# **Advanced Remote Sensing**

# 1. Loading the data

The first step is to load the necessary data into eCognition software. Firstly, the main image is loaded ("QB\_MGI31\_subset\_SBG.tif") and it is important to choose the correct visualization settings, as seen in image 1.

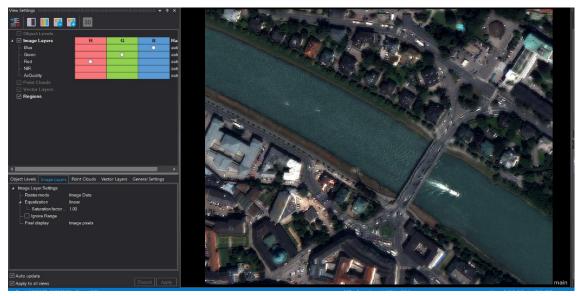


Image 1 – Visualization settings for the main image

After it, the second image is loaded, as an extra layer data. This image will be assigned as a fifth layer for our main image, as an air quality image layer (see image 2).

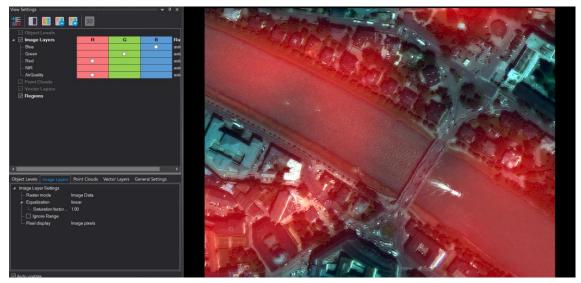


Image 2 – Second loaded image as a fifth layer

2. Chessboard segmentation

We add a new process to the Process Tree which is a chessboard segmentation (a way to segment an image by dividing it into an orthogonal matrix, like a chessboard). The input settings and the Process Tree can be seen on image 3.

Edit Process			×			
Name Automatc ChessBoardSegmentation	Algorithm Description Split the pixel domain or an image object domain into square image objects.			Process Tree		
Algorithm	Algorithm parameters	Value		Segmentation	ating 'ChessBoardSegmentation'	Set
chessboard segmentation	Parameter Object Size Level Name	10 ChessBoardSegmentation		eness bound. To ene	and chessed usegmentation	Aige
Domain	Cever Name Overwrite existing level Thematic Layers	Yes				
Parameter     Value       Condition        Map     From Parent						
Loops & Cycles				Main	_ n v	-
Loop while something changes only     Number of     1				Feature Object features Geometry	Value Extent	
Segmentation Preview		Execute Ok Car	ncel	Number of pixels	<no image="" object="" selec.<="" td=""><td></td></no>	

Image 3 – Chessboard segmentation settings and Process Tree

Selecting some object features, such as mean for each layer, and then selecting an image object (result from the chessboard segmentation), we can have a small statistic analysis (mean analysis) of the image object – see image 4.

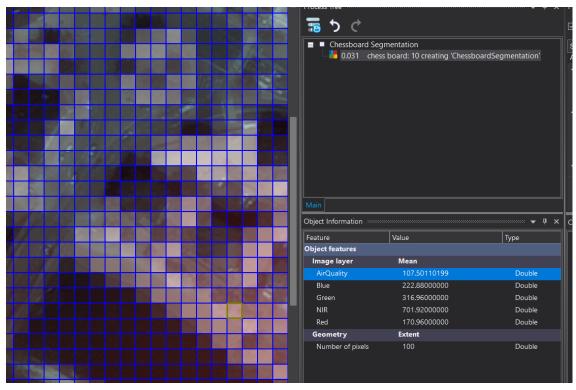


Image 4 - Mean values for each selected image object

3. More object features

Just like we add the mean as an object feature, we can add more and they can be found under Select displayed Features window. We are adding the following features (see image 5):

- Mean value of the blue band;
- Maximum pixel value of the blue band;
- Area of the selected object;
- Shape Index of the selected object;

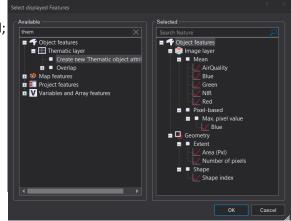


Image 5 – Adding features

It can also be interesting to compare different image objects (pixels) with different spectral values using the added object pictures, as seen on image 6.

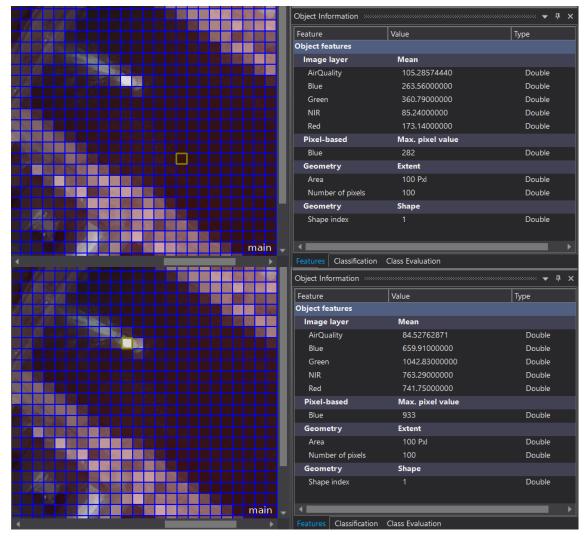


Image 6 – Comparison between two different image objects using the new added features

Besides the already prepared eCognition features, it is also possible to customize your own features. For example, it would be interesting to customize a new feature representing the NDVI (Normalized Difference Vegetation Index):

$$NVDI = \frac{(NIR - RED)}{(NIR + RED)}$$

To do so we create a new customize feature, following the input settings, as seen on image 7.

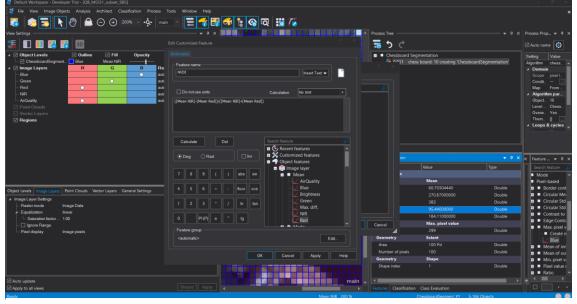


Image 7 – Input settings to create a customize feature to calculate NDVI

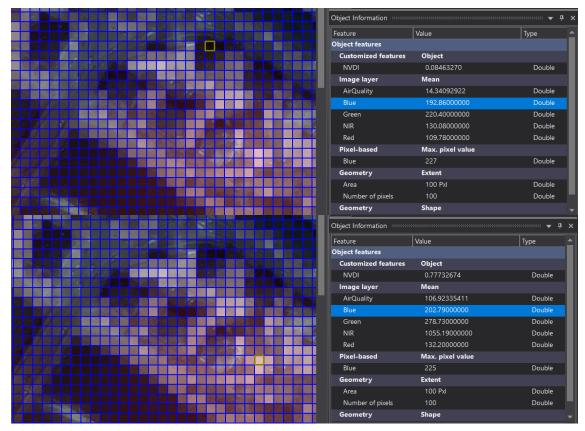


Image 8 – Comparing the values between two different image objects of different image areas

**4. Question 1:** What becomes obvious if you compare the values for the two objects of the chessboard segmentation?

It becomes obvious that chessboard segmentation is not ideal for analyzing and comparing image objects because it does not divide the image based on its features. Instead, it arbitrarily divides the image into uniform tiles. For example, when comparing NDVI between vegetation and water, it is harder to find objects that solely contain vegetation or water pixels, as each object may mix pixels from different features. This can lead to inaccurate comparisons, as the segmentation does not reflect the true boundaries of the study objects.

5. Question 2: Which features don't make that much sense in this case?

It doesn't make sense to use any features that rely on meaningful object boundaries since the segmentation doesn't take in consideration boundaries or object properties. Any geometry features, such as area or number of pixels don't make sense in this context because they will always be the same since the segmentation divides the images into same size and shape objects. Also, the shape index of the selected object feature doesn't make much sense because all the features objects that result from a chessboard segmentation have the same shape.

# 6. Multiresolution segmentation

To proceed with a supervised classification, it is better to have a multiresolution segmentation rather than a chessboard segmentation. Thus, we will delete the chessboard classification level and create a new process for the multiresolution segmentation, following the input settings as seen on image 9.

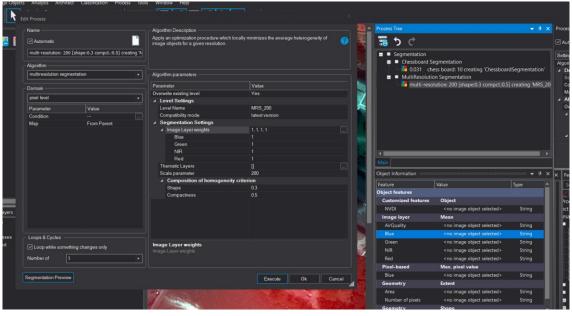


Image 9 - Multiresolution segmentation input settings

Using this segmentation method, it also makes more sense to use and compare certain image object with most of the object features talked before. See image 10 to look at the differences between two different image object and how their features differ from each other.

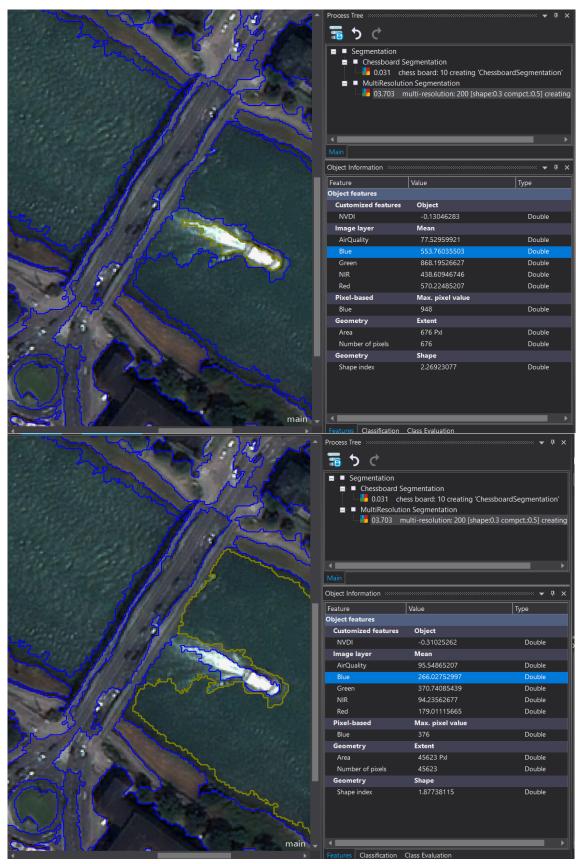


Image 10 – Comparison between two different image objects coming from a multiresolution segmentation

### 7. Updating NDVI features

Having an improved segmentation, we should find a good threshold value to distinguish between vegetation and non vegetation areas and water bodies. After updating the values and displaying them on the segmentated image, we'll get a similar result as the one shown on image 11.

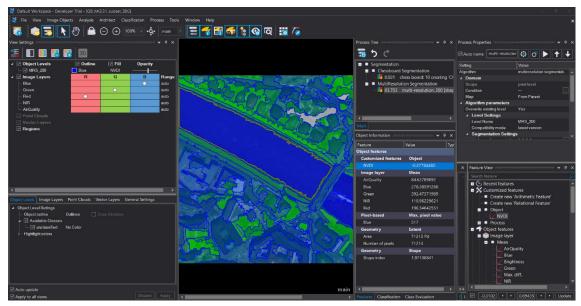


Image 11 – Updated NDVI values and displaying them

## 8. Creating and assigning classes

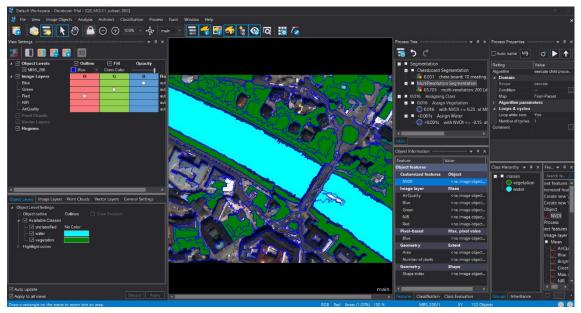
With a good segmentation done, we can start creating classes, like vegetation or water. We can create a process to assign classes concerning each class already created. This algorithm will make sure that our image will have assigned classes according to the NDVI parameters created just before. See images 12 and 13 for the input settings for both vegetation and water assignment class.

_ Name					Process Tree		× Process Properties
Automatic		Assign all objects in the image	object domain to the class specified by the Use class parameter.	?	<u>ਜ਼</u> 5 ਟ		Auto name Assign Water
do					Segmentation		Setting
- Algorithm		Algorithm parameters —			🖬 🔳 Chessboar	d Segmentation	Algorithm
assign class		Parameter	Value		MultiReso	lution Segmentation	⊿ Domain
		Use class	water		🛓 🗖 Assigning Cla		Scope
- Domain					😑 💷 Assign Ve	getation	Condition
image object level					🗌 🔘 with N	VDI >= 0.25 at MRS_200: v	eç Map
					💷 🗖 Assign Wa	iter	Algorithm parameters
Parameter	Value						✓ Loops & cycles
Level	MRS_200				•		Loop while something cha
Class filter Condition	none NVDI <= -0.15				Main		Number of cycles
Condition	NVDI <= -0, 15 From Parent						
Region	From Parent				Object Information		×
Max. number of objects	all				Feature	Value T	VP .
Samples only	No				Object features		
				- 12	Customized featur	es Object	
				<del>-</del>	NVDI	-0.27784480	Class Hierarchy
CLOOPS & Cycles					Image layer	Mean	· · · · · · · · · · · · · · · · · · ·
Loop while something c	hanges only				AirQuality	84.62789892	■ ■ classes
Number of 1					Blue	278.38591268	vegetation
					Green	392.47371969	water
					NIR	110.96229621	
			Execute Ok	Cancel	Red	196.34642551	
				la v	Pixel-based	Max. pixel value	
lable Classes			A state of the sta	A A	Blue	517	

Image 12 – Input settings to assign class water

Name			Algorithm Description			Pro			Process Properties	
Automatic			Assign all objects in the image object	t domain to the class specified by the U	se class parameter. 🛛 😗		≣⊅ぐ		Auto name Assign Veget	ati
do							Segmentation			
			Algorithm parameters				Segmentation Chessboard Segmentation	amentation	Setting	
Algorithm			Parameter	Value			MultiResolutio		Algorithm	
assign class		· .	Use class	vegetation			Assigning Class	in Segmentation	▲ Domain Scope	
Domain			Cae class	regenation		<b>_</b>	<ul> <li>Assign Artifici</li> </ul>	al Surfaces	Condition	
							Assign Vegeta		Map	
image object level		· .					Assign Water		Algorithm parameters	
Parameter	Value					- 4			▲ Loops & cycles	
Level	MRS_200								Loop while something cha.	
Class filter	none								Number of cycles	
Condition	NVDI >= 0.25					<u> </u>	lain		Comment	
Мар	From Parent					Ot	bject Information			
Region	From Parent						eature	Value Typ		
Max. number of objects Samples only	all No						biect features	value iyp		
Samples only							Customized features	Object		
							NVDI	-0.27784480		
Loops & Cycles							Image layer	Mean	Class Hierarchy	r 4
Loop while something	hanges only						AirQuality	84.62789892	classes	
	and iges only						Blue	278,38591268	artifical_surfaces	
Number of 1							Green	392,47371969	vegetation	
							NIR	110.96229621		
				Execute	Ok Cancel	R.	Red	196.34642551		

Image 13 – Input settings to assign class vegetation



The result should be similar to the one shown on image 14.

Image 14 – Result after assigning water and vegetation classes

We can create a new feature called Relative border to... any already created class, such as water or vegetation. This will show the index value of how much an image object is in contact with another object with an assigned class. The best example for it is the boat on the river, which is surrounded by water assigned image object – take a look at image 15 to see the values for the boat image object.

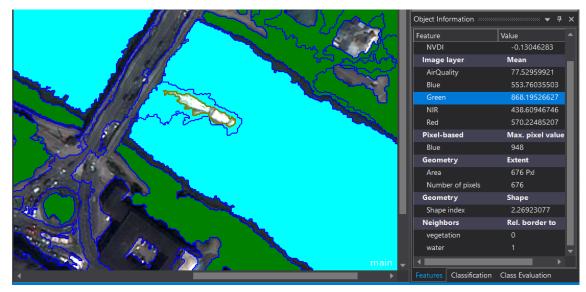


Image 15 – Relation between boat image object and water image object

Knowing this, we can create a new class specially for the boat and use the assign class algorithm along with the relative border to feature to classify it. We will use the unique relationship with the water to have the perfect class assignment (see image 16).

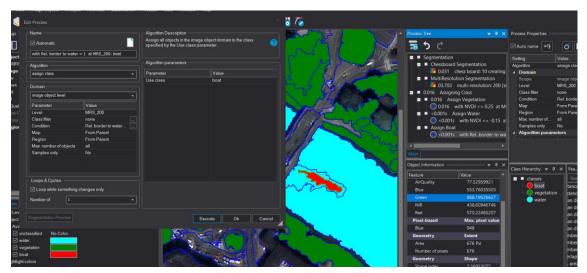


Image 16 – Assigning a class to the boat with relative border to feature

#### 9. Using air quality level

Now we will take advantage of the air quality level to improve our classification and class assignment. To do so we will create two new sub classes for the vegetation class (high\_air\_quality and low\_air\_quality). After it we will use the assign class algorithm again but taking into consideration the new created classes and the air quality image layer. In the end, the algorithm will divide the classified image objects as vegetation into high and low air quality. Take a look at image 17 for the input settings and image 18 for the final result.

Name		Algorithm Description		Process Tree • R × Process Proper	
Automatic		Assign all objects in the specified by the Use cl	image object domain to the class ass parameter.	0 Auto name	Ass
do					Val
Algorithm		Algorithm parameters -		Segmentation     Setting     Algorithm	exe
assign class		Parameter	Value	🗌 🔡 🖉 MultiResolution Segmentation 🔹 Domain	
		Use class	high_air_quality	Condition	
Domain				0.016 with NVDI >= 0.25 at MI Map	Fre
image object level				Algorithm	parameter
Parameter	Value MRS 200			O < 0.001s with NVDI <= -0.15 at 4 Loops & c     O     Construction     Constructio	
Level Class filter	vegetation			Assign Boat     Loop while     Assign Boat     Number of     Number of	
Condition	Mean AirQuality >= 50			Assign air guality Comment	
Мар	From Parent				
Region Max. number of objects	From Parent				
Samples only	No				
				Object Information - 4 × Class Hierarchy	• <del>•</del> • ×
- Loops & Cycles				Feature Value AirQuality 77.52959921	5
Loop while something of	handes only				
Number of 1					
Number of					high_air_q low_air_qu
				Red 570.22485207 wa	
			Execute Ok Cano	el Pixel-based Max. pixel value	
				* 🗑 72	
- Name		C Algorithm Description -		Process Tree	artias
Automatic	-	Assign all objects in the	image object domain to the class		
		specified by the Use cla	iss parameter.	🖉 📅 🔊 🦿 🖂 🗹 Auto nam	e 😳
vegetation with Mean AirQ	uality < 50 at MRS_200: low_			Setting	Va
Algorithm		Algorithm parameters –		Chessboard Segmentation     Algorithm	ast
assign class		Parameter	Value	Domain     Dependence     Depen	ims
Domain		Use class	low_air_quality	a 🖬 0.016 Assign Vegetation Level	MF
image object level				O 0.016 with NVDI >= 0.25 at MRS_200: v Class file	
Parameter	Value			Condition	n Me Fro
Level	MRS_200			COURS WINTWORK = 0.15 at MIKS_20     Region	Fre
Class filter	vegetation			oldstand          O<0.001s	
Condition	Mean AirQuality < 50			Samples	
Мар	From Parent			Algorith	im paramet
Region	From Parent			0.079 vegetation with Mean AirQuality	
Max. number of objects Samples only	all No				
Samples only				a the second	
				Class Hierar.	- n ×
Loops & Cycles					ses boat
Loop while something c	hanges only				vegetation
Number of 1	•			AirQuality 77.52959921 _	high_air_
					low_air_
			Execute Ok Cance	Green 866.19526627 -	water
				NIR 438.60946746 _	

Image 17 – Input settings to assign class algorithm for the high and low air quality

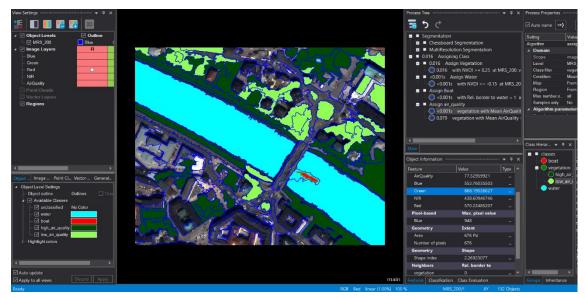


Image 18 – Image result of the class assignment

10. Question 3: How many objects were classified as "water"?

To answer this question we should create a new feature (Map Features > Objects > Number of Classified Objects > choose water class). We have 3 objects classified as water, as seen on image 19.

Feature	Value	Type	
Pixel-based	Max. pixel value		
Blue	376	Double	
Geometry	Extent		
Area	45623 Pxl	Double	
Number of pixels	45623	Double	
Geometry	Shape		I
Shape index	1.87738115	Double	ł
Neighbors	Rel. border to		ł
vegetation		Double	ł
water	0.21571072	Double	ł
Project/Map features			1
Objects	Number of classified o	objects	ł
water		Integer	,
Features Classification	Class Evaluation		

Image 19 – Number of object classified as water

**11. Question 4:** What is the area of the whole vegetation class (if you select the vegetation class to generate the feature, it will summarize the values from the grouped sub-classes)?

To answer this question, we should create a new feature (Map Features > Objects > Area of Classified Objects > choose the whole vegetation class). The whole vegetation class has 158660 pixels or 57117.6 m2, as seen on image 20.

Object Information			•	<b>д</b>	×
Feature	Value	Туре			
Geometry	Extent				
Area	71213 Pxl	Double			
Number of pixels	71213	Double			
Geometry	Shape				
Shape index	1.97108841	Double			
Neighbors	Rel. border to				
vegetation		Double			
water		Double			
Project/Map features					
Objects	Number of classified object	5			
water		Integer			
Objects	Area of classified objects				
vegetation	57117.59999999 m²				
Features Classification	Class Evaluation				

Image 20 – Area of the whole vegetation class

#### 12. New multiresolution segmentation

We will create a new multiresolution segmentation on a new level using the input settings seen on image 21. After executing this process, it is interesting to compare the two multiresolution segmentation that were performed so far (see image 22 for the comparison between the two of them).

ame		Algorithm De	escription					
Automatic		Apply an op	timization procedure	which locally minimizes the objects for a given resolution.	?	🚡 5 C		
at MRS_200: multi-resoluti	ion: 50 [shape:0.5 com	pct.:0.5				📮 🔳 Segment		
gorithm ———		Algorithm pa	rameters				board Segmentation Resolution Segmentation	
nultiresolution segmentati	on	- Parameter		Value			.703 multi-resolution: 200 [shap	per0.3 compct r0.51 creating 'MB
		Level S	ettings				d MultiResolution Segmentation	
omain		Level Na	ime	MRS_50			MRS_200: multi-resolution: 50 [sh	
nage object level		- Level Us		Create below		■ ■ 0.015 A		aperato competitorol creating i
			bility mode	latest version			Assign Vegetation	
arameter	Value		ntation Settings				1s Assign Water	
evel	MRS_200		e Layer weights				1s Assign Boat	
ass filter ondition	none	Themati					1s Assign air_quality	
ap	From Parent	Scale pa		50			is rosign an_quanty	
	From Parent From Parent		position of homog					
legion lax. number of objects	all	Shap		0.5				
amples only	No	Com	pactness					
ampies only	INO							
						Main		
oops & Cycles						Object Information	1	<b>→</b> 7
Loop while something c	hanges only					Feature	Value	Туре
Imber of 1						Geometry	Extent	
						Area	71213 Pxl	Double
						Number of pix	els 71213	Double
mentation Preview			Exec	ute Ok Ca	ncel	Geometry	Shape	
						Shape index	1.97108841	Double
						Neighbors	Rel. border to	
						vegetation		Double

Image 21 – Multiresolution segmentation input settings

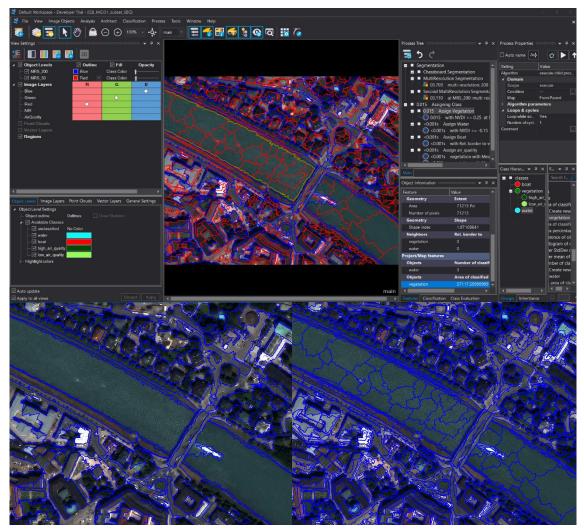


Image 22 – Comparison between the two multiresolution segmentations. Blue segmentation at top image and bottom image at left with scale parameter of 200 and red segmentation at top image and bottom image at right with scale parameter of 50

We will use a new feature called Existence of super object, which is directly connected wit the vegetation class, since it has two sub classes in it. We will use a the class vegetation and a distance value of 1. The result can be seen on image 23.

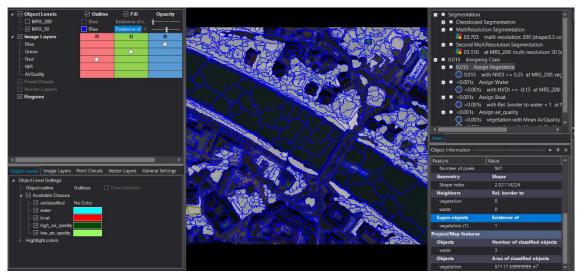


Image 23 - Result of Existence of feature visualization

13. Question 5: What is the meaning of the distance value when you create the feature?

The distance value is basically a Boolean value, meaning that 1 is considered true and 0 false. In this context, since we created the feature related to the vegetation class it will show if the class is either vegetation (distance value 1) or non vegetation (distance value 0). Also, the display colors on the image represent the vegetated or non vegetated areas. See image 24 to better understand the distance value.

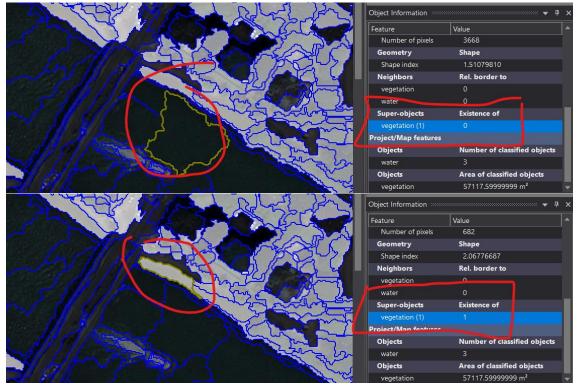


Image 24 – Comparison of Existence of feature