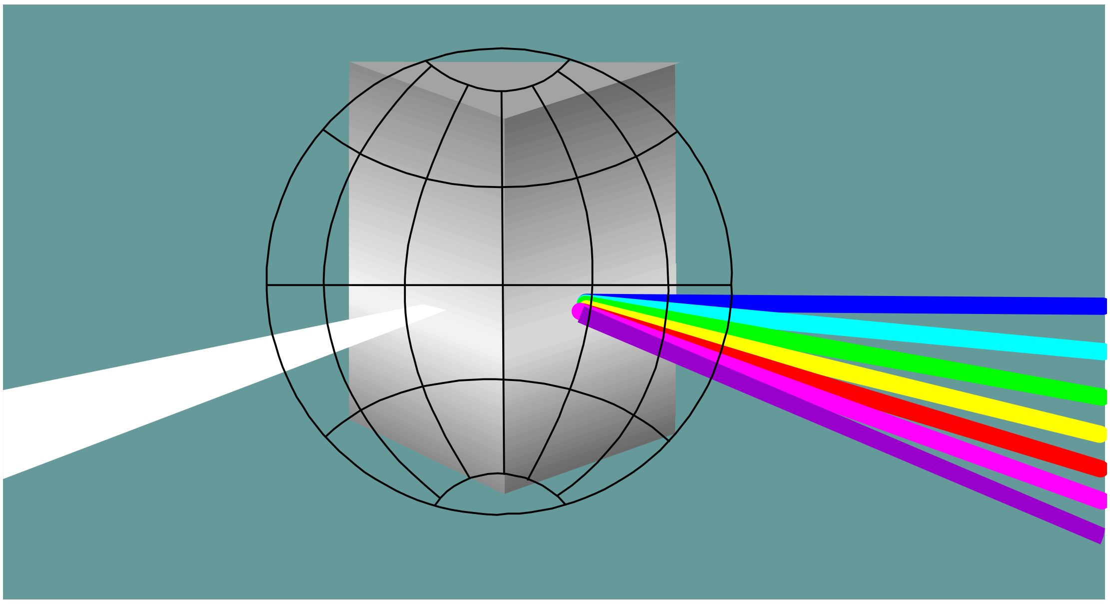
MarineTools

A QGIS plugin for marine data processing



A collection of software for the processing, analysis and enhancement of marine acoustic data for seafloor characterisation and mapping.

Version 1.93

Compiled by

Tim Le Bas

National Oceanography Centre

U.K.

Contents

Introduction

Geographic Information Systems provide many basic tools and techniques for manipulation of data. It is therefore for the user to combine these tools to achieve the desired result. Finding the correct tool can sometimes be challenging; in addition to then finding the appropriate inputs for the tool. Many of the tools are tailored for terrestrial data inputs and can be inappropriate for marine data or may require alteration. Creation of a dedicated set of tools for marine data was suggested. The "MarineTools" toolbar has been created for two mainstream GIS packages: ESRI's ArcPro v3.2 and QGIS v3.8 and above. Both toolbars are a collation of standard tools and published methods that are typically used with marine data. The inventory is not exhaustive and is actively being added to, updated, and corrected. However, the ethos is to make these standard tools available to all users with a simple graphical user interface (GUI) and produce output which can be viewed, questioned and maybe reproduced with different or more appropriate parameters or data.

The code for the QGIS plugin is written in Python v3. It is open-source code and can be edited or altered by users if desired. This allows for alteration of assumptions, fixing issues, updating of tools, and changing fixed parameters. Many workers create processing models (using model builder) and it is these which can be added to the toolbar, adding new functionality. Similarly, users may extract code for newer processing techniques. The structure of the software is relatively complex and does require a certain level and knowledge of Python software and programming. A QGIS plugin structure explanation is provided on how to make a plugin.

There are three categories for the tools within the QGIS plugin:

1) Bathymetry specific techniques
    - Bathymetry Morphometry – geomorphological derivatives
    - Alter Raster Values – filter out values and using NoData (null) values
    - Spike Filter – removing spurious datapoints
    - Synthetic Sidescan imagery (from bathymetry grid)
2) Interpretation tools
    - Benthic Terrain Modelling (BTM)
    - CoMMa Delineation - feature outlining
    - MBES Segmentation - OBIA on bathymetry and backscatter datasets
    - OBIA (Object Based Image Analysis)
    - Random Forest – prediction of occurrences of features
3) Other tools for data manipulation and survey assistance.
    - Grey Level Co-occurrence Matrices – second order imagery derivatives
    - Estimate new MBES Coverage –estimate over low resolution bathymetry
    - Raster Coverage – polygon of bathymetry coverage
    - Survey Lines – designing survey tracks
    - CSV to Grid – converting csv files to GIS ready files
    - Listing of properties of layers – list of project's datafiles properties

Installation

MarineTools for QGIS is available via the plugins downloads. It is currently an experimental plugin and can be found by searching for MarineTools in the “Not Installed” section of the plugins. Installation is quick. The plugin will then be installed locally in the <user>\AppData\Roaming\QGIS\QGIS3\profiles\default\python\plugins directory.

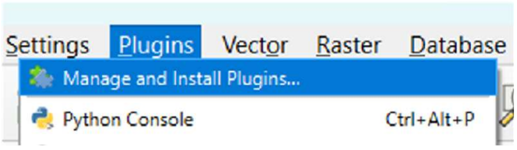
The MarineTools plugin installation has two steps:

1) Installation of Orfeo Toolbox – a QGIS standard processing toolbar
2) Installation of MarineTools

Installation of Orfeo Toolbox

Download the installation software for your machine from:

1. Download the installation software from **https://www.orfeo-toolbox.org/download/**
2. Install the **OTB-9.1.1-Win64** software somewhere accessible
3. Open QGIS
4. In QGIS Plugins – Manage and Install Plugins…
5. Select Not Installed
6. Search for “orfeo”
7. Select and tick Orfeo Toolbox provider
8. Select Settings
9. Tick Show also Experimental Plugins
10. Select Not Installed
11. In QGIS Settings – Options – Processing – Providers – OTB
12. Double click the “OTB application folder” and change it via the  icon to the /lib/otb/applications folder of Orfeo Toolbox Provider (OTB):
    e.g. **C:/Users/<name>/Documents/Software/OTB-9.1.1-Win64/lib/otb/applications**
13. Double click the “OTB folder” and add the main folder via the  icon:
    e.g. **C:/Users/<name>/Documents/Software/OTB-9.1.1-Win64**
14. Click OK to exit Settings



Installation of MarineTools Plugin

1. In QGIS Plugins – Manage and Install Plugins…
2. Select Not Installed
3. In the search bar type “**Marine Tools**”
4. Click and select Marine Tools and click on Install Experimental Plugin

Bathymetry Morphometry

Rationale:

This tool creates several output raster derivatives from a single banded raster bathymetry grid. There are several “Do?” tick options for each of the functions so that the user can specify which outputs are wanted and which options are not required.

Usage:

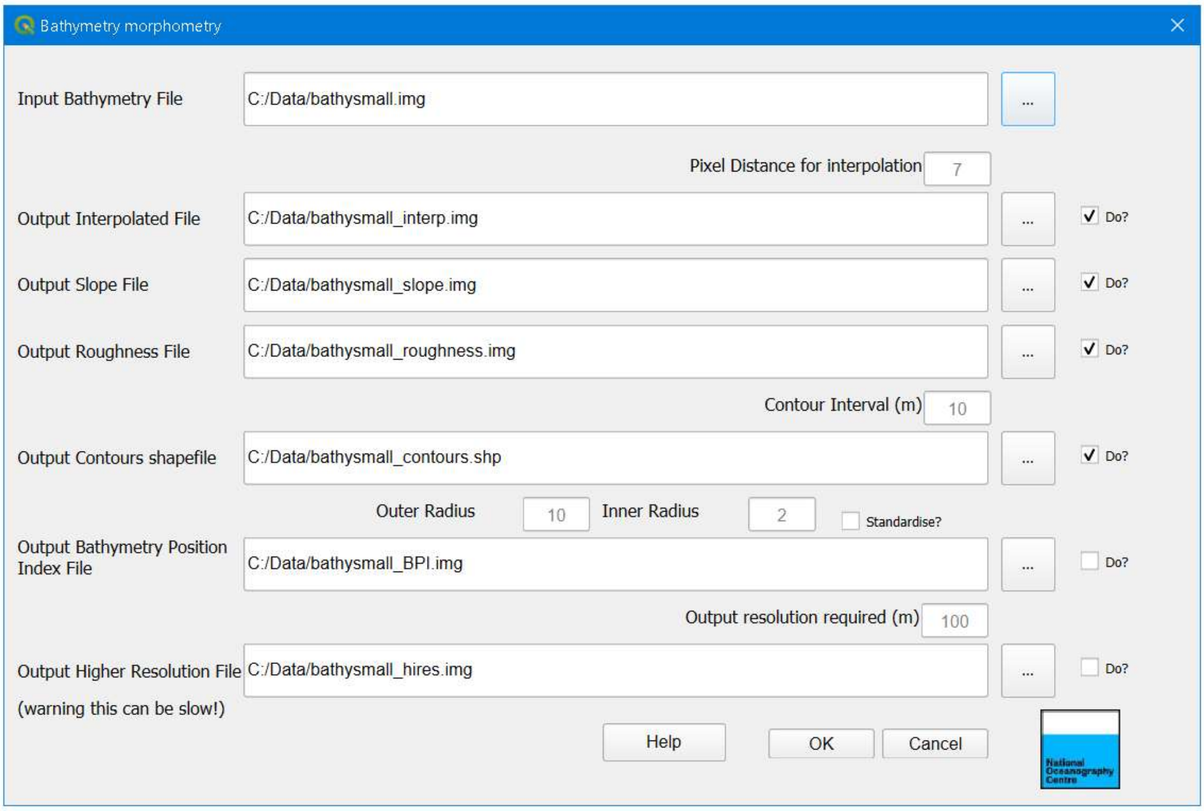
There are six options:

- To create an interpolated bathymetry grid with less or no gaps in its coverage. A smoothing or filling factor in pixels is applied, calculating the average pixel value for the area but only applying this value to pixels that had NoData. It does not alter pixel values of known depths. A default value of 7 pixels kernel size is suggested but can be changed by the user if required.
- To create a slope map from the input bathymetry file, or the interpolated file if created
- To create a roughness map from the input bathymetry file, or the interpolated file if created
- To create a set of vector contours at the specified contour interval. Small contours, of length less than ten times the pixel resolution, are removed as are often small anomalies of less than 6 pixels in areal extent. A default contour interval of 10m is suggested but can be changed by the user if required.
- To create a Bathymetry Position Index (BPI) raster grid from the input bathymetry file, or the interpolated file if created. It calculates the difference between the central value and an average value for an annulus around that pixel and requires two input values, the inner and outer radii. This can be used for broad or fine scale structures delimitation depending on the values used. Results can be standardised making one standard deviation around the mean having a value of -100 or 100.
- To create a higher resolution raster grid from the input file or the interpolated file if created. Higher resolution is created with interpolation between the original grid squares. A default grid resolution of 100m is suggested but can be changed by the user if required. This interpolation is processing intense and can take some time. This option produces some intermediate files in the tempMT directory.

The user will have to change the “Do?” tick boxes to select which of the derivatives are wanted.

The input raster file will suggest output raster filenames (in the same directory) but with an additional descriptive postscript added to the filename. These can be edited by the user if desired.

Example:



Bathymetry morphometry
Input Bathymetry File
C:/Data/bathysmall.img
Pixel Distance for interpolation
7
Output Interpolated File
C:/Data/bathysmall_interp.img
Do?
Output Slope File
C:/Data/bathysmall_slope.img
Do?
Output Roughness File
C:/Data/bathysmall_roughness.img
Do?
Contour Interval (m)
10
Output Contours shapefile
C:/Data/bathysmall_contours.shp
Do?
Outer Radius
10
Inner Radius
2
Standardise?
Output Bathymetry Position Index File
C:/Data/bathysmall_BPI.img
Do?
Output resolution required (m)
100
Output Higher Resolution File
(warning this can be slow!)
C:/Data/bathysmall_hires.img
Do?
Help
OK
Cancel

Alter Raster Values

Rationale:

This tool is to be used to prepare raster imagery for more detailed processing in other tools. Raster imagery for this tool is assumed to be single band grids of data values and have some numeric values in each grid square or a NoData (or null) mark for the grid square.

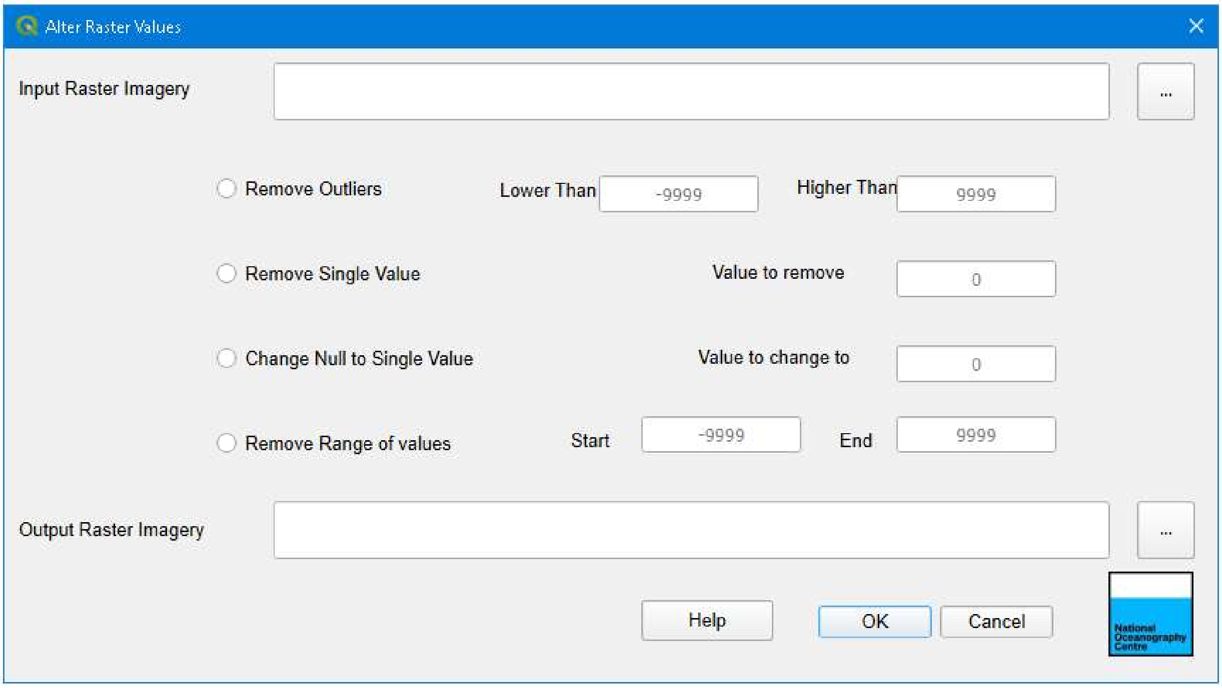
Usage:

There are four options:

- To remove outlier values, which are either too high or too low to be useful or credible. Values found will be replaced with a NoData marker
- To change a specific value to a NoData marker
- To change a NoData marker to a specific value
- To remove a range of values and replace them with a NoData marker.

The user will have to click on one of the radio buttons (circles) to choose one of the options

The input raster file will suggest an output raster filename in the same directory but with an additional "_out" added to the filename. This can be edited by the user if desired.



Spike Removal

Rationale:

This simple tool filters a bathymetry raster grid by removing spurious depth points. The criteria for being a 'spurious depth' value is its value being more than a given threshold value away from the local median value.

Usage:

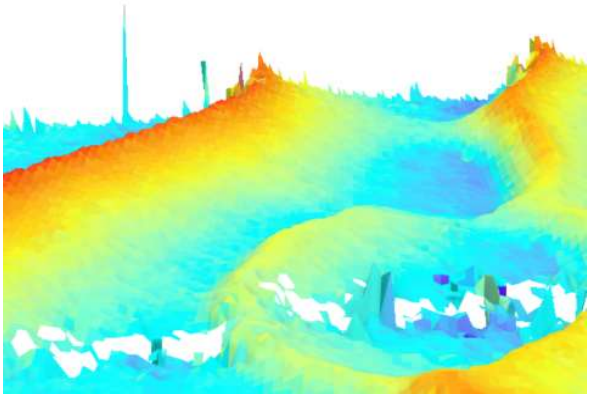
There are two parameters required for this tool:

- The radius of a circle of an area in pixels from which a median value is calculated.
- A threshold value of the maximum deviation from the median surface which is allowed

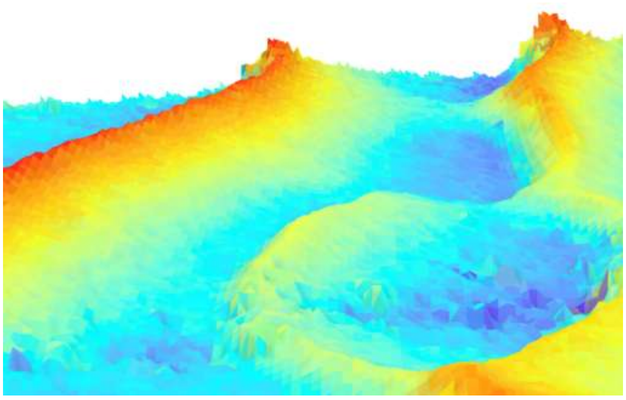
Output are two raster files and have default filename of "_despiked" plus the radius and threshold used for the filter. The first is the despiked bathymetry grid and the second is a slope grid which shows more clearly the effect of the despiking algorithm and the parameters used.

Example:

3D image before filtering



3D image after filtering



Synthetic Imagery

Rationale:

This creates what a sidescan sonar system should see if run over the given bathymetry. It uses a navigation line or points to orientate the theoretical acoustic response from the seafloor. It assumes that all the standard processing techniques for removing systematic characteristics have been applied – such as the beam pattern and TVG (Time Varied Gain). If the navigation point file has altitude values these can be used to show the theoretical returns and can be compared with actual sidescan imagery.

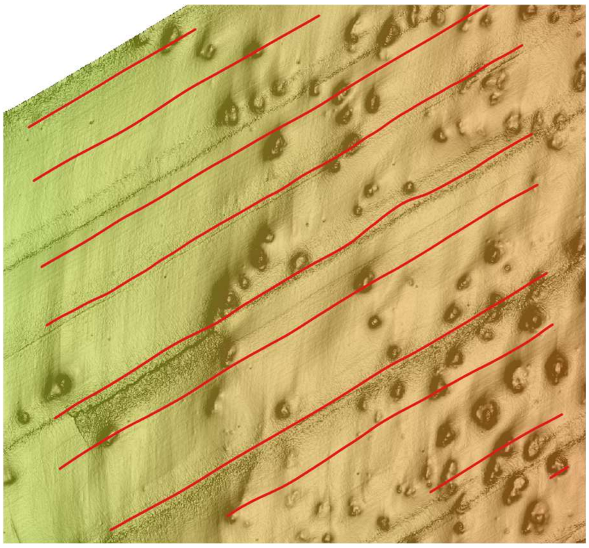
Usage:

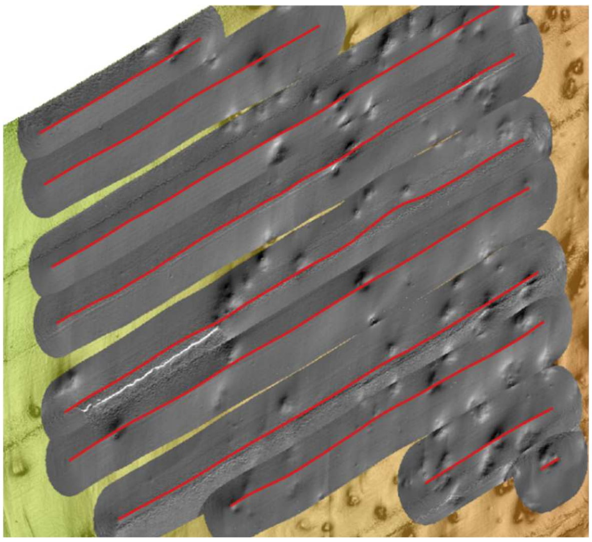
There are four parameters required for this tool:

- A bathymetry grid file. Depths can be positive or negative but must be in metres. This will produce the synthetic sidescan imagery where data exists.
- A trackline for prediction (or comparison). This can be a line or series of points. If it is a series of points the depth of the AUV can be in a field to be selected (as it is likely to be previously collected variable?).
- There are 3 ways to describe the possible ways the synthetic imagery can be created. This describes the effective depth of the sidescan transducers.
    - A variable depth of the vehicle (already known)
    - A predicted constant altitude of the vehicle above the seafloor
    - A constant depth value of the vehicle below the sea surface (hull mounted or underwater vehicle).
- A range parameter in metres. This is the maximum slant range distance that the imagery is seeing.

Output is a raster file and has default filename of "_synthetic_" plus the range.

Example:





High resolution AUV bathymetry and red tracklines - Synthetic Sidescan Imagery

Benthic Terrain Modeler (BTM)

Rationale:

Analyses benthic terrain for the purposes of classifying surficial seafloor characteristics that may be used in studies of benthic habitat, geomorphology, prediction of benthic fish species distribution, marine protected area design, and more

Benthic Terrain Modeler (BTM) is an application originally developed in 2005 at Oregon State University under a cooperative agreement with the NOAA Coastal Services Centre's (CSC) GIS Integration and Development program (Wright et al., 2005). The application provides a set of geoprocessing tools to analyse benthic terrain for the purposes of classifying surficial seafloor characteristics that may be used in studies of benthic habitat, geomorphology, prediction of benthic fish species distribution, marine protected area design, and more. From an input grid of multibeam bathymetry the user may create additional grids of slope, bathymetric position index or BPI (a variation on the topographic position index approach of Guisan et al. 1999 and Weiss, 2001), and seafloor rugosity (after Jenness, 2003 and Iampietro and Kvitek, 2002).

The Benthic Terrain Modeller contains a set of customized scripts that automatically creates grids of broad and fine scale standardized BPIs, and slope from an input bathymetric data set. Using these data and a defined dictionary of characteristics, the tool classifies the bathymetric structures into feature classes. If no dictionary exists yet, the tool will create a dictionary based on standard definitions and can be subsequently edited and the tool re-run.

Usage:

The input raster file will output two files: first an output raster file of the classes file and secondly a vector polygon file of the classes. Default names are suggested, being in the same directory but with an additional "_zones.img" added to the filename for the raster output and "_BTM.shp" for the vector polygon output file. These can be edited by the user if desired.

Four variables are required for the tool:

- The broad scale BPI inner radius – for pixel size of smaller features to be ignored when large scale features are being measured
- The broad scale BPI outer radius – for pixel size of large-scale features
- The fine scale BPI inner radius– for pixel size of very small features (noise) to be ignored when smaller scale features are being measured
- The fine scale BPI outer radius – for pixel size of small-scale features

Default values are given for initial use but can be changed by the user for possible later runs of the tool.

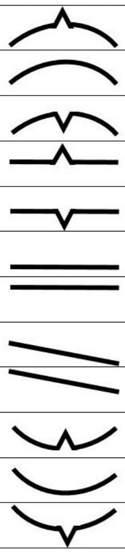
Two tick boxes are provided:

- To delete as many intermediate files as possible (held in a subdirectory called "tempMT").
- To make depth data positive values (depth data is often held as negative values). This BTM tool requires positive values

A default classification dictionary is created which defines the criteria for the different classes. This file is called "Dictionary.csv" and can be edited by the user if desired. An example is shown below where the depth boundary between shallow and deep is 5.8m and slope boundary of 11.8º.

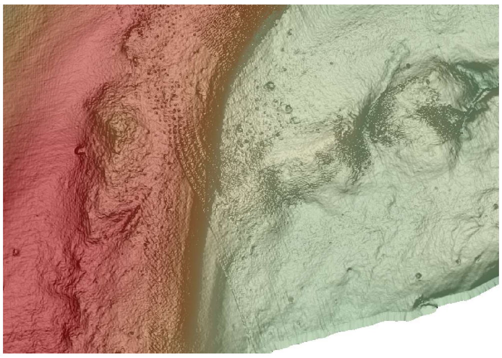
Class	Zone	BroadBPI_	BroadBPI_	FineBPI_Lc	FineBPI_U	Slope_Low	Slope_Upp	Depth_Lov	Depth_Upper
1	Peak	-10000	-100	-10000	-100	0	90	0	12000
2	Ridge	-10000	-100	-100	100	0	90	0	12000
3	Ridge Trough	-10000	-100	100	10000	0	90	0	12000
4	Flat with Ridge	-100	100	-10000	-100	0	90	0	12000
5	Flat with Trough	-100	100	100	10000	0	90	0	12000
6	Flat (Deeper)	-100	100	-100	100	0	11.81052	5.856267	12000
7	Flat (Shallower)	-100	100	-100	100	0	11.81052	0	5.856267
8	Steep (Deeper)	-100	100	-100	100	11.81052	90	5.856267	12000
9	Steep (Shallower)	-100	100	-100	100	11.81052	90	0	5.856267
10	Depression with Ridge	100	10000	-10000	-100	0	90	0	12000
11	Depression	100	10000	-100	100	0	90	0	12000
12	Depression with Trough	100	10000	100	10000	0	90	0	12000

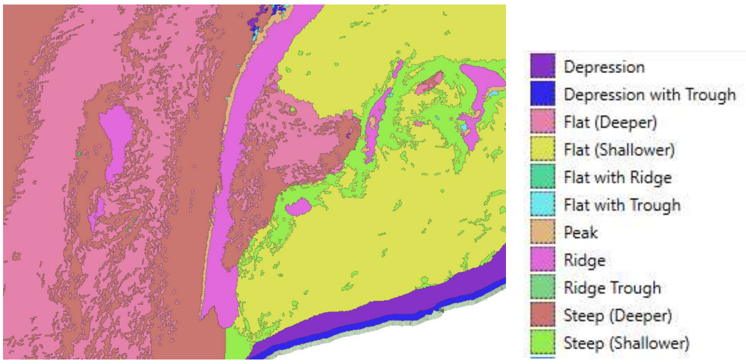
Graphically this equates to:

Zone	Profile
Peak	
Ridge	
Ridge Trough	
Flat with Ridge	
Flat with Trough	
Flat (Deeper)	
Flat (Shallower)	
Steep (Deeper)	
Steep (Shallower)	
Depression with Ridge	
Depression	
Depression with Trough	

The program is complex and requires many intermediate steps. It produces many intermediate files and can take many minutes to complete and create the final output.

Example: Bathymetry and Classified features





CoMMa Feature Delineation

Rationale:

The CoMMa delineation tool can be applied directly to a bathymetric DEM and, when the general seascape is otherwise flat, it might be sufficient to isolate and correctly delineate the targeted features. However, if this is not the case, and the interference of sloping topography or other underlying large-scale landforms can distort the signal of the targets, an LTP derivative is used to isolate a specific wavelength thought to best delineate the feature of interest.

The “Boundary-based Delineation" tool, takes advantage of the hydrological algorithm “Fill”. This algorithm will fill up positive or negative enclosed relief, according to the relief of the targeted features. Then the filled output will be subtracted from the original DEM, creating a flat surface with only the potential features of interest. Then a user-defined threshold, the Vertical Cutoff, is used to define the contour line that Is going to be the base of the delineation.

Once all the areas of confined morphologies are delineated based on the Vertical Cutoff threshold, a sequence of steps is followed and features that do not satisfy the criteria set by the user-defined thresholds will be excluded. The threshold covers aspects related to the minimum vertical relief and dimensions expected for the targeted feature.

The delineation of some features may fail to include the top of positive features or the base of the negative features, by focusing on areas of change of relief – creating holes within the delineating polygons. If that occurs, the user can choose to have it automatically removed.

Usage:

The input file for this tool can either be a bathymetric DEM or an LTP (Local Topographic Position) or BPI (Bathymetric Posotion Index) grid. This input name will automatically provide an output name for the polygons delineating the features. It will append the suffix “_Delin” and the parameters used to define the features, so that the user can identify which parameters were used on specific outputs.

The tool can identify features that have a positive or negative profile.

There are five main parameters required for this analysis, plus several other processing options:

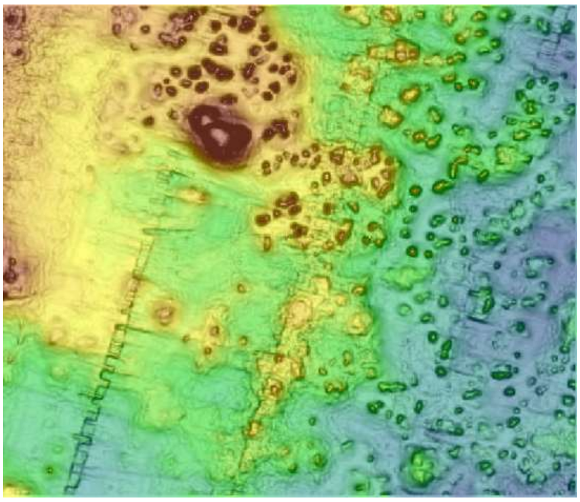
- Vertical Cutoff - The value that will be used as the confining boundary and used to delineate the features. This value will be in meters if the delineation is fully based on the source DEM, OR on the derivative unit, if the latter is used. When the targeted features are well disconnected for other features, this value should be as low was possible, to best capture the geometry of the feature’s boundary. However, the user may want to use a higher value to reduce the likelihood of adjacent features being mapped as a conjoined polygon.
- Minimum Vertical Threshold - Only features with a vertical relief greater than the Minimum Vertical Relief value will be mapped.
- Minimum Width - Minimum Width threshold allows to exclude features based on their size. Only features with width greater the Minimum Width value will be mapped.
- Minimum W/L Ratio - The Minimum Width/Length Ratio threshold allows to exclude features based on their shape. Width and Length are defined by the features’ Minimum

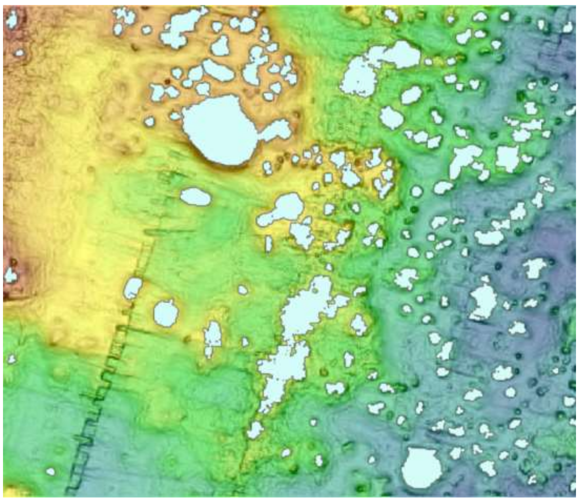
Bounding Geometry (MBG) It should be noted that the Minimum Width/Length Ratio value can range from 1 (for a circle-shaped feature) to almost 0 (for a very elongated feature).

- Buffer to apply to the delineation - The Buffer Distance value is applied to the initial polygons created based on the feature's internal contour line correspond ing to the Vertical Cutoff. The Buffer Distance should reflect approximately the distance, in plan view, from the line delineated to the actual rim/edge of the features.
- Level of Smoothing - The user can decide whether to apply smoothing and simplification to the initial (or buffered) polygons. Light, heavy or no simplification and smoothing options are offered. Simplification is calculated using the Retain weighted effective areas (Zhou-Jones) algorithm, with tolerance (T) based on the size of the polygon (light T = width feature / 10; heavy T = width feature / 5). Smoothing is done using the PAEK algorithm with tolerance (T) based on the polygon's size (light T = width feature / 5; heavy T = width feature / 2).
- Geomorphon raster input – can be left blank or a file added. A geomorphon file is created by the geomorphon tool from the DEM or from an LPT-derived raster. Every cell of the geomorphon raster will have an integer value corresponding to a specific landform type: Flat—cell value 1, Peak—cell value 2, Ridge—cell value 3, Shoulder—cell value 4, Spur—cell value 5, Slope—cell value 6, Hollow—cell value 7, Footslope—cell value 8, Valley—cell value 9, Pit—cell value 10.
- Minimum Geomorphons ratio - if a geomorphon file is defined features with a ratio geomorphons area/total area smaller than this threshold will be excluded from the delineation.
- Delete internal holes - when checked, holes inside a delineated feature are removed. The delineation of some features, especially based on some LTPs layers, may fail to include the top of positive features or the base of the negative features, by focusing on the main areas of change of relief – creating holes within the delineating polygons.
- Delete temporary files - when checked all the files within the temp folder will be deleted. It should be noted that some of these "intermediate" files could be useful to understand the reason behind an unexpected output. If the tool is run multiple times the temp files will be overwrite, to avoid excessive use of disk space. If is required to compare the temp files created with different parameters, then different workspaces should be selected.

Output is a single polygon vector shapefile and its default filename is the same as the bathymetry filename with "_Delin" added to the name. Format output is .shp (plus its associated companion files).

Arosio et al. (2024) CoMMa: a GIS geomorphometry... *Geomorph.*, 458, 109227. 10.1016/j.geomorph.2024.109227





MBES Segmentation

Rationale:

This tool creates a segmented vector polygon output file which characterises and segments the combination of bathymetry and corresponding backscatter mosaic grids. Segmentation is a method to aggregate pixels together to create a thematic map. The segmentation process creates a set of polygons defined by the statistics associated with the input images. Clusters of the imagery pixels are created in 3-dimensional space and created into classes. Aggregation into geographic regions (polygons) is done according to a minimum polygon size rule, and clustering rules.

The segmentation process uses a technique of k-means clustering and uses the iterative removal of outliers via a sieving process.

The bathymetry file is processed to give a slope and roughness derivatives. It is these two derivatives files which are used in conjunction with backscatter imagery which are the 3 inputs to the object-based imagery analysis (OBIA) process.

The resulting shapefile of polygons may take some time depending on the data complexity, file sizes, and number of clusters. Each polygon will have attributes of:

- VALUE – The Class Number of each polygon
- backs_mean – The average of all pixel values in the polygon from the backscatter input file
- backs_stde – The standard deviation of pixel values in the polygon from the backscatter input file
- slope_mean – The average of all pixel values in the polygon from the calculated slope data
- slope_stde – The standard deviation of pixel values in the polygon from calculated slope data
- rough_mean – The average of all pixel values in the polygon from the calculated roughness data
- rough_stde – The standard deviation of pixel values in the polygon from calculated roughness data

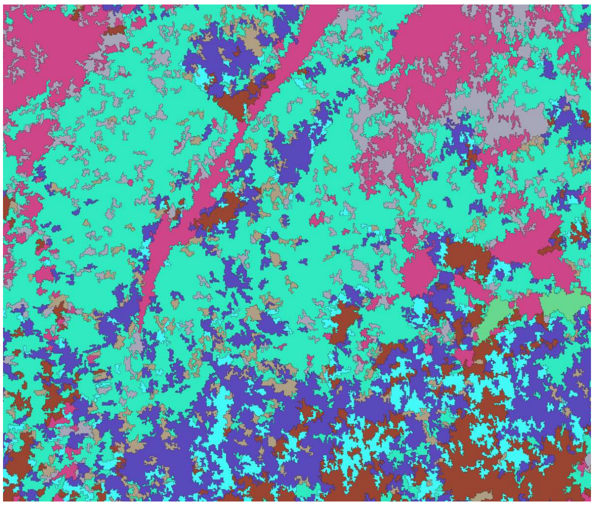
Usage:

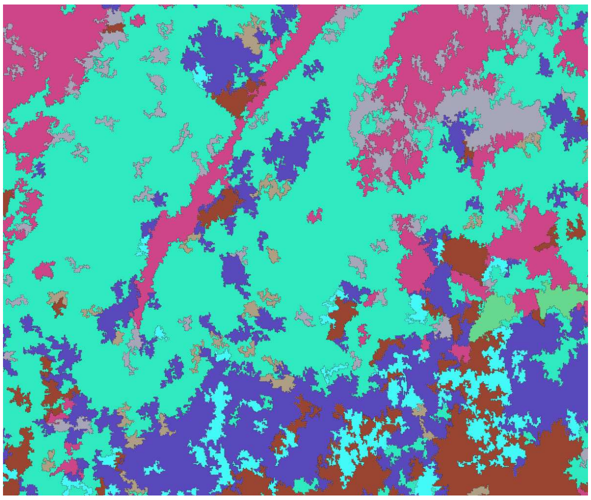
There are four parameters required for this analysis:

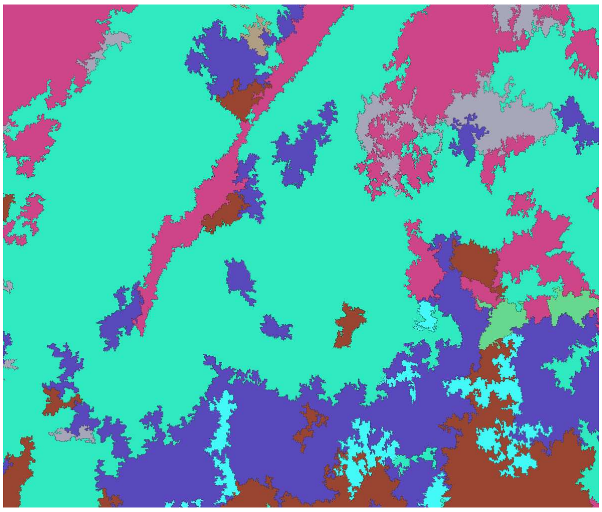
- A bathymetry single layer grid either from a pulldown list or by the directory and filename.
- A backscatter single layer grid either from a pulldown list or by the directory and filename.
- The number of classes is the number of expected characteristic clusters of data in n-dimensional space. A default value of 10 is suggested but can be altered by the user. Often the number of output classes actually produced is less than the number entered due to the small areal extent of a particular class and therefore deemed insignificant.
- The minimum size is a value in pixels for the smallest areal extent of any polygon class. Small polygons are merged with the largest neighbouring polygon.

Output is a single polygon vector shapefile and its default filename is the same as the bathymetry filename with "_MBESseg" added to the name. Format output is .shp (plus its associated companion files).

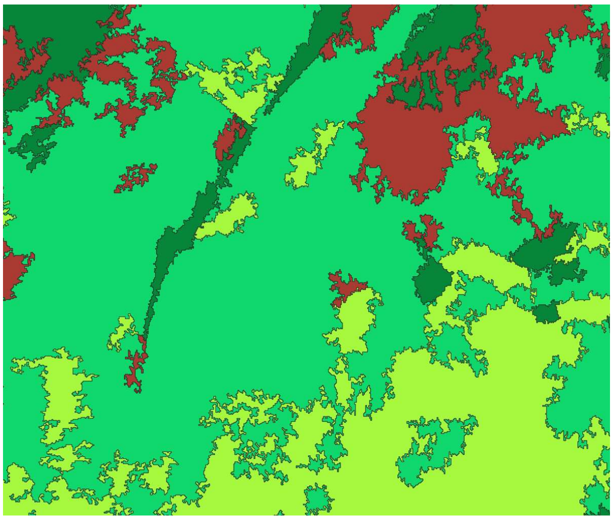
It is envisioned that this tool may be run several times to get the level of interpretation desired. Example below shows 10 classes but a change in the minimum size of polygons (from 30 to 100 to 400).







If the number of classes is reduced from 10 to 5 the classification boundaries are changed, and the analysis gives a similar but differing result:



OBIA (Object Based Image Analysis)

Rationale:

This tool creates a segmented vector polygon output file which characterises and segments the combination of imagery raster images (grids). Segmentation is a method to aggregate pixels together to create a thematic map. The segmentation process creates a set of polygons defined by the statistics associated with the input images. Clusters of the imagery pixels are created in n-dimensional space and created into classes. Aggregation into geographic regions (polygons) is done according to a minimum polygon size rule, and clustering rules.

The segmentation process uses a technique of k-means clustering and uses the iterative removal of outliers via a sieving process.

The resulting shapefile of polygons may take some time depending on the data complexity, file sizes, and number of clusters. Each polygon will have attributes of:

- VALUE – The Class Number of each polygon
- fileX_mean – The average of all pixel values in the polygon from input file X (1 to 5)
- fileX_stde – The standard deviation of pixel values in the polygon from input file X (1 to 5)

Usage:

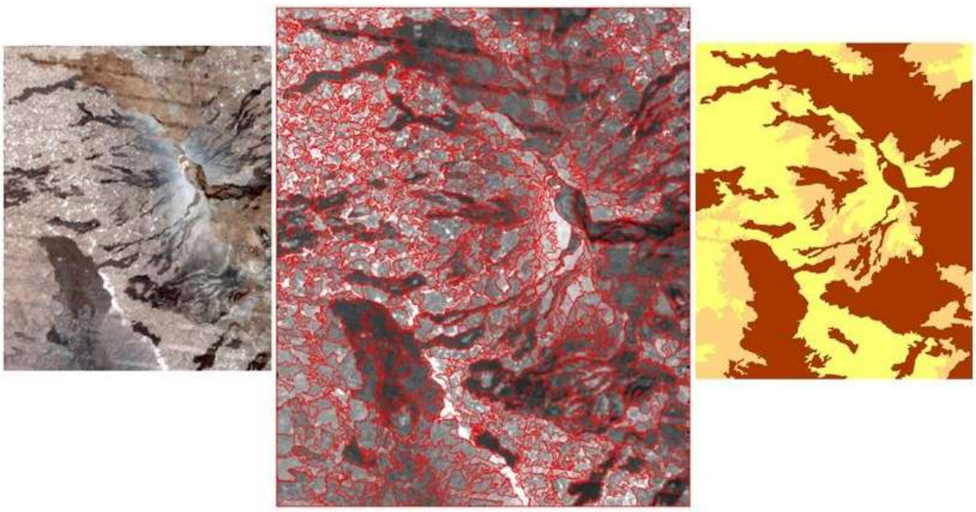
There are two parameters required for this analysis:

- The number of classes is the number of expected characteristic clusters of data in n-dimensional space. A default value of 10 is suggested but can be altered by the user. Often the number of output classes actually produced is less than the number entered due to the small areal extent of a particular class and therefore deemed insignificant.
- The minimum size is a value in pixels for the smallest areal extent of any polygon class. Small polygons are merged with the largest neighbouring polygon.

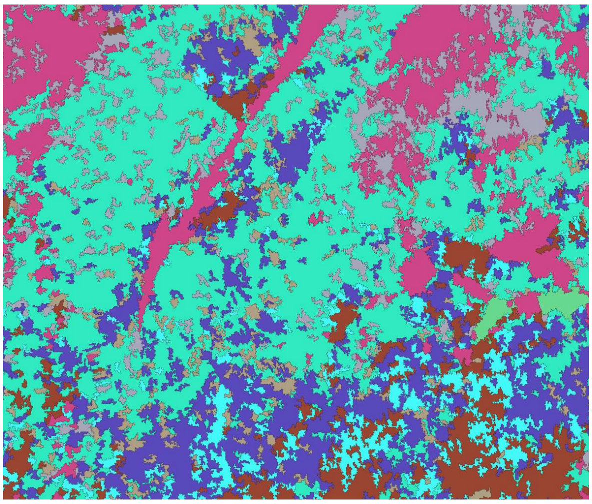
Inputs are single layered files – not geodatabases. A maximum of 5 input files are allowed. Output is a single polygon vector shapefile and its default filename is the same as the first input filename with "_OBIA" added to the name. Format output is .shp (plus its associated companion files).

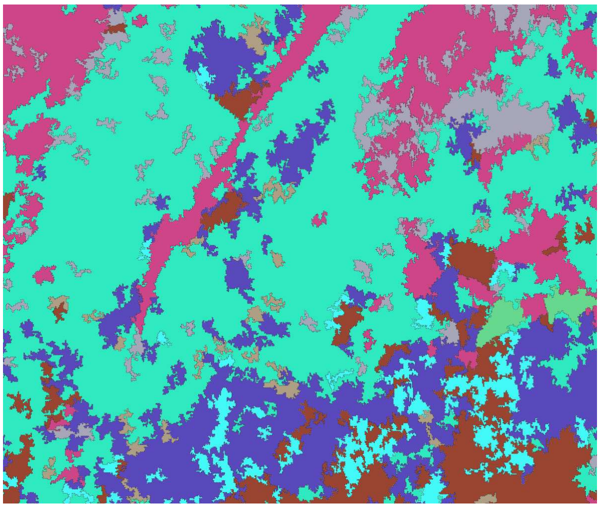
Examples:

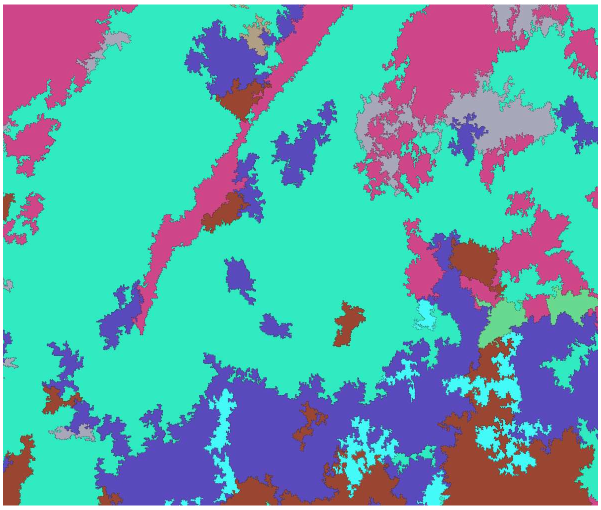
Satellite imagery bands – Before, Segmented boundaries, and Final Interpretation.



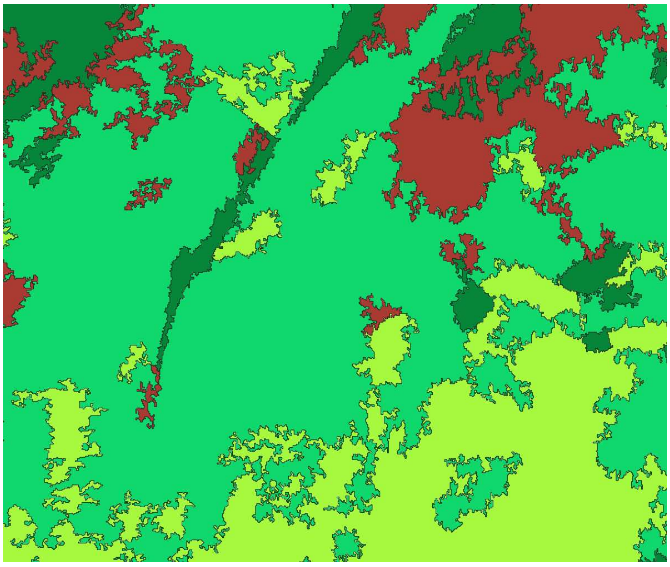
It is envisioned that this tool may be run several times to get the level of interpretation desired. Example below shows 10 classes but a change in the minimum size of polygons (from 30 to 100 to 400).







If the number of classes is reduced from 10 to 5 the classification boundaries are changed, and the analysis gives a similar but differing result:



Random Forest

Rationale:

To predict the likelihood of a distribution of a characteristic or feature over an area where some sample data is available but not so many individual samples to cover the whole area. It uses a method called “Random Forest” to create a prediction for data. Takes a file of groundtruth data and predicts presence data. If the groundtruth data has just presence data but no absence data the program will make an equal number of randomly placed absence points in the geographic area defined by the first raster image. The groundtruth data will have a mixture of groundtruth points with “Presence” or “Absence” criteria. The user can enter the definition of the presence points and all other points will be counted as absence.

The program will create a prediction raster image at a given resolution of the likelihood of the pixel being definitely defined as Presence (value 1) or Absence (value 0) and many values in between. Accuracy assessment can be determined by statistical values.

Values provided are:

- Errors of commission - sometimes also called "false positives."
- Errors of omission - a mistake that consists of not including something such as an amount or fact that should be included. Errors of omission are likely to be more common than errors of commission.
- Producer's Accuracy - the map accuracy from the point of view of the map maker (the producer). This is how often real features on the ground that are correctly shown on the classified map or the probability that a certain land cover of an area on the ground is classified correctly.
- The User's Accuracy - the accuracy from the point of view of a map user, not the map maker. The User's accuracy essentially tells us how often the class on the map will actually be present on the ground. This is often referred to as reliability.
- Overall Accuracy - the proportion of reference sites were mapped correctly. It is usually expressed as a percent, with 100% accuracy being a perfect classification where all reference sites were classified correctly.
- Cohen's Kappa - a metric that compares an Observed Accuracy with an Expected Accuracy (random chance). It considers random chance (0%) i.e. agreement with a random classifier, to 100% perfect prediction (with the given data), which generally means it is less misleading than simply using accuracy as a metric.

Usage:

There are several parameters required for this analysis:

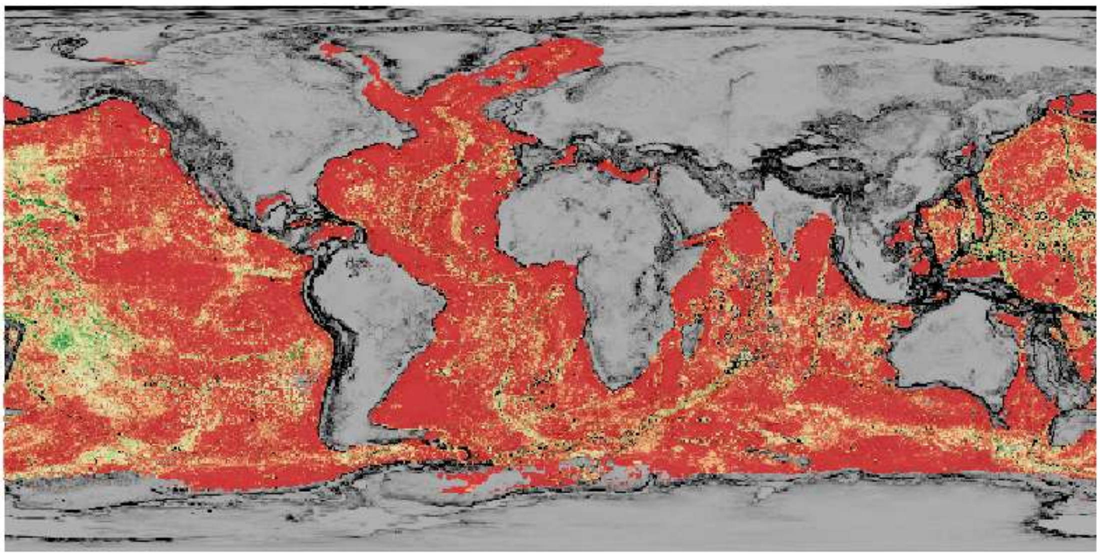
- The groundtruth point sample data giving the feature to be predicted. The point data must have an attribute field that allows determination of presence or absence of the feature. Examples of this are: attribute “Class” has values of “Crust” or “Not Crust”, or attribute “Coral” has values greater or less than 50%. The projection of the input sample data should be the same as the input predictor grids.
- A pulldown menu of the attributes in the Presence (and Absence) datafile.
- The criteria for prediction presence. This can either be a text criterion (using single quotes such as **= ‘Crust’**) or a numeric criterion (such as **> 49.9**)

- A proportion of the input point sample data is held back from the model production and prediction for testing of the final output. A random selection is made of the input sample data being a percentage of the total number of input points.
- Next is the predictor grid files required to do the prediction. Initially, it is suggested that many predictors are entered, but in later predictions, some of those input layers can be omitted due to not be of much use in the final predicted output. All the predictors should be single layered grids in .img or .tif format.
- The extent box can be used to reduce the area of the final prediction grid. This is used to speed up the process or for testing the data. Default extent is the extent of the first predictor grid entered. Output will not be calculated for a pixel if there is a missing value in any of the input predictive grid.
- If all input points are used in the prediction points outside the extent box will used in the production of the prediction model and subsequently affect the output prediction.
- The resolution of the final grid can be in metres or degrees but should be in the same units as the input predictor grids.
- The resulting output grid has values between 1 and 0 for presence and absence. Intermediate values give a level of likelihood. For the statistical calculation of binary presence and absence a cutoff value of 0.5 is often used, but can be nuanced by biasing the resulting division.
- Output is a single polygon vector shapefile and its default filename is the same as the first input sample point filename with "_prediction_<PresenceField>" added to the name. This is default but can be edited by the user.
- The software will create many intermediate files and will put these is a temporary directory /tempMT. Most of these can be deleted by the software but some cannot be removed until exiting QGIS
-

The prediction output grid is accompanied by three other files:

- "RF_Model.Rdata" has the model information used in the prediction
- "RF_Model_importance.png" has a graph of the importance of each of the variable in the model production
- "RF_Model_summary.txt"" has the model statistics.
-

For example:



```
MODEL SUMMARY:
==============

Call:
 randomForest(formula = Presence2 ~ AgeMap_M00 + KEnergy_01 + Producti02,
      ntree = 500s, replace = FALSE, importance = TRUE)

               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 1

          Mean of squared residuals: 0.326
                    % Var explained: 45.6

Estimated predictor variable importance:

           %IncMSE IncNodePurity
AgeMap_M00   0.312           151
KEnergy_01   0.267           152
Producti02   0.198           113

Presence Absence File used = SplAndSedPacificNoDup.shp
Working in directory       =
C:/Users/tlb/Documents/Data/world/Q_RFtest/testimgPacific
Using presence identity      "Class" = 'Crust'
Number of samples used     = 1739
A random selection of      = 1304 samples for model creation
The remainder of samples   = 435 for model testing (25.0 %)
Input files used:
                           = AgeMap_Ma_x100Pacific.img  AgeMap_M00
                           = KEnergy_maxBothPacific.img  KEnergy_01
                           = Productivity2019nullPacific.img  Producti02
Area extent predicted      = -178.5500, -137.2166,-7.1166,35.1499
Output resolution of grid  = 0.2

Confusion Matrix (based on 50% value presence absence)
                      Presence  Absence
  Predicted Presence     43       17
  Predicted Absence      38       333

                      Errors of    Errors of    Producer     User
                      Commission   Omission     Accuracy     Accuracy
  Predicted Presence   0.28         0.46         0.53         0.71
  Predicted Absence    0.10         0.04         0.95         0.89

  Overall Accuracy       87.238 %
  Cohen's Kappa          53.566 %
```

Grey Level Co-occurrence Matrices

Rationale:

This tool creates Grey-Level Co-occurrence Matrices (GLCM) grids for an input grid, such as a backscatter mosaic. Matrices include: Contrast, Dissimilarity, Homogeneity, Energy, Entropy, Angular Second Momentum (ASM), Standard Deviation, and Mean. Options include pixel box size, sampling distance, grey levels and pixel pair angle. These equate to different types of texture measurements of the input imagery. They are sometimes referred to as second-order derivatives.

Usage:

This tool requires six user inputs:

- An input image. This image will be converted into an 8 bit grey-scale image for processing. This bathymetry file must be in a meter projection such as UTM or Mercator and not geographic coordinates.
- Eight GLCMs are available for calculation, the pull-down menu lists all of them.
- Sampling Box is the size of the area in pixels over which the texture is measured. Default size is 5 pixels square. A larger box will result in lower resolution detection.
- Sampling distance is the pixel to pixel distance measuring the image's calculation of texture. Default size is 1 pixel. A larger value may result in smoother changes in texture.
- Number of grey-levels is set as 8 as default. It reduces the number of texture levels. Greater values will slow the calculation process and may be limited by the available memory of the processor
- The pixel pair angle can be altered by the user to any direction. It is the direction that comparison and calculations are made for the texture mapping equations.

The output raster filename is automatically generated from the input filename plus the GLCM type and its 4 parameters. This can be edited by the user

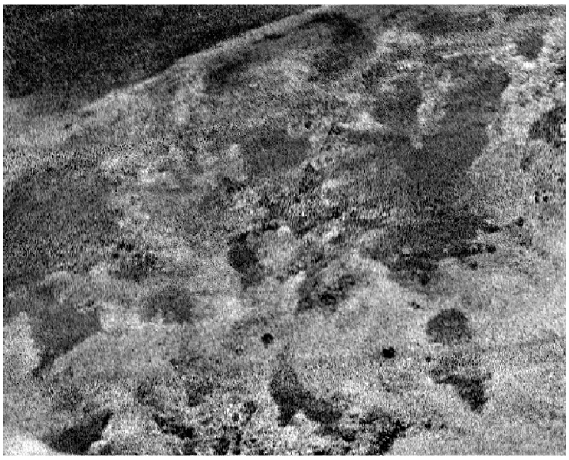
The equations and some descriptors for the GLCMs are:

$Contrast = \sum_i \sum_j (i-j)^2 * P(i,j)$	Contrast, also known as inertia, is a measure that describes the degree to which the intensity of different pixels in an image varies. If contrast is high, it means that there is a lot of variation in pixel intensity.
$Dissimilarity = \sum_i \sum_j \lvert i-j \rvert * P(i,j)$	Measures the average difference in intensity between neighbouring pixels. High dissimilarity values indicate greater heterogeneity in texture.
$Homogeneity = \sum_i \sum_j \frac{P(i,j)}{1+(i-j)^2}$	Homogeneity is a measure that reflects the degree to which the matrix elements are close to the diagonal of the GLCM matrix. The homogeneity value will be high if the matrix elements that have high values are located near the diagonal.

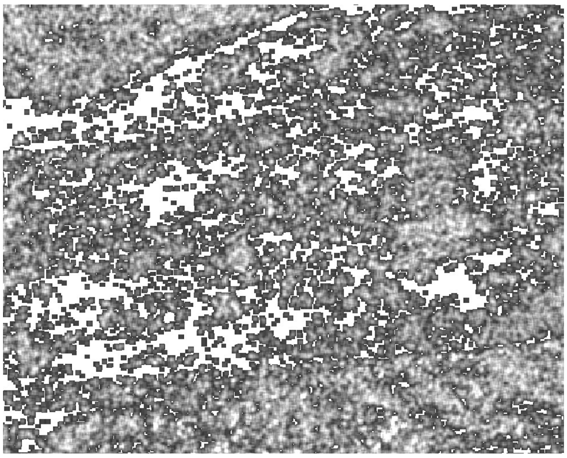
$Energy = \sqrt{\sum_i \sum_j P(i,j)^2}$	The energy value will be high if the elements in the image tend to have the same or uniform values.
$Entropy = -\sum_i \sum_j P(i,j) * \log(P(i,j))$	Entropy is a measure of the complexity of the texture or information contained in an image. Higher entropy values indicate higher texture or information complexity.
$Correlation = \frac{\sum_i \sum_j [ij * P(i,j)] - \mu_x * \mu_y}{\sigma_x * \sigma_y}$ *Similar to Standard Deviation and Variance*	Correlation is a measure that describes the extent of the relationship between two pixels. A high correlation indicates a strong dependency between two pixels.
$Angular\ Second\ Momentun = \sum_i \sum_j P(i,j)^2$	ASM is very similar to energy value output, where values will be high if the elements in the image tend to have the same or uniform values.
$Mean = \sum_i \sum_j P(i,j)$	Mean measures the frequency of the combination with neighbouring pixel values.

Example:

Multibeam backscatter



Contrast – 5x5 box 1 pixel sampling



Homogeneity – 5x5 box 1 pixel sampling

Entropy – 5x5 box 1 pixel sampling

Estimate new MBES Coverage

Rationale:

This tool creates a vector polygon output file which shows the estimated coverage of a MBES system predicted to be acquired over the area. It uses low resolution bathymetric data, such as GEBCO as a guide to swath width. It is not an exact representation of the actual coverage but a guide to placing tracklines.

Usage:

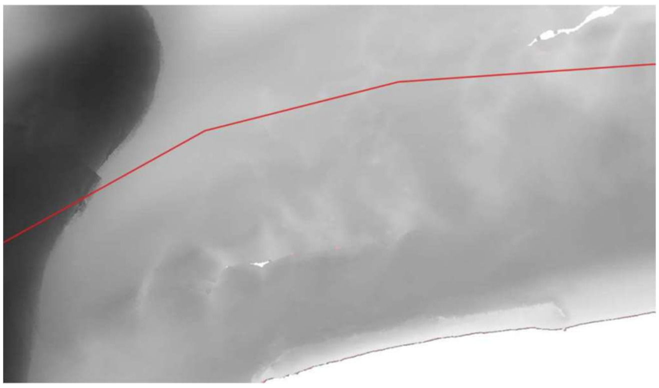
This tool requires three inputs::

- A background bathymetry grid to be used for approximate depth estimation. This bathymetry file must be in a meter projection such as UTM or Mercator and not geographic coordinates. GEBCO data is suggested for deep water.
- A trackline shapefile of the suggested survey line. It can be a single line or multiple lines.
- The total swath angle for the MBES. A default value of 140º is provided but can be changed by the user.

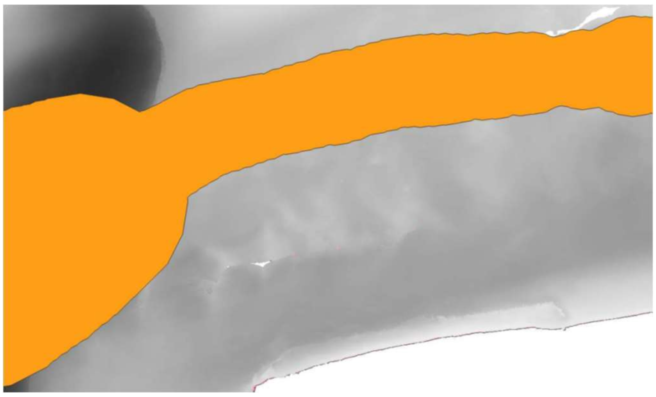
Output is a single polygon vector shapefile and its default filename is the same as the trackline filename with "_coverage" added to the name. Format output is .shp (plus its associated companion files).

Example:

Bathymetry data and trackline



Swath coverage (140º)



Raster Coverage

Rationale:

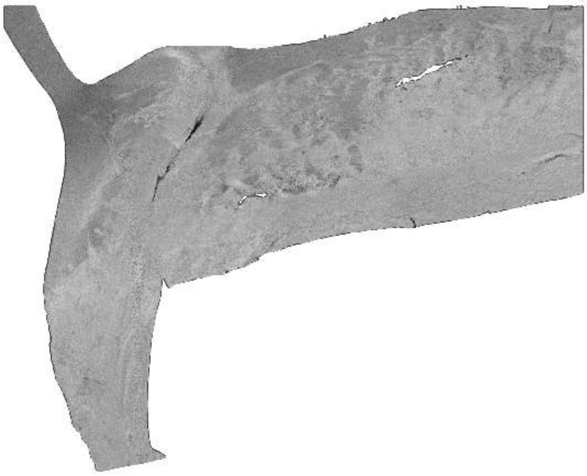
This tool creates a vector polygon output file which shows the coverage of a raster imagery datafile.

Usage:

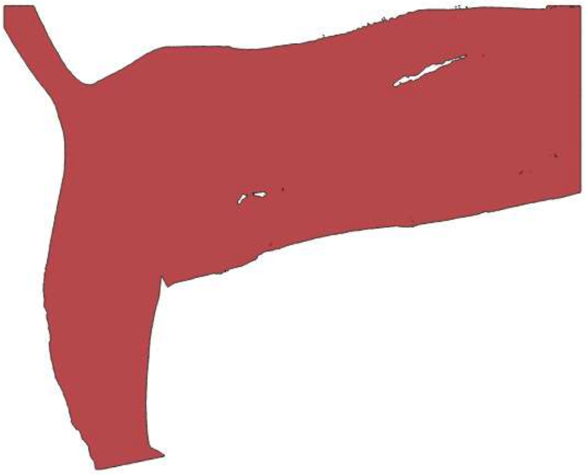
The input for this tool is a raster image grid, such as bathymetry or backscatter. It creates a polygon around the data and assumes that no coverage is defined by NoData pixels. If no data is defined by zero or another value, these will have to be changed to NoData.

Example:

Bathymetry data



Polygon coverage



Survey Lines

Rationale:

This tool creates a continuous survey of tracklines at a given azimuth and spacing to cover a user-defined area. Output consists of two vector shapefiles; one showing the trackline and one consisting of the waypoints to be used. An input polygon gives the area to be covered but the user can define whether the waypoints are on the edge of the area (giving coverage outside the area) or can be slightly inside the boundary as the swath will be wide enough to get to the polygon edge.

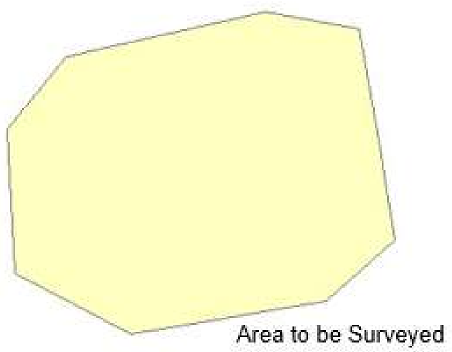
Usage:

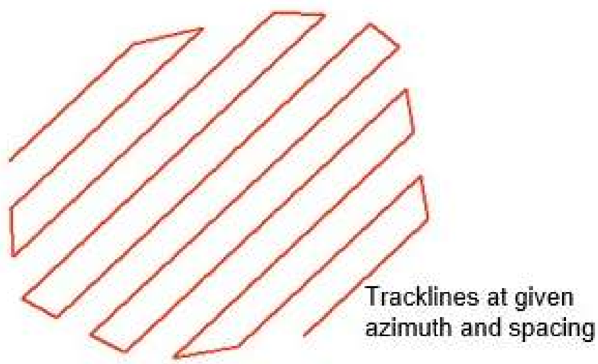
There are several parameters required for this tool:

- A polygon shapefile of the area to be surveyed. A simple polygon is best (e.g. rectangular but can be any shape or complexity. Concave edges may result in survey lines outside the area unless specified to be kept inside the polygon. In these cases the survey lines may not always be the most efficient.
- Direction (azimuth) of the survey lines – defined as 0 being North/South, 90 being East/West. Values should be (0.0 to 179.99). Default value is 0 but the user can change this as desired.
- Line spacing in metres. Default value is 100m but the user can change this as desired.
- The “Outside area coverage” tick box allows the waypoints to be on the polygon edge rather than within the area by the width of the swath (being half the line spacing).
- The “Start at far end” tick box reverses order of the output waypoints shapefile
- The “Reverse lines” tick box changes the start point of each main line so that the survey is flipped.
- The “Allow concave shape” tick box when ticked stops coverage lines in the concave area of the polygon. Useful when going round islands.

Outputs are two vector shapefiles and have default filenames and put in the same directory as the is the area polygon shapefile. The waypoints filename has “_wayPoints” added to the name, and the survey lines filename has “_surveyLines” added to the name. Format outputs are .shp (plus its associated companion files).

Example:





CSV to Grid

Rationale:

This tool converts Comma Separated Values files (.csv) into raster grids or vector point files. The files are often output from Excel or similar program in text form. The data needs to have columns Latitude and Longitude values in decimal degrees (or X and Y values in some map known projection and datum system), together with a column of values for gridding (e.g. depth, temperature etc.) The tool will look in a single directory for all the .csv files and try to convert all of them into grids or point files. The tool will calculate the best grid resolution from the input values – based on the median spacing between data points. It will also output a shapefile of the data points in the .csv file. The input files will need a top row of column descriptors.

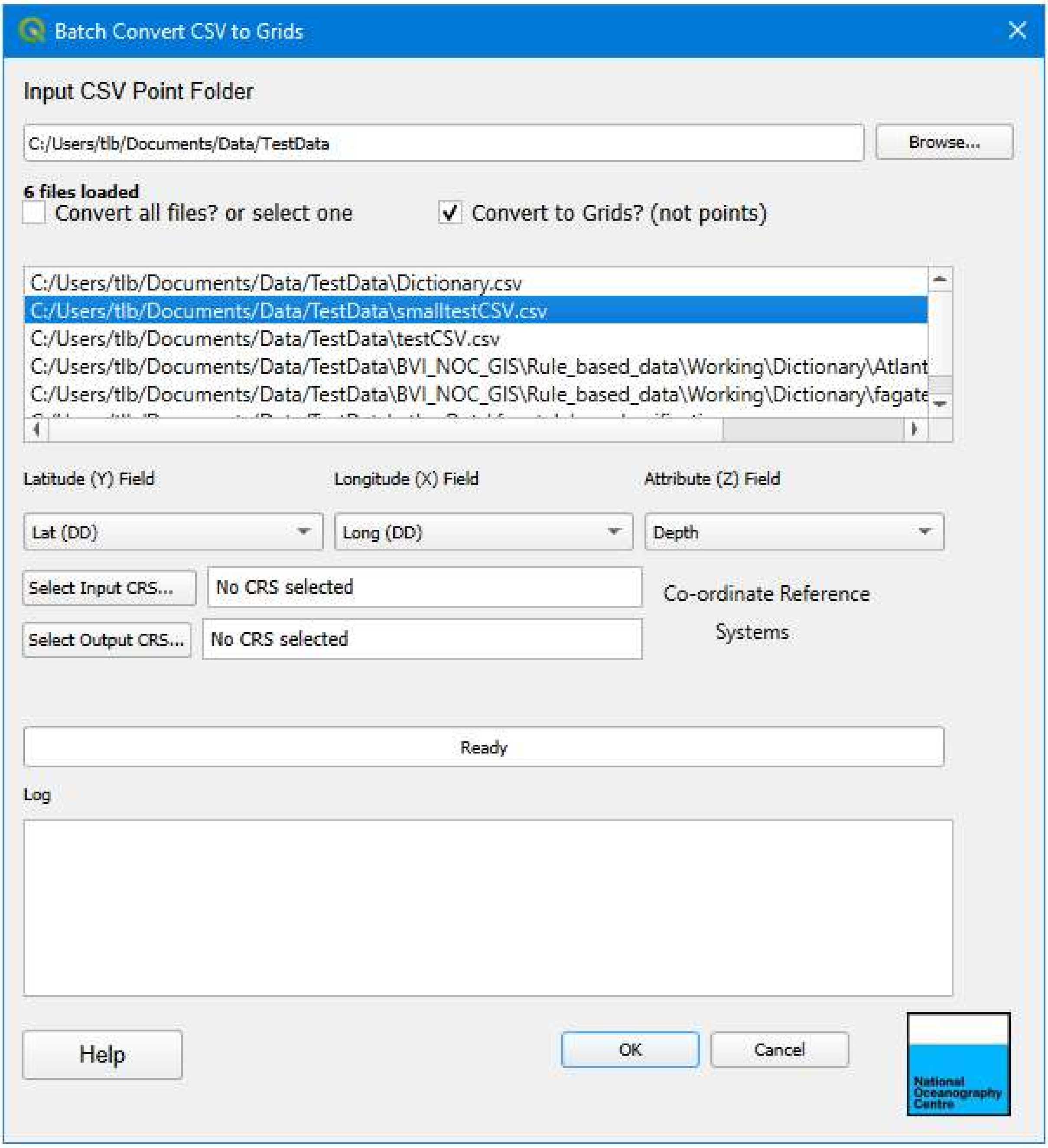
Usage:

There are several parameters required for this tool:

- A folder or directory where the input .csv files can be found. This is defined by clicking on the "Browse…" button and selecting the appropriate directory. It will take a couple of seconds for the directory to be found and searched. The files will be displayed in the box below
- The user can decide whether to convert all the files displayed or only one. Individual files can be selected and highlighted by a mouse click.
- The user can also decide whether the conversion of the .csv file(s) are to a shapefile of points or to a grid.
- The Latitude (Y), Longitude (X), and Attribute (Z) fields will be populated with the top line values of the first .csv file found. If any field is incorrect for usage the pulldown arrows will allow the user to choose more appropriate columns. If multiple files are to be converted they must have the same fields as the first file.
- The input values for X and Y need a co-ordinate reference system (CRS) to be defined. The "Select Input CRS…" button will open a new window and the user can select the appropriate CRS of the input file. Latitude and Longitude with WGS84 datum is default (and known as EPSG: 4326) and if left as "No CRS selected" this projection will be assumed. There are thousands of possible projections and datums available.
- The output points and grid also need a co-ordinate reference system (CRS) to be defined. The "Select Output CRS…" button will open a new window and the user can select the appropriate CRS of the output files. Latitude and Longitude with WGS84 datum are default (and known as EPSG: 4326) and if left as "No CRS selected" this projection will be assumed. It is recommended that a meter-based projection system and datum is used for outputs, such as UTM (Universal Transverse Projection) for the appropriate longitude zone.
- When the "Apply" button is clicked the progress bar will sweep across and log messages will appear in the dialog box below. Processing may take a couple of minutes depending on the file sizes involved.

Output is either a vector shapefile and raster grid or just he former (per input .csv file or files) and have default filenames as the input files except with filetype of .shp and .img respectively. They will be located in the same directory as the original .csv file.

Example:


Batch Convert CSV to Grids
Input CSV Point Folder
C:/Users/tlb/Documents/Data/TestData
Browse...
6 files loaded
Convert all files? or select one
Convert to Grids? (not points)
C:/Users/tlb/Documents/Data/TestData\Dictionary.csv
C:/Users/tlb/Documents/Data/TestData\smalltestCSV.csv
C:/Users/tlb/Documents/Data/TestData\testCSV.csv
C:/Users/tlb/Documents/Data/TestData\BVI_NOC_GIS\Rule_based_data\Working\Dictionary\Atlant
C:/Users/tlb/Documents/Data/TestData\BVI_NOC_GIS\Rule_based_data\Working\Dictionary\fagate
Latitude (Y) Field
Longitude (X) Field
Attribute (Z) Field
Lat (DD)
Long (DD)
Depth
Select Input CRS...
No CRS selected
Co-ordinate Reference Systems
Select Output CRS...
No CRS selected
Ready
Log
Help
OK
Cancel
National Oceanography Centre

Listing of properties of layers

Rationale:

This tool creates a table of the properties of all the layers in the project. The properties include:

- Layer name
- Filename (with extension)
- Directory
- Projection name
- EPSG code of projection
- Format of layer e.g. raster layer or type of vector layer
- If a vector layer, the number of features
- If a vector layer, the type of features (point, line or polygon)
- X minimum
- X maximum
- Y minimum
- Y maximum
- If a raster layer, the Minimum Pixel Value
- If a raster layer, the Maximum Pixel Value
- If a raster layer, the Number of rows in the image
- If a raster layer, the Number of columns in the image
- If a raster layer, the pixel resolution in the X direction
- If a raster layer, the pixel resolution in the Y direction

The table will be stored in a file called LayerList.csv

Usage:

This tool requires one user input:

- An output directory. There will be a pulldown list of all the directories used in the project. The user can choose which directory to store the layer list .csv file.

The output LayerList file is automatically loaded onto project and can be viewed by its attribute table, or by any text or spreadsheet program.

Example:

	Layer	Filename	Directory	Projection	EPSG	Format	N
1	AS5M127andM...	AS5M127andM131_interp_treeFilt_7_...	C:/Users/tlb/Do...	WGS 84 / UTM ...	EPSG:32629	HFA/Erdas Ima...	
2	AS5M127andM...	AS5M127andM131_interp_treeFilt_slo...	C:/Users/tlb/Do...	WGS 84 / UTM ...	EPSG:32629	HFA/Erdas Ima...	
3	AS5M131_NoN...	AS5M131_NoNadir2.img	C:/Users/tlb/Do...	WGS 84 / UTM ...	EPSG:32629	HFA/Erdas Ima...	
4	Autosub5_Norb...	Autosub5_Norbit_Janus_NavM131.shp	C:/Users/tlb/Do...	WGS 84	EPSG:4326	ESRI Shapefile	7
5	part127bathy	part127bathy.tif	C:/Users/tlb/Do...	WGS 84 / UTM ...	EPSG:32629	GTiff/GeoTIFF	
6	partM131	partM131.shp	C:/Users/tlb/Do...	WGS 84	EPSG:4326	ESRI Shapefile	5

No_Features	Type	Xmin	Xmax	Ymin	Ymax	MinValue	MaxValue	Rows	Columns	Xres	Yres
		447073.0	453729.0	5344408.0	5348888.0	-510.14...	-321.8...	6656	4480	1.0	1.0
		447073.0	453729.0	5344408.0	5348888.0	0	39.530...	6656	4480	1.0	1.0
		446427.0	455276.75	5344312.25	5348868.0	0	253	35399	18223	0.25	0.25
788931	Point	-9.709937	-9.620965	48.251111	48.290629						
		450387.0	451691.0	5346482.0	5347628.0	-398.37...	-355.1...	1304	1146	1.0	1.0
57239	Point	-9.667461	-9.652724	48.270027	48.279805						

R programming in QGIS

Rationale:

The R software system has many useful numerical and statistical functions that can be used in the environment of QGIS. Linking the two systems is most advantageous to utilise the power of both software whilst using the tools.

Installation:

- To install R, go to https://www.cran.r-project.org
- Click Download R for <your system>.
- Install R Click on “install R for the first time”.
- It is suggested for the current set of tools to use a previous release (3.6.2) rather than the latest version
- Click on “Previous releases”
- Click on “R 3.6.2 “
- Click on “R-3.6.2-win.exe “
- Save the file to some user-controlled diskspace (e.g. Windows downloads)

Index of /bin/windows/base/old/3.6.2

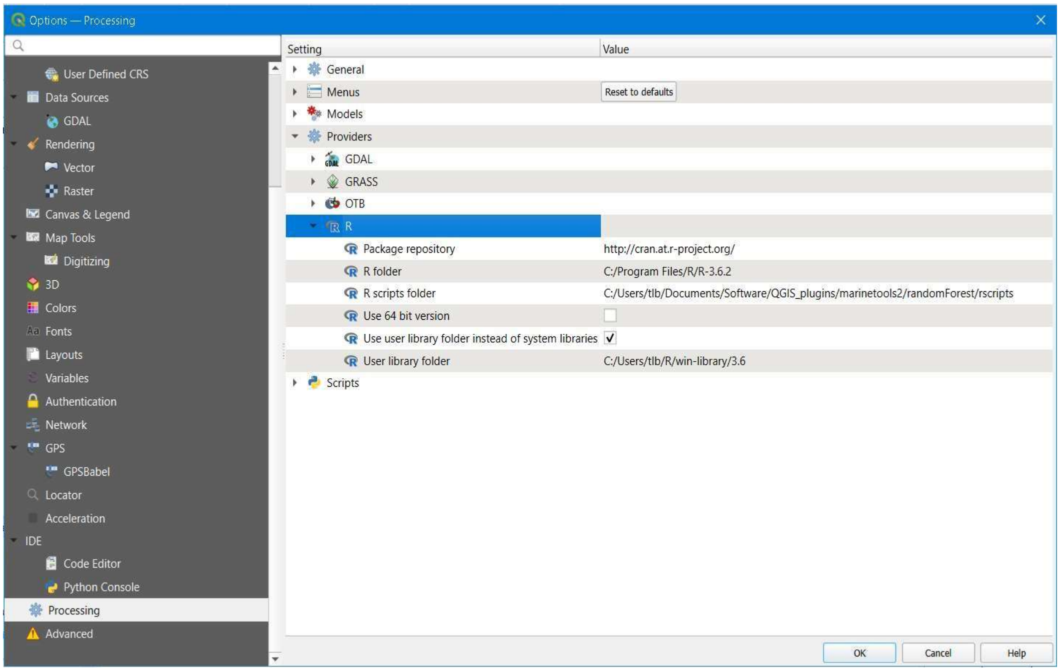
Name	Last modified	Size	Description
Parent Directory		-	
CHANGES.R-3.6.2.html	2019-12-12 10:43	39K	
NEWS.R-3.6.2.html	2019-12-12 10:43	457K	
R-3.6.2-win.exe	2019-12-12 10:43	82M	
R.css	2023-01-19 13:32	1.8K	
README.R-3.6.2	2019-12-12 10:43	8.5K	
SVN-REVISION.R-3.6.2	2019-12-12 10:43	46	
md5sum.txt	2019-12-12 10:43	50	
release.html	2019-12-12 10:43	90	
rw-FAQ.html	2019-12-12 10:43	103K	

Apache Server at cran.r-project.org Port 443

- Run the executable just downloaded with Administrative privilege (Run as administrator). e.g. In Windows this will install the software in C:\Program Files\R\R-3.6.2
- In QGIS **Plugins – Manage and Install Plugins… - Not installed**
- Search for “Processing R Provider”

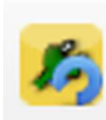
- Click on “Install plugin”

- Click Close
- In QGIS **Settings – Options – Processing – Providers – R**
- Double click the “User library folder” and change it to a user held library e.g. C:/Users/<user>/R/win-library/3.6
- Double click the “R folder” and add the main folder
- e.g. C:/Program Files/R/R3.6.2
- Click on OK



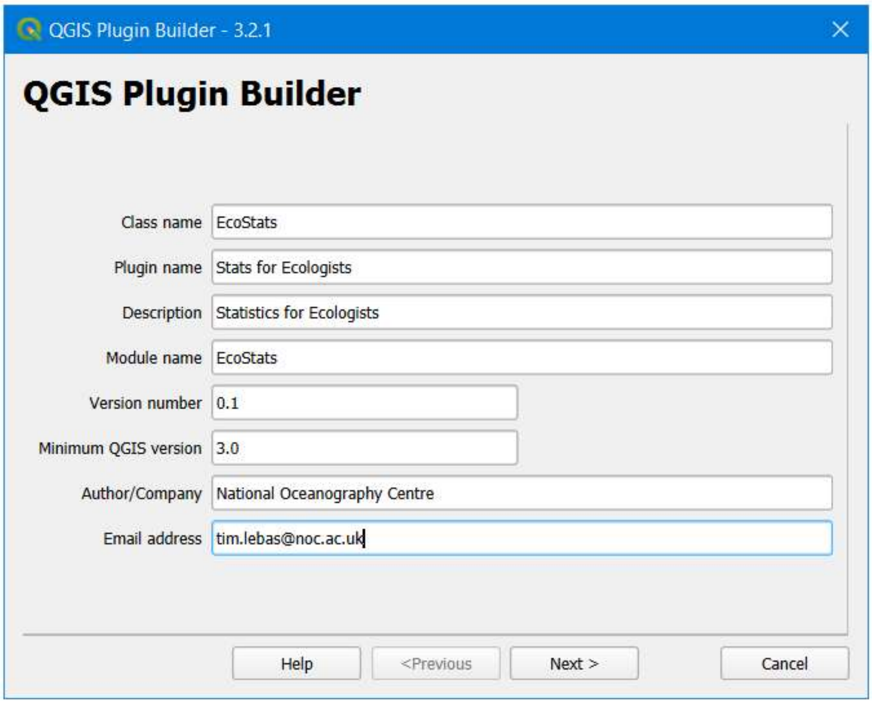
Making a QGIS plugin

There a few pre-requisites before programming can start:

- Download OSgeo4W installer - https://trac.osgeo.org/osgeo4w/
- Download QTCreator - https://download.qt.io/archive/qtcreator/10.0/10.0.2/qt-creator-opensource-windows-x86_64-10.0.2.exe This may come packaged with QGIS already.
- In QGIS – install plugins Plugin Builder  and Plugin Reloader 
- Set environment variable QGIS_PYTHONPATH for your id, to your software building directory:
- If you wish to publish your plugin (worldwide) you will need an OsGeo UserID using: https://id.osgeo.org/ldap/create to get a “mantra” and it may take a few days for it to be ratified.
- For information: The default (system) plugin directory is in:

  C:/Users/<user>/AppData/Roaming/QGIS/QGIS3/profiles/default/python/plugins

Plugin Builder



This creates a directory “ecostats”:

EcoStats.py - main program

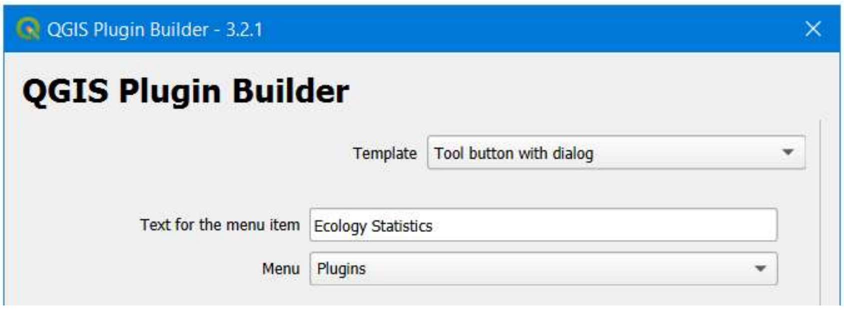
EcoStats_dialog_base.ui - GUI program

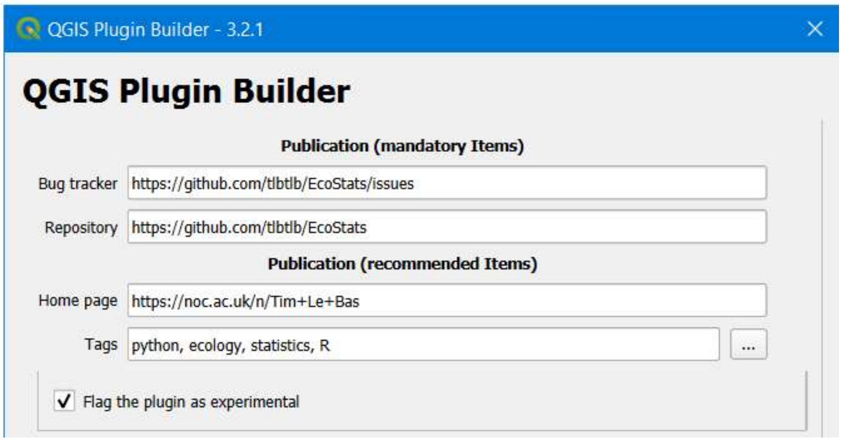
metadata.txt - ancillary data for publication

resources.qrc - icon forwarder

resources.py

EcoStats_dialog.py - program

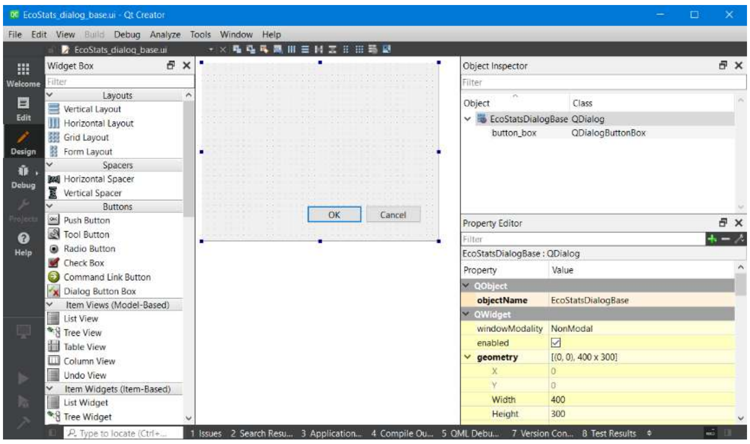




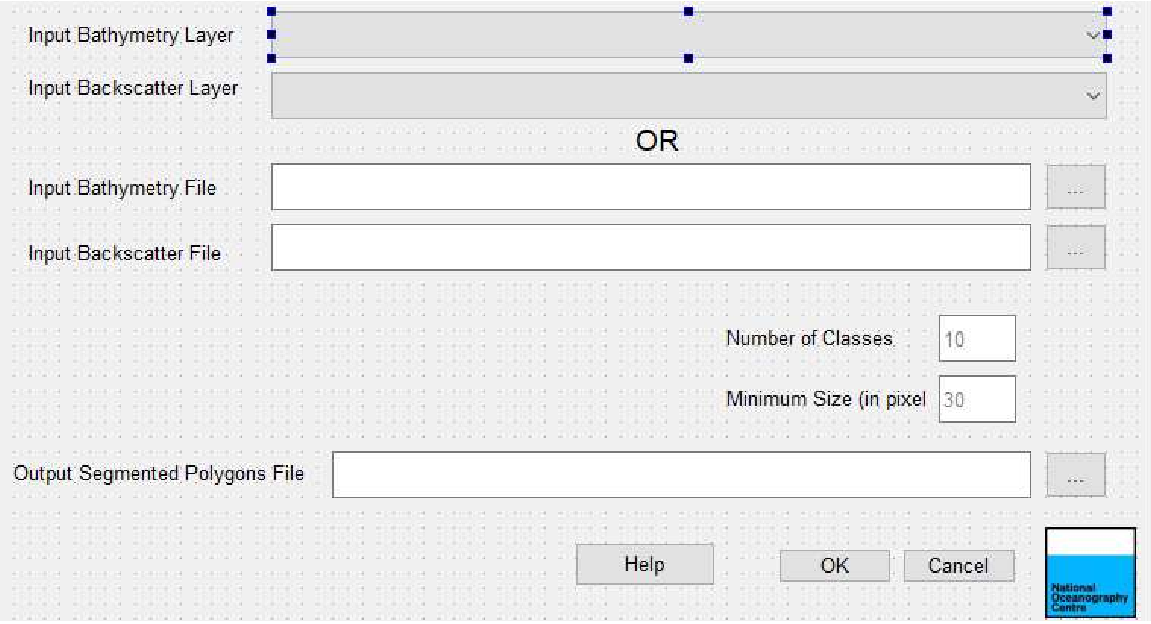
Remember to have the Bug tracker and Repository ready before publication and probably put on Github.

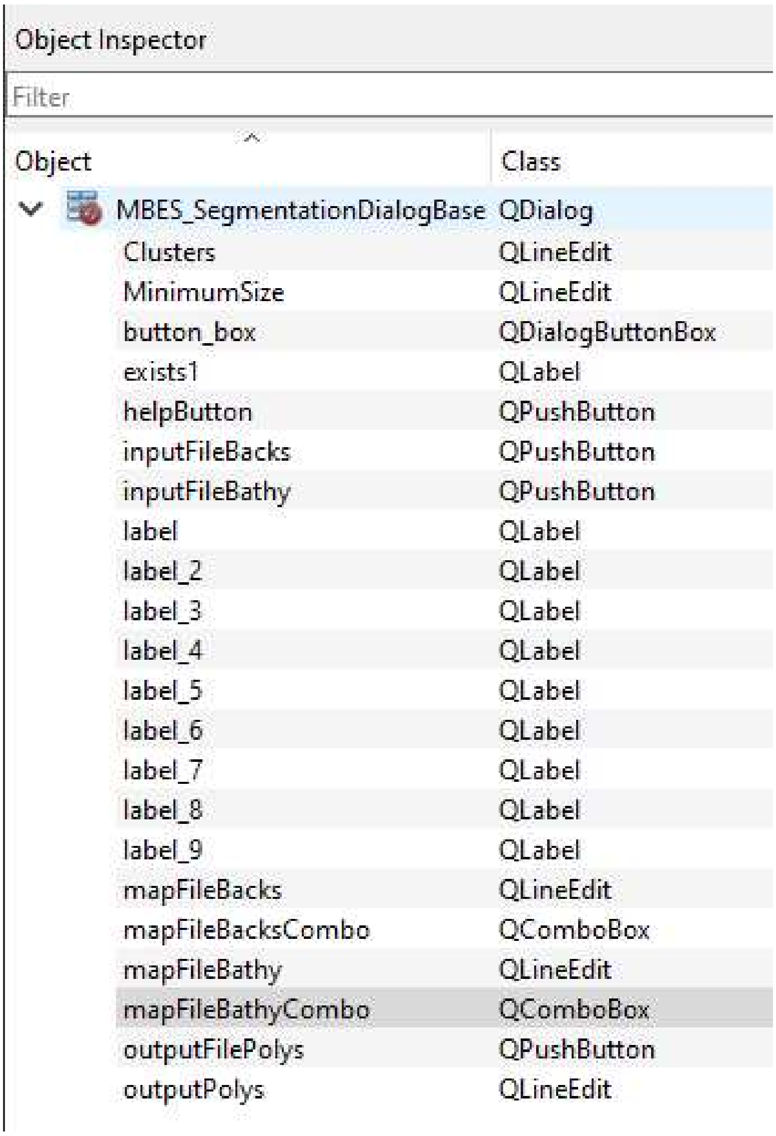
Making the GUI

- Open “Qt Creator”
- File –> Open File or Project….
- C:/Users/tlb/Documents/Software/QGIS_plugins\ecostats\EcoStats_dialog_base.ui
- Drag and drop item such as:
    - Combo Box – for visible map layers input
    - Line Edit – for files and text
    - Label – for text and icons on GUI
    - PushButton – for action button
    - Check Box or Radio Button – for Boolean action
- Right hand side for property editor
- Take note of button names!



Example:


Input Bathymetry Layer
Input Backscatter Layer
OR
Input Bathymetry File
Input Backscatter File
Number of Classes
10
Minimum Size (in pixel
30
Output Segmented Polygons File
Help
OK
Cancel
National Oceanography Centre


Object Inspector
Filter
Object
Class
MBES_SegmentationDialogBase QDialog
Clusters QLineEdit
MinimumSize QLineEdit
button_box QDialogButtonBox
exists1 QLabel
helpButton QPushButton
inputFileBacks QPushButton
inputFileBathy QPushButton
label QLabel
label_2 QLabel
label_3 QLabel
label_4 QLabel
label_5 QLabel
label_6 QLabel
label_7 QLabel
label_8 QLabel
label_9 QLabel
mapFileBacks QLineEdit
mapFileBacksCombo QComboBox
mapFileBathy QLineEdit
mapFileBathyCombo QComboBox
outputFilePolys QPushButton
outputPolys QLineEdit

Changing the plugin Icon

- Default icon is  icon.png
- Change the resources.qrc file and

  new icon 

```
<RCC>
    <qresource prefix="/plugins/EcoStats" >
        <file>icon.png</file>
    </qresource>
</RCC>
```

- Open “OSGeo4W Shell” window

  cd C:/Users/tlb/Documents/Software/QGIS_plugins\ecostats
  pyrcc5 -o resources.py resources.qrc

Making Default Output names

- For LineEdit in run

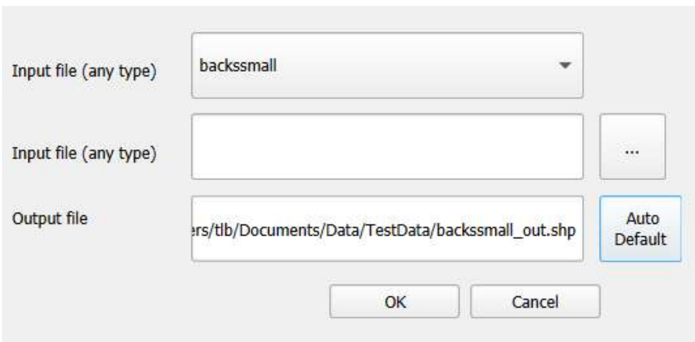
```
self.dlg.pushButton_1.clicked.connect(self.select_input_file1)
```

```
def select_input_file1(self):
    filename, _filter = QFileDialog.getOpenFileName(selfMT.dlg, "Select
                          input bathymetry file ","", '*.img *.tif*')
    selfMT.dlg.mapFileBathy.setText(filename)
    autoname = filename[:-4]+"_out.shp"
    selfMT.dlg.lineEdit.setText(autoname)
```

- For ComboBox in run

```
        layers = QgsProject.instance().mapLayers().values()
        self.dlg.comboBox.clear()
        self.dlg.comboBox.addItems([layer.name() for layer in layers])
        self.dlg.pushButton.clicked.connect(self.combo_change)
```

```
def combo_change(self):
    selectedLayerIndex = self.dlg.comboBox.currentIndex()
    a=0
    layers = QgsProject.instance().mapLayers().values()
    for layer in layers:
        if a == selectedLayerIndex:
            autoname = str(layer.source())[:-4]+"_out.shp"
            self.dlg.lineEdit.setText(autoname)
            a=a+1
LineText = self.dlg.lineEdit.text()
```



Running of a program and using functions

```
if DoContours:
    # make contours output to filename4 and use newInterp
    dict = processing.run("native:rasterlayerproperties",
                           {'INPUT':newInterp,'BAND':1})
    cellsizeX = float(dict['PIXEL_WIDTH'])
    cutoff = cellsizeX * 10.0
    result = processing.run("gdal:contour"
           {'INPUT':newInterp,'BAND':1,'INTERVAL':Interval,
           'FIELD_NAME':'ELEV','CREATE_3D':False,'IGNORE_NODATA':False,
           'NODATA':None,'OFFSET':0,'OUTPUT':'TEMPORARY_OUTPUT'})
    newContour1 = result['OUTPUT']
    result = processing.run("qgis:exportaddgeometrycolumns",
           {'INPUT':newContour1,'CALC_METHOD':0,
           'OUTPUT':'TEMPORARY_OUTPUT'})
    newContour2 = result['OUTPUT']
    processing.run("native:extractbyattribute",
           {'INPUT':newContour2,'FIELD':'length','OPERATOR':2,
           'VALUE':cutoff,'OUTPUT':filename4})
    fname = os.path.dirname(str(filename4))
    vlayer = QgsVectorLayer(str(filename4),str(filename4[len(fname)+1:]),
           "ogr")
    QgsProject.instance().addMapLayer(vlayer)

if DoRoughness:
    # Calculate Roughness output to filename5 and use newInterp
    newRoughness = newdir + "/newRoughness"+rand+".img"
    processing.run("gdal:roughness",
           {'INPUT':newInterp,'BAND':1,'COMPUTE_EDGES':False,
           'OPTIONS':'','OUTPUT':filename5})
    fname = os.path.dirname(str(filename5))
    vlayer = QgsRasterLayer(str(filename5),str(filename5[len(fname)+1:]))
    QgsProject.instance().addMapLayer(vlayer)
```

Publishing a plugin

- An OSGeo UserID needed
- Tidy directory removing unnecessary files and directories.
- Add a LICENCE file (standard GNU text?)
- Zip up the directory of the tool – e.g. ecostats.zip
- Open https://plugins.qgis.org/plugins/ and login
- Select "PLUGINS" at top
- Click on "Upload a plugin"
- Click the Experimental
- Browse and select the zip file
- Click on "Upload"
- It will be checked by someone in the QGIS administration before publishing

- Remove directories

__pycache__
help
i18n
scripts
Test

- and files:

Makefile
pb_tool.cfg
plugin_upload.py
pylintrc
README.html
README.txt
resources.qrc