Global Lakes Sentinel Services

Grant number 313256

GLaSS Training material, Lesson #2

Tools for GLaSS data analysis

Tools for statistical analysis of EO data

Ana Ruescas (BC)

Water Insight · SYKE · EOMAP · VU University Amsterdam · Consiglio Nazionale delle Ricerche · Brockmann Consult GmbH · Tartu Observatory space research centre · Brockmann Geomatics Sweden AB

Lesson summary

The aim of the lesson is to familiarize EO data users with the tools developed for the GLaSS project that are available in BEAM/SNAP. These tools are focused on image classification and statistical analysis of data. The training material is prepared for the understanding of the optical water tool (GLaSS Deliverable 3.3, 2014), and the image classification method (Magic Wand and Prediction tool, (GLaSS Deliverable 3.6, 2014) included currently in BEAM 5 and will be transferred into SNAP in the short term.

GLaSS training material outline

This lesson is part of the GLaSS training material. The complete training material outline is listed on http://data.waterinsight.nl/GLaSS/trainingmaterials/GLaSS_lesson_outline.pdf.

Note that the lessons logically follow up on each other, the later lessons might require skills that can be acquired during the earlier lessons.

GLaSS tools installation in BEAM 5

This training unit, as well as the other GLaSS training units -which deal with use cases as real examples of research and application for water quality on lakes- make use of one open source and free costs tool called BEAM.

BEAM stands for Basic ERS&ENVISAT (A)ATSR and MERIS TOOLBOX, and it is an open-source toolbox and development platform for viewing, analysing and processing of remote sensing raster data. Originally developed to facilitate the utilisation of image data from Envisat's optical instruments, BEAM now supports a growing number of other raster data formats such as GeoTIFF and NetCDF as well as data formats of other EO sensors such as MODIS, AVHRR, AVNIR, PRISM and CHRIS/Proba. Various data and algorithms are supported by dedicated extension plug-ins.

BEAM is installed easily in the three main systems Microsoft Windows, Unix and Mac OS. The tools developed in GLaSS will be available for the last version of BEAM 5 just by updating the plug-ins. Once BEAM has been installed, it is only necessary to access the *Module Manager* in the *Help* label and check for Available Modules or Module Updates to install the new plug-ins or to update them if there are new versions. For more help on this issue please visit: http://www.brockmann-consult.de/cms/web/beam/forum

Table of contents

List of abbreviations

Abbreviation	Description
BEAM	Basic ERS&ENVISAT (A)ATSR and MERIS TOOLBOX
CDOM	Colour Dissolved Organic Matter
C2R	Case 2 Regional Algorithm
EO	Earth Observation
ESA	European Space Agency
GPT	Graph Processing Tool
MERIS	Medium Resolution Imaging Spectrometer
OWT	Optical Water Types
ROI	Region Of Interest
SNAP	Sentinel Application Platform
VISAT	The desktop application used for EO data visualisation, analysing and processing.
WQ	Water Quality

1 Introduction

The BEAM Earth Observation Toolbox and Development Platform, is a collection of executable tools and Application Programming Interfaces (APIs) which have been developed to facilitate the utilization, viewing and processing of a variety of remotely sensed data.

The main components of the BEAM are:

- VISAT - An intuitive desktop application used for EO data visualization, analyzing and processing.
- A set of scientific data processors running either from the command-line or invoked by VISAT.
- The command-line tool gpt (graph processing tool) is used to execute processing graphs made up of operators' nodes developed using the BEAM GPF (see Java API below).
- A data product converter tool pconvert allows a user to convert raw data products to the BEAM-DIMAP standard, GeoTIFF, HDF-5 format or RGB images.
- A Java™ API which provides ready-to-use components for development of remote sensing related application and plug-in points for new BEAM extension modules. Besides a number of extension points such as product reader and writers, the BEAM API comprises the Graph Processing Framework (GPF), which is used to rapidly create raster data processors. The VISAT Rich Client Platform is used to develop rich GUI user interface applications based on BEAM VISAT.

In the framework of the GLaSS project, new tools for data mining and classification have been developed. Two of them will be publicly available in BEAM 5 and transferred into the SNAP in the near future:

1. An optical pre-classification algorithm that can distinguish the main water types and facilitate selection of water quality (WQ) algorithms was developed and implemented in BEAM, taking as basis, the fuzzy Optical Water Type processor (Moore et al., 2014).
2. A prediction tool focusing so far on the supervised learning and modelling was implemented in BEAM. The training data consist of a set of training examples (pixels, ROIs, etc.). In supervised learning, each example is a pair consisting of an input object and a desired output value (label). A supervised learning algorithm analyzes the training data and produces an inferred function ("Train model"), which can be used for mapping new examples ("Apply to images"). An optimal scenario will allow the algorithm correctly determine the class labels for pixels not in the set of training data.

2 This lesson

2.1 Research question

In this lesson two tools for the classification and the statistical analysis of EO data are presented: optical water types and image classification approaches, with semi-automatic pixel selection.

2.2 Training objectives / skills to gain in this lesson

The aim of the lesson is to familiarize EO data users with the tools developed within the GLaSS project and made available in BEAM/SNAP for data classification and statistical analysis of lakes.

2.3 Required software and data

Software and tools

To complete this lesson tasks, the following tools and software are required or suggested:

- BEAM (required, see Lesson #1), to be downloaded from: http://www.brockmann-consult.de/cms/web/beam/releases
- When the tools are transfer, these will also be available for the SNAP toolboxes.

Downloadable files

- GLaSS_Training_Lesson2.pdf
  The main document of the lesson including exercises and questions.
- GLaSS_Training_Lesson2_DataAndTools.zip
  Supplied data and tools, described below.

The zip-file with supplied data and tools contain:

- The subset of the MERIS image on the Lake Balaton:
  - subset_1_of_MER_FSG_1PNBCG20080706_092309_000000172070_00079_33199_0001.dim
  - subset_1_of_MER_FSG_1PNBCG20080706_092309_000000172070_00079_33199_0001.data

3 Hands-on exercises with BEAM

3.1 Optical Water Classification Tool

Introduction

The optical water classification (OWT) tool is an adaptation of a spectrally based optical pre-classification scheme (Moore et al., 2014) for global lakes that calculates clusters from in situ spectra collected by partners and advisors of the EU-GLaSS project. The tool assigns the water type class that corresponds to the best match between the remotely sensed and the in situ spectrum. Moore et al. (2001, 2014) designed a fuzzy logic spectral classification scheme that was adapted for lakes. In situ hyperspectral data were used to characterize optically distinct water classes *a priori.* The aggregated data come from multiple sources and covers a wide range of concentrations, also for colour dissolved organic matter (CDOM) and suspended sediments (SPM).

Tool overview

The classification method by Moore et al. (2014) had been implemented in BEAM including three new GLaSS classifications: a classification with five clusters (5C), a classification with 6 clusters (6C) and a classification in which the spectra are normalised to their total intensity before clustering: the 6C_normalised clustering. Tim Moore provided these three datasets to Brockmann Consult, who implemented the classification method with these new values in the OWT tool in BEAM 5, so that we have a processor tool for GLaSS partners to use embedded into BEAM.

In Figure 1 we can see the results of the GLASS_6C and the GLASS_6C_NORMALISED clusters. The GLASS_6C class 1 is representative of clear water slightly affected by chlorophyll pigments (peak in around band 550 nm). Chlorophyll dominated waters with increase in the pigment concentration are represented by class 1, class 3 and class 4. Classes 5 and 6 transition to bright sediment dominated waters. Class 2 waters are relatively dark in the whole spectrum and occur in peatlands with high humic (CDOM) absorption. The normalised classification shows different results because the reflectance values were normalised integrating the area under the curve. This removes magnitude effects and focuses more on the spectral shape.

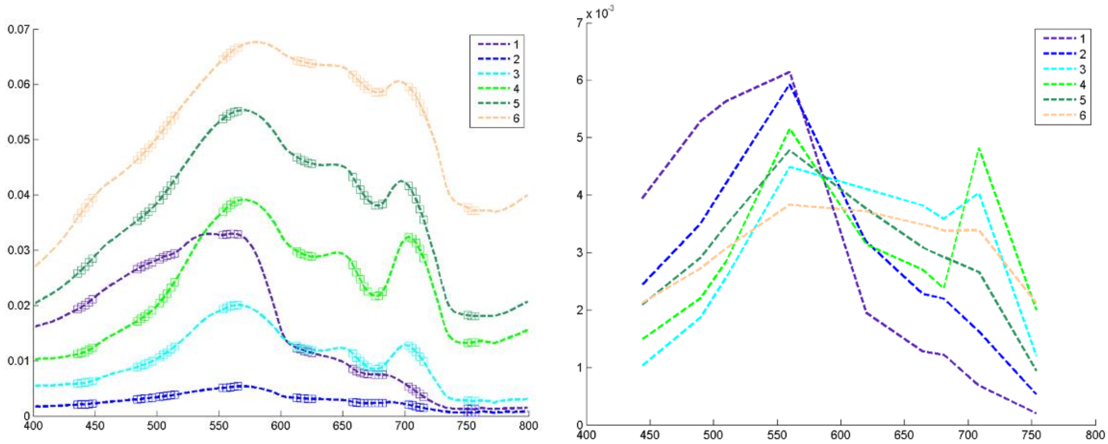


Figure 1. GLASS_6C and the GLASS_6C_NORMALISED clusters

The OWT can be found under VISAT's *Processing* tab, on the *Thematic Water Processing* sub-label. It is called *MERIS OWT classification* (Figure 2). The interface with the users consist on a window with the input and output parameters (source and target products), selection of directory, format of the output product and the option whether the derived product should be open in VISAT or not. The interface of *Processing Parameters* allows the user select the type of OWT and indicates the prefix used to identify the reflectance bands on the input product. In the example case the prefix is "reflec". Reflectance can be somewhat different when produced with different atmospheric correction schemes (e.g. standard MEGS, CoastColour, C2R, etc.). The user can opt for OWT input as RADIANCE_REFLECTANCES or IRRADIANCE_REFLECTANCES (ESA compatible).

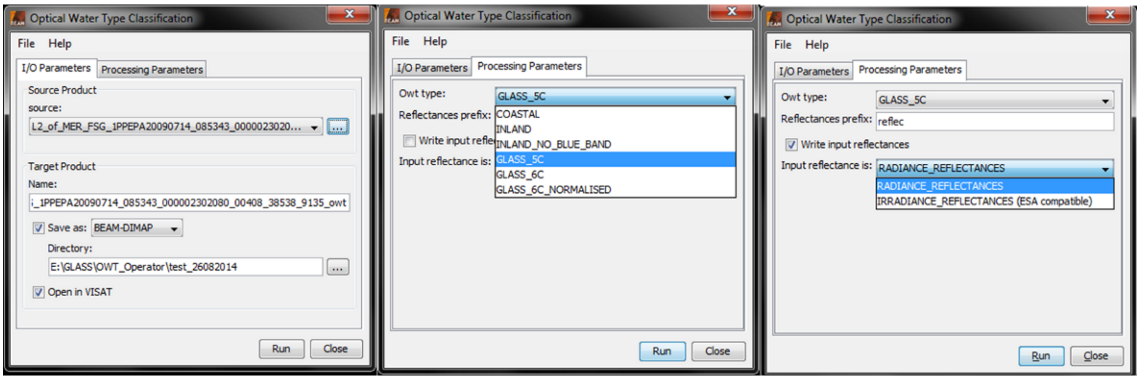


Figure 2. OWT tool overview

Remote sensing or water leaving radiance reflectance (Equation 1) is expressed in sr-1 units (theta_s is solar zenith angle).

Rrs(+)= Lw/(E0*cos(theta_s)) (Equation 1)

Water leaving (irradiance) reflectance is implemented in BEAM according to Equation 2 and is dimensionless.

rho_w(+)= RLw = Lw*PI//(E0*cos(theta_s)) (Equation 2)

Based on this input, the OWT Tool will convert RLw into Rrs, or not. Internal calculations will also convert the satellite signal from Rrs(+) to Rrs(-) (Equation 3), and perform normalisation, so that remote sensing data sets, and in situ classes were treated equally, and there is spectral matching on equal grounds.

Rrs (0,-)= Rrs(0,+)/(0.52+1.7*Rrs(0,+)) (in sr-1) (Equation 3)

The OWT processor expects to find the reflectances following a naming convention, which is <prefix>_i, where <prefix> is a user-defined text and "i" is the MERIS band number, starting with 1. For instance "reflec_2_*". The <prefix> can be specified here in this text field.

The output file consists of one band per each cluster, without and with normalisation, one band indicating the dominant class, the class sum and the normalised class sum band (Figure 3).

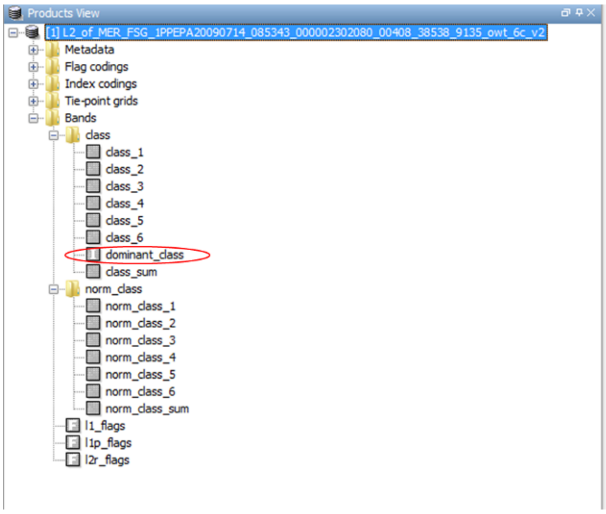


Figure 3. Output bands after OWT classification

Working with data

The image that we will use in this exercise is a subset of a MERIS Full Resolution image taken on Lake Balaton the 6 of July of 2008 (Figure 4). The exercise use one image called:

”subset_1_of_MER_FSG_1PNBCG20080706_092309_000000172070_00079_33199_0001.dim”

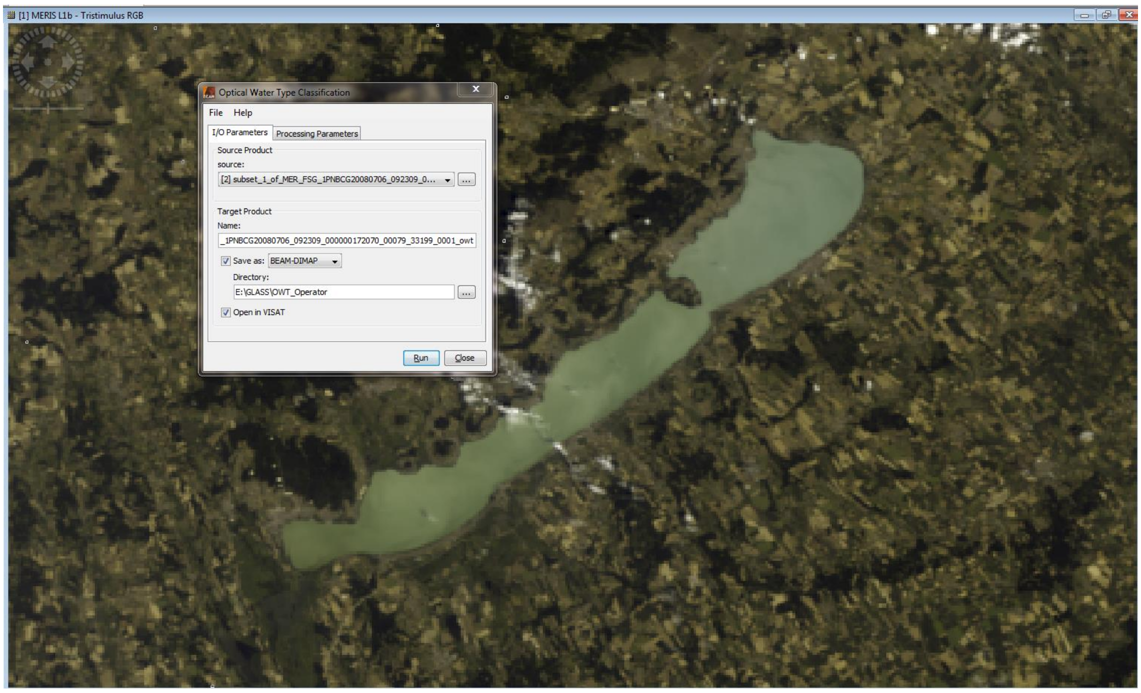


Figure 4. RBB composition of MERIS FSG, Lake Balaton image on 6 July 2008

For running the OWT tool, a Level 2 image is necessary, because the classification processor was trained with in situ reflectance data. BEAM has several L2 Processors. For this exercise we will use a Coast Colour Atmospheric Correction processor to derive this L2 data.

Go to the *Processing* label and select in the *Thematic Water Processing* section the *MERIS CoastColour Processors/CoastColour Atmospheric Correction* (Figure 5).

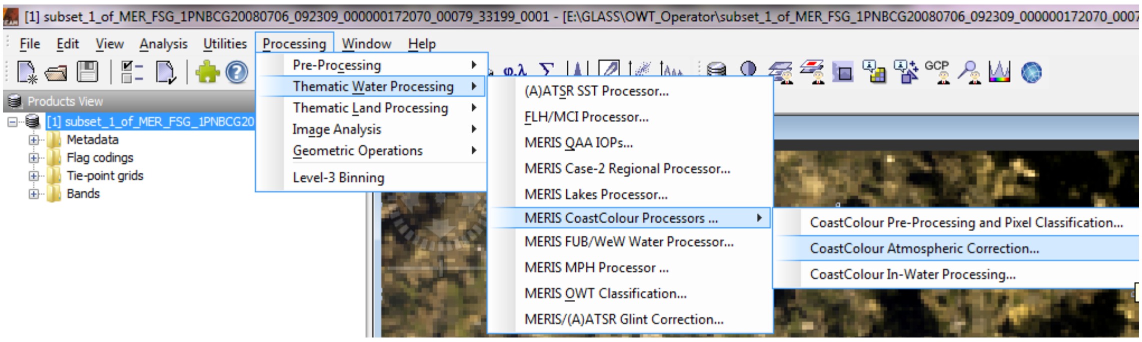


Figure 5. Selecting the CoastColour Atmospheric Correction processor

The CoastColour Atmospheric Correction processor uses the L1 image and applies an atmospheric correction to derive surface reflectance from the top of atmosphere (TOA) radiances of the L1. In the I/O parameters dialog box, select the L1 image and a name for the new image that will be generated (Figure 6). In the example shown in Figure 6, the name of the output image has been modified to change the number 1 after the *_FSG_1PNB* prefix by a 2 *_FSG_2PNB* to indicate a L2 image will be generated.

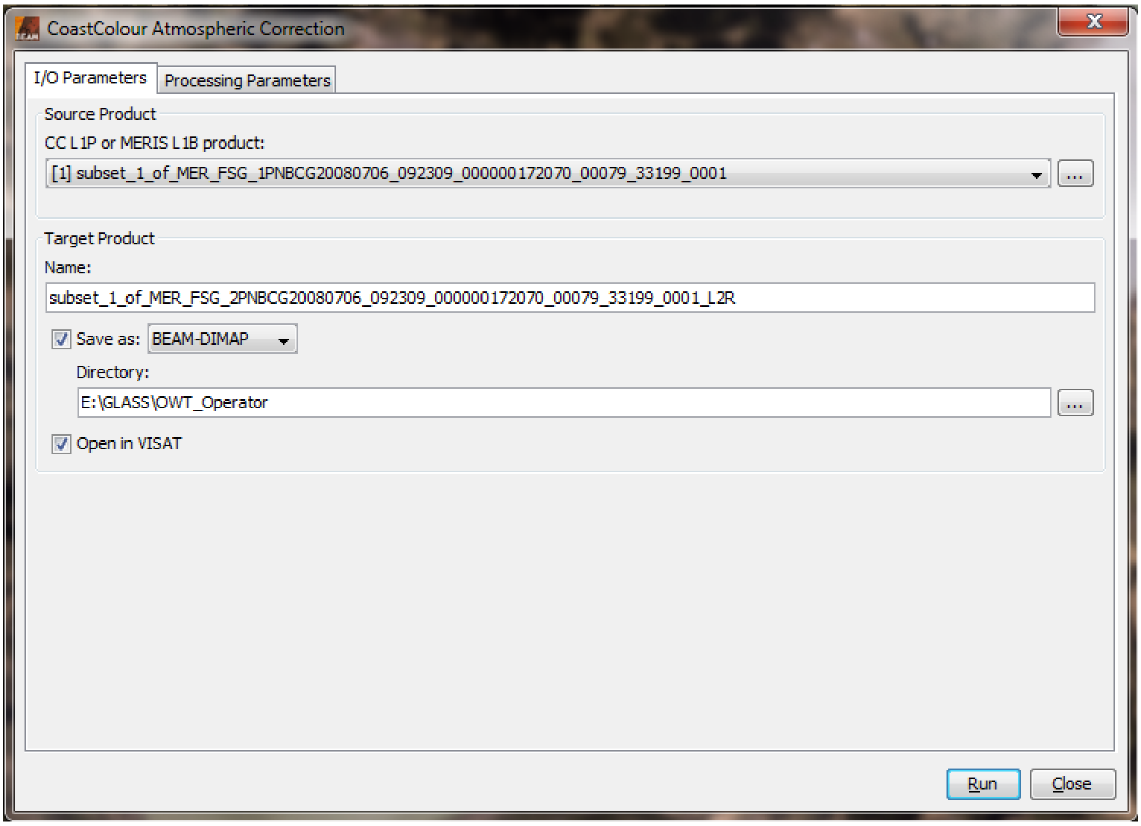


Figure 6. CoastColour Atmospheric Correction processor I/O parameters

The Processing Parameters shows all possible options (Figure 7). For the present exercise we will leave the options to default. There is only one detail we need to check now and that is the last line in the dialog box. When saving the results, radiance reflectance or irradiance reflectance can be chosen. Select one of the two and keep in mind your selection for the next steps. The explanation about the difference can be found in the Tool Overview section.

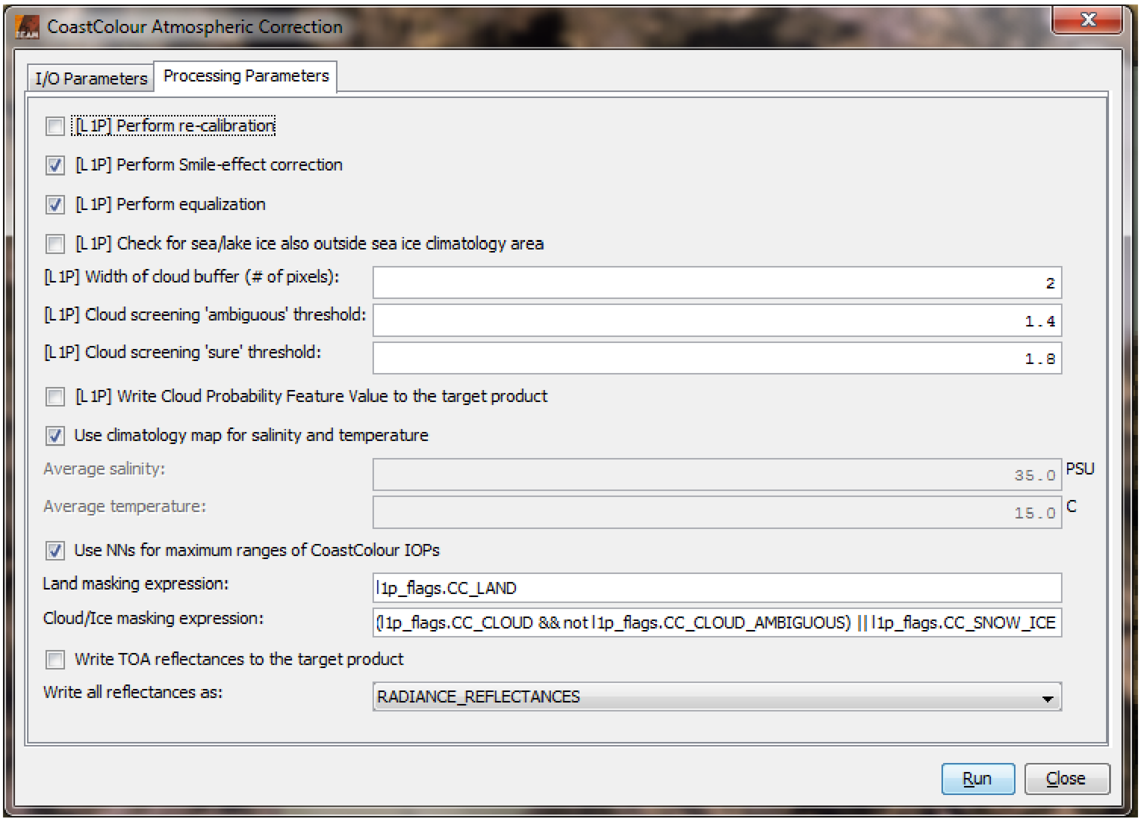


Figure 7. CoastColour Atmospheric Correction Processing Parameters window

To open the OWT tool open BEAM 5.0-VISAT's *Processing* tab, and select the *Thematic Water Processing* sub-label. It is called *MERIS OWT classification* (Figure 8).

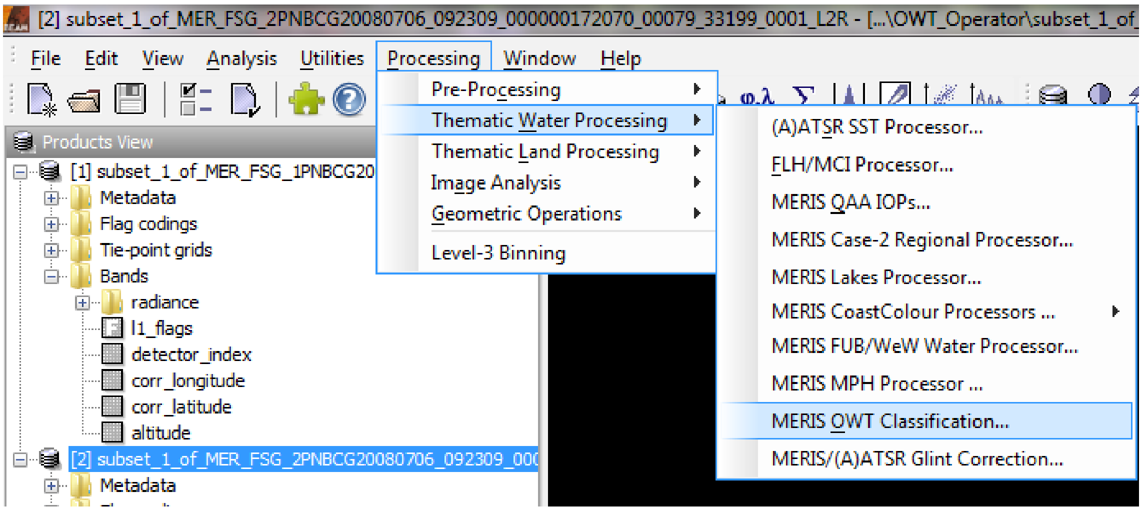


Figure 8. Opening the MERIS OWT classification tool in BEAM 5.0

Figure 9 shows how to select the source product. It is also possible to open the image previously in the *Product View* and it will be listed automatically in the source product. The target product name is the same as the source name with the prefix "_owt" by default. This name can be change by the user by clicking twice in the "Name" dialog window. The output format can also be selected (HDF, NETCDF4, DIMAP, CSV or GeoTIFF), and the directory where the output data should be stored.

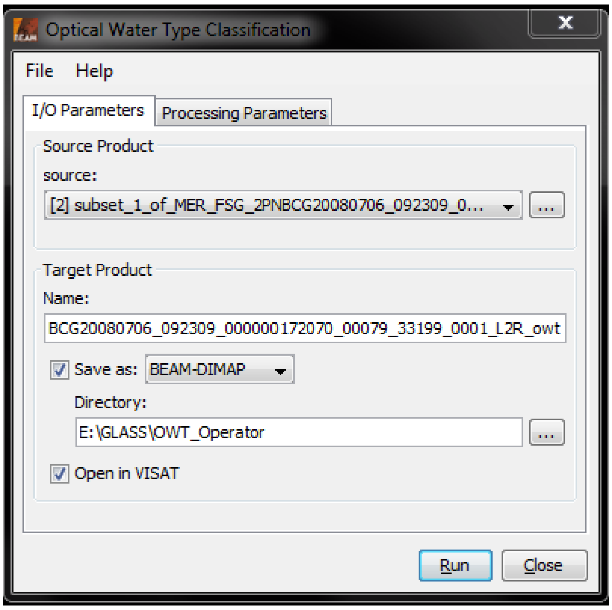


Figure 9. Optical water type classification I/O parameters

Figure 10 shows the processing parameters label of the tool. There are several OWT classification types developed for GLaSS and other projects (CoastColour and Diversity II). The three options developed and implemented for GLaSS are the GLASS_5C, GLASS_6C, GLASS_6C_NORMALISED.

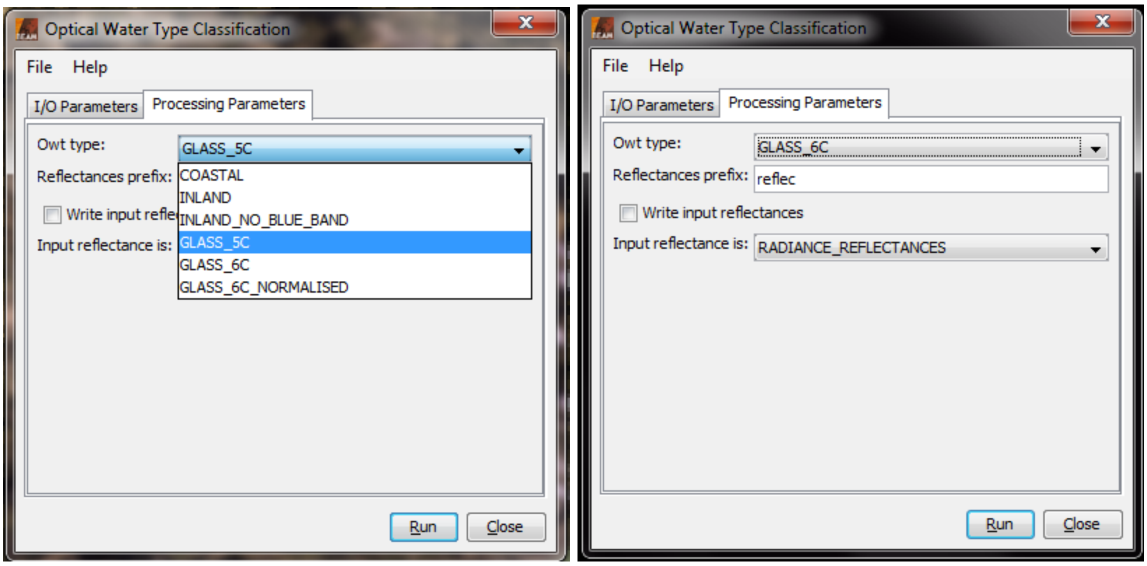


Figure 10. Processing parameters label of the OWT tool

The reflectance prefix dialog window has been described in the *Tool Overview* section. The "Input reflectance" dialog refers to the type of units in which the reflectances are written. Here the user should recall in which format the reflectance bands were saved in the previous step, and make the same selection here. There is always the possibility of opening the metadata of the image and look for the out reflectance units (Figure 11).

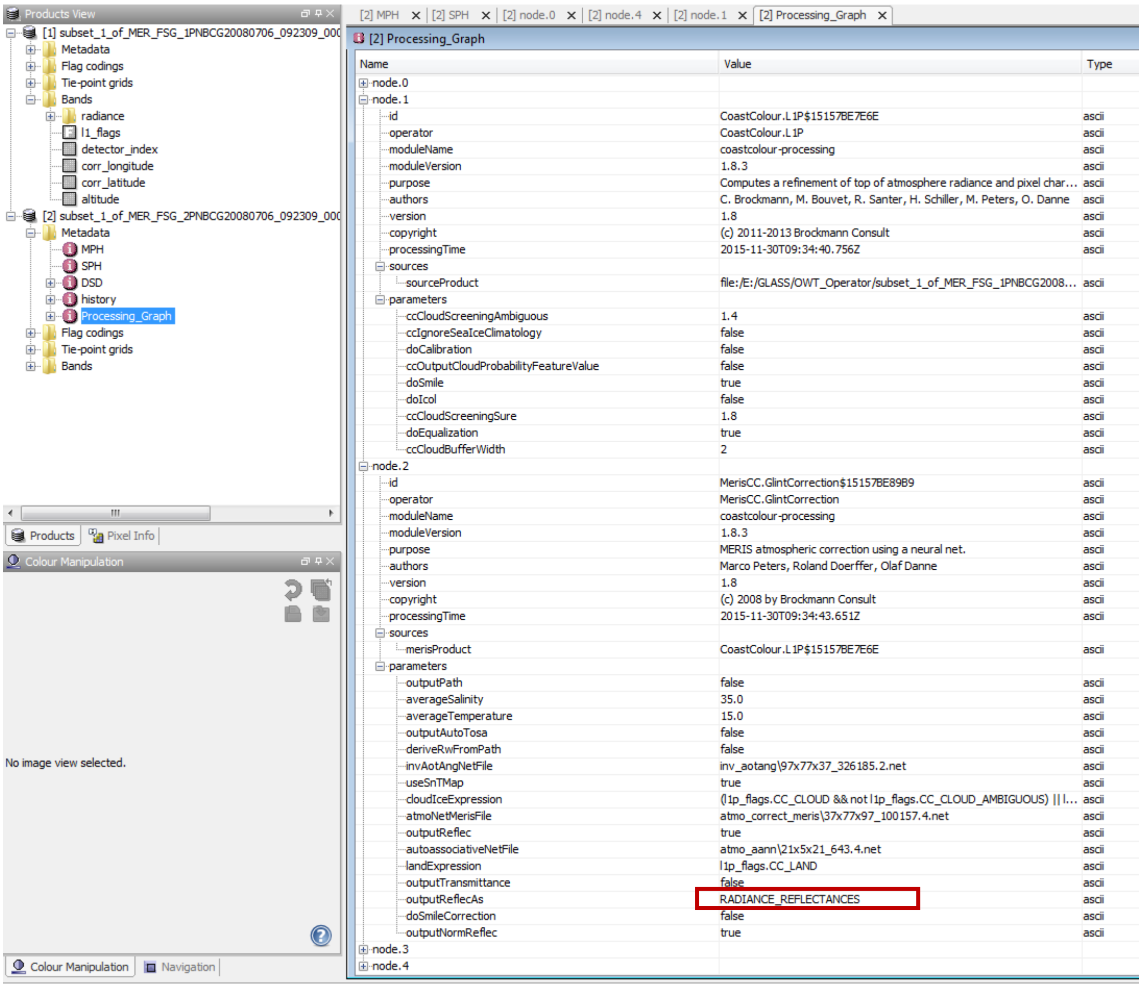


Figure 11. Metadata visualisation in BEAM 5.0

This is a very important step because the OWT tool makes internal calculations depending on the type of the input unit, as we have seen before.

Now the OWT type can be selected, for instance the GLASS_6C, specify the reflectance prefix ("reflec") and the input reflectance. After a few seconds the image will be processed (Figure 12).

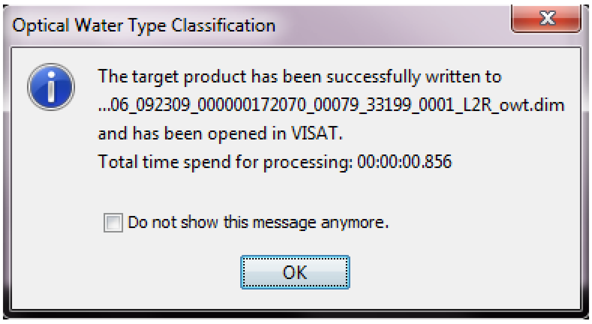


Figure 12. OWT total time spent for processing message

The user can click now on the output file in the *Products View* to see the content of the OWT file (Figure 13).

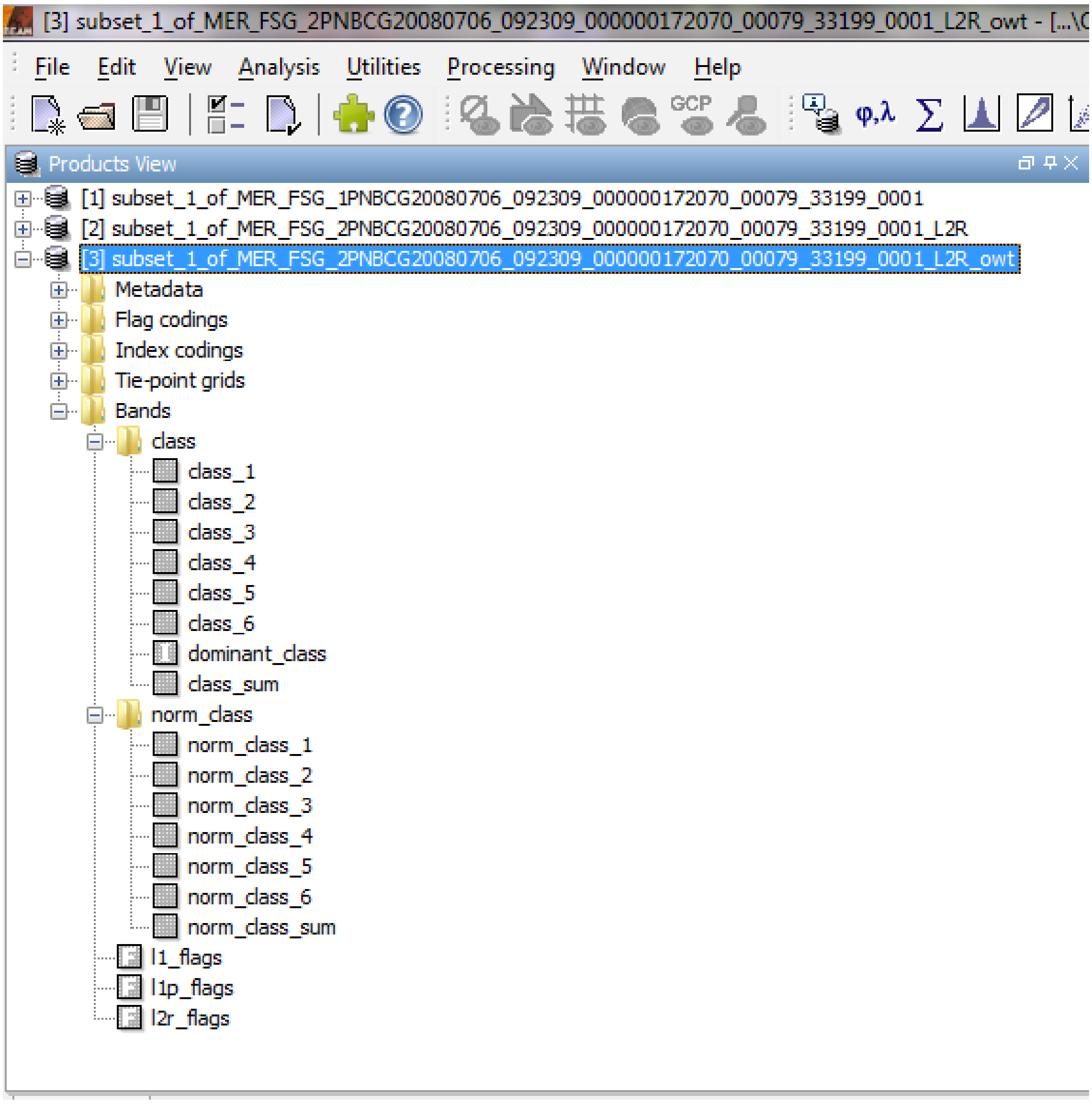


Figure 13. Content of OWT output file

Visualisation of the results

The output file consists of one band per each cluster, without and with normalisation, one band indicating the dominant class, the class sum and the normalised class sum (Figure 13). All pixels with reflectance data are evaluated and a probability of pertaining to the different clusters is calculated. These are the membership maps (Figure 14). In some cases it is possible that some pixels do not have membership in any of the OWTs. These are assigned to NaN and depict in black (Figure 14 and Figure 15).

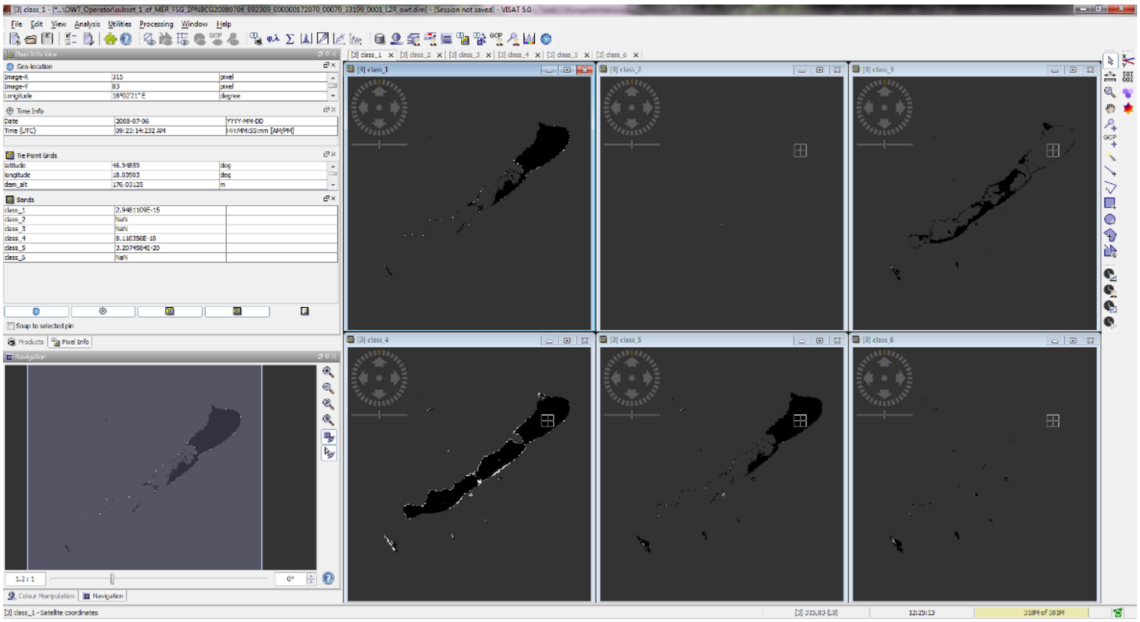
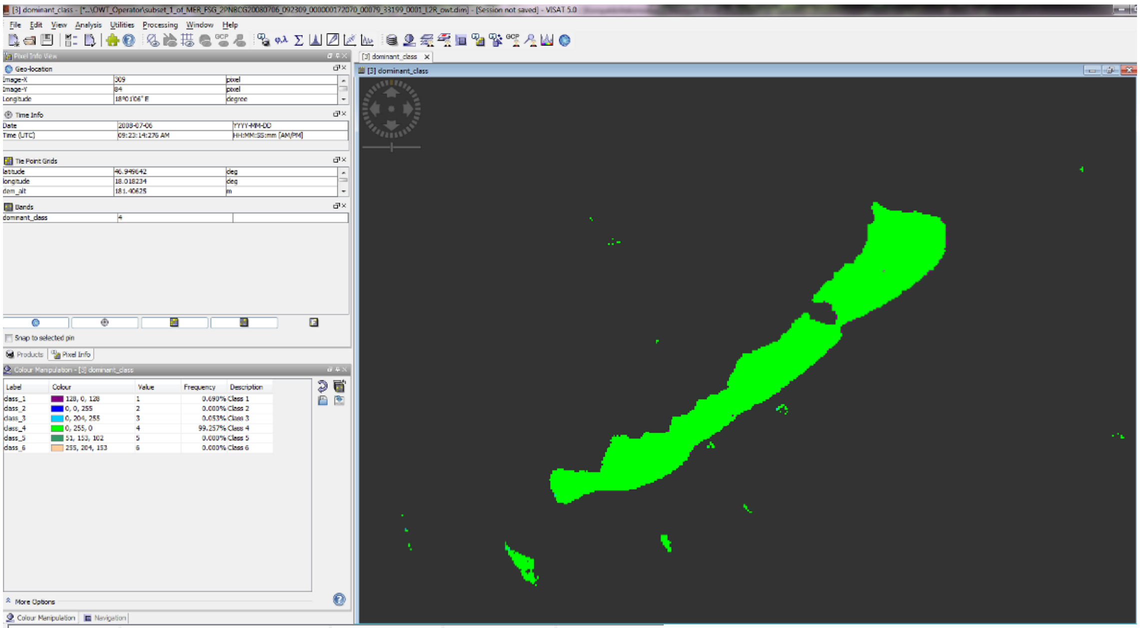


Figure 14. Overview of OWT GLASS_6C results on the Lake Balaton image

Bands	
class_1	2.9481109E-15
class_2	NaN
class_3	NaN
class_4	8.110356E-10
class_5	3.2074584E-20
class_6	NaN

Figure 15. Detail on pixel information of the probability maps

The dominant class band shows which of the OWT classes are dominant for each pixel, that means, which one has the maximum membership probability (Figure 16). In the image used here, it is clear that the majority of the pixels for this image belong to class 4. If we refer to Figure 1, we can observe that class 4 corresponds to a water type that contains relatively high values of chlorophyll, probably due to an algal bloom during this July scene (Figure 17). Citing Palmer et al. (2015) "*Annual phytoplankton blooms are particularly severe during the late summer and early fall months (August/September), especially in Basin 1 but commonly extending to Basins 2 and 3 and occasionally to Basin 4, and smaller winter/spring blooms also commonly occur (Hajnal and Padisák, 2008, Mózes et al., 2006 and Présing et al., 2008).*" (Figure 18).



Label	Colour	Value	Frequency	Description
class_1	128, 0, 128	1	0.690%	Class 1
class_2	0, 0, 255	2	0.000%	Class 2
class_3	0, 204, 255	3	0.053%	Class 3
class_4	0, 255, 0	4	99.257%	Class 4
class_5	51, 153, 102	5	0.000%	Class 5
class_6	255, 204, 153	6	0.000%	Class 6

Figure 16. Dominant class image for lake Balaton and detail of color scale bar and legend

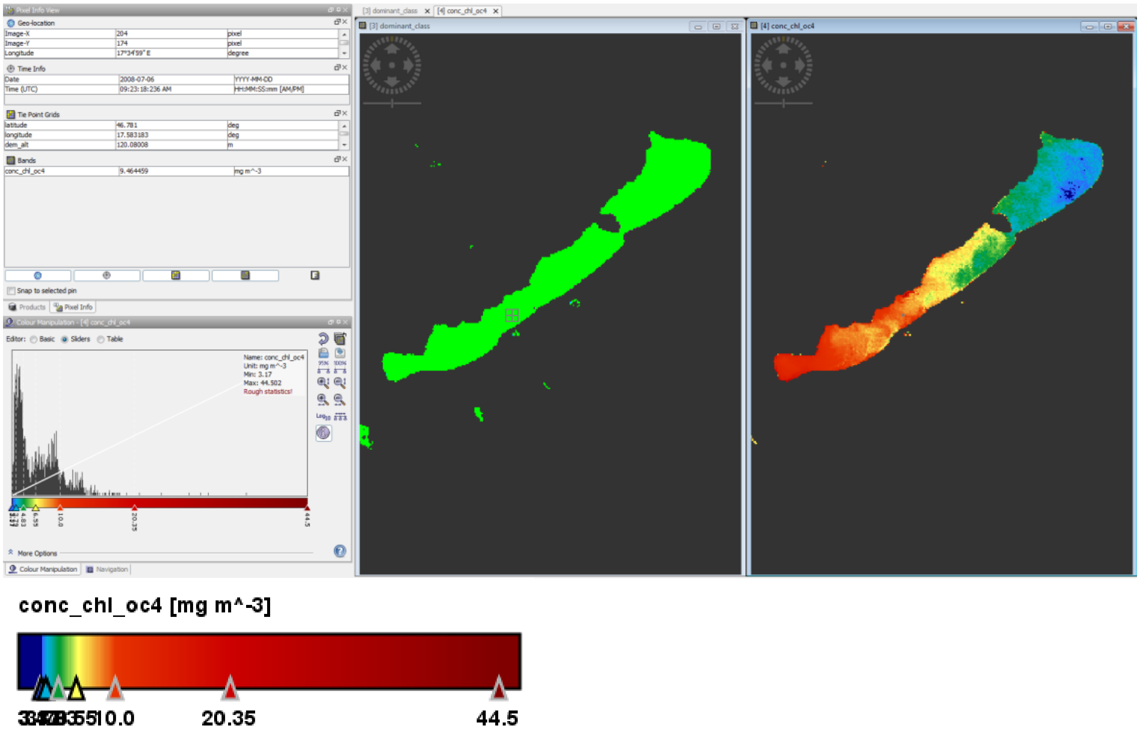


Figure 17. Dominant class versus an estimate of chlorophyll concentration (oc4) in mg m^{-3}

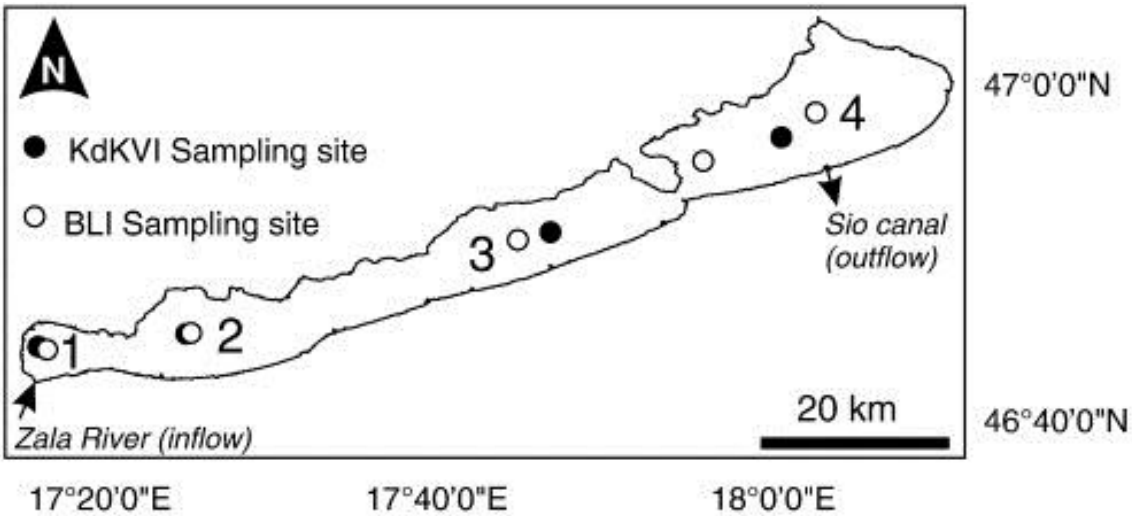
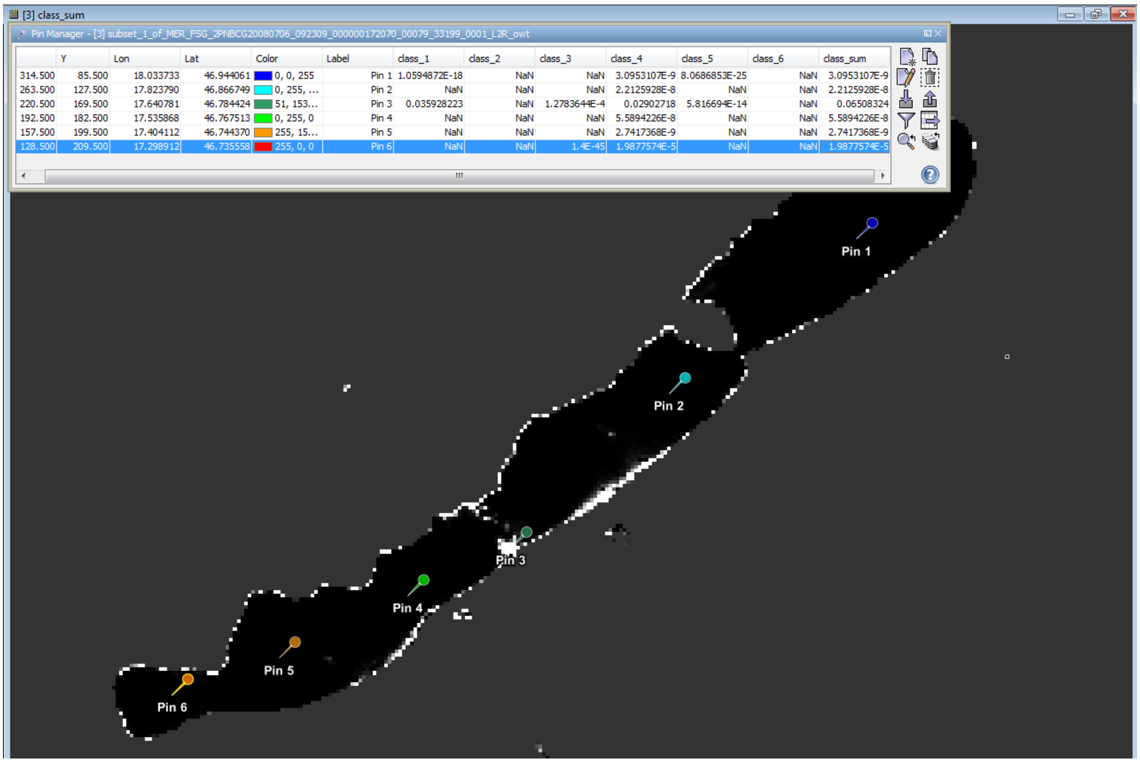


Figure 18. Lake Balaton, Hungary. The four main basins (1–4), regularly sampled sites, the Zala River (main inflow) and the Sio canal (main outflow) are indicated (from Palmer et al , 2015)

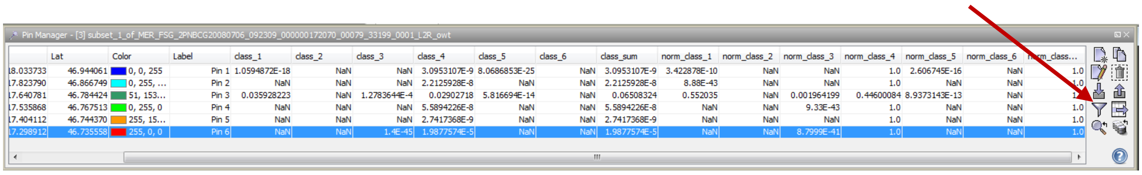
The class sum band is the addition of all class memberships. In Figure 19 a few pins have been selected on the different parts of the lake. The class sum numbers can be seen in the "Pin manager" table as the sum of the individual class probabilities.



	Y	Lon	Lat	Color	Label	class_1	class_2	class_3	class_4	class_5	class_6	class_sum
314.500	85.500	18.033733	46.944061	0, 0, 255	Pin 1	1.0594872E-18	NaN	NaN	3.0953107E-9	8.0686853E-25	NaN	3.0953107E-9
263.500	127.500	17.823790	46.866749	0, 255, ...	Pin 2	NaN	NaN	NaN	2.2125928E-8	NaN	NaN	2.2125928E-8
220.500	169.500	17.640781	46.784424	51, 153...	Pin 3	0.035928223	NaN	1.2783644E-4	0.02902718	5.816694E-14	NaN	0.06508324
192.500	182.500	17.535868	46.767513	0, 255, 0	Pin 4	NaN	NaN	NaN	5.5894226E-8	NaN	NaN	5.5894226E-8
157.500	199.500	17.404112	46.744370	255, 15...	Pin 5	NaN	NaN	NaN	2.7417368E-9	NaN	NaN	2.7417368E-9
128.500	209.500	17.298912	46.735558	255, 0, 0	Pin 6	NaN	NaN	1.4E-45	1.9877574E-5	NaN	NaN	1.9877574E-5

Figure 19. Visualisation of the values of 6 pins by class and class_sum

The normalisation used is the simple division of the class by the class sum; therefore the norm_class_sum always should be 1 when the pixel has been classified in any of the clusters. In Figure 1 the normalised class have been added to the *Pin manger* table by clicking on the filter icon 



Pin Manager - [3] subset_1_of_MER_FSG_2PNBCG20080706_092309_000000172070_00079_33199_0001_L2R_owt

	Lat	Color	Label	class_1	class_2	class_3	class_4	class_5	class_6	class_sum	norm_class_1	norm_class_2	norm_class_3	norm_class_4	norm_class_5	norm_class_6	norm_class...
18.033733	46.944061	0, 0, 255	Pin 1	1.0594872E-18	NaN	NaN	3.0953107E-9	8.0686853E-25	NaN	3.0953107E-9	3.422878E-10	NaN	NaN	1.0	2.606745E-16	NaN	1.0
17.823790	46.866749	0, 255, ...	Pin 2	NaN	NaN	NaN	2.2125928E-8	NaN	NaN	2.2125928E-8	8.88E-43	NaN	NaN	1.0	NaN	NaN	1.0
17.640781	46.784424	51, 153...	Pin 3	0.035928223	NaN	1.2783644E-4	0.02902718	5.816694E-14	NaN	0.06508324	0.552035	NaN	0.001964199	0.44600084	8.9373143E-13	NaN	1.
17.535868	46.767513	0, 255, 0	Pin 4	NaN	NaN	NaN	5.5894226E-8	NaN	NaN	5.5894226E-8	NaN	NaN	9.33E-43	1.0	NaN	NaN	1.0
17.404112	46.744370	255, 15...	Pin 5	NaN	NaN	NaN	2.7417368E-9	NaN	NaN	2.7417368E-9	NaN	NaN	NaN	1.0	NaN	NaN	1.0
17.298912	46.735558	255, 0, 0	Pin 6	NaN	NaN	1.4E-45	1.9877574E-5	NaN	NaN	1.9877574E-5	NaN	NaN	8.7999E-41	1.0	NaN	NaN	1.0

Figure 20. Pin manager showing pin values for class and norm_class

The number of classes depends highly on the input in situ data used for the generation of the LUT tables (see Moore et al. 2001); and it is possible and even likely, that the data collected do not represent all conditions

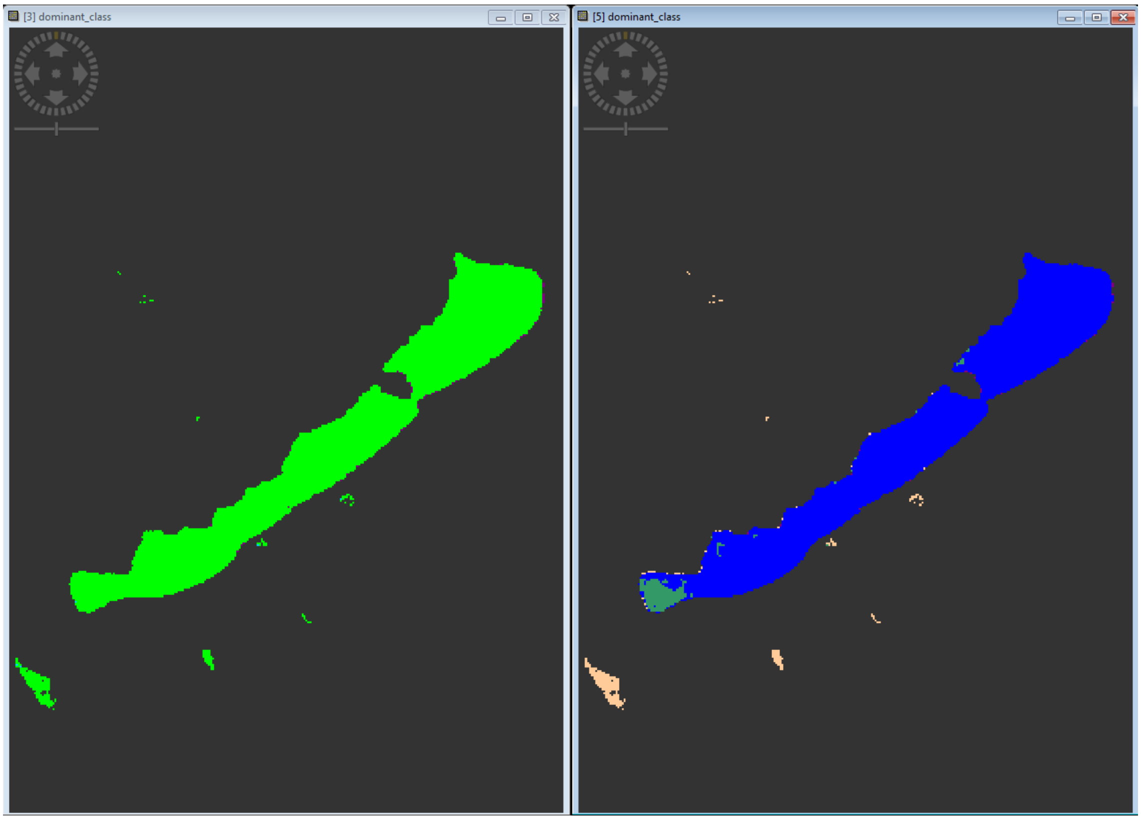


Figure 21. Dominant class resulted from the GLASS_6C (left) and the GLASS_6C_NORMALISED (right) classifications

We can also run the OWT using the GLASS_6C_NORMALISED and compare the results (Figure 21). The dominant class is now class 2, whose properties are now more based on the shape of the spectra than on the reflectance magnitude, indicated again that in this image we have chlorophyll dominated waters in practically all the basin except in the area 1 (Figure 18), where the inflow of the Zala River is present and increases the sediment content of these waters.

3.2 Prediction tool

Introduction

Related to the cluster analysis, one of the main purposes of satellite remote sensing is to interpret the observed data and classify features. In addition to the approach of photointerpretation, quantitative analysis, which uses computer to label each pixel to particular spectral classes (called classification), is commonly used. There are two broads of classification procedures: supervised classification and unsupervised classification. In the supervised classification the analyst has available sufficient known pixels to generate representative parameters for each class of interest. This step is called training. Once trained, the classifier is then used to attach labels to all the image pixels according to the trained parameters. The most commonly used supervised classification is maximum likelihood classification (MLC), which assumes that each spectral class can be described by a multivariate normal distribution (Figure 22).

The training data consist of a set of training examples (pixels, ROIs, etc.). In supervised learning, each example is a pair consisting of an input object and a desired output value (label). A supervised learning algorithm analyses the training data and produces an inferred function (train model), which can be used for mapping new examples (apply to image). An optimal scenario will allow for the algorithm to correctly determine the class labels for pixels not in the training data set. This requires the learning algorithm to generalize from the training data to unseen situations in a reasonable way.

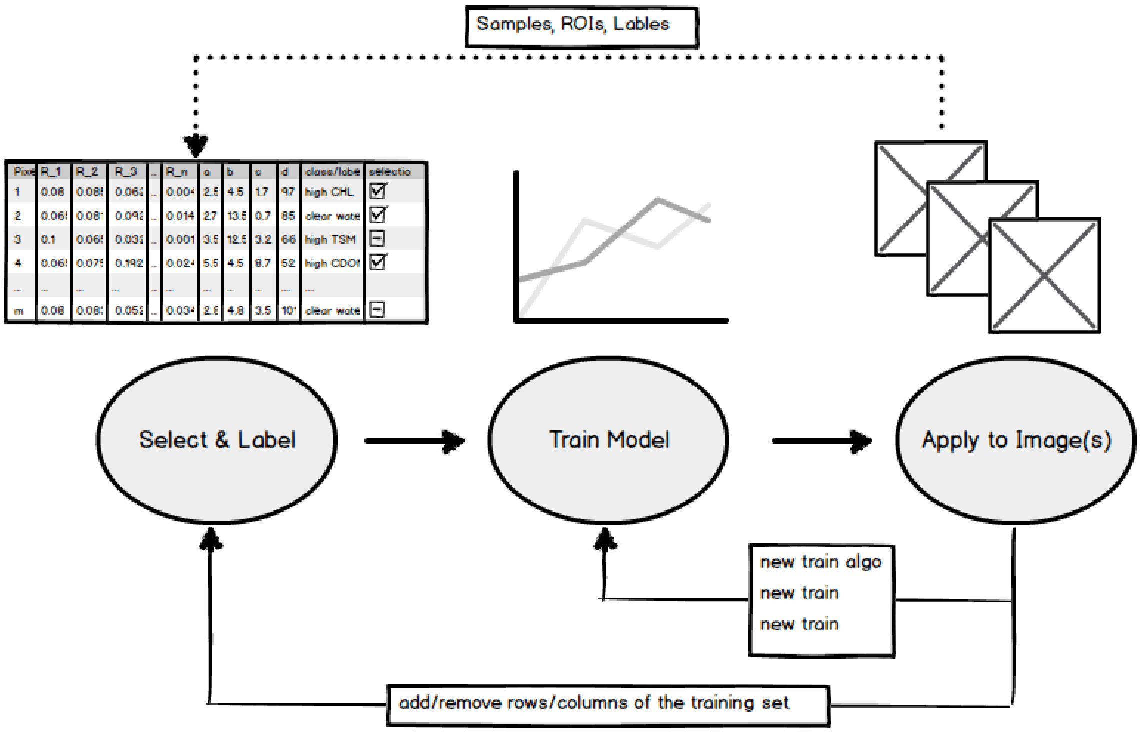


Figure 22. Concept for the supervised classification in BEAM

The installation of the *Prediction Tool* in BEAM is done via the *Module Manager*, which is stared under the *Help* menu. In the tab *Available Modules,* the two entries “Data Model Application Operator” and “Data Modeling” need to be selected and the *Install bottom* needs to be clicked and finally with *OK* the processed started. The Prediction Tool will be installed next time that VISAT is opened.

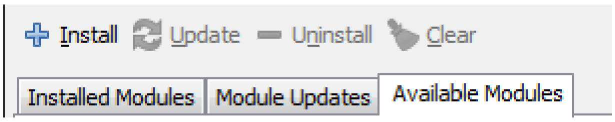


Figure 23: Zoom of the Module Manager for installing the prediction tool within BEAM.

The exercise use one image called:

”subset_1_of_MER_FSG_1PNBCG20080706_092309_000000172070_00079_33199_0001.dim”

This is a MERIS FSG subset of the Lake Balaton image for the 6 of July of 2008. To read the image in BEAM-VISAT choose the *File/Open Product* option. This image will be the source of all input information needed for the training of the model, and it will be later used when the model is applied by the model operator.

Selection and labelling

The first step is to determine the type of training examples. The user needs to decide what kind of data is to be used as a training set. In the case of remote sensing, the information is extracted from a pixel or a group of pixels or region of interest. The second step will be to gather the training set. The training set has to be representative enough of the set of data used as input. We will select the pixels for the training using some of the BEAM tools already available.

The training set must be representative of the target classes of interest. The aim of the training is to derive a representative sample of the spectral signatures or other parameters of interest of each class. Thus, the quality of the training datasets is directly influencing the performance of the algorithm and so the results. The training data is usually derived from an image using *a priori* knowledge of the scene. Several spatial sampling objects can be used to select the training data: a single pixel, polygons or blocks of pixels, similar contiguous pixels, pixels following certain arithmetic expressions, etc.

In the prediction tool implemented in BEAM, the training data is introduced in the model by creating masks (Figure 24). The image to be classified has to be selected previously and the mask manager should be opened. Training pixels are saved into masks using different tools for their acquisition. A mask is a product node similar to a band or tie-point grid. It has a unique name and comprises an image (raster data) whose sample data type is boolean. Each data product may comprise virtually any number of masks.

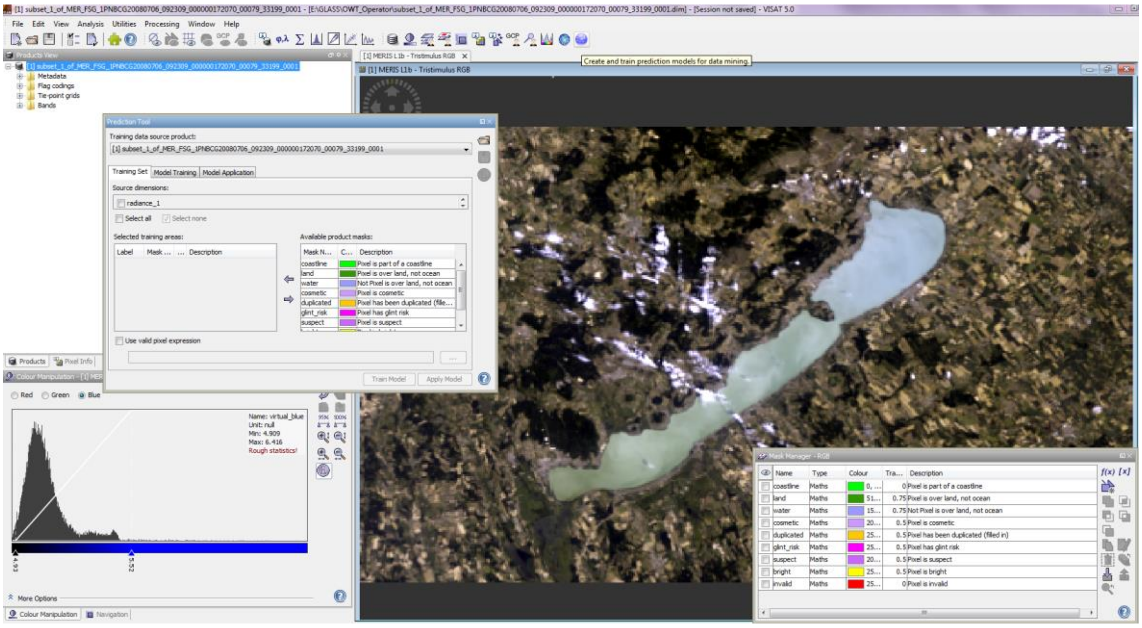


Figure 24. Overview of the Prediction Tool training set and mask manager

Training areas

There are several ways of selecting pixels using the mask creation tool or others (pins, geometries, magic wand selection). The common thing to all of them is that they have to be stored as mask of the product. This creation of mask and storage is made automatically in BEAM when using the *Mask Manage*r, the pins, geometry tools or the magic wand. It is also possible to import vector data or even create a mask for tool windows.

The exercise will be made with the image called: “subset_1_of_MER_FSG_1PNBCG20080706_092309_000000172070_00079_33199_0001.dim”

This is a MERIS FSG subset of the Lake Balaton image for the 6 of July of 2008. Open one band or make and RGB. To open the mask manager go to *View/Tool Windows/Mask manager.*

The Mask Manager

Masks are organized by the “*Mask Manager*” tool (Figure 25). The manager allows creating new masks, editing mask properties and delete existing masks. It also allows for creating new masks based on logical combinations of existing masks. Furthermore masks may be imported and exported. If an *Image View* is selected, the manager tool window can also be used to control the visibility for the currently displayed band. Each mask becomes automatically available in the raster data analysis tools, such as the Statistics, Histogram, and Scatter Plot tool windows.

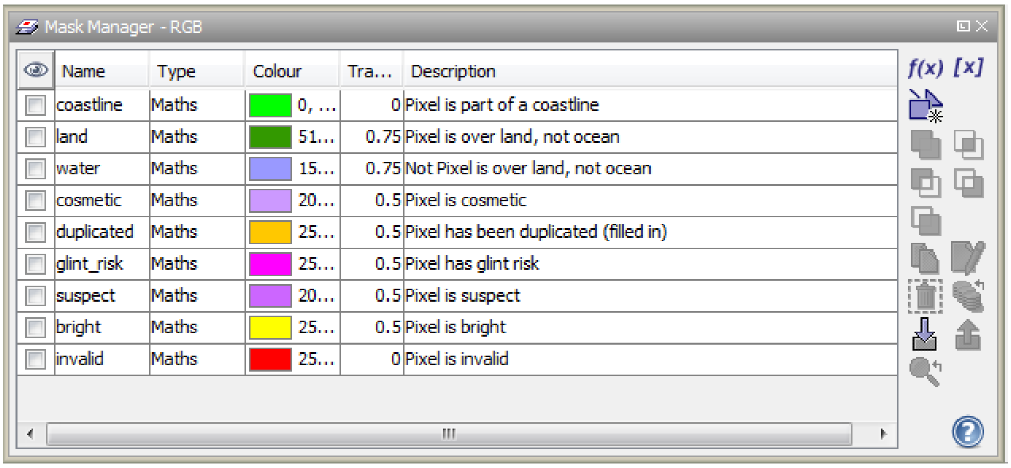


Figure 25. The mask manager

The Pin Manager

Click on the icon  to open the *Pin Manager*. A pin is a marker for a certain geographical position within a geo-referenced image and can be used to store the pixel information. New pins can be created with the Pin Tool  and removed using the delete pin command in the Edit menu or using directly the Pin Manager. The Pin Manager displays all pins stored in the current product within a table and provides some pin related operations. Pins are automatically stored in the *Mask Manager* as a mask derived from a geometry (a point or groups of points).

We can place some of the pins in the image (Figure 26): select the Pin Tool and place two pins in the lake, rename them using the Pin Manager and look for them in the Mask Manager. To change the color of the pins is also possible within the Pin Manager.

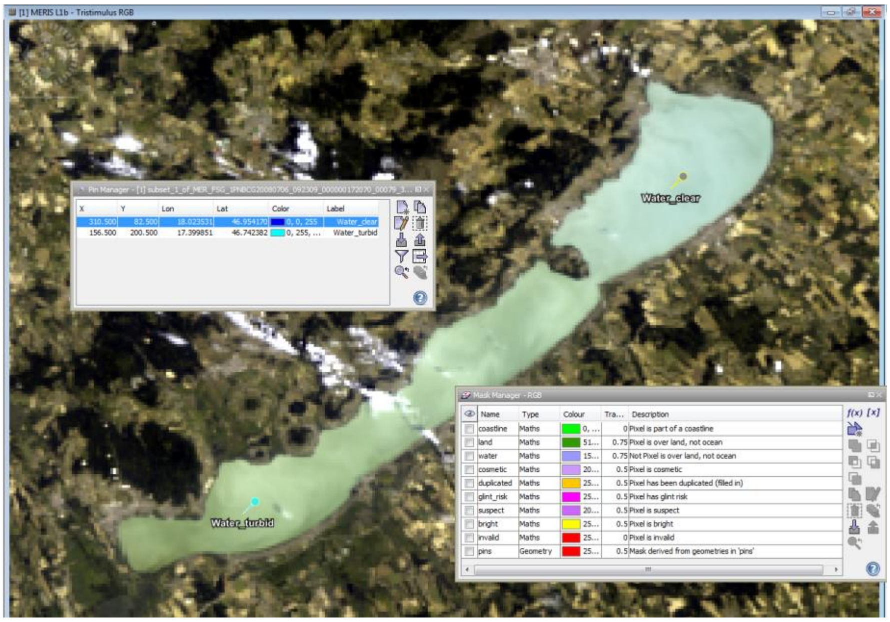


Figure 26. Pin Manager showing the two pins in the lake on the left. Mask Manager with the pins added as a mask.

Geometries

Once an image view is opened, new geometries can be created by using the various drawing tools provided by VISAT through the Interactions Toolbar: lines, polylines, rectangles, ellipses and polygons. Geometries are organized in geometry containers. All geometries within one container belong together and are handled as one geometry or mask (concerning statistics, training definition etc.). Once a new geometry container has been added to the data product, an associated geometry mask is created. The associated mask will always have the same name as the geometry container.

In the Utilities label select Create Vector Data Container (Figure 27). A new window will be open and we can set a name for the vector container and a short description. For instance in the Figure 28 we have created a new container named DarkVeg, which will contain groups of pixels associated with “vegetation that appears dark” in the image, because they are forest, or shadows on vegetation. We can now start selecting the pixels using the geometry tools (Figure 29). We could create several of these vector data containers with pixels pertaining to several surfaces.

Another way of creating geometries for masks is importing vector data in BEAM: files of comma separated values files, ESRI shapefiles, MERMAID extraction files, SeaBASS data and SeaDAS track data can be imported, and automatically masks are generated.

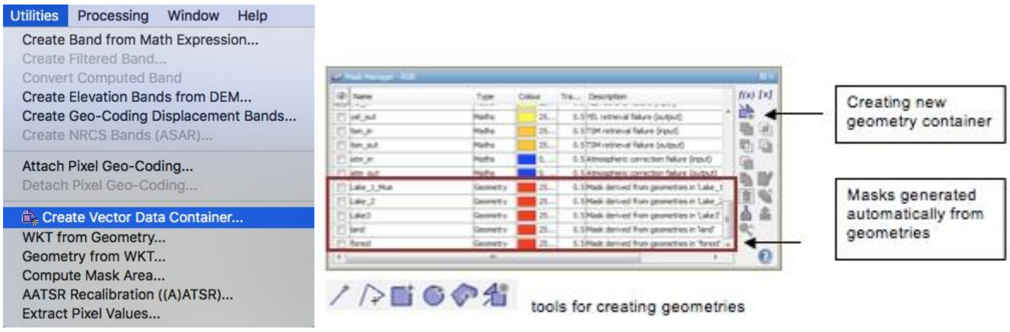


Figure 27. Create vector data container in VISAT

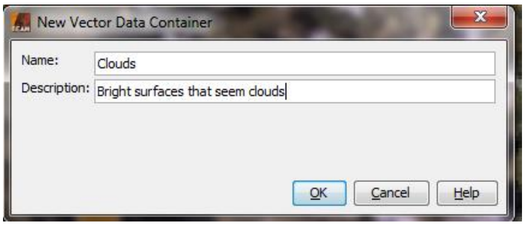


Figure 28. Create a new data vector container

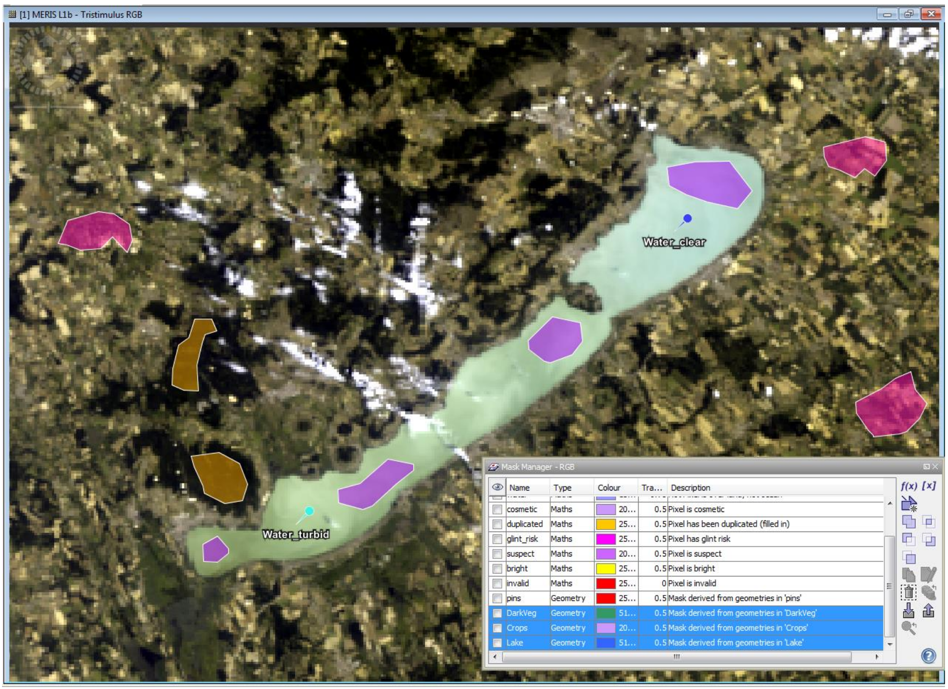


Figure 29. With the polygon tool, several geometries have been drawn inside the DarkVeg, Crops and Lake containers. The vector containers are stored in the mask manager by default as masks

Magic Wand

The Magic Wand tool is accessible by clicking on the icon  on the right panel of the VISAT view window. When the tool is opened, several options can be selected (Figure 30). The first step to make good use of the tool is to open the image of interest and check in the *Colour Manipulation* window or using the *Histogram tool*, the distribution of the data. This is important because the Tolerance range should be changed from the default (0 – 1) to values closer to the ones in the image. For instance, if the image is in the range of 0 to 256 values, to set the threshold from 0 to 100 would be a good way to start. Afterwards, click on a representative pixel with the right characteristics of the region of interest you want to identify. There are several options in the tool that can be selected depending on the characteristics of the region of interest and pixel features. In the Annex I of the GLaSS Deliverable 3.6 (2014) there is a detailed explanation of each one of the options for the transformation of the spectrum and the inclusion/exclusion test.

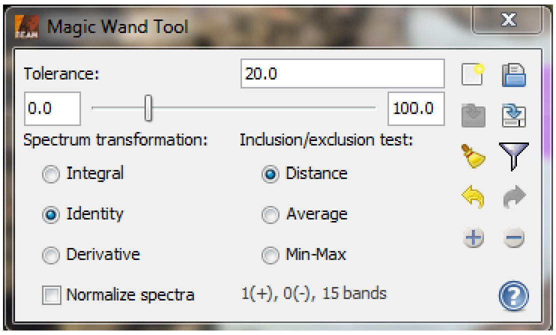


Figure 30. Magic Wand Tool window

Because we are working with an RGB view, we will set the range between 0 and 100. We are going to check for all pixels that could be clouds, so point at one cloud with the cursor to compile the information about the clouds, and set the Tolerance range in to 10. Results of this operation should be similar of the ones shown in the Figure 31. Play around with the different specrtum transformations and inclusion/exlcusion test and select the more appropriate for a good selection of the clouds.

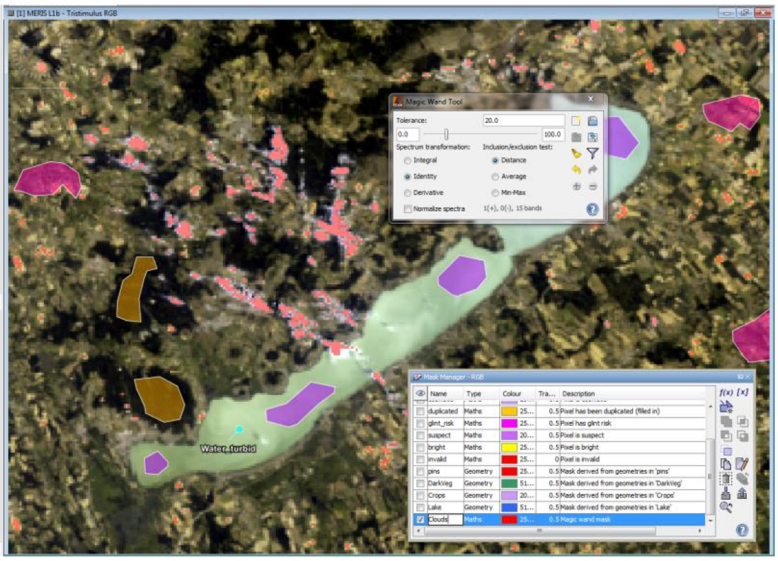


Figure 31. Cloud pixels selected with the Magic Wand tool. Pixel selected are marked in red, the Mask Manager shows the mask automatically created after selection of the pixels of interest

<u>Valid pixel expression</u>

To define valid pixels is a necessary pre-processing step before data extraction and further analysis. This step helps to detect unwanted pixels from the training data set and thus improves the statistics of the model. Unwanted pixels, i.e. not valid pixels, could be pixels affected by clouds, cloud shadow, shallow areas or correspond to any other criteria that excludes pixels that are erroneously interpreted at some step in the selected line of processing, e.g. calibration or atmospheric correction errors. It will be up to the user to define the appropriate valid pixel expression as the line of processing can vary very much. The valid pixel expression can be specified in the prediction model window, which is further described below. An example is given in Figure 32, which displays the chlorophyll_ a concentration (FUB) in Lake Mälaren before and after masking of valid pixels. The valid pixel definition can also reduce the training data set to a certain region of the image by drawing a bounding box.

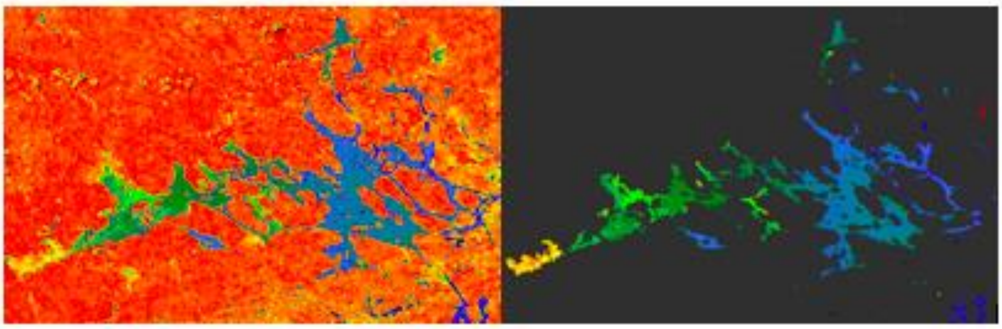


Figure 32. Chlorophyll_a in Lake Mälaren before and after masking of valid pixels

Model training

Once the definition of the training data has been done as explained in the previous section, the *Prediction Tool* can be opened by clicking on the icon. The Training Set label is highlighted first, and the user needs to select the masks that contain the training classes (Figure 33). The Training Set label has a link to the Mask Manager, from where the classes are selected and transferred into the Training Areas section. A valid pixel expression can be already stated here, in order to select only valid pixels from each of the defined training classes (Figure 34). In this case we have assigned as valid pixel expression all those pixels that are not considered to be invalid by the standard processor (not l1_flags.INVALID). The source dimensions are the input bands to be used by the model to extract information about the training classes. For the present exercise you can select the 15 radiance bands.

Once the training classes have been selected, the model is chosen and trained. In Figure 35, two selections are available: select the Model Category and Choose Method. The model category refers to the prediction model to be used. Only the Supervised Classification is now active. The method refers to the statistical and cluster analysis to be done on the training data. Only the Maximum Likelihood is available now. Once both boxes have been selected, the model can be trained (click on Train Model label).

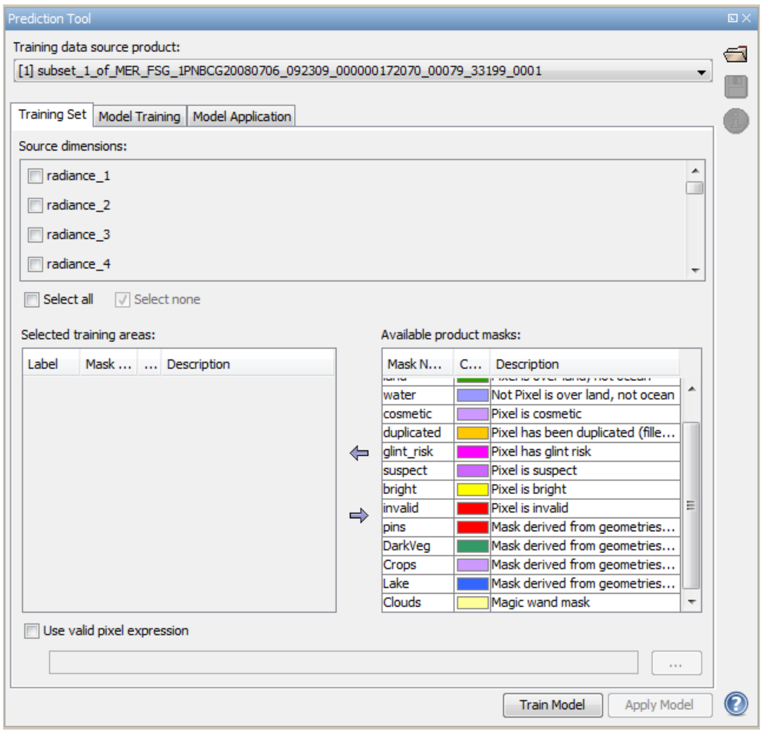


Figure 33. Training Set label from the Prediction Tool window. The reference image or source is selected and the corresponding source dimensions are shown in a list format- the available product mask are shown in the table on the right

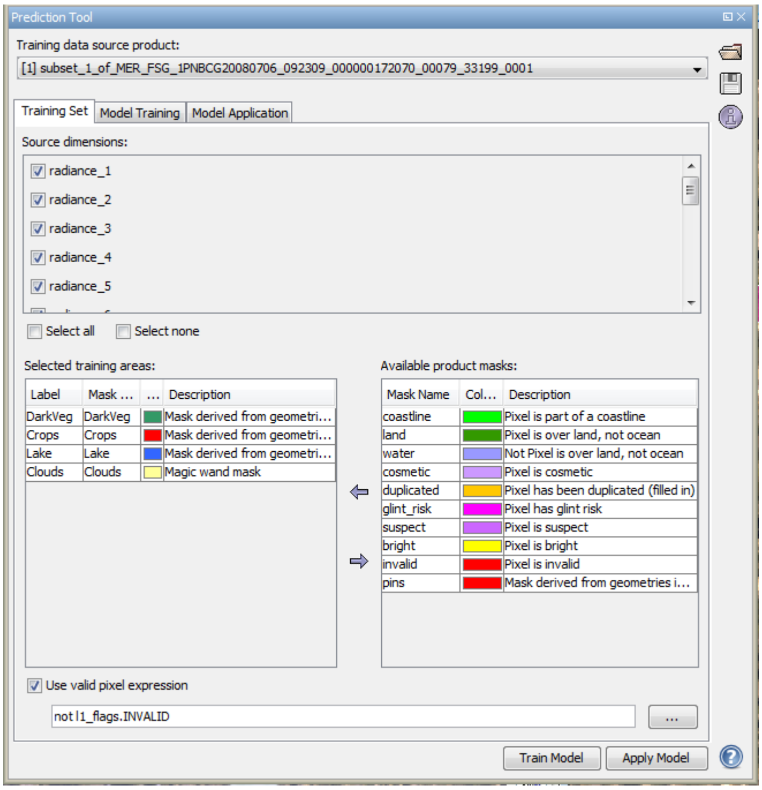


Figure 34. Training areas selected and a valid pixel expression introduced for the training set

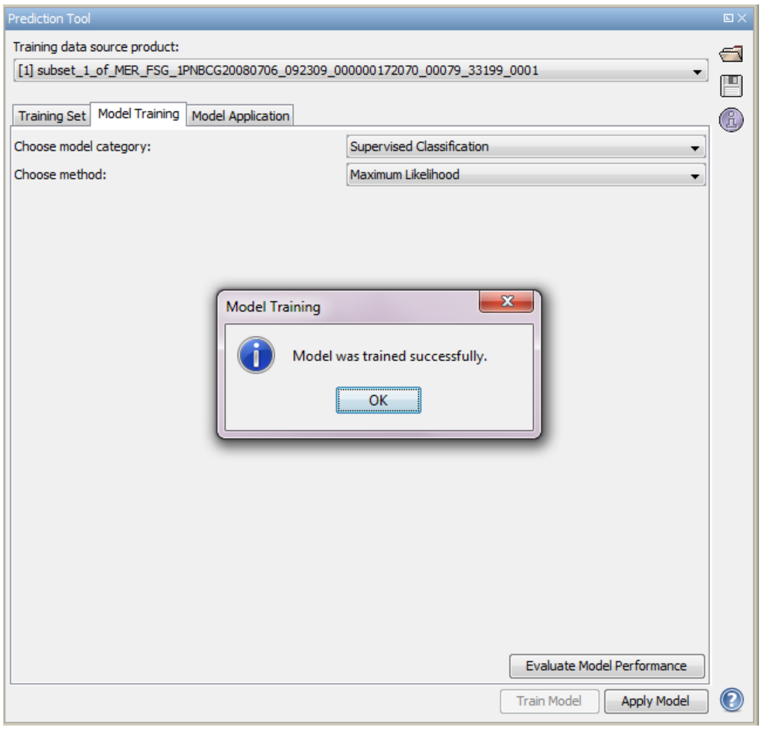


Figure 35. Model Training selection from the Prediction Tool window

The validation of the trained model is done by several metrics. The first one of these is a confusion matrix. A confusion matrix displays the relationship between the reference data (or ground truth) and the classification result. It is a square matrix with the number of rows and columns corresponding to the number of classes. The table displays which features have been classified onto which class. If there is a perfect fit, all values that do not lie on the main diagonal are 0. Figure 36 shows an example of a confusion matrix generated with the training data. The rows show the features of the reference data, the columns show the classification results. Further measures are the producer and the user accuracies. Both measures are calculated for each class and can be derived from the confusion matrix. The producer accuracy is the fraction of correctly classified pixels with regard to all pixels of that ground truth class, which means, it is the accuracy: for each class of ground truth pixels (row), the number of correctly classified pixels is divided by the total number of ground truth or test pixels of that class. On the bottom of the Figure 36 you can see the accuracy values derived for the confusion matrix above. It can be seen that the DarkVeg class has a producer accuracy of 94%, since is the result of 555/588 features that have been classified correctly (18 and 3 have been classified as Crops and Clouds, respectively). The user accuracy is also known as the reliability of the classification. It is the fraction of correctly classified pixels with regard to all pixels classified as this class in the classified image (in this case only the training data). For instance, for the DarkVeg class, the reliability is 555/555 = 1, in percentages 100%.

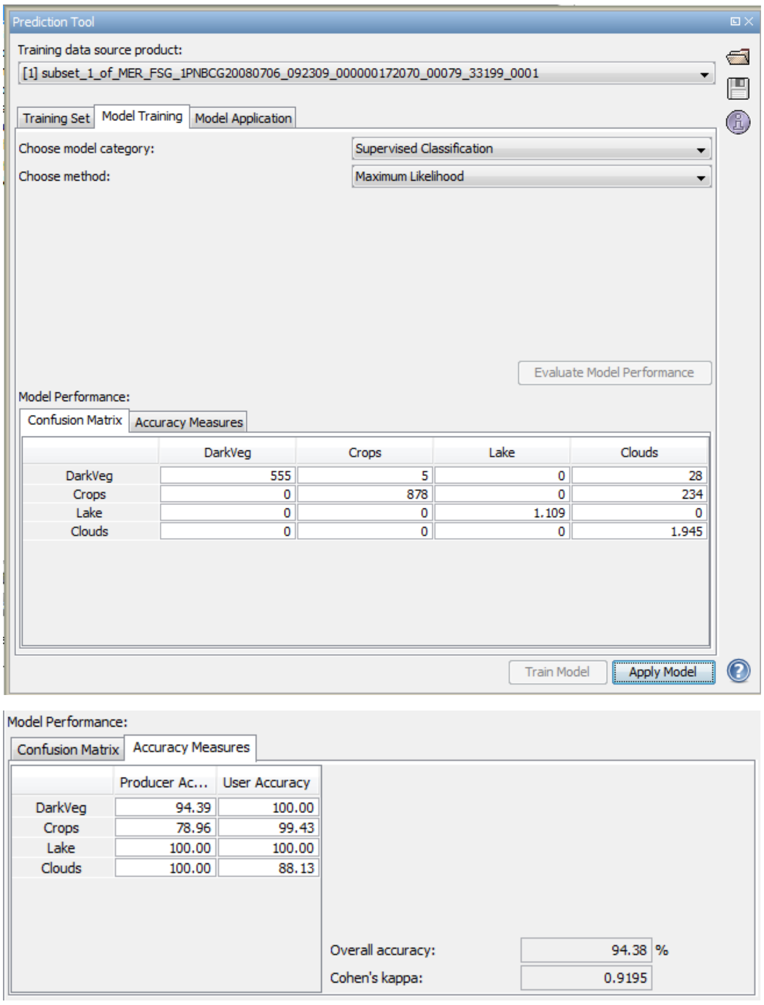


Figure 36. Model performance metrics and statistics

The overall accuracy is the ratio of correctly classified features against the total sum of features. Finally, the kappa coefficient κ, also called Cohen's kappa (COHEN 1960), is calculated from the confusion matrix. The kappa coefficient is a more robust measure, as it considers agreement by chance. The Kappa coefficient is always less than or equal to 1. A value of 1 implies perfect agreement and values less than 1 imply less than perfect agreement. In rare situations, Kappa can be negative. It is rare that we get perfect agreement. Different people have different interpretations about what is a good level of agreement.

Here is one possible interpretation of Kappa (Altman, 1991):

- Poor agreement = Less than 0.20
- Fair agreement = 0.20 to 0.40
- Moderate agreement = 0.40 to 0.60
- Good agreement = 0.60 to 0.80
- Very good agreement = 0.80 to 1.00

With the derived statistics and the confusion matrix, the user can decide to go further and apply the model to the full image, or to improve the training set if the results are not sufficient.

Apply and save the model

The last label corresponds to the Model Application (Figure 37). This is the section of the tool that allows to apply the trained model to the input data, and to define the desired output. The output, in this case the supervised classification, can be written in the input file as additional band, with a desired suffix, or can be saved as an independent product (Target Product, Figure 36). The directory can also be indicated here. The maximum likelihood classifier allows you to choose probability thresholds for each of the classes.

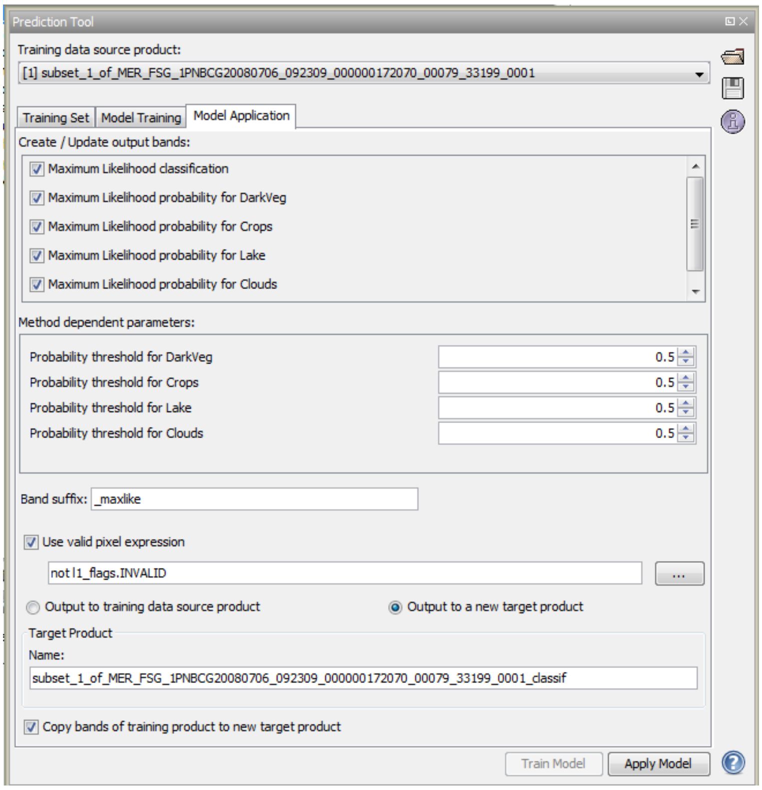


Figure 37. Model Application label from the Prediction Tool window

The model can be saved using the icon. A previously saved model can also be loaded for further adjusting using the button.

When the model has been conveniently saved, it can be applied to other images. This can be done with the Data Model Application Operator. In BEAM/VISAT it is located under Processing/Image Analysis. In the first tab the user can choose the product on which to apply the model as well as the target product to save the results to (Figure 38). In the Processing Parameters label, the model that shall be used can be loaded and configured. Figure 39 shows what the tab looks like when it is first opened. It allows the setting of model-independent parameters: A band suffix, a valid pixel expression and the choice to copy the bands from the source product to the target product. Also, a model can be loaded.

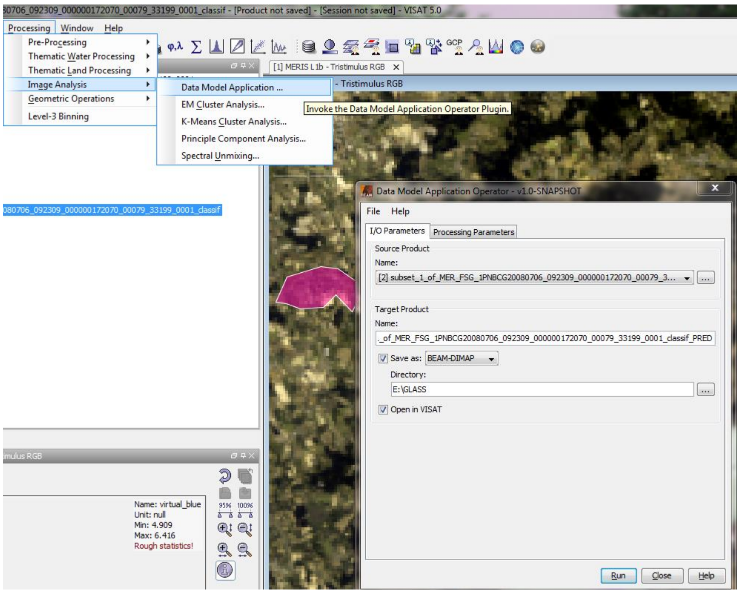


Figure 38. The Input/Output settings of the Data Model Application Operator

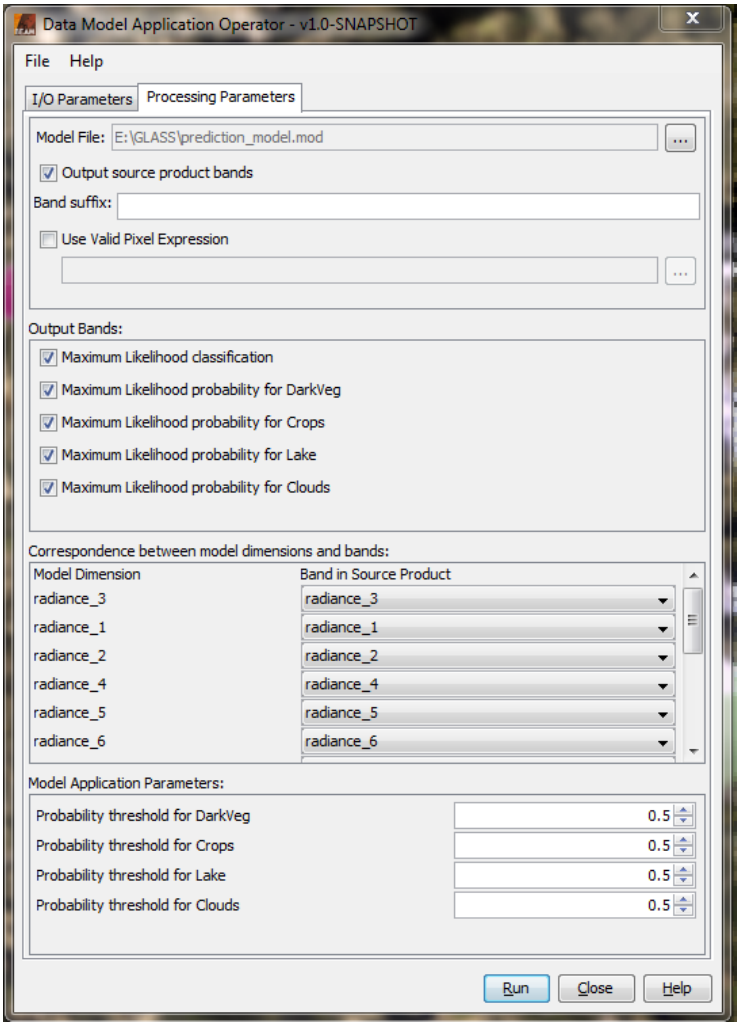


Figure 39. The Processing Parameters after a Maximum Likelihood model has been loaded.

When the model is loaded the user can set which bands to include in the target product and the parameters which are specific to the model. In correspondence between model dimensions and bands, the user can choose which source dimension in the model shall correspond to which band in the source product. This allows to apply the model also on source products which bands are differently named than the ones in the training product.

Clicking Run will apply the model and create a target product which can be shown in VISAT.

As the Data Modeler Application Operator is integrated into the BEAM Graph Processing Framework, it can also be accessed from the command line. To do so, one must navigate to the BEAM bin directory. Typing gpt DataModeler.Op –h will display information on how to use the operator here. The GPF enables the operator to be run in batch mode.

Visualisation of the results

The results of the classification are shown in Figure 40. On the top left of the figure there is the RGB composite to compare. On the top center are the results of the classification, and we can observe there is an overestimation of the clouds. This is due to the brightness of some of the fields that should have been classified as crops. The other four images are the probability maps for the four classes, and we can see here that the cloud probability is too high for all the image. A better definition of the clouds, or a different band combination or use of new variables is recommended in this case to improve the classification. Another option for understanding how the classification will result is looking at the statistical information. On the prediction tool main window there is an information icon that gives access to a table with the mean and covariance value per band and class used in the classification (Figure 41).

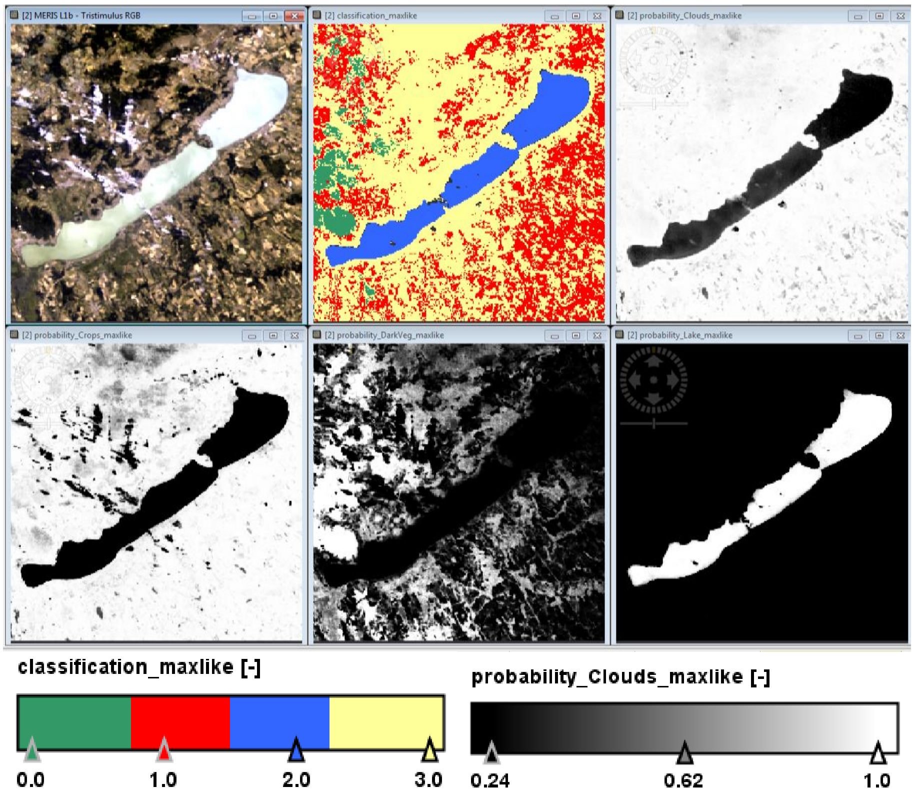
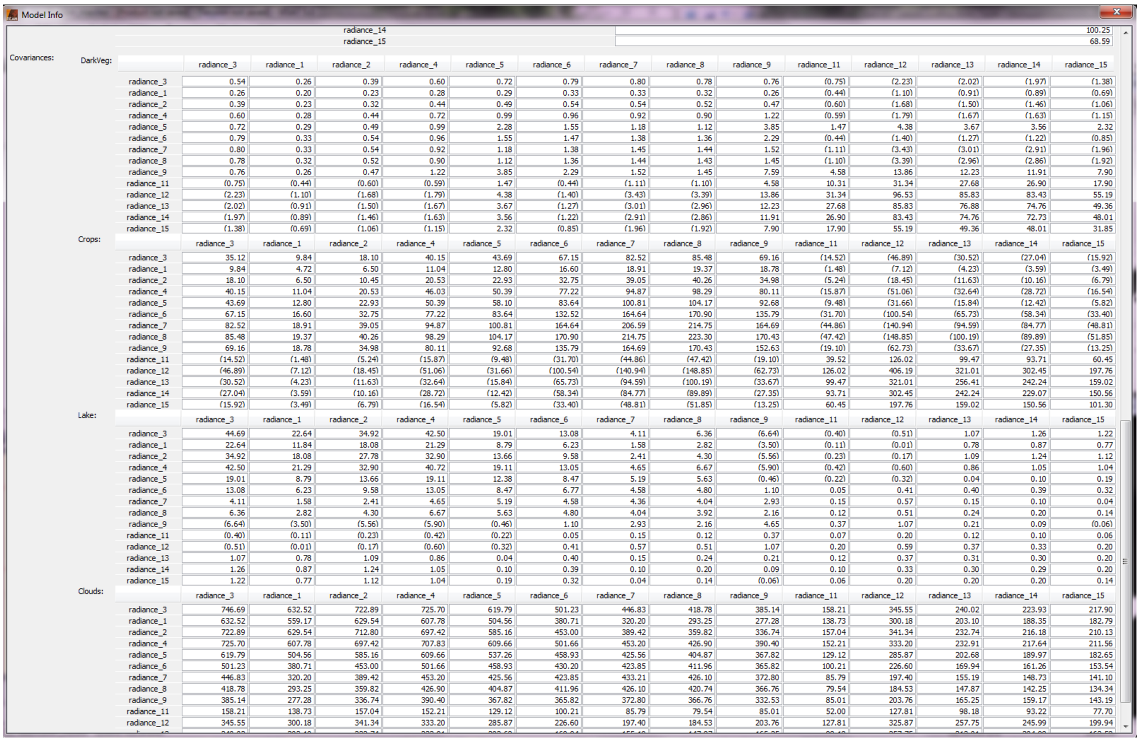


Figure 40. Results of the Maximum Likelihood Classification: from top left to bottom right, RGB of radiance, maxlike classification, clouds, crops, dark vegetation and lake probability classifications



Model Info

radiance_14	100.25
radiance_15	68.59

Covariances:

DarkVeg:

	radiance_3	radiance_1	radiance_2	radiance_4	radiance_5	radiance_6	radiance_7	radiance_8	radiance_9	radiance_11	radiance_12	radiance_13	radiance_14	radiance_15
radiance_3	0.54	0.26	0.39	0.60	0.72	0.79	0.80	0.78	0.76	(0.75)	(2.23)	(2.02)	(1.97)	(1.38)
radiance_1	0.26	0.20	0.23	0.28	0.29	0.33	0.33	0.32	0.26	(0.44)	(1.10)	(0.91)	(0.89)	(0.69)
radiance_2	0.39	0.23	0.32	0.44	0.49	0.54	0.54	0.52	0.47	(0.60)	(1.68)	(1.50)	(1.46)	(1.06)
radiance_4	0.60	0.28	0.44	0.72	0.99	0.96	0.92	0.90	1.22	(0.59)	(1.79)	(1.67)	(1.63)	(1.15)
radiance_5	0.72	0.29	0.49	0.99	2.28	1.55	1.18	1.12	3.85	1.47	4.38	3.67	3.56	2.32
radiance_6	0.79	0.33	0.54	0.96	1.55	1.47	1.38	1.36	2.29	(0.44)	(1.40)	(1.27)	(1.22)	(0.85)
radiance_7	0.80	0.33	0.54	0.92	1.18	1.38	1.45	1.44	1.52	(1.11)	(3.43)	(3.01)	(2.91)	(1.96)
radiance_8	0.78	0.32	0.52	0.90	1.12	1.36	1.44	1.43	1.45	(1.10)	(3.39)	(2.96)	(2.86)	(1.92)
radiance_9	0.76	0.26	0.47	1.22	3.85	2.29	1.52	1.45	7.59	4.58	13.86	12.23	11.91	7.90
radiance_11	(0.75)	(0.44)	(0.60)	(0.59)	1.47	(0.44)	(1.11)	(1.10)	4.58	10.31	31.34	27.68	26.90	17.90
radiance_12	(2.23)	(1.10)	(1.68)	(1.79)	4.38	(1.40)	(3.43)	(3.39)	13.86	31.34	96.53	85.83	83.43	55.19
radiance_13	(2.02)	(0.91)	(1.50)	(1.67)	3.67	(1.27)	(3.01)	(2.96)	12.23	27.68	85.83	76.88	74.76	49.36
radiance_14	(1.97)	(0.89)	(1.46)	(1.63)	3.56	(1.22)	(2.91)	(2.86)	11.91	26.90	83.43	74.76	72.73	48.01
radiance_15	(1.38)	(0.69)	(1.06)	(1.15)	2.32	(0.85)	(1.96)	(1.92)	7.90	17.90	55.19	49.36	48.01	31.85

Crops:

	radiance_3	radiance_1	radiance_2	radiance_4	radiance_5	radiance_6	radiance_7	radiance_8	radiance_9	radiance_11	radiance_12	radiance_13	radiance_14	radiance_15
radiance_3	35.12	9.84	18.10	40.15	43.69	67.15	82.52	85.48	69.16	(14.52)	(46.89)	(30.52)	(27.04)	(15.92)
radiance_1	9.84	4.72	6.50	11.04	12.80	16.60	18.91	19.37	18.78	(1.48)	(7.12)	(4.23)	(3.59)	(3.49)
radiance_2	18.10	6.50	10.45	20.53	22.93	32.75	39.05	40.26	34.98	(5.24)	(18.45)	(11.63)	(10.16)	(6.79)
radiance_4	40.15	11.04	20.53	46.03	50.39	77.22	94.87	98.29	80.11	(15.87)	(51.06)	(32.64)	(28.72)	(16.54)
radiance_5	43.69	12.80	22.93	50.39	58.10	83.64	100.81	104.17	92.68	(9.48)	(31.66)	(15.84)	(12.42)	(5.82)
radiance_6	67.15	16.60	32.75	77.22	83.64	132.52	164.64	170.90	135.79	(31.70)	(100.54)	(65.73)	(58.34)	(33.40)
radiance_7	82.52	18.91	39.05	94.87	100.81	164.64	206.59	214.75	164.69	(44.86)	(140.94)	(94.59)	(84.77)	(48.81)
radiance_8	85.48	19.37	40.26	98.29	104.17	170.90	214.75	223.30	170.43	(47.42)	(148.85)	(100.19)	(89.89)	(51.85)
radiance_9	69.16	18.78	34.98	80.11	92.68	135.79	164.69	170.43	152.63	(19.10)	(62.73)	(33.67)	(27.35)	(13.25)
radiance_11	(14.52)	(1.48)	(5.24)	(15.87)	(9.48)	(31.70)	(44.86)	(47.42)	(19.10)	39.52	126.02	99.47	93.71	60.45
radiance_12	(46.89)	(7.12)	(18.45)	(51.06)	(31.66)	(100.54)	(140.94)	(148.85)	(62.73)	126.02	406.19	321.01	302.45	197.76
radiance_13	(30.52)	(4.23)	(11.63)	(32.64)	(15.84)	(65.73)	(94.59)	(100.19)	(33.67)	99.47	321.01	256.41	242.24	159.02
radiance_14	(27.04)	(3.59)	(10.16)	(28.72)	(12.42)	(58.34)	(84.77)	(89.89)	(27.35)	93.71	302.45	242.24	229.07	150.56
radiance_15	(15.92)	(3.49)	(6.79)	(16.54)	(5.82)	(33.40)	(48.81)	(51.85)	(13.25)	60.45	197.76	159.02	150.56	101.30

Lake:

	radiance_3	radiance_1	radiance_2	radiance_4	radiance_5	radiance_6	radiance_7	radiance_8	radiance_9	radiance_11	radiance_12	radiance_13	radiance_14	radiance_15
radiance_3	44.69	22.64	34.92	42.50	19.01	13.08	4.11	6.36	(6.64)	(0.40)	(0.51)	1.07	1.26	1.22
radiance_1	22.64	11.84	18.08	21.29	8.79	6.23	1.58	2.82	(3.50)	(0.11)	(0.01)	0.78	0.87	0.77
radiance_2	34.92	18.08	27.78	32.90	13.66	9.58	2.41	4.30	(5.56)	(0.23)	(0.17)	1.09	1.24	1.12
radiance_4	42.50	21.29	32.90	40.72	19.11	13.05	4.65	6.67	(5.90)	(0.42)	(0.60)	0.86	1.05	1.04
radiance_5	19.01	8.79	13.66	19.11	12.38	8.47	5.19	5.63	(0.46)	(0.22)	(0.32)	0.04	0.10	0.19
radiance_6	13.08	6.23	9.58	13.05	8.47	6.77	4.58	4.80	1.10	0.05	0.41	0.40	0.39	0.32
radiance_7	4.11	1.58	2.41	4.65	5.19	4.58	4.36	4.04	2.93	0.15	0.57	0.15	0.10	0.04
radiance_8	6.36	2.82	4.30	6.67	5.63	4.80	4.04	3.92	2.16	0.12	0.51	0.24	0.20	0.14
radiance_9	(6.64)	(3.50)	(5.56)	(5.90)	(0.46)	1.10	2.93	2.16	4.65	0.37	1.07	0.21	0.09	(0.06)
radiance_11	(0.40)	(0.11)	(0.23)	(0.42)	(0.22)	0.05	0.15	0.12	0.37	0.07	0.20	0.12	0.10	0.06
radiance_12	(0.51)	(0.01)	(0.17)	(0.60)	(0.32)	0.41	0.57	0.51	1.07	0.20	0.59	0.37	0.33	0.20
radiance_13	1.07	0.78	1.09	0.86	0.04	0.40	0.15	0.24	0.21	0.12	0.37	0.31	0.30	0.20
radiance_14	1.26	0.87	1.24	1.05	0.10	0.39	0.10	0.20	0.09	0.10	0.33	0.30	0.29	0.20
radiance_15	1.22	0.77	1.12	1.04	0.19	0.32	0.04	0.14	(0.06)	0.06	0.20	0.20	0.20	0.14

Clouds:

	radiance_3	radiance_1	radiance_2	radiance_4	radiance_5	radiance_6	radiance_7	radiance_8	radiance_9	radiance_11	radiance_12	radiance_13	radiance_14	radiance_15
radiance_3	746.69	632.52	722.89	725.70	619.79	501.23	446.83	418.78	385.14	158.21	345.55	240.02	223.93	217.90
radiance_1	632.52	559.17	629.54	607.78	504.56	380.71	320.20	293.25	277.28	138.73	300.18	203.10	188.35	182.79
radiance_2	722.89	629.54	712.80	697.42	585.16	453.00	389.42	359.82	336.74	157.04	341.34	232.74	216.18	210.13
radiance_4	725.70	607.78	697.42	707.83	609.66	501.66	453.20	426.90	390.40	152.21	333.20	232.91	217.64	211.56
radiance_5	619.79	504.56	585.16	609.66	537.26	458.93	425.56	404.87	367.82	129.12	285.87	202.68	189.97	182.65
radiance_6	501.23	380.71	453.00	501.66	458.93	430.20	423.85	411.96	365.82	100.21	226.60	169.94	161.26	153.54
radiance_7	446.83	320.20	389.42	453.20	425.56	423.85	433.21	426.10	372.80	85.79	197.40	155.19	148.73	141.10
radiance_8	418.78	293.25	359.82	426.90	404.87	411.96	426.10	420.74	366.76	79.54	184.53	147.87	142.25	134.34
radiance_9	385.14	277.28	336.74	390.40	367.82	365.82	372.80	366.76	332.53	85.01	203.76	165.25	159.17	143.19
radiance_11	158.21	138.73	157.04	152.21	129.12	100.21	85.79	79.54	85.01	52.00	127.81	98.18	93.22	77.70
radiance_12	345.55	300.18	341.34	333.20	285.87	226.60	197.40	184.53	203.76	127.81	325.87	257.75	245.99	199.94

Figure 41. Information about a trained model. For each class, the means and the covariance matrices are listed

4 More information and further reading

This lesson is based on the following report:

- GLaSS Deliverable 3.3, 2014. Optical pre-classification method. VU/VUmc, BC, WI. Available via: www.glass-project.eu/downloads
- GLaSS Deliverable 3.6, 2014. Data mining BEAM module. BC. Available via: www.glass-project.eu/downloads

For more information on GLaSS, and to download all public reports: www.glass-project.eu.

References

- GLaSS Deliverable 3.3, 2014. Optical pre-classification method. VU/VUmc, BC, WI. Available via: www.glass-project.eu/downloads
- GLaSS Deliverable 3.6, 2014. Data mining BEAM module. BC. Available via: www.glass-project.eu/downloads
- Moore, T.S, Campbell J.W., Feng, H. (2001) A fuzzy logic classification scheme for selecting and blending ocean color algorithms. IEEE Transactions on Geoscience and Remote Sensing 39(8): 1764-1776
- Moore, T.S., Dowell, M.D., Bradt, S., Ruiz Verdu, A. (2014). An optical water type framework for selecting and blending retrievals from bio-optical algorithms in lakes and coastal waters. Remote Sensing of Environment 113(11), 2424–2430.

Colophon

Global Lakes Sentinel Services

GLaSS is funded by the European Commission (FP7)

Grant number 313256

GLaSS Training material, Lesson #2

Tools for GLaSS data analysis

Tools for statistical analysis of EO data

Ana Ruescas (BC)

2016

GLaSS Consortium

No	Name	Short Name
1	WATER INSIGHT BV	WI
2	SUOMEN YMPARISTOKESKUS	SYKE
3	EOMAP GmbH & Co.KG	EOMAP
4	STICHTING VU-VUMC	VU/VUmc
5	BROCKMANN CONSULT GMBH	BC
6	CONSIGLIO NAZIONALE DELLE RICERCHE	CNR
7	TARTU OBSERVATORY	TO
8	BROCKMANN GEOMATICS SWEDEN AB	BG