eCognition® 8.0: Guided Tour Level 1

Getting started – Example: Simple building extraction

Classification of buildings using elevation and RGB data

Version 3.0

Legal Notices

Copyright and Trademarks

The contents of this manual are subject to change without notice. Although every effort has been made to ensure the accuracy of this manual, we can not guarantee the accuracy of its content. If you discover items which are incorrect or unclear, please contact us using this form: http://www.ecognition.com/content/training-inquiries

Find more information about the product or other tutorials at

http://www.ecognition.com/

http://www.ecognition.com/community

Release Notice

Release date: August 2010

Table of Contents

Introduction

About this Guided Tour

Welcome to the Guided Tour 'Getting started – Example: **Simple building extraction'.**

This tour is written for **novice users** of the eCognition software.

This Guided Tour will focus on the **basic steps** involved in developing a Rule Set using **eCognition Developer**.

Further information about eCognition products is available on the website
http://ecognition.com.

Requirements

To perform this tutorial you will need

- **eCognition Developer** installed on a computer.

All steps of this Guided Tour, except the batch processing, can be done using **eCognition Developer or its trial version** (http://www.ecognition.com/products/trial-software).

This edition is designed for self-study.

Data included with the Guided Tour

Image data

Image data can be found in the folder '**.../Data**'.

There are two **.img** files available for this Guided Tour,
'RGB_Level1_Simple_Example.img' representing the aerial RGB data and
'DSM_Level1_Simple_Example.img' representing the LiDAR data.

Please note that the geoinformation of the data has been deleted.

Data courtesy of Woolpert, a premier supplier of ADS40 imagery and LiDAR data

Rule Sets

Rule sets are available in the **Rule Sets** folder. Whenever the Guided Tour refers to a Rule Set it is to be found in this folder.

Project

The ready set up project can be found in the folder **project**.

Import template

Import templates can be found in the folder '**Data**'. Whenever the Guided Tour refers to an import template it is to be found in this folder.

How to use this Guided Tour

Introduction

Symbols at the side of the document

Information

If the side is hachured and 'Introduction' is added, this indicates that the text is giving a general introduction or methodology about the following Lesson, method or exercise.

If the side is hachured and 'Information' is added, this indicates that the text is giving information about the following exercise.

Action!

If this symbol is shown, you have to follow the numbered items in the text. If you just want to work through the exercises without reading the theory part, follow only this sign.

✓ **Settings Check**

If this symbol is shown, compare the settings shown in the screenshot with the settings in the according dialog box in Developer.

✓ **Rule Set Check**

If this symbol is shown check the screenshot of the Process Tree with the content of the Process Tree in Developer.

✓ **Result Check**

If this symbol is shown check the screenshot aside with the result Developer. It should look similar.

Symbols for next step in Rule Set development

The following symbols show you always the next step to do. They will be prompted in the text.

Get the big picture

This diagram expresses that the next step to do is to develop the core strategy, containing the choice of the data to use. Get the big picture:

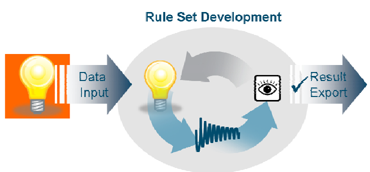


Figure 1: Get the big picture

Import data

This diagram expresses that the next step to do is to import your data:

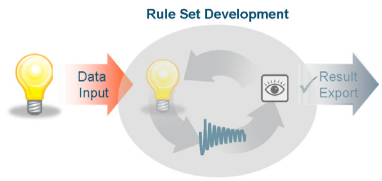


Figure 2: Import data

Develop strategy

This diagram expresses that the next step to do is to develop your next Rule Set development strategy step:

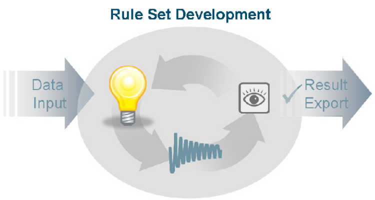


Figure 3: Develop strategy

Translates Strategy into Rule Set

This diagram expresses that the next step to do is to translate your strategy into a Rule Set:

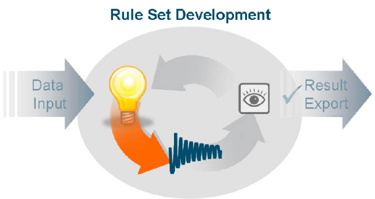


Figure 4: Translate into Rule Set

Review result

This diagram expresses that the next step to do is to review your results:

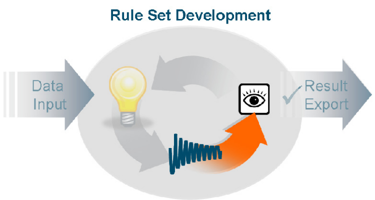


Figure 5: Review Result

Refine/expand strategy

This diagram expresses that the next step to do is to go back and develop your next strategy step, because the result did not meet your requirements:

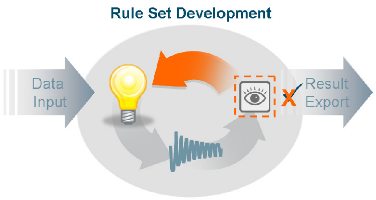


Figure 6: Refine/expand strategy

Ready for export

This diagram stands for the stage where you are satisfied with the result and you can export:

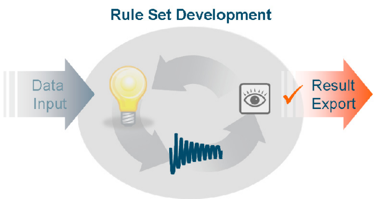
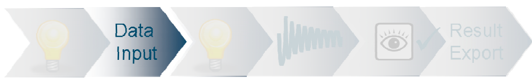


Figure 7: Result is OK, ready for export.

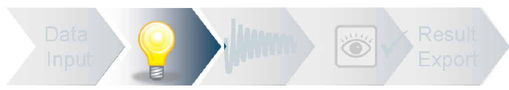
Symbols for stage in Rule Set development

There are also diagrams in the header of the document showing the current stage you are:

Get the big picture

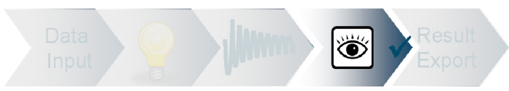


Import the data

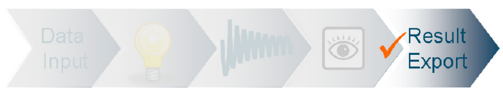


Develop strategy step

Write Rule Set



Review intermediate result



Export result

Lesson 1 Introduction to Rule Set development

This Lesson covers the following content

- → *Get the big picture*
- → *Which data to use?*
- → *Develop strategy*
- → *Translate the strategy into processes*
- → *Review your intermediate results*

1.1 Get the big picture

The most important tool for creating a Rule Set is your **expert knowledge**, for example as a remote sensing professional or a geographer, and the ability to translate your recognition process into the eCognition language: the Cognition Technology Language.

Behind every image analysis there is a methodology.

- 1 Get the big picture of the general analysis task
- 2 Choose the data
- 3 Develop a strategy
- 4 Translate the strategy into a Rule Set
- 5 Review the results
- 6 Refine the strategy and Rule Set if necessary
- 7 Export the results

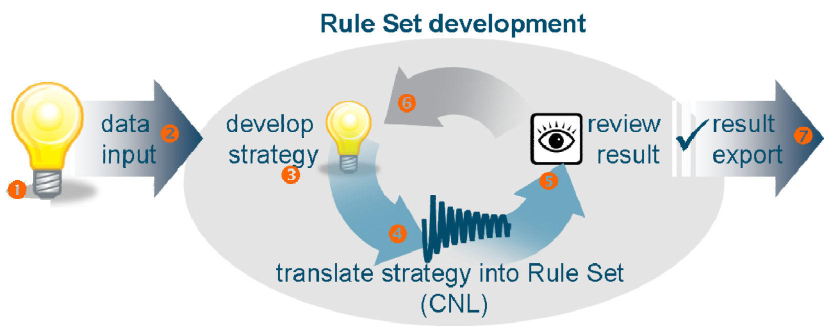
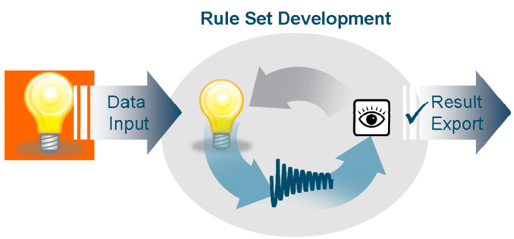


Figure 8: The Rule Set development process.

The diagram and the symbols above will guide you through the whole tour. The individual symbols will be visible in the header of the page, with the current stage highlighted.
Whenever there is a change e.g. from strategy development to Rule Set writing etc. the diagram above will show you the next phase.

1.2 Which data to use?



First step in Rule Set development is choosing the data to be used for analysis. In this Guided Tour buildings are extracted.

Data requirements:

- The data should have a **fine resolution** to get correct outlines.
- Due to the large variability of building types and roof tops, classification only based on spectral information is quite tricky. But **elevation information is a very consistent** source of information to classify buildings.

⇨ eCognition allows integration of data from **different sensors** and with **different resolutions**. In this example high resolution spectral images and a lower resolution DSM (Digital Surface Model) will be loaded together without problems.

⇨The software works with **objects**, therefore the resolution has direct impact on the object creation.

⇨The most stabile information about buildings is their **different elevation compared to their surrounding**. LiDAR data contains accurate elevation information in high resolution.

⇨Spectral information will help to separate buildings from other elevated objects like trees.

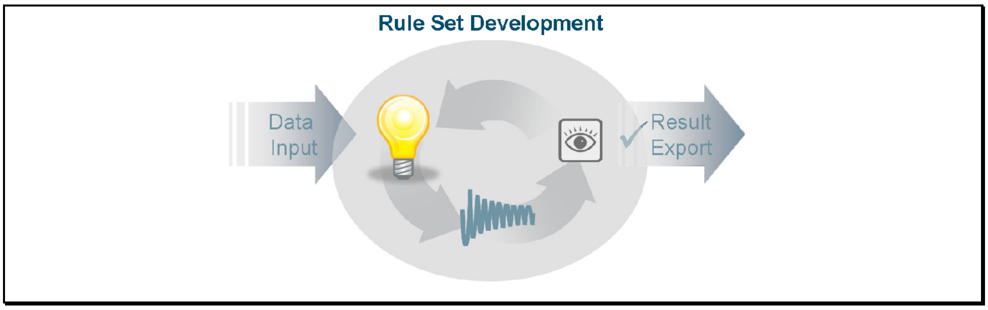
Conclusion:

To extract buildings a combination of spectral (here RGB) and LiDAR data is an optimum.

Efficiency in development

It is important to choose a **representative subset** of your data to develop an image analysis routing. The first step always is to **evaluate your complete data** set and then choose an area you want to start with. Working with a representative subset saves development time, because the processing times are lower than testing every step on the whole data set. Nonetheless it is recommended to test the Rule Set on **regular basis and not too late (!)** on the complete set, to avoid surprises at the end of the calculated development time.

1.3 Develop strategy



To find the best suited features and algorithms, always ask yourself, **why do I recognize something as a building, tree, lake?** Ask: how do I have to modify the objects so that they fulfill the criteria?

General rules

- Developing a Rule Set is an iterative process
- Always start with the class with the most significant features.
- Only process in the domain you are interested in.

eCognition works with objects. The image pixels are grouped together and as a result, much more information is available. Information like the spectral signature of the whole object, the shape and size and also context information is available. All these attributes can be used and combined for classification.

Tools to develop strategies

It is essential to know how to retrieve the necessary information to describe a class or to set up rules for processing, since the raw image may not clearly show the information that is needed. eCognition Developer contains tools that support your recognition process.

In eCognition Developer **visualization tools** and tools giving back **feature values** help you to display the information contained. Simply said: What you do not see/know, you can not describe. If the information is hidden in the data, use image filters provided in the software (like an edge filter) to be able to use this information as a rule.

The visualization tools…

…help to evaluate the data and/or the result

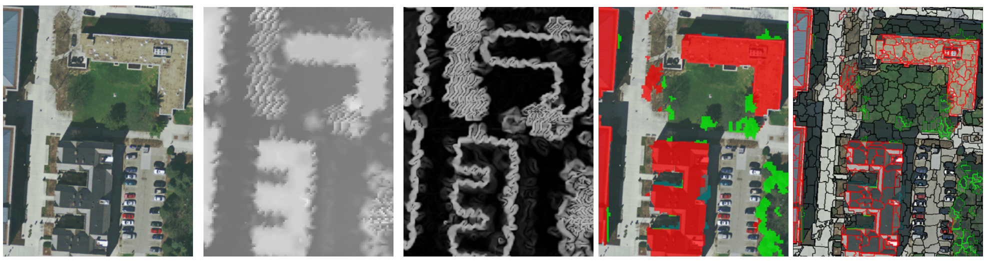


Figure 9: Different visualization possibilities of the loaded data and the classification results.

The ‘Feature View’…

…helps to evaluate the feature values for the entire scene

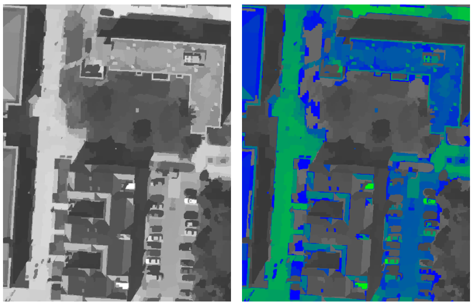


Figure 10: Viewer displaying feature values in grey range or in a color range.

The ‘Image Object Information’ window…

…helps to evaluate values for individual objects or evaluate classifications.

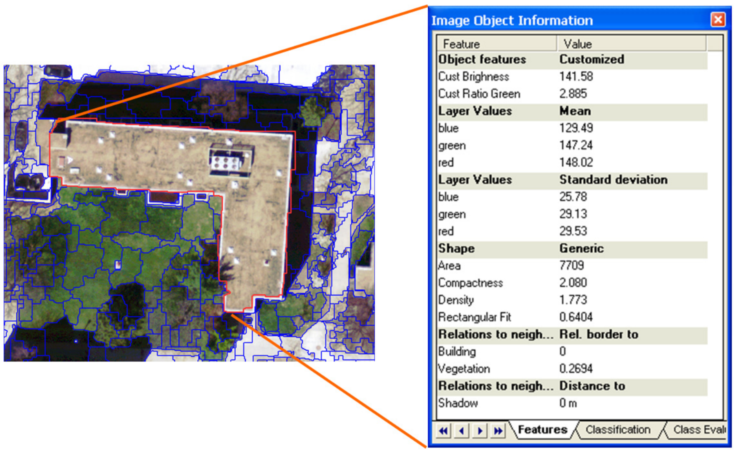
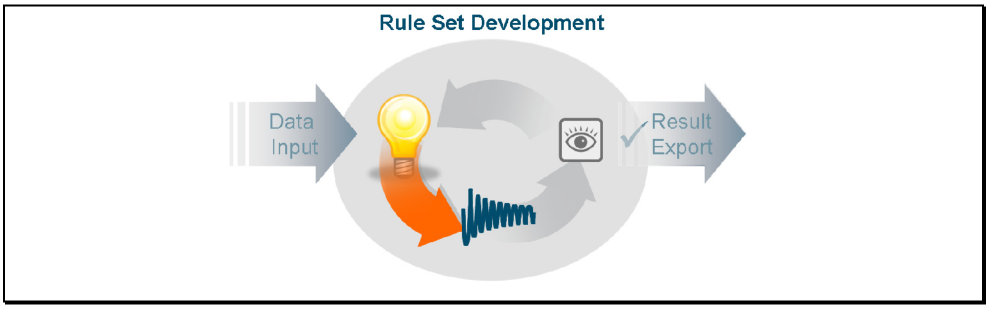


Figure 11: ‘Image Object Information’ window.

1.4 Translate the strategy into processes



To be able to write a Rule Set you have to translate your strategy into processes. This is done in the ‘Process Tree’ window. There you can add, edit and sort the processes.

In the ‘Process Tree’ window you can define:

- how shall be processed (algorithm)
- what and where shall be processed (domain)
- under which conditions (domain condition)

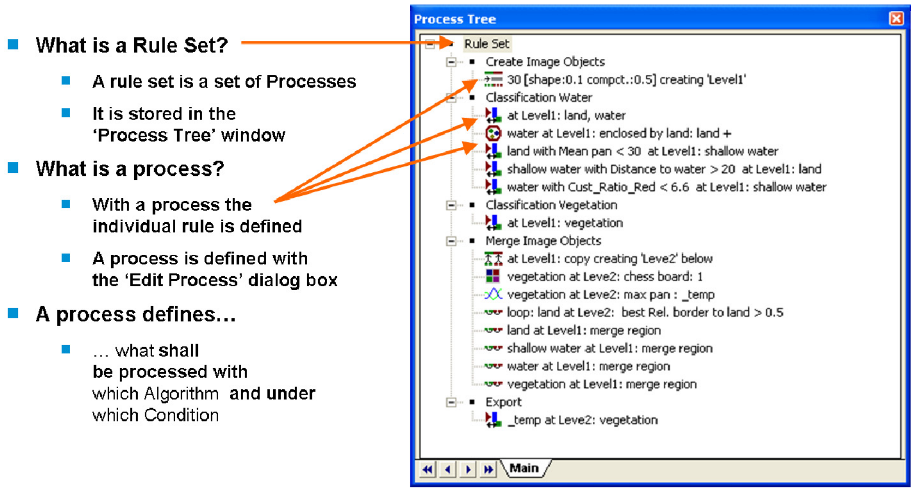
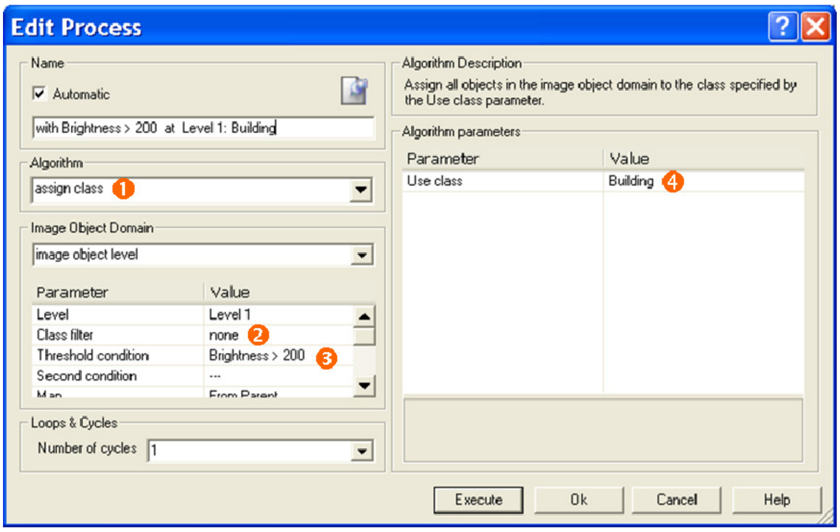


Figure 12: Diagram about the Process Tree and the individual processes.



Example: Assign ❶ all image objects ❷brighter than 200 ❸ to the class 'Building' ❹

A single process represents an individual operation of an image analysis routine for an image or subset. Thus, it is the **main working tool** for developing rule sets. A single process is the **elementary unit** of a rule set providing a solution to a specific image analysis problem.

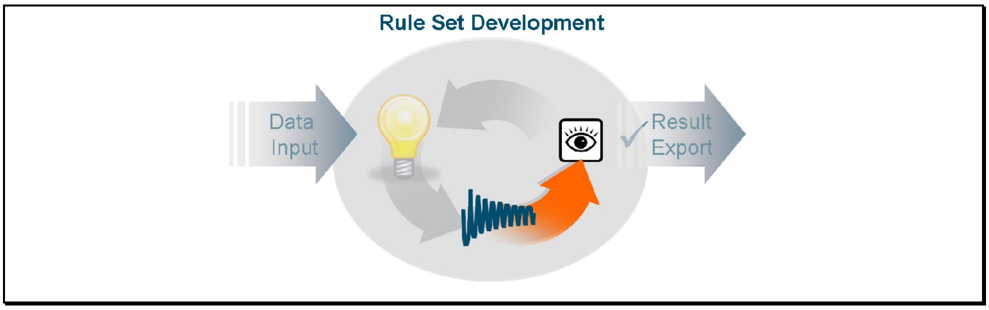
Every single process has to be **edited** to define an algorithm ① to be **executed** on an **image object domain** ②

Combine single processes in a **sequence** by building a **Rule Set**. You can organize a process sequence with parent and child processes which are executed in a defined order.

You can also **load** an existing Rule Set, **save** your rule set, and **execute** individual processes or the complete rule set.

Developing a Rule Set does not require you to write any code, rather one selects from a set of predefined algorithms within the graphical user-interface.

1.5 Review your intermediate results



As already mentioned, developing a Rule Set is an iterative process. You start with a base strategy, implement the actual rules in the software, check the result, then you step back and refine/expand the strategy, modify the Rule Set and again check the result. Until you are finally satisfied with the outcome.

Figure 13: Reviewing your intermediate results is a crucial step to develop the next steps of your analysis…

Figure 14:…or to decide that the accuracy of the classification is ready for export

Lesson 2 Get the big picture of building extraction analysis task

This Lesson covers the following content

→ *How buildings are represented in spectral image data?*

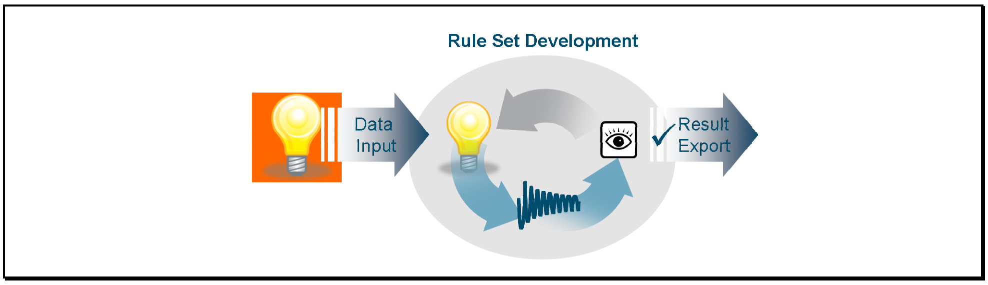
→ *Do buildings have significant shape characteristics?*

→ *Can buildings be separated using context information?*

→ *Is elevation information stabile characteristic for buildings?*

→ *Conclusion: Which data to be used? Which general ideas?*

→ *Overview Guided Tour 1: Simple example of building extraction*



To get the big picture you have to think about which general and consistent characteristics are contained…

- …in the data…
- …in the object-shape…
- …and whether there are context based characteristics.

2.1 How buildings are represented in spectral image data?

Roofs seen from above have a wide spectral variety, from colored roof tiles, to metal roofs.

→ Spectral Information about buildings is **inconsistent** information

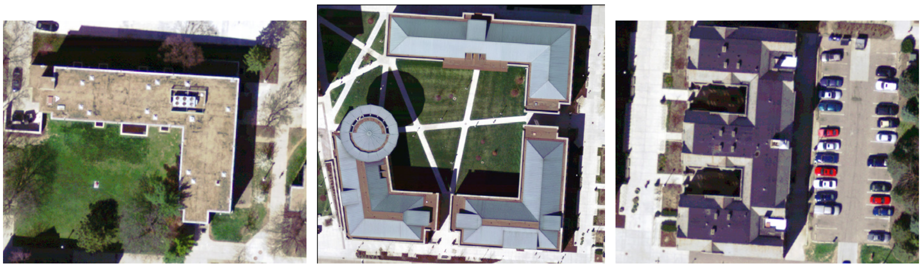


Figure 15: Example for different types of buildings.

2.2 Do buildings have significant shape characteristics?

Buildings have also a wide variety in shape and in size. They can be rectangular to circled, small family houses to bigger industry buildings. At least buildings have compared to other classes a quite large size.

→ **Partially inconsistent** information

2.3 Can buildings be separated using context information?

Buildings have shadows, as they are elevated.

Roofs are generally not covered by water or vegetation.

→ **consistent** context information

2.4 Is elevation information stabile characteristic for buildings?

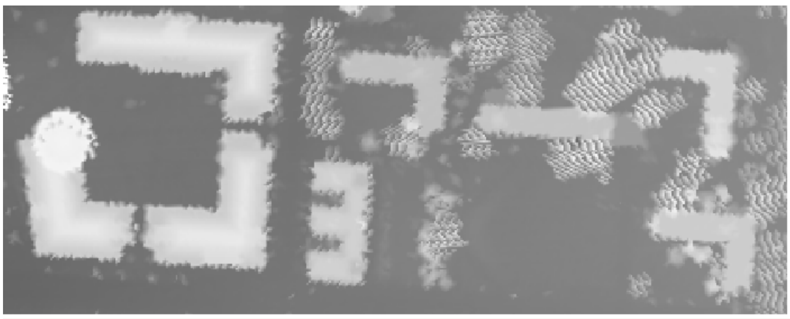


Figure 16: The DSM image layer.

Buildings have a higher elevation than their surrounding.

At the edges of a building there is a very steep slope. The elevation changes suddenly.

→ **Consistent** Information

2.5 Conclusion: Which data to be used? Which general ideas?

The most consistent and relevant characteristics of buildings is their elevation. Therefore elevation data (converted from LiDAR data) is chosen for the analysis task 'Building extraction'. The elevation information is used for segmentation and classification.

Only elevation is not sufficient to classify buildings correctly. Additional spectral information is needed, to separate e.g. buildings from trees. The elevation data used has a resolution of 2,5 feet, the aerial photo has 0,5. The spectral image layer will help to get more detailed outlines due to its higher resolution.

2.6 Overview Guided Tour 1: Simple example of building extraction

In the first Guided Tour a rather flat area will be classified. This has the advantage that the initial classification of elevated objects can take place using fixed height values.

 Figure 17: Classification result after first classification step.	• The classification will start by simply classifying all objects higher than a certain value.
 Figure 18: Classification result after second classification step.	• As a result trees are classified too. To get rid of these trees, their different appearance in the DSM is used to separate them from buildings. Additionally the spectral features for vegetation and shadow will be used to clean up the classification.
	• After that, missing parts of the buildings will be classified using context and size information.
 Figure 19: Final classification result.	As a result all buildings in the subset are classified with a good accuracy according the building outline. •

Lesson 3 Importing data

This lesson covers the following content

- → *Open eCognition Developer Rule Set Mode*
- → *Option 1: Use workspace and import template*
- → *Option 2 (Trial version): Creating an individual Project*

Introduction

All necessary data and Rule Set components are stored in the eCognition **project file (.dpr)**. The project files can be managed in the **workspace**. A workspace file (.dpj) contains image data references, projects, exported result values and references to the used Rule Sets. Furthermore, it comprises the import and export templates, result states, and metadata.

NOTE:

If you are working with the Trail Version, you have to jump to chapter 3.3 to **create** and **save** projects individually. The workspace functionality is not available.

Or if you have access to the full licensed product you will learn how to **manage** projects in the workspace. Then the data in this Guided Tour will be imported in the workspace using the 'Customized Import' tool.

3.1 Open eCognition Developer Rule Set Mode

Introduction

When starting eCognition Developer you can optionally start it in either:

- Quick Map Mode

or

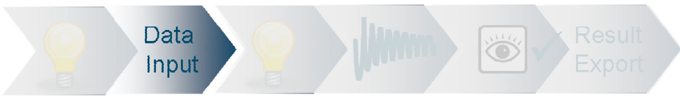
- Rule Set Mode

The Quick Map Mode is designed to allow a user **solving simple analysis tasks** without having to get involved with Rule Set development. The main steps in analyzing an image are creating objects, classifying objects and exporting results. For each of these steps, a **small assortment of predefined actions** is available.

The **Rule Set Mode** provides all necessary tools to **develop own Rule Sets**. In all the Guided Tours, the Rule Set Mode is necessary to accomplish the exercises.

Action!

1. Start eCognition Developer.
2. Select 'Rule Set Mode'.



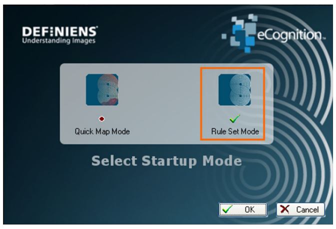


Figure 20: Select 'Rule Set Mode' to have all tools available.

Settings Check

3.2 Option 1: Use workspace and import template

This chapter covers the following content

→ *Create a new workspace*

→ *Import data in the workspace with an import template*

→ *How to open a created project*

→ *How to save a project*

3.2.1 Create a new workspace

Information

In the 'View Settings' toolbar there are **4 predefined View Settings** available. Each view setting is specific to the different phases of a Rule Set development workflow.

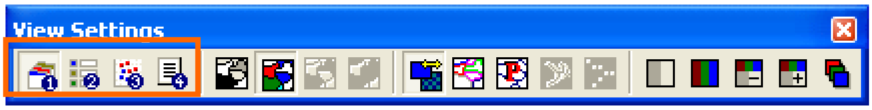


Figure 21: View Settings toolbar with the 4 predefined view setting buttons: Load and Manage Data, Configure Analysis, Review Results, Develop Rulesets.

To create, open or modify a workspace, make sure that you are in the 'Load and Manage Data' view.

Action!

3. Select the predefined view setting number 1 '**Load and Manage Data'** from the 'View Settings' toolbar.

4. To **create a new workspace**, do one of the following:

   - Click the '**Create New Workspace button**' on the toolbar.
   - Choose '**File > New Workspace**' from the main menu bar.

The 'Create New Workspace' dialog box opens.

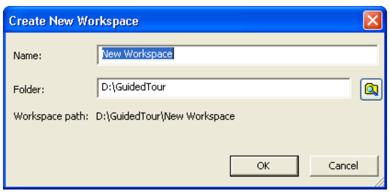


Figure 22: 'Create New Workspace' dialog box.

5. **Enter a Name** for the new workspace. The default name is ‘New Workspace’.
6. Browse for a **location to save** the new workspace.
7. Confirm with ‘**OK**’ to create the workspace. It is displayed as the root folder in the left pane (the tree view) of the workspace window.

3.2.2 Import data in the workspace with an import template

Information

There are several ways to import data into a workspace. You can create an **individual** project, import an **existing** project, use a predefined **import template** or the highly flexible ‘Customized Import’ tool.

In this Guided Tour the project used contains an RGB image file plus the LiDAR DSM.

Every channel, **red, green, blue** and the **DSM** shall have an **alias** to make further processing more transferable. This means the loaded data will not appear with its default naming Layer1, Layer2 etc. but have meaningful names like red, green, blue and DSM.

Besides that areas containing **no data values** have to be defined.

All these settings are already defined in the customized import template file (in **xml format**) which is used for import.

Action!

1. **Right-click** in the **left** workspace window and choose ‘**Customized Import**’ from the menu.

The ‘Customized Import’ dialog box opens.

2. Make sure that the ‘Workspace’ ① **tab** is active
3. Click on the ‘**Load**’ ② button and browse to the ‘**…GuidedTour1_eCognition8_GettingStarted_Example_SimpleBuildingExtraction\\Data**’ folder. Select the ‘**CustoImport_GuidedTour_Level1.xml**’.

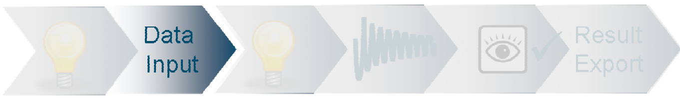
The settings defined in the .xml are loaded

4. Define the ‘**Root Folder**’ ③ by browsing to the folder the data is stored.
5. Define ‘**RGB_Level1_Simple_Example.img**’ as ‘Masterfile’ ④.

Note:

Select ‘No’ for the upcoming ‘Resolve conflicts’ error message. The information about different number of image layers refers also to the additionally to be loaded panchromatic layer.

6. Confirm with ‘OK’ ⑤.



✓ **Settings Check**

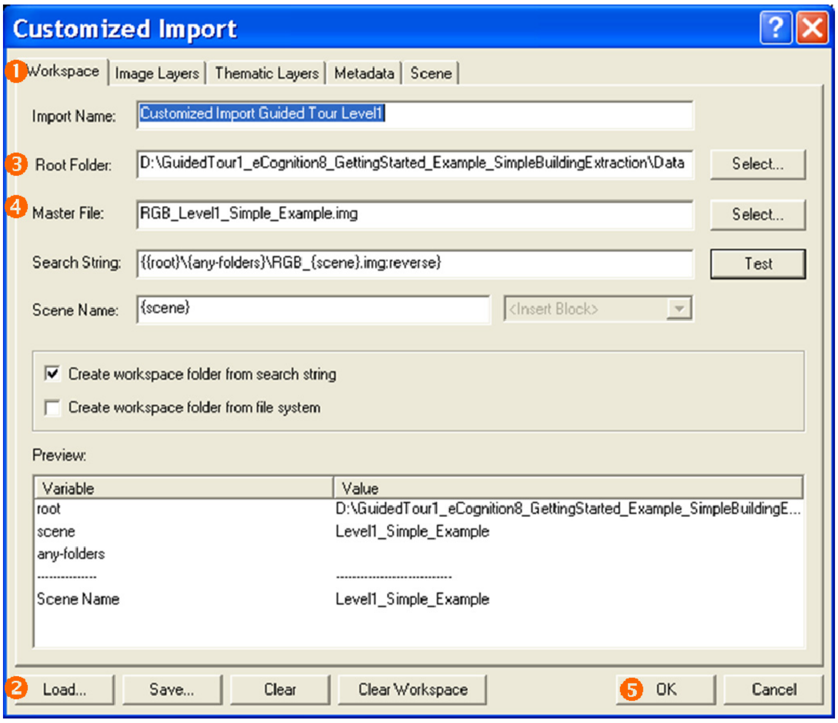


Figure 23: 'Customized Import' dialog box with settings to import the data.

> **Note:**
>
> Alternatively you can right-click in the left workspace window and select 'Import Existing Project' and browse to the folder where the Guided Tour1 data is stored. There you can find the 'Level1_Simple_Example.dpr' project file.

3.2.3 How to open a created project

There are several methods to open a project from the workspace

- **Double-click** on the project in the 'Workspace' window.
- **Right-click** on the project and select 'Open' from the main menu.

> **Note:**
>
> The currently opened project is marked in the 'Workspace' window with an asterisk.

3.2.4 How to save a project

Save the changes of a project by any of these methods:

- Click on the '**Save Project**' button.
- Or press Crtl + S on your keyboard.

NOTE:

As there is no undo command, it is recommended that you save a project prior to any operation that could lead to the unwanted loss of information, such as deleting an object layer or splitting objects. To retrieve the last saved state of the project, close the project without saving and reopen.

3.3 Option 2 (Trial version): Creating an individual Project

This chapter covers the following content

→ *Define the data to be loaded*

→ *Overview: 'Create Project' dialog box*

→ *Define layer alias*

→ *Assign 'No Data' values*

→ *Confirm the settings to create the project*

3.3.1 Define the data to be loaded

Action!

1. Click on the '**Create New Project**' button or go to main menu 'File>New Project'

The 'Create Project' dialog box together with the '**Import Image Layer**' dialog box opens.

2. Navigate to the folder '…\GuidedTour1_eCognition8_GettingStarted_Example_SimpleBuildingExtraction\ **Data'**.
3. Mark the following image files and click '**Open**'.
   - **RGB_Level1_Simple_Example.img04mar_multi.img**
   - **DSM_Level1_Simple_Example.img**

The '**Create Project**' dialog box opens.

4. In the 'Create Project' dialog box in the field 'Name' either enter a meaningful name for the project, e.g. '**Building extraction**'. Or keep the default naming according to the first loaded image file 'RGB_Level1_Simple_Example'.

3.3.2 Overview: 'Create Project' dialog box

The 'Create Project' dialog box has 4 main sections.

The general settings section 1:

- the geocoding information is displayed if the '**Use geocoding**' check box is selected and the resolution is automatically detected and displayed in the 'Resolution' field.
- the **unit** is detected automatically if **auto** is selected from the drop down list.
- The **unit** is automatically set to meters, but can be changed by selecting another one from the drop down list.

> **NOTE:**
>
> The geocoding for the data is not available with the data belonging to this Guided Tour.

- A subset of the loaded images can be selected by clicking the **Subset Selection** button. The 'Create Subset' dialog box opens.

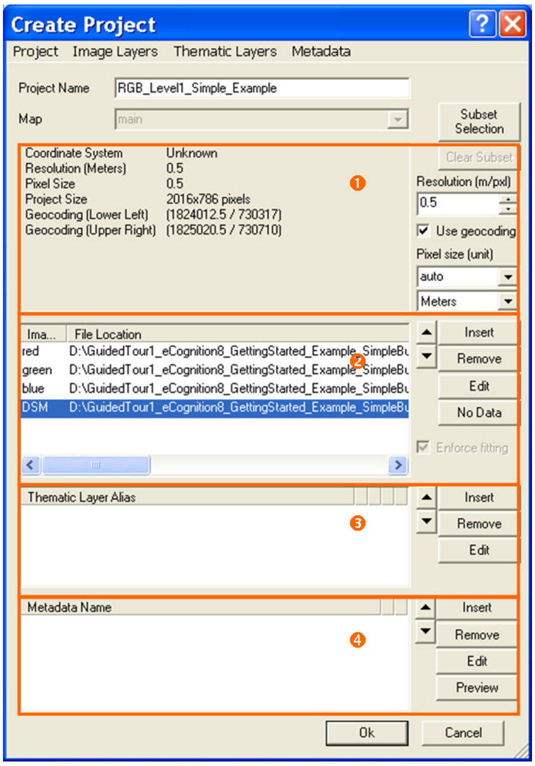


Figure 24: 'Create New Project' dialog box.

The image layer options section 2:

Introduction

- All preloaded **image layers** are displayed along with their properties. To select an image layer, click it. To select multiple image layers, press Ctrl or the Shift key and click on the image layers.
- To **edit a layer** double-click or right-click an image layer and choose '**Edit**'. The '**Layer Properties**' dialog will open. Alternatively you can click the '**Edit**' button.
- To **insert an additional image layer** you can click the '**Insert**' button or **right-click** inside the image layer display window and choose '**Insert**' on the context menu.
- To **remove** one or more image layers, select the desired layer(s) and click '**Remove**'.
- To change the order of the layers select an image layer and use the up and down arrows.
- To set **No Data** values for those pixels not to be analyzed, click '**No Data**'. The '**Assign No Data Values**' dialog box opens.

The thematic layer options section 3:

- To **insert** a thematic layer, you can click the '**Insert' button** or right-click inside the thematic layer display window I and choose 'Insert' from the context menu.
- **To edit** a thematic layer works similar to editing image layers described above.

The meta data options section 4:

Here you can load additional information data as an .ini file, if available.

3.3.3 Define layer alias

Information

In order to generate Rule Sets that are transferable between different datasets, the loaded channels have to have aliases assigned to them.

Action!

1. To assign a layer alias, select the **layer** in the 'Create Project' dialog box and **double-click** it.

NOTE:

The 'Layer Properties' dialog opens. Important for all 2 dimensional image analysis, as it is standard for Earth Sciences, the lower part of the dialog box 'Multidimensional Map Parameters' can be **ignored**.

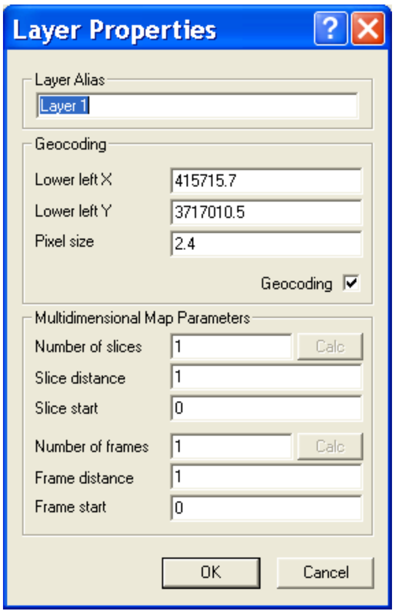


Figure 25: The 'Layer Properties' dialog box.

2. Assign for layer 1 'blue'.

3. Confirm the alias with 'OK'

Assign the following **aliases** to the other layers,:

- **Layer1 →red**
- **Layer2 →green**
- **Layer3 →blue**
- **Layer4 →DSM**

3.3.4 Assign 'No Data' values

Information

If there are areas with 0 or other artificial values not containing any relevant information, they can be set as 'No Data' areas. These areas will not be processed in any way, no objects will be created for these areas. In the current example, there are areas, where the RGB contains no information but the DSM, these areas are set to be 'No Data'.

Action!

4. Click on the '**No Data**' button.

5. In the field '**Global No Data Value**' switch on the check-box and enter the value 0.

This indicates that an area is defined as 'No Data' if any of the layers contains a 0 value pixel.

6. Confirm the settings with 'OK'.

3.3.5 Confirm the settings to create the project

7. After all alias are defined, click '**OK**' at the bottom of the 'Create Project' dialog box.

The new project is created.

Lesson 3 covered the following content

- → *Open eCognition Developer Rule Set Mode*
- → *Option 1: Use workspace and import template*
- → *Option 2 (Trial version): Creating an individual Project*

Lesson 4 Developing the core strategy by evaluating the data content

This lesson covers the following content

- → *The image visualization tools*
- → *Evaluate the data content*

Introduction

To create an initial strategy it is essential to **get familiar with the data**. eCognition has layer mixing tools that help to make the important information visible. You can zoom in and out to the scale of detail you are interested in and display the data individually or as layer combinations like RGB.

Two groups of tools are essential here: The zooming functions and the layer mixing tools. Additionally you have the possibility to open multiple viewers, for example to have on the one hand side the RGB displayed and on the other hand the DSM.

4.1 The image visualization tools

This chapter covers the following content

- → *Zooming into the scene*
- → *Display the DSM*
- → *Open a second viewer window to compare data*

Information

'Layer Mixing' tools are available in the 'View Settings' toolbar. You can display individual image layers or combinations and you can edit the equalization of a channel.

4.1.1 Zooming into the scene

Information

Depending on the level of detail you are working on or for switching between **detail and overview** you need the zooming functions of **eCognition Developer**.

They are all grouped in the '**Zoom**' toolbar.

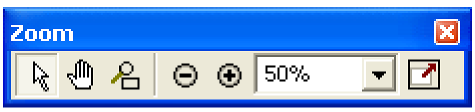


Figure 26: Zoom Toolbar

Action!

1. If you are working with the workspace, open the created project by **double-clicking** on it in the workspace window. Or, right-click on it and select 'Open' from the context menu.
2. Zoom in and out of the loaded scene, use the panning function to evaluate the data.

4.1.2 Display the DSM

Information

You have the possibility to change the display of the loaded data using the '**Edit Layer Mixing**' dialog box. This enables you to display the individual channels of a combination.

The first three bands of the loaded data are displayed by default in the **Red**, **Green** and **Blue** color guns, which is correct in this case. The third channel represents the elevation data.

Action!

3. To open the 'Edit Image Layer Mixing', do one of the following:
   - From the '**View**' menu, select '**Image Layer Mixing**'
   - or click on the '**Edit Image Layer Mixing**' button in the 'View Settings' toolbar.

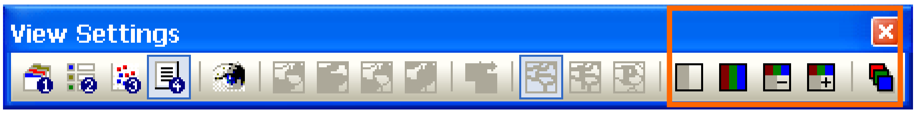


Figure 27: Layer mixing buttons in the 'View Settings' toolbar

Settings Check

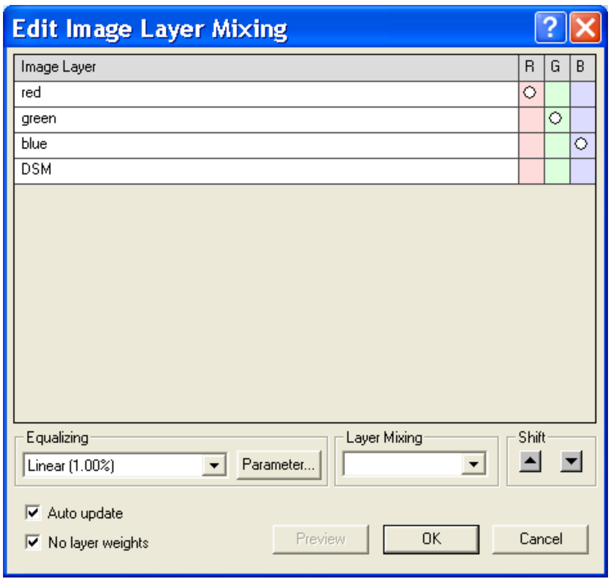


Figure 28: 'Edit Layer Mixing' dialog box.

Action!

4. Click on the dots for red, green and blue layers to deactivate them and click in one field of the DSM channel to activate it.
5. At the bottom of the '**Edit Layer Mixing**' dialog box, click '**OK**'.

The image will now be displayed using the view settings you specified.

Image Layer	R	G	B
red			
green			
blue			
DSM	○		

Figure 29: Only DSM is displayed.

Settings Check

4.1.3 Open a second viewer window to compare data

To compare the DSM and the RGB data, open a second viewer.

1. From the '**Window'** menu, select either **'Split Horizontally**' or '**Split Vertically**' to open a second viewer window.

Action!

A second window opens.

2. **To link** the two windows go to the menu '**Windows**' and select **'Side by Side View**'.

Now two viewer windows are open (see Figure 31) showing the same area of the loaded data. If now zooming in one window the other will follow and show the same content.

Result Check

4.2 Evaluate the data content

This chapter covers the following content

→ *Evaluate the elevation data*

→ *Evaluate the RGB data*

4.2.1 Evaluate the elevation data

Introduction

The elevation data show, that the buildings have higher elevation than the surrounding ground. This is the most obvious information and therefore it will be used for initial classification. In the area of the subset the buildings have different heights. At the end the buildings are separated into different height classes.

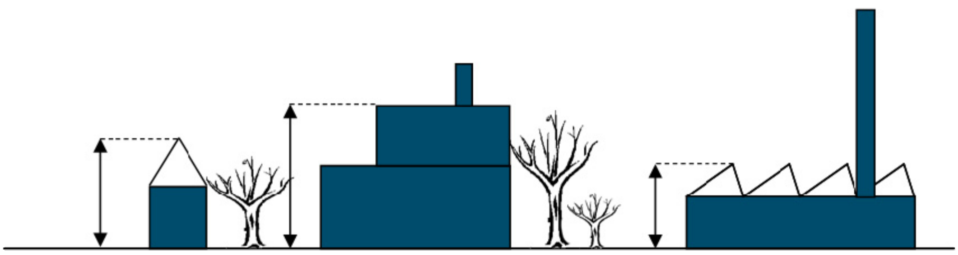


Figure 30: A stabile feature to describe buildings is their difference in height.

Introduction

Besides the buildings, the trees are elevated objects too, but displaying the DSM you can see that leaf-off trees, in contrast to buildings, give back very heterogeneous elevation information. This heterogeneity can be used as a feature to separate the trees from buildings.

Figure 31: Left: DSM layer; Right: RGM layer mix.

4.2.2 Evaluate the RGB data

In the RGB data trees and buildings appear different too. Trees have a lower size than the buildings and in the RGB data contains the information that meadow is shining through the branches. This information separates them from the buildings too.

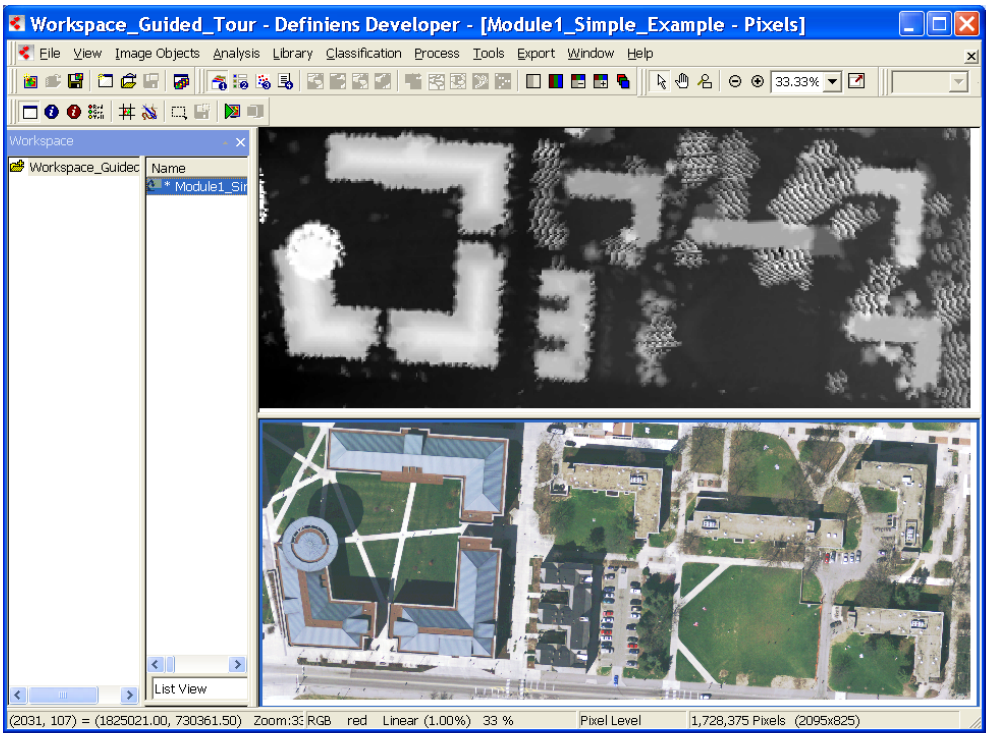
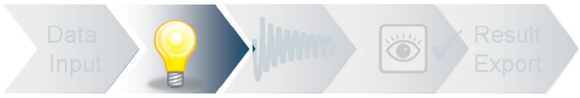


Figure 32: Loaded data with the workspace, displayed in two viewers. Above: the DSM layer; Below: the RGB layer mix.



This example shows that height information from the LiDAR data provides key information to extract buildings. However, since trees and buildings both are elevated, only additionally using their different appearance in LiDAR and RGB layer separates buildings from all other objects with elevation.

The rules in words:

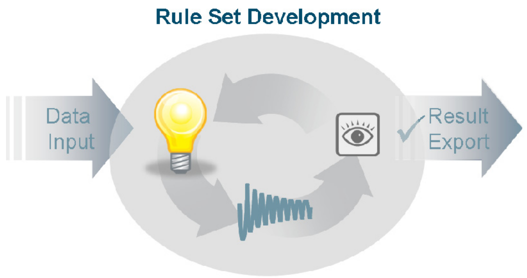
- Buildings are higher than a certain level -> Use the height information from the DSM
- Trees, which have the same elevation as buildings, have a heterogenic elevation information, are smaller and have spectral vegetation features. -> Use heterogeneity and the size information and classify elevated objects with vegetation features in a separate class.

Lesson 4 covered the following content

→ *The image visualization tools*

→ *Evaluate the data content*

Next step: Develop Strategy for first analysis step!



Lesson 5 Creating image objects

This lesson covers the following content

- → *Strategy for creating suitable image objects*
- → *Translate strategy into Rule Set – use algorithm 'multiresolution segmentation'*
- → *Review the created image objects*

5.1 Strategy for creating suitable image objects

Information

The fundamental step of any eCognition image analysis is a segmentation of a scene—representing an image—into image objects. Thus, initial segmentation is the subdivision of an image into separated regions represented by basic unclassified image objects called 'image object primitives'.

To get most realistic image objects they have to represent the necessary elevation information and they should be not too big and not too small for the analysis task.

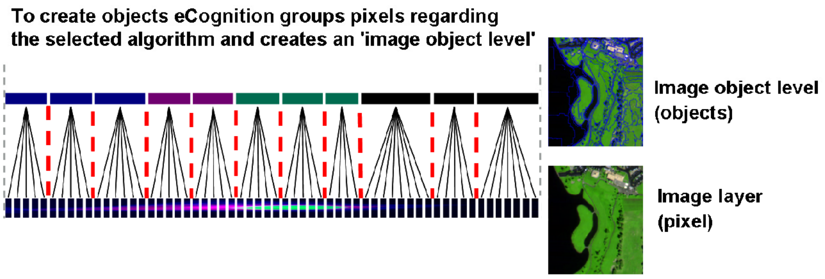
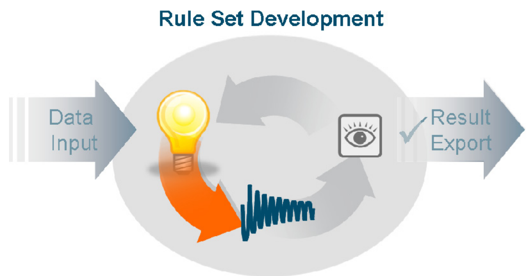
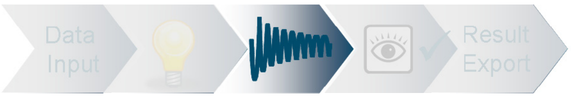


Figure 33: Diagram of relationship between 'Image Layer' and 'Image Object Level'.

Next step: Translate Strategy into Rule Set!





5.2 Translate strategy into Rule Set – use algorithm 'multiresolution segmentation'

This chapter covers the following content

→ *Why 'multiresolution segmentation' algorithm?*

→ *Creating objects using the multiresolution segmentation*

5.2.1 Why 'multiresolution segmentation' algorithm?

Information

The first crucial decision you have to make is which algorithm to use for creating objects. The initially created objects are the basis for all further analysis.

In the first lesson of this Guided Tour, the 'multiresolution segmentation' is chosen to get most "real-world" objects in a simple and rather knowledge-free way.

Simply said, the multiresolution segmentation groups areas of similar pixel values into objects. Therefore homogeneous areas result in larger objects, heterogeneous areas in smaller ones. How homogeneous/heterogeneous the objects are allowed to get is operated by the 'scale parameter'.

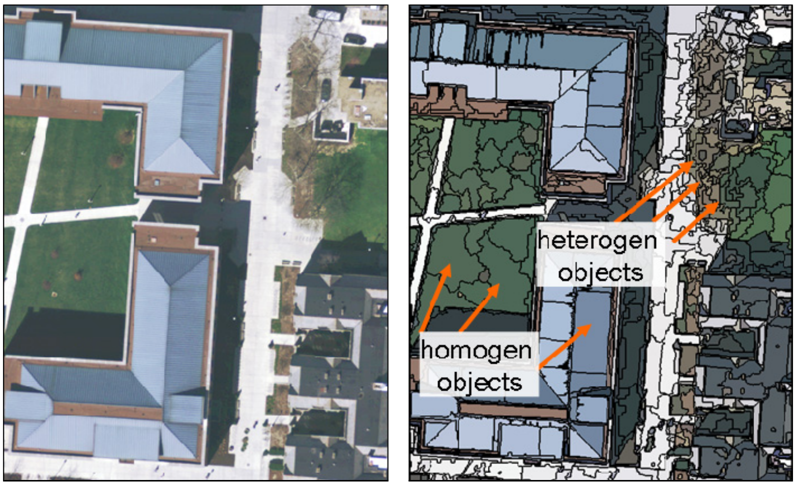


Figure 34: Left RGB layer mix, Right: image object outlines. Homogeneous areas result in bigger objects, heterogeneous areas in smaller.

Creating objects using the multiresolution segmentation

Information

Which layers to be used for creating objects?

The basis for creating image objects is the input-data. According to the data and the algorithm, objects result in different object shapes. The first to evaluate is which layers contain the important information.

Here in the current example there are two types of image data, the RGB and the DSM. In most segmentation algorithms it can be chosen whether to use all data available or only specific layers. This depends on where the important information is contained. Here all data is used for image object creation, the information contained in the RGB and in the DSM.

Which scale parameter to be set?

The ‘scale parameter’ is an abstract term. It is the restricting parameter to stop the objects from getting too heterogeneous.

In the current example we start with rather small objects to get the exact outlines of the buildings and to cover also the details (like windows and chimneys) as well as also the fractal structure of the leaf-off branches of the trees. For the ‘scale parameter’ there is no definite rule, you have to use trial and error to find out which ‘scale parameter’ results in the objects you need for further classification.

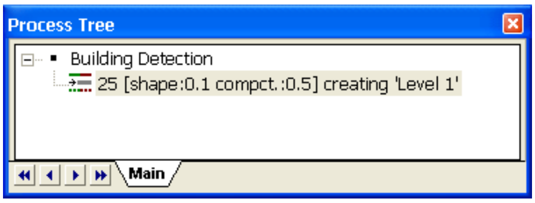


Figure 35: Process Tree with process to create image objects using the ‘multiresolution segmentation’ added.

.

5.2.2 Insert rule for object creation

This chapter covers the following content

→ *Insert a parent process as wrapper for all underlying processes*

→ *Insert the multiresolution segmentation process as a child process*

Information

In the 'View Settings' toolbar there are **4 predefined view setting**s available, specific to the different phases of a Rule Set development workflow.

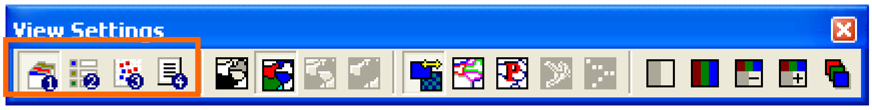


Figure 36: 'View Settings' toolbar with the 4 predefined view setting buttons: Load and Manage Data, Configure Analysis, Review Results, Develop Rulesets.

Action!

1. Select the predefined view setting number 4 '**Develop Rulesets**' from the 'View Settings' toolbar.

For the 'Develop Rulesets' view, per default one viewer window for the image data is open, as well as the '**Process Tree**' and the '**Image Object Information**' window, the '**Feature View**' and the '**Class Hierarchy**'.

Information

Insert a parent process as wrapper for all underlying processes

A so called 'parent process' is added to the Process Tree and will serve as a wrapper for all following processes. This has the advantage that you can execute the whole sequence from this process.

Action!

2. **Right-click** in the '**Process Tree**' window.

Result Check

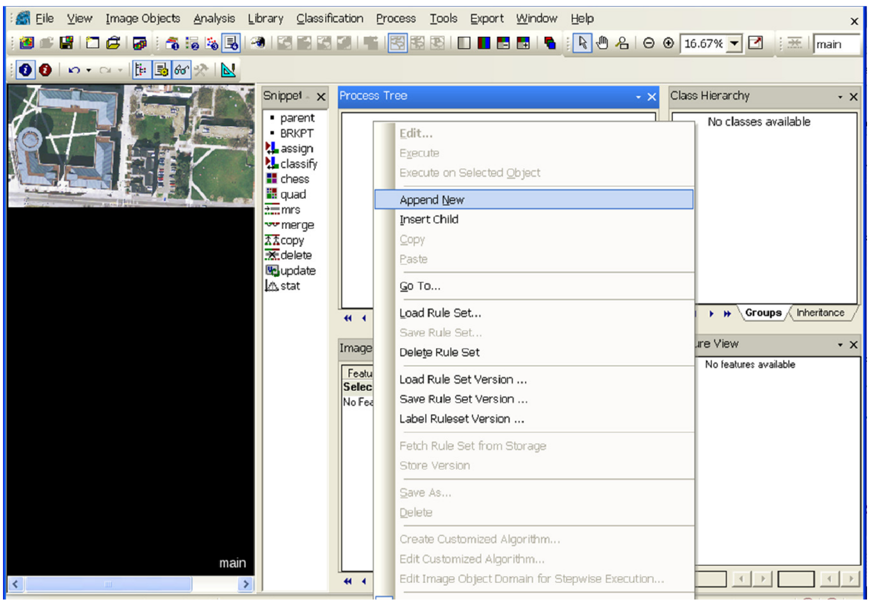


Figure 37If you right-click in the Process Tree window, a context menu appears.

Action!

3. Select '**Append New**' from the context menu.

The 'Edit Process' dialog box opens.

4. In the '**Name**' field enter the name '**Building Detection**' and confirm with '**OK**'.

> **Note:**
>
> For all 'parent processes' which serve only as a wrapper for underlying Processes, you can edit the name. For all Processes containing an algorithm, keep the auto naming!

Insert the multiresolution segmentation process as a child process

Information

According to the algorithm you choose, the parameter fields on the right side of the 'Edit Process' dialog box change.

For the 'multiresolution segmentation' you can edit the name of the level to be created, the 'scale parameter' and the so called 'homogeneity criterion'.

Action!

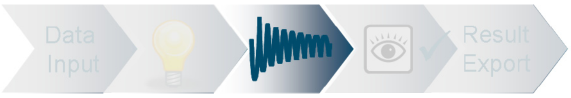
1. Select the inserted process and **right-click** on it. Choose '**Insert Child**' form the context menu.
2. In the '**Algorithm**' field choose (1) '**multiresolution segmentation**' from the list.

> **TIP:**
>
> Type in the 'Algorithm' field the first letters of the algorithm you want to use, a list of suitable algorithms is provided.

3. Keep 'pixel level' (2) in the 'Image Object Domain'.

> **Note:**
>
> For the first segmentation you can not change that. If you perform a second segmentation step, you can choose if you want to start from the domain 'pixel level' again or from an already existing image object Level.

4. In the field '**Level Name**' (2) type '**Level 1**'. This will be the name of the image object level to be created.
5. In the field '**Scale parameter**' (4) enter 25.
6. Confirm the settings with 'OK'. The process is now added to the Process Tree.



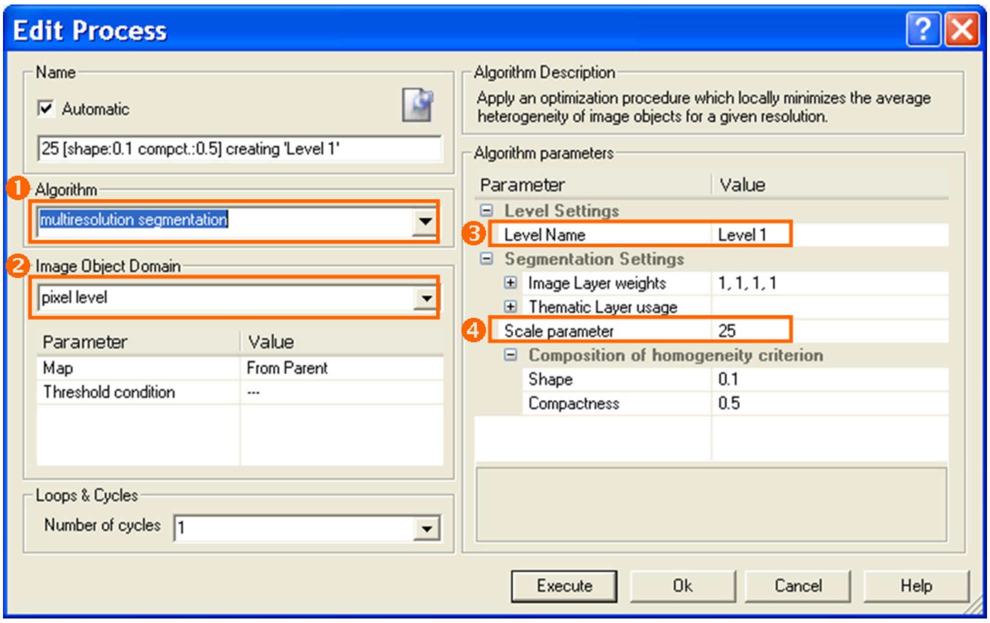


Figure 38: 'Edit Process' dialog box with settings to create 'Level 1' using the multiresolution segmentation algorithm.

Settings Check

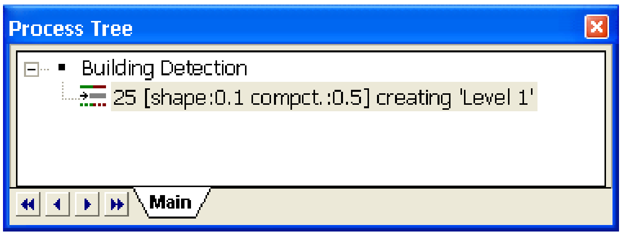


Figure 39: Process Tree with parent process 'Building Detection' and child process for creating 'Level 1'.

Rule Set Check

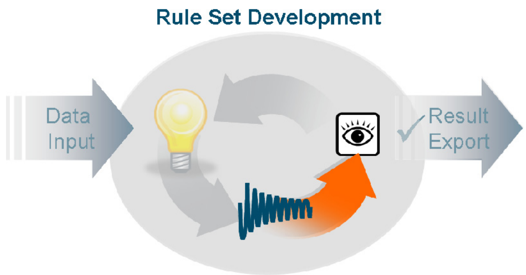
Execute the 'multiresolution segmentation' process

7. **Right-click** on the classification process and select '**Execute'** from the context menu. Alternatively press F5 on your keyboard.

Action!

An image object level 'Level1' is created according to the settings in the 'multiresolution segmentation' process.

Next step: Review Result!



5.3 Review the created image objects

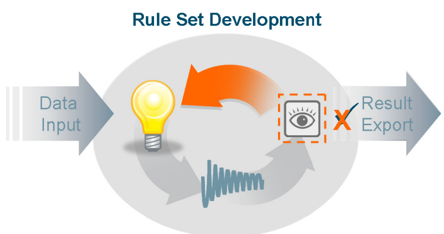
Result Check

The created objects represent the buildings even in their detailed outlines and also the roof parts.

Figure 40: Outlines of objects created with 'multiresolution segmentation' algorithm.

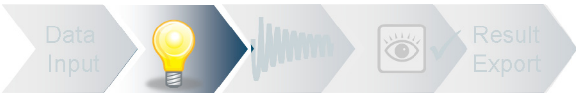
These objects will be the basis to classify in a first step all elevated objects in the scene.

Next step: Develop next analysis step!



Lesson 5 covered the following content

- → *Strategy for creating suitable image objects*
- → *Translate strategy into Rule Set – use algorithm 'multiresolution segmentation'*
- → *Review the created image objects*



Lesson 6 Initial classification: classifying all elevated objects

This lesson covers the following content

- *Strategy to classify buildings based on elevation information*
- *Translate the strategy into Rule Set– Object mean of DSM, algorithm 'assign class'*
- *Review the classification result– buildings and trees are classified*

6.1 Strategy to classify buildings based on elevation information

Introduction

As described in the chapter 'Evaluate the loaded data' the assumption is made that buildings are always elevated. The subset used for this Guided Tour has not much change in terrain elevation, so simply the definite elevation above sea level is used as a threshold to classify elevated objects.

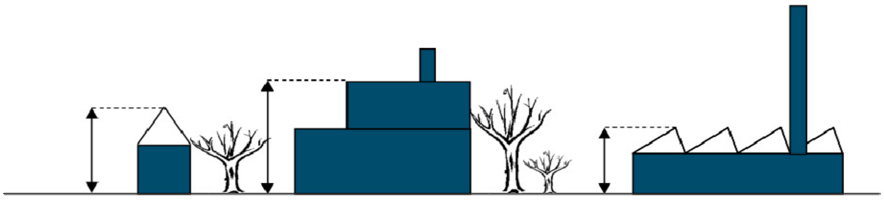
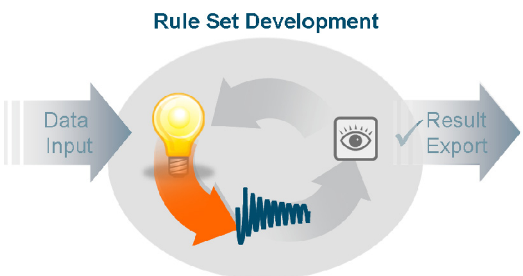


Figure 41: First step is to classify all elevated objects, buildings and trees.

Now this knowledge has to be transfered into the Rule Set!

Next step: Translate Strategy into Rule Set!



6.2 Translate the strategy into Rule Set– Object mean of DSM, algorithm 'assign class'

This chapter covers the following content

→ *Find the feature and the threshold to represent all elevated objects (mean value of DSM)*

→ *Write rule for classifying all elevated objects*

→ *Review the classification result– buildings and trees are classified*

6.2.1 Find the feature and the threshold to represent all elevated objects (mean value of DSM)

Information

Every object contains lots of feature values. There are:

- Object features like layer features, shape features and position features
- Class-related features: Context information about neighbors, sub objects below and super objects above
- Scene features: Features relative to scene, like the overall brightness, etc.
- Process-related features: Features with which you can expand the domain concept to deal only with individual objects, not the whole class or level
- Metadata: Deal with external, additional information, if available
- Feature variables : Used for defining individual names for features

For the initial classification of elevated objects, the mean value of the object concerning the DSM is used.

Use the 'Feature View' tool

Information

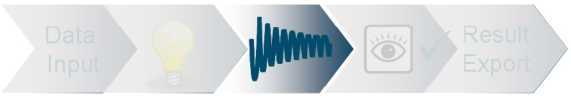
The most crucial part in Rule Set development is to find the optimal features and values for classifying image objects in the one or the other class.

The 'Feature View' is a tool that helps finding the **optimal features** and to determine **threshold values** for **classification**.

With the 'Feature View' the values for all objects are displayed in the viewer in **grey values**. But there is also the possibilities to show value ranges in **color**.

Note:

If not already open, there are several possibilities to open the 'Feature View' tool: Choose in the menu '**Tools>Feature View', alternatively** select the 'Feature View' button  from the '**Tool' toolbar**.



Select the feature

1. In the 'Feature View' window browse to '**Object features>Layer Values>Mean**'.

2. **Double-click** on 'DSM'.

3. Move your **cursor** over the objects in the viewer and the **exact feature values** for the object appears.

Action!

As you can see the **elevated objects** appear quite **bright**, this means they have **high values** for the feature 'Mean DSM'.

Result Check

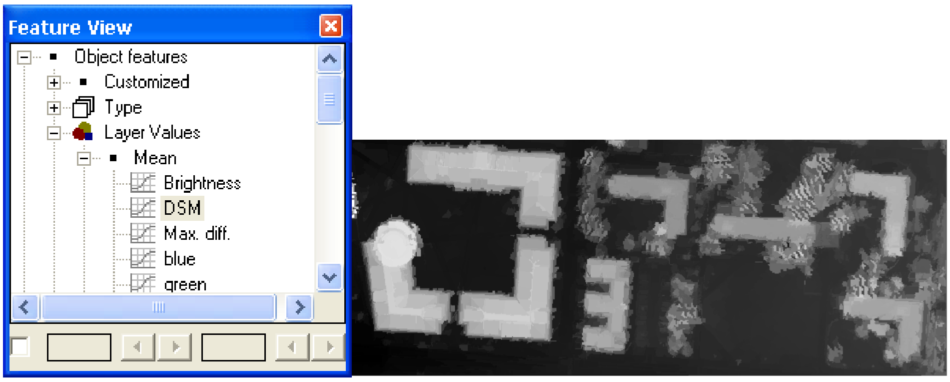


Figure 42: All objects appear now in **grey values** representing the **respective DSM value**.

But how to get the correct threshold to separate elevated and not elevated objects?

Update the feature range

Information

Besides moving the mouse over the objects and guessing a threshold value, you can use the **color displayed range** and so find the threshold value range.

First the whole range for the feature 'Mean DSM' has to be calculated internally (from minimum to maximum value).

1. **Select** the feature '**Mean DSM**' in the "Feature View" and **right-click** on it.

2. From the menu select '**Update Range**'.

Action!

Visualize a certain area of the whole feature range

1. Click the **check box** at the bottom of the 'Feature View' window.

Result Check

This activates the display of the feature range and the **minimum value** (here 737.92) is displayed in the left box and the **maximum value** (here 810.87) is displayed in the right box.

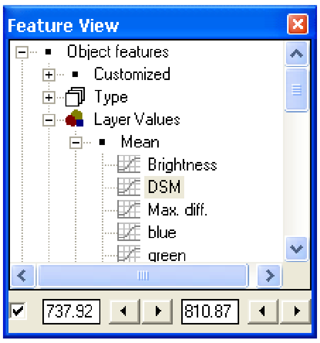


Figure 43: All image objects are **colored** in a smooth transition from **blue** (low values) to **green** (high values).

Information

With the **arrows** beside the value boxes you can **increase or decrease** the start and end of the range. All objects which are not within this range will be displayed in grey again.

Action!

2. To isolate the **high values (areas with high elevation)**, click the up arrow 1 to the right of the minimum value.

This will **increase the low end** of the range. Only objects within this new range are now displayed in color.

3. Continue until you reach the value **765** or type it in.

Result Check

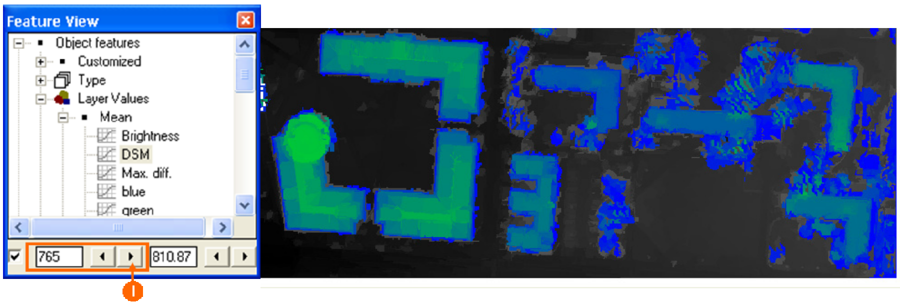


Figure 44: All image objects within a value range from 765 to 810 are **colored**, all objects outside this range (lower than 765) are displayed in grey values.

The rule derived from the feature and the values can be formulated like this: all objects with a mean elevation of more than 765 are possible buildings.

NOTE:

Be sure to update the range of feature values each time you select a different feature. Otherwise, the range of the recent feature is used.

6.2.2 Write rule for classifying all elevated objects

Information

Formulate the rule in words:

All objects with a mean elevation of more than 765 are buildings.

Prepare the Rule Set structure

Action!

1. In the Process Tree right-click on the last process and select '**Append New**' from the context menu.
2. Insert '**Classification of Buildings**' in the 'Name' field and confirm with OK.

3. In the Process Tree right-click on the 'Classification of Buildings' process and select '**Insert Child**' from the context menu.

4. Insert '**Classification based on elevation**' in the 'Name' field and confirm with 'OK'.

Rule Set Check

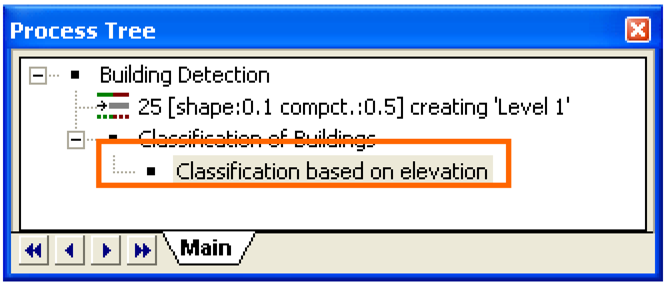


Figure 45: Process Tree with the two parent processes 'Classification of Buildings' and 'Classification based on elevation'.

Create a class in the 'Class Hierarchy' window

Action!

1. **Right-click** in the 'Class Hierarchy' window and select '**Insert Class**' from the context menu.

The 'Class Description' dialog box opens.

2. Enter 'Building' in the 'Name' field. Keep the default color.

3. Confirm with 'OK'.

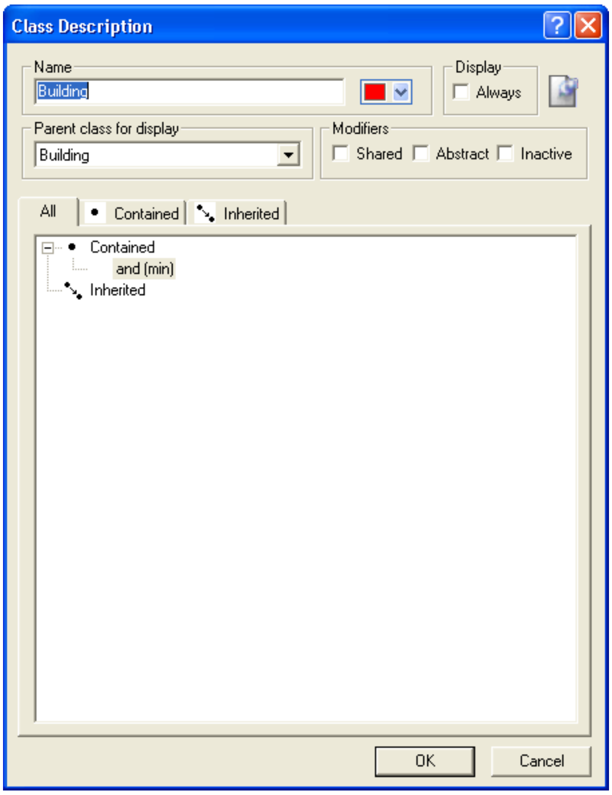


Figure 46: The 'Class Description' dialog box. The name is defined as 'Building'. Red is chosen as color.

The class 'Building' is added to the 'Class Hierarchy' window.

Result Check

Figure 47: Class Hierarchy with class 'Building' inserted.

Insert process to classify

Action!

1. **Right-click** on the process 'Classification based on elevation' and select '**Insert Child**' from the context menu.

Choose algorithm and object domain

2. Choose '**assign class**' 1 from the algorithm drop-down list.
3. Check if '**Level 1**' 2 is set as level domain
4. In the field 'Class Filter' keep '**none**' 3.

Define condition

5. Click in '**Threshold condition**' field and again click on the '**...**' 4 next to it.

The '**Select Single Feature**' dialog box opens.

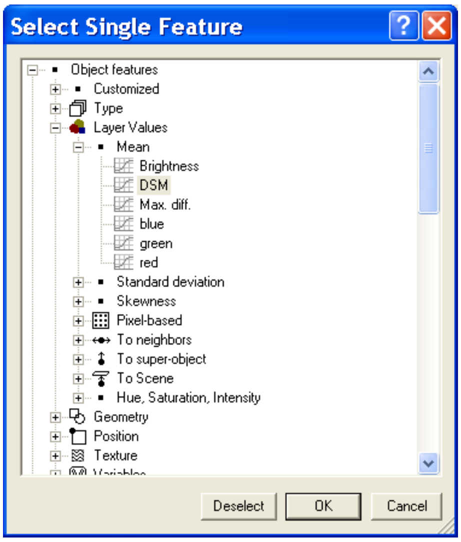


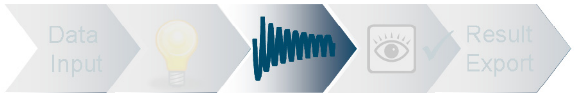
Figure 48: The 'Select Single Feature' dialog box.

6. Browse to '**Object features>Layer Values>Mean>DSM**' and double-click on it.
7. Choose the larger or equal than (**>=**) as operator and enter the value **765**.
8. Confirm with '**OK**'.

The '**Edit threshold condition**' dialog box opens.

Settings Check

The condition is added to the process.



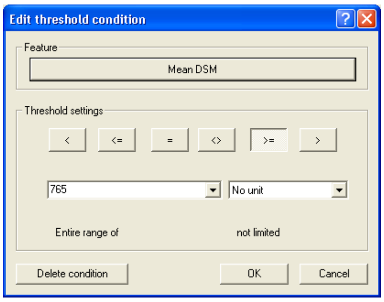


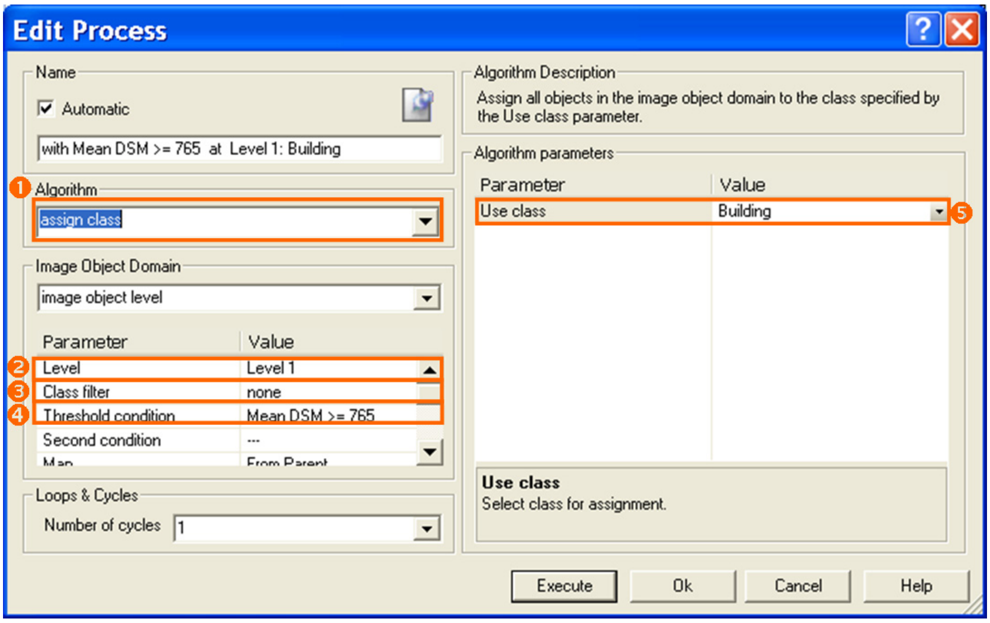
Figure 49: The 'Edit threshold condition' dialog box.

Define the target class

9. In the '**Algorithm Parameter**' section in the right pane, select '**Building**' 5 from the drop-down list.
10. Confirm the process settings with 'OK'.

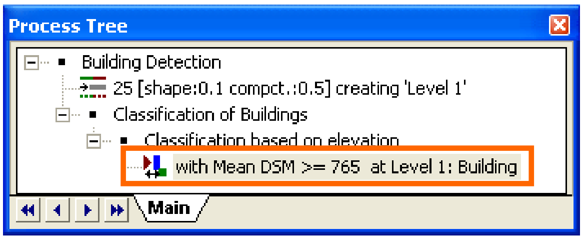
Action!

The process is now added to the Process Tree.



Settings Check

Figure 50: 'Edit Process' dialog box with settings to classify objects with 'Mean DSM' higher than 765 to the class 'Building'.



Rule Set Check

Figure 51: Process Tree with process for classification added.

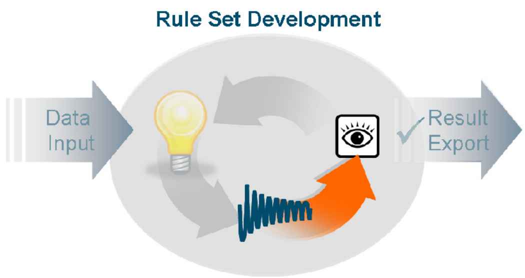
Execute the classification process

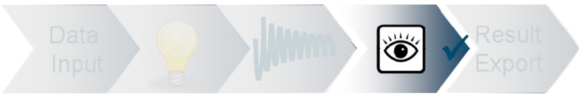
Action!

11. Right-click on the classification process and select ‘Execute’ from the context menu. Alternatively press F5 on your keyboard.

All objects fitting in the rule defined in the process are assigned to the class ‘Building’.

Next step: Review Result!





6.3 Review the classification result– buildings and trees are classified

Action!

1. In the 'View Settings' toolbar select the '**View Classification**' button and make sure that the 'Show or Hide Outlines' button is deselected.
2. Move the cursor over the classification and the assigned class will appear as a tool tip next to it.
3. Select the '**Pixel View or Objects Mean View**' button to switch on and off the transparency view.
4. Select the '**Show or Hide Outlines**' button and the outlines will be displayed in the classification colors.

Figure 52: Classification view not transparent, transparent and with outlines view switched on.

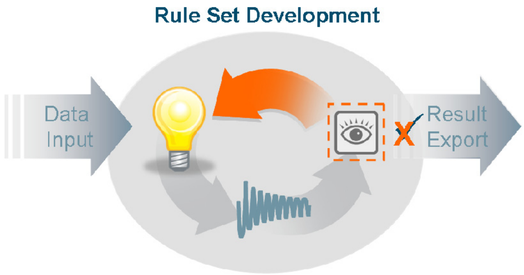
Result Check

All elevated objects are elevated. Besides the buildings there are also trees classified. This means add additional rules to refine the result must be added.

You have to develop a new strategy to find additional rules to minimize the misclassifications.

Figure 53: Classification result.

Next step: Develop next analysis step!



Lesson 6 covered the following content

- → *Strategy to classify buildings based on elevation information*
- → *Translate the strategy into Rule Set– Object mean of DSM, algorithm 'assign class'*
- → *Review the classification result– buildings and trees are classified*

Lesson 7 Refinement based on DSM: un-classifying trees

This lesson covers the following content

- → *Strategy to separate buildings from trees: use standard deviation of DSM*
- → *Translate the strategy into Rule Set – Standard deviation of DSM, algorithm 'assign class'*
- → *Review the classification result –trees are de-classified; some vegetation and shadows objects are still misclassified*

7.1 Strategy to separate buildings from trees: use standard deviation of DSM

Introduction

In a second processing step buildings have to be separated from trees, as both are elevated objects. Here again the DSM information will be used.

Evaluate object features for DSM to find a separating feature

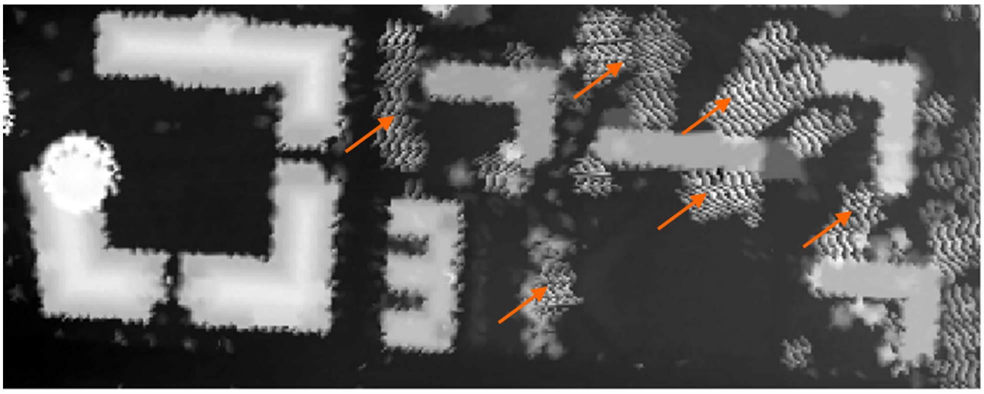
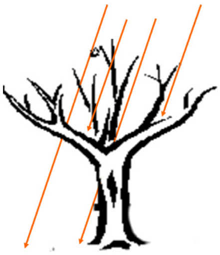


Figure 54: Image layer 'DSM'. Trees appear more heterogeneous than buildings.

If you evaluate the DSM image layer you can see that trees have very high elevation values close to very low elevation values. This is due to the leaf-off tree branches. The laser hits a branch and gives back a high elevation value, between the branches the laser goes through onto the ground, this results in a low elevation value. So it is characteristic that there are significant changes in elevation close to each other.



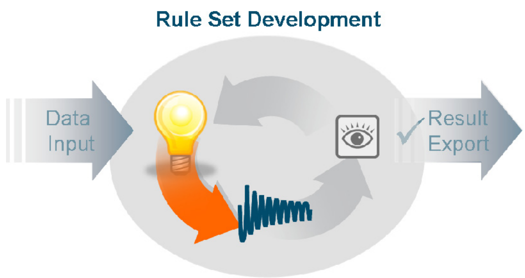
Introduction

Figure 55: The laser hits one time the branch, the other time it goes onto the ground. Due to this effect, the leaf-off trees appear heterogeneous.

Which feature represents the characteristics of leaf-of trees?

The features of the category 'Mean' will not help here. A very suitable feature to describe heterogeneity or homogeneity is the 'standard deviation'.

Next step: Translate Strategy into Rule Set!



'.

7.2 Translate the strategy into Rule Set – Standard deviation of DSM, algorithm 'assign class'

This chapter covers the following content

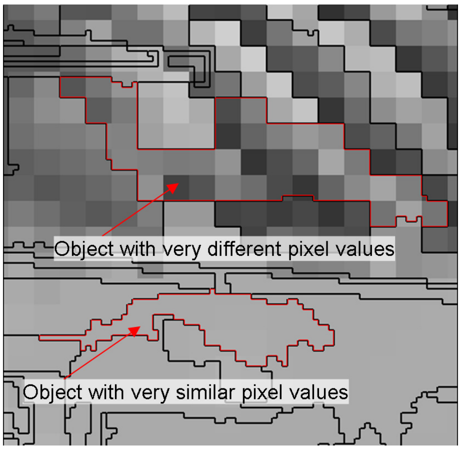
- → *Find the feature and the threshold to separate buildings from trees (Stddev. of DSM)*
- → *Write rule to separate buildings from trees*
- → *Review the classification result –trees are de-classified; some vegetation and shadows objects are still misclassified*

7.2.1 Find the feature and the threshold to separate buildings from trees (Stddev. of DSM)

Information

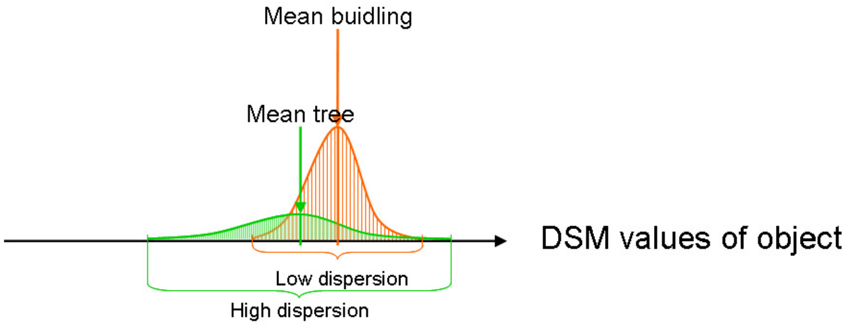
The feature 'standard deviation of DSM'

The standard deviation of pixel values of an object gives back information whether the pixel values of an object are all quite similar or if there are significant differences.



For the leaf-off trees high differences in elevation is expected, the standard deviation of these areas should be high. In opposite, the building surfaces should give back low values for standard deviation.

Information



The pixels of building objects are close around the mean value of the object, this means that the dispersion is low. In opposite the pixels of a tree object have a wide dispersion around the mean of the object.

Find threshold using the 'Feature View' tool

Action!

1. Browse to the feature '**Object features>Layer Values>Standard deviation>DSM**'.
2. Right-click on it and select 'Update Range' from the context menu.
3. Switch on the check box at the bottom of the 'Feature View' window.
4. Increase the lower range to the value 6. Most of the objects which are actually trees have values above 6, most of the building objects have values below.

Rule Set Check

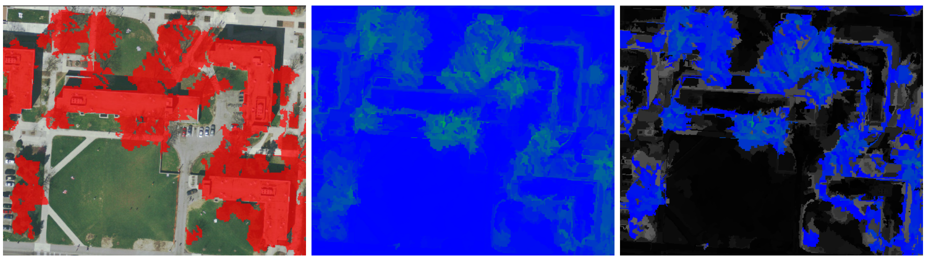


Figure 56: Left: classification view; Middle: whole range of the standard deviation of DSM; Right: high standard deviation values are highlighted in color.

The trees appear in green; this indicates high values for the standard deviation of DSM. If you increase the low values, only the trees remain colored.

7.2.2 Write rule to separate buildings from trees

Introduction

Formulate the rule in words:

Unclassify all 'Building' objects with standard deviation above 6.

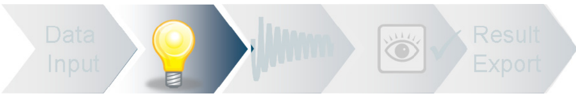
Only one condition is used, so again use the algorithm 'assign class'.

Action!

1. Select the last process in the Process Tree, right-click and select '**Append New**' from the context menu.

Choose algorithm and object domain

2. Choose '**assign class**' from the algorithm list.



3. Click in the field 'Class Filter' and again click on the '…' button next to it. Select '**Building**' as domain by switching on the check-box next to it.

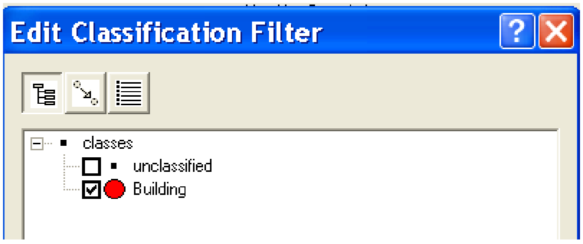


Figure 57: The class 'Building' is selected.

Tip:

If you want to change to the list view in the 'Edit Classification Filter' dialog box, simply select the last button. Then you can select the classes by double-clicking

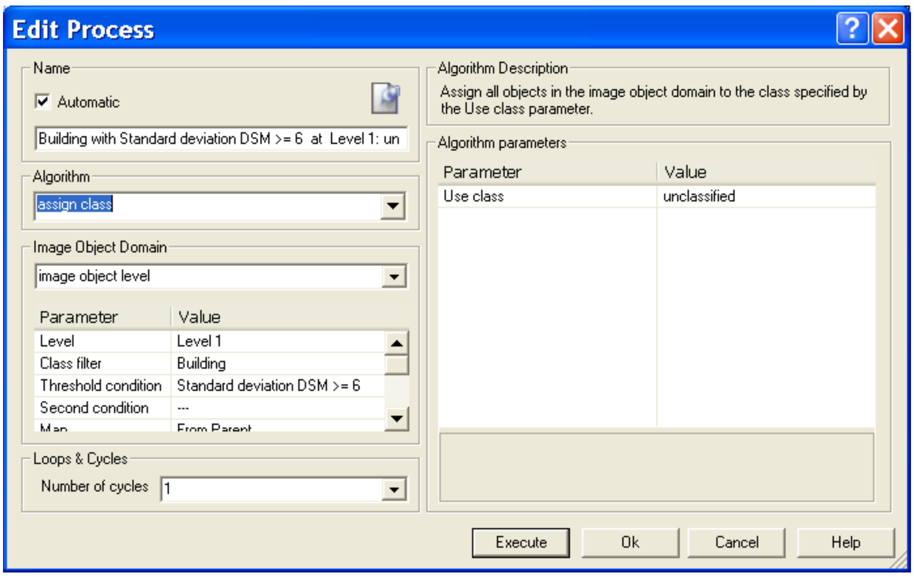
Define condition

4. Click in '**Threshold condition**' field and again click on the '**…**' next to it.
5. Browse to '**Object features>Layer Values>Standard deviation>DSM**' and double-click on it.
6. In the 'Edit threshold condition' choose the **larger or equal than (>=)** as operator and enter the value **6**.
7. Confirm with '**OK**'.

Define the target class

8. In the 'Parameter' section in the left pane, keep '**unclassified**'.
9. Confirm the process settings with 'OK'.

The process is now added to the Process Tree.



Settings Check

Figure 58: 'Edit Process' dialog box with settings to un-classify all 'Building' objects with a standard deviation of DSM larger than 6.

Rule Set Check

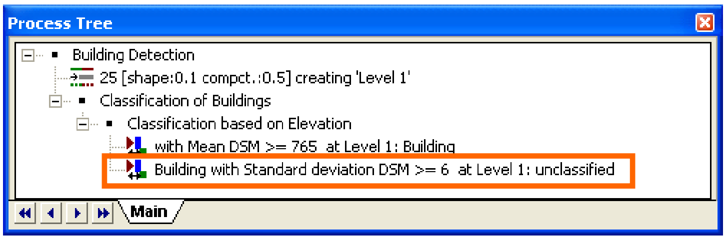


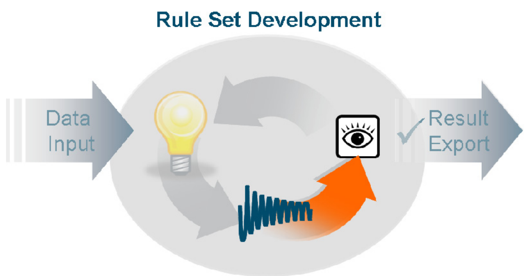
Figure 59: Process Tree with second process for classification added.

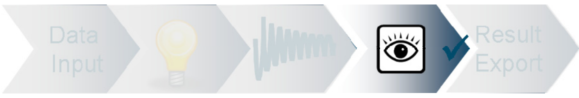
Execute the classification process

Action!

10. Right-click on the process and select '**Execute**' from the context menu. Alternatively press F5 on your keyboard.

Next step: Review Result!





7.3 Review the classification result – trees are de-classified; some vegetation and shadows objects are still misclassified

Result Check

All 'Building' objects with a value higher than 6 for the feature 'standard deviation of DSM' are unclassified again. But still there are too many objects classified as 'Building'. The classification is not ready for export!

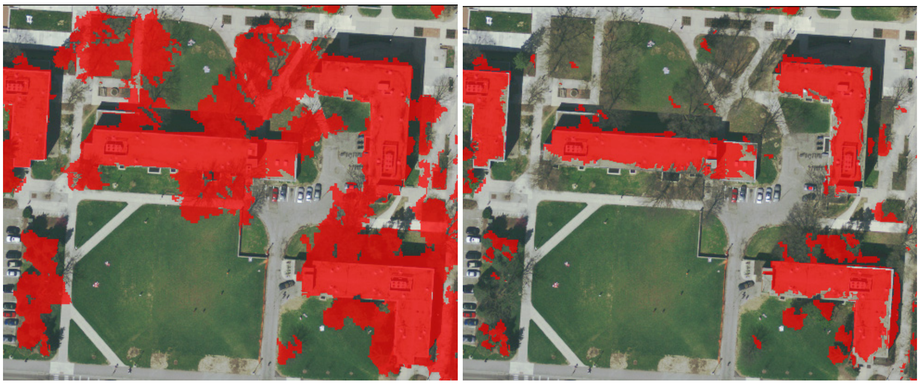


Figure 60: Left: Classification before refinement; Right: Classification after refinement.

This means you have to go back to the 'Develop Strategy' stage to find additional rules to minimize the misclassifications.

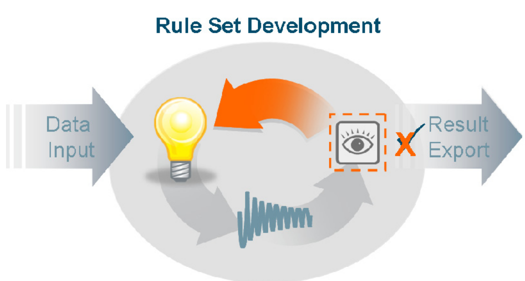
Lesson 7 covered the following content

→ *Strategy to separate buildings from trees: use standard deviation of DSM*

→ *Translate the strategy into Rule Set – Standard deviation of DSM, algorithm 'assign class'*

→ *Review the classification result –trees are de-classified; some vegetation and shadows objects are still misclassified*

Next step: Translate Strategy into Rule Set!



Lesson 8 Refinement based on spectral information

This lesson covers the following content

- *Strategy to refine buildings based on spectral information – the spectral layer ratio*
- *Translate the strategy into Rule Set –Ratio of green, algorithm 'assign class'*
- *Review the classification result – the vegetated areas are de-classified; small objects still misclassified*

8.1 Strategy to refine buildings based on spectral information – the spectral layer ratio

Introduction

After refining the classification using the standard deviation of the DSM layer still some trees are classified as 'Building'. Some of them are coniferous trees, so they are not leaf-off and therefore give back a quite homogeneous elevation area.

But these coniferous trees have a significant spectral difference to buildings. Roofs are usually not covered by vegetation; therefore a feature must be found that represents vegetation in a stabile way.

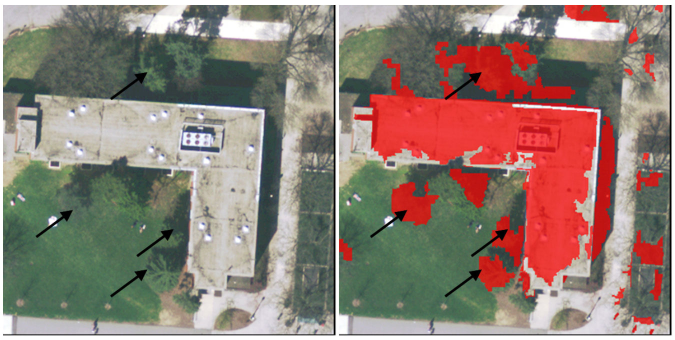
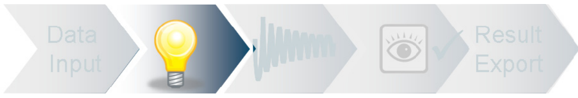


Figure 61: The black arrows are pointing to some trees not de-classified with the previous step.

Figure 62: RGB layer mixing

It is obvious that the green layer contains significant information about vegetation, but only in comparison with the other 2 layers.



Introduction

Figure 63: The individual layers, red, green, blue

One way of comparing image layers is to create a ratio. The relevant ratio would be "green/(red+green+blue)".

This knowledge must be transfered into the Rule Set now.

Outlook on Process Tree:

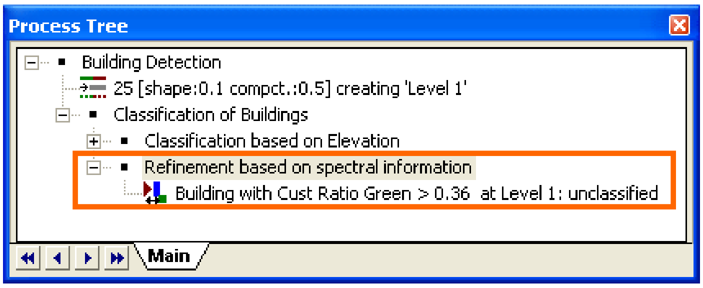
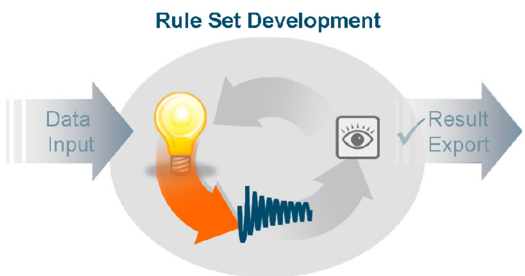


Figure 64: Process Tree processes added to refine classification based on spectral feature.

Next step: Translate Strategy into Rule Set!



8.2 Translate the strategy into Rule Set –Ratio of green, algorithm 'assign class'

This chapter covers the following content

→ *Find the feature and the threshold for spectral refinement (ratio green)*

→ *Translate strategy in a rule for refinement based on a spectral feature*

→ *Review the classification result – the vegetated areas are de-classified; small objects still misclassified*

8.2.1 Find the feature and the threshold for spectral refinement (ratio green)

Information

Within eCognition Developer there are standard features, always listed in the feature view, but there is also the possibility to **customize features** for your specific needs. You can create, save and load such 'Customized Features.' Here in this example a feature is needed which expresses the ratio: "**green/(red+green+blue)**".

Note:

The software provides a standard ratio calculation. These ratio features are calculated on basis of all inserted layers. This means in our case that the DSM will be included in the calculation.

Load a customized feature

Action!

1. Right-click in the 'Feature View' and select '**Load**'.
2. Browse to the folder 'RuleSets' in your Guided Tour directory and select '**CustRatioGreen.duf**'.

The customized feature is added to the 'Feature View'

3. Browse to '**Object Features>Customized**'.

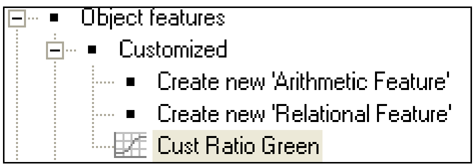


Figure 65: The customized feature listed in the 'Feature View'.

4. **Right-click** on the feature 'CustRatioGreen' and select '**Edit**'.

The 'Edit Customized Feature' dialog box opens.

Check the calculation for the customized feature

5. Check the correct calculation: [Mean green]/([Mean blue]+[Mean green]+[Mean red]).

The mean value for the layer green of the object is compared to the overall brightness of the object.

Result Check

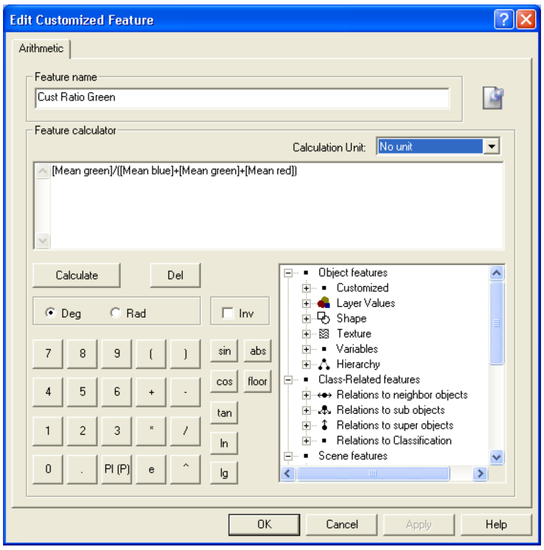


Figure 66: The loaded customized feature in the 'Edit Customized Feature' dialog box.

6. Close the dialog by clicking 'OK'.

Action!

Find threshold to separate the class 'Vegetation'

1. Browse to the feature '**Object features>Customized>CustRatioGreen**' and double-click on it.

Result Check

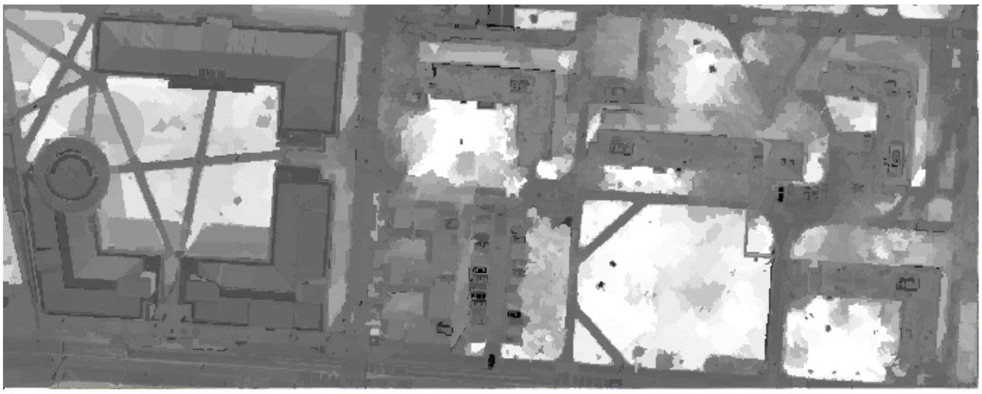


Figure 67: 'Feature View' of the customized feature 'Cust Ratio Green'

The vegetated areas appear very bright, this means the objects have high values. The amount of green in comparison to the other image layers is here dominant.

Action!

2. Right-click on it and select '**Update Range**' from the context menu.
3. Switch on the **check box** at the bottom of the 'Feature View' window.
4. **Increase the lower range** to the value 0.36. Most of the vegetated areas have values above 0.36.

Result Check

Figure 68: RGB layer mixing.

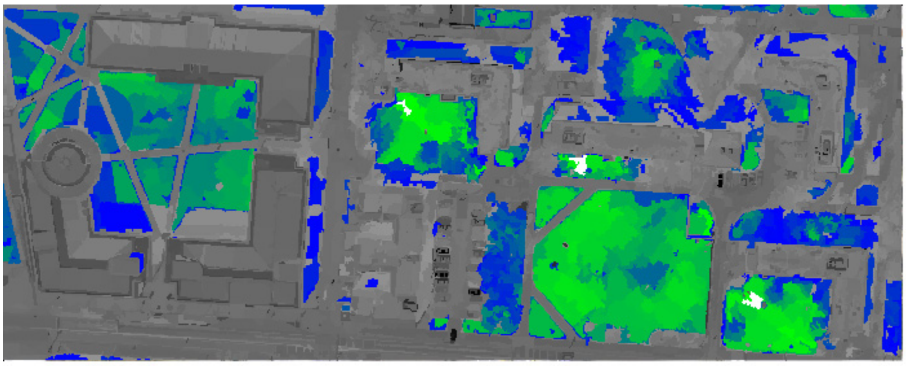


Figure 69: 'Feature View' range (above 0.36) of the customized feature 'Cust Ratio Green'

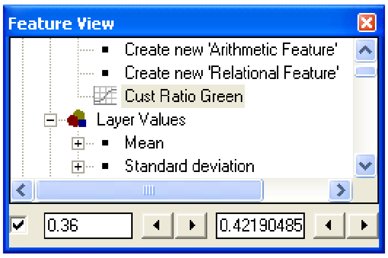


Figure 70: Vegetation is represented by values higher than 0.36 for the feature 'CustRatioGreen'.

As a rule from the feature and the range it can be formulated that:

Unclassify all 'Building' objects with a value for customized ratio green of more than 0.36.

8.2.2 Translate strategy in a rule for refinement based on a spectral feature

Prepare the Rule Set structure

Action!

1. In the Process Tree right-click on the last process and select '**Append New**' from the context menu.
2. Insert '**Refinement based on spectral information**' in the 'Name' field and confirm with 'OK'.

Insert process to classify

1. In the Process Tree right-click on the 'Refinement based on spectral information' and select '**Insert Child**' from the context menu.

Choose algorithm and object domain

Again only one condition for classification is used, so the algorithm 'assign class' is appropriate.

2. Choose '**assign class**' from the algorithm list.
3. As 'Class Filter' select '**Building**'.

Define condition

4. Click in '**Threshold condition**' field and again click on the '**...**' next to it.
5. Browse to '**Object features>Customized>Cust Ratio Green**' and double-click on it.
6. In the 'Edit threshold condition' choose the **larger or equal than (>=)** as operator and enter the value **0.36**.
7. In the field 'Active class' keep '**unclassified**'.

Confirm the settings

8. Confirm with '**OK**'.

Settings Check

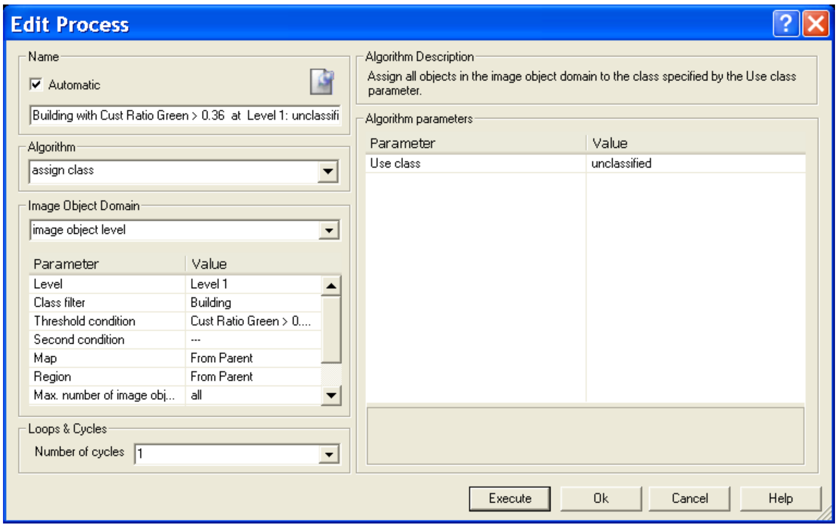


Figure 71: 'Edit Process' dialog box with settings to un-classify 'Building' objects with a value for the customized feature higher than 0.36.

Rule Set Check

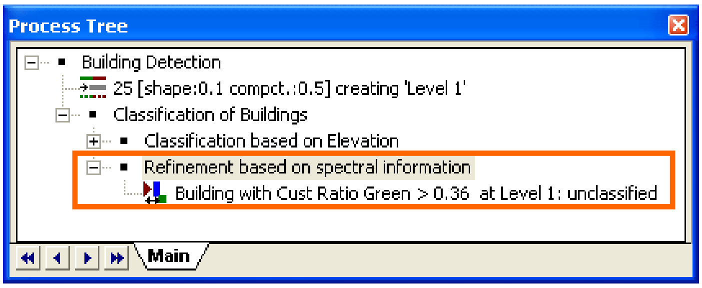


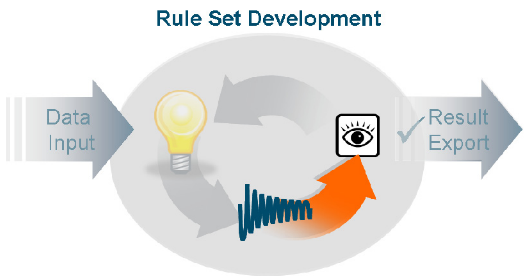
Figure 72: Process Tree with parent process 'Refinement based on spectral information' added and with child processes for classification.

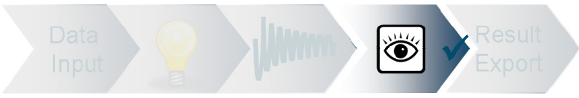
Execute the classification process

Action!

9. Right-click on the process and select '**Execute**' from the context menu. Alternatively press F5 on your keyboard.

Next step: Review Result!





8.3 Review the classification result – the vegetated areas are de-classified; small objects still misclassified

All 'Building' objects with a value higher than 0.36 for the feature 'Cust Ratio Green' are now unclassified again.

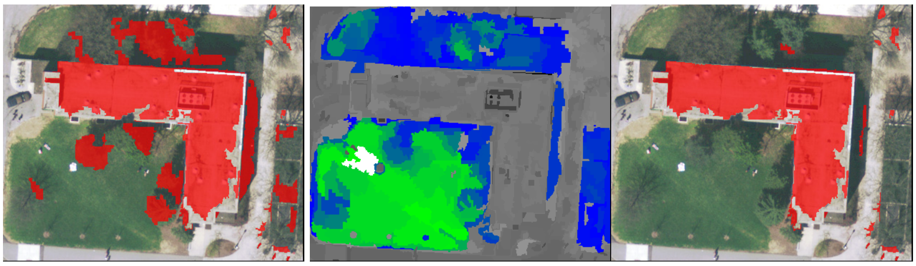


Figure 73: Left: Classification view before refinement; Middle: Feature View for Ratio Green; Right: Classification after refinement.

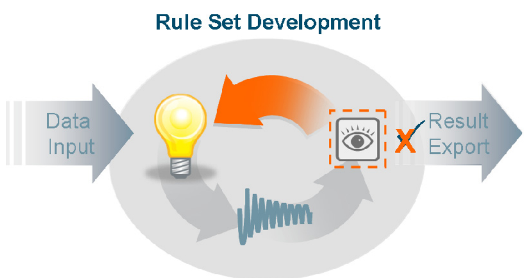
But still there are objects misclassified as 'Building' and some building areas are not yet classified. The classification is not ready for export yet!

This means the strategy has to be refined, additional rules must be added to minimize the misclassifications.

Lesson 8 covered the following content

- *Strategy to refine buildings based on spectral information – the spectral layer ratio*
- *Translate the strategy into Rule Set –Ratio of green, algorithm 'assign class'*
- *Review the classification result – the vegetated areas are de-classified; small objects still misclassified*

Next step: Develop next analysis step!



Lesson 9 Refinement based on context

This lesson covers the following content

- → *Strategy to refine buildings based on context information –surrounding neighbor objects*
- → *Translate Strategy in a rule for refinement based on a context feature*
- → *Review the classification result – only small objects remain misclassified*

9.1 Strategy to refine buildings based on context information – surrounding neighbor objects

Introduction

Some areas of the buildings have not been classified yet or were de-classified again because they fulfilled one of the conditions before.

Some of the not classified objects are highly surrounded by 'Building' objects. If an unclassified object has a high common border to 'Building' objects it should also belong to the class 'Buildings'.

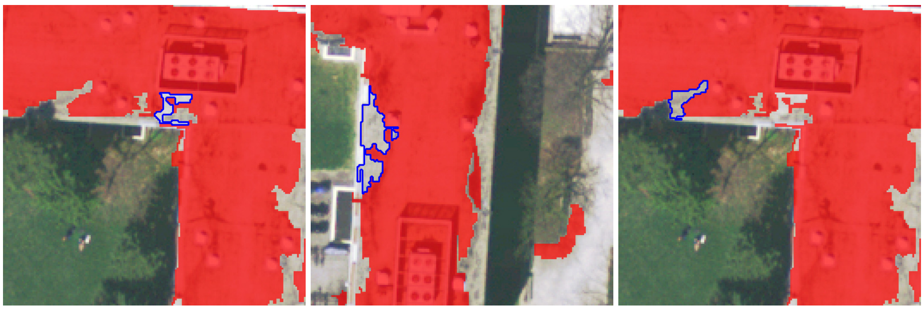


Figure 74: Some unclassified objects are highly surrounded by 'Building' objects.

This knowledge must be transferred into the Rule Set now.

Within eCognition Developer you can express neighborhood relationships of objects using the 'Class-Related' (context) features. This is possible within the image object level, for super-objects above or sub-objects below.

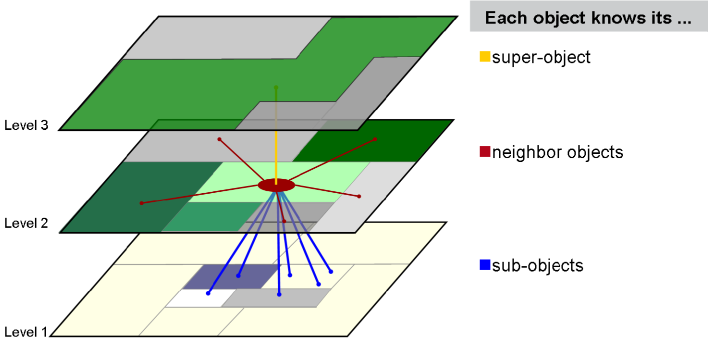


Figure 75: Diagram of 'Image Object Hierarchy' and network.

In the current example neighborhood within the same level will be analyzed.

The rule must describe the situation that if an unclassified object has a high common border to 'Building' objects it belongs to the class 'Building' too.

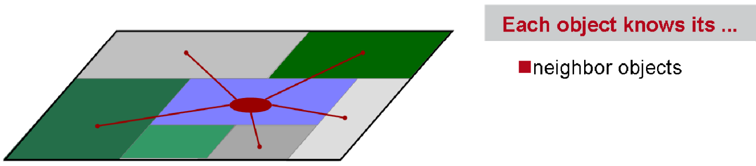


Figure 76: All objects have neighborhood relationships, which can be used as information for classification.

This knowledge must be transferred into the Rule Set now.

Outlook on Process Tree:

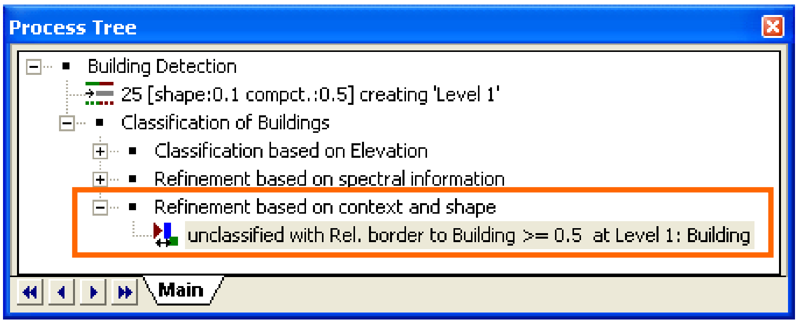
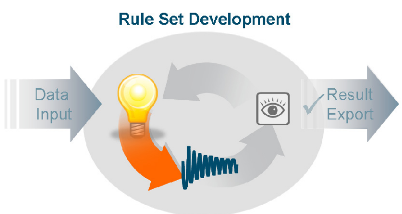


Figure 77: Process Tree with processes added for classification based on context information.

Next step: Translate Strategy into Rule Set!



9.2 Translate Strategy in a rule for refinement based on a context feature

This chapter covers the following content

- → *The class-related feature 'Relative border to'*
- → *Prepare the Rule Set structure*
- → *Insert process to classify*
- → *Review the classification result – only small objects remain misclassified*

9.2.1 The class-related feature 'Relative border to'

Information

The class-related feature 'Relative border to' describes the amount of the overall border of an object compared to the border to a specific class.

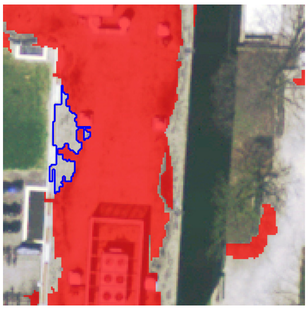


Figure 78: The selected object (blue outlines) has a border to 'Buildings' about 0.6 and a border to 'unclasssified' about 0.4

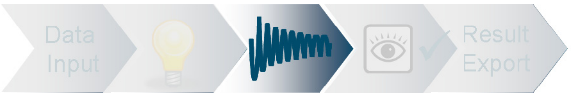
The range of the feature reaches from **0 to 1**.

- Objects with no border to 'Building' objects have the value 0
- Objects with a low common border to 'Building' objects have low values
- Objects with a high common border to 'Building' objects have high values
- Objects which are completely surrounded by 'Building' objects have the value 1

Create the feature 'Relative border to 'Buildings''

Action!

1. In the 'Feature View' window browse to '**Class-Related features>Relations to neighbor objects> 'Relative Border to**'.



Note:

Per default the list of class-related features is empty. You first have to create the class you want to evaluate. If you had a lot of classes and all relationships to all classes would be displayed always, this would be very confusing. This is why you have to create the individual relationships on demand.

2. **Double-click** on 'Create new 'Rel. border to''.

The 'Create Rel. border to' dialog box opens.

3. From the 'Value' drop-down list select the class '**Building**', as the border to 'Building' objects must be evaluated.

4. Confirm the settings with 'OK'.

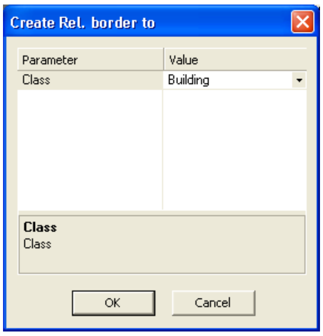


Figure 79: The 'Create Rel. border to' dialog box with the class 'Building' selected.

Settings Check

The feature is added to the 'Feature View' list.

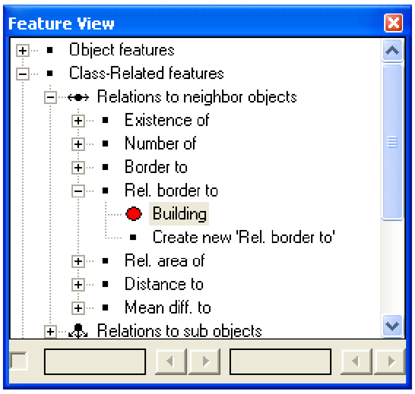


Figure 80: Feature View with new feature 'Rel. border to 'Building'' created.

Result Check

Find the threshold

1. Double-click the feature '**Class-Related features>Relations to neighbor objects> 'Relative Border to 'Building'**'.

2. Right-click on it and select '**Update Range**' from the context menu.

3. Switch on the **check box** at the bottom of the 'Feature View' window.

4. **Increase the lower range** to the value **0.5**.

Action!

Result Check

Objects appearing in a color range form blue to green, have at least half of its overall border common with 'Building' objects. Objects with a lower common border to 'Building' objects appear now in a grey range.

If the value is too low, too many neighboring objects would be included.

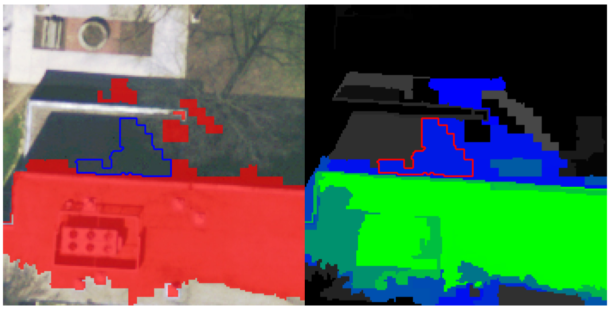


Figure 81: The selected object has a Rel. border to 'Buildings' of 0.3.

If the value is too high, too few neighboring objects would be included.

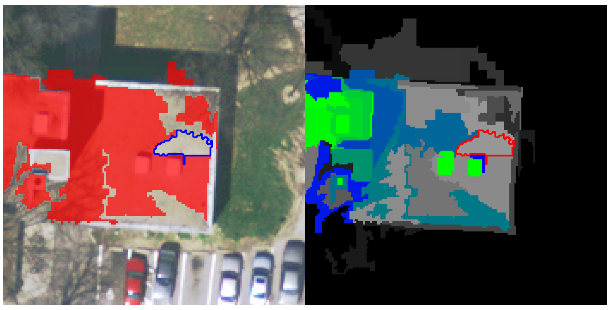


Figure 82: The selected object has a relative border to 'Buildings' of 0.6.

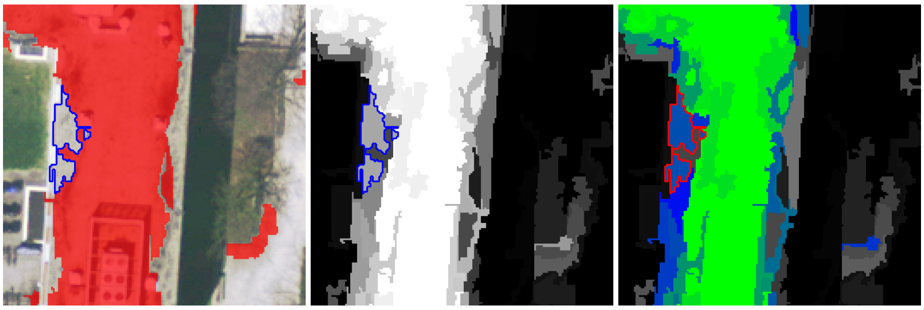


Figure 83: The selected object has a Rel. border to 'Buildings' of 0.5.

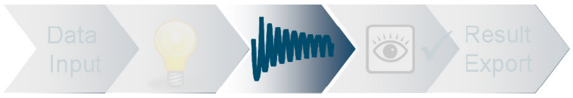
As a rule from the feature and the range it can be formulated that:

Unclassified objects with a relative border to 'Building' objects **more than 0.5** shall be classified as 'Building' too.

9.2.2 Prepare the Rule Set structure

Action!

1. In the Process Tree right-click on 'Refinement based on spectral information' and select 'Append New' from the context menu.
2. Insert 'Refinement based on context information' in the 'Name' field and confirm with 'OK'.



9.2.3 Insert process to classify

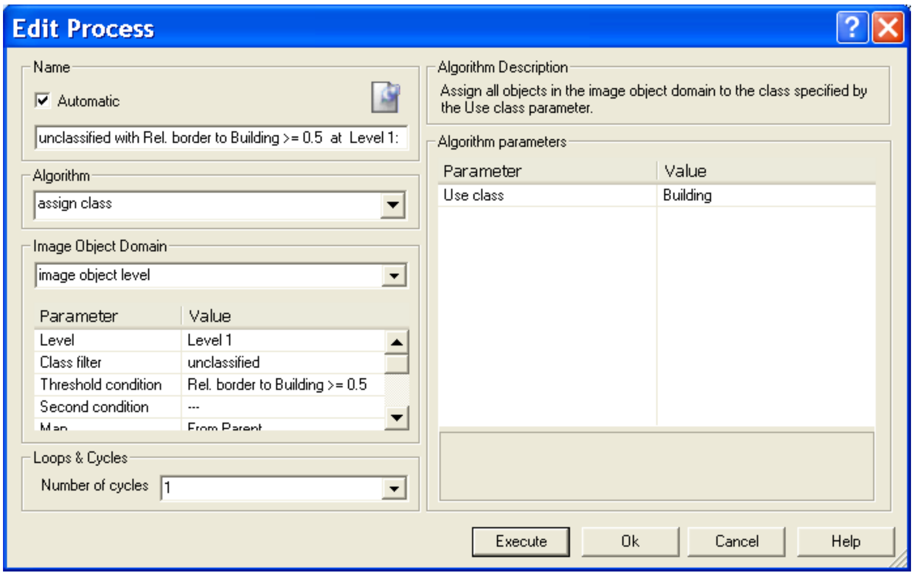
1. In the Process Tree right-click on the 'Refinement based on context information' and select 'Insert Child' from the context menu.

Choose algorithm and object domain

2. Choose '**assign class**' from the algorithm list.
3. Define 'unclassified' as 'Class filter'.

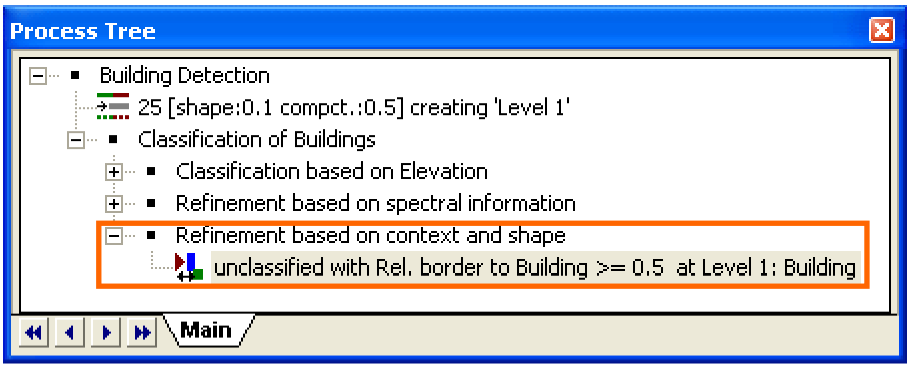
Define condition

4. Click in the 'Threshold condition' field and browse to '**Class-Related features>Relations to neighbor objects> 'Rel. border to Building**' and double-click on it.
5. In the 'Edit threshold condition' choose the **larger or equal than (>=)** as operator and enter the value **0.5**.
6. Confirm with '**OK**'.



Settings Check

Figure 84: 'Edit Process' dialog box with settings to classify all 'unclassified' objects with a high common border to 'Buildings' as 'Buildings' too.



Rule Set Check

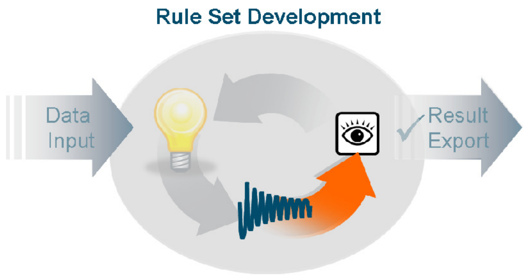
Figure 85: Process Tree with parent process 'Refinement based on context and shape' added and with child processes for classification.

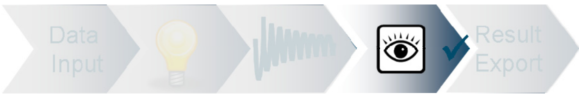
Execute the classification process

Action!

7. Right-click on the process and select '**Execute**' from the context menu. Alternatively press F5 on your keyboard.

Next step: Review Result!





9.3 Review the classification result – only small objects remain misclassified

Unclassified objects with a relative border to 'Building' objects more than 0.5 are classified as 'Building' too.

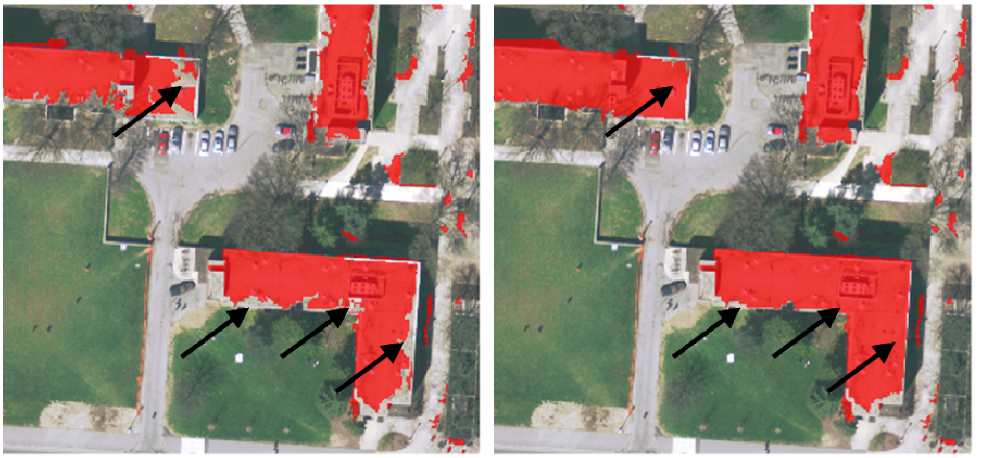
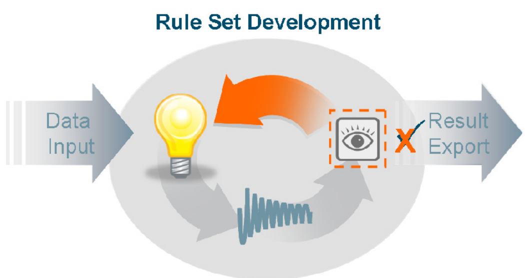


Figure 86: Classification view before and after the refinement.

But still there are too many misclassifications. The classification is not ready for export!

This means the development strategy has to be refined to find additional rules to minimize the misclassifications.

Next step: Develop Strategy for fist analysis step!



Lesson 9 covered the following content

→ *Strategy to refine buildings based on context information –surrounding neighbor objects*

→ *Translate Strategy in a rule for refinement based on a context feature*

→ *Review the classification result – only small objects remain misclassified*

Lesson 10 Refinement based on shape

This lesson covers the following content

- → *Strategy to refine buildings based on shape information – generalize objects, separate them by size*
- → *Translate strategy in a rule for refinement based on a shape feature*
- → *Review the classification result*

10.1 Strategy to refine buildings based on shape information – generalize objects, separate them by size

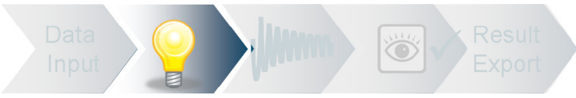
Introduction

There are still some misclassified objects, but they are very small. The size of the 'object of interest' is the separating condition here, based on the assumption, that buildings have a certain size.

Figure 87: Outlines of unmerged objects.

As all objects are now still as small as they were created by the initial multiresolution segmentation, they have to be merged first according to their classification. After that the area feature can be used to separate the big buildings from the small misclassified objects.

Figure 88: Outlines of merged 'Building' objects.



This means you have to add two rules:

Rule 1) Merge all 'Building' objects.

Rule 2) Unclassify all too small 'Building' objects.

This knowledge must be transferred into the Rule Set now.

Outlook on Process Tree:

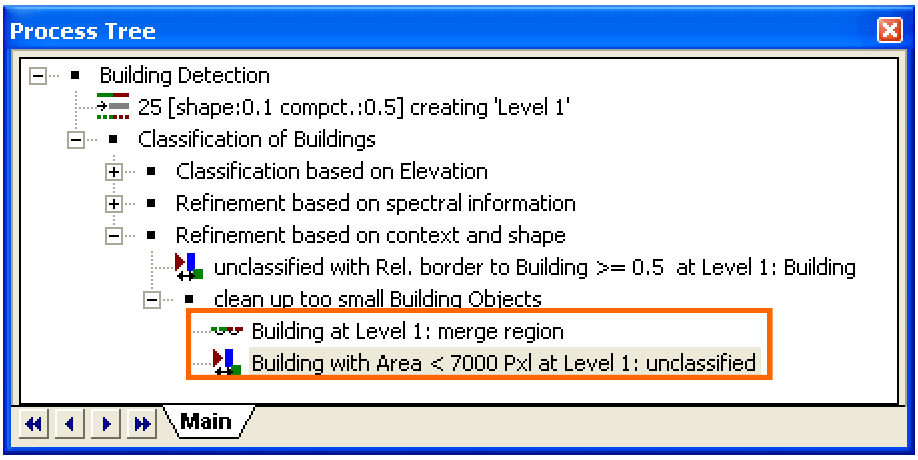
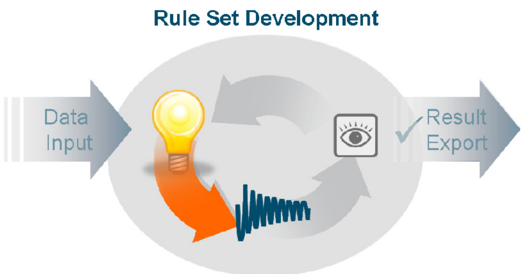


Figure 89: Process Tree with process added to merge all 'Building' objects and to un-classify too small 'Building' objects.

Next step: Translate Strategy into Rule Set!



10.2 Translate strategy in a rule for refinement based on a shape feature

This chapter covers the following content

- → *Merge the image objects*
- → *Find the feature and the threshold for refinement based on shape*
- → *Translate strategy in a rule for refinement based on the 'Area' feature*
- → *Review the classification result*

10.2.1 Merge the image objects

Information

eCognition Developer provides several algorithms to merge image objects. The most simple one is the algorithm '**merge region**'. With this algorithm you can merge **neighboring objects** according to their class.

Prepare the Rule Set structure

Action!

1. In the Process Tree right-click on the last process and select '**Append New**' from the context menu.
2. Insert '**clean up too small Building objects**' in the 'Name' field and confirm with 'OK'.

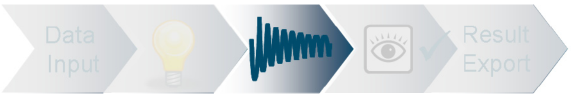
Insert process to merge 'Building' objects

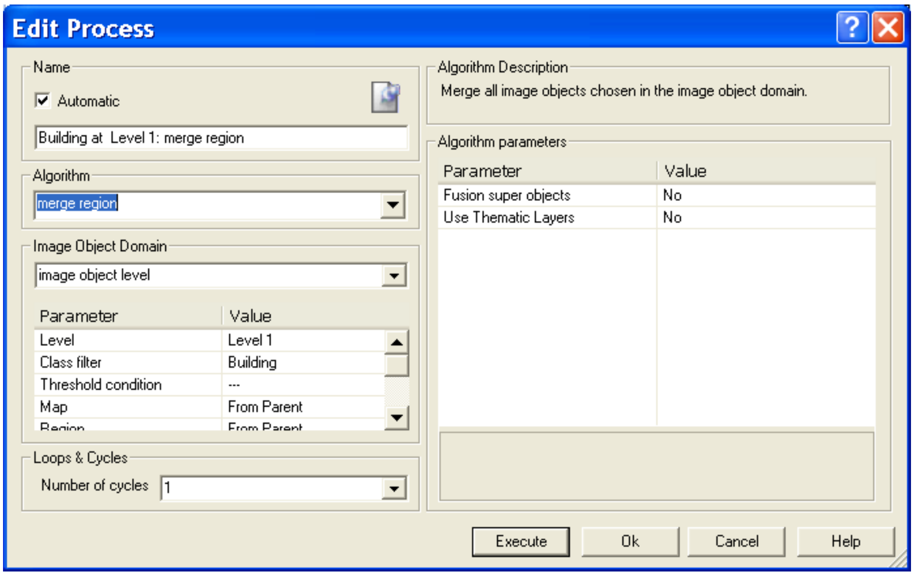
1. In the Process Tree **right-click** on the 'Clean up too small Building objects' and select '**Insert Child**' from the context menu.

Choose algorithm and object domain

Action!

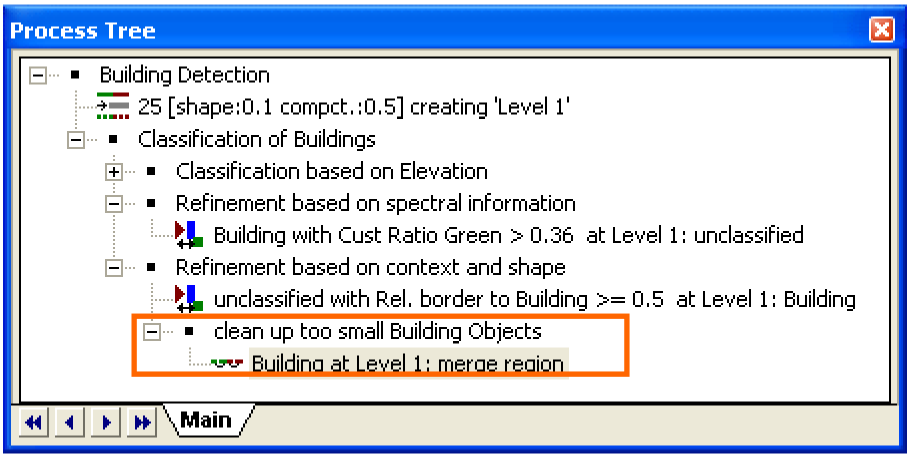
2. Choose '**merge region**' from the algorithm list.
3. Select '**Building**' as 'Class filter'.
4. Keep 'Threshold condition' empty.
5. Confirm with '**OK**'.





Settings Check

Figure 90: 'Edit Process' dialog box with settings to merge all 'Building' objects.



Rule Set Check

Figure 91: Process Tree with parent process 'clean up too small 'Building' objects added and with child processes to merge 'Building' objects.

Execute the classification process

6. Right-click on the process and select '**Execute**' from the context menu. Alternatively press F5 on your keyboard.

Action!

10.2.2 Find the feature and the threshold for refinement based on shape

Information

Within eCognition Developer objects can also be evaluated according to their shape characteristics, like their area.

Action!

1. In the 'Feature View' window browse to '**Object features>Geometry>Extent**'.
2. Double-click on the feature 'Area'.

Result Check

Figure 92: Feature View of the feature 'Area'.

The buildings appear very bright, this means the objects have high values. The area of buildings is significantly higher than the area of not building objects.

Action!

3. Right-click on it and select '**Update Range**' from the context menu.
4. Switch on the **check box** at the bottom of the 'Feature View' window.
5. **Increase the lower range** to the value 7000. All of the not 'Building' objects have values below 7000.

Result Check

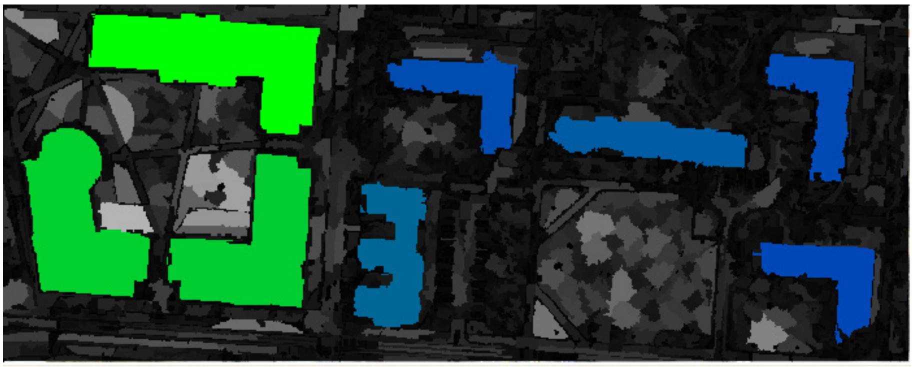


Figure 93: Feature View range of the feature 'Area'.

As a rule from the feature and the range it can be formulated that:

Unclassify all 'Building' objects with a value for the feature 'Area' of less than 7000.

10.2.3 Translate strategy in a rule for refinement based on the 'Area' feature

Insert process to classify

1. In the Process Tree right-click on the last process and select 'Append New' from the context menu.

Action!

Choose algorithm and object domain

Again only one condition is used, so use the algorithm 'assign class'.

2. Choose '**assign class**' from the algorithm list.

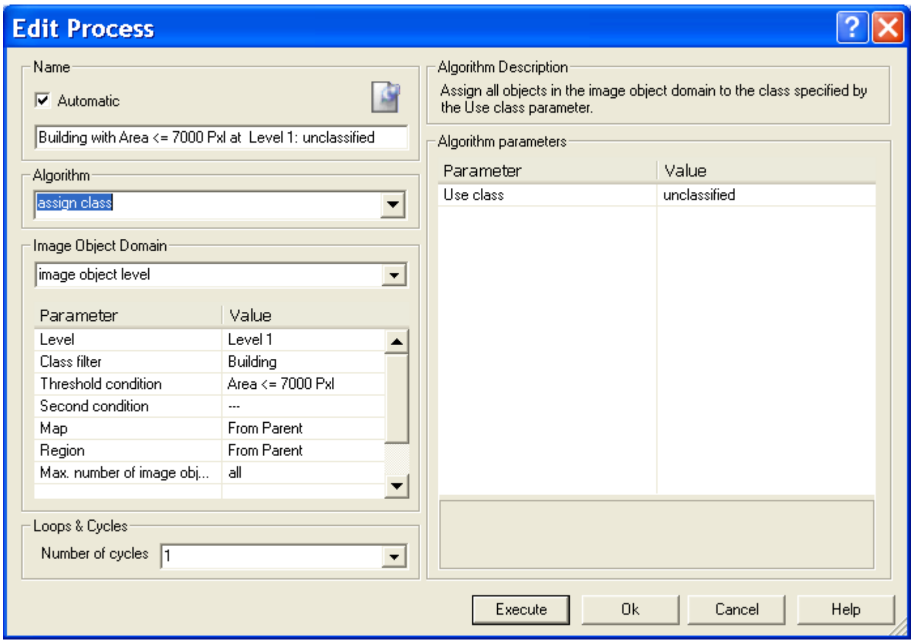
3. Select 'Building' as 'Class filter'.

Define condition

4. In the 'Threshold condition' field browse to '**Object Features> Geometry>Extent>Area**' and double-click on it.

5. In the 'Edit threshold condition' choose the **smaller than (<)** as operator and enter the value **7000**.

6. Confirm with **OK**.



Settings Check

Figure 94: 'Edit Process' dialog box with settings to un-classify all 'Building' objects smaller than 7000 pixel.

Rule Set Check

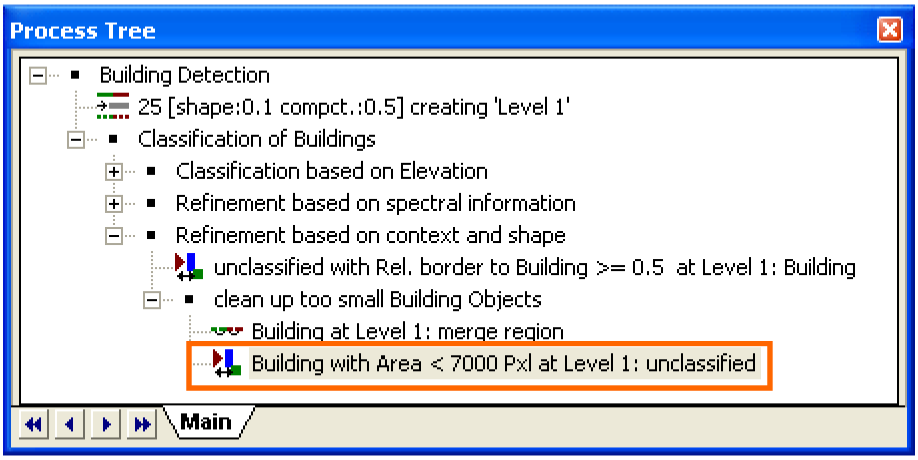


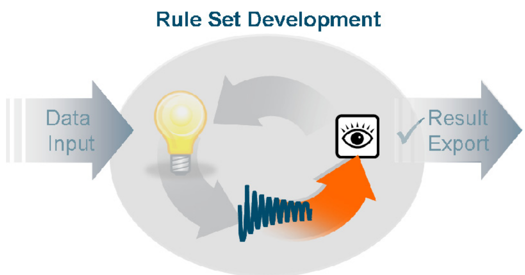
Figure 95: Process Tree with process to un-classify 'Building' objects.

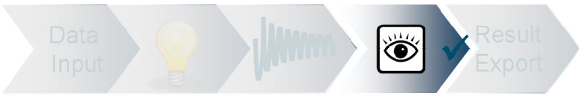
Execute the classification process

Action!

7. Right-click on the process and select '**Execute**' from the context menu. Alternatively press F5 on your keyboard.

Next step: Review Result!





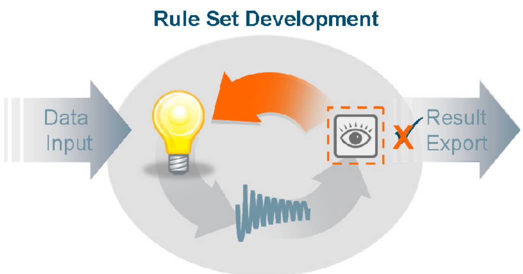
10.3 Review the classification result

No misclassified 'Building' object remains. The classification is ready for export!

Figure 96: Classification before and after the refinement.

This means the development is ready for the next step, the 'Export'.

Next step: Develop next analysis step!



Lesson 10 covered the following content

→ *Strategy to refine buildings based on shape information – generalize objects, separate them by size*

→ *Translate strategy in a rule for refinement based on a shape feature*

→ *Review the classification result*

Lesson 11 Exporting the result in a vector layer

This lesson covers the following content

- → *Insert process to merge 'unclassified' objects*
- → *Insert process to export shape file with attributes*
- → *Review the exported result*

Introduction

The buildings are classified correctly now and there are no misclassifications. The classification is ready for export. In this Guided Tour the classification will be exported as a vector .shp file. The buildings should be exported together with their size information.

Outlook on Process Tree:

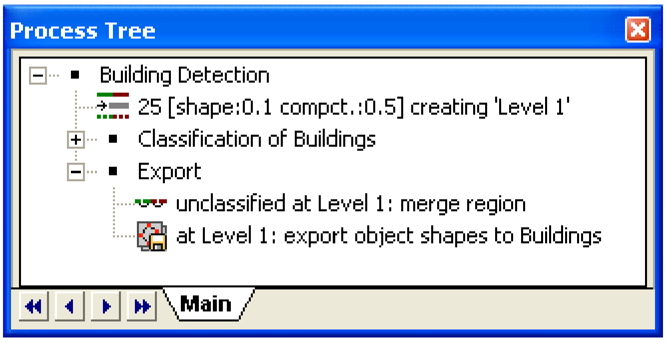


Figure 97: Process Tree with process added to export a vector layer of the result.

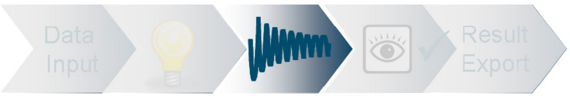
11.1 Insert process to merge 'unclassified' objects

This chapter covers the following content

- → *Prepare the Rule Set structure*
- → *Insert process to merge all 'unclassified*

Information

To clean up and reduce the number of objects in the scene first all unclassified objects must be merged.



11.1.1 Prepare the Rule Set structure

Action!

1. In the Process Tree collapse the process 'Classification of Buildings' and **right-click** on it. Select '**Append New**' from the context menu.
2. Insert 'Export' in the 'Name' field and confirm with ok.

11.1.2 Insert process to merge all 'unclassified' objects

1. In the Process Tree right-click on the 'Export' and select 'Insert Child' from the context menu.
2. Choose '**merge region**' from the algorithm list.
3. Select '**unclassified**' as 'Class filter'.
4. Keep '**no condition**'.
5. Confirm with **OK.**
6. **Execute** the process.

Settings Check

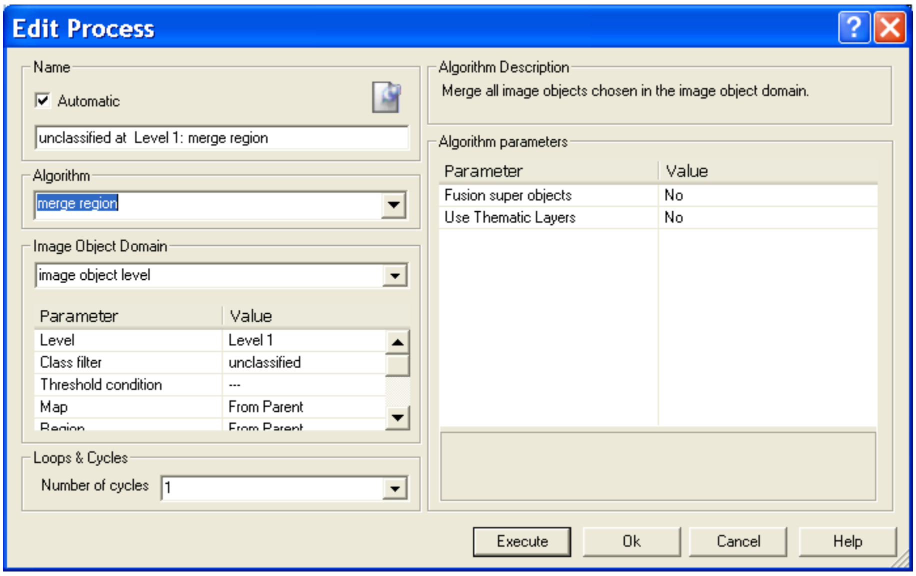


Figure 98: 'Edit Process' dialog box with settings to merge all 'unclassified' objects.

Rule Set Check

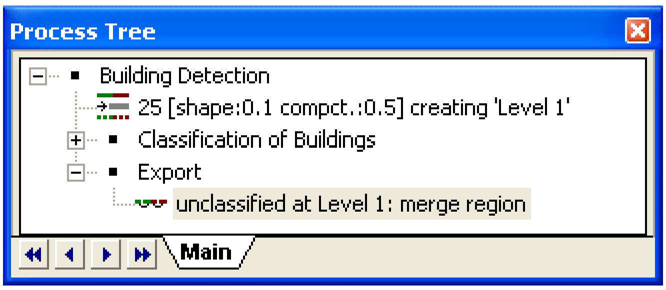


Figure 99: Process Tree with parent process 'Export' added and with child processes to merge 'unclassified' objects.

11.2 Insert process to export shape file with attributes

This chapter covers the following content

→ *Create the feature 'Class name'*

→ *Insert process to export vector file*

eCognition offers several algorithms to export e.g. current views, statistics and also vector layers.

11.2.1 Create the feature 'Class name'

Information

When exporting a file, per default no feature is defined to be additionally exported. To export now the class name to a vector file or any other object related file (object statistic or thematic raster file), the new feature '**Class name**' has to be added to the feature list (like area feature of other features).

The '**Class Name'** feature returns the name

- of the class of an image object
- or the name of the super-class of an image object
- or the name of the class of the super-object of an image object

This is controlled by the 'Distance' value.

Action!

1. In the 'Feature View' window browse to '**Class-Related features>Relations to Classification'**.
2. Double-click on '**Create new 'Class name''**.

The 'Class name' dialog box opens.

3. Keep the **default settings**.
4. Confirm the settings with 'OK'.

The feature is added to the 'Feature View' list.

✓

Result Check

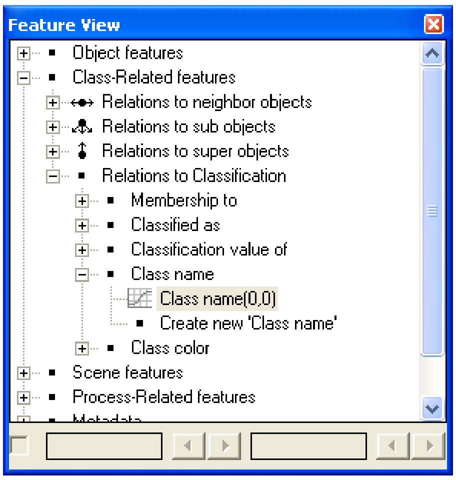


Figure 100: The feature 'Class name' in the 'Feature View' tree.

11.2.2 Insert process to export vector file

1. In the Process Tree right-click on the last process and select 'Append New' from the context menu.

Action!

Choose algorithm and object domain

2. Choose '**export vector layers**' from the algorithm list.
3. Keep the '**Class filter**' and 'Threshold condition' as default.

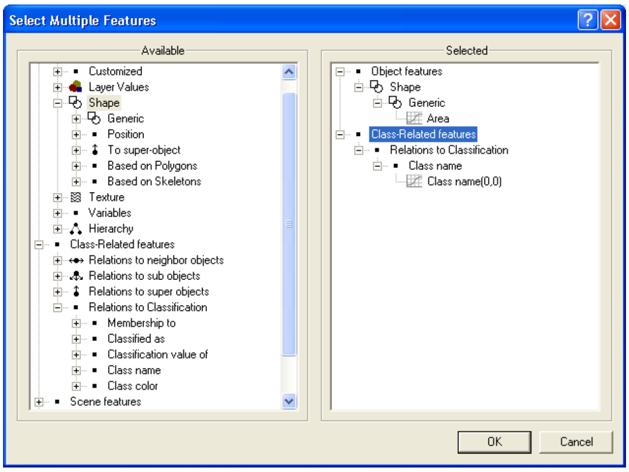
Define the parameter

4. In the field 'Export mode' keep 'Use export item'.
5. In the field 'Export item name' insert '**Building**'. This will be the name of the exported file.
6. Click in the field '**Attribute table**'.

The 'Select Multiple Features' dialog box opens.

7. Browse to the '**Class-Related features>Relations to Classification>Class name**' and double-click on it to move it to the 'Selected' window.
8. Do the same for the 'Area' feature (Object Features>Geometry>Extent>Area). Confirm with OK.

The features are added to the export specifications.



Result Check

Figure 101: The 'Select Multiple Features' dialog box.

9. In the algorithm parameter field 'Shape Type' change from 'Points' to '**Polygons**'.
10. Keep all other default settings.
11. After clicking 'OK', the 'Edit Attribute Table Columns' window appears, where other settings are possible.
12. **Execute** the process.

Action!

The shape file together with its attributes is exported to the location where the image data is stored.

Settings Check

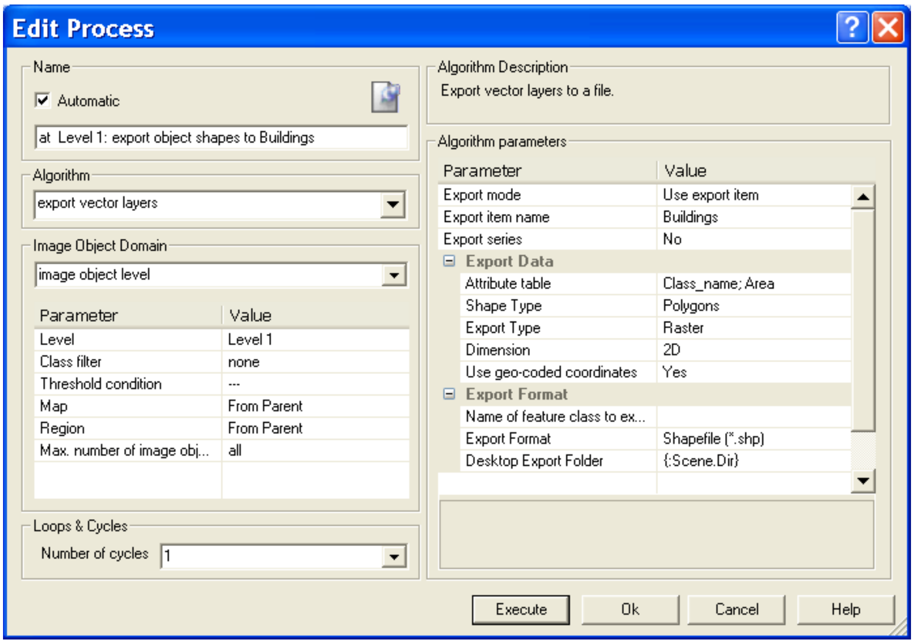


Figure 102: 'Edit Process' dialog box with settings to export a vector layer.

Rule Set Check

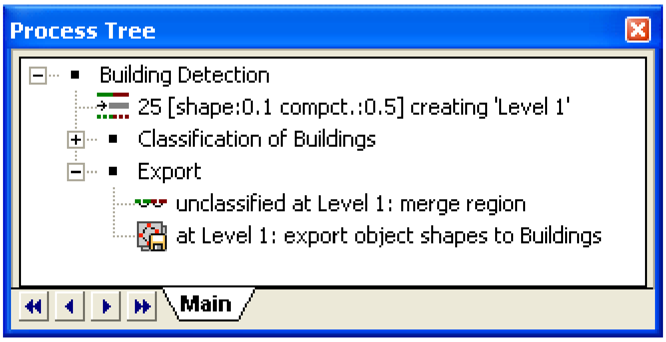


Figure 103: Process Tree with process added to export a vector layer of the result.

Lesson 11covered the following content

→ *Insert process to merge 'unclassified' objects*

→ *Insert process to export*

11.3 Review the exported result

A shape file has been exported. Review it with ArcGIS or simply check the exported .dbf file.

Figure 104: Final classification of the buildings

	A	B
1	Class_name	Area_Pxl
2	unclassified	1390.000000000000000
3	Building	84173.000000000000000
4	Building	28798.000000000000000
5	Building	29098.000000000000000
6	Building	34217.000000000000000
7	Building	69489.000000000000000
8	Building	67429.000000000000000
9	Building	38365.000000000000000
10	unclassified	1201171.000000000000000
11	Building	28430.000000000000000
12		

Figure 105: Exported .dbf table containing the area and the classification of every object.

Lesson 11 covers the following content

- → *Insert process to merge 'unclassified' objects*
- → *Insert process to export shape file with attributes*
- → *Review the exported result*

Where to get additional help and information?

eCognition Community and Rule Set Exchange platform

The eCognition community helps to share knowledge and information within the user, partner, academia and developer community to benefit from each other's experience. The community contains next to other content:

- **Wiki**: collection of eCognition related articles (e.g. Rule Set tips and tricks, strategies, algorithm documentation...).
- **Discussions**: ask questions and get answers.
- **File exchange**: share any type of eCognition related code such as Rule Sets, Action Libraries, plug-ins...
- **Blogs**: read and write insights about what's happening around our industry..

Share your knowledge and questions with other users interested in using and developing image intelligence applications for Earth Sciences at http://community.ecognition.com/.

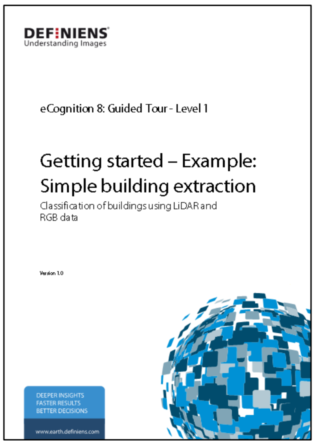
User Guide and Reference Book

Together with the software a User Guide and a Reference book is installed. You can access them in the Developer interface in the main menu 'Help>eCognition Developer User Guide' or Reference Book.

The Reference Book lists detailed information about algorithms and features, and provides general reference information.

Both are also available in the Wiki section of the community http://community.ecognition.com/home/app_wiki

Additional Guided Tours and Tutorials



If you are interested in further auto learning material you can work through the other Guided Tours you can download them on the eCognition community (http://community.ecognition.com/home/app_wiki).

Overview over the current Guided Tours:

- Level1_GuidedTour_LiDAR_SimpleExample
- Level2_GuidedTour_MappingImperviousSurface
- Level3_GuidedTour_LiDAR_AdvancedExample

Tutorials can be found in the Wiki section of the community
http://community.ecognition.com/home/training-material

eCognition Training

eCognition Training Services offer a carefully planned curriculum that provides hands-on, real-world training. We are dedicated to enhancing customers' image analysis skills, helping these organizations to accomplish their goals. Tailored courses are available to meet the needs of customers engaged in Life and Earth Sciences.

Our courses are held in our classrooms around the world and at customer's sites. We offer regular trainings as Open Classes, where anyone can register and as In-Company Training. We also offer Customized Courses to satisfy customer's unique image analysis needs, thereby maximizing the training effect.

For more information please see our website at
http://www.ecognition.com/learn/trainings .

eSeminars

Join one of the live, online eSeminars:

http://www.ecognition.com/learn/trainings/web-based-self-study-training

Consulting

eCognition Consulting Services experts have deep insight into products, best practices and professional project management skills. As a result, we are able to deliver rapid implementations that maximize the return on investment and minimize total cost of ownership. Consulting can also provide full-service, turnkey solutions, including project management, off-site rule development, QA and implementation reviews. Customers can count on our consultants' considerable expertise in the application of technology.

Buy Software and Services?

If you want to purchase software or if you want to know more about pricing please visit http://www.ecognition.com/buy and get in contact with our sales team.