User Guide of GuidosToolbox

Peter Vogt

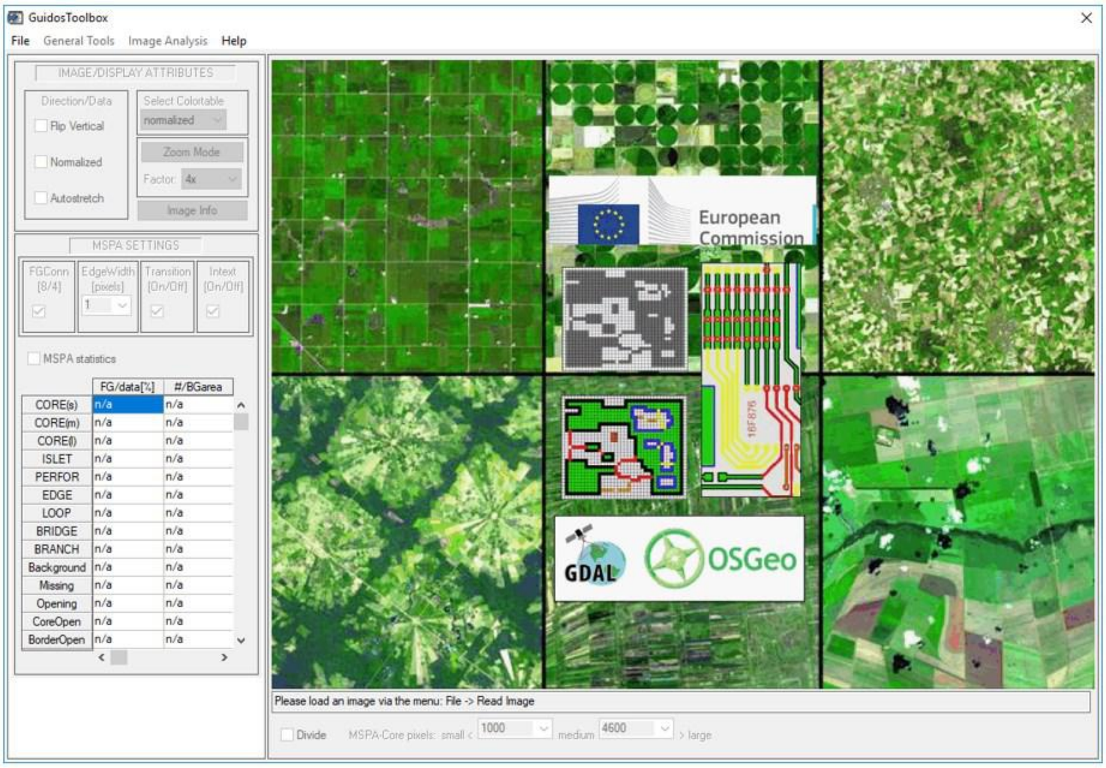
European Commission
Joint Research Centre (JRC)
Via Enrico Fermi 2749, TP 261
I-21027 Ispra (VA), Italy

Release 2.7, Revision 2, August 2018

GuidosToolbox (**G**raphical **U**ser **I**nterface for the **D**escription of image **O**bjects and their **S**hapes - GTB) provides generic image processing tools.

All tools are based on geometric concepts only and can thus be applied to any kind of raster data.

This document describes the various menus and features of GuidosToolbox:



Contents

The system requirements are a PC or a Mac with at least 2 GB of system memory, the more the better. This application has been tested on a variety of current Linux distributions, Intel-MacOS, and Microsoft Windows® XP SP3 – Windows 10 platforms.

Support for MS-Windows 32-bit operating systems will be discontinued in the near future.

The latest version of this application, including manual and installation instructions, can be obtained from the GuidosToolbox homepage.

Citation reference for GuidosToolbox:

Vogt P., Riitters, K., 2017. GuidosToolbox: universal digital image object analysis. European Journal of Remote Sensing 50:1, 352-361, DOI: 10.1080/22797254.2017.1330650. Software available for free at: http://forest.jrc.ec.europa.eu/download/software/guidos

Please add the following reference if you use MSPA in your work:
Soille P., Vogt P., 2008. Morphological segmentation of binary patterns. Pattern Recognition Letters 30, 4:456-459, DOI: 10.1016/j.patrec.2008.10.015

Windows and menus

The following graphical elements can be used to interact with GuidosToolbox:

- a horizontal menu bar (top panel)
- a window to set different attributes of the image and its graphical display (top left panel)
- a window to set MSPA parameters and statistics (bottom left panel)
- a viewport (right panel)
- a window for image coordinates & values (bottom right panel)

1. The menu bar of GuidosToolbox

The top menu bar offers four pull-down menus:

- File
- General Tools
- Image Analysis
- Help

1.1. The File pull-down menu

The File pull-down menu offers the following options:

- Read Image
- Save Image
- Batch Process
- Change
- Exit

1.1.1 Read Image

This menu is used to read your input data:

- **GeoTiff: The default image type is GeoTiff** (a ".tif"-formatted file having a geoheader - information on projection, etc.). GeoTiff can be read or pre/post-processed by any image processing (IP) software.

- **Generic:** Image of formats like tif, png, bmp, jpeg, etc.

- **IP Software:** IP software (ESRI, ARC, IDRISI, etc.) raster image formats like img, bil, etc. The included Gdal will try to convert these formats into

GeoTiff. Alternatively, go back into your IP software and export the image to GeoTiff.

- **ENVI:** An image from an ENVI session (extension ".hdr"). GuidosToolbox will not use the geo-information of the ENVI-data. If you want to maintain the geo-information you should export the data in ENVI to GeoTiff and read this format in GuidosToolbox.

1.1.2 Save Image

This menu is used to save your processed data. The options are similar as in the Read menu. Additional options are:

- **Display Snapshot:** This option will save a snapshot of the current viewport in GuidosToolbox. This may be useful when working with a large image and a quick-look of the processed image is sufficient. When saving a MSPA result, the filenames will include the settings of the 4 MSPA parameters. If MSPA statistics were enabled, an additional file containing these statistics will be saved with the same notation and the additional suffix '.txt'.

- **KML:** This option will export to kml-format for visualization in Google Earth, requiring an image in the projection EPSG:4326 (WGS 84). To re-project either use reproject to GoogleEarth or start the GTB Terminal and use the command gdalwarp and the appropriate EPSG reference codes. The result is stored in a zip-archive. Extract this archive at any location and load the included kml-file into Google Earth. When exporting a MSPA or a FAD image, a customized header plus legend is added automatically.

The default data directory is the subdirectory 'data' located in the main GuidosToolbox directory.
Note: Please use the default GeoTiff format for maximum compatibility.

1.1.3 Batch Process

This option allows for automatic processing of multiple files in batch mode. Please put all batch input files into a dedicated subdirectory inside the GTB-'data' directory and avoid empty spaces in file names or directory path names. Input files must be in the default format (Geo-)Tiff. GTB will save the output files in batch-processing specific directories located next to the batch-input directory, i.e., batch-input files directory: **GuidosToolbox/data/batch-input/** and resulting output files (for example for Accounting) are in the directory: **GuidosToolbox/data/batch-ACC**/
Batch-processing options are similar as in the Image Analysis menu.

Objects:

- **Accounting:** Select a series of images for batch processing of **Accounting**.

- **Parcellation:** Select a series of images for batch processing of **Parcellation**.

Pattern:

- **Morphological:** This option will process the user-selected files using the following options:
  - **SPA3/5/6:** Calculate the spatial pattern analysis resulting in 3/5/6 classes and save the resulting maps and statistics in the input directory. Images processed with SPA3/SPA5/SPA6 are not subject to size limitations as is the case for the full version of MSPA.
  - **MSPA:** This option will open a window to select the MSPA parameters and a switch for statistical output. The resulting filenames will include the settings of the 4 MSPA parameters and, if selected, separate statistics.

- **Moving Window:** This menu provides options for batch processing of several moving window algorithms. The dimension of the moving window can be specified via a popup window (**Figure 3**).

Fragmentation:

This menu provides options for batch processing of several fragmentation analysis algorithms featured in the Image Analysis – **Fragmentation** section.

Distance:

- **Euclidean Distance:** Batch-process images for **Euclidean Distance** resulting in maps and a summary of the distance histogram properties.

Network:

- **MSPA-based:** This menu provides options for batch processing of several network analysis algorithms featured in the Image Analysis – **Network** section. All options require MSPA input images, non-MSPA input images are ignored and cannot be processed.

- **ConeforInputs:** This option will generate the node file (area per object) and the connection file (pairwise distance between objects) in the format required by Conefor. Object area and distance will be calculated for objects of value 2 only (if needed, use *Recode* to reassign object values). The distances can be calculated as Euclidean edge-to-edge distance (8/4-connected) or centroid-to-centroid distance between all foreground objects of the raster image.

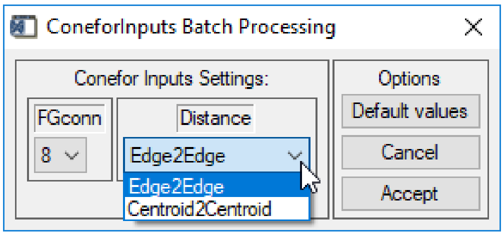


Figure 1: *ConeforInputs*: Select 8/4-connectivity and distance measures.

Note: When using the option ***ConeforInputs*** the connectivity is defined via the *pairwise distance* between image objects. This is different to the option **MSPA ConeforInputs**, where the connectivity is defined via the MSPA-detected *structural connectors* (Bridges)

Recode:

Batch recoding for a series of images. First, select a sample image to set up the recoding table. Batch-recoding requires a recoding table with 256 entries in order to deal with all kind of class allocations. A recoding table can be set up from scratch or saved and restored later. Here, please use the naming scheme GTBrecode*.sav, and replace the placeholder * with a meaningful name. After the recoding table has been defined, select a series of images to which this recoding table should be applied.

1.1.4 Change

This menu provides the following two options for change analysis. A popup window allows for selecting the two input images, which are then processed:

- **FOS:** This option will calculate changes in fragmentation classes between two **FOS** images. The summary statistics for the fixed, user-selected observation scale are saved in txt and csv-format.

- **FAD:** This option will compare two **FAD** image sets and calculate changes in fragmentation classes over all observation scales as well as the multi-scale images. All changes are summarized in a bar-plot image and further detailed in observation-scale specific summary statistics provided in txt and csv-format. The output files are:
  a) fadchange_barplot.png: FAD-bar chart summary
  b) fadchange_{7,13,27,81,243,mscale}.txt: FAD summary statistics by scale

c) fadchange_{7,13,27,81,243,mscale}.csv: FAD change summary statistics by scale in comma-separated value format for import into a spreadsheet application

The first table in the summary statistics shows the number of pixels being in one of the FAD-classes or in the background at time A and at time B. If the area of a pixel is known and constant throughout the image (equal-area projection) then the number of pixels times the pixel-area corresponds to actual area measures. Gross area gain equals to the number of pixels changing from background at time A to any foreground class at time B. Similarly, gross area loss equals the number of pixels changing from any foreground class at time A to background at time B. The net area change is the sum of gross area gain and gross area loss.
The second table in the summary statistics shows the relative changes within the foreground (at both dates) only. Entries along the diagonal of the matrix are zero because these entries correspond to pixels being in the same class at time A and B, equivalent to no change. Matrix entries below the diagonal represent fragmentation increase expressed with positive percentages. Fragmentation decrease is expressed with negative percentages and found above the matrix diagonal. With this setup, the second table shows the relative contribution of each possible fragmentation class change for each fraction, fragmentation increase and fragmentation decrease.

- **LM heatmap:** This option will calculate the difference of the occurrence frequency [%] in the heatmaps of image *B* minus image *A*. The resulting delta heatmap image and the corresponding csv-table are saved into the GTB/data directory.

- **Simple Change:** This option will calculate a simple by-pixel difference of image *B* minus image *A*, starting at the top left corner of the image. The dimension of the resulting image is equivalent to the smaller image of the two input images.

- **Morphological Change:** This option will conduct a Morphological Change Detection (MCD), described in Seebach et al. (2013). MCD will remove unwanted spurious changes caused by mis-registration between classified maps and their thematic accuracies. MCD requires that both input images are binary MSPA-compliant maps (2b – FG, 1b – BG, 0b – Missing (optional)) in GeoTiff format, having the same projection, pixel-resolution, location, and an area in common. The resulting GeoTiff image will show the essential changes in the common area of input image *A* and *B*. The output values have the following assignment: 11b – BG/BG, 12b – BG/FG (gain), 21b – FG/BG (loss), 22b – FG/FG, 176b – neglected, spurious changes, 255b – undetermined (missing data in either input image). The title bar lists net percent change values for FG (Foreground) and FGi (Foreground interior area: 1-pixel eroded FG), with negative values indicating a net loss. Finally, *Elasticity* (Riitters et al., 2015) is the ratio of FGi to FG. For example, in case of forest loss higher elasticity values indicate higher fragmenting effects on the remaining forests.

1.1.5 Exit

Quit the program and return to the operating system.

1.2. The General Tools pull-down menu

The General Tools pull-down menu provides generic image processing utilities, which may be useful for a variety of purposes. The following options are available:

- Preprocessing
- Convolution
- Equalization
- Thresholding
- Edge Enhance
- Morphological
- GIS Software
- Original Image
- Switch Cursor
- Undo/Redo

1.2.1 Preprocessing

The Preprocessing menu provides several generic image-processing routines targeted to reassign image pixel values. These routines can also be used for setting up MSPA-compliant input images. The following options are available:

- **Convert → Byte/Integer/Long:** Convert data to the selected data type.
- **RGB → Single Band:** Convert RGB-image to single-band image.
- **Reproject for GoogleEarth:** If the geo-header information of the currently loaded GeoTiff file has a EPSG code and it is different to 4326 then this option can be used to reproject the image to EPSG:4326 (WGS 84).
- **Recode:** A table showing new and current unique pixel values is displayed. The new value entries in the left column can be reassigned to match the desired recoding of image class values (*recode.c*, K.Riitters, personal communication 2015). The *Set all to* dropdown menu can be used to assign all classes to a single value. The *Save* button can be used to save a new recode table to a file GTrecode*.sav (please keep the prefix

GTrecode and the extension *.sav* and only modify the placeholder *). Any previously saved recoding table GTrecode*.sav can be restored via the *Restore* button. Here, only those entries that match the current class values will be restored. This option is only available for images of data type Byte. Please note that any new value in the table will only be taken over after pressing the Enter key or after leaving the entry field.

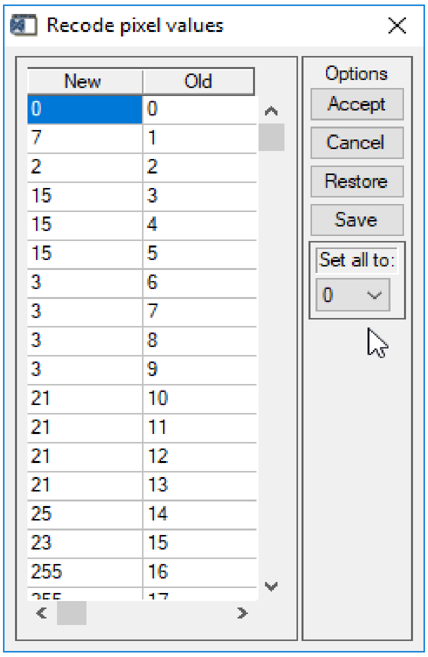


Figure 2: The *Recode* GUI showing existing class values in the right column (Old) and new assignments in the left column (New).

- **Cost Marker Image:** This menu allows defining a marker image for the Cost analysis (p. 42). Instructions to mark start/target objects as points or polygons as well as missing data are provided via popup windows.

- **Threshold: FG/BG:** Apply a threshold to a gray-scale image to obtain a MSPA-compliant binary image with foreground and background (FG/BG) values assigned to 1 byte and 2 byte, respectively.

- **Group: FG/BG/Missing:** Group a sequence of gray-scale image values into FG/BG/Missing.

- **X → FG/BG/Missing:** Assign an individual pixel value to FG/BG/Missing.

- **Invert:** Exchange the current assignment of FG/BG/Missing.

- **Add 1b:** Add 1 byte to all current image pixel values.

- **Subtract 1b:** Subtract 1 byte of all current image pixel values.

Note: The *Image Info* box in the main window can be used to control the output of the Preprocessing functions or to query image properties.

1.2.2 Convolution

The Convolution menu offers the choice for one of the following image convolution filters:

- **Median:** Median filtering is effective in removing salt and pepper noise, (isolated high or low values). The median is the middle value of a given data array, which should not be confused with the average value. A set of predefined median filter box sizes is available in the submenu.

- **Boxcar:** Similar to Median, this filter computes the average value instead.

- **Lee:** The Lee filter technique (Lee, 1986) will smooth additive image noise by generating statistics in a local neighborhood and comparing them to the expected values. A set of predefined box sizes for the Lee filter is available in the submenu.

- **Sigma:** This filter computes the mean and standard deviation of pixels in a box centered at each pixel of the image, but excluding the center pixel. If the center pixel value exceeds one standard deviation from the mean, it is replaced by the mean in the 5×5 box overlaying the pixel of investigation.

- **Hilbert:** The Hilbert function outputs a series that has all periodic terms phase-shifted by 90 degrees. This transform has the interesting property that the correlation between a series and its own Hilbert transform is mathematically zero.

- **User-Defined:** A popup window allows the user to select or customize a kernel within dimensions within [3, 501] to be applied to the current image. The kernel size (*ks*) is an uneven number because it is measured with respect to the center pixel; for example, *ks=5* corresponds to a moving window with x/y-dimensions of 5 pixels (the center pixel ± 2 pixels in x/y-direction). Some predefined kernel sizes can be selected to analyze typical area measures *A* in [m^2] at spatial resolution *res* in [m], in general: $ks = \frac{\sqrt{A}}{res}$ For example, if we have satellite data with a spatial resolution of *25m* and want to apply an analysis window area of *1000 acres* (4046856 m^2) we need *ks=81* ($\frac{\sqrt{4046856}}{25} = 80.47$, closest uneven number is 81), or you insert your custom size, i.e., an analysis window of 1km^2 on satellite data having 30m resolution: *ks=33* ($\frac{\sqrt{1000000}}{30} = 33.33$, closest uneven number is 33).

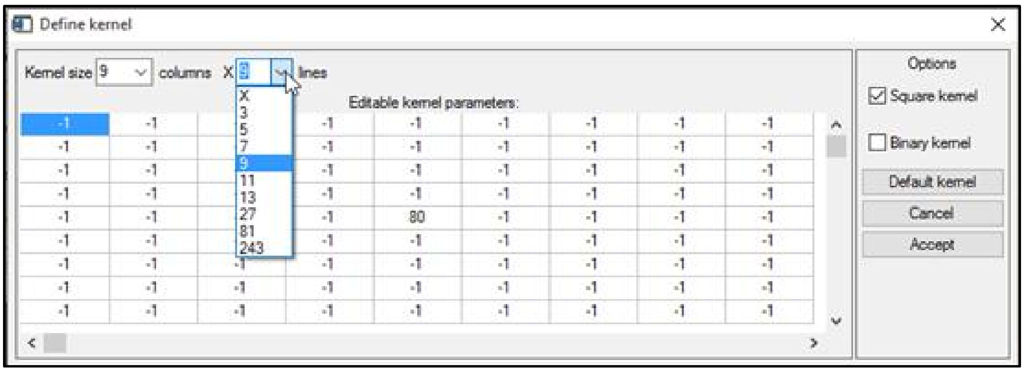


Figure 3: The *Define Kernel* interface showing options to define the kernel dimension within the range [3, 501], select a square and/or a binary kernel and a reset option to go back to the default kernel for the current application.

1.2.3 Equalization

The Equalization menu offers the choice for one of the following image processing algorithms:

- **Contrast:** The image contrast can be adjusted in an interactive mode. This option will close the main window and open up a new, dual window interface (Figure 4). The window on the left displays the histogram of the brightness values of the image and two color bars, which can be dragged with the mouse to set a lower and upper limit for the intensity range into which the image will be rescaled. The result of this procedure is shown in the preview window on the right, which can be saved in a variety of image formats from the *Controls → Save image as* menu. Furthermore, the maximum pixel density may be reset from the *Controls → Max pixel density* menu to display the histogram in an adequate way. Finally, choosing *Quit/Accept* will close this interface, return to the main interface, and cancel/apply the selected settings to the current image.

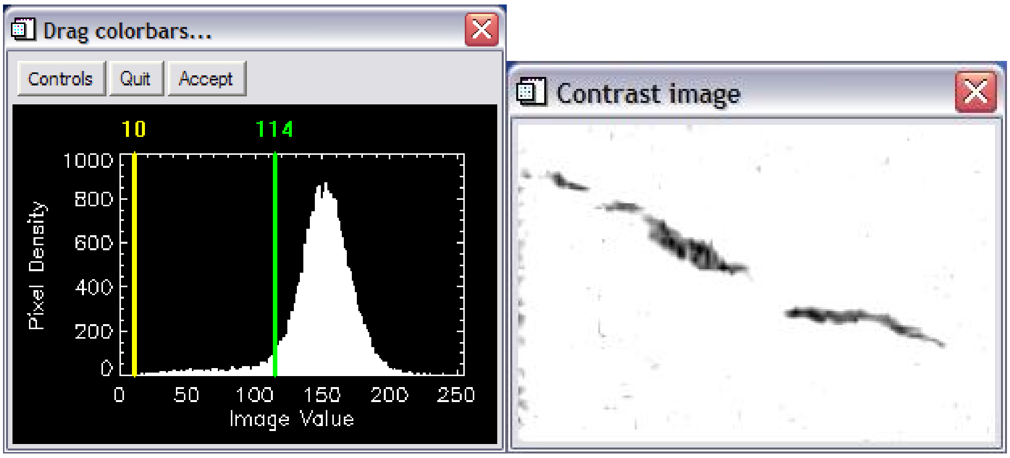


Figure 4: The *Contrast* interface showing on the left the histogram of the brightness values including the two color bars to define the contrast range; and on the right the preview of the selected contrast range applied to the current image.

- **Histogram Equalization:** This function is used to obtain the density distribution of the input array. The histogram is integrated to obtain the cumulative density-probability function and a histogram-equalized image is returned.

- **Adaptive Histogram Equalization:** This function applies contrast enhancement based on the local region surrounding each pixel. Each pixel is mapped to an intensity value, which is proportional to its rank within the surrounding neighborhood.

1.2.4 Thresholding

This option will open a dual window (**Figure 5**). The left window displays a histogram of the brightness values from the current image. A color bar indicating the current threshold value can be moved with the mouse to set a new threshold value, which is displayed in the right window of this interface. The *Controls* menu provides the option to save the current preview to an image in a variety of image formats and to redefine the maximum pixel density in order to best display the brightness value distribution of the histogram. Finally, the *Quit* or *Accept* buttons are used to cancel/apply the selected threshold. Please note that this process results in a binary image and can therefore only be applied once.

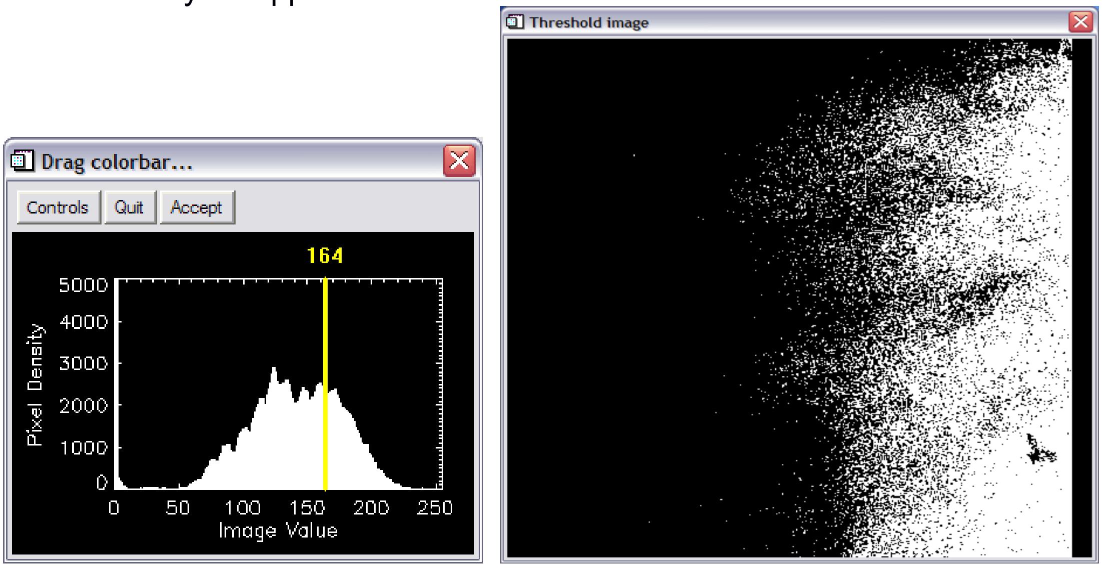


Figure 5: The *Thresholding* interface provides a color bar to set and preview a brightness threshold value on the current image.

The result of using this interface is a binary image having foreground and background (FG/BG) values assigned to 0/1 byte, respectively. This is different to the option in *Preprocessing* → *Threshold* where the resulting binary image is MSPA-compliant having foreground and background (FG/BG) values assigned to 1/2 byte.

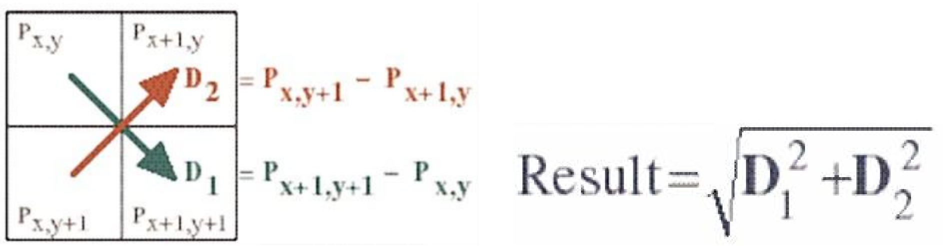
1.2.5 Edge enhance

This Edge enhance menu provides a series of edge enhance filters, which are described in the following:

- **Canny:** This filter produces results similar to the Sobel operator, but is separable (Canny, 1986). This means that it can be performed in two passes, computing the derivative in the horizontal and the vertical direction. They are then combined to calculate the gradient or the difference of the two components.

- **Laplace:** The image is convolved with the kernel $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$

- **Roberts:** The diagram below shows the principle of the Roberts operator (1965) where two differences in orthogonal directions are combined to determine the gradient.



- **Sobel:** Similar to the Roberts filter, the Sobel filter computes the derivatives in two orthogonal directions, which are combined as the square root of the sums of their squares to obtain a result independent of orientation.

- **Sharpen:** This filter adds the Laplace filter of the image to the image itself. The *Define Kernel* window (**Figure 3**) offers the option to use a different user-defined filter.

- **Unsharp masking:** This filter subtracts the original image from a smoothed version of the original image. Different smoothing box sizes are selectable from the submenu.

- **Skeleton:** This filter combines thresholding with the Sobel filter.

1.2.6 Morphological

Mathematical morphology is a methodology of analyzing digital image objects based on their shape attributes. A discussion of this topic is beyond the scope of this manual. A suggested reference is Soille, 2004; In short, a user-selectable object of predefined size and shape (structuring element) is defined and the image is scanned for the presence or absence of this shape of interest (some examples in **Table 1**). This option will open a new window (**Figure 6**) providing a series of morphological filters (Dilate, Erode, Open, Close, Tophat, Gradient), structuring element types (horizontal, vertical, diagonal up, diagonal down, circular) and sizes (1-10). Any combination of filter and structure type and size can be selected and tested by pressing the *Test* button in the upper panel. The *Define kernel* window (**Figure 3**) allows for selection of the default or user-defined kernel to be applied to the previously selected settings. The result of this process is displayed in the graphical display of this window. Other combinations may be tested at any time after restoring the original state in the display via the *Reset* button. When using the *Tophat*-filter, the contrast of the resulting image can be adjusted with the sliders in the horizontal panel above the display. Finally, the *Quit/Accept* button will close this window and cancel/apply the selected filter settings to the current image in the main window.

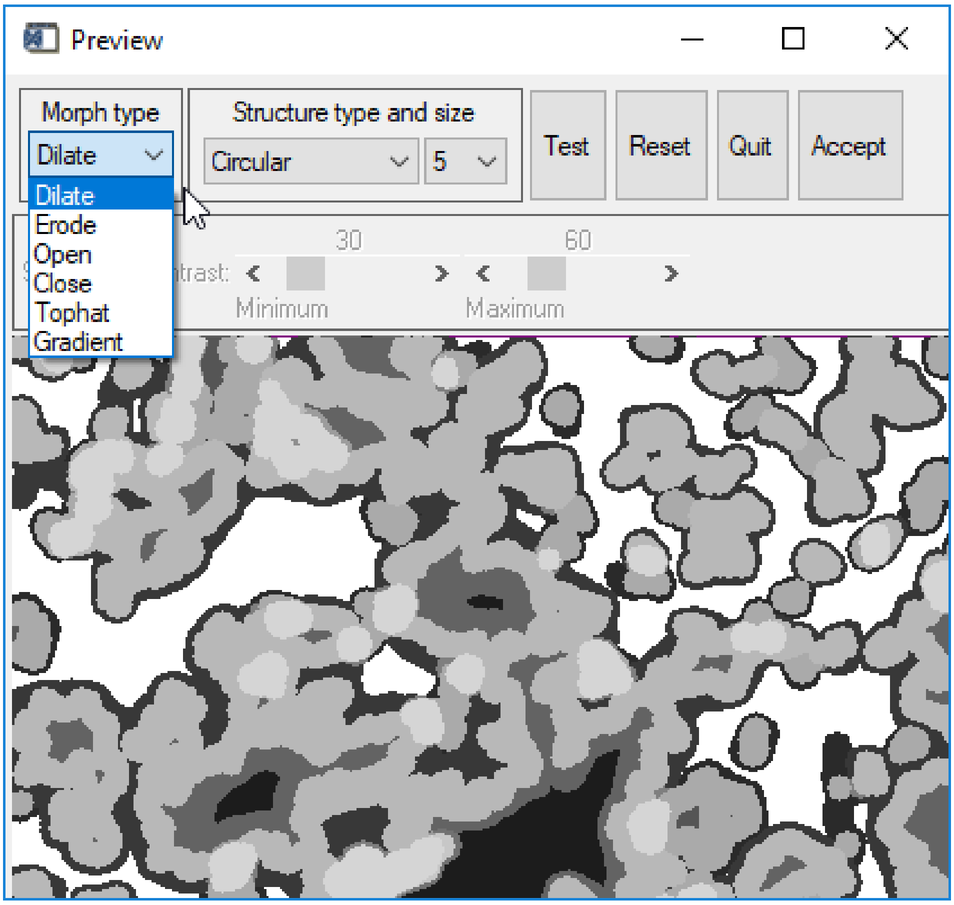


Figure 6: The *Morphological filter* interface providing a preview for a variety of combinations of filters with structuring element type and sizes.

Filter	Description
Dilate	fills holes of size equal to or smaller the structuring element.
Erode	removes islands smaller than the structuring element
Open	an erosion followed by a dilation, pixels are removed
Close	a dilation followed by an erosion, pixels are added
Tophat	= image–open(image): Shows brightness peaks on inverted image
Gradient	= dilate(image)–erode(image): Highlight structure boundaries

Table 1: Description of some basic morphological filters.

1.2.7 GIS Software

The GIS Software menu provides access to the following GIS related software packages on the host PC running GuidosToolbox:

- **GTB Terminal:** This option will open a separate terminal to access all GDAL-commands. Use this terminal to work on your data, for example to reproject (gdalwarp), format conversion (gdal_translate), get information (gdalinfo), and much more.

- **OpenEV Viewer:** This option will open a separate window with the image viewer OpenEV. Note that OpenEV is not available in the Mac version of GuidosToolbox.

- **QGIS**: This option will open QGIS if it is found in the operating system.

- **Conefor**: This software permits quantifying the importance of habitat patches and links for landscape connectivity. In GuidosToolbox, you may use **ConeforInputs** or **MSPA ConeforInputs** to setup input files for further graph-theory analysis in Conefor.

- **IMPACT Toolbox**: This software collection offers a combination of remote sensing techniques, photo interpretation, and processing technologies in a portable and stand-alone GIS environment allowing users to accomplish all necessary pre-processing steps for the production of a land cover map from Earth observation data.

- **Nestedness**: Nestedness is a measure of order in an ecological system, referring to the order in which the number of species is related to area or other factors. The more a system is "nested" the more it is organized.

1.2.8 Original Image

Selecting the *Original Image* button will restore the initially loaded image.

1.2.9 Switch Cursor

This option can be used to change the appearance of the mouse cursor within the viewport of GuidosToolbox.

1.2.10 Undo/Redo

GuidosToolbox stores the settings of one (1) processing step only. The *Undo* button may be used to undo the last processing step. If selected, this button will change to *Redo* allowing reverting to the previous step.

1.3. The Image Analysis pull-down menu

The Image Analysis pull-down menu offers the following options:

- Objects
- Pattern
- Fragmentation
- Distance
- Cost
- Network

1.3.1 Objects

This menu provides several options for the analysis of individual image object attributes.

- **Accounting:** This option will label and calculate the area of all foreground objects (coded with 2 byte) according to user-specified area thresholds:

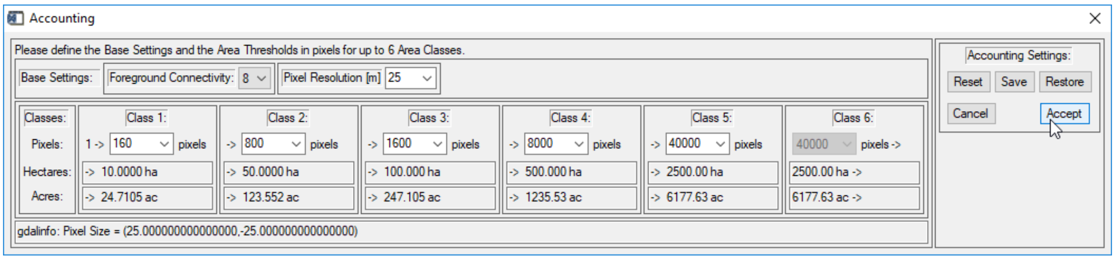


Figure 7: The *Accounting* interface providing user-selectable fields for Foreground Connectivity, Pixel Resolution and up to six foreground-object size classes.

Figure 7 shows the *Accounting* interface where the user can select or insert custom values to define the area thresholds (in pixels) of up to six object size categories. The bottom panel shows the spatial pixel resolution

retrieved via the *gdalinfo* command. Depending on the geoheader, it might be in meters, degrees, any other unit, or not available at all and for this reason should be considered for information only. The initial 5 area thresholds are derived by GTB by calculating the range from the smallest to the largest object area and scaling the range into 5 sections starting from the smallest object area + [0.001, 0.01, 0.1, 0.5, 0.9] * range. The *Reset* button can be used to re-assign the initial automatic scaling thresholds. The *Save* and *Restore* button save/restore custom user settings. The area interval of Class 1 starts with the minimum object area up to and including the area threshold specified by the user. The following Classes start at the area threshold of the previous Class + 1 pixel up to the Class threshold. Class 6 accounts for all objects larger than the area threshold of Class 5. The interface will automatically adjust the thresholds to ensure a positive progression of the threshold settings. Neighboring classes can be collapsed into a single class by assigning the same area threshold value. Combined with the pixel resolution, the two bottom rows show the upper area limit of each class in hectares and/or acres. Please note that this information is only valid when using an image having an equal area projection and a properly specified information on the pixel resolution.
Via the *Accounting* interface, the user can set each area threshold to define specific area classes in pixel, hectare, or acre. To apply the user-defined settings press the *Accept* button.

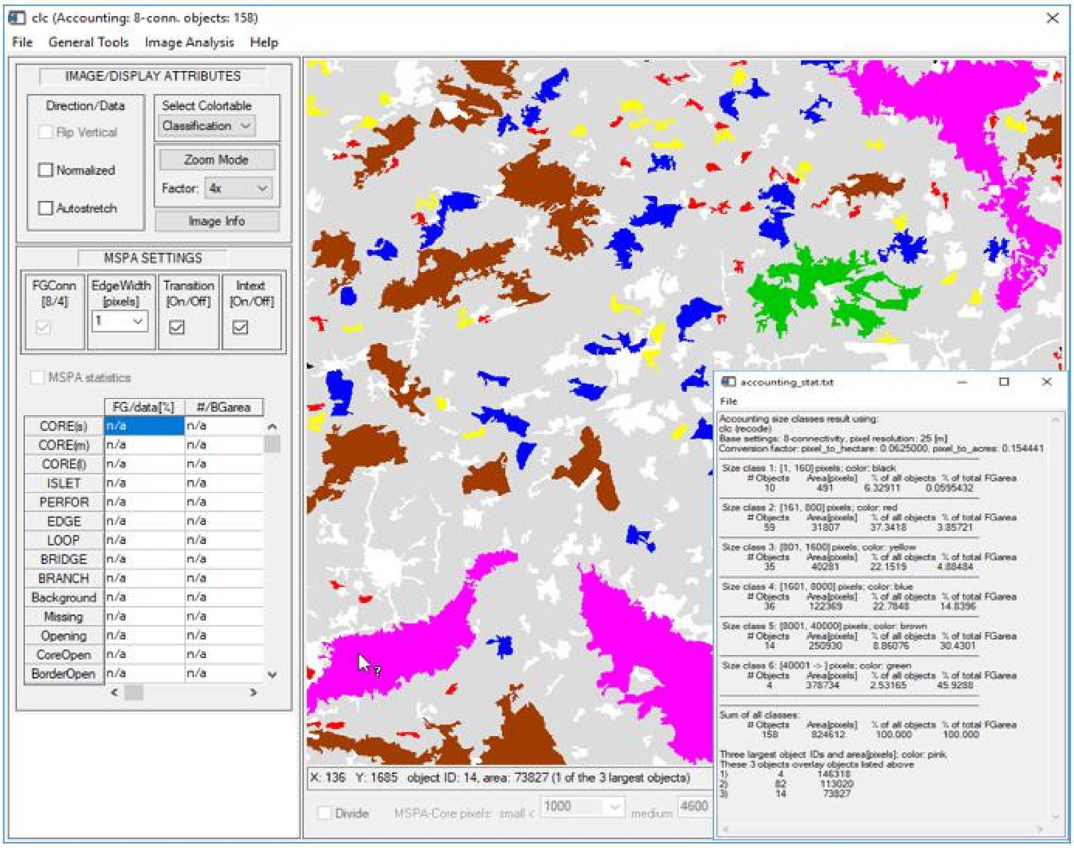


Figure 8: *Accounting* result showing different area categories in different colors and a statistical summary for the number of objects, their area and proportions in up to six different area classes, and highlighting the 3 largest objects.

The result is an image showing all objects in up to six different area classes, each with its own color, background in grey and potential missing data in white. In addition, the three largest objects are highlighted in pink color. Moving the mouse cursor over an object will show the object ID and its area in the information panel below the viewport. For example, in **Figure 8** the mouse cursor is over the pink object at the position 136/1685, the unique object ID is 14 and the area of object 14 is 73827 pixels, thus falling into size class 6, shown in green color but because it is one of the 3 largest objects it is shown in pink color.

Saving a Accounting image will produce the following files with the prefix <name>_accounting:
a) prefix.tif: Same graphics as displayed in GuidosToolbox
b) prefix.txt (only when saving the entire image): Table listing the number and area [in pixels] for the objects per size class (see **Figure 8**).

In addition, the user can select to add the following optional two files:
c) prefix_ids.tif: The unique identifiers of the objects in positive integer values and pixel values for missing (-4) and Background (0).
d) prefix_pixels.tif: The area in pixels for each object.

More information can be found in the Accounting Product Sheet.

Note: When running Accounting in batch-mode, no automatic scaling settings are calculated and the user-selected settings are applied to all input images regardless of their actual object area range and/or pixel resolution.

- **Parcellation:** On a given categorical map, Parcellation provides a summary of number, area, and some simple metrics for each class as well as the entire image (see **Table 2** below) with:
  $a_{i,j}$ = area of object *i* in class *j*
  n = number of unique objects in a given class
  A_j = total area of a given land cover class *j*
  A = image data area: all classes in the image = image – missing pixels

  Parcellation provides a statistical summary file (txt/csv-format) with details for each unique class found in the image: class value, total number of objects, total area plus the full image summary, (see **Table 2** below):
  - Class: Sequential number of individual land cover classes
  - Value: Land cover class value in the image
  - Count: Total number of patches in the given land cover class j
  - Area: Total area of a given land cover class j:
    A_j (Unit: area, measured in pixels)
  - APS: Average Patch Size of a given land cover class j:
    $\mathrm{APS} = \frac{1}{n} * \sum_{i=1}^{n} a_{i,j}$ (Unit: area, measured in pixels)
  - AWAPS: Area Weighted APS of a given land cover class j:

$\text{AWAPS} = \frac{1}{A_j} * \sum_1^n a_{i,j}^2$ (Unit: area, measured in pixels)

- AWAPS/data: AWAPS with respect to the entire data area A:
  $\text{AWAPS} = \frac{1}{A} * \sum_1^n a_{i,j}^2$ (Unit: area, measured in pixels)
- DIVISION: Degree of division of a given land cover class j:
  $\text{DIVISION} = 1 - S = 1 - \sum_1^n \left(\frac{a_{i,j}}{A_j}\right)^2$ (Unit: index in [0, 1])
- PARC: A logarithmic conversion and normalization of the Division complement $S = \sum_1^n (\frac{a_{i,j}}{A_j})^2$:
  a) Convert/constrain S into the range $R = [-\ln(1.0e^{-6}) \leq -\ln(S)]$
  b) Normalize: $Parc = \frac{100}{-\ln(1.0e^{-6})} * R$ (Unit: index in [0, 100] %)

The final line (**Table 2**) provides the overall summary for the entire image (top-left image in **Figure 9**), i.e.: The parcels were investigated using the 8-connectivity rule: there are 26 individual classes and 967 individual objects. The total area of all objects (= data area) is 4000000 pixels, the Average Patch Size (APS) is 4136 pixels, the Area Weighted Average Patch Size (AWAPS) is 450657 pixels, and the degree of division is 0.8873 corresponding to a Parcellation (PARC) for the entire image of 15.8%.

Class	Value	Count	Area[pixels]	APS	AWAPS	AWAPS/data	DIVISION	PARC[%]
1	1	2	2792	1396.0000	1998.3600	1.3948	0.2843	2.4207
2	2	201	432930	2153.8800	24287.2000	2628.6600	0.9439	20.8507
3	3	35	56446	1612.7400	3655.9400	51.5909	0.9352	19.8106
4	4	5	6131	1226.2000	1335.9000	2.0476	0.7821	11.0293
5	5	1	418	418.0000	418.0000	0.0437	0.0000	0.0000
6	6	2	14708	7354.0000	7779.0500	28.6036	0.4711	4.6105
7	7	14	10134	723.8570	783.5970	1.9852	0.9227	18.5281
8	8	3	1779	593.0000	701.8210	0.3121	0.6055	6.7325
9	9	6	6687	1114.5000	1979.0300	3.3084	0.7040	8.8130
10	10	15	25139	1675.9300	4503.9000	28.3059	0.8208	12.4460
11	11	28	36026	1286.6400	1773.8400	15.9761	0.9508	21.7950
12	12	61	2013006	33000.1000	875187.0000	440439.0000	0.5652	6.0291
13	18	139	278701	2005.0400	7531.4400	524.7550	0.9730	26.1377
14	20	48	80188	1670.5800	2995.8400	60.0575	0.9626	23.7932
15	21	53	77036	1453.5100	2042.6800	39.3399	0.9735	26.2749
16	23	108	308382	2855.3900	38797.6000	2991.1200	0.8742	15.0047
17	24	111	398163	3587.0500	29455.4000	2932.0100	0.9260	18.8483
18	25	93	118067	1269.5400	2976.5900	87.8591	0.9748	26.6402
19	26	4	6717	1679.2500	2505.7000	4.2077	0.6270	7.1374
20	27	1	675	675.0000	675.0000	0.1139	0.0000	0.0000
21	29	1	3828	3828.0000	3828.0000	3.6634	0.0000	0.0000
22	32	1	1022	1022.0000	1022.0000	0.2611	0.0000	0.0000
23	35	2	1577	788.5000	941.6470	0.3712	0.4029	3.7324
24	36	14	57507	4107.6400	22376.6000	321.7020	0.6109	6.8321
25	40	1	2092	2092.0000	2092.0000	1.0941	0.0000	0.0000
26	41	18	59849	3324.9400	32760.4000	490.1690	0.4526	4.3618
8-conn. Parcels:		967	4000000	4136.5049		450657.9072	0.8873	15.8036

Table 2: Example for tabular statistics of Parcellation.

The MSPA parameter 1 *FGconn* can be used to set the 4 or 8-connectivity rule for the image objects.

The input image for the Parcellation analysis must be a raster map with at least two land cover classes. A class value of 0 (zero) is reserved for masking no-data pixels.

In general, the value of Parcellation will increase with increasing number of feature classes, number of objects per class, and decreasing object areas.

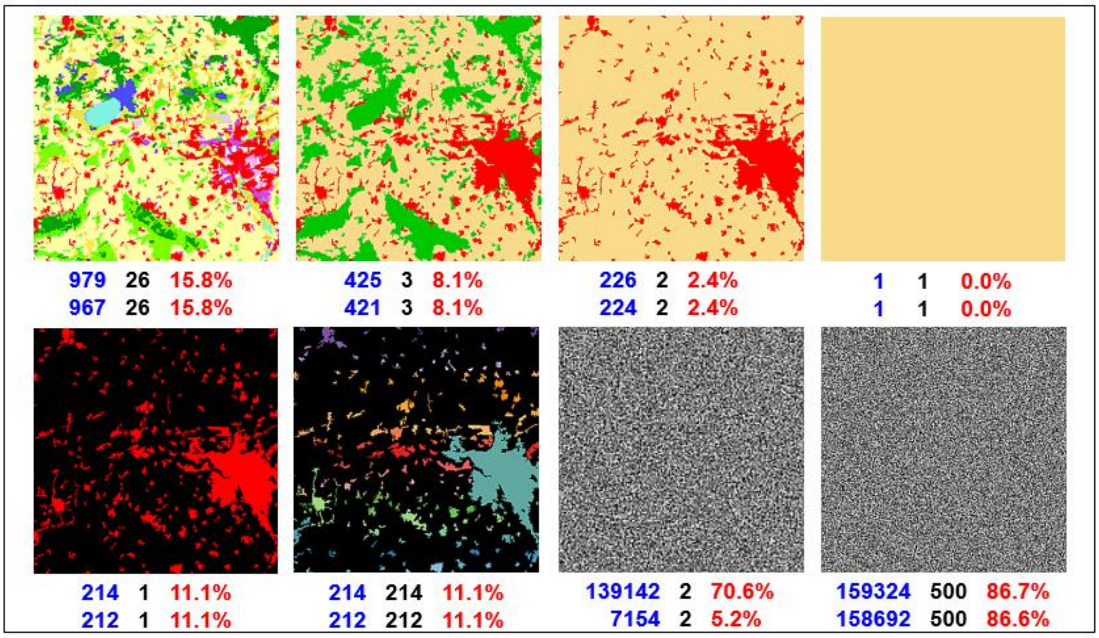


Figure 9: Example images showing: the number of image objects (blue), the number of individual classes (black) and the value of Parcellation (red). Results of the top (bottom) row below each image correspond to using the 4- (8-) connectivity rule for the image objects.

The left image of the top row in **Figure 9** shows a CORINE land cover image of dimension 2000^2 pixels having a total of 26 individual land cover classes. When using the 4-connectivity rule the image is composed of 979 parcels. When adding the diagonal direction of the 8-connectivity rule some neighboring parcels of the same land cover type become combined resulting in a lower total of 967 individual parcels. **Table 2** provides further details for the latter case on the relative Parcellation of each land cover class. The next three examples in the top row of **Figure 9** show the effect of joining the existing 26 land cover classes into 3, 2 and finally a single land cover class: the decrease of the total number of parcels and classes is directly related to the value of Parcellation and reaching the minimum at the lower boundary condition represented by a fully homogeneous land cover. However, this expected behavior is different when a substantial amount of missing pixels or no-data areas are present in the image, represented in the black color in the first two images of the bottom row in **Figure 9**: assigning all parcels into a single land cover class or each parcel into a different one should not, and in fact it does not change the Parcellation value of the image data area. This behavior is also evident from the Parcellation formula, which is simply building the sum of the relative contribution of each parcel over each land cover class. The third image in the bottom row of **Figure 9** shows a random distribution of two land cover classes generated for an image size of dimension 1024^2 pixels. The purpose of this image is to illustrate the potentially strong impact of applying the 4- or 8-connectivity rule, here resulting in a drastic difference in the total number of parcels and the Parcellation itself. Obviously, this behavior is very pronounced when only 2 classes are present, resulting in a high probability of the same land cover class being present in a diagonal neighborhood. The probability to encounter the same land cover class in

the direct neighborhood is far less with more classes present in the image. The final image in **Figure 9** confirms this thought showing a very similar number of parcels and Parcellation on an image of dimension 400^2 pixels and being composed of 500 individual classes in a random distribution. A further increase in the number of individual classes will obviously increase the value of Parcellation up to the boundary condition of 100% when each pixel has its own individual class value.
More information can be found in the [Parcellation Product Sheet](#).

- **Contortion:** Contortion describes the degree of regularity of a foreground object perimeter. Its value corresponds to the number of times a given object perimeter direction changes its sign.

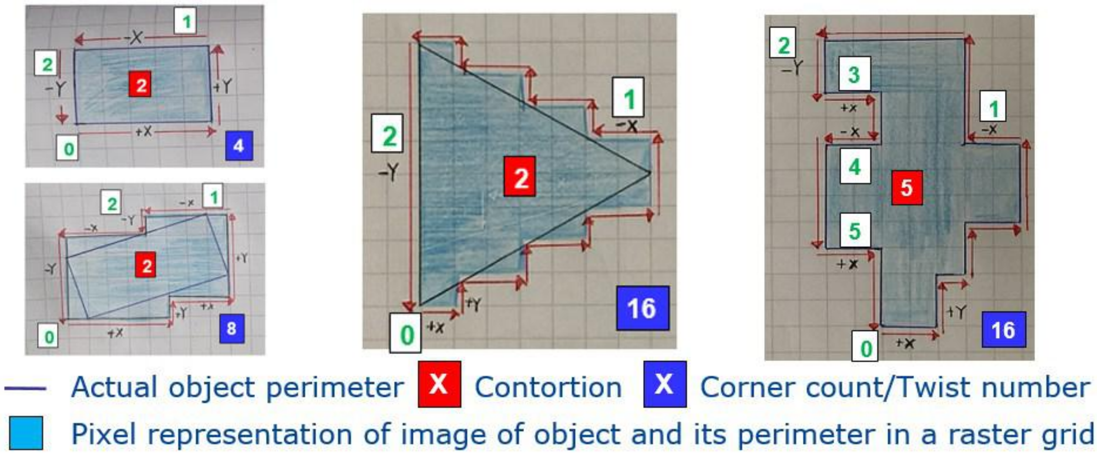


Figure 10: Contortion: measuring the number of directional changes in x-/y-direction.

Contortion can be used for shape description, in particular to distinguish random shapes from regular shapes. This is achieved by following the entire perimeter line, which includes the final angle at the starting point. **Figure 10** shows a sketch of the raster representation for a series of simple objects. Starting at the lower left border point, and following the entire perimeter outline of the object, the number of the directional changes in the X-/Y-direction are shown in white boxes. The final Contortion value for each object is shown in a red box while the number of perimeter corner counts is shown in a blue box. For example, when following the perimeter of a rectangle the direction changes one time in the horizontal and one time in the vertical direction: the contortion of a rectangle is 2, regardless of the orientation of the rectangle in the rectangular grid representation (see left part of **Figure 10**). Similarly, the contortion of the letter E is 6. An important feature of contortion is the invariance with respect to the length of diagonal lines. For example, two triangles oriented in the same direction but of different size will have a different number of pixel corners points due to the staircase representation along the diagonal lines. However, they both have the same contortion value of 2. The same is true for circles of different size. In satellite images, regularly shaped objects such as buildings and agricultural fields will have low contortion values while natural objects are more likely to be non-regular and hence have higher contortion values.

Since contortion investigates the object perimeter, the default calculation is set to exclude small or narrow objects where the perimeter will collapse into a line and contortion will be calculated only on objects having a core area. In addition, the user can define a minimum object area to be considered for the calculation via the MSPA parameter 2 *EdgeWidth* drop-down menu. Either select a pre-defined value or specify a custom value in the first entry of the drop-down menu. Pressing the Enter key will then apply the new custom minimum area value and recalculate the contortion. In the resulting image, small neglected objects are denoted as *omitted* and displayed in black color. The title bar shows the fraction of calculated versus total number of foreground objects in the image. The Divide Range panel below the GuidosToolbox viewport can be used to group the contortion value range into small/medium/large using either pre-defined or custom values in the respective threshold drop-down menu.

1.3.2 Pattern

This menu provides access to the following pattern analysis tools:

Morphological:
This option will process the user-selected file using the following options:

- **SPA3:** This option will process a binary input image resulting in a simplified version of the full MSPA with a map and statistics for three (3) morphological classes: The two foreground classes *Core* and *Margin* plus the background class *Core-Opening* (see **Figure 11**). SPA3 input images are *not* subject to the MSPA input image size limitations.
  The main purpose of SPA3 is to focus on Core, non-Core and Core-Openings only.

- **SPA5:** This option will process a binary input image resulting in a simplified version of the full MSPA with a map and statistics for five (5) morphological classes: The four foreground classes *Core, Edge, Perforation, Margin* plus the background class *Core-Opening* (see **Figure 11**). SPA5 input images are *not* subject to the MSPA input image size limitations.
  The main purpose of SPA5 is to provide more details compared to SPA3 by adding information on the internal and external Core-boundaries.

- **SPA6:** This option will process a binary input image resulting in a simplified version of the full MSPA with a map and statistics for six (6) morphological classes: The five foreground classes *Core, Edge, Perforation, Islet, Margin* plus the background class *Core-Opening* (see **Figure 11**). SPA6 input images are *not* subject to the MSPA input image size limitations.

The main purpose of SPA6 is to provide more details compared to SPA5 by adding information on isolated small fragments, which are too small to contain Core area.

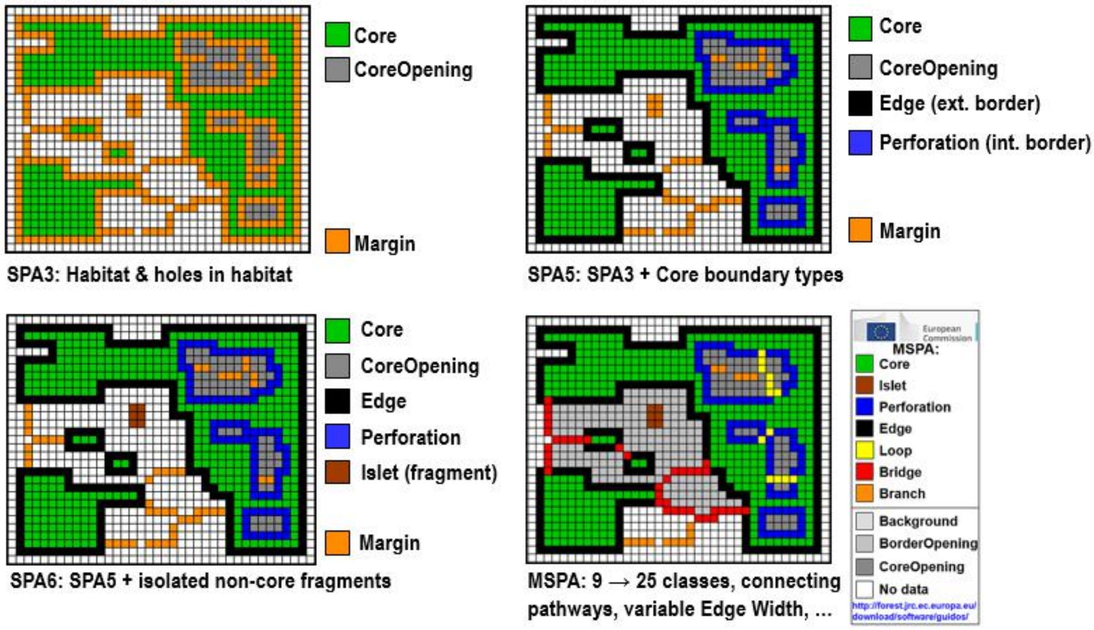


Figure 11: Spatial Pattern Analysis (SPA) including detection of holes inside Core and providing a total of 3-classes (top-left), 5-classes (top-right), 6-classes (bottom-left), or MSPA with 9 to 25 classes and additional fine-tuning options.

More information can be found in the Morphology Product Sheet.

- **MSPA:** This option will process a binary input image using the mathematical morphology algorithm described in Soille & Vogt, 2009. Compared to SPA3/5/6 the full version of MSPA provides many more morphometric details of the foreground patches as well as detecting holes within foreground but not inside Core. The 10 classes shown in **Figure 12** are only the visual result while the numeric result can provide details on up to 25 different morphometric classes. The full MSPA version has many additional features to fine-tune the analysis, for example, using the 4- or 8-connectivty rule for the foreground pixels, using an EdgeWidth larger than 1 pixel, showing transition pixels of connecting pathways and distinguishing background inside from background at the boundary of foreground. Further details on input requirements, processing options, resulting pixel values and morphological classes are summarized in the dedicated **MSPA Guide** or Online. This guide also provides information on the operating system specific MSPA input image file size limitations, likewise available in the Help → About GTB menu.

  Note: Selecting MSPA statistics will conduct summary statistics for the foreground pixels and the openings within foreground, more details can be found in the section MSPA Statistics.

  Standalone MSPA-plugins with documentation and installers are also available for ArcGIS, R, and QGIS from the MSPA-website.

The main purpose of MSPA is to provide additional details on connecting pathways, branches, and internal features and a user-selectable, scale-specific spatial pattern analysis.

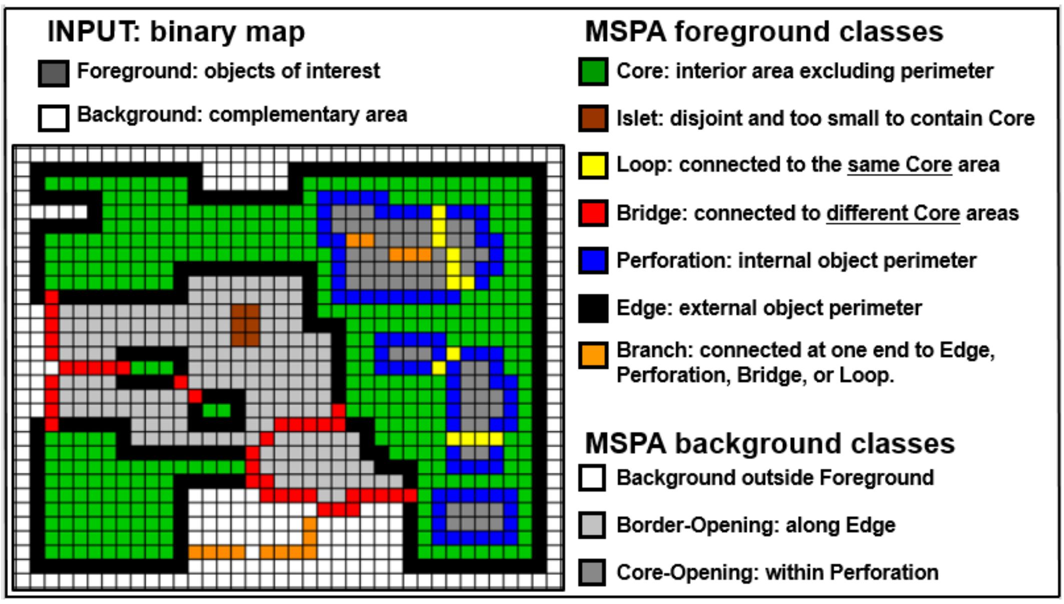


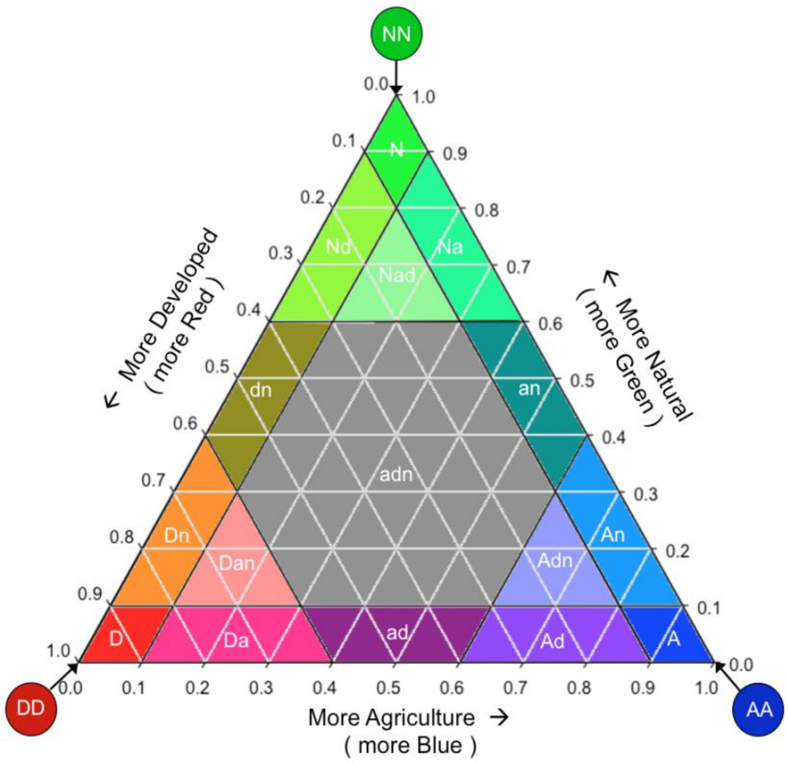
Figure 12: MSPA: Overview of the various foreground and background MSPA pattern classes derived from a binary input mask.

- **MSPA Tiling:** This option will perform a MSPA-analysis with an automated buffered tiling of a single image which is larger than the standard maximum file-size (MS-Windows: 10000^2 pixels; Linux/MacOS: Larger than 10000^2 pixels, depending on the available free RAM). After verifying MSPA-compliance, a window will open, where the MSPA parameters can be specified. No statistics will be calculated for these large images.

Moving Window:
This menu provides access to several moving window algorithms, implemented via the generic image-convolution program *spatcon.c* (K. Riitters, personal communication, 2015; and demonstrated in earlier papers: Riitters et al. 1997, 2000, 2002, 2009). Via an initial pop-up window, the user can specify the size of a square (kernel) window, which is then overlaid over each pixel of the input image, the selected metric is calculated for the area of the window, and the result is re-assigned to the center pixel of the overlaid window in the output image. All texture indices are derived from analyzing the attribute adjacency table (Musick and Grover 1991), where Fij (i,j = 1 to t) is the frequency of adjacent pixel pairs with land-cover types {i,j}. When forming the attribute adjacency table, adjacency is evaluated in the four cardinal directions, each edge is counted once, the order of pixels in pairs is not preserved, and pairs involving a missing pixel are not included (Riitters et al. 1996b). The metrics P2 and LM are calculated from the proportions of cell values in the window. For those metrics, missing cell values are not included in the calculation, and the calculation result is missing if all cells in the window

are missing. The metrics P22 and P23 are calculated from the cell|cell adjacency values (edge) proportions in the window. We define *N* as the total number of edges between all pixels in cardinal directions, and the subset *n* as the number of edges that have foreground on one side or the other. All edges (N-n) that do not have foreground on either side are excluded in the metric calculation. If there are missing cells in the (kernel) window, the edges involving missing cells are not included. As a result, the total number of edges is less than *N* and the total number of edges involving foreground may be less than *n*, if missing cells are adjacent to foreground cells. P22 is class-level contagion (one row of the attribute adjacency table), while Shannon/SumD is landscape-level contagion accounting for all classes in the image (the entire attribute adjacency table). The following moving-window processing options are available:

- **LM** (Landscape Mosaic, **Figure 13**): A tri-polar classification of a location accounting for the relative proportions of three classes in the window surrounding that location. The classification model uses the critical values of 10%, 60%, and 100% along each axis to partition the tri-polar space into 19 mosaic classes. More details on the concept of Landscape Mosaic and application examples can be found in Wickham et al. (1994) and Riitters et al. (1996a, 2000, 2009).



Value [byte] - Class Name	Color	RGB
0 – Missing		255/255/255
1 – A		000/000/255
2 – D		255/000/000
3 – N		000/255/000
4 – Ad		128/000/255
5 – An		000/128/255
6 – Dn		255/128/000
7 – Da		255/000/128
8 – Na		000/255/128
9 – Nd		128/255/000
10 – Adn		128/128/255
11 – Dan		255/128/128
12 – Nad		128/255/128
13 – ad		128/000/128
14 – an		000/128/128
15 – dn		128/128/000
16 – adn		128/128/128
17 – NN		000/191/000
18 – AA		000/000/191
19 – DD		191/000/000

Figure 13: The Landscape Mosaic (LM) tri-polar classification scheme showing 19 mosaic classes and their proportions of the 3 land cover categories *Natural*, *Agriculture*, and *Developed*. A capital letter denotes more than 60% and a small letter a contribution in the range of [10-60]% of a given land cover category.

The result of the moving window analysis is a geographic image showing the 19 landscape Mosaic classes for each pixel at the user-selected observation scale (**Figure 14** left). A second output shows a statistical summary as a heatmap (image and csv-file). The statistical summary is for an expanded, 103-class version of LM. In the expanded version, each of the 100 sub-triangles in **Figure 13** is a separate class, and three additional classes correspond to the external circles (AA, DD, NN in **Figure 13**). The heatmap image (**Figure 14**, right) shows the relative frequencies of pixels in each of the 103 categories (see also **Figure 15**). The accompanying csv-file contains the numeric values from the heatmap image.

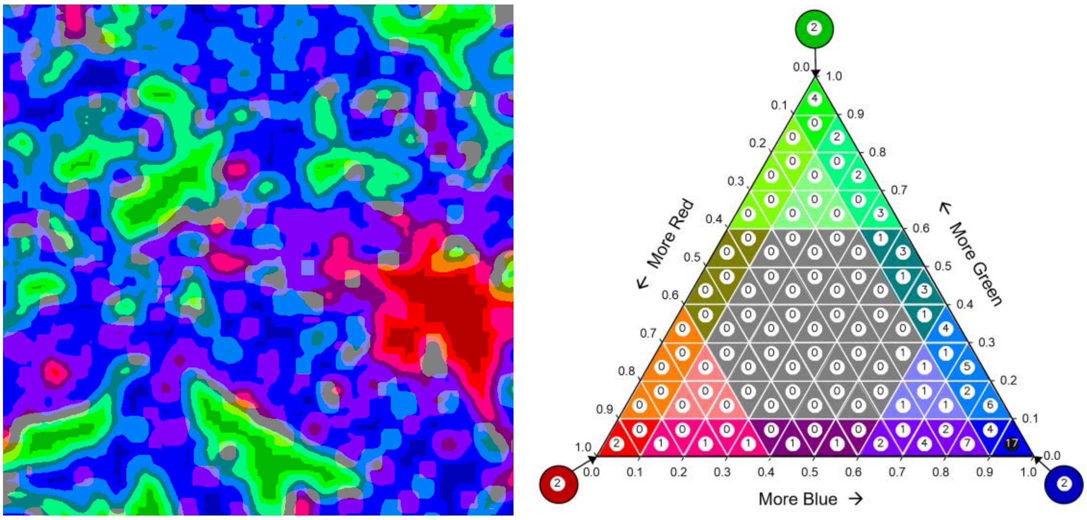


Figure 14: The Landscape Mosaic (LM) image (left) and corresponding heatmap (right) showing LM-pixel occurrence frequencies. The maximum frequency is highlighted in a black circle. In this example, 21% (i.e., 17% plus 4%) of all LM-pixels are in the Mosaic class A (Agriculture).

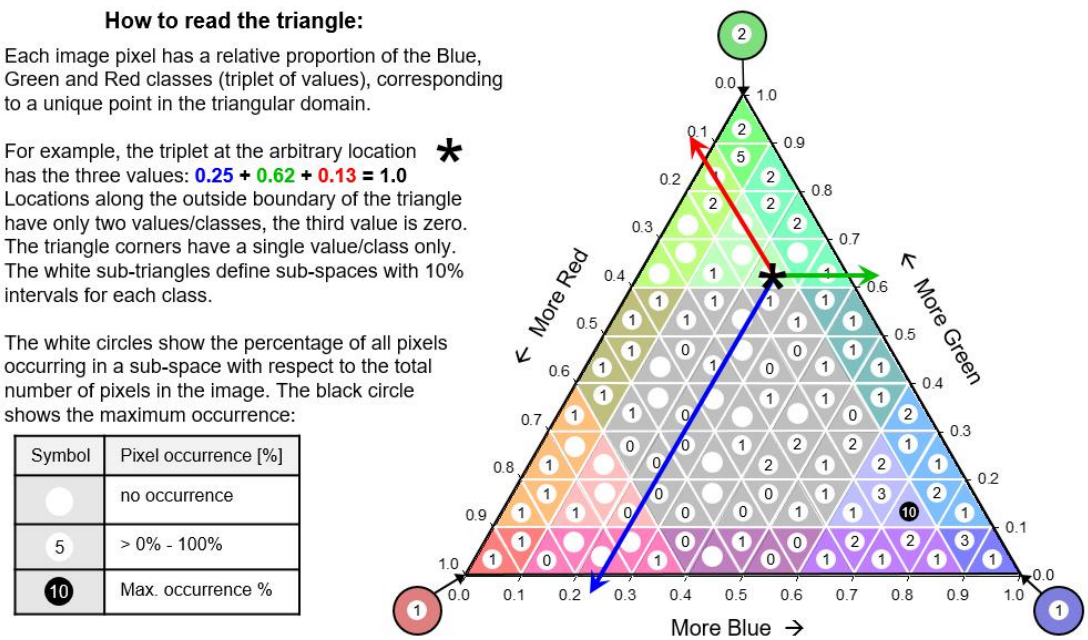


Figure 15: Explanatory chart to read out Landscape Mosaic pixel value triplets and to locate these triplets in the ternary domain space.

The input image for LM must be a Byte array with no more than 3 target classes with the assignment **AND** (1-**A**griculture, 2-**N**atural, 3-**D**eveloped) plus an optional class for missing values (0 Byte).

More information can be found in the LM Product Sheet.

Note: GuidosToolbox provides the additional option Dominance, which combines LM-analysis over a set of five observation scales into a summary layer showing the dominant triplet of land cover type across multiple observation scales.

- **P2:** The proportion of foreground pixels (2b) in the moving window. More details on the concept of P2 and P22 (below) and application examples can be found in Riitters et al. (1997, 2002). Click Intext for FG-masking and Divide for range grouping.

  The input image for P2 must be *MSPA-compliant*: A Byte array having 0b – missing (optional), 1b – background, 2b – foreground.

- **P22:** The proportion of adjacent pixel pairs in cardinal directions that include at least one foreground pixel, for which both pixels are foreground (2b|2b). Note: If n = 0, then P22 is missing. P22 estimates the conditional probability that, given a foreground pixel, its neighbor is also foreground. Click Intext for FG-masking and Divide for range grouping.

  The input image for P22 must be *MSPA-compliant*: A Byte array having 0b – missing (optional), 1b – background, 2b – foreground.

- **P23:** The proportion of adjacent pixel pairs in cardinal directions that include at least one foreground pixel (2b), for which the neighboring pixel is *interesting background* (2b|3b). Here, the original data background (1b) is subdivided into non-interesting background (1b) and interesting background (3b). A P23 map shows foreground (2b) fragmented by *interesting background (3b)*. More details on the concept of P23 and application examples can be found in Wade et al. (2003) and Riitters et al. (2012a). In a similar fashion one can divide the background not into two but into *x* subclasses. Then setup a loop, recode the input map accordingly with the interesting background subclass *x* set to 3b, and run P23 in a loop for each of these recoded maps in order to retrieve the impact of each background subclass *x* on the foreground class. For example, a land-cover map may contain the four classes Forest, Developed, Agriculture, and Water. If we choose Forest as foreground, then we have: $1=P_{FF}+P_{FD}+P_{FA}+P_{FW}$. Here, P_{FF} is the proportion of pixel pairs having Forest-Forest (P22 in GuidosToolbox), and the other 3 components describe the proportions of Forest to Developed, Agriculture, and Water, respectively. These 3 components (P23 in GuidosToolbox) then describe how the foreground (Forest) is fragmented by each of the remaining background land-cover types. Click Intext for FG-masking and Divide for range grouping.

Note: If all background is interesting, then P23 = (1 – P22). By taking different subsets of background as *interesting*, it is possible to partition the total fragmentation (1 – P22) into components attributable to different background classes.

The input image for P23 must be a Byte array having the classes:
- Missing, 0b (optional),
- Non-interesting background, 1b (optional),
- Foreground, 2b (mandatory),
- Interesting background, 3b (mandatory).

- **Shannon:** Shannon edge-type evenness, the classical overall contagion measure used in Landscape Ecology literature, see Li and Reynolds, (1993).
The input image must be a Byte array with optional missing pixels set to 0 byte.

- **SumD:** An alternative contagion measure is the sum of the main diagonal ("same-class") elements of the attribute adjacency matrix; see Wickham and Riitters (1995) and Riitters et al. (1996). In contrast to the Shannon index, this index is not affected by adjacencies among different classes, and the same value is obtained for all of the ways pixel pairs could be tallied. The input image must be a Byte array with optional missing pixels set to 0 byte.

1.3.3 Fragmentation

Fragmentation can be seen as the spatial heterogeneity, or the spatial composition and arrangement of foreground objects in an image. It accounts for the number of objects and the distance between them, hence addressing foreground and background characteristics at the same time. Due to its holistic nature the description of fragmentation is rather complex and, in the case of landscapes, usually defined for a given species of interest and as such very specific. Moreover, current fragmentation definitions are only descriptive and for this reason do not allow *quantifying* the degree or changes of fragmentation for a given image. For quantifying fragmentation, we apply different concepts. All provide normalized values in the range [0 – 100] %. The Entropy requires a minimum image dimension of 500 pixels in any direction. In the Divide Range panel below the viewport, the fragmentation values can be grouped into small/medium/large. The title bar in GuidosToolbox shows fragmentation values for the entire image, the foreground only, and the range of minimum-maximum values.
The aim is to *describe* fragmentation using different concepts. Based on the nature of the selected approach each concept will provide different results, focusing on the user-selected aspect of fragmentation. Besides *quantifying* the fragmentation state on a given map these measures permit the comparison of the degree of fragmentation of different sites, the *quantification*

of changes in fragmentation over time, and *monitoring* as well as *measuring progress* in planning programs and political directives.

Index:

The Fragmentation submenu ***Index*** provides methodologies resulting in a normalized fragmentation value (index) for the entire image.

- **Entropy:** This option will calculate the **Entropy** of the entire image.

- **Hypsometry:** The *Normalized Hypsometric Curve* (NHMC) is the Hypsometric Curve (**Figure 21**) scaled by the maximum distance in the foreground and in the background.

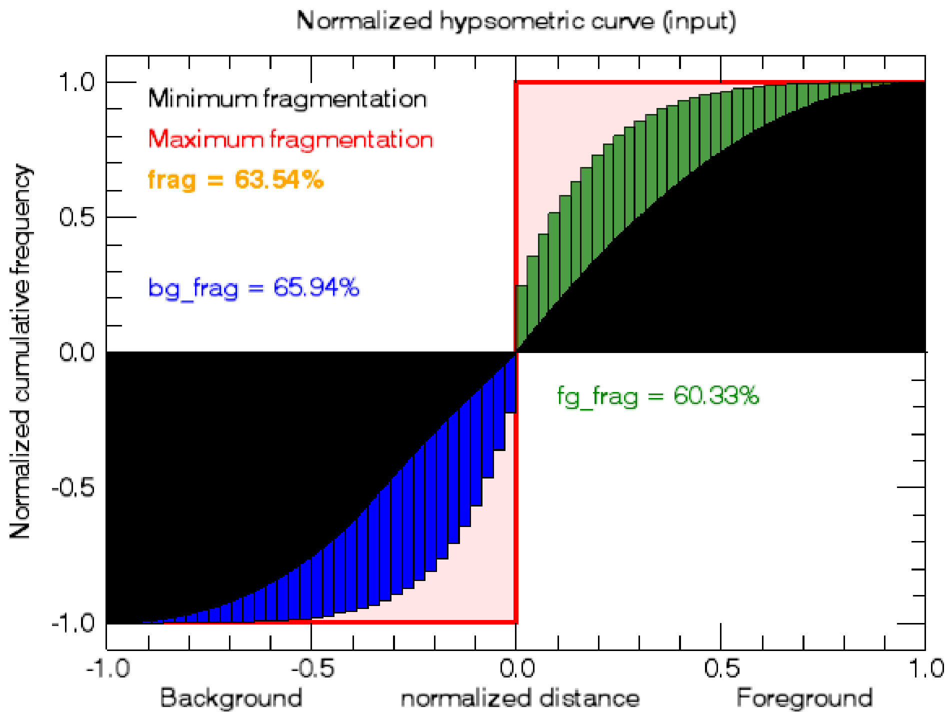


Figure 16: Normalized Hypsometric Curve for minimum (black), maximum (red), and actual fragmentation state in the foreground (green) and in the background (blue).

Figure 16 shows the NHMC for three images and the following three conditions:

- minimum fragmentation (black): NHMC of an image with maximum foreground aggregation: All foreground pixels accumulated to a circle.
- actual image (blue/green): NHMC of the actual image
- maximum fragmentation (red): NHMC of a checkerboard image of the same size as the actual image and with 50% coverage. This theoretical maximum condition for fragmentation is characterized by all foreground as well as all background pixels having a distance of 1 and thus resulting in the step-function outlined in red in **Figure 16**.

For a given image, the degree of fragmentation (Kozak et al. (2018)) corresponds to the area covered between minimum fragmentation (black) and maximum fragmentation (red). In **Figure 16**, this area is highlighted in blue for the background and in green for the foreground. Accounting for the dual nature of fragmentation (foreground is fragmented by background and vice versa) the degree of fragmentation for a given image is defined by the weighted sum of fragmentation in the foreground and the background:

Frag (hypso) = (bg_area/100.0 * bg_frag) + (fg_area/100.0 * fg_frag)

The so-defined fragmentation provides values in the range of [0, 100]%, accounting for and summarizing key fragmentation aspects: Duality, perforations, amount, division, and dispersion of image objects.

Note: Both fragmentation indices described here are provided for historical reasons. They are now superseded by the Map/MultiScale-products, in particular the **FAD/FOS** approach.

Map:

The Fragmentation submenu ***Map*** provides methodologies resulting in spatial distribution of normalized fragmentation values.

- **Entropy**: In thermodynamics, Entropy describes the degree of disorder in a system. Transferring this concept into spatial geometry (raster images), we can use entropy as a descriptor for spatial fragmentation. Starting from the classical definition of entropy in information theory H = -∑ Pi * log(Pi) (Shannon, 1948) we define the discrete set of probabilities *Pi* as the probability that the difference between 2 adjacent pixels is equal to *i* and log is the base 2 logarithm. The original entropy definition has been implemented in many ways and it is important to distinguish the above definition of Pi from other commonly used indicators such as Shannon's diversity index or the Eveness index (where Pi is the proportion of *species*) and variations of contagion indices (where Pi is the proportion of different type of pixel edges). In short, in Shannon's original concept Pi refers to percentages of species classes in categorical maps, as defined in the species diversity literature. In contrast, here we investigate differences between cell values in all 8 directions (that is, the values of i), which is meaningful because raster images are continuous variables where their magnitude has meaning. While the entropy in the edge-type evenness is derived from the attribute adjacency table, the spatial entropy here is calculated on spatial tiles and assuming 8-connectivity for the foreground pixels.
  For a given amount of foreground area, an image with a single compact foreground object has minimum entropy while the entropy reaches its maximum value when the given area is split into the maximum number and dispersed equally over the entire image. Maximum entropy is thus found

for a checkerboard distribution. These two boundary conditions define the possible range of fragmentation in the image. The spatial entropy is calculated by averaging calculations using box size tiles of 50 and 33 starting from the center of the image. This approach replaces the precise but much slower performing moving window computation. Finally, a smoothing filter is applied in order to return a spatial contiguous per pixel distribution. The result shows the normalized fragmentation as a function of spatial entropy.

- **Contagion**: Contagion describes the degree of clumpiness of image objects. With this definition, fragmentation can be defined as the complement of contagion (1 – contagion); an image region with high contagion is equivalent to having low fragmentation. The contagion of foreground objects is calculated via the moving window metric P22 and box size 49, providing a statistically meaningful data sample. The result shows the normalized fragmentation as a function of spatial contagion.

  Both, Entropy and Contagion can both be seen as local aggregation measures but with an important difference: While Contagion will consider the Foreground objects only, the Entropy based fragmentation assessment is based on the *simultaneous* evaluation of Foreground and Background together (duality). For example, an image with predominant Background cover (i.e. 95%) and a few isolated Foreground objects (5%) will result in high fragmentation values for Contagion. For Entropy, this image will have low fragmentation values because the dominant area coverage (Background) is only slightly fragmented by the Foreground. In fact, we will get the same low fragmentation value for Entropy when using the inverted image with 95% Foreground cover and 5% background; in this case, the dominant area coverage is now *Foreground*, which is only slightly fragmented by the *Background*. In short, Entropy derives Fragmentation for the interplay of Fore- and Background while Contagion will focus on the Foreground only.
  Note: Entropy and Contagion are provided for historical reasons. They are now superseded by the **FAD/FOS** approach.

- **FOS:** The Fixed Observation Scale (FOS) analysis calculates fragmentation as a function of the foreground area density, equivalent to **FAD** but at a user-selected observation scale. In this module, the user can define the observation scale by specifying the pixel resolution (in meters) and the edge length of the square moving window in a GUI:

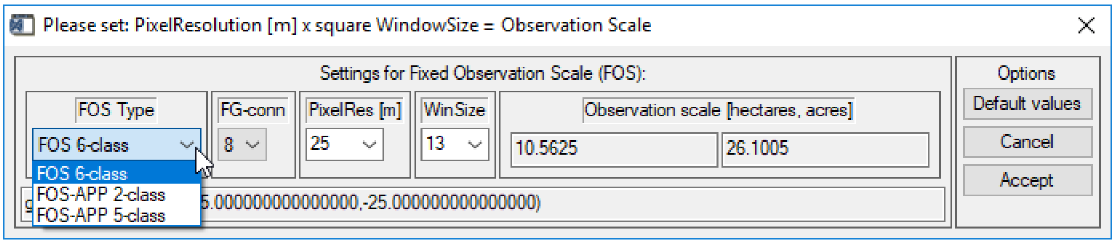


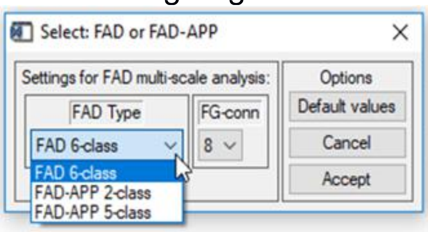
Figure 17: FOS interface: Set or select the spatial resolution of the data and the moving window size to define a Fixed Observation Scale for the fragmentation analysis.

The selected settings are then used to calculate the area of the moving window (observation scale), which is displayed in hectares and acres. The bottom panel of the GUI provides information on the pixel size retrieved via the *gdalinfo* command. Please note that this information is not automatically used for calculating the observation scale but displayed only as a guide for the user. The GDAL reported pixel size may be in meters or degrees and can only be retrieved from properly formatted geotiff headers. The image above shows a combination of pixel resolution and window size resulting in an observation scale of ~10 hectares. The user can choose settings from the drop-down menu or insert custom values, followed by the Enter key. After the definition of the analysis type and the user-selected observation scale, click on *Accept* to apply this scale for the fragmentation analysis.
More information can be found in the FAD/FOS Product Sheet.

MultiScale:

The submenu ***MultiScale*** provides methodologies conducting a fragmentation analysis at a variety of different observation scales.

- **FAD**: The Foreground Area Density (FAD) is constructed by measuring **P2** over five observation scales using a moving window analysis with square neighborhood areas of length 7, 13, 27, 81, 243 pixels and applying foreground masking (Riitters et al., 2002, 2012a, 2012b). The result is a set of five maps (one for each observation scale) showing the P2/FAD values for each foreground pixel. The resulting maps list the FAD (density) values either per-pixel (FAD 6-class) or the Average-Per-Patch (FAD-APP) per patch. For visual clarity, the FAD values are displayed color-coded into the following fragmentation classes:



FAD 6-class	Color	FAD range
1-Rare		FAD < 10%
2-Patchy		10% ≤ FAD < 40%
3-Transitional		40% ≤ FAD < 60%
4-Dominant		60% ≤ FAD < 90%
5-Interior		90% ≤ FAD < 100%
6-Intact		FAD = 100%

FAD-APP 5-class	Color	FAD range
1-Rare		FAD < 10%
2-Patchy		10% ≤ FAD < 40%
3-Transitional		40% ≤ FAD < 60%
4-Dominant		60% ≤ FAD < 90%
5-Interior		90% ≤ FAD ≤ 100%

FAD-APP 2-class	Color	FAD range
1-Separated		FAD < 40%
2-Continuous		40% ≤ FAD ≤ 100%

Figure 18: Summary of FAD/FAD-APP fragmentation class thresholds, names, and color assignment. FAD is a per-pixel classifier while FAD-APP summarizes the average density value per patch.

The statistical summary (**Figure 19**) lists the proportion of each fragmentation class and scale. The map legend and the aggregated results for the entire image are displayed as a cumulative bar chart. The observation scales 1, 2, 3, 4, 5 correspond to the window lengths 7, 13, 27, 81, 243 pixels.

```
FAD: Foreground Area Density summary analysis for image:
C:\GuidosToolbox\data\NFinland_A.tif
================================================================================
8-conn FG: area, # patches, aps [pixels]: 1099835.0, 600, 1833
Fragmentation class: foreground proportion at observation scale/area:
Observation scale:     1         2          3          4          5        mscale
Neighborhood area:    7x7      13x13      27x27      81x81    243x243
================================================================================
         Rare:      0.0033     0.0242     0.2450     0.3786     0.3976     0.0186
       Patchy:      1.9327     3.5649     4.9045     7.1666     9.1789     2.8774
 Transitional:      7.5076     8.9589    11.3673    13.3002    15.8747    10.3957
     Dominant:     24.1627    36.2559    45.5856    60.1687    71.3635    58.0553
     Interior:     14.1438    21.5234    28.4999    18.8959     3.1853    28.6529
       Intact:     52.2498    29.6727     9.3977     0.0900     0.0000     0.0000
================================================================================
```

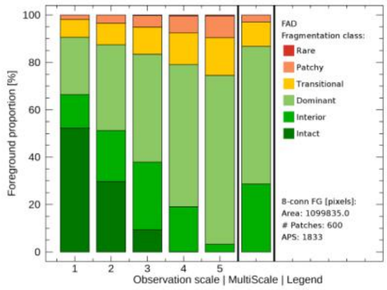


Figure 19: Statistical summary table and bar-plot example for FAD 6-class.

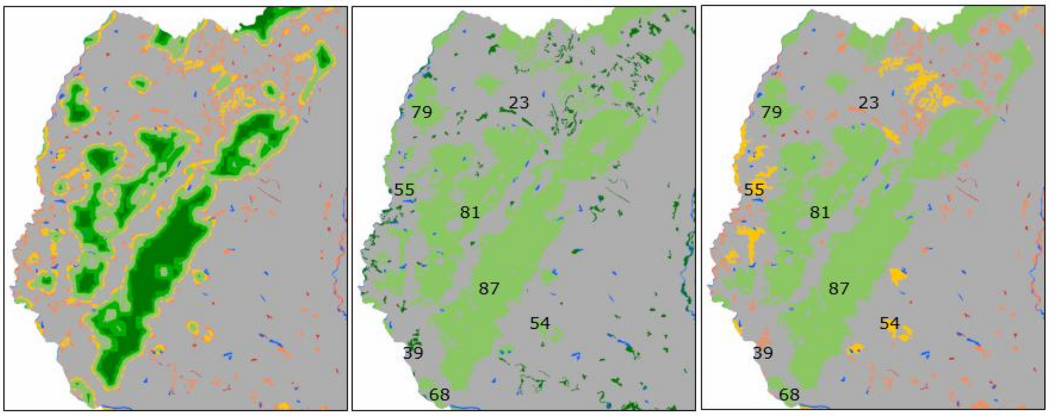


Figure 20: FAD versus FAD-APP: Forest density in the western part of Slovakia, using CORINE 2012 data and an observation scale of 23x23 pixels ~ 529 ha.
Left: FOS 6-class: per-pixel density and the default 6-class FAD color-table.
Center: FOS-APP 2-class: per-patch average density and 2-class FAD-APP color-table.
Right: Same as center panel but applying the FAD-APP 5-class color-table. The center and right panel include average per patch (APP) density values for some selected patches.

Figure 20 shows the difference between the per-pixel FAD and the per-patch FAD-APP analysis. By design, a per-pixel moving window assessment scheme will assign lower density values to pixels along the patch boundary. This feature may be of interest to differentiate intact areas from less intact areas inside larger patches. Users preferring to see a single and constant density value for each patch should select the option FAD-APP. Here, the density values of all pixels of a given patch are used to calculate the Average density Per Patch (APP), which is then re-assigned to all pixel values of the given patch. In addition, and with APP being designed as a more simplified version of the original FAD/FOS, the number of color-classes is reduced to either five, or two using a threshold of 40% in density to distinguish Separated (fragmented, dark-green) from Continuous (non-fragmented, light-green) foreground cover (**Figure 18**); a setup chosen by [Forest Europe](#) for the definition of indicator 4.7 Forest Fragmentation. As with FAD/FOS, the APP option provides actual density values in [1, 100] %, just averaged by patch.
More information can be found in the [FAD/FOS Product Sheet](#).

The viewport image shows the multi-scale FAD assessment: the average FAD value across all 5 observation scales, which is displayed color-coded into the user-selected fragmentation classes. When moving the mouse cursor into the viewport, the information panel below the viewport will display the cursor coordinates followed by the multi-scale FAD value and the sequence of FAD values across the 5 observation scales.

The input image for the FAD analysis must have the values 1 byte for background and 2 byte for foreground. All background data pixels will fragment the foreground. Optionally, the input can have the values:
0 byte – missing data. These pixels will show in white color in the output image.
3 byte – specific background. These background pixels (e.g. water) will show in dark blue color in the output image.
4 byte – non-fragmenting background. In contrast to all other background pixels, these background pixels will not have a fragmenting effect on the foreground. These pixels will show in light blue color in the output image.

Note: Because input data pixels with 4 byte do not have a fragmenting effect, changing background to non-fragmenting background will lead to a reduction of foreground fragmentation and consequently change the fragmentation statistics.

The FAD analysis can be saved as a single multi-scale image when using the option Display Snapshot or KML. When using the option GeoTiff or Generic the full result is saved into a directory having the following files:

a) <name>_FAD_{7/13/27/81/243}.tif: FAD-values at 5 observation scales
b) <name>_FAD_barplot.png: FAD-bar chart summary
c) <name>_FAD_mscale.txt: FAD summary statistics
d) <name>_FAD_mscale.csv: FAD summary statistics in comma-separated value format for import into a spreadsheet application
e) <name>_FAD_mscale.tif: FAD summary multi-scale image

The resulting images have the following dedicated value assignments:
1-100 byte: Foreground density
101 byte: Background pixels (gray color)
102 byte: Missing pixels (white color)
105 byte: Specific background pixels (dark blue)
106 byte: Non-fragmenting background pixels (light blue)

Saving a zoomed area is disabled. When saving a FAD analysis all saved files refer to the entire extent of the original input image.

Note:
1. The five observation scales are arbitrary; they were selected to span a wide range of scales (window area) and to approximate a geometric progression of window areas with scale.
2. Please use the **FOS** algorithm to conduct a single-scale FAD fragmentation analysis at a user-specified scale.

3. The results displayed in the viewport may suggest that FAD may also be useful for labeling different morphological parts of the foreground patches (Riitters 2005). However, such labeling is better accomplished using the MSPA algorithm.

- **Dominance:** Dominance is constructed by measuring the **Landscape Mosaic** (LM) over five observation scales using a moving window analysis with square neighborhood areas of length ks = 7, 13, 27, 81, 243 pixels and applying foreground masking. The result is a set of five maps (one for each observation scale) showing the LM values for each foreground pixel. The LM values on each map are grouped into a summary LM that shows the same 19 mosaic classes as a across-scale analysis. The summary value for a given pixel is the result of applying the tri-polar LM-classification to the average values of each of the three variables in the tri-polar model.

  The input image for Dominance must be a Byte image with no more than 3 target classes and the assignment **AND** (1-**A**griculture, 2-**N**atural, 3-**D**eveloped) plus an optional class for missing values (0 Byte). Dominance can be saved as a single multi-scale image when using the option Display Snapshot or KML. When using the option GeoTiff or Generic the full result is saved into a directory having the following files:

  a) <name>_lm_ks.tif: LM-image at 5 observation scales

  b) <name>_lm_ks.csv/png/sav: LM-heatmaps at 5 observation scales

  c) <name>_lm_mscale.tif/csv/png/sav: LM-image and heatmaps across 5 observation scales

  d) heatmap_legend.png: explanatory chart, **Figure 15**

1.3.4 Distance

This menu provides several options for distance analysis.
Note: When using geotiff data the distance value in pixels is only meaningful for images having **equal-area projection**!

- **Euclidean Distance:** This option calculates the approximate Euclidean distance map for both, the background and the foreground. If the Divide button below the viewport is not active, a click on the Intext button will trigger the display of two summary graphs:
  a) Distance histogram: including foreground and background specific information on average distance (adf/adb), total number of objects (fgo/bgo), and maximum distance found in the image. Here, Background data have a negative sign to distinguish them from foreground data. **b) The Hypsometric Curve:** the normalized cumulative frequency of the Euclidean distance histogram:

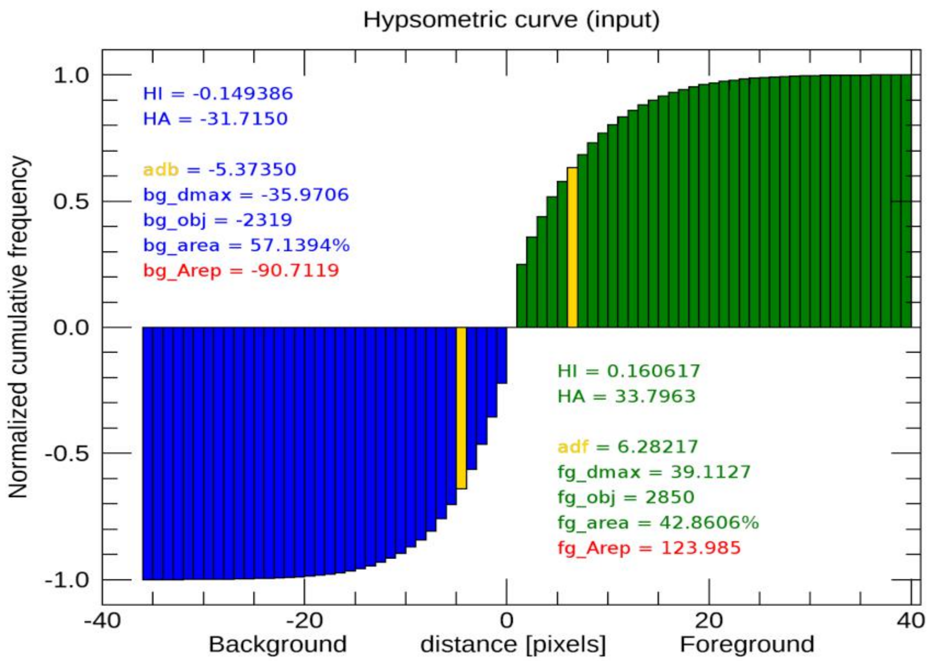


Figure 21: Hypsometric Curve for the Euclidean distance distribution in the foreground and in the background and related distance measures.

When viewing the image distance distribution as a pseudo-elevation map the Hypsometric Curve summarizes the relief or contour lines in the foreground, the same process is known as bathymetry in the background. **Figure 21** shows the normalized cumulative frequencies of Euclidean distances as well as the following related statistics for the background and the foreground:

- *Hypsometric Index (HI)*: adb/bg_max   or   adf/fg_max
- *Hypsometric Area (HA)*: Integral area under the curve
- *Average distance (adb/adf)*
- *Maximum distance (bg_max/fg_max)*
- *Number of objects (bg_obj/fg_obj)*
- *Total area of objects (bg_area/fg_area)*
- *Representative area (bg_Arep/fg_Arep)*

The *Representative Area* is calculated for a vicarious object of circular shape with a radius of adf/adb. In a similar way, a representative square object could be defined having the edge length sqrt(Arep). As with the *Average Distance* (adf/adb), changes in the *Representative Area* are indicative for fragmentation processes.

The GuidosToolbox title bar shows adf/adb and the spatial distance distribution is displayed in the viewport. For each pixel, this map shows the shortest distance to the nearest foreground/ background boundary. The distance is provided in pixel units since the actual spatial pixel resolution is unknown. The color code is designed to mimic a pseudo elevation map: Blue colors represent ocean (background), yellow/red/green colors

represent land or mountains (foreground), and a value of zero is assigned to the coastline (intersection of foreground/background). Saving a Euclidean distance image will save the actual distance image as well as the histogram and the HMC bar plot images. As in any other image, the zoom function can be used to retrieve more detailed pixel information via the mouse cursor.
More information can be found in the Distance Product Sheet.

- **Influence Zones:** An influence zone is defined as the outside equal distance delimiter line (iso-distances) separating selected foreground objects. The boundary of an influence zone is derived by applying a morphological watershed operator to the Euclidean distance map of the background area in the image. Considering the gray scale (8-connected Euclidean distance) image as a surface, each local minimum can then be thought of as the point to which water falling on the surrounding region drains. The boundaries of the watersheds lie on the tops of the ridges. Small objects in the original image can produce spurious minima in the gradients, which leads to over-segmentation. For this reason, the default minimum object size is set to 5000 pixels. Omitted objects smaller than the minimum object size are displayed in pale blue color. Objects for which influence zones are calculated are displayed in alternating colors. Potential missing data is displayed in white and the influence zone boundaries (watershed lines) in black color. The information panel below the viewport allows setting buffer zones for both, foreground and background. In summary, the influence zones provide a segmentation of the background and buffer zones can be added to define *core areas* as well as *outreach zones* of any size.

  Influence zones are driven by the following 3 parameters:
  1. Minimum area: Influence zones are calculated for foreground objects larger or equal to the specified minimum area in pixels. The threshold of small objects to be excluded from the calculation can be set to any value via the MSPA parameter 2 *EdgeWidth* drop-down menu. You can either select a pre-defined value or specify a custom value in the first entry of the drop-down menu. A new custom value will only be assigned after the Enter key has been pressed.
  2. FG Buffer zone: A non-zero value for the foreground buffer zone corresponds to the perimeter width of the foreground objects to be excluded from the calculation. Consequently, this parameter can be used to define core-foreground objects. In this case the title-bar will show, and the calculation will be conducted for foreground *core* objects having the minimum area specified via the MSPA parameter 2 *EdgeWidth.*
  3. BG Buffer zone: A non-zero value for the background buffer zone will add a dark grey colored buffer zone of the specified width around, and if sufficiently large holes are present, inside the selected core or foreground objects. The boundary of the background buffer zone is depicted in pink color. Background buffer zones will terminate at the boundary of the influence zones.

After changing any of the three influence zones parameters the *Divide* box under the viewport is changed to Off. Tick the *Divide* box to On in order to apply any new influence zone settings.
Clicking the *Divide* box switch to Off will reset both, the foreground and background settings to their default value of zero, omitting buffer zones and showing influence zones only.
Note: A minimum object area value of one (1) will calculate influence zones for *all* foreground objects. Since influence zones are defined to describe the **outer** region of objects they are calculated for **filled foreground objects**. For this reason, objects insides holes of surrounding objects will have the same object ID and hence, the total number of objects when calculating influence zones may be smaller compared to the total number for object labeling.
Saving an influence zone image will produce the following two files with the prefix <name>_influence_<Y>_<b1>_<b2>_<zoom>, where Y stands for the selected minimum object size in pixels, b1 and b2 stand for the selected foreground and background buffer zone width, respectively, and the suffix *zoom* will be added if a subregion of the image is saved:
a) prefix.tif: Same graphics as displayed in GuidosToolbox
b) prefix_ids.tif: Image showing the unique identifiers of the objects, buffer zones and watersheds, etc. using negative values for specific data and positive values for the individual objects, i.e. the following notation:
-6: Buffer zone
-5: Buffer zone boundary
-4: Missing data
-3: Watershed
-2: Omitted Foreground pixel
-1: Hole in Foreground
0: Background
1-x: Unique object identifier

➢ **Proximity:** This option will measure the vicinity between neighboring objects. As with the Influence Zones, the objects of interest are defined with 8-connectivity and a minimum area, which can be set via the MSPA parameter 2 *EdgeWidth*. Objects smaller than the minimum size are displayed in light blue and the watershed line in black. When browsing with the mouse pointer through the image, the information panel below the viewport will list the component ID and the associated area in pixels, if we are in the background, or over an omitted foreground object.

Placing the mouse pointer over the watershed line will list the x/y-coordinates of the watershed location and the following three measures (**Figure 22** below illustrates a sample situation):

1. Proximity: The proximity value is equivalent to the shortest distance, measured in pixels, needed to connect neighboring objects. Along the watershed line, this will be two objects. At watershed intersection points, the proximity value corresponds to the shortest distance needed to connect all neighboring objects.

2. CAG: The **C**onnected **A**rea **G**ain (CAG) is a measure of how much area is gained when establishing a connection between disjoint components at a given location. In general, and when combining two objects with area *a* and *b* the combined *reachable* area becomes $\sqrt{(a+b)^2} = \sqrt{a^2 + 2ab + b^2}$. Here, the area gained through the establishment of the new connection is $\sqrt{2ab}$ when connecting 2 components, $\sqrt{2ab + 2ac + 2bc}$ when connecting 3 components, etc. For example, in **Figure 22** below the proximity between the two components with ID 4 and ID 3 is 18 pixels. The connected area gain at this location would be:
$CAG = \sqrt{2ab} = \sqrt{2 * 5598 * 16314} = 13515.$

3. CAG_rel [%]: The relative CAG [%] is defined as the ratio of CAG to the total area of all components. Larger percentage values imply that a larger amount of area will be connected and hence indicate a higher efficiency of a connection established at this given location.

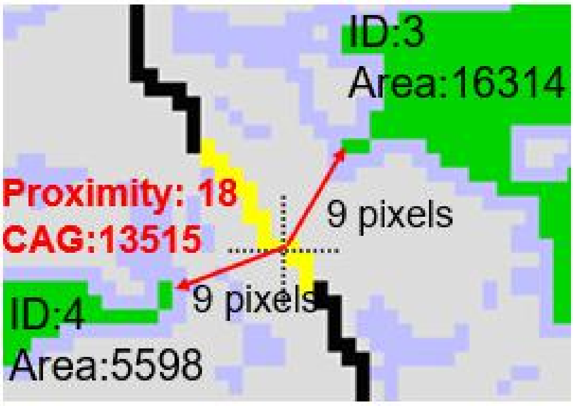


Figure 22: Example showing *Proximity* and *CAG* (connected area gain) between two components.

The left dropdown menu in the panel below the viewport, *CoreZone,* can be used to neglect branches by defining core zones of the objects. The dropdown menu on the right, *Prox_max,* can be used to define a maximum proximity value of interest. Proximity values equal to or smaller than *Prox_max* will be highlighted in yellow on the watershed line. The minimum proximity value is shown in red color and the total number of pixels having this minimum value is listed in the title bar description above the viewport. A mouse-click on the *Divide* button will reset the *CoreZone* and *Prox_max* fields to their default values of zero. The Proximity module can be used to examine the vicinity of neighboring objects while the CAG measures the (relative) importance for any proximity value. In combination, these tools may be useful to locate and evaluate restoration pathways of dissected landscape patches.

Proximity statistics for the entire image may be obtained by clicking on MSPA-parameter 4 **Intext**, providing the following three actions (**Figure 23**):

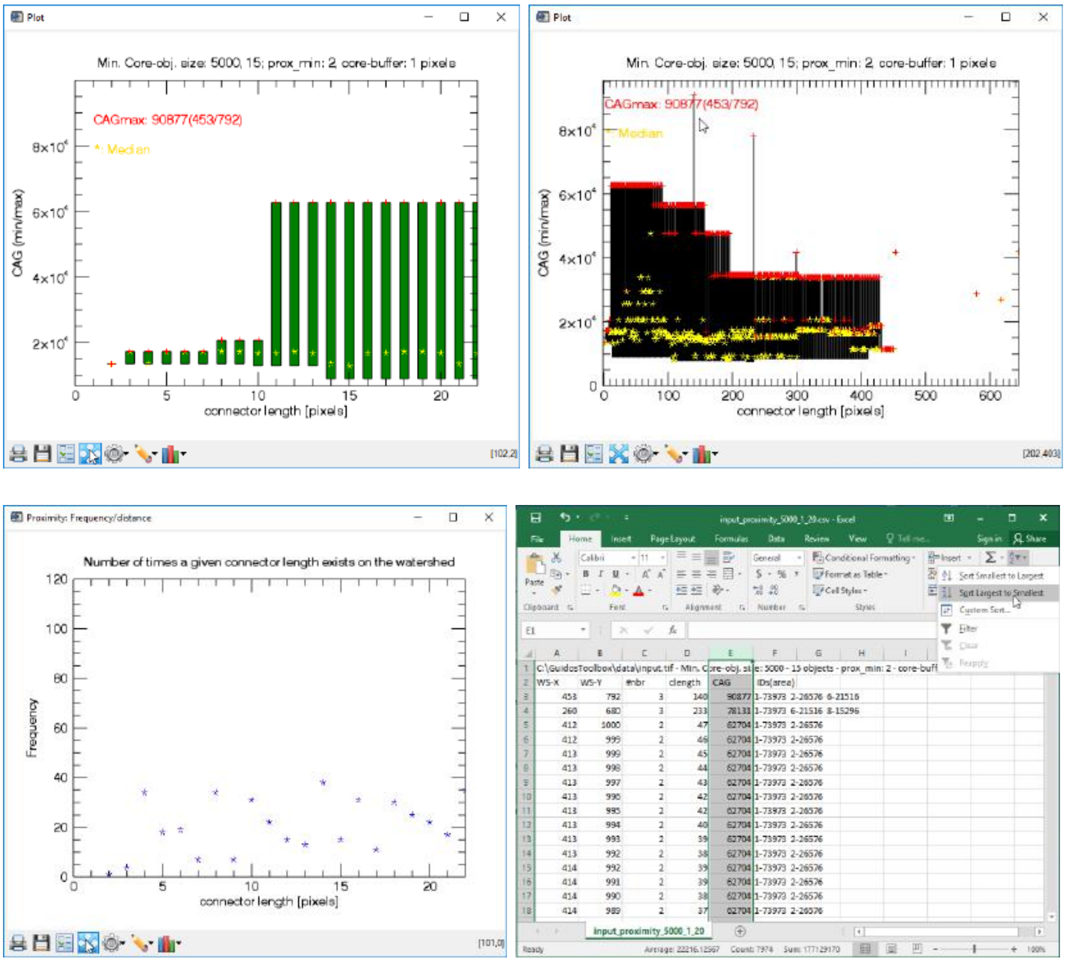


Figure 23: Example showing *Proximity* and *CAG* statistic summaries for the entire image. Top: CAG range for each Proximity (connector length) value (to expand to the full proximity range click the blue cross in the bar plot – top right). Bottom left: Proximity frequency along the watershed. Bottom right: Spreadsheet summary.

a) **CAG histogram:** A new graphic window plot (Figure 23 top left) showing the range of CAG values (minimum, median, maximum) encountered for each proximity value encountered along the watershed. The default proximity range is [0, prox_max]. Click the blue cross icon in the bar-plot image to display the full proximity range (Figure 23 top right). In addition, the maximum CAG found in the image and its x/y-location is printed out in red color in the bar plot image.

b) **Frequency histogram:** A new graphic window showing the number of times a given proximity value was found along the watershed. The default proximity range is [0, prox_max]. Click the blue cross icon in the bar-plot image to display the full proximity range.

c) **Spreadsheet summary:** A summary spreadsheet is written to disk providing comma separated information on location; proximity; CAG; number, ID, and area of objects for each watershed location. This csv-file can be imported into a spreadsheet application and then sorted to the property of interest; for example, to find the x/y-location providing the highest CAG or to list all CAG values and spatial locations for a given connector length (proximity).

Saving the proximity analysis will produce the following five files with the prefix <name>_proximity_<Y>_<b1>_<b2>_<zoom>, where Y stands for the selected minimum object size in pixels, b1 and b2 stand for the selected values in the *CoreZone* and *Prox_max* fields, respectively, and the suffix *zoom* will be added if a subregion of the image is saved:
a) prefix_CAG.tif: Image showing CAG values along the watershed.
b) prefix_viewport.tif: Same graphics as displayed in GuidosToolbox
c) prefix.csv: Proximity/CAG statistic summary spreadsheet file (csv).
d) prefix_ws.tif: Image showing the proximity values on the watershed.
e) prefix_ids.tif: Image having positive values for the unique identifiers of the objects and negative values for specific data, i.e. the following notation:
- 4: Missing data
- 3: Watershed
- 2: Omitted Foreground pixel
- 1: Hole in Foreground
0: Background
1-x: Unique object identifier

1.3.5 Cost

The modules in this section are designed to conduct a Cost analysis, requiring the following two input images:

a) **Resistance map**: A single-band image of datatype Byte providing a relative resistance value within [1, 100] byte for each pixel. Resistance values of 0 are not allowed. If the resistance map contains values larger than 100 byte then the user should setup an appropriate re-scaled resistance map within [1, 100] byte, for example using the **Recode** option. A resistance map could be a land-cover map, where for a given species a specific resistance value is assigned to each land-cover class. The land-cover class specific resistance value can then be seen as a measure of the difficulty for that species to traverse a pixel within that land-cover class.

b) **Marker map**: A single-band image of datatype Byte with the same dimensions as the resistance map and having the following values:
0b: Background
1b: Start object A (either a single pixel or the area of an object)
2b: Target object B (optional, only needed for Cost Map AB)
3b: No or missing data, or areas that should be neglected (optional)
The start/target objects A and B must be unique and must not overlap.

These two input maps are combined and the cost map is calculated with a generalized geodesic distance function (Soille 1994). The Cost Marker map can be setup via the dedicated user interface: **Cost Marker Image**.

The Cost submenu provides the following three options:

- **Cost Map A:** This option will calculate and display the cost map starting from object A.

- **Cost Map AB:** This option will calculate a cost map starting from object A and a second one starting from object B, which are then combined into the final cost map AB. Pixels having the minimum value of this map are defined as *Least Cost Range* pixels and the skeleton of this range is called the *Least Cost Path*. The latter is a subset of the Least Cost Range and shows one trajectory of minimum cost between the start and target objects A and B.

The title bar will inform about the minimum and maximum cost values encountered in the image. The cost inside a start/target object is set to zero. The division panel below the viewport can be used to group the cost surface into the ranges small/medium/large using either pre-defined or custom values in the respective threshold drop-down menu. This option may be of interest to highlight cost corridors between A and B with a slightly increased minimum cost (i.e. 1.1 x least cost) in order to visualize and/or investigate potential alternative connecting pathways. While positive values show the actual cost the following values are reserved for specific assignments:
- 3: Pixels that cannot be reached from or are between the start/target object
- 2: Missing data
- 1: Start/target object A/B
0: Least Cost Path

Depending on the selected file format saving a cost map will provide:
- **GeoTiff**: <input filename>_costmap<A or AB>-data/viewport.tif, a twin set of Geotiff-images of data type:
  a) Long integer with the actual cost map
  b) Byte matching the visual display of the GuidosToolbox viewport. When linked against the corresponding data image in a GIS this image may be useful to visualize the (user-selected) cost ranges.
- **Generic-Tiff**: <input filename>_costmap<A or AB>.tif, two images:
  a) Long integer Tiff-image with the actual cost map
  b) Byte PNG-image matching the visual display of the viewport.
- **Generic-PNG**: <input filename>_costmap<A or AB>.png:
  Long integer with the actual cost values.

The image files are accompanied by <input filename>_costmap<A or AB>.txt, providing a summary of the cost map options, minimum/maximum cost and, if selected, the cost range grouping thresholds.

- **Reconnect:** This option provides an interface to examine potential reconnecting pathways and/or to detect stepping-stones between existing components. This module works in three steps:

1) **Define components of interest:** Components of interest are assumed to have a minimum area (default is 5000 pixels). Similar to the Influence Zones the components of interest can be defined by area (via the MSPA-parameter 2 EdgeWidth) and degree of removing boundary pixels to define a Core component (CoreZone in Division panel below the viewport).
   Resistance: The cost calculation is based on the pixel-based relative resistance values in the image.
   - All **foreground pixels** (components of interest shown in green and the omitted FG-pixels shown in pale blue) have a resistance of 1%.
   - Potential **missing pixels** (shown in white) have an infinite resistance.
   - All **background pixels** (grey) have the same resistance, which can be assigned in the *BGresist* box field below the viewport.

   Please click the ***Divide*** checkbox to apply any new Reconnect settings.

2) **Define start/target component:** Clicking on MSPA-parameter 1 **FGConn** will open a new window with information how to setup the marker image for the least cost path (LCP) calculation. Follow the instructions provided below the viewport, select the start/target component of interest with a mouse pointer click (zoom in if needed) and then press the Enter-key or Space bar for selection. Additional information windows are shown to guide the user in this process.

3) **Calculate Reconnection:** after a successful start/target definition, clicking on MSPA-parameter 1 **FGConn** will calculate the least cost path (LCP), which is then displayed in purple color in the image. The title bar provides additional LCP-information: the number of background pixels required to establish this path (*RestorePixels*); *CAG* (Connected Area Gain, absolute and relative, see **CAG**); and the component IDs encountered along this LCP.

The Reconnect module could be of interest in planning scenarios to investigate the impact of background resistance changes (i.e., land cover changes) or the introduction of new obstacles on the movement potential within the image area. It may also be useful for the detection of landscape elements functioning as stepping-stones between targets of interest. The number of *RestorePixels* can be seen as a proxy for the expense and the relative CAG as an efficiency proxy to establish a given reconnecting pathway. Finally, the information of intermediate encountered component IDs could be of interest to evaluate which components will become involved in the process of reconnecting.

1.3.6 Network

The MSPA-analysis can be converted into a Network setup for further analysis in a graph-theory application, here Conefor (Saura, 2009a).
A ***Network* is composed of *Nodes*** (↔ MSPA class: **Core**) **and *Links*** (↔ MSPA class: **Bridge** = connectors between different Cores) and **the remaining MSPA classes are neglected**. A connected set of nodes and links is called a ***Component***.

The following Network options are available **after a MSPA-analysis**:

- **NW Components:** Individual components of the network are displayed in alternating colors. The color black is used for node-only components having no links. The **ECA**, *Equivalent Connected (Node/Core) Area*, is the square root of PCnum= $\sum_{i=1}^{N}(node\ area\ of\ component_i)^2$, the overall network connectivity. ECA is measured in units of area, i.e. hectare, or in the generic case in units of pixels. **ECA_rel** is the normalized or relative ECA with respect to ECA_max= $\sum_{i=1}^{N}(node\ area\ of\ component_i)$, when all components are fully connected. When looking at landscape habitats, ECA_rel is also known as *ARH* (amount of reachable habitat) or *PRH* (percentage of reachable habitat). ECA and ECA_rel describe the degree of network connectivity in the image.
  The information window below the main display shows the unique component identifier, the total area of the component, and the contribution of links. Saving the result will produce the following three files:
  a) <name>_nw.tif: Same graphics as displayed in GuidosToolbox
  b) <name>_nw_nwdata.tif: Image with component IDs
  c) <name>_nw.txt (only when saving the entire image): Statistics for each component, total area, and contributed area of links, ECA, ECA_rel.

- **Node/Link Importance:** This option will show the connectivity importance for each node and each link of the network. The connectivity importance is calculated according to equation 4 in Saura (2009b) and having the following three contributions: $dPC = dPCintra + dPCflux + dPCconnector$, where the importance corresponds to the term dPCconnector only. The information window below the main display shows the node/link ID, and its absolute and relative connectivity importance. The top [1, 5, 10]% relative importance of nodes/links are displayed in decreasing intensity of green and red color.
  Saving the result will produce the following four files:
  a) <name>_cs.tif: Same graphics as displayed in GuidosToolbox
  b) <name>_cs_conn.tif: Connectivity importance for each node and link.
  c) <name>_cs_ids.tif: Unique identifier of each node and link where nodes have a negative sign to distinguish them from links.
  d) <name>_cs.txt (only when saving the entire image): Statistics for each component, its nodes and links, area, and connectivity importance.

- **MSPA ConeforInputs:** This option will setup and save the two input files *nodes_mspa_<input>.txt* and *links_mspa_<input>.txt* for further analysis in Conefor (Saura, 2009a). Use this option for detailed graph-theory analysis, which is beyond the network connectivity importance provided within GuidosToolbox.
  Note: When using this option **MSPA ConeforInputs** the **connectivity is defined via** the *MSPA-detected structural connectors* (***MSPA-Bridges***). This is different to the option **ConeforInputs**, where the **connectivity is defined via the *pairwise distance*** of image objects.

1.4. The Help pull-down menu

The Help pull-down menu offers the following options:

- Documentation
- Online
- Bug Report
- About GTB

1.4.1 Documentation

- **GTB Manual:** This option will open the GuidosToolbox Manual in a separate window. The manual provides general information on the organization of the graphical elements within GuidosToolbox and the nature and functionality of the various menus and options.

- **MSPA Guide:** This option will open the MSPA Guide in a separate window. The guide contains important, detailed information on the input data requirements for the processing of MSPA, the MSPA parameters, the resulting MSPA image output, and on the use of the MSPA-standalone version. Please read this document carefully. It contains all MSPA related information, GuidosToolbox is only a graphical interface, designed to facilitate MSPA processing.

- **Changelog:** This option displays recent changes and feature additions for the current version of GuidosToolbox.

- **Disclaimer:** This option will display the GuidosToolbox license conditions. It will also provide related references to be included when this application is used in a publication or commercial product.

1.4.2 Online

- **News:** This option displays current ongoing activities and upcoming changes to be included in a future version or revision of GuidosToolbox.

- **Homepage:** This option will open the GuidosToolbox homepage in a web browser providing current information on the GuidosToolbox software collection.

- **Check for Updates:** Use this option to check for & install GTB upgrades:
    - **Program release**: Includes major changes within the libraries of the programing framework requiring a fresh installation of GTB.
    - **Revision release**: A small patch, which will either fix issues found in the current program release and/or add new features. A revision release can be installed automatically into the existing installation.

- **GWS (GTB Workshop):** Use this option to install or upgrade the GuidosToolbox Workshop material. This material contains presentations with many details on the motivation, design, functioning, and application fields of the different methodologies available in GTB. It is complemented by key reference publications, as well as sample data sets and instructions to illustrate using these tools. This package is used during the 1-2 day GTB training courses but it can also be easily followed by the interested user of GTB. The workshop material can be installed automatically into an existing installation.

- **GTB Product Sheets:** Use this option to open the GTB homepage at the section Product Sheets providing links to pdf-files having further documentation, explanation and application examples of dedicated GTB products.

1.4.3 Bug Report

This option will setup a text file for bug reporting, including date and program version, details on the currently loaded image, operating system summary and instructions to complete the report. If you find a problem, please follow the instructions and email the finalized report to the developer. Please provide precise and concise information allowing duplicating your issue, which will facilitate detecting and fixing the problem in a future update.

1.4.4 About GTB

This option provides information on the currently installed program/revision version of GTB, homepage and contact information, the operating system dependent additional software, and the maximum image dimensions for various processing routines supported in GuidosToolbox. On Linux and MacOS the maximum supported image dimension for MSPA is recalculated dynamically accounting for the amount of available free RAM in the operating system.

2. The Image/Display Attributes window (top left panel)

This panel allows changing the image and display attributes of the image shown in the viewport in the right panel.

The left side provides:
- Flip Vertical: Select this option to vertically flip the image.
Note: This option is not applicable for geotiff images.

- Normalized: Display the image values using either their apparent values (default) or normalizing them into the range [0, 100].

- Autostretch: This switch will scale the present image values into [min, max]. This feature can be used to visualize images with small contrast range when using color tables spanning the entire range of [0, 255] byte.

The right side provides:
- Select Colortable: A series of predefined color tables and the option to setup a user-defined color table adjustable via a dedicated window interface.

- Zoom Mode/Factor: These settings are used to specify a rectangular *Region Of Interest* (ROI), a sub region of the image: Prior to the definition of the ROI, a zoom factor in the range of [1, 10] should be selected from the *Factor* drop-down menu. Next, a rectangular ROI is defined by holding down the left mouse button and dragging the mouse inside the graphic display. The selected region is outlined in green color and constantly updated until the mouse button is released. The selected zoom factor is then applied to the selected area and displayed in the viewport. The *Zoom Mode* button changes to *Quit Zoom*, providing the option to return to the display showing the entire image extent.

- Image Info: A separate window will display details of the currently loaded image, such as data type, number of bands, unique pixel values, and geoheader information including projection name and EPSG-code, if a GeoTiff image was loaded.

3. The MSPA window (bottom left panel)

This window is divided into the following two segments:

3.1. MSPA Parameters

This window allows changing the settings of the four MSPA parameters (more details can be found in **Help → MSPA Guide**):

1. **FGconn:** The default setting for the connectivity of the foreground pixels is 8-connectivity (cardinal and diagonal directions) but may also be constrained to 4-connectivity (cardinal directions only).

2. **EdgeWidth:** The MSPA analysis scale driving the distance of the non-Core boundary classes (default: 1); the selected value is equivalent to the resulting boundary width in pixels. The x-entry in the drop-down menu can be used to insert a custom value within [0, 100]. A new custom value will only be assigned after the Enter key has been pressed.

3. **Transition:** Transition pixels are those pixels of an Edge or a Perforation where the Core area intersects with a Loop or a Bridge. The default value (1: tick mark set) is to show transition pixels as Loop or Bridge pixels connecting to the Core area. However, doing so will interrupt the visual integrity of a closed Edge or Perforation perimeter. The closed perimeter display can be maintained by switching transition to off (0: tick mark unset). Please note that when transition is off, short Bridges of 2 pixels will not be visible since they are hidden under the Edge/Perforation pixels.
   Note: Changing the Transition setting will change the visual appearance only but not the actual pixel values or statistics of the MSPA image.

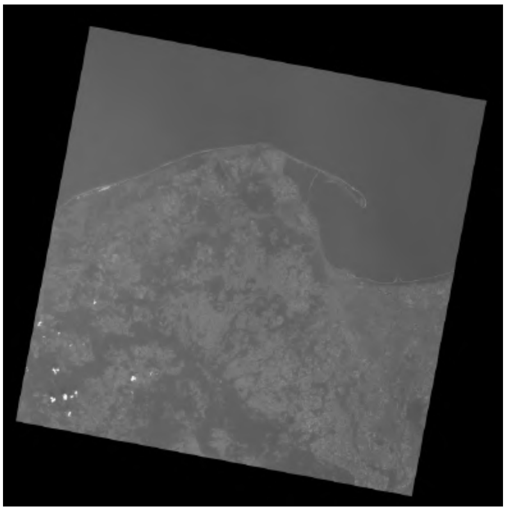
4. **Intext:** This parameter allows distinguishing internal from external background, where internal background is defined as being surrounded by blue Perforation pixels. The default is to enable this distinction, which will add a second layer of classes to the seven basic classes. All classes, with the exception of Perforation, which by default is always internal (105 byte), can then appear in internal or external background.
   Note: Setting Intext=1 is needed to retrieve information on ***Openings*** in the Foreground. Please read the **Help → MSPA Guide** for further details.

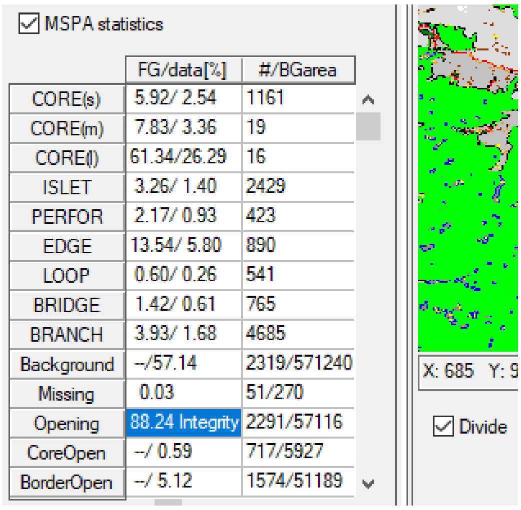
Note: The 4-parameter fields are also used for non-MSPA purposes (i.e.: SPA3/5/6, Cost and Distance analysis), please consult the respective section for further information.

3.2. MSPA Statistics

The MSPA analysis requires a (pseudo-) binary input image. In general, the rectangular image area is composed of the following two parts:
1) Data area (required): Pixels with information and assigned to be either background or foreground.
2) Missing area (optional): Pixels without information.

In the left image below, the data-area corresponds to pixels where information is available (the rectangular area composed of grey-scale pixels). The no-data area, where pixel information is not available, are the cloud pixels (white pixels at the bottom left of the data area) and the boundary segments (black) between the image boundaries and the grey data area. Please note that a particular land cover type could be defined as 'Missing' in order to exclude it from the MSPA analysis.





MSPA statistics

	FG/data[%]	#/BGarea
CORE(s)	5.92/ 2.54	1161
CORE(m)	7.83/ 3.36	19
CORE(l)	61.34/26.29	16
ISLET	3.26/ 1.40	2429
PERFOR	2.17/ 0.93	423
EDGE	13.54/ 5.80	890
LOOP	0.60/ 0.26	541
BRIDGE	1.42/ 0.61	765
BRANCH	3.93/ 1.68	4685
Background	--/57.14	2319/571240
Missing	0.03	51/270
Opening	88.24 Integrity	2291/57116
CoreOpen	--/ 0.59	717/5927
BorderOpen	--/ 5.12	1574/51189

Data (grey)/no-data(black/white) MSPA statistics window

Figure 24: Left: The rectangular image area is composed of data and no-data area. Right: Example of entries in the MSPA statistics window.

With this definition, we have:
Image = Data + Missing = (Foreground + Background) + Missing
The MSPA statistics window shows simple statistics of the seven basic MSPA Foreground classes, Missing, Background and Openings:
Left column: Shows the percentage = number of class pixels per foreground area and per data area.
Right column: Shows its frequency = number of unique objects of the given class and, where applicable, the area = number of background pixels covered.

In the example above and for the class ISLET we find:
- 3.26% of the *Foreground* area are Islet pixels
- 1.40% of the *Data* area (= Foreground + Background) are Islet pixels
- 2429: There are 2429 Islets (regardless of their individual size) in this image.

Similarly, for the class PERFORATION the statistics show that the image has a total of 423 perforations, and all perforation pixels together make up 2.17%

of the Foreground area, or 0.93% of the Data area (foreground area + background area). The total Background area enclosed by Perforation pixels (Core-Openings) amounts to 5927 pixels. All MSPA statistics are calculated for closed foreground boundaries.

Openings: A given foreground cover may contain openings, which can be divided into *Core-openings* (dark-gray, 100 byte; completely inside Core and surrounded by blue Perforation pixels) and *Border-openings* (medium-gray, 220 byte; next to foreground boundaries and *not* completely inside Core). Statistics for both type of openings are displayed in the MSPA statistics window and opening pixels are assigned with the values 100/220 byte, **if Internal = 1**. If Internal = 0 no distinction is made between regular background and Openings. Finally, the *integral Foregound* (iFG) is defined as the sum of foreground + all openings.
Forest Integrity is a measure accounting for the area of all openings within the integral forest area (= forest + openings). Forest Integrity is 100% if the forest has no openings at all. In the example image above, 11.76% of the integral forest area are openings, which implies that the forest integrity is reduced to 88.24%.
There are 2291 openings in the Foreground area with a total area of 57116 pixels. 717 of the 2291 openings are *Core-Openings* within the Core area of the forest covering a total area of 5927 pixels (area inside of Perforations). The remaining 1574 openings are *Border-Openings* (see bottom panel in MSPA statistics in **Figure 24**) located along the outside forest boundary (Edge) and covering a total area of 51189 pixels.

If the *Divide* checkbox below the viewport is ticked, the statistics for the MSPA class Core will be provided individually for small/medium/large Core areas.
Please note that the purpose of these basic statistics is to provide a quick summary only. For example, the statistics do not account for the distinction of internal/external Core classes. The statistics will be reset in case of changing a MSPA parameter affecting the statistics. More details on the MSPA classes is available in **Help → MSPA Guide**.

Note: MSPA-plugins with documentation and installers are available for **ArcGIS**, **R**, and **QGIS**. They can be downloaded from the MSPA-website.
MSPA is also available as a standalone executable for inclusion into custom scripts. More details on the use and settings of the standalone version can be found on the last page of the **Help → MSPA Guide**.

4. The Viewport window (top right panel)

This window displays the original, processed or the zoomed area of the image. The image is either displayed in its original size or automatically downsized to fit the viewport of GuidosToolbox. Any processing is performed on the original non-zoomed image. The icon of the mouse pointer inside the viewport can be changed via the General Tools → Switch Cursor option.

5. The Pixel Locator/Value panel

This panel below the viewport shows the pixel coordinates, value and type for the current location of the mouse pointer in the viewport. This feature works in full display as well as in Zoom Mode. It will also list related specific class names when investigating certain image types in the viewport, i.e. MSPA, FAD or MCD.

6. The Divide Range panel

This panel below the Pixel Locator/Value panel allows dividing the data range of the following image types into the 3 groups: small/medium/large. The two thresholds defining these groups are defined via two drop-down menus. The user can either select predefined settings or enter custom thresholds to define threshold values for the 3 groups. A new custom value will only be assigned after the Enter key has been pressed.

- MSPA: Divide the MSPA core area [pixels],
- Fragmentation: Divide the Fragmentation range [%],
- Contortion: Divide the Contortion range,
- Cost: Divide the Cost range,
- Distance: Divide the distance range [pixels] for foreground and background.
- Influence Zones: Define the width of buffer zones into the Foreground (to define Core-objects) and into the Background (buffer zones ranging outside of Foreground or Core objects),
- Proximity: Define CoreZones and maximum proximity values of interest.
- Reconnect: Define CoreZones and relative resistance for the background pixels.

7. Limitations and known issues

The following list summarizes known limitations and provides suggestions for potential issues arising when using GuidosToolbox:

- **MS-Windows 32-bit:** Certain features in GTB require large array processing, which may exceed the available memory allocation and hence lead to wrong results or even a crash of GTB. Support for MS-Windows 32-bit operating systems will be discontinued in the near future.
- **Maximum image dimensions:** The supported maximum dimensions in x and y are listed under Help → **About GTB**. On Linux and MacOS, the maximum size for MSPA image processing is recalculated dynamically accounting for the currently available amount of RAM in the system. To increase the potential maximum image dimensions, the user should exit from any other running applications occupying system memory.
- **Cost Analysis:** This type of analysis is implemented using the data type long integer. For very large images, and depending on the average resistance values, the maximum of this data type may be superseded. In this case, it will not be feasible to conduct a cost analysis.
- **GTB window size:** The size of the program window is driven by the currently loaded image dimensions and, at present, cannot be maximized to fit the entire screen.
- **Save Image → kml does not work:** On certain 64-bit versions of MS-Windows the option to save an image in kml-format may not work due to a conflict with other GIS-software installed in the system. A patch to address this issue is available in the folder *C:\GuidosToolbox\guidos_progs*. If needed, the user should double-click the file *fix_saveaskml.bat* and provide the administrator password to apply the patch.
- **GDAL:** The version included in GuidosToolbox is 1.11.3. Newer versions with additional features can be obtained from the GdalBinaries website.
- **Data folder on network drive:** Some users have reported issues when processing images stored on network drives. Image data should preferably be stored in the default *GuidosToolbox\data* folder or on a local hard drive.
- **Concurrent use of external software:** Simultaneous opening/processing of the same raster file in GuidosToolbox and an external software (i.e., Erdas, ENVI, ArcMap, QGIS) should be avoided.
- **Concurrent use GTB:** Running multiple instances of GTB from the same installation directory may cause overwriting temporary files within the program directory and thus resulting in program crashes or erroneous results. Multiple instances of GTB on the same machine can be run but must be executed from different installation directories.
- **Batch processing under MS-Windows:** A MS-Windows system-inherent problem will limit the number of files for batch processing (only 32000 bytes of string can be read). This problem can be avoided by shortening the length for the full path to the image files or bypassed completely by using the Linux or the MacOS version of GuidosToolbox instead.
- **Undo/Redo:** Since only one processing step is saved in the activity history, the user is advised to save intermediate results to the hard drive.

8. Acknowledgments

The author would like to thank all those for the help and suggestions during the design phase of GuidosToolbox. Parts of this software package are based on conceptual ideas, which can be found in the excellent IDL programs of David Fanning. I am deeply grateful to Pierre Soille for sharing in-depth knowledge into the beautiful world of mathematical morphology and especially for all his time and effort in writing the MSPA code. Sincere thanks go to Kurt Riitters for his invaluable advice on the pattern algorithms; Santiago Saura for providing details and help when implementing the network analysis routines; Frank Warmerdam for permitting to include the excellent FWTools package in the Linux version of GuidosToolbox; Bernd Eckhardt for advice and constructive suggestions improving the user-experience. Finally, a big thank you to Dario Simonetti for his guidance and help with getting GDAL to work on MS-Windows.

This software package includes research products designed and developed within the context of the Collaborative Research Arrangement between the United States Department of Agriculture, United States Forest Service (Forest Service Agreement No. 14-MU-11330110-001) and the Joint Research Centre of the European Commission (Collaborative Research Arrangement No. JRC 33385).

9. References

Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions Pattern Analysis Machine Intelligence (PAMI)* 8(6):679-698.

Kozak, J., Ziolkowska, E., Vogt, P., Dobosz, M., Kaim, D., Kolecka, N., Ostafin, K. (2018). Forest-Cover Increase Does Not Trigger Forest-Fragmentation Decrease: Case Study from the Polish Carpathians. Sustainability, 10(5), 1472. DOI: 10.3390/su10051472

Lee (1986). *Optical Engineering* 25(5), 636-646

Li, H., and Reynolds, J.F. (1993). A new contagion index to quantify patterns of landscapes. *Landscape Ecology* 8:155-162.

Musick, H. B., and H. D. Grover. (1991). Image textural measures as indices of landscape pattern. Pages 77-103 in M. G. Turner and R. H. Gardner, editors. 1991. *Quantitative Methods in Landscape Ecology*. Springer-Verlag.

Riitters, K. H., J. D. Wickham, and K. B. Jones. (1996a). A Landscape Atlas of the Chesapeake Bay. Tennessee Valley Authority, Norris, TN. 29 pp.

Riitters, K. H., O'Neill, R.V., Wickham, J.D., and K. B. Jones. (1996b). A note on contagion indices for landscape analysis. *Landscape Ecology* 11 no. 4, pp. 197-202.

Riitters, K. H., R. V. O'Neill, and K. B. Jones. (1997). Assessing habitat suitability at multiple scales: a landscape-level approach. *Biological Conservation* 81:191-202.

Riitters, K. H., Wickham, J.D., Vogelmann, J.E., and K. B. Jones. (2000). National land-cover pattern data. *Ecology* 81: 604.

Riitters, K.H.; Wickham, J.D.; O'Neill, R.V. [and others]. (2002). Fragmentation of continental United States forests. *Ecosystems* 5:815–822.

Riitters, K.H. (2005) Downscaling indicators of forest habitat structure from national assessments. Ecological Indicators 5:273-279.
DOI 10.1016/j.ecolind.2005.04.003

Riitters, K.H.; Wickham, J.D.; Wade, T.G. (2009). An indicator of forest dynamics using a shifting landscape mosaic. *Ecological Indicators* 9:107-117.
DOI 10.1016/j.ecolind.2008.02.003

Riitters, K.H.; Coulston, J.W.; Wickham, J.D. (2012a). Fragmentation of forest communities in the eastern United States. *Forest Ecology and Management* 263:85-93. DOI 10.1016/j.foreco.2011.09.022

Riitters, K.H.; Wickham, J.D. (2012b). Decline of forest interior conditions in the conterminous United States. Scientific Reports 2, Article number: 653.
DOI 10.1038/srep00653

Riitters, K.; Wickham, J.; Costanza, J.K.; Vogt, P. (2015). A global evaluation of forest interior area dynamics using tree cover data from 2000 to 2012 Landscape Ecology. DOI 10.1007/s10980-015-0270-9

Roberts, L. G. (1965). Machine perception of three-dimensional solids, in *Optical and Electro-Optical Information Processing*

Saura, S. and Torné, J. (2009a). Conefor Sensinode 2.2: a software package for quantifying the importance of habitat patches for landscape connectivity. *Environmental Modelling & Software* 24: 135-139.
DOI: 10.1016/j.envsoft.2008.05.005

Saura, S. and Rubio, L. (2009b). A common currency for the different ways in which patches and links can contribute to habitat availability and connectivity in the landscape. *Ecography*. DOI 10.1111/j.1600-0587.2009.05760.x

Seebach, L., Strobl, P., Vogt, P., Mehl, W., San-Miguel-Ayanz, J. (2013) Enhancing post-classification change detection through morphological post-processing – a sensitivity analysis. *International Journal of Remote Sensing*, 34 (20), 7145–7162. DOI: 10.1080/01431161.2013.815382

Shannon, C. E. (1948) A mathematical theory of communication. *The Bell System Technical Journal*, 27, 379–423 and 623–656

Sobel, I. (1970). Camera Models and Machine Perception, AIM-21. Stanford Artificial Intelligence Lab, Palo Alto.

Soille, P. (1994). Generalized geodesy via geodesic time. *Pattern Recognition Letters* 15(12), pp. 1235-1240. DOI: 10.1016/0167-8655(94)90113-9

Soille, P. (2004). Morphological Image Analysis: Principles and Applications. Second Edition, Springer-Verlag, Berlin. DOI: 10.1007/978-3-662-05088-0

Soille, P. and Vogt, P. (2009). Morphological segmentation of binary patterns. *Pattern Recognition Letters*. DOI: 10.1016/j.patrec.2008.10.015

Vogt, P., 2015: Quantifying landscape fragmentation. In: SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO, 17. (SBSR), 2015, João Pessoa. Anais... São José dos Campos: INPE, 2015. p. 1239-1246. Internet. ISBN 978-85-17-0076-8. Available from: <http://urlib.net/8JMKD3MGP6W34M/3JM47SP>

Vogt P., Riitters, K., 2017. GuidosToolbox: universal digital image object analysis. European Journal of Remote Sensing 50:1, 352-361, DOI: 10.1080/22797254.2017.1330650.

Wade, T.G.; Riitters, K.H.; Wickham, J.D.; Jones, K.B. (2003). Distribution and causes of global forest fragmentation. *Ecology and Society* 7(2):7, [online] URL: http://www.ecologyandsociety.org/vol7/iss2/art7

Wickham, J.D., and Riitters, K.H. (1995). Sensitivity of landscape metrics to pixel size. *International Journal of Remote Sensing* 16:3585-3594.

Wickham, J. D. and D. J. Norton. (1994). Mapping and analyzing landscape patterns. *Landscape Ecology* 9:7-23. DOI: 10.1007/BF00135075