensitive to spatial inconsistency of the training dataset. The occurrence probabilities computation at an intermediate scale is performed in a ***second preliminary step*** of the supervised spectral classification procedure (see section 3.3.2).

As for the machine learning classification algorithm in itself, it relies on the classical maximum likelihood principle. The algorithm and associated parameters are described in section □, along with the entire procedure.

3.3.1 Preliminary step 1 – Training dataset preparation

The training dataset shall provide representative spectral signatures for each land cover class of interest for the 240*240 km^2 search areas centered on the classification areas. It is derived from the reference land cover database, which is a key auxiliary dataset built in this project (section 3.2.1). This reference database consists of a set of global, regional and local reference land cover maps selected as the most accurate ones available for a given region, with the highest spatial resolution and with a CCI-compatible legend. Its spatial resolution is 20 m.

The workflow of this preliminary step for training dataset preparation is provided in Figure 3-6. It is organized in three steps:

- The first step consists in aggregating the reference LC database from 20 m to 300 m according to decision rules meaningful from a CCI legend and LCCS compatibility;
- The second step consists in applying a morphological filter to the reference database in order to ensure training dataset as "pure" as possible (i.e. not contaminated by "border effects");
- The third step consists in extracting, for each eroded land cover class, representative spectral signatures at the search area level.

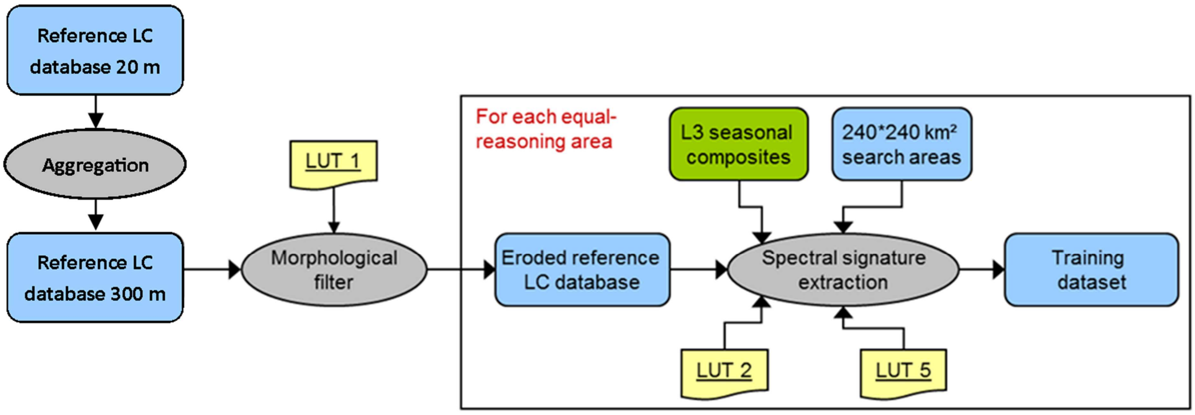

Figure 3-6: Activity diagram illustrating the training dataset preparation: (1) the reference land cover database is aggregated to 300 m, (2) the reference land cover database is eroded and (3) spectral signatures are extracted or each eroded class

The aggregation step is the process of resampling the reference LC database at 20 m to 300 m. It relies on pre-defined rules that associate proportions of the 20-m LC labels to reference LC labels in the resulting 300 m x 300 m pixel. Those rules aim at mimicking the proportions of LC types that form the definition of the global LC map itself. The underlying hypothesis of this aggregation step is that each 20-m pixel is labelled according to the LCCS legend with minimum ambiguity between similar classes.

The morphological filter is based on the erosion principle, i.e removing pixels along the boundaries of each land cover class. However, eroding each of the class with the same number of pixels would erase the thin classes from the reference. The erosion was therefore designed in such a way that at least one pixel remains from each group of adjacent pixels belonging to the same class.

The morphological filter keeps only the pixels with the most neighbours of the same class. A fixed neighbourhood of 3*3 pixels was used for all strata. The filter is composed of two passes. The first pass counts the number of pixels from the same class than the central pixel (example on Figure 3-7 B). The second pass erases the label of the central pixel if it has less neighbours than another pixel of the same class in the neighbourhood (example on Figure 3-7 D). This specificity allows conserving isolated pixels in the reference layer if there is no larger group in the neighbourhood, which is not the case with a standard erosion filter (example on Figure 3-7 C).

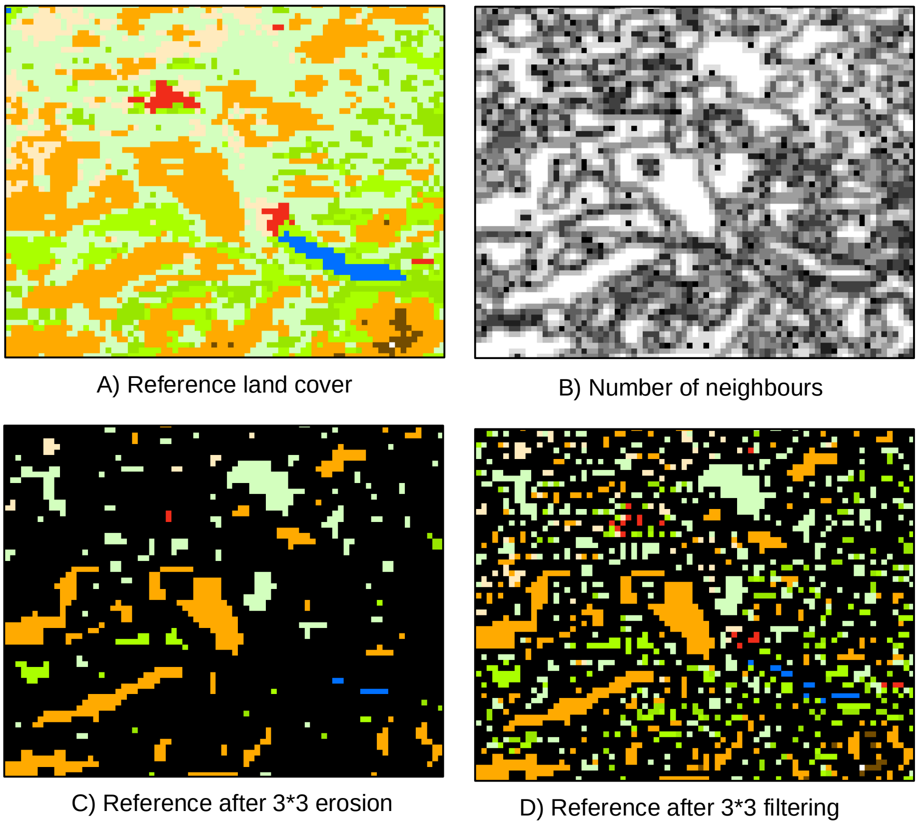

Figure 3-7: Results of morphological erosion and majority neighbours filtering in a rural landscape

This composite filter results in a new raster file which contains the new eroded reference land cover database. This new database is then used to extract spectral signature for each class and thus build training dataset.

This second step is operated at the 240*240 km^2 search area level while running with the generic parameters of the machine learning algorithm which are defined at the stratum level. These parameters are contained in the LUT 5 and include, for each stratum, the L3 seasonal composites and the spectral channels from which extracting the spectral signatures. Inside each search area, for each class and from the specified seasonal composites and spectral channels, the algorithm extracts reflectance values to define representative spectral distributions that will serve as training dataset.

The LUT 5 contains the following fields:

- **NB_ST**, which lists the numbers (IDs) ranging between 1 and 22 and corresponding to each equal-reasoning area;
- **Startdate_Season(i)**, specifying the exact date which marks the start of the seasonal composite i (i ranging from 1 to 3 according to the stratum);
- **Enddate_Season(i)**, specifying the exact date which marks the end of the seasonal composite i (i ranging from 1 to 3 according to the stratum);

- **Channels**, specifying the spectral channels to use as input for the machine learning classification algorithm.

Table 3-14: Parameters associated with the machine learning classification algorithm (contained in LUT5)

NB_ST	CHANNELS
1	3, 5, 7, 9, 10, 13
2	3, 5, 8, 9, 11, 13
3	5, 7, 14
4	5, 6, 9, 10, 11, 15
5	2, 3, 6, 7, 9, 15
6	5, 7, 14
7	3, 5, 7, 9, 10, 13
8	3, 5, 7, 9, 11, 13
9	5, 7, 14
10	5, 7, 14
11	5, 7, 14
12	6, 7, 8, 9, 10, 11
13	8, 9, 10, 11, 14, 15
14	2, 3, 8, 9, 14, 15
15	3, 6, 9, 11, 13, 15
16	1, 2, 9, 10, 11
17	1, 4, 6, 9, 10, 11
18	5, 7, 14
19	3, 4, 9, 10, 11, 12
20	4, 7, 11, 12, 14, 15
21	5, 7, 14
22	5, 6, 10, 15

- ***Algorithm assumptions and limitations***

None

- ***Input and output data***

Input and output data associated with the application of the aggregation of the reference LC database, the morphological filter on the reference and the generation of the training dataset (through the extraction of spectral signature by class) are described in Table 3-16, Table 3-17 and Table 3-17, respectively.

Table 3-15: Input and output data of the 1st preliminary step of the supervised spectral classification, for the aggregation of the 20-m reference to 300 m.

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
Ref20_LC	Global reference land cover database at 20 m, where each pixel is associated with a land cover class of the LCCS CCI-legend through an ID	IN	Short	[0 .. 255]
LUT 1	Look-up table describing the ID of each land cover class	IN	/	/
Ref300_LC	Global reference land cover database aggregated at 300 m, where each pixel is associated with a land cover class of the LCCS CCI-legend through an ID	OUT	Short	[0 … 255]

Table 3-16: Input and output data of the 1st preliminary step of the supervised spectral classification, for the application of the morphological filter on the reference.

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
Ref300_LC	Global reference land cover database at 300 m, where each pixel is associated with a land cover class of the LCCS CCI-legend through an ID	IN	Short	[0 .. 255]
LUT 1	Look-up table describing the ID of each land cover class	IN	/	/
Ref_LC_Training	Reference land cover database, processed with the morphological filter. Each pixel is associated with a land cover class of the LCCS CCI-legend or with a no data value through an ID	OUT	Short	[0 … 255]

Table 3-17: Input and output data of the 1st preliminary step of the supervised spectral classification, for the spectral signature extraction

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
L3_StartDate_EndDate_SR_<n> n = 1,..,15 (MERIS)	Global raster of surface reflectance in each bands of the seasonal composites GeoTiff format	IN	None	[0 … 1]
Ref_LC_Training	Reference land cover database, processed with the morphological filter. Each pixel is associated with a land cover class of the LCCS CCI-legend or	IN	None	[0 … 255]

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
	with a no data value through an ID			
GRID_240	Raster grid for each stratum defining the 240*240 km^2 search areas localization	IN	None	[0 … ~60000]
LUT 2	Look-Up-Table describing the location of the 22 strata	IN	/	/
LUT 5	Look-Up-Table describing the parameters of the machine learning algorithm	IN	/	/
ROI_<NB_LAB>_<GRID>_StartDate_EndDate_SR - NB_LAB representing each class of the LCCS legend - GRID representing the search area - StartDate and EndDate defining the source seasonal composite	Pure training dataset for each class ("NB_LAB") and for each search area ("GRID"), consisting in representative reflectance values distributions in specific seasonal composites (defined by "StartDate" and "EndDate")	OUT	None	[0 … 1]

- ***Parameters***

Table 3-18 provides the parameters needed to run the morphological filter. The parameters needed to extract the spectral signatures correspond to the generic ones defined for the machine learning algorithm and are thus contained in the LUT 5.

Table 3-18: Parameter needed in the 1st preliminary step of the supervised spectral classification, for the application of the morphological filter on the reference

PARAMETERS	DESCRIPTION	INTENT (IN, OUT, INOUT)	FORMAT	RANGE
k	Size of the window inside which the filter is applied	IN	Short	[0 … 255]

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
algorithm REF20_LC_Aggregation is

input:
REF20_LC: reference land cover layer at 20m
LUT1
```

```
output:
REF300_LC: reference land cover layer aggregated at 300m

/*Resample the REF20_LC to 300 m following the CCI LC legend stored in LUT1 */
/*Save results*/
Save REF300_LC
```

Algorithm 3-3. Aggregation algorithm to retrieve the label of the reference land cover database at 300 m (algorithm associated with the preliminary step 1 of the spectral supervised classification algorithm)

```
algorithm Morphological_filter is
     input:
     REF300_LC: reference land cover layer aggregated at 300 m
     LUT1
     output:
     REF_LC_Eroded: eroded reference land cover layer
     variables:
     i = 0 (“i” being the row index for the central pixel)
     j = 0 (“j” being the column index for the central pixel)
     m = 0 (“m” being the row index for the 3*3 neighborhood window)
     n = 0 (“n” being the column index for the 3*3 neighborhood window)
     COUNT = 0



/*First loop to count pixels associated with a specific label NB_LAB inside the
3*3 neighborhood window*/
for each pixel p(i,j) do:
     Read values (NB_LAB) in the reference land cover layer REF300 and the
     corresponding label (LABEL) in LUT 1
     ID(p(i,j)) = “NB_LAB”
     Read values (NB_LAB) in the reference land cover layer REF300 for each pixel
     inside the 3*3 neighborhood and increment the counter COUNT if pixels have
     the same value NB_LAB than the central pixel p(i,j):
     for m = -4:4
           for n = -4:4
                 if ID(p(i+m,j+n)) = “NB_LAB”
                       COUNT = COUNT+1
                 end if
           end for n
     end for m
     Write intermediate result for the central pixel p(i,j):
     COUNT(p(i,j)) = “COUNT”
end for p
/*Second loop to select the most represented pixel inside the 3*3 window*/
for each pixel p(i,j) do:
     Read values (NB_LAB) in the reference land cover layer REF300 and the
```

```
    corresponding label (LABEL) in LUT 1
    ID(p(i,j)) = “NB_LAB”

    Initialize intermediate variable:
    NewID(p(i,j)) = ”NB_LAB”

    Read the count of this value (= intermediate result of loop 1):
    COUNT(p(i,j)) = “COUNT”

    Look at the maximum of the COUNT value for the pixels that have the same
    ID than the central pixel p(i,j). If one of these count value is larger than
    the count of the central pixel, set the NewID of the central pixel p(i,j) to
    NoData:


    for m = -4:4

          for n = -4:4

                if ID(p(i+m,j+n)) = “NB_LAB”

                      /* if there is another pixel of this class with more
                      neigbours than the central pixel, set central pixel to
                      NoData */

                      if COUNT(p(i+m,j+n)) > COUNT(p(i,j))

                            NewID(p(i,j)) = NoData

                            /* We can exit the loop in the neigbborhood here*/

                      end if

                end if

          end for m

    end for n

end for p
```

Algorithm 3-4. Morphological filter application on raster file to erode clusters (algorithm associated with the preliminary step 1 of the spectral supervised classification algorithm)

```
algorithm Spectral_Signature_Extraction is

    input:
    MERIS FR and RR and SPOT-VGT single-year and multi-year seasonal composites
    used to extractreflectance values:
    L3_<sensor>_StartDate_EndDate_SR_<n>
    L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_EndMonthDay_SR_<n>
    GRID_240: 240*240 km² grid used to define the classification areas
    REF_LC_Eroded: eroded reference land cover layer
    Stratification layer and LUT 2
    LUT 4
    LUT 5

    output:
    ROI_<NB_LAB>_<GRID>_<sensor>_MY_StartDate_EndDate_SR<n>: pure training
    dataset for each class (“NB_LAB”) and for each search area (“GRID_240”),
    consisting in representative reflectance values distributions in multi-year
    seasonal composites (defined by “StartDate” and “EndDate”)
    ROI_<NB_LAB>_<GRID>_<sensor>_SY<Y>_StartDate_EndDate_SR<n>: pure
    training dataset for each class (“NB_LAB”) and for each search area
```

```
    ("GRID_240"), consisting in representative reflectance values distributions
    in single-year seasonal composites (defined by "StartDate" and "EndDate")

/*Organization of the ROI preparation step, based on the multi-year strategy*/
for each stratum NB_ST (1 to 22 – LUT 2) do:
    Read the field "MY_Strategy" of the LUT 4
    if MY_Strategy = MY_S1 do:
        run Spectral_Signature_Extraction_S1 algorithm
    elseif MY_Strategy = MY_S2 do:
        run Spectral_Signature_Extraction_S2 algorithm
    end if
end for NB_ST
/*Run the Spectral_Signature_Extraction_S1 algorithm, for strata concerned by the
MY_S1*/
for each stratum NB_ST do:
    Read the input data corresponding to NB_ST in the LUT 5 and
    identifies:
    - the suitable seasonal composites (Si and Ei defining the start and end
    dates of the compositing period; i being the number of composites specified
    for each stratum)
    - the suitable spectral channels CHij (j being the channels for the
    composite i)
    Open the corresponding channels
    for each search area GRID_240 do:
        for each class NB_LAB in the reference land cover REF_LC_Eroded do:
            for each pixel p, do:
                Read input data (seasonal composite defined by Si and Ei,
                spectral channels defined by CHij_BASE)
                Append the ROI of the class NB_LAB with the corresponding
                reflectance values:
                ROI_<NB_LAB>_<GRID>_<sensor>_MY_<Si,Ei>_SR<n> =
                [ROI_<NB_LAB>, Ref(p,Si,Ei,CHij)]
            end for p
        end for NB_LAB
    end for GRID_240
end for NB_ST
/*Run the Spectral_Signature_Extraction_S2 algorithm, for strata concerned by the
MY_S2*/
for each stratum NB_ST do:
    Read the input data corresponding to NB_ST in the LUT 5 and
    identifies:
    - the suitable seasonal composites (Si and Ei defining the start and end
    dates of the compositing period; i being the number of composites specified
    for each stratum)
    - the suitable spectral channels CHij (j being the channels for the
```

```
     composite i)
     Open the corresponding channels

     for each year Y, do:

          for each search area GRID_240 do:

               for each class NB_LAB in the reference REF_LC_Eroded do:

                    for each pixel p, do:

                         Read input data (seasonal composite defined by Si and
                         Ei, spectral channels defined by CHij)

                         Append the ROI of the class NB_LAB with the
                         corresponding reflectance values:
                         ROI_<NB_LAB>_<GRID>_<sensor>_SY<Y>_<Si,Ei>_SR<n> =
                         [ROI_<NB_LAB>, Ref(p,Si,Ei,CHij)]

                    end for p

               end for NB_LAB

          end for GRID_240

     end for Y

end for NB_ST
```

Algorithm 3-5. Spectral signature extraction to generate training dataset (algorithm associated with the preliminary step 1 of the spectral supervised classification algorithm)

3.3.2 Preliminary step 2 – Computation of land cover classes' occurrence probabilities at the stratum level

As already mentioned in the introduction, the machine learning classification is locally adjusted: the algorithm is run within classification areas, using training dataset collected within 240*240 km^2 search areas. In order to minimize the sensitivity of the algorithm to the possible spatial inconsistency of the training dataset, the algorithm also makes use of a priori occurrence probabilities for each land cover class defined at the spatial scale of the search areas.

The occurrence of each land cover class is computed inside each search area through a comparison with the reference land cover database. These occurrences will serve as a priori information for the machine learning algorithm and complete the classical training dataset made of spectral signatures.

- ***Algorithm assumptions and limitations***

None

- ***Input and output data***

Table 3-19 presents the input and output data associated with the step of land cover classes' occurrence computation.

Table 3-19: Input and output data of the 2nd preliminary step of the supervised spectral classification, for the LC classes occurrence computation

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
REF_LC	Reference land cover layer where each pixel is associated with a land cover class through an ID	IN	None	[0 … 255]
GRID_240	Raster grid for each stratum defining the 240*240 km^2 search areas localization	IN	None	[0 … ~60000]
LUT 1	Look-Up-Table describing the CCI LCCS land cover legend	IN	/	/
OCC_<NB_LAB>_<GRID> - NB_LAB representing each class of the LCCS legend - GRID representing the search area	Pure training dataset for each class ("NB_LAB") and for each search area ("GRID"), consisting in representative reflectance values distributions in specific seasonal composites (defined by "StartDate" and "EndDate")	OUT	None	[0 … 1]

- ***Parameters***

No specific parameter is required.

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
algorithm Class_Occurrence is

     input:
     GRID_240: 240*240 km² grid used to define the search areas
     REF_LC: reference land cover layer
     Stratification layer and LUT 2
     LUT 1

     output:
     OCC_<NB_LAB>_<GRID_240>: text file (one for each search area GRID_240)
     listing the occurrence of each land cover class NB_LAB inside each search
     area GRID_240

for each stratum NB_ST (1 to 22 – LUT 2) do:

     for each search area in the input data GRID_240


          for each for each pixel p in the search area, do:

               Read values (NB_LAB) in the reference land cover layer REF_LC and
               the corresponding label (LABEL) in LUT 1
```

```
            Write the result in a temporary file
        end for p
        Compute the occurences of each value NB_LAB
       Write the results in the corresponding text file:
       OCC_<NB_LAB>_<GRID_240>

     end for each GRID_240
end for NB_ST
```

Algorithm 3-6. Class frequency computation inside search areas (algorithm associated with the preliminary step 2 of the spectral supervised classification algorithm)

3.3.3 Machine learning spectral classification

This section describes the machine learning algorithm in itself, and more precisely the classifier it uses to predict the output class of all pixels.

The algorithm makes use of the most common classifier, which is the Maximum Likelihood (ML), and relies on a statistical approach. The ML classifier assumes that each spectral class can be described by a multivariate normal distribution. The ML algorithm therefore takes advantage of both the mean vectors and the multivariate spreads of each class, and would be able to also identify elongated classes. More precisely, the Probability Density Function (PDF) of each class is estimated under a Gaussian assumption. It is rather simple and rapid but it assumes that there is only one population per class (i.e. a unique spectral signature by land cover class). This assumption seems realistic since the algorithm is applied at a local scale.

ML classification is a statistical decision criterion to assist in the classification of overlapping signatures; pixels are assigned to the class of highest probability.

As already mentioned, the supervised algorithm is run with parameters defined at a larger scale while being applied at a more local scale. On one hand, the input EO data (i.e. seasonal composites in a definite number of spectral channels) are defined at the ***equal-reasoning area*** level. On the other hand, the algorithm is processed using moving ***classification areas*** using (i) training data valid within 240*240 km^2 ***search areas*** centered on the classification areas and (ii) a priori land cover classes' occurrence probabilities defined at the same scale than the ***search areas***.

Classification areas inside which the machine learning algorithm is run are not the same for all strata. This information is included in the LUT 6 (Table 3-20), which includes the following fields:

- **NB_ST**, which indicates the number of the stratum;
- **CLASSIF_AREA,** specifying the size (in km*km) of the areas in which running the ML algorithm.

Table 3-20: Parameters associated with the classification areas of the machine learning algorithm (contained in LUT 6)

NB_ST	CLASSIF_AREA
1	120 *120
2	120 *120
3	120 *120
4	120 *120
5	120 *120
6	60 *60
7	120 *120
8	120 *120
9	120 *120
10	120 *120
11	60 *60
12	60 *60
13	60 *60
14	120 *120
15	60 *60
16	120 *120
17	120 *120
18	120 *120
19	120 *120
20	120 *120
21	120 *120
22	120 *120

The principle of this machine learning classification strategy is illustrated in Figure 3-8.

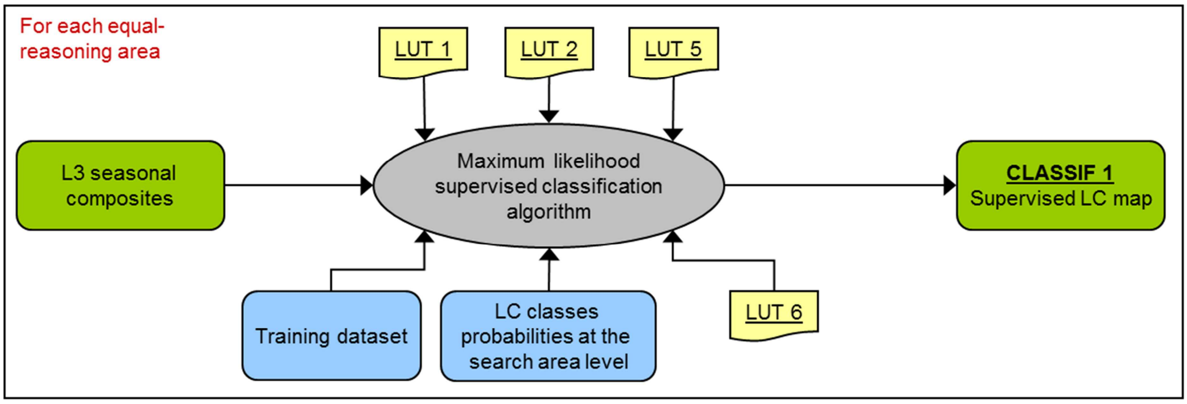

Figure 3-8: Activity diagram illustrating the Gaussian Maximum Likelihood supervised classification algorithm developed in the CCI-LC project

The machine learning classification algorithm is applied on L3 seasonal composites independently for each equal-reasoning area (NB_ST).

The principle of the maximum likelihood classification relies on the Bayes theorem. In the context of the supervised classification of remote sensing data, the likelihood of a land cover class for a given set of spectral values can be derived from the a priori probability and the distribution of the spectral values for this class. The computation of the a priori probability is described in the second preliminary step; the parameters of the distribution of the spectral values are derived from the training datasets. For each pixel, the label is assigned to the class with the largest probability. It is worth noting that the largest probability is stored together with the label in order to be used as a quality flag.

As an output, the machine learning classification algorithm creates, for each stratum (NB_ST), a raster file (CLASSIF1) where each pixel is associated to a land cover class through the NB_LAB identifier (see LUT 1 in Table 3-10). It also produces an auxiliary layer providing the classification probability associated with each pixel (CL1_PROB). This probability informs about the confidence in the classification output.

Local classification could lead to tiling artefacts because the classifiers change for each classification area. However, this is mitigated by the use of a search area that is larger than the classification area and a bilinear interpolation of the a priori probabilities in a continuous way. While the training of the classifier based on spectral signature is locally based, the a priori brings some consistency and the classifiers parameters are seamlessly changing. The size of the classification area was set by tuning the method in order to remove tiling artefact with optimal processing time. In addition, the tiling artefacts can also be avoided by shifting the starting point of the classification areas year after year.

- ***Algorithm assumptions and limitations***

The Gaussian ML approach underlying the supervised classification algorithm makes the assumption that each spectral class can be described by a Gaussian distribution. It assumes that classes in the input data have a Gaussian distribution and that signatures were well selected; this is not always a safe assumption.

- ***Input and output data***

Input and output data associated with this machine learning algorithm are described in Table 3-21.

Table 3-21: Input and output data of the step 1a of the classification chain, i.e. the spectral machine learning classification

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
L3_StartDate_EndDate_SR_<n> n = 1,..,15 (MERIS)	Global raster of surface reflectance in each bands of the seasonal composites GeoTiff format	IN	None	[0 … 1]
ROI_<NB_LAB>_<GRID>_StartDate_EndDate_SR - NB_LAB representing each class of the LCCS legend - GRID representing the search area - StartDate and EndDate defining the source seasonal composite	Pure training dataset for each class ("NB_LAB") and for each search area ("GRID"), consisting in representative reflectance values distributions in specific seasonal composites (defined by "StartDate" and "EndDate")	IN	None	[0 … 1]
OCC_<NB_LAB>_<GRID> - NB_LAB representing each class of the LCCS legend - GRID representing the search area	Pure training dataset for each class ("NB_LAB") and for each search area ("GRID"), consisting in representative reflectance values distributions in specific seasonal composites (defined by "StartDate" and "EndDate")	IN	None	[0 … 1]
LUT 1	Look-up table describing the ID of each land cover class	IN	/	/
LUT 2	Look-Up-Table describing the location of the 22 strata	IN	/	/
LUT 4	Look-Up-Table indicating which strata require the generation of multi-year seasonal composites	IN	/	/
LUT 5	Look-Up-Table describing the parameters of the machine learning algorithm	IN	/	/
LUT 6	Look-Up-Table providing the	IN	/	/

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
	classification areas for each stratum			
CLASSIF1	Land cover map resulting from the supervised classification algorithm, where each pixel is associated with a land cover class through an ID	OUT	None	[0 … 255]
Code_CLASSIF1	Classification probability associated with the label selected for each pixel	OUT	None	0 … 1

- ***Parameters***

The classifier can be tuned to adjust the size of the processing window, the size of the search window and the weight (W) of the a priori information with respect to the spectral information.

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
algorithm Machine_Learning_Classification is

     input:
     MERIS single-year and multi-year seasonal composites used to extract
     reflectance values:
     L3_<sensor>_StartDate_EndDate_SR_<n>
     L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_EndMonthDay_SR_<n>

     ROI_<NB_LAB>_<GRID_240>_<sensor>_MY_StartDate_EndDate_SR<n>: pure training
     dataset for each class ("NB_LAB") and for each search area ("GRID_240"),
     consisting in representative reflectance values distributions in multi-year
     seasonal composites (defined by "StartDate" and "EndDate")
     ROI_<NB_LAB>_<GRID_240>_<sensor>_SY<Y>_StartDate_EndDate_SR<n>: pure
     training dataset for each class ("NB_LAB") and for each search area
     ("GRID_240"), consisting in representative reflectance values distributions
     in single-year seasonal composites (defined by "StartDate" and "EndDate")

     OCC_<NB_LAB>_<GRID_240>: text file (one for each search area GRID_240)
     listing the occurrence of each land cover class NB_LAB inside each search
     area GRID_240

     Stratification layer and LUT 2
     GRID_240: 240*240 km2 grid used to define the classification areas
     LUT 4
     LUT 5
     LUT 6

     output:
     CLASSIF_1_MY: raster file resulting from the spectral supervised
     maximum likelihood classification where each pixel is associated with a land
     cover class (described with an ID = NB_LAB and a name = LABEL). It results
```

```
     from the algorithm applied on multi-year seasonal composites
     Code_CLASSIF1_MY: raster provididng for each pixel the probability
     that the pixel is well classified. It results from the algorithm applied on
     multi-year seasonal composites
     CLASSIF_1_SY<Y> (Y ranging from 1 to 10): raster file resulting from
     the spectral supervised maximum likelihood classification where each pixel
     is associated with a land cover class (described with an ID = NB_LAB and a
     name = LABEL). It results from the algorithm applied on single-year seasonal
     composites
     Code_CLASSIF1_SY<Y> (Y ranging from 1 to 10): raster provididng for
     each pixel the probability that the pixel is well classified. It results
     from the algorithm applied on single-year seasonal composites

     intermediate variables:
     PRIOR(i,j) = a priori information used in the algorithm, which comes from
     the preliminary step 2. It consists in the land cover classes probability
     inside the search area GRID_240(i,j).

/*Organization of the supervised step, based on the multi-year strategy*/

for each stratum NB_ST (1 to 22 – LUT 2) do:

     Read the field "MY_Strategy" of the LUT 4

     if MY_Strategy = MY_S1 do:

          run Supervised_ML_S1 algorithm

     elseif MY_Strategy = MY_S2 do:

          run Supervised_ML_S2 algorithm

     end if

end for NB_ST


/*Run the Supervised_ML_S1 algorithm, for strata concerned by the MY_S1*/

for each stratum NB_ST do:

     Read the size of the classification area in LUT 6

     for each classification area CLASSIF_AREA do:

          Read the input data corresponding to NB_ST in the LUT 5:

          - STARTDATE_Si = exact date which marks the start of the seasonal
          composite i to use in the unsupervised classification, i being the
          number of composites specified for each stratum

          - ENDDATE_Si = exact date which marks the end of the seasonal
          composite i to use in the unsupervised classification, i being the
          number of composites specified for each stratum

          - CHij = j channels for the composite i

          Open the corresponding channels

          /*Compute the parameters of the Gaussian distribution for each ROI*/

          Identify the corresponding search area in GRID_240

          ROI_<NB_LAB>_<CLASSIF_AREA>= ROI_<NB_LAB>_<GRID_240>_<sensor>_
          MY_StartDate_EndDate_SR<n>

          for each training dataset ROI_<NB_LAB>_<GRID>, do:

               /*Calculate the covariance*/
```

```
            COV_ROI_<NB_LAB> = covariance of ROI_<NB_LAB>_<GRID>

            /*Calculate the inverse of the covariance*/
            ICOV_ROI_<NB_LAB> = inverse of COV_ROI_<NB_LAB>

            /*Calculate the mean of the ROI*/
            MEAN_ROI_<NB_LAB> = mean of ROI_<NB_LAB>_<GRID>

            /*Calculate the final statistic* ("powel" being the power and
            "numel" being the number of elements*/
            NORMC_ROI_<NB_LAB> = pow(2*PI,
            numel(CHij)/2)*determinant(COV_ROI_<NB_LAB>)

        end for ROI_<NB_LAB>_<GRID>

        /*Perform the classification for each pixel*/

        for each pixel p(i,j) in CLASSIF_AREA, do:

            /*Read the a priori information about class occurrence*/
            PRIOR(i,j)  = read the value at pixel p(i,j) in the text file
            OCC_<NB_LAB>_<GRID_240>

            /*Compute the likelihood at pixel location for each class*/
            /*Loop through the classes NB_LAB*/
            for k = 1:NB_LAB

                PRIOR(k) = read the frequency of class k in the classes
                occurrence file OCC_<NB_LAB>_<GRID_240>

                REF(i,j) = read the reflectance values at pixel p(i,j) in the
                input L3 seasonal composites that have been opened

                /*Compute the Probability Density Function*/
                PDFVAL(k) = exp(-0.5 * ((SPEC - MEAN_ROI_<k>) *
                ICOV_ROI_<k> * transpose( SPEC - MEAN_ROI_<k>))
                PDFVAL(k) = PDFVAL(k) / NORMC_ROI_<NB_LAB>

            end for k

            /*Get label and probability for the maximum likelihood value*/
            ("argmax" being the position of the maximum in a list)

            LABEL = argmax( PRIOR(i,j)* ^W PDFVAL(i,j))
            PROBVAL = PDFVAL(LABEL)/ sum(PDFVAL)

            Write LABEL in the output CLASSIF1_MY
            Write PROBVAL in the output Code_CLASSIF1_MY

        end for p(i,j)

    end for GRID

end for NB_ST


/*Run the Supervised_ML_S2 algorithm, for strata concerned by the MY_S2*/

for each stratum NB_ST do:

    Read the size of the classification area in LUT 6

    for each classification area CLASSIF_AREA do:

        Read the input data corresponding to NB_ST in the LUT 5:

        - STARTDATE_Si = exact date which marks the start of the seasonal
        composite i to use in the unsupervised classification, i being the
        number of composites specified for each stratum
```

```
- ENDDATE_Si = exact date which marks the end of the seasonal
composite i to use in the unsupervised classification, i being the
number of composites specified for each stratum

- CHij_BASE = j channels for the MERIS (and possibly SPOT-VGT)
composite i

for each year Y, do:

     Open the corresponding channels

     /*Compute the parameters of the Gaussian distribution for each
     ROI*/

     Identify the corresponding search area in GRID_240

     ROI_<NB_LAB>_<GRID>= ROI_<NB_LAB>_<GRID_240>_<sensor>_
     SY<Y>_StartDate_EndDate_SR<n>

     for each training dataset ROI_<NB_LAB>_<GRID>, do:

          /*Calculate the covariance*/
          COV_ROI_<NB_LAB> = covariance of ROI_<NB_LAB>_<GRID>

          /*Calculate the inverse of the covariance*/
          ICOV_ROI_<NB_LAB> = inverse of COV_ROI_<NB_LAB>

          /*Calculate the mean of the ROI*/
          MEAN_ROI_<NB_LAB> = mean of ROI_<NB_LAB>_<GRID>

          /*Calculate the final statistic* ("powel" being the power and
          "numel" being the number of elements*/
          NORMC_ROI_<NB_LAB> = pow(2*PI,
          numel(CHij)/2)*determinant(COV_ROI_<NB_LAB>)

     end for ROI_<NB_LAB>_<GRID>

     /*Perform the classification for each pixel*/

     for each pixel p(i,j) in GRID_50, do:

          /*Read the a priori information about class occurrence*/
          PRIOR(i,j)  = read the value at pixel p(i,j) in the text file
          OCC_<NB_LAB>_<GRID_240>

          /*Compute the likelihood at pixel location for each class*/
          /*Loop through the classes NB_LAB*/
          for k = 1:NB_LAB

               PRIOR(k) = read the frequency of class k in the classes
               occurrence file OCC_<NB_LAB>_<GRID_240>

               REF(i,j) = read the reflectance values at pixel p(i,j)
               in the input L3 seasonal composites that have been
               opened

               /*Compute the Probability Density Function*/
               PDFVAL(k) = exp(-0.5 * ((SPEC - MEAN_ROI_<k>) *
               ICOV_ROI_<k> * transpose( SPEC - MEAN_ROI_<k>))
                PDFVAL(k) = PDFVAL(k) / NORMC_ROI_<NB_LAB>

          end for k

          /*Get label + probability for the maximum likelihood value*/
          ("argmax" being the position of the maximum in a list)

          LABEL = argmax( PRIOR(i,j)*PDFVAL(i,j))
          PROBVAL = PDFVAL(LABEL)/ sum(PDFVAL)

          Write LABEL in the output CLASSIF1_SY<Y>
```

```
                    Write PROBVAL in the output Code_CLASSIF1_SY<Y>
                end for p(i,j)
            end for Y
        end for GRID
end for NB_ST
```

Algorithm 3-7. Supervised Maximum Likelihood classification algorithm

3.3.4 Multi-temporal approach

For strata concerned by the MY_S2 (see Table 3-12 and section 3.1.2), the classification algorithm has to be run multiple times (one for each year of interest) and the multiple single-year land cover maps have to be aggregated in a multi-year land cover map.

A superposition between the single-year land cover products (raster files CLASSIF1_<Year>) is operated. For each pixel, a histogram of class frequency is computed, which is then interpreted according to the majority voting principle. As a result, a multi-year land cover class (i.e. a unique class number NB_LAB and name LABEL) is associated with each pixel and an output multi-year land cover product (raster file CLASSIF1_MY) is generated.

The histogram interpretation process also associates each pixel with an ambiguity code (AMB_CODE_MY) that quantifies the frequency of occurrence of the land cover class (NB_LAB and LABEL) finally associated with the pixel according to the decision rules. This code can stand for an indicator of the land cover label.

- ***Algorithm assumptions and limitations***

None

- ***Input and output data***

Input and output data associated with the aggregation of single-year land cover maps are described in Table 3-22.

Table 3-22: Input and output data of the aggregation of single-year land cover maps (multi-year approach)

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
CLASSIF1_SY_<Year>	Raster at the stratum level resulting from the spectral classification algorithm run on single-year seasonal composites, where each pixel is associated with a land cover class ID (NB_LAB)	IN	None	[0 … 255]

Code_CLASSIF1_SY_<Year>	Classification probability associated with the label selected for each pixel	OUT	None	0 … 1
LUT 1	Look-Up-Table describing the CCI LCCS land cover legend	IN	/	/
CLASSIF1_Histo	Text file (one for each equal-reasoning area) containing for each pixel the land cover classes frequency	INOUT	Long	[0 … 100]
CLASSIF1_MY	Land cover map resulting from the aggregation of single-year spectral land cover maps, where each pixel is associated with a land cover class through an ID	OUT	None	[0 … 255]
Code_CLASSIF1_MY	Classification probability associated with the label selected in the land cover map resulting from the aggregation of single-year spectral land cover maps	OUT	None	0 … 1
AMB_CODE_CLASSIF1_MY	Frequency of a same land cover class observed over the multiple aggregated years, thus reflecting, at the pixel level, the reliability of the CLASSIF1_MY	OUT	None	0 … 10

- ***Parameters***

No parameters are needed to process this aggregation. Yet, a critical input is the set of pre-defined combination rules. They are provided in Table 3-23.

Table 3-23. Rules defined to combine single-year maps into a unique multi-year map

```
Combination rules are:
• for i = 1:n:
     /*If Urban label appears more than 4 years, final label is urban*/
     if ∑U >= 4:
          NB_LAB_MY(i) = 190
          Ambiguity(i) = ∑U
     /*If Sum(crops) >= Sum(Vegetation), final label is mosaic crop/vegetation*/
     elseif (∑C + 0.65*∑MC + 0.35*∑MV) >= (∑V + 0.65*∑MV + ∑MF + ∑MG + 0.15*∑S):
          NB_LAB_MY(i) = 30
          Ambiguity(i) = int(∑C + 0.65*∑MC + 0.35*∑MV)
     /*If Sum(crops) < Sum(Vegetation), final label is mosaic vegetation (MC, MF
```

```
          or MG)*/
          elseif (∑C + 0.65*∑MC + 0.35*∑MV) < (∑V + 0.65*∑MV + ∑MF + ∑MG + 0.15*∑S):
               /*If Sum(crops) is between 20 and 50%, final label is mosaic
               vegetation/crop*/
               a = (∑C + 0.65*∑MC + 0.35*∑MV)
               b = (∑V + 0.65*∑MV + ∑MF + ∑MG + 0.15*∑S)
               if ( 20%*(a+b) < a ) & ( 50%*(a+b) > a ):
                    NB_LAB_MY(i) = 40
                    Ambiguity(i) = int(∑C + 0.65*∑MC + 0.35*∑MV)
               /*If Sum(forest) is higher than Sum(grassland), final label is mosaic
               Forest-Shrub/Grassland */
               elseif (∑F + 0.65*∑MF + 0.35*∑MG + 0.15*∑S) > (∑G + 0.65*∑MG + 0.35*∑MF
               + 0.15*∑S):
                    NB_LAB_MY(i) = 100
                    Ambiguity(i) = int(∑F + 0.65*∑MF + 0.35*∑MG + 0.15*∑S)
               /*If Sum(forest) is lower than Sum(grassland), final label is mosaic
               Grassland/Forest-Shrub*/
               elseif (∑F + 0.65*∑MF + 0.35*∑MG + 0.15*∑S) < (∑G + 0.65*∑MG + 0.35*∑MF
               + 0.15*∑S):
                    NB_LAB_MY(i) = 120
                    Ambiguity(i) = int(∑G + 0.65*∑MG + 0.35*∑MF + 0.15*∑S)
               end
          end
end
```

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
algorithm SingleYear_To_MultiYear_Aggregation is
     input:
     CLASSIF1_SY<Y> (Y ranging from 1 to 10): raster file resulting from
     the spectral supervised maximum likelihood classification where each pixel
     is associated with a land cover class (described with an ID = NB_LAB and a
     name = LABEL). It results from the algorithm applied on single-year seasonal
     composites
     Code_CLASSIF1_SY<Y> (Y ranging from 1 to 10): raster provididng for
     each pixel the probability that the pixel is well classified. It results
     from the algorithm applied on single-year seasonal composites
     LUT 1

     output:
     CLASSIF1_MY: raster where where each pixel is associated with a
     land cover class (described with an ID = NB_LAB and a name = LABEL). It
     results from the combination of single-year maps.
     Code_CLASSIF1_MY : raster provididng for each pixel the probability that the
     pixel is well classified. It results from an aggregation of the probability
     obtained with the single-year seasonal composites using the years classified
     with the label selected in the multi-year map
     AMB_CODE_CLASSIF1_MY: raster providing for each pixel the
```

```
     reliability of the land cover map. It results from the combination of
     single-year maps.

for each pixel p, do:

     Read values (NB_LAB) in each of the input single-year maps: CLASSIF_1_SY<Y>

     Compute the histogram H of the corresponding single-year labels (NB_LAB)
     based first on the frequency and second on the chronology (most recent year
     first)

     Write the results in a text file “Histo” (which is a temporary output)

     Identify the majority label “LAB_Maj” and its frequency “LAB_Maj_freq”,
     which could vary from 1 to 10

     /*Identify the final label of the multi-year map, and associate and
     ambiguity code corresponding to the occurrence of the final label in the
     single-year maps*/

     if LAB_Maj_freq ≥ 6:
          NB_LAB_MY = LAB_Maj
          Ambiguity = LAB_Maj_freq
     else:
          1) Assuming the following grouping of the labels:
          * Crop labels = 10, 20
          * Vegetation labels = 50, 60, 70, 80, 90, 120, 120, 160, 170, 180
          * Forest labels = 50, 60, 70, 80, 90, 120, 160, 170
          * Grassland label = 130
          * Sparse labels = 140, 150
          * Urban label = 190
          * Mosaic crop/vegetation label = 30
          * Mosaic vegetation/crop label = 40
          * Mosaic Forest-Shrub/Grassland label = 100
          * Mosaic Grassland/Forest-Shrub label = 110
          2) Calculate intermediate variables that are the occurrence of the
          previous groups:
          * ΣC= sum(Crop labels)
          * ΣV = sum(Vegetation labels)
          * ΣF = sum(Forest labels)
          * ΣG = sum(Grassland labels)
          * ΣS = sum(Sparse labels)
          * ΣU = sum(Urban labels)
          * ΣMC = sum(Mosaic Crop/Vegetation labels)
          * ΣMV = sum(Mosaic Vegetation/Crop labels)
          * ΣMF = sum(Mosaic Forest-Shrub/Grassland labels)
          * ΣMG = sum(Mosaic Grassland/Forest-Shrub labels)
          3) Run the combination rules (see Table 3-23)
     end

     /*Compute the classification probabilities corresponding to the multi-year
     map*/

     If NB_LAB_MY is not a mosaic label (10, 20, 50 to 90, 120 to 220):
          * Identify the single-years that are classifed with the unique label
          derived from the combination rules and store the associated
          classification probabilities Code_Classif1_SY<Y>(NB_LAB_MY)
          * Store these values in a vector “Codes_Mean” (which is an temporary
          output)
     Elseif NB_LAB_MY is a mosaic label (30, 40, 100, 110):
          if NB_LAB_MY is 30:
               * Identify the single-years that are classifed either as 10, 20,
```

```
            30, 40 and store the associated classification probabilities
            Code_Classif1_SY<Y>(NB_LAB_MY)
            * Store these values in a vector “Codes_Mean” (which is an
            temporary output)
            * Calculate the mean of this vector:
            Code_MY = ∑(codes(lab(10,20))+ ∑(codes(lab30)*0.65)+∑(codes(lab40)*0.35) / Ambiguity
       elseif NB_LAB_MY is 40:
            * Identify the single-years that are classifed either as 10, 20,
            30, 40 and store the associated classification probabilities
            Code_Classif1_SY<Y>(NB_LAB_MY)
            * Store these values in a vector “Codes_Mean” (which is an
            temporary output)
            * Calculate the mean of this vector:
            Code_MY = ∑(codes(lab(10,20))+ ∑(codes(lab30)*0.35)+∑(codes(lab40)*0.65) / Ambiguity
       elseif NB_LAB_MY is 100:
            * Identify the single-years that are classifed either as 50 to
            100, 110, 120, 150 and store the associated classification
            probabilities
            Code_Classif1_SY<Y>(NB_LAB_MY)
            * Store these values in a vector “Codes_Mean” (which is an
            temporary output)
            * Calculate the mean of this vector: Code_MY =
            ∑(codes(lab(50 to 90,120,160,170))+ ∑(codes(lab100)*0.65)+∑(codes(lab110)*0.35)+∑(codes(lab150)*0.15 / Ambiguity
       elseif NB_LAB_MY is 110
            * Identify the single-years that are classifed either as 100, 110,
            130, 150 and store the associated classification probabilities
            Code_Classif1_SY<Y>(NB_LAB_MY)
            * Store these values in a vector “Codes_Mean” (which is an
            temporary output)
            * Calculate the mean of this vector:
            Code_MY = ∑(codes(lab 130))+ ∑(codes(lab100)*0.35)+∑(codes(lab110)*0.65)+∑(codes(lab150)*0.15) / Ambiguity
       end
    end

    Write the unique label, NB_LAB_MY, in the corresponding output raster file
    CLASSIF1_MY

    Write the corresponding classification probability, CODE_MY, in the
    corresponding output raster file Code_CLASSIF1_MY

    Write the ambiguity code, Ambiguity, reflecting the number of single-years
    classified with the selected label in the corresponding output raster file
    AMB_CODE_CLASSIF1_MY

end for p
```

Algorithm 3-8. Aggregation of single-year LC maps (derived from the spectral supervised algorithm) into a multi-year LC map

3.4 Detailed processing scheme of the unsupervised spectral classification (step 2)

Using an unsupervised classification algorithm allows reaching a rather high degree of automation while reducing the processing time. These advantages were successfully demonstrated in the GlobCover experience [RD.11, RD.20].

Unsupervised image classification is a classification process based solely on the image statistics, without availability of training data or other a priori knowledge of the area. The unsupervised algorithm used in this project relies on the clustering principle. Clustering is the task of assigning a set of pixels into clusters so that the pixels in the same cluster are more similar to each other than to those in other clusters. The assignation is based on natural groupings present in the reflectance values. The basic premise is that reflectance values within a given land cover class should be close together in the measurement space, whereas pixels belonging to different land cover classes should be comparatively well-separated [RD.12].

Clustering is not an algorithm in itself but a general task, which can be achieved by various algorithms that differ significantly in their notion of what constitutes a cluster and how to efficiently find them. The unsupervised classification algorithm used in this project relies on the ISODATA clustering technique, which represents each cluster by a single mean vector. More detail about this algorithm (and about all the classification procedure) is given hereafter in section 3.4.1.

Directly linked with the unsupervised algorithm is the labelling procedure, which aims at transforming the spectral clusters into LC classes. The unsupervised algorithm indeed generates spectrally separable clusters, for which the LC label is not known. The LC class associated with each cluster needs to be determined in a further step, by comparing the cluster to some auxiliary reference dataset.

The requirements for continuity and consistency in the long-term require an objective and automated labelling procedure. This challenge was already successfully addressed in the framework of the GlobCover project thanks to the use a global reference dataset and the definition of a set of generic decision rules. This procedure is detailed in the section 3.4.2.

3.4.1 Unsupervised ISODATA algorithm

This section describes the unsupervised classification algorithm used in the project, which relies on the ISODATA clustering technique.

The ISODATA algorithm is an iterative optimization clustering procedure, also called the migrating means technique. It is based upon estimating some reasonable assignment of the pixel vectors into candidate clusters and then moving them from one cluster to another in such a way that the Sum of Squared Error (SSE) is progressively reduced.

The algorithm is implemented by the following set of basic steps:

- First, the procedure starts by randomly selecting C points in the multidimensional input data space that will serve as candidate cluster centres:

$\mathbf{m}_i$, i = 1 … C;

- Second, each pixel in the image (or segment of image) to classify is assigned to the nearest candidate cluster. This assignment is based on the minimization of the Euclidean distance function between that pixel and the candidate cluster centers $\mathbf{m}_i$;
- Third, after each iteration, the new set of means that result from the grouping produced in step 2 are computed;
- Fourth, the entire process is repeated. After each iteration, a new mean is calculated for each cluster, based on the actual spectral locations of the pixels in the cluster. Then, these new means are used for defining clusters in the next iteration. The process continues until there is little change between iterations, i.e. if the normalised percentage of pixels whose assignments are unchanged since the last iteration reaches a convergence threshold or the maximum number of iterations is reached.

The ISODATA principle is illustrated in Figure 3-9.

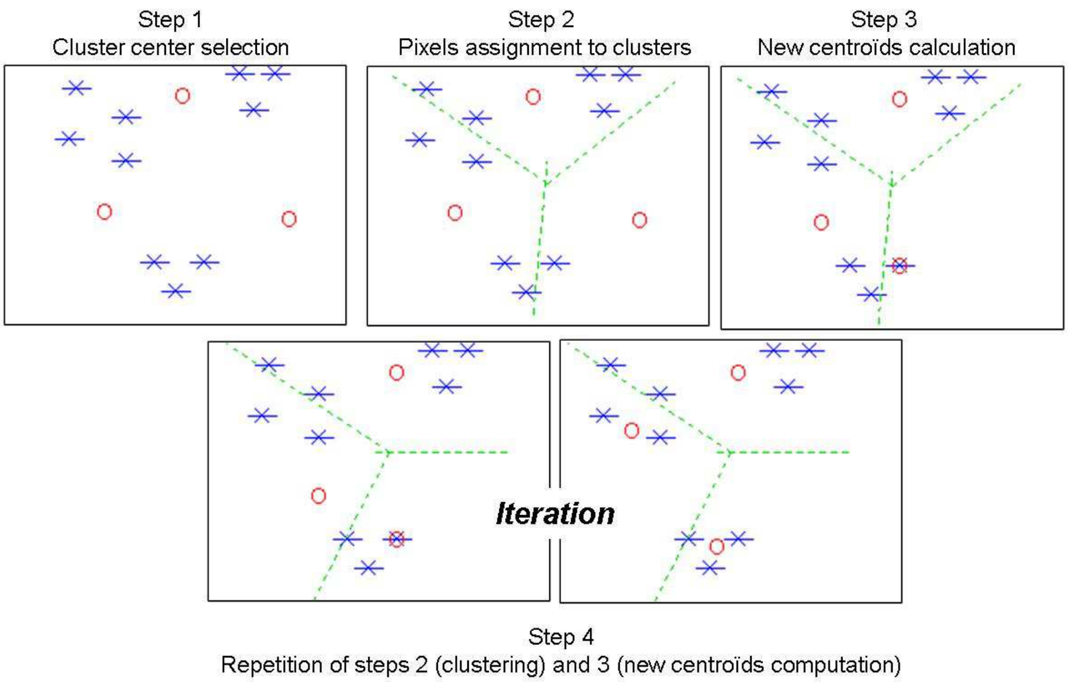

Figure 3-9: Principle of the ISODATA clustering technique

The ISODATA algorithm is applied to L3 seasonal composites independently for each equal-reasoning area (NB_ST). For each stratum, specific compositing periods and spectral channels were selected. These parameters are included in the LUT 7 (Table 3-24). As an output, the ISODATA algorithm creates, for each stratum (NB_ST), an output raster file (L4_<NB_ST>_Clusters_Spectral) where each pixel is associated to a spectrally homogeneous but unlabelled cluster (NB_Cluster).

The LUT 7 contains the following fields:

- **NB_ST**, which lists the numbers (IDs) ranging between 1 and 22 and corresponding to each equal-reasoning area;
- **Startdate_Season(i)**, specifying the exact date which marks the start of the seasonal composite i (i ranging from 1 to 3 according to the stratum);
- **Enddate_Season(i)**, specifying the exact date which marks the end of the seasonal composite i (i ranging from 1 to 3 according to the stratum);
- **Channels**, specifying the spectral channels to use as input for the machine learning classification algorithm;
- **N**, indicating the maximum number of clusters to generate;
- **NB_PIX**, indicating the minimum number of pixels in a cluster.
- ***Algorithm assumptions and limitations***

The ISODATA clustering technique relies on the assumption that each LC class can be well-represented by a single mean vector.

- ***Input and output data***

Input and output data associated with this unsupervised classification process are described in Table 3-24.

Table 3-24: Input and output data of the step 1b of the classification chain, i.e. the spectral unsupervised (ISODATA) classification algorithm

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
L3_StartDate_EndDate_SR_<n> n = 1,..,15 (MERIS)	Global raster of surface reflectance in each bands of the seasonal composites GeoTiff format	IN	None	[0 … 1]
LUT 2	Look-Up-Table describing the location of the 22 strata	IN	/	/
LUT 4	Look-Up-Table indicating which strata require the generation of multi-year seasonal composites	IN	/	/
LUT 7	Look-Up-Table describing the parameters of the unsupervised algorithm	IN	/	/
L4_<NB_ST>_Clusters_Spectral	Raster at the stratum level resulting from the unsupervised classification algorithm where each pixel is	OUT	None	[0 … 255]

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
	associated with a cluster ID (NB_Cluster)			

- ***Parameters***

Two parameters are associated with the ISODATA process, which are described in Table 3-25.

Table 3-25: Parameter of the ISODATA algorithm

PARAMETERS	DESCRIPTION	INTENT (IN, OUT, INOUT)	FORMAT	RANGE
NB_IT	Number of iterations to be performed	IN	Short	[0 ... 255]
T	Percentage of pixels remaining unchanged between iterations	IN	Short	[0... 100]

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
algorithm ISODATA_for_spectral_unsupervised_classification is

     input:
     MERIS single-year and multi-year seasonal composites used to extract
     reflectance values:
     L3_<sensor>_StartDate_EndDate_SR_<n>
     L3_<sensor>_MY_StartYear_EndYear_StartMonthDay_EndMonthDay_SR_<n>

     Stratification layer and LUT 2
     LUT 4
     LUT 7

     parameters:
     NB_IT = 1200
     T = 1.0

     output:
     L4_MY_<NB_ST>_Clusters: raster at the stratum level where each pixel is
     associated with an homogeneous spectral cluster. It results from the
     ISODATA classification algorithm applied on multi-year seasonal composites
     L4_SY<Y>_<NB_ST>_Clusters (Y ranging from 1 to 10): 10 raster files at
     the stratum level where each pixel is associated with an homogeneous
     spectral cluster. It results from the ISODATA classification algorithm
     applied on single-year seasonal composites

/*Organization of the ISODATA step, based on the multi-year strategy*/

for each stratum NB_ST (1 to 22 – LUT 2) do:

     Read the field "MY_Strategy" of the LUT 4

     if MY_Strategy = MY_S1 do:

          run ISODATA_S1 algorithm

     elseif MY_Strategy = MY_S2 do:
```

```
          run ISODATA_S2 algorithm
     end if
end for NB_ST


/*Run the ISODATA_S1 algorithm, for strata concerned by the MY_S1*/
for each stratum NB_ST do:
     Read the input data corresponding to NB_ST in the LUT 7:
     - STARTDATE_Si = exact date which marks the start of the seasonal
     composite i to use in the unsupervised classification, i being the
     number of composites specified for each stratum
     - ENDDATE_Si = exact date which marks the end of the seasonal
     composite i to use in the unsupervised classification, i being the
     number of composites specified for each stratum
     - CHij = j channels for the composite i
     Open the corresponding channels
     Read the algorithm parameters corresponding to NB_ST in the LUT 7:
     - N = the maximum number of clusters to generate
     - NB_IT = number of iterations to be performed
     - NB_PIX = minimum number of pixels in a cluster
     If the reflectance value of CHij(REFij) >< 0 do:
           Run the ISODATA function (see Algorithm 3-10) on the input data
           L3_<sensor>_StartDate_EndDate_SR_<n>
           Write results in the raster L4_MY_<NB_ST>_Clusters
     End if


/*Run the ISODATA_S2 algorithm for strata concerned by the MY_S2*/
for each stratum NB_ST do:

     Read the input data corresponding to NB_ST in the LUT 7:
     - STARTDATE_Si = exact date which marks the start of the seasonal
     composite i to use in the unsupervised classification, i being the
     number of composites specified for each stratum
     - ENDDATE_Si = exact date which marks the end of the seasonal
     composite i to use in the unsupervised classification, i being the
     number of composites specified for each stratum
     - CHij = j channels for the composite i
     Open the corresponding channels
     Read the algorithm parameters corresponding to NB_ST in the LUT 7:
     - N = the maximum number of clusters to generate
     - NB_IT = number of iterations to be performed
     - NB_PIX = minimum number of pixels in a cluster
     for each year Y, do:
          if the reflectance value of CHij(REFij) >< 0 do:
                Run the ISODATA function (see Algorithm 3-10) on the input data
                L3_<sensor>_StartDate_EndDate_SR_<n>
                Write results in the raster L4_SY<Y>_<NB_ST>_Clusters
          end if
```

```
    end for Y
End for NB_ST
```

Algorithm 3-9. Unsupervised classification algorithm

```
algorithm ISODATA_function is

    input EO data: suitable seasonal composites and spectral ca
    seasonal composites (start and end dates provided in dedicated LUT)
    spectral channels (channels selection provided in dedicated LUT)

    input parameters (stored in a dedicated LUT):
    N = the maximum number of clusters to generate
    NB_IT = number of iterations to be performed
    NB_PIX = minimum number of pixels in a cluster
    T = percentage of pixels remaining unchanged between iterations

    output:
    raster associating each pixel with a spectral cluster

Read input data (seasonal composites and spectral channels to classify)

Read algorithms parameters values in dedicated LUTs

/* Determines the number of arbitrary clusters (C) corresponding to 130% of the
number of classes required as output (N)*/

C = 1.3 * N

/*Starts the iterative clustering procedure*/

for  NB_IT = 1 to “NB_IT” do:

    while (C > N) do:

        All the pixels of the image are assigned to the nearest candidate
        cluster and a new set of means is computed

        Remove all the classes with a number of pixels lower than the
        specified parameter “NB_PIX”

        if (T<1%) do:

            The class with the minimum number of pixels is removed

        end if

    end while

    while (C = N) and (T<1%) do:

        The class with the minimum number of pixels is removed

        All the pixels of the image are assigned to the nearest candidate
        cluster and a new set of means is computed

    end while

end for NB_IT
```

Algorithm 3-10. ISODATA function

3.4.2 Automatic reference-based labelling

The ISODATA algorithm has interpreted, for each stratum, L3 seasonal composites into a set of clusters which have the property to be spectrally homogeneous but which are not identified by a land cover label. Transforming these clusters into LC classes (identified by a number NB_LAB and a name LABEL – see LUT 1 in Table 3-10) is the objective of this labelling step. This is done through a comparison between the cluster raster file (L4_<NB_ST>_CLUSTERS_Spectral) and the auxiliary LC reference database.

The auxiliary LC reference database is the one built as a preliminary step (see section 3.2.1) and is the same than the one used to define the training sample in the supervised classification approach (see section 3.3.1). As already mentioned, this reference database (REF_LC) consists of a set of global, regional and local reference land cover maps selected as the most accurate ones available for a given region, with the highest spatial resolution and with a CCI-compatible legend. It is associated with the CCI LC legend, i.e. with the NB_LAB, LABEL and color codes contained in the LUT 1 (Table 3-10).

A superposition of the cluster raster (L4_<NB_ST>_CLUSTERS) with the reference layer (REF_LC) is operated. For each cluster, a histogram of class frequency is computed (Figure 3-10). The most represented classes inside the cluster are identified and ranked using (i) the number of pixels they cover (NB_Pix1, NB_Pix2, etc.) and (ii) their label (NB_LAB1, NB_LAB2, etc.).

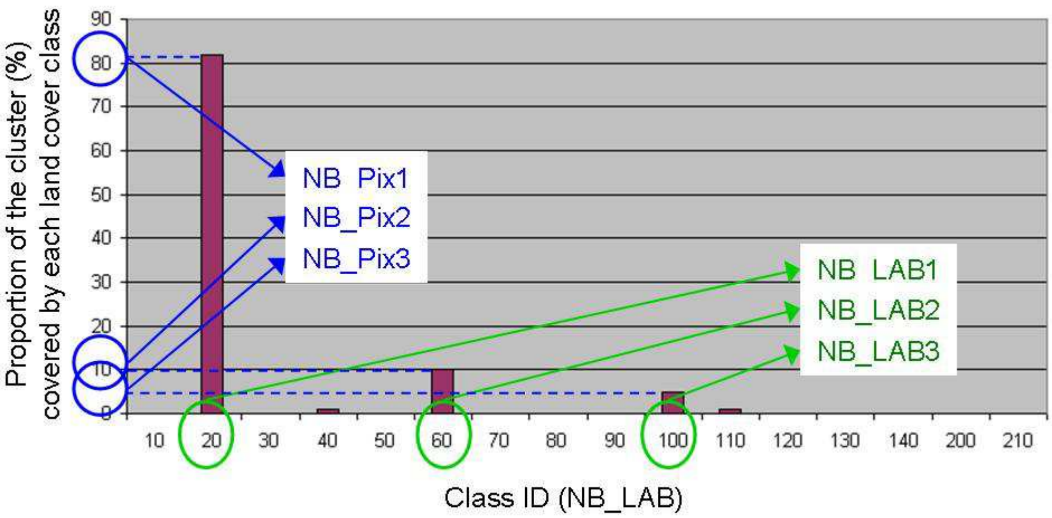

Figure 3-10: Histogram of class frequency interpretation: most represented classes are identified

The class frequency histogram is then interpreted according to a set decision rules which are defined a priori. As a result, a unique LC class (i.e. a unique class number NB_LAB and name LABEL) is associated with each cluster (NB_Cluster) and an output raster file (CLASSIF_2) is generated. The histogram interpretation process also associates each cluster with an ambiguity code (CODE) that characterizes the ambiguity of the interpretation, and thus the reliability of the associated LC label.

- ***Algorithm assumptions and limitations***

None

- ***Input and output data***

Input and output data associated with the automated labelling procedure are described in Table 3-26.

Table 3-26: Input and output data of the step 2 of the classification chain, which is the spectral unsupervised classification process, for the automated labelling procedure

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
L4_<NB_ST>_Clusters_Spectral	Raster at the stratum level resulting from the unsupervised classification algorithm where each pixel is associated with a cluster ID (NB_Cluster)	IN	None	[0 … 1024]
REF_LC	Reference land cover layer where each pixel is associated with a land cover class through an ID	IN	None	[0 … 255]
LUT 1	Look-Up-Table describing the CCI LCCS land cover legend	IN	/	/
L4_<NB_ST>_Histo	Text file (one for each equal-reasoning area) containing for each cluster (NB_Cluster) the land cover classes frequency (with indices NB_Pixi and NB_LABi)	INOUT	Long	[0 … 100]
CLASSIF 2	Land cover map resulting from the unsupervised classification approach (ISODATA algorithm + labelling process), where each pixel is associated with a land cover class through an ID	OUT	None	[0 … 255]
Code_Classif2	Ambiguity code that characterizes the ambiguity of the classification process, thus reflecting, at the pixel level, the reliability of the CLASSIF 1B	OUT	None	0 … 10

- ***Parameters***

No parameters are needed to process this aggregation. Yet, a critical input is the set of pre-defined labelling rules that are used to interpret the histograms. They are provided in Table 3-27.

Table 3-27. Decision rules defined to label clusters resulting from the ISODATA unsupervised algorithm in the spectral classification step

```
Decision rules are:

• if (NB_Pix1_GL > 85%), do:
     CODE = 1
     * if (NB_Pix1 > 60%), do:
          NB_LAB(CLASSIF_1B) = NB_LAB1
     * else, do:
          NB_LAB(CLASSIF_1B) = NB_LAB1_GL
• if (70% < NB_Pix1_GL ≤ 80%) do:
     CODE = 2
     * if (NB_LAB1_GL = 210) and (159 < NB_LAB2_GL < 189), do:
          NB_LAB(CLASSIF_1B) = NB_LAB2_GL
     * if (NB_LAB1_GL = 200) and (NB_LAB2_GL = 190) and (NB_Pix1_GL > 10%), do:
          NB_LAB(CLASSIF_1B) = NB_LAB2_GL
     * if (NB_Pix1 > 60%), do:
          NB_LAB(CLASSIF_1B) = NB_LAB1
     * else, do:
          NB_LAB(CLASSIF_1B) = NB_LAB1_GL
• if (60% < NB_Pix1_GL ≤ 70%) do:
     CODE = 3
     * if (NB_LAB1_GL = 210), do:
          NB_LAB(CLASSIF_1B) = NB_LAB2_GL
     * if (NB_LAB1_GL = 200) and (NB_LAB2_GL = 190) and (NB_Pix1_GL > 10%), do:
          NB_LAB(CLASSIF_1B) = NB_LAB2_GL
     * if [(49 < NB_LAB1_GL < 99) or (NB_LAB1_GL = 120)] and [(NB_LAB2_GL = 130)
     or (99 < NB_LAB2_GL < 119)], do:
          NB_LAB(CLASSIF_1B) = 100
     * if (NB_LAB1_GL = 130) and (49 < NB_LAB2_GL < 129), do:
          NB_LAB(CLASSIF_1B) = 110
     * if (9 < NB_LAB1_GL < 29) and [(49 < NB_LAB2_GL < 99) or (119 < NB_LAB2_GL
     < 139)], do:
          NB_LAB(CLASSIF_1B) = 30
     * if [(49 < NB_LAB1_GL < 99) or (119 < NB_LAB1_GL < 139)] and (9 <
     NB_LAB2_GL < 29), do:
          NB_LAB(CLASSIF_1B) = 40
     * else, do:
          ▪if (NB_Pix1 > 60%), do: NB_LAB(CLASSIF_1B) = NB_LAB1
```

```
        ▪else, do: NB_LAB(CLASSIF_1B) = NB_LAB1_GL
• if (40% < NB_Pix1_GL ≤ 60%) and (NB_Pix2_GL > 20%) do:
    CODE = 4
    * if (NB_LAB1_GL = 210), do:
        NB_LAB(CLASSIF_1B) = NB_LAB2_GL
    * if (NB_LAB1_GL = 200) and (NB_LAB2_GL = 190) and (NB_Pix1_GL > 10%), do:
        NB_LAB(CLASSIF_1B) = NB_LAB2_GL
    * if [(49 < NB_LAB1_GL < 99) or (NB_LAB1_GL = 120)] and [(NB_LAB2_GL = 130)
    or (99 < NB_LAB2_GL < 119)], do:
        NB_LAB(CLASSIF_1B) = 100
    * if (NB_LAB1_GL = 130) and (49 < NB_LAB2_GL < 129), do:
        NB_LAB(CLASSIF_1B) = 110
    * if (9 < NB_LAB1_GL < 29) and [(49 < NB_LAB2_GL < 99) or (119 < NB_LAB2_GL
    < 139)], do:
        NB_LAB(CLASSIF_1B) = 30
    * if [(49 < NB_LAB1_GL < 99) or (119 < NB_LAB1_GL < 139)] and (9 <
    NB_LAB2_GL < 29), do:
        NB_LAB(CLASSIF_1B) = 40
    * else, do:
        ▪if (NB_Pix1 > 60%), do: NB_LAB(CLASSIF_1B) = NB_LAB1
        ▪else, do: NB_LAB(CLASSIF_1B) = NB_LAB1_GL
• if (40% < NB_Pix1_GL ≤ 60%) and (NB_Pix2_GL ≤ 20%) and (NB_Pix1_GL+NB_Pix2_GL >
  50%), do:
    CODE = 5
    * if (NB_LAB1_GL = 210), do:
        NB_LAB(CLASSIF_1B) = NB_LAB2_GL
    * if (NB_LAB1_GL = 200) and (NB_LAB2_GL = 190) and (NB_Pix1_GL > 10%), do:
        NB_LAB(CLASSIF_1B) = NB_LAB2_GL
    * if [(49 < NB_LAB1_GL < 99) or (NB_LAB1_GL = 120)] and [(NB_LAB2_GL = 130)
    or (99 < NB_LAB2_GL < 119)], do:
        NB_LAB(CLASSIF_1B) = 100
    * if (NB_LAB1_GL = 130) and (49 < NB_LAB2_GL < 129), do:
        NB_LAB(CLASSIF_1B) = 110
    * if (9 < NB_LAB1_GL < 29) and [(49 < NB_LAB2_GL < 99) or (119 < NB_LAB2_GL
    < 139)], do:
        NB_LAB(CLASSIF_1B) = 30
    * if [(49 < NB_LAB1_GL < 99) or (119 < NB_LAB1_GL < 139)] and (9 <
    NB_LAB2_GL < 29), do:
        NB_LAB(CLASSIF_1B) = 40
    * else, do:
        ▪if (NB_Pix1 > 60%), do: NB_LAB(CLASSIF_1B) = NB_LAB1
```

```
            ▪else, do: NB_LAB(CLASSIF_1B) = NB_LAB1_GL
• if (40% < NB_Pix1_GL ≤ 60%) and (NB_Pix2_GL ≤ 20%) and (NB_Pix1_GL+NB_Pix2_GL ≤
  50%), do:
      CODE = 6
      * if (NB_LAB1_GL = 210), do:
            NB_LAB(CLASSIF_1B) = NB_LAB2_GL
      * if (NB_LAB1_GL = 200) and (NB_LAB2_GL = 190) and (NB_Pix1_GL > 10%), do:
            NB_LAB(CLASSIF_1B) = NB_LAB2_GL
      * if [(49 < NB_LAB1_GL < 99) or (NB_LAB1_GL = 120)] and (S_100110120 > 20%),
      do:
            ▪ if (NB_Pix1_GL > S_100110120), do: NB_LAB(CLASSIF_1B) = 100
            ▪ if (NB_Pix1_GL ≤ S_100110120), do: NB_LAB(CLASSIF_1B) = 110
      * if (NB_LAB1_GL = 130) and (S_50to120 > 20%), do:
            ▪ if (NB_Pix1_GL > S_50to120), do: NB_LAB(CLASSIF_1B) = 110
            ▪ if (NB_Pix1_GL ≤ S_50to120), do: NB_LAB(CLASSIF_1B) = 100
      * if (9 < NB_LAB1_GL < 29) and (S_50to90_120130 > 20%), do:
            ▪ if (NB_Pix1_GL > S_50to90_120130), do: NB_LAB(CLASSIF_1B) = 30
            ▪ if (NB_Pix1_GL ≤ S_50to90_120130), do: NB_LAB(CLASSIF_1B) = 40
      * if [(49 < NB_LAB1_GL < 99) or (119 < NB_LAB1_GL < 139)]  and (S_1020 >
      20%), do:
            ▪ if (NB_Pix1_GL > S_1020), do: NB_LAB(CLASSIF_1B) = 40
            ▪ if (NB_Pix1_GL ≤ S_1020), do: NB_LAB(CLASSIF_1B) = 30
      * if (9 < NB_LAB1_GL < 29) and (S_3040 > NB_Pix1_GL), do:
            NB_LAB(CLASSIF_1B) = 40
      * if [(49 < NB_LAB1_GL < 99) or (NB_LAB1_GL = 120)] and (S_100110 >
      NB_Pix1_GL), do:
            NB_LAB(CLASSIF_1B) = 100
      * if (NB_LAB1_GL = 130) and (S_100110 > NB_Pix1_GL), do:
            NB_LAB(CLASSIF_1B) = 110
      * else, do:
            ▪if (NB_Pix1 > 60%), do: NB_LAB(CLASSIF_1B) = NB_LAB1
            ▪else, do: NB_LAB(CLASSIF_1B) = NB_LAB1_GL
• if (NB_Pix1_GL ≤ 40) and (NB_Pix2_GL > 20%) and (NB_Pix1_GL+NB_Pix2_GL > 50%),
  do:
      CODE = 7
      * if (NB_LAB1_GL = 210), do:
            NB_LAB(CLASSIF_1B) = NB_LAB2_GL
      * if (NB_LAB1_GL = 200) and (NB_LAB2_GL = 190) and (NB_Pix1_GL > 10%), do:
            NB_LAB(CLASSIF_1B) = NB_LAB2_GL
      * if [(49 < NB_LAB1_GL < 99) or (NB_LAB1_GL = 120)] and [(NB_LAB2_GL = 130)
```

```
    or (99 < NB_LAB2_GL < 119)], do:
          NB_LAB(CLASSIF_1B) = 100
    * if (NB_LAB1_GL = 130) and (49 < NB_LAB2_GL < 129), do:
          NB_LAB(CLASSIF_1B) = 110
    * if (9 < NB_LAB1_GL < 29) and [(49 < NB_LAB2_GL < 99) or (119 < NB_LAB2_GL
    < 139)], do:
          NB_LAB(CLASSIF_1B) = 30
    * if [(49 < NB_LAB1_GL < 99) or (119 < NB_LAB1_GL < 139)] and (9 <
    NB_LAB2_GL < 29), do:
          NB_LAB(CLASSIF_1B) = 40
    * else, do:
          ▪if (NB_Pix1 > 60%), do: NB_LAB(CLASSIF_1B) = NB_LAB1
          ▪else, do: NB_LAB(CLASSIF_1B) = NB_LAB1_GL
• if (NB_Pix1_GL ≤ 40) and (NB_Pix2_GL > 20%) and (NB_Pix1_GL+NB_Pix2_GL ≤ 50%),
  do:
    CODE = 8
    * if (NB_LAB1_GL = 210), do:
          NB_LAB(CLASSIF_1B) = NB_LAB2_GL
    * if (NB_LAB1_GL = 200) and (NB_LAB2_GL = 190) and (NB_Pix1_GL > 10%), do:
          NB_LAB(CLASSIF_1B) = NB_LAB2_GL
    * if [(49 < NB_LAB1_GL < 99) or (NB_LAB1_GL = 120)] and (S_100110120 > 20%),
    do:
          ▪ if (NB_Pix1_GL > S_100110120), do: NB_LAB(CLASSIF_1B) = 100
          ▪ if (NB_Pix1_GL ≤ S_100110120), do: NB_LAB(CLASSIF_1B) = 110
    * if (NB_LAB1_GL = 130) and (S_50to120 > 20%), do:
          ▪ if (NB_Pix1_GL > S_50to120), do: NB_LAB(CLASSIF_1B) = 110
          ▪ if (NB_Pix1_GL ≤ S_50to120), do: NB_LAB(CLASSIF_1B) = 100
    * if (9 < NB_LAB1_GL < 29) and (S_50to90_120130 > 20%), do:
          ▪ if (NB_Pix1_GL > S_50to90_120130), do: NB_LAB(CLASSIF_1B) = 30
          ▪ if (NB_Pix1_GL ≤ S_50to90_120130), do: NB_LAB(CLASSIF_1B) = 40
    * if [(49 < NB_LAB1_GL < 99) or (119 < NB_LAB1_GL < 139)]  and (S_1020 >
    20%), do:
          ▪ if (NB_Pix1_GL > S_1020), do: NB_LAB(CLASSIF_1B) = 40
          ▪ if (NB_Pix1_GL ≤ S_1020), do: NB_LAB(CLASSIF_1B) = 30
    * if (9 < NB_LAB1_GL < 29) and (S_3040 > NB_Pix1_GL), do:
          NB_LAB(CLASSIF_1B) = 40
    * if [(49 < NB_LAB1_GL < 99) or (NB_LAB1_GL = 120)] and (S_100110 >
    NB_Pix1_GL), do:
          NB_LAB(CLASSIF_1B) = 100
    * if (NB_LAB1_GL = 130) and (S_100110 > NB_Pix1_GL), do:
```

```
        NB_LAB(CLASSIF_1B) = 110
    * else, do:
        ▪if (NB_Pix1 > 60%), do: NB_LAB(CLASSIF_1B) = NB_LAB1
        ▪else, do: NB_LAB(CLASSIF_1B) = NB_LAB1_GL
• if (NB_Pix1_GL ≤ 40) and (NB_Pix2_GL ≤ 20%) and (NB_Pix1_GL+NB_Pix2_GL > 50%),
  do:
    CODE = 9
    * if (NB_LAB1_GL = 210), do:
        NB_LAB(CLASSIF_1B) = NB_LAB2_GL
    * if (NB_LAB1_GL = 200) and (NB_LAB2_GL = 190) and (NB_Pix1_GL > 10%), do:
        NB_LAB(CLASSIF_1B) = NB_LAB2_GL
    * if [(49 < NB_LAB1_GL < 99) or (NB_LAB1_GL = 120)] and [(NB_LAB2_GL = 130)
    or (99 < NB_LAB2_GL < 119)], do:
        NB_LAB(CLASSIF_1B) = 100
    * if (NB_LAB1_GL = 130) and (49 < NB_LAB2_GL < 129), do:
        NB_LAB(CLASSIF_1B) = 110
    * if (9 < NB_LAB1_GL < 29) and [(49 < NB_LAB2_GL < 99) or (119 < NB_LAB2_GL
    < 139)], do:
        NB_LAB(CLASSIF_1B) = 30
    * if [(49 < NB_LAB1_GL < 99) or (119 < NB_LAB1_GL < 139)] and (9 <
    NB_LAB2_GL < 29), do:
        NB_LAB(CLASSIF_1B) = 40
    * else, do:
        ▪if (NB_Pix1 > 60%), do: NB_LAB(CLASSIF_1B) = NB_LAB1
        ▪else, do: NB_LAB(CLASSIF_1B) = NB_LAB1_GL
• if (NB_Pix1_GL ≤ 40) and (NB_Pix2_GL ≤ 20%) and (NB_Pix1_GL+NB_Pix2_GL ≤ 50%),
  do:
    CODE = 10
    * if (NB_LAB1_GL = 210), do:
        NB_LAB(CLASSIF_1B) = NB_LAB2_GL
    * if (NB_LAB1_GL = 200) and (NB_LAB2_GL = 190) and (NB_Pix1_GL > 10%), do:
        NB_LAB(CLASSIF_1B) = NB_LAB2_GL
    * if [(49 < NB_LAB1_GL < 99) or (NB_LAB1_GL = 120)] and (S_100110120 > 20%),
    do:
        ▪ if (NB_Pix1_GL > S_100110120), do: NB_LAB(CLASSIF_1B) = 100
        ▪ if (NB_Pix1_GL ≤ S_100110120), do: NB_LAB(CLASSIF_1B) = 110
    * if (NB_LAB1_GL = 130) and (S_50to120 > 20%), do:
        ▪ if (NB_Pix1_GL > S_50to120), do: NB_LAB(CLASSIF_1B) = 110
        ▪ if (NB_Pix1_GL ≤ S_50to120), do: NB_LAB(CLASSIF_1B) = 100
    * if (9 < NB_LAB1_GL < 29) and (S_50to90_120130 > 20%), do:
```

```
        ▪ if (NB_Pix1_GL > S_50to90_120130), do: NB_LAB(CLASSIF_1B) = 30

        ▪ if (NB_Pix1_GL ≤ S_50to90_120130), do: NB_LAB(CLASSIF_1B) = 40

   * if [(49 < NB_LAB1_GL < 99) or (119 < NB_LAB1_GL < 139)]  and (S_1020 >
   20%), do:

        ▪ if (NB_Pix1_GL > S_1020), do: NB_LAB(CLASSIF_1B) = 40

        ▪ if (NB_Pix1_GL ≤ S_1020), do: NB_LAB(CLASSIF_1B) = 30

   * if (9 < NB_LAB1_GL < 29) and (S_3040 > NB_Pix1_GL), do:

        NB_LAB(CLASSIF_1B) = 40

   * if [(49 < NB_LAB1_GL < 99) or (NB_LAB1_GL = 120)] and (S_100110 >
   NB_Pix1_GL), do:

        NB_LAB(CLASSIF_1B) = 100

   * if (NB_LAB1_GL = 130) and (S_100110 > NB_Pix1_GL), do:
        NB_LAB(CLASSIF_1B) = 110

   * else, do:

        ▪if (NB_Pix1 > 60%), do: NB_LAB(CLASSIF_1B) = NB_LAB1

        ▪else, do: NB_LAB(CLASSIF_1B) = NB_LAB1_GL
```

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
algorithm Labelling_Spectral_Classification is

     input:
     L4_MY_<NB_ST>_Clusters: raster at the stratum level where each pixel is
     associated with an homogeneous spectral cluster (NB_Cluster). It results
     from the ISODATA classification algorithm applied on multi-year seasonal
     composites
     L4_SY<Y>_<NB_ST>_Clusters (Y ranging from 1 to 10): raster files at
     the stratum level where each pixel is associated with an homogeneous
     spectral cluster (NB_Cluster). It results from the ISODATA classification
     algorithm applied on single-year seasonal composites
     REF_LC: reference land cover layer
     Stratification layer and LUT 2
     LUT 1

     output:
     CLASSIF2_MY: raster file resulting from the labelling procedure where
     each pixel is associated with a land cover class (described with an ID =
     NB_LAB and a name = LABEL). It results from the unsupervised classification
     step (ISODATA + labelling) applied on multi-year seasonal composites
     Code_Classif2_MY: raster providing for each pixel the ambiguity level
     of the whole unsupervised classification approach. It results from the
     unsupervised classification step (ISODATA + labelling) applied on multi-year
     seasonal composites
     CLASSIF2_SY<Y> (Y ranging from 1 to 10): raster file resulting from
     the labelling procedure where each pixel is associated with a land cover
     class (described with an ID = NB_LAB and a name = LABEL). It results from
     the unsupervised classification step (ISODATA + labelling) applied on
     single-year seasonal composites
```

```
    Code_CLASSIF2_SY<Y> (Y ranging from 1 to 10): raster providing for
    each pixel the ambiguity level of the whole unsupervised classification
    approach. It results from the unsupervised classification step (ISODATA +
    labelling) applied on single-year seasonal composites

for each stratum NB_ST (1 to 22 – LUT 2) do:

    Read input data:
    L4_MY_<NB_ST>_Clusters
    L4_SY<Y>_<NB_ST>_Clusters

    for each cluster raster, do:

        for each spectral cluster NB_Cluster do:

            Read all pixels belonging to this NB_Cluster in the input data
            L4_<MY or SY>_<NB_ST>_Clusters

            For these pixels, read values (NB_LAB) in the reference land
            cover layer REF_LC and the corresponding label (LABEL) in LUT 1

            Compute the frequency histogram H of corresponding reference
            labels (NB_LAB)

            Write the results in the text file L4_<NB_ST>_Histo (which is an
            intermrediate output)

            Identify the 2 most represented classes:
            - NB_Pix1 = H(1) (with NB_LAB1 as corresponding label)
            - NB_Pix2 = H(2) (with NB_LAB2 as corresponding label)

            Compute the sum of NB_Pixi for the labels (NB_LABi) belonging to
            the same tens (10 to 19, 20 to 29, 30 to 39,…), standing for
            “global classes)

            Compute the frequency histogram H of reference labels
            corresponding to these global classes, identify the 2 most
            represented classes amongst these new global classes and compute
            varying sums amongst these new global classes:
            - H_GL = frequency histogram of corresponding global labels
            - NB_PIX1_GL= H_GL(1) (with NB_LAB1_GL as corresponding label)
            - NB_PIX2_GL= H_GL(2) (with NB_LAB2_GL as corresponding label)
            - S_1020 = sum(NB_Pix_GL(NB_LAB_GL=10,20)
            - S_3040 = sum(NB_Pix_GL(NB_LAB_GL=30,40)
            - S_50to90 = sum(NB_Pix_GL(NB_LAB_GL=50,60,70,80,90)
            - S_50to90_120130 = sum(NB_Pix_GL(NB_LAB_GL=50,60,70,80,
            90,120,130)
            - S_50to120 = sum(NB_Pix_GL(NB_LAB_GL=50,60,70,80,90,100, 110,
            120)
            - S_100110 = sum(NB_Pix_GL(NB_LAB_GL=100,110)
            - S_100110120 = sum(NB_Pix_GL(NB_LAB_GL=100,110,120)
            Run the labelling rules (see Table 3-27)

            Write the unique label (NB_LAB) derived from the decision rules
            in the corresponding output raster file CLASSIF2_<MY or SY>
            Write the ambiguity code (CODE) derived from the decision rules
            in the corresponding output raster file Code_CLASSIF2_<MY or
            SY>
        end for NB_Cluster

    end for each cluster raster

end for NB_ST
```

Algorithm 3-11. Automated labelling procedure, as applied in the spectral unsupervised classification step

3.4.3 Multi-temporal approach

For strata concerned by the MY_S2 (see LUT 4 in Table 3-12, section 3.2.3.1), the classification algorithm has to be run multiple times (one for each year of interest) and the multiple single-year LC maps have to be aggregated in a multi-year land cover map. This process is identical to the one applied for the machine learning algorithm (see section 3.3.4).

- ***Algorithm assumptions and limitations***

None

- ***Input and output data***

Input and output data associated with the aggregation of single-year land cover maps are described in Table 3-28.

Table 3-28: Input and output data of the aggregation of single-year land cover maps (multi-year approach)

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
CLASSIF2_SY_<Year>	Raster at the stratum level resulting from the unsupervised spectral classification algorithm run on single-year seasonal composites, where each pixel is associated with a land cover class ID (NB_LAB)	IN	None	[0 … 255]
Code_CLASSIF2_SY_<Year>	Ambiguity code associated with the label selected for each pixel	OUT	None	0 … 10
LUT 1	Look-Up-Table describing the CCI LCCS land cover legend	IN	/	/
CLASSIF2_Histo	Text file (one for each equal-reasoning area) containing for each pixel the land cover classes frequency	INOUT	Long	[0 … 100]
CLASSIF2_MY	Land cover map resulting from the aggregation of single-year spectral land cover maps, where each pixel is associated with a land cover class through an ID	OUT	None	[0 … 255]
Code_CLASSIF2_MY	Classification probability associated with the label selected in the land cover map resulting from the aggregation of single-year spectral land cover maps	OUT	None	0 … 1

AMB_CODE_CLASSIF2_MY	Frequency of a same land cover class observed over the multiple aggregated years, thus reflecting, at the pixel level, the reliability of the CLASSIF2_MY	OUT	None	0 … 10

- ***Parameters***

No parameters are needed to process this aggregation. Yet, a critical input is the set of pre-defined combination rules. They are the same than for the spectral supervised algorithm (Table 3-23).

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
algorithm SingleYear_To_MultiYear_Aggregation is

     input:
     CLASSIF2_SY<Y> (Y ranging from 1 to 10): raster file resulting from
     the spectral unsupervised classification where each pixel is associated with
     a land cover class (described with an ID = NB_LAB and a name = LABEL). It
     results from the algorithm applied on single-year seasonal composites
     Code_CLASSIF2_SY<Y> (Y ranging from 1 to 10): raster providing for
     each pixel the ambiguity of the classification. It results from the
     algorithm applied on single-year seasonal composites
     LUT 1

     output:
     CLASSIF2_MY: raster where where each pixel is associated with a
     land cover class (described with an ID = NB_LAB and a name = LABEL). It
     results from the combination of single-year maps.
     Code_CLASSIF2_MY : raster providing for each pixel the ambiguity of the
     classification algorithm. It results from an aggregation of the ambiguity
     code obtained with the single-year seasonal composites using the years
     classified with the label selected in the multi-year map
     AMB_CODE_CLASSIF2_MY: raster providing for each pixel the
     reliability of the land cover map. It results from the combination of
     single-year maps.

for each pixel p, do:

     Read values (NB_LAB) in each of the input single-year maps: CLASSIF2_SY<Y>

     Compute the histogram H of the corresponding single-year labels (NB_LAB)
     based first on the frequency and second on the chronology (most recent year
     first)

     Write the results in a text file “Histo” (which is a temporary output)

     Identify the majority label “LAB_Maj” and its frequency “LAB_Maj_freq”,
     which could vary from 1 to 10

     /*Identify the final label of the multi-year map, and associate and
     ambiguity code corresponding to the occurrence of the final label in the
     single-year maps*/

     if LAB_Maj_freq ≥ 6:
          NB_LAB_MY = LAB_Maj
          Ambiguity = LAB_Maj_freq
```

```
else:
     1) Assuming the following grouping of the labels:
     * Crop labels = 10, 20
     * Vegetation labels = 50, 60, 70, 80, 90, 120, 120, 160, 170, 180
     * Forest labels = 50, 60, 70, 80, 90, 120, 160, 170
     * Grassland label = 130
     * Sparse labels = 140, 150
     * Urban label = 190
     * Mosaic crop/vegetation label = 30
     * Mosaic vegetation/crop label = 40
     * Mosaic Forest-Shrub/Grassland label = 100
     * Mosaic Grassland/Forest-Shrub label = 110
     2) Calculate intermediate variables that are the occurrence of the
     previous groups:
     * ΣC= sum(Crop labels)
     * ΣV = sum(Vegetation labels)
     * ΣF = sum(Forest labels)
     * ΣG = sum(Grassland labels)
     * ΣS = sum(Sparse labels)
     * ΣU = sum(Urban labels)
     * ΣMC = sum(Mosaic Crop/Vegetation labels)
     * ΣMV = sum(Mosaic Vegetation/Crop labels)
     * ΣMF = sum(Mosaic Forest-Shrub/Grassland labels)
     * ΣMG = sum(Mosaic Grassland/Forest-Shrub labels)
     3) Run the combination rules (see Table 3-23)
end

/*Compute the classification probabilities corresponding to the multi-year
map*/

If NB_LAB_MY is not a mosaic label (10, 20, 50 to 90, 120 to 220):
     * Identify the single-years that are classifed with the unique label
     derived from the combination rules and store the associated
     classification probabilities Code_CLASSIF2_SY<Y>(NB_LAB_MY)
     * Store these values in a vector “Codes_Mean” (which is an temporary
     output)
Elseif NB_LAB_MY is a mosaic label (30, 40, 100, 110):
     if NB_LAB_MY is 30:
          * Identify the single-years that are classifed either as 10, 20,
          30, 40 and store the associated classification probabilities
          Code_CLASSIF2_SY<Y>(NB_LAB_MY)
          * Store these values in a vector “Codes_Mean” (which is an
          temporary output)
          * Calculate the mean of this vector:
```

$$\text{Code\_MY} = \frac{\sum(codes(lab(10,20))) + \sum(codes(lab30)*0.65) + \sum(codes(lab40)*0.35)}{Ambiguity}$$

```
     elseif NB_LAB_MY is 40:
          * Identify the single-years that are classifed either as 10, 20,
          30, 40 and store the associated classification probabilities
          Code_CLASSIF2_SY<Y>(NB_LAB_MY)
          * Store these values in a vector “Codes_Mean” (which is an
          temporary output)
          * Calculate the mean of this vector:
```

$$\text{Code\_MY} = \frac{\sum(codes(lab(10,20))) + \sum(codes(lab30)*0.35) + \sum(codes(lab40)*0.65)}{Ambiguity}$$

```
     elseif NB_LAB_MY is 100:
          * Identify the single-years that are classifed either as 50 to
          100, 110, 120, 150 and store the associated classification
          probabilities
          Code_CLASSIF2_SY<Y>(NB_LAB_MY)
          * Store these values in a vector “Codes_Mean” (which is an
```

```
            temporary output)
            * Calculate the mean of this vector: Code_MY =
            ∑(codes(lab(50 to 90,120,160,170))+∑(codes(lab100)*0.65)+∑(codes(lab110)*0.35)+∑(codes(lab150)*0.15
            ------------------------------------------------------------------------------------------------
                                              Ambiguity
        elseif NB_LAB_MY is 110
            * Identify the single-years that are classifed either as 100, 110,
            130, 150 and store the associated classification probabilities
            Code_CLASSIF2_SY<Y>(NB_LAB_MY)
            * Store these values in a vector "Codes_Mean" (which is an
            temporary output)
            * Calculate the mean of this vector:
            Code_MY = ∑(codes(lab 130))+∑(codes(lab100)*0.35)+∑(codes(lab110)*0.65)+∑(codes(lab150)*0.15)
                      ---------------------------------------------------------------------------------
                                                    Ambiguity
        end
    end

    Write the unique label, NB_LAB_MY, in the corresponding output raster file
    CLASSIF2_MY

    Write the corresponding classification probability, CODE_MY, in the
    corresponding output raster file Code_CLASSIF2_MY

    Write the ambiguity code, Ambiguity, reflecting the number of single-years
    classified with the selected label in the corresponding output raster file
    AMB_CODE_CLASSIF2_MY

end for p
```

Algorithm 3-12. Aggregation of single-year LC maps (derived from the spectral unsupervised algorithm) into a multi-year LC map

3.5 Detailed processing scheme of the land cover maps merging (step 3)

At this stage of the classification chain, two global land cover maps have been produced:

- the map called CLASSIF1_MY, resulting from the machine learning classification approach detailed throughout the section 3.3;
- the map called CLASSIF2_MY, resulting from the unsupervised classification approach detailed throughout section the 3.4.

In both cases, the "multi-year" map was obtained either by classifying multi-year composites or by aggregating single-year LC maps into a multi-year map. This is these multi-year LC maps that will be combined in this step. The combination will be done according to objective decision criteria to generate a unique land cover map (CLASSIF_3). This unique LC map is the output of the CCI classification chain. The decision criteria between CLASSIF_1 and CLASSIF_2 are class-specific and are provided in a set of decision rules.

- ***Algorithm assumptions and limitations***

None

- ***Input and output data***

Input and output data associated with this merging process are described in Table 3-29.

Table 3-29: Input and output data for the step 3 of the classification chain, i.e. the merging of land cover maps obtained by the supervised and unsupervised classification approaches

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
CLASSIF1_MY	Land cover map resulting from the supervised classification algorithm, where each pixel is associated with a land cover class through an ID	IN	None	[0 … 255]
Code_CLASSIF1_MY	Classification probability associated with the label selected for each pixel	IN	None	[0 … 1]
CLASSIF2_MY	Land cover map resulting from the unsupervised classification approach (ISODATA algorithm + labelling process), where each pixel is associated with a land cover class through an ID)	IN	None	[0 … 255]
Code_CLASSIF2_MY	Ambiguity code that characterizes the ambiguity of the classification process, thus reflecting, at the pixel level, the reliability of the classification	IN	None	[0 … 10]
CLASSIF3	Land cover map resulting from the spectral classification approach (both supervised and unsupervised), where each pixel is associated with a land cover class through an ID	OUT	None	[0 … 255]
Code_CLASSIF3	Quality flag that characterizes, at the pixel level, the reliability of the classification	OUT	None	[0 … 100]
Source_CLASSIF3	Flag that indicates, at the pixel level, if the land cover label is derived from the supervised (CLASSIF1) or unsupervised (CLASSIF2) algorithm	OUT	None	[1, 2]

- ***Parameters***

No parameters are required to merge the supervised and unsupervised maps. Yet, a critical input of this step is the set of decision rules that guides the merging. They are provided in Table 3-30.

Table 3-30. Decision rules defined to merge the supervised and unsupervised land cover maps

```
Decision rules are:
if CLASSIF1_MY = (160, 170, 160 or 190), do:
     CLASSIF3 = CLASSIF1_MY
     Code_CLASSIF3 = Code_CLASSIF1_MY
     Source_CLASSIF3 = 1
```

```
if CLASSIF2_MY = (30 or 40) and CLASSIF1_MY = (10 or 20), do:
     CLASSIF3 = CLASSIF1_MY
     Code_CLASSIF3 = Code_CLASSIF1_MY
     Source_CLASSIF3 = 1
if CLASSIF2_MY = (100 or 110) and CLASSIF1_MY = (50, 60, 70, 80 or 90), do:
     CLASSIF3 = CLASSIF1_MY
     Code_CLASSIF3 = Code_CLASSIF1_MY
     Source_CLASSIF3 = 1
```

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
algorithm CLASSIF_1_2_Merging is
     input:
     CLASSIF1_MY: raster where each pixel is associated with a land cover class
     (described with an ID = NB_LAB and a name = LABEL). It results from the
     supervised classification step
     Code_CLASSIF1_MY : raster providing for each pixel the ambiguity level of
     the supervised classification approach.
     CLASSIF2_MY: raster where each pixel is associated with a land cover class
     (described with an ID = NB_LAB and a name = LABEL). It results from the
     unsupervised classification step
     Code_CLASSIF2_MY : raster providing for each pixel the ambiguity level of
     the unsupervised classification approach.

     output:
     CLASSIF3: raster where each pixel is associated with a land cover class
     (described with an ID = NB_LAB and a name = LABEL). It results from the
     merging of the previous CLASSIF_1 and CLASSIF_2 maps
     Code_CLASSIF3 : raster providing for each pixel the reliability of the
     land cover map obtained through the whole classification chain (both
     supervised and unsupervised algorithms)
     Source_CLASSIF3 : raster indicating for each pixel if the land cover
     label is derived from the supervised or unsupervised classification
     algorithm (CLASSIF_1 or CLASSIF_2)

for each pixel p do:
     CLASSIF3 = CLASSIF2_MY
     Code_CLASSIF3 = Code_CLASSIF2_MY
     Source_CLASSIF3 = 2
     Update the CLASSIF_3, Code_CLASSIF3 and Source_CLASSIF3 raster
     files in running the decision rules (see Table 3-30)
end for p
```

Algorithm 3-13. Supervised and unsupervised land cover maps merging

3.6 Detailed processing scheme of the post-classification editions (step 4)

The last step of the classification chain consists in bringing improvements thanks to existing external datasets. It should allow (i) correcting for some errors resulting from the algorithms and (ii) taking most benefit of existing high quality thematic products.

It is impossible to document for once the corrections that are made manually, as they differ according to the classified year and they change as soon as the reference LC database is modified (i.e. at each map update).

Yet, it is possible to document the integration of existing external dataset and it is done here below for the key datasets: the MERIS RR map, the CCI.

The integration is done using a C++ code specifically developed which defines the rules for each dataset.

- ***Algorithm assumptions and limitations***

None

- ***Input and output data***

Input and output data associated with this merging process are described in Table 3-31.

Table 3-31: Input and output data for the step 4 of the classification chain, i.e. the addition of key thematic information in the classification output

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
CLASSIF3_FR	Land cover map resulting from CCI classification chain applied on MERIS FR seasonal composites, where each pixel is associated with a land cover class through an ID	IN	None	[0 … 255]
Code_ CLASSIF3_FR	Quality flag that characterizes, at the pixel level, the reliability of the CLASSIF3_FR	IN	None	[0 … 100]
Source_CLASSIF3_FR	Flag that indicates, at the pixel level, if the land cover label of the CLASSIF3_FR is derived from the supervised or unsupervised classification algorithm	IN	None	[1, 2]
CLASSIF3_RR	Land cover map resulting from the CCI classification chain applied on MERIS RR seasonal composites, where each pixel is associated with a land cover class through an ID	IN	None	[0 … 255]

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
Code_CLASSIF3_RR	Quality flag that characterizes, at the pixel level, the reliability of the CLASSIF3_RR	IN	None	[0 … 100]
Source_CLASSIF3_RR	Flag that indicates, at the pixel level, if the land cover label of the CLASSIF3_RR is derived from the supervised or unsupervised classification algorithm	IN	None	[1, 2]
CCI_WB_300m	CCI Water Body product at 300m spatial resolution, where each pixel is associated with a water – no water status	IN	None	[1, 2]
Global_Cropland_Extent	Binary product resampled at 300m spatial resolution, where each pixel is associated with a crop – no crop status	IN	None	[0 … 255]
MODIS_Urban_Extent	Binary product resampled at 300m spatial resolution, where each pixel is associated with a urban – no urban status	IN	None	[0 … 255]
JRC Global Human Settlement Layer	Binary product resampled at 300m spatial resolution, where each pixel is associated with a urban – no urban status	IN	None	[0 … 255]
Global_Mangrove_Atlas	Binary product resampled at 300m spatial resolution, where each pixel is associated with a mangrove – no mangrove status	IN	None	[0 … 255]
REF_LC	Reference land cover layer where each pixel is associated with a land cover class through an ID	IN	None	[0 … 255]
CCI_Baseline_LC_Map	Land cover map resulting from the edition of the step 3 output (CLASSIF 3) through a posteriori addition of external dataset, where each pixel is associated with a land cover class through an ID	OUT	None	[0 … 255]
Code_CCI_Baseline_LC_Map	Quality flag that characterizes, at the pixel level, the reliability of the CCI_Baseline_LC_Map	OUT	None	[0 … 100]
Source_CCI_Baseline_LC_Map	Flag that indicates, at the pixel level, the source of the label of the CCI_Baseline_LC_Map	OUT	None	[1, 10]

- ***Parameters***

No specific parameters need to be implemented.

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
    algorithm Assembling_Reference is

        input:
        CLASSIF3_FR: raster where each pixel is associated with a land cover class
        (described with an ID = NB_LAB and a name = LABEL). It results from the
        merging of the previous CLASSIF_1 and CLASSIF_2 maps (obtained with MERIS
        FR)
        Code_CLASSIF3_FR : raster providing for each pixel the reliability of the
        land cover map obtained through the whole classification chain (both
        supervised and unsupervised algorithms run on MERIS FR composites)
        Source_CLASSIF3_FR : raster indicating for each pixel if the land cover
        label is derived from the supervised or unsupervised classification
        algorithm (CLASSIF_1 or CLASSIF_2) run on the MERIS FR composites
        CLASSIF3_RR: raster where each pixel is associated with a land cover class
        (described with an ID = NB_LAB and a name = LABEL). It results from the
        merging of the previous CLASSIF_1 and CLASSIF_2 maps (obtained with MERIS
        RR)
        Code_CLASSIF3_RR : raster providing for each pixel the reliability of the
        land cover map obtained through the whole classification chain (both
        supervised and unsupervised algorithms run on MERIS RR composites)
        Source_CLASSIF3_RR : raster indicating for each pixel if the land cover
        label is derived from the supervised or unsupervised classification
        algorithm (CLASSIF_1 or CLASSIF_2) run on the MERIS RR composites
        CCI_WB_300m: global raster rile where each pixel is associated with a water
        – no water status
        Global_Cropland_Extent: global raster rile where each pixel is associated
        with a crop – no crop status
        MODIS_Urban_Extent: global raster rile where each pixel is associated with
        a urban – no urban status
        GHSL_Urban_Extent: global raster rile where each pixel is associated with
        a urban – no urban status
        Global_Mangrove_Atlas: global raster rile where each pixel is associated
        with a mangrove -  no mangrove status
        REF_LC: reference land cover layer


        output:
        CCI_Baseline_LC_Map: raster file where each pixel is associated with a land
        cover class (described with an ID = NB_LAB and a name = LABEL
        Code_CCI_Baseline_LC_Map : raster providing for each pixel the reliability
        of the CCI_Baseline_LC_Map (from 0 to 100)
        Source_CCI_Baseline_LC_Map : raster providing for each pixel the the source
    of the label of the CCI_Baseline_LC_Map

for each pixel p, do:

     CCI_Baseline_LC_Map = CLASSIF3_FR
     Code_CCI_Baseline_LC_Map = Code_CLASSIF3_FR
     Source_CCI_Baseline_LC_Map = Source_CLASSIF3_FR

     /*Fill the gaps of the FR classification with the RR classification*/
```

```
     if NB_LAB(CCI_Baseline_LC_Map) = 0:
          CCI_Baseline_LC_Map = CLASSIF3_RR
          Code_CCI_Baseline_LC_Map = Code_CLASSIF3_RR
          Source_CCI_Baseline_LC_Map = Source_CLASSIF3_RR + 2 /*meaning the sources
          3 and 4 will indicate supervised algorithm on RR and unsupervised
          algorithm on RR, respectively*/
     end

     /*Make the CCI_Baseline_LC Map consistent with the CCI Water Body product*/
     if CCI_WB_300m = 2 and CCI_Baseline_LC_Map >< (20, 160, 170, 180, 210, 220):
          CCI_Baseline_LC_Map = 210
          Code_CCI_Baseline_LC_Map = 100
          Source_CCI_Baseline_LC_Map = 5
     end

     /*Add cropland areas in the CCI_Baseline_LC Map based on high-quality global
     cropland mask*/
     if Global_Cropland_Extent = 2 and CCI_Baseline_LC_Map >< (10, 11, 12, 20):
          CCI_Baseline_LC_Map = 10
          Code_CCI_Baseline_LC_Map = 100
          Source_CCI_Baseline_LC_Map = 6
     end

     /*Add urban areas in the CCI_Baseline_LC Map based on high-quality global urban
     mask*/

     if GHSL_Urban_Extent== 1 // GHSL_Urban_Extent is urban and CCI_Baseline_LC_Map
>< (190):
          CCI_Baseline_LC_Map = 190
          Code_CCI_Baseline_LC_Map = 100
          Source_CCI_Baseline_LC_Map = 7
     else    if    ((MODIS_Urban_Extent)    =    2    AND    GHSL_Urban_Extent==0))    and
CCI_Baseline_LC_Map >< (190):
          CCI_Baseline_LC_Map = 190
          Code_CCI_Baseline_LC_Map = 100
          Source_CCI_Baseline_LC_Map = 7

     end

     /*Add urban areas in the CCI_Baseline_LC Map based on high-quality global
     mangrove mask*/
     if Global_Mangrove_Atlas = 2 and CCI_Baseline_LC_Map >< (170):
          CCI_Baseline_LC_Map = 170
          Code_CCI_Baseline_LC_Map = 100
          Source_CCI_Baseline_LC_Map = 8
     end

     /*Fill the possible remaining gaps of the CCI Baseline LC maps using the
     reference LC database*/
     if NB_LAB(CCI_Baseline_LC_Map) = 0:
          CCI_Baseline_LC_Map = REF_LC
          Code_CCI_Baseline_LC_Map = 100
          Source_CCI_Baseline_LC_Map = 9
     end

end
```

Algorithm 3-14. A posteriori addition of external datasets.

4 GENERATION OF GLOBAL ANNUAL LC MAPS

The method implemented to derive global annual LC maps from the baseline global LC map is illustrated in Figure 4-1. The generation of these maps is organized in 3 major steps: (i) the annual change detection at 1 km, (ii) the change delineation at 300 m and (iii) the baseline update. These processes are detailed in the following sections.

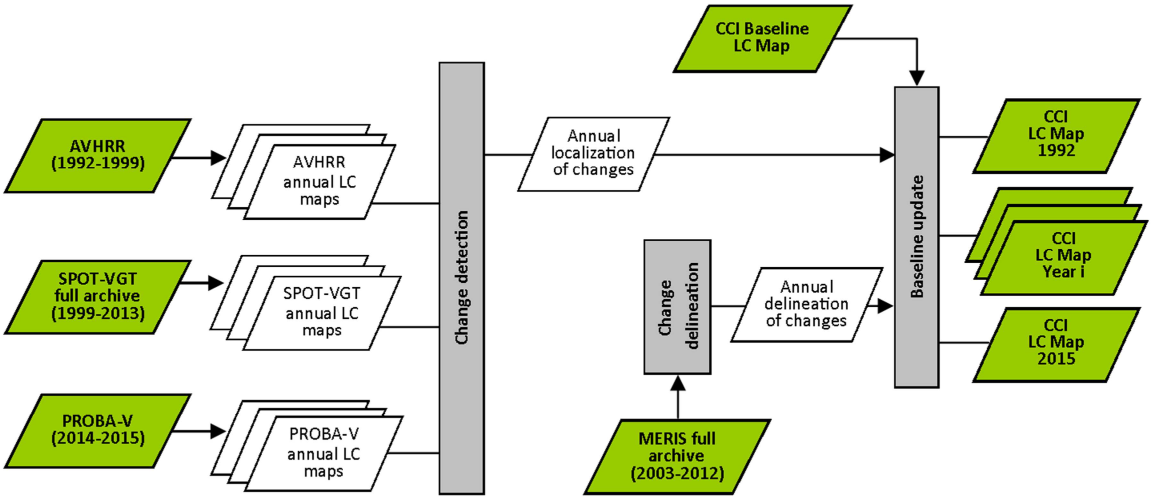

Figure 4-1: Schematic representation of the methodology developed to derive global annual LC maps from the baseline global LC map.

4.1 Change detection at 1 km

The first step of the change module consists in mapping the dynamics of the land surface by analysing, on a per-pixel basis, annual time series of 1-km global classifications from 1992 to 2015. The algorithm uses as input AVHRR classifications from 1992 to 1999, SPOT-VGT annual classifications from 1999 to 2013 and PROBA-V annual classifications from 2014 to 2015. These annual classifications are unsupervised spectral classifications using the ISODATA algorithm and automatic reference-based labelling as described in section 3.4.1 and 3.4.2.

With their 1 km resolution, analysing the sequence of global LC classifications over time allows capturing the dominant land cover transitions. Yet, in order to avoid false change detections due to the inter-annual variability in classifications, each change has to be confirmed over more than two successive years in the classification time series.

In the most dynamic regions of the world, more than one land cover change can be detected between 1992 and 2015. Most of the pixels are associated with 0, 1, 2 or 3 land cover changes, knowing that each change needs to last at least two years to be detected.

The change detection method allows determining (i) the year of change and (ii) the type of change that is observed which will eventually serve as input to the change delineation step in section 4.2. It is applied on a time window of 14 years (W in Table 4-4) to allow a robust, yet flexible, change detection when the LC map time series is extended.

- ***Change detection between IPCC classes***

The change is detected on a pixel basis between groups of classes which are slightly different from the CCI LC legend (described in LUT 1 in Table 3-10). The classes considered here are: cropland, forest, grassland, wetland, settlement and others (shrubland, sparse vegetation, bare area, water). This grouping correspond to the main Intergovernmental Panel on Climate Change (IPCC) land categories [RD.19], which was a requirement expressed by the climate users [AD.3, AD.4]. The correspondence between these groups of classes is defined in Table 4-1. It is used to generate the input "IPCC_Grouping".

Table 4-1: Correspondence between the IPCC classes and the LCCS classes of the LC maps legend.

CLASSES CONSIDERED FOR THE CHANGE DETECTION	CCI LAND COVER CLASSES	
Cropland	10, 11, 12	Rainfed cropland
	20	Irrigated cropland
	30	Mosaic cropland (>50%) / natural vegetation (tree, shrub, herbaceous cover) (<50%)
	40	Mosaic natural vegetation (tree, shrub, herbaceous cover) (>50%) / cropland (< 50%)
Forest	50	Tree cover, broadleaved, evergreen, closed to open (>15%)
	60, 61, 62	Tree cover, broadleaved, deciduous, closed to open (> 15%)
	70, 71, 72	Tree cover, needleleaved, evergreen, closed to open (> 15%)
	80, 81, 82	Tree cover, needleleaved, deciduous, closed to open (> 15%)
	90	Tree cover, mixed leaf type (broadleaved and needleleaved)
	100	Mosaic tree and shrub (>50%) / herbaceous cover (< 50%)
	160	Tree cover, flooded, fresh or brakish water
	170	Tree cover, flooded, saline water
Grassland	110	Mosaic herbaceous cover (>50%) / tree and shrub (<50%)
	130	Grassland
Wetland	180	Shrub or herbaceous cover, flooded, fresh-saline or brakish water

Classes considered for the change detection		CCI Land Cover classes	
Settlement		190	Urban
Other	Shrubland	120, 121, 122	Shrubland
	Sparse vegetation	140	Lichens and mosses
		150, 152, 153	Sparse vegetation (tree, shrub, herbaceous cover)
	Bare area	200, 201, 202	Bare areas
	Water	210	Water

- ***Twelve types of change are detected at 1 km***

The change detection method is applied over twelve types of changes (Table 4-2). Note that changes in the urban areas class are derived in a separate module (see section Table 4-4).

Table 4-2: Descriptions for each type of change, ID and band number associated in the output of the change detection at 1km

Change Type	ID	Band number	Description
Forest Loss	101	1	Changes from forest classes to crops, shrub, grasses, sparse vegetation, wetlands and bare areas
Forest Gain	102	2	Changes from crops, shrub, grasses, sparse vegetation, wetlands and bare areas to forest classes
Cropland Loss	103	3	Changes from crops classes to grasses, sparse vegetation and bare areas
Cropland Gain	104	4	Changes from grasses, sparse vegetation and bare areas to crop classes
Grassland Loss	105	5	Changes from grasses to sparse vegetation and bare areas, and from sparse vegetation to bare areas
Grassland Gain	106	6	Changes from sparse vegetation and bare areas to grasses, and from bare areas to sparse vegetation
Shrubland Loss	107	7	Changes from shrub to crops, grasses, sparse vegetation and bare areas
Shruland Gain	108	8	Changes from crops, grasses, sparse vegetation and bare areas to shrub
Wetland Loss	109	9	Changes from wetland to crops, grasses, shrub, sparse vegetation and bare areas
Wetland Gain	110	10	Changes from crops, grasses, shrub, sparse vegetation and bare areas to wetland
Water Loss	111	11	Changes from water to crops, forests, grasses, shrub, sparse vegetation, wetland and bare areas
Water Gain	112	12	Changes from crops, forests, grasses, shrub, sparse vegetation, wetland and bare areas to water

- ***Decision rules***

The change detection method is guided by decision rules which analyse the suite of annual classifications and determine if there has been or not a LC change during this period. These decision rules follow a general structure but are also fine-tuned per stratum and type of change. Those specific decision rules concern for example the number of years for which change should be confirmed over the period as well as the expected IPCC classes observed as a result of the change (Table 4-3).

The stratification parameters are contained in the LUT 2, described earlier in Table 3-11.

Table 4-3: Decision rules applied in the change detection algorithm at 1km for each stratum by type of change.

CHANGE TYPE	SPECIFIC AND GENERIC DECISION RULES PER STRATUM
Forest Loss	Specific decision rules for strata 1 – 6, 14, 16, 20
	Specific decision rules for strata 7 – 13, 15, 17 – 19, 22
	Specific decision rules for stratum 21
Forest Gain	Specific decision rules for strata 2 – 12, 14 – 20, 21
	Specific decision rules for stratum 13
	Specific decision rules for stratum 1
	Specific decision rules for stratum 21
Cropland Loss	Generic for all strata
Cropland Gain	Specific decision rules for strata 1 – 11, 14, 16, 17, 19 – 21
	Specific decision rules for stratum 18
	Specific decision rules for strata 12, 13, 15, 22
Grassland Loss	Generic for all strata
Grassland Gain	Generic for all strata
Shrubland Loss	Specific decision rules from strata 1 – 20 and 22
	Specific decision rules for stratum 21
Shrubland Gain	Generic for all strata
Wetland Loss	Generic for all strata
Wetland Gain	Generic for all strata
Water Loss	Generic for all strata
Water Gain	Generic for all strata

The generic decision rules for the change detection at 1 km over the 14-y window follow a series of three conditionals statements for each type of change which aim at:

- Identifying if a potential change exists between two different groups of IPCC classes by sequentially scanning the series of annual LC classifications;

- Identifying if, among the potential changes, some are confirmed over time by observing the same post-change class over several years;
- Identifying the year of change as the year of the first LC class that does not belong anymore to the original group of IPCC class.

- ***Algorithm assumptions and limitations***

The common pre-processing chain applied over the various sensors provides stable surface reflectance from sensor to the other. The annual LC classifications reached a high level of accuracy allowing a comparison of LC labels over a time series.

- ***Input and output data***

Input and output data associated with the change detection process are described in Table 4-4.

Table 4-4: Input and output data for the change detection at the stratum level

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
IPCC_Grouping_1_14_<NB_ST>_<W>	Matrix for each stratum (NB_ST) converting the LCCS legend to the IPCC legend based on Table 4-1Table 4-1 on each time window (W)	IN	None	[1…10]
M_1_14_<NB_ST>_<W>(i)	Matrix at the stratum level comprising the annual SPOT-VGT and PROBA-V classifications reshaped in vectors, for each stratum (NB_ST) and each time window (W). i = pixel	IN	None	[0…220]
ChangeYear_<NB_ST>_<W> (band)	Multi-band raster at the stratum level of the year of change by type of change (1 band by type of change) for each time window (W).	INOUT	None	[2004… 2017]
ChangeType_<NB_ST>_<W> (band)	Multi-band raster at the stratum level of the type of change for each time window (W). Each code corresponds to the change type ID in Table 4-3	OUT	None	[101… 114]

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
algorithm Generic_decision_rules is


for each pixel i, do:
       for each type of change tc from Table 4-1, band (where tc and band are the id
and band number given in Table 4-2), do:
              group the original LCCS classes in PCC groups using Table 4-1
              scan the series of annual LC classifications inside W
              if  potential  change  exists  between  two  different  groups  of  IPCC
classes:
                 if potential change is confirmed:
                    identify  the  first  year  of  change  in  LC  and  record  it  in
ChangeYear_ST_W(band)
                    record the type of change in layer ChangeType_ST_W(band)
```

Algorithm 4-1. Generic decision rules applied in the change detection at 1 km on a 14-year window.

```
algorithm Change_detection is

input:
M_1_14_<NB_ST>_<W>(i): matrix (one for each stratum NB_ST in LUT2 and for each time
window (W) comprising 14 annual classifications reshaped in vectors.(i = pixel, j =
year)
IPCC_Grouping_1_14_<NB_ST>_<W>  :  matrix  (one  for  each  stratum  NB_ST  and  for  each
time  window  (W)  compromising  14  annual  classifications  reshaped  in  vectors  and
converted to the IPCC legend.
input/output:
ChangeYear_<NB_ST>_<W> (band): multi-band raster (one for each stratum NB_ST in LUT2
and for each time window (W) compromising the year of change for each type of change
(one band by type of change)
output:
ChangeType_<NB_ST>_<W> (band): multi-band raster (one for each stratum NB_ST and for
each time window (W) compromising the type of change (one type of change by band).
for each stratum <NB_ST> do:
   for each window <W> do:
      for each pixel i, do:
          for each type of change <TC>, do :
             read the input data from M_1_14_<NB_ST>_<W>(i),
IPCC_Grouping_1_14_<NB_ST>_<W>
             run the generic decision rules of algorithm generic_decision_rules for
each <TC>
             run the specific decision rules according to Table 4-4.
             write the corresponding year_of_change in the raste
ChangeYear_<NB_ST>_<W> (band)
             write the corresponding <TC> ID in the raster ChangeType_<NB_ST>_<W>
(band)
          end
      end
   end
end
```

Algorithm 4-2. Change detection at 1 km on a 14-year window.

4.2 Change delineation at 300 m

The change information extracted from the 1 km time series was enhanced thanks to the higher spatial resolution of MERIS and PROBA-V between 2003 and 2015. The classification algorithm was the supervised machine learning algorithm described in section 3.3. The inputs images are the multiyear and annual composites from MERIS and PROBA-V described in section 3.2.3.2. For the MERIS data, multi-annual change-based composites have also been used, as described below.

4.2.1 Multi-annual change-based composite

The rationale of the multi-annual change-based composite is the same as the rationale behind the compositing process: increasing the number of observations for a given seasonal stage will reduce the noise of the signal. However, multi-annual composite assume that no land cover change occurred inside the pixel during the compositing period, which could make changed pixels undetectable. The multi-annual change-based composite aims at mitigating this issue by splitting the compositing period at the date of change in areas where the change is likely to occur.

- ***Algorithm assumptions and limitations***

There are two assumptions for this algorithm. First, it assumes that the land cover is stable between two dates of change. Second, it relies on stable radiometric corrections from year to year. The main limitation of the algorithm is the absence of data during the compositing period.

- ***Input and output data***

The input data for the multi-annual change based composite are the same as the input data of the multi-year change-based composite (see Section 3.2.3.2) with an additional layer: the 1 km change date derived in section 4.1 accompanied by a buffer described hereafter in section 4.4.

- ***Pseudo-code representation***

```
algorithm Change_delineation_300m is

input
Reflectance_raw_<DATE>_MERIS/PROBA
Buffer_ChangeYr_<NB_ST>

output
Reflectance_<YEAR>_<SEASONALITY>_MERIS/PROBA

For each strata in NB_ST, for each pixel:

  change_years = extract from Buffer_ChangeYr_<NB_ST>

  for each years_range around/between changes_years: # if no change_year => 1 period

    for each seasonality:

      sum = 0
      number = 0

      for each date in years_range and in seasonality:
```

```
if data in Reflectance_raw_<DATE>_MERIS/PROBA:

  sum = sum + Reflectance_raw_<DATE>_MERIS/PROBA
  number = number + 1

mean = sum / number

for each year in years_range:

  Reflectance_<YEAR>_<SEASONALITY>_MERIS/PROBA = mean
```

Algorithm 4-3. Change delineation at 300 m.

4.3 Change detection for the urban class

The methodology described in section 4.1 did not show satisfactory results to classify the urban class both spatially and temporarily. A new algorithm was therefore dedicated to urban change detection, for the 300 m era (i.e. 2003 onwards). Before 2003, the 1 km spatial resolution brought by AVHRR and SPOT-VGT did not allow for robust urban change detections. Between 1992 and 1999 and between 2000 and 2003, the urban footprint therefore corresponds to the GHSL 1990 and GHSL 2000, respectively. The urban footprint is stable inside and expands between each of these two periods.

From 2003 onwards, the urban change detection relied on a series of annual machine learning spectral classifications of MERIS 300 m from 2004 and 2011 and PROBA-V 300 m from 2012 and 2015. A filling procedure aims overcoming data gaps present on a yearly basis. Spatio-temporal consistency in the change detection was ensured by 2 constraints: (i) the urban footprint is only allowed to expand over time; (ii) the minimum and maximum urban footprints are constrained by the Landsat-based Joint Research Centre Global Human Settlement Layer (GHSL) [RD.23] resampled to 300 m for year 2000 and the GHSL of 2014, respectively. In addition, the Global Urban Footprint [RD.24] was fused to the GHSL 2014 to compensate for urban area omissions.

- ***Algorithm assumptions and limitations***

Assumptions are the following:

- urban footprints expand over time, with no destruction;
- the GHSL datasets for years 2000 and 2014 represent the urban state correctly;
- the error implied by using the GHSL of 2014 in 2015 is minimal at global scale.

Limitations concern:

- the lack of change detections between 1992 and 1999 and between 2000 and 2003;
- a temporal shift in the urban change detection could be present in highly cloud-covered areas resulting from the gap filling procedure.

- ***Input and output data***

Input and output data associated with the change detection for the urban class are described in Table 4-5.

Table 4-5: Input and output data for the change delineation at 300 m.

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
SUP_CLASSIF_300m_<sensor>_<year> where: <sensor> is “MERIS” or “PROBAV” <year> is in range [2003, 2015]	Land cover map resulting from the supervised classification algorithm, where each pixel is associated with a land cover class through an ID	IN	None	[0 … 255]
GHSL_<year> where <year> is 2000 OR 2014	Landsat-based binary layer with urban presence/absence for year <year> at ~38 m spatial resolution	IN	None	[0 … 190]
GHSL_300m_<year> where <year> is 2000 OR 2014	Binary layer with urban presence/absence for year <year> at 300 m spatial resolution	INOUT	None	[0 … 190]
U_Mask_<Year> where <year> is range [2003, 2015]	binary layers with urban presence/absence for year <year>	OUT	None	[0 … 190]

- ***Equations***

No specific equation was implemented.

- ***Pseudo-code representation***

```
algorithm Change_detection_Urban_300m is

     input:

     SUP_CLASSIF_300m_MERIS_<Year>: supervised classifications from 2003 to 2011
     SUP_CLASSIF_300m_PROBAV_<Year>: supervised classifications from 2012 to 2015
     GHSL_2000: binary layer with urban presence/absence for year 2000
     GHSL_2014: binary layer with urban presence/absence for year 2014
     GUF: binary layer with urban presence/absence

     output:

     U_Mask_<Year>: binary mask with urban presence/absence per year (Non-urban: 0 ;
Urban: 190)

     variables:

     Y: the year inside the temporal series. Varies between 2003 and 2015
     LAB_CL = the class label from the supervised classifications
“SUP_CLASSIF_300m_MERIS” and “SUP_CLASSIF_300m_PROBAV”
```

```
    Prop_CLU = proportion of the class urban weighted by the total number of years
inside the time series
    Y_TOT = total number of years in the temporal series.
    Y_END = last year of the time series
    Y1_CL = first year of appearance of the urban class within the time series

  * Resample the GHSL layers to 300 m resolution using a majority resampling*

  gdalwarp -of  GTiff -co COMPRESS=LZW -co TILED=YES -r Mode -tr 0.002777777701187
0.002777777701187 GHSL_2000.tif GHSL_300m_2000.tif

  gdalwarp -of  GTiff -co COMPRESS=LZW -co TILED=YES -r Mode -tr 0.002777777701187
0.002777777701187 GHSL_2014.tif GHSL_300m_2015.tif

  * Add the GUF to the GHSL layer 2014*

  for each stratum NB_ST (1 to 22 – LUT 2) do:
     for each for each pixel p in the search area, do:
        Read classification labels (LAB_CL) from 2003 to 2015:

        *Apply the gap-filling algorithm*


           For y in range(2003,1,2015):

              If GHSL_2000 == 190:
                  U_Mask_<Year> == 190

              If GHSL_2014 == 190:

                  if LAB_CLy == 190 (Urban):
                      if isNaN(LAB_CLy-1) AND isNaN(LAB_CLy+1):
                         LAB_CLy = NaN
                  if isNaN(LAB_CLy):
                      if (LAB_CLy-1 == 190) AND (LAB_CLy+1== 190):
                         LAB_CLy = 190

        *Compute the proportion of urban class in the time series*

           Prop_CLU = (sum(LAB_CL==190)/(sum (LAB_CL==190, LAB_CL!=190))* Y_TOT
           Y1_CL = Y_END-int(Prop_CLU)

        * Write the urban masks for each year of the time series*

           For Y < Y1_CLU:
              U_Mask_<Year> = 0
           Else:
              U_Mask_<Year> == 190
```

Algorithm 4-4. Change detection for urban at 300 m.

4.4 Baseline update

The changes were identified first at 1 km based on AVHRR, SPOT-VGT and PROBA-V classifications. For years between 2004 and 2015, the 1-km spots of change were further delineated at 300 m with MERIS and PROBA-V 7-day time series. This information is now used to derive annual LC maps from the baseline LC map.

This "dating" process is done at 300 m spatial resolution when possible, i.e. between 2004 and 2015 which are covered by MERIS data and PROBA-V at 300 m. Global LC maps from 1992 to 2003 are only imaged with AVHRR and SPOT-VGT data. The changes specific to these years will therefore appear at 1 km in the map. The change detected with AVHRR between 1992 and 1999 is limited to the "Forest loss and gain" change types, due to low accuracy in reflectances.

The baseline is updated using the layers produced in steps 4.1 and 4.2. Under the change detection (at 1km), the annual classifications (at 300m or 1km depending on the year) are compared to the baseline. If the classification indicates a change during a specific year, the label of the baseline can be modified from this specific year, if it respects certain rules.

The detected changes from step 4.1 are cleaned with the combination of a connection algorithm, followed with an erosion algorithm. It removes isolated pixels more susceptible to be a false change detection. The remaining changes are extended with a buffer (to avoid artefacts coming from the 1km to 300m change of resolution). Inside this buffer the rules are more restrictive. The buffer is not applied for years at 1km of resolution (1992-2003).

The following sections describe the baseline update in a generic manner for both changes at 1-km and 300 m spatial resolution.

- ***Algorithm assumptions and limitations***

None

- ***Input and output data***

Input and output data associated with the change detection process are described in Table 4-6.

Table 4-6: Input and output data for the change delineation and baseline update

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
CCI_Baseline_LC_Map	Land cover map resulting from the edition of the step 3 output (CLASSIF 3) through a posteriori addition of external dataset, where each pixel is associated with a land cover class through an ID	IN	None	[0 … 255]
Buffer_ChangeYr_1km_<NB_ST>	Multi-band raster of the year of land cover change (value 0 indicating no change), available at the stratum level. Several bands are used if several changes occur inside a given pixel. The MERIS & PROBAV changes are merged with the SPOTVGT changes. A majority buffer of 5x5 km is applied around the pixels of step 4.1, after the application of	IN	None	[1992 … 2015]

DATA	DESCRIPTION	INTENT (IN, OUT, INOUT)	PHYSICAL UNIT	RANGE
	a connection and an erosion algorithm. The buffer is tagged to discern originals from added. GeoTiff format.			
Buffer_ChangeTy_1km_ <NB_ST>	Multi-band raster of the thirteen types of change (value 0 indicating no change) available at the stratum level The comments are the same as for ChangeYr GeoTiff format.	IN	None	[0 … 210]
Classif_<YEAR>_AVHRR	Raster files where each pixel is associated with a land cover class, identical as in step 4.1.	IN	None	[0 … 255]
Classif_<YEAR>_SPOTVGT	Raster files where each pixel is associated with a land cover class, identical as in step 4.1.	IN	None	[0 … 255]
Classif_<YEAR>_MERIS	Raster file where each pixel is associated with a land cover class, generated in step raster files where each pixel is associated with a land cover class, identical as in step 4.2.	IN	None	[0 … 255]
Classif_<YEAR>_PROBAV	Raster files where each pixel is associated with a land cover class	IN	None	[0 … 255]
CCI_Urban_Mask_<YEAR>	Raster files where a value of 0 or 190 (urban class) is set annually	IN	None	0 or 190
CCI_LC_Map_<year> where <year> can vary from 1992 to 2015	Land cover map representative of year <year>, derived from the Baseline map with the outputs of the change detection, where each pixel is associated with a land cover class through an ID	OUT	None	[0 … 210]

- ***Parameters***

The algorithm is driven by a set of decision rules that are used to modify the baseline map and derive the global annual LC mas. Those rules are based on the labels before the change (BEF_CLASS), the type of change (CHGT_TYPE) and the label of the fate of the change (FATE_CLASS). The links between those labels are given in Table 4-7.

Table 4-7: Decision rules to convert the LC baseline to annual LC maps using the change layers.

CHGT_TYPE	BEFCLASS	FATE _CLASS
Forest Loss	50 to 100, 160 or 170	10 to 40, 110, 120, 130, 150, 180 or 200
Forest gain	10 to 40, 110, 120, 130, 150, 180 or 200	50 to 100, 160 or 170
Cropland loss	10 to 40	110, 130, 150 or 200
Cropland gain	110, 130, 150 or 200	10 to 40
Grassland loss	110, 130 150	150, 200 200
Grassland gain	150, 200 200	110, 130 150
Shrubland loss	120	10 to 40, 110, 130, 150, 200
Shrubland gain	10 to 40, 110, 130, 150, 200	120
Wetland loss	180	10 to 40, 110, 120, 130, 150, 200
Wetland gain	10 to 40, 110, 120, 130, 150, 200	180
Water loss	210	10 to 40, 50 to 100, 160 or 170, 110, 120, 130, 150, 180, 200
Water gain	10 to 40, 50 to 100, 160 or 170, 110, 120, 130, 150, 180, 200	210
Urban gain	10 to 40, 50 to 100, 160 or 170, 110, 120, 130, 150, 180, 200	190

Table 4-8: Paramaters for the baseline update.

PARAMETERS	DESCRIPTION	INTENT (IN, OUT, INOUT)	FORMAT	RANGE
Change_threshold	Percentage of minimum valid classes to validate a period between changes or first/last periods	INOUT	float	[0..1]
Change_threshold_buffer	Identical to change_threshold, applied only under the buffer	INOUT	float	[0..1]
Weight_<year>	A weight fixed by annual classification quality and spatial resolution based on expert knowledge, different for each year. E.g. classifications resulting from AVHRR are more	INOUT	Float	[0..1]

PARAMETERS	DESCRIPTION	INTENT (IN, OUT, INOUT)	FORMAT	RANGE
	unstable and are at 1km.			

- ***Equations***

No specific equations need to be implemented.

- ***Pseudo-code representation***

```
algorithm Baseline_Update is

      inputs (see Table 4-6 for description):

      Buffer_ChangeYr_<NB_ST>
      Buffer_ChangeTy_<NB_ST>
      // According to the year of change, one of those layers will be selected as
      input. In the pseudo-code, the generic term “Classif” is used.
      Classif_<YEAR>:
        Case YEAR in 1992-1998:
          Classif_<YEAR>_AVHRR
        Case YEAR in 1999-2003:
          Classif_<YEAR>_SPOTVGT
        Case YEAR in 2004-2012:
          Classif_<YEAR>_MERIS
        Case YEAR in 2013-2015:
          Classif_<YEAR>_PROBA-V
      CCI_Baseline_LC_Map
      LUT in Table 4 18 that contains, for each change type, the associated labels
      for the change class (named FATE_CLASS) and the “before the change class”
      (named BEF_CLASS) to be tested. If two changes are detected,variables are
      describes according the number of the change : BEF_CLASS1, FATE_CLASS1 =
      BEF_CLASS2, FATE_CLASS2.

      Parameters

      Change_threshold = 20%
      Change_threshold_buffer = 80%
      # list from 1992 to 2015
      Weights = [0.2 0.2 0 0.2 0.2 0.2 0.2 0.3 0.3 0.3 0.3 0.3 1 1 1 1 1 1 1 1 0.5
      0.5 1 1]

      outputs
      CCI_LC_Map_[year]: raster file where each pixel is associated with a land
      cover class (described with an ID = NB_LAB and a name = LABEL) representative
      of <year> varying from 1992 to 2015.

For each strata in NB_ST:
For each pixel:

  change_years = extract from Buffer_ChangeYr_<NB_ST>

  change_types = extract from Buffer_ChangeTy_<NB_ST>

  doChange = true

  if no change_years or (change_years < 2004 and under the buffer):
```

```
    doChange = false
    (skip next for loop)
  for each period around/between changes_years:
    expected_list_bef = from Table 4-21 corresponding to change_types[period]
    expected_list_fate = from Table 4-21 corresponding to change_types[period]
    proportion = 0
    possible_classes = empty list
    for each YEAR in period:
      class = Classif_YEAR
      weight = Weights[YEAR]
      if class in expected_list_bef and expected_list_fate:
        proportion = proportion + weight
        add class to possible_classes
    proportion = proportion / sum(weights in period)
    if proportion >= Change_threshold: (change_threshold_buffer if under the buffer)
      for each YEAR in period:
        tmp_CCI_LC_Map_<YEAR> = majority in possible_classes or baseline
        # baseline is chosen if the thematic accuracy is better
        # baseline must be placed in one of the periods
    else:
      doChange = false
  if doChange:
    CCI_LC_Map_<YEAR> = tmp_CCI_LC_Map_<YEAR> (for all YEARS)
  Else:
    CCI_LC_Map_<YEAR> = CCI_Baseline_LC_Map

  # application of the urban mask
  If CCI_Urban_Mask_<YEAR> = urban:
    CCI_LC_Map_<YEAR> = urban class
Write CCI_LC_Map_<year> in the corresponding output raster files
```

Algorithm 4-5. Baseline update

5 QUALITY CONTROL PROCEDURE

5.1 Introduction and scope

A confidence-building procedure must be integrated into the classification chain and will consist in a systematic quality control of the CCI LC products. Importance of the quality control procedure is highlighted by the fact that previous global LC products, although of good overall quality, exhibit in some areas macroscopic errors that could have been avoided by a careful review of the draft products. Errors affecting accuracy of thematic maps can be either thematic - confusion between the LC classes (wrong label, missing classes) or spatial (wrong position of the boundary between classes, disappearance of small patches).

By the elimination of macroscopic errors, the systematic quality control is intended to increase the overall acceptance of the LC product by users. Quality control is also a way of assessing if the remotely sensed data have been correctly classified, i.e. if the errors are due to limitations of data quality rather than to inappropriate classification procedures. Moreover, quality control assesses spatial distribution of errors and presents achieved performances on maps. The procedure contributes to increase the quality of classification by using the results of the analysis to remove errors and improve the output.

5.2 Systematic protocol and error detection

The quality control is a mandatory step before the release of any final product. Qualitative control is based on a systematic descriptive protocol. Within the procedure, the global map is divided into regular grids of which each cell is visually examined and its accuracy documented in terms of type of error, landscape pattern, reference material used, etc. Maps were evaluated by equal reasoning areas used in the classification chain. Naturally, landscape patterns from one area to another have different degree of heterogeneity. For example, in the central part of the Amazon Basin or in the heart of the Sahara, the grid cells are much more homogeneous and could thus be much larger than in the complex landscapes of Western Europe. However, considering that purpose of quality control is rapid assessment of map product, the cell size will be based on geographic grid. Higher attention will be given during the quality control to cells with heterogeneous areas. Another important factor which was taken into consideration is focus more on areas of major discrepancies between global LC products.

In the assessment, grid cell size covering landscapes between 45° S and 45° N were of 1×1 degree. Areas north of 45° N and South of 45° were of 1×2 degrees. As illustration, Figure 5-1 presents the grid used for systematic quality control exercise over central Africa. Each cell is associated with an alphanumeric identification code providing information about latitude and longitude of lower left corner of the cell (Figure 5-2). Purpose of using grid based on geographic coordinates is to provide clear information about location of the cell and ease structure of the database.

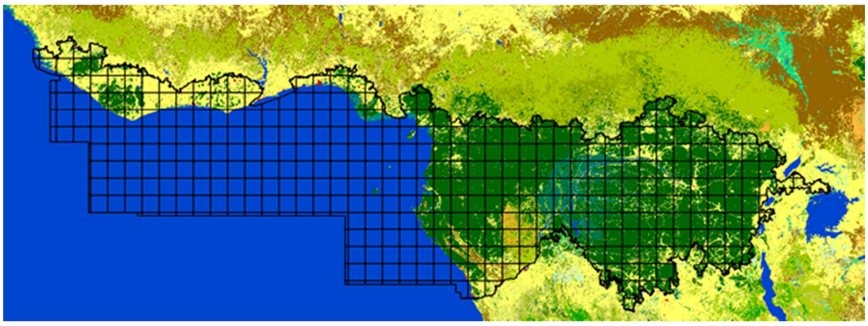

Figure 5-1: Regular grid used for systematic quality control over central Africa.

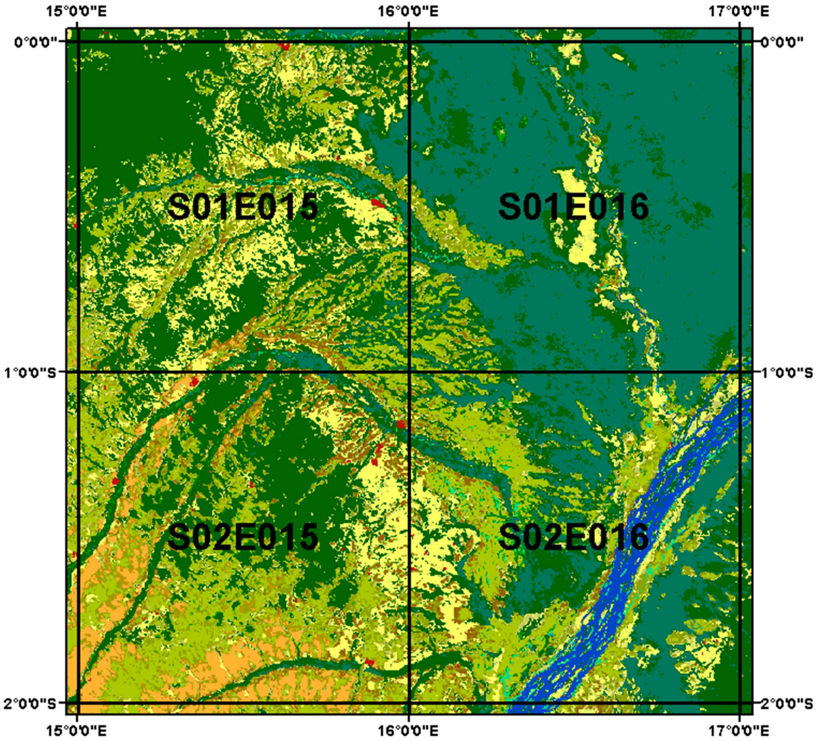

Figure 5-2: Example of quality control cells and identification label.

For the systematic quality assessment, different reference materials were used, including single-date coarse resolution images, detailed thematic maps, and quick-look imagery derived from fine-resolution sensors (Google Earth). Pre-processing and classification procedures applied to multi-date imagery often lead to the loss of many spatial details that are clearly visible on original images or temporal synthesis. This loss of detail is particularly obvious when long time series of derived parameters such as vegetation indexes are used as input for the classification. For each cell, image products will be overlaid over the product to evaluate and compare using image/map analysis tools such as blend, flicker and swipe.

5.3 Quality control

In the systematic quality control procedure, each cell is examined and characterized in detail by a few parameters: the composition and the spatial pattern of the cell, type of reference dataset used, the overall quality of the cell, and the typology of any errors. These parameters are summarized and recorded into database (Figure 5-4).

5.3.1 Land Cover

The cell composition is a key factor affecting the accuracy of a map because some LC classes (e.g. evergreen forests, deserts, water bodies) are easier to discriminate than others (e.g. deciduous forests or woodlands, grasslands, extensive agriculture). Information on the composition of the cell contributes to a better understanding of the errors and can help to stratify cell patterns representation. LC class composition will be calculated for each cell separately and stored in the database.

5.3.2 Spatial pattern

It is widely recognized that the spatial pattern of the landscape influences the appearance or disappearance of LC classes at varying resolution as well as the area estimates derived from coarse resolution maps. In order to assess spatial landscape composition over each cell, LC area fragmentation patterns will be evaluated. Fragmentation of specific area on land surface refers to the geometric complexity of a landscape as determined by the characteristics of the forest/non forest interface [RD.22] and can be applied on any of LC patterns combination. Four types of fragmentation patterns are recognized: linear, insular, diffuse and massive (Figure 5-3).

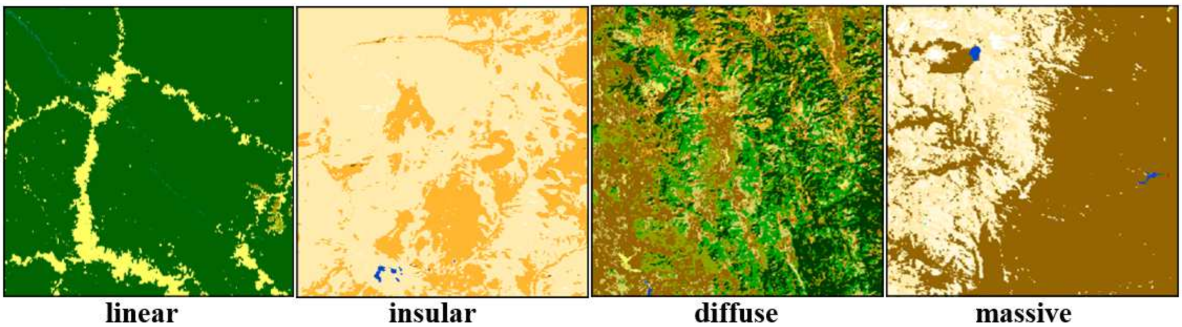

Figure 5-3: LC fragmentation patterns

5.3.3 Reference dataset quality

Quality assessment of each cell highly depends on the availability and quality of reference dataset based on which is the quality of the product evaluated. In the evaluation procedure, two main types of reference datasets are used: high resolution satellite images (or aerial photographs) and existing high resolution thematic maps. Best available dataset is selected.

5.3.4 Overall quality

As a first approximation, the overall quality of each cell will be categorized in qualitative classes using a nominal scale - good, moderate, low. As with qualitative labeling of heterogeneity, a catalogue of representative cases will be provided in order to ensure consistency. The labeling of overall quality, once performed for all the cells, allows for a synthetic spatial representation of the quality of the product.

Validation Cell ID	Evaluated LC Product	LC Classes	Expert Name
Spatial Pattern	**Reference Dataset Type**	**Overall Quality**	**Type of Error**
linear	satellite/aerial image	good	wrong label
insular	thematic map	moderate	wrong delineation
diffuse	image and map	low	wrong label and delineation
massive			

Figure 5-4: Summary of data parameters and values recorded in quality control database.

5.4 Typology of errors

Ascertaining the nature of the errors occurring in the cell is of primary importance. Statistical accuracy assessment merges in the category “error” many different cases that quality control can easily document. Such information can be profitably used for improving the map during the updating phase.

The main cases that can be found in global products are the following:

- The delineation of a LC feature is accurate, but the label is wrong. In this case, the type of confusion must be specified in order to derive a thematic “distance” between the right and the wrong labels. It is, for example, generally more problematic to classify tropical forests as grasslands than to classify woodlands as savannas;
- The proportions of labels present in the cell are generally correct, but the delineation of the various features is wrong. If this case is the most frequent, it means that the spatial resolution (and eventually the pre-processing steps) precludes any accurate delineation of LC features. The extreme case of this category occurs when no clear structures appear on the map. The LC map then corresponds more to a climatic stratification;
- One important LC feature is missing in the map or a feature is mapped while it is not present in the field. This is a particular case combining a wrong label and an inaccurate delineation of the LC features. For example, it happens when specific features are introduced by erroneous ancillary data.

5.5 Presentation of the results

Following the quality control procedure when all the cells have been visited, obtained result from various fields have been stored in the database. This database is available as a separated excel file [AD.9].

This section presents the results such as a spatial distribution of errors, their categorization and proportion. Additionally, other parameters as data quality and spatial distribution of landscape patterns are presented. Results of the evaluation indicates possible causes of errors when interaction between different parameters and quality of data are outlined.

5.5.1 Error distribution within land cover classes

During the quality control procedure LC categories per each detected error were identified. As a next step were these errors corrected. Either the type of modification was a change of label or delineation of area covered by specific LC class, information about the categories concerning change was recorded. Figure 5-5 presents error distribution within CCI LC classes for all detected errors.

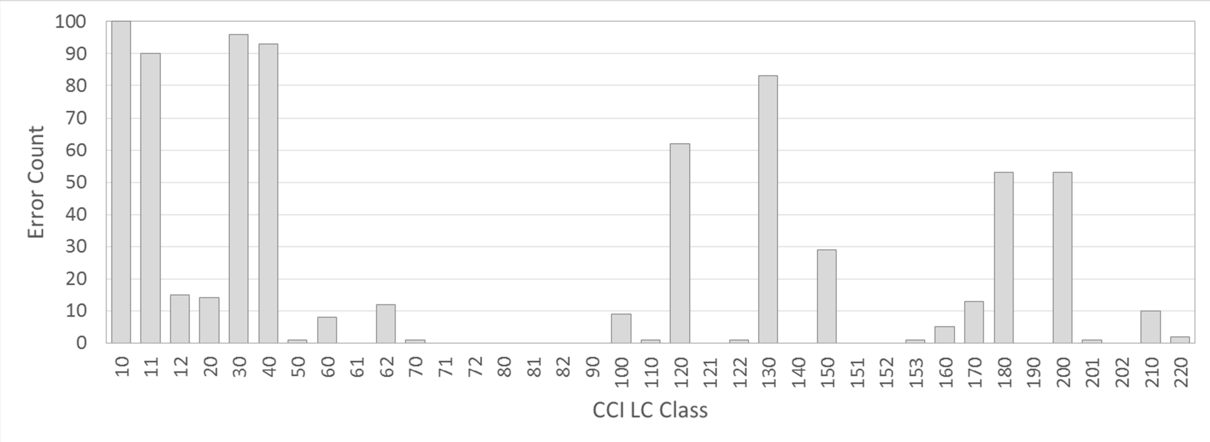

Figure 5-5: Distribution of errors within CCI LC classes – count of error occurrence regardless of area.

5.5.2 Dataset quality and spatial distribution of errors

During the quality control procedure dataset quality was evaluated per each cell (Figure 5-7) and assigned nominal category (good, moderate and low). For each error detected within the cells labeled as of low quality, suitable correction method was applied. In case when wrong LC labels were identified, impacted area was marked by polygon and particular rules for reclassification of impacted pixels were applied within the region. In case of obvious wrong LC delineation caused mostly by application of not optimal methods in classification procedure or data source errors, areas of LC patches had to be delineated manually. Again, for these newly delineated areas, specific rules were applied to perform correction to the final product. Certain areas in the world are characteristic by occurrence of combined error typology and complex correction rules had to be applied. After error identification and correction, spatial distribution of cells covering the areas where errors occurred has been recognized. These cells were classified based on type of error and are displayed in Figure 5-8. One of the common type of error was overestimation of crop categories occurring in large areas of northern Eurasia and in southern part of South America. Similarly, cropland was often misclassified in selected mountain areas in Europe and Asia where high altitude grassland is present (Figure 5-6). Then, error type of large pixel blocks occurred frequently in mountain areas of South America mainly due to low availability of input image data in high resolution. Wrong labeling of LC categories was recognized in dry areas of southern Africa. Also, several cases of sharp transition between LC categories caused by errors in classification were identified in different parts of sub-Saharan Africa.

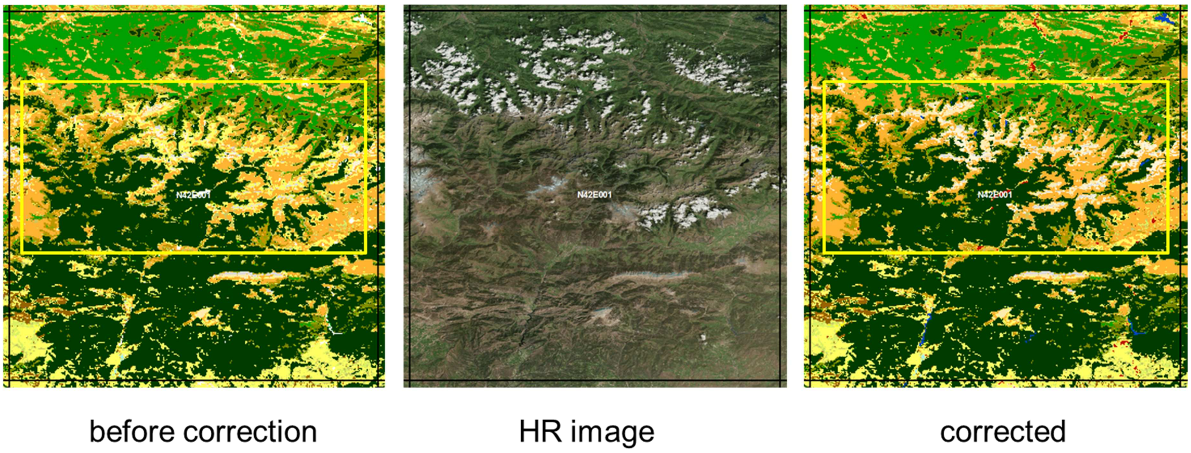

Figure 5-6: Example of error detected and corrected over Pyrenees, Europe.

In addition to identification and correction of errors in the world with low quality of LC data, areas of moderate data quality were selected. These areas are characteristic by difference in LC definition from different reference data sources, but have not clearly wrong delineation or labelling and in overall, are considered as correct. However, identification of these areas could serve to considerations in future development of LC product.

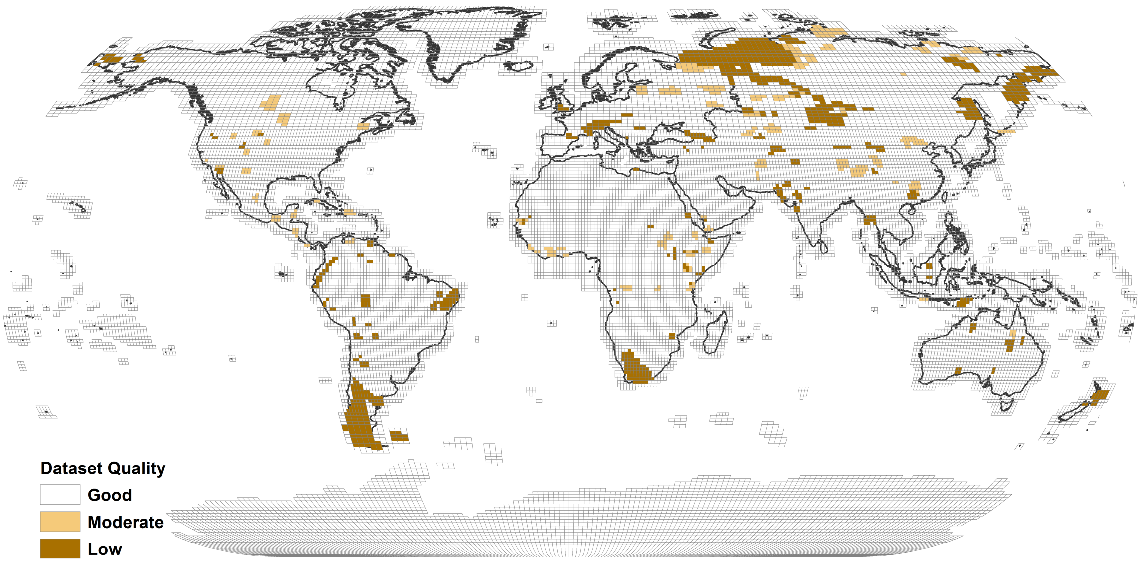

Figure 5-7: Map of global evaluation of LC dataset quality.

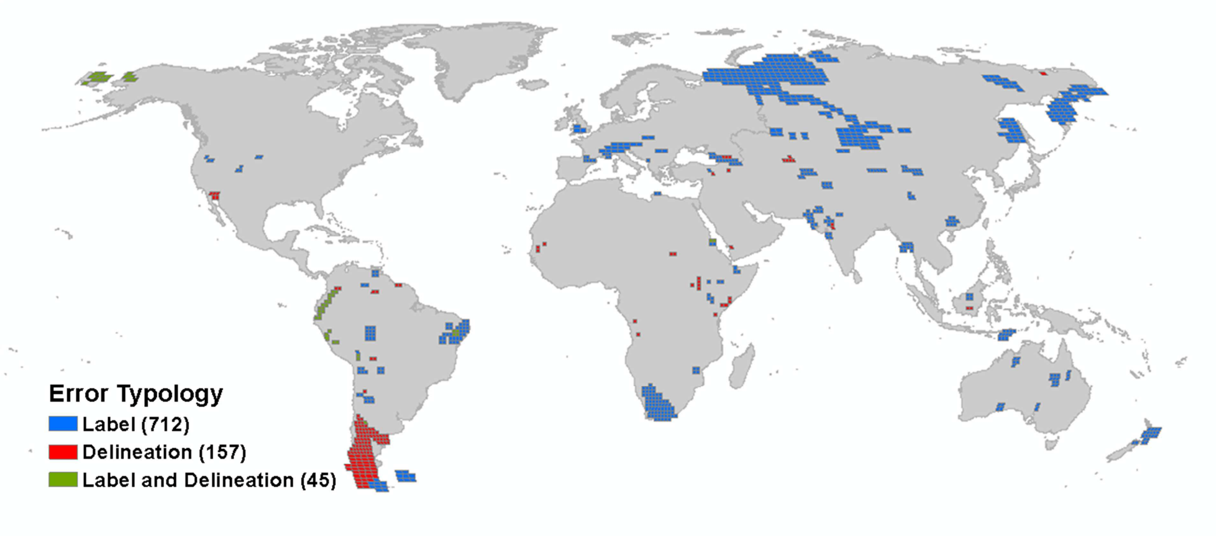

Figure 5-8: Map of error typology identified globally during systematic quality control.

In total, two assessment of quality were performed during the quality control of the global LC dataset. In the second run, additional errors were detected and corrected. However, further parameters were not assessed. Figure 5-9 displays spatial distribution of these errors.

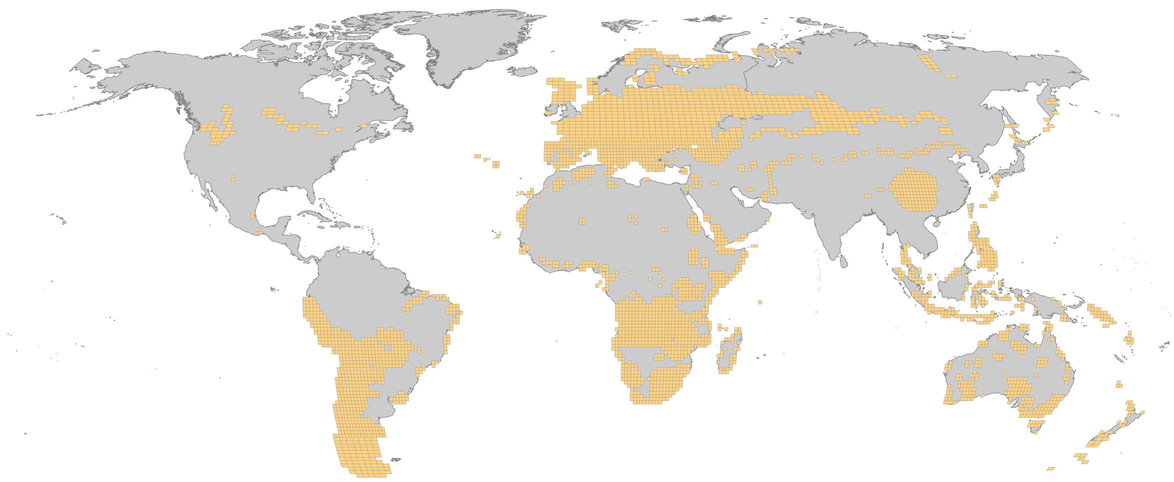

Figure 5-9: Map of errors identified globally during second quality assessment of the quality control procedure.

5.5.3 Landscape spatial pattern

During the quality control procedure, additional parameters concerning LC spatial pattern has been evaluated. Cells used for error detection were classified based on four types of landscape patters as displayed in Figure 5-10. The proportions of pattern types are stated in the legend of map. Light cell boundary color indicates the cells associated with detected errors.

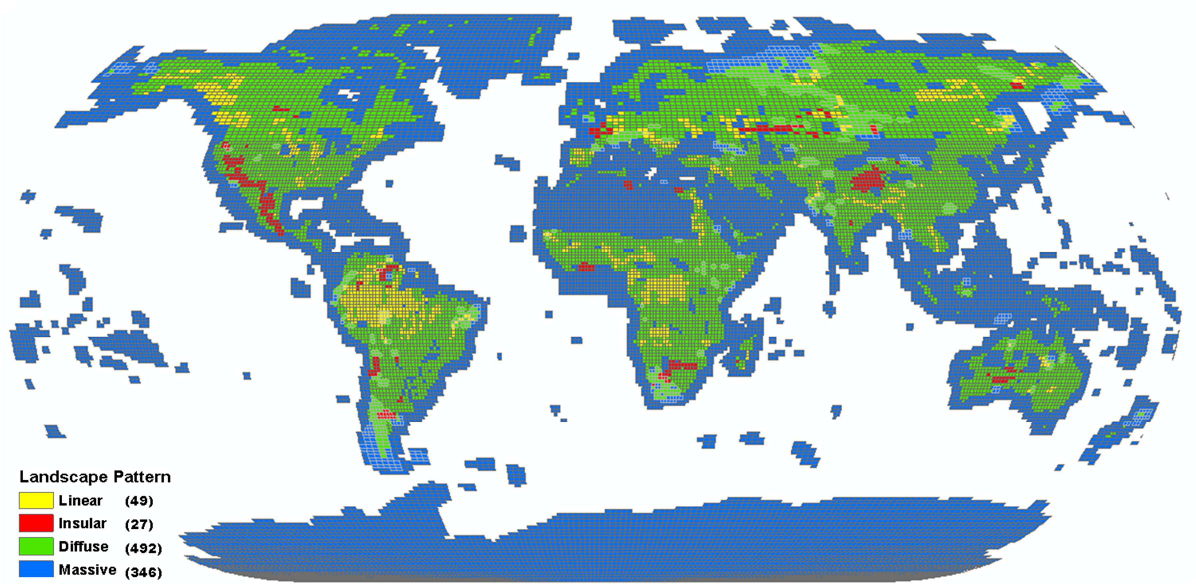

Figure 5-10: Map of landscape spatial pattern assessed globally during systematic quality control. Map is classified into four fragmentation patterns; legend refers to cell count per each pattern. Cells contained errors are highlighted by light cell boundaries.

At the end of the quality control procedure, following the data error specification and localization, all discovered errors were systematically corrected. In addition, following check list summarize all actions concerned quality control procedure:

- ✓ Systematic quality control of map products
- ✓ Check of the data documentation and format
- ✓ Check of data visualized through the viewer
- ✓ Check of the download functionality and the downloaded data (visualization)