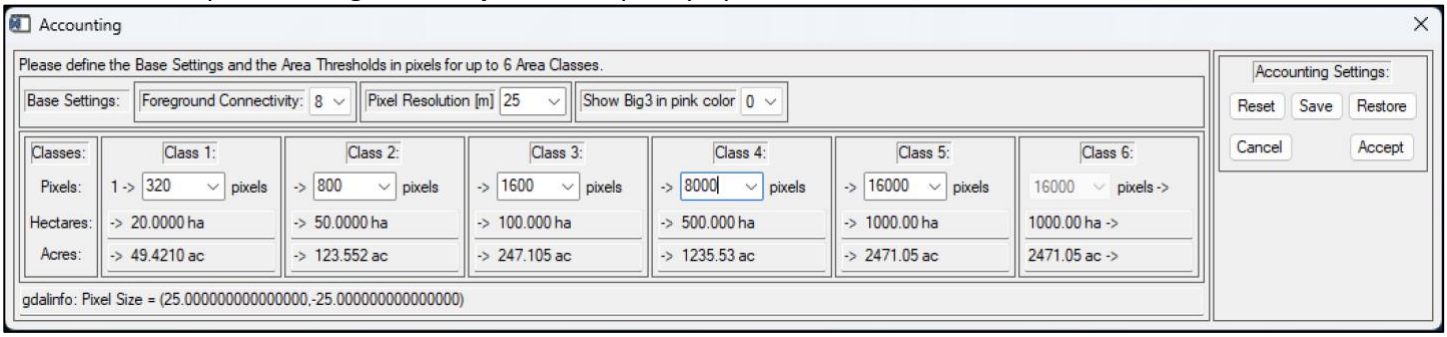
GTB-tools in container: Image Analysis → Objects
Task: summarize shape and area of image objects

Accounting (← click for product sheet)

Question: what is the size class distribution of the foreground objects (2 byte)?

How: load a map with foreground objects and specify up to six size class thresholds:



The chart above shows six user-selected thresholds in number of pixels, which are automatically converted into patch size classes up to 20, 50, 100, 500, 1000, >1000 hectare.

Result:

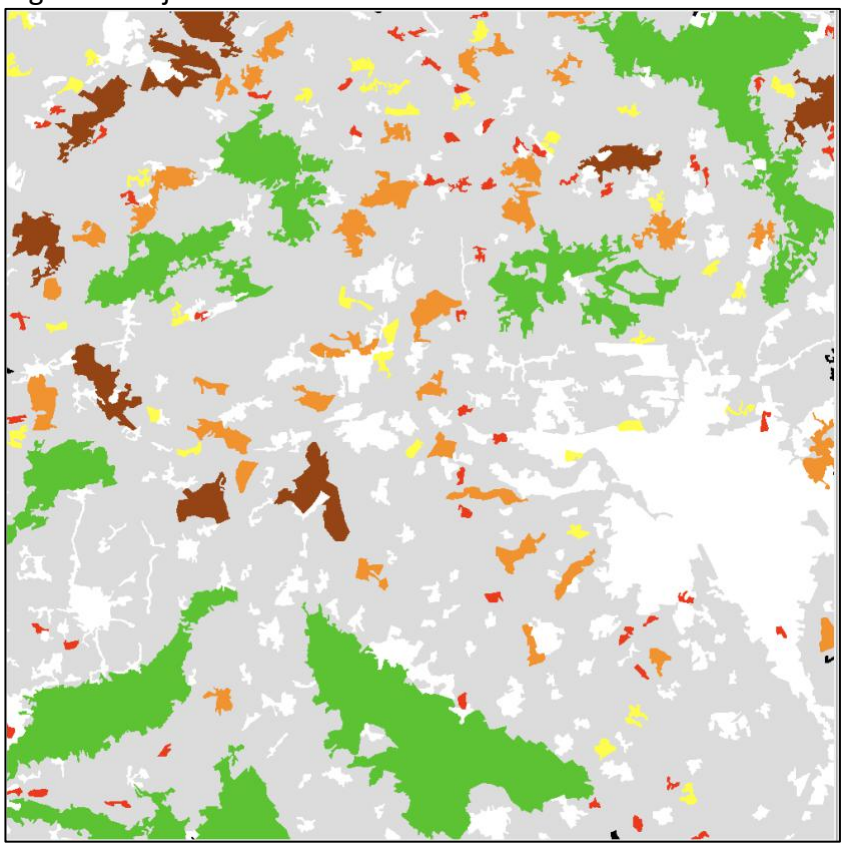
Statistics: number, area, and proportion of foreground objects in each size class; total number/area of foreground patches, average patch size and area of largest foreground object.

Map: color-coded map showing location of foreground objects in each size class.

```
Accounting size classes result using:
clc_fm
Base settings: 8-connectivity, pixel resolution: 25 [m]
Conversion factor: pixel_to_hectare: 0.0625000, pixel_to_acres: 0.154441
-------------------------------------------------------------------------
Size class 1: [1 - 320] pixels; color: black
     # Objects      Area[pixels]    % of all objects  % of total FGarea
          14             1531          8.86076        0.18566308
-------------------------------------------------------------------------
Size class 2: [321 - 800] pixels; color: red
     # Objects      Area[pixels]    % of all objects  % of total FGarea
          55            30767          34.8101        3.7310881
-------------------------------------------------------------------------
Size class 3: [801 - 1600] pixels; color: yellow
     # Objects      Area[pixels]    % of all objects  % of total FGarea
          35            40281          22.1519        4.8848428
-------------------------------------------------------------------------
Size class 4: [1601 - 8000] pixels; color: orange
     # Objects      Area[pixels]    % of all objects  % of total FGarea
          36           122369          22.7848        14.839585
-------------------------------------------------------------------------
Size class 5: [8001 - 16000] pixels; color: brown
     # Objects      Area[pixels]    % of all objects  % of total FGarea
           9           100038          5.69620        12.131524
-------------------------------------------------------------------------
Size class 6: [16001 -> ] pixels; color: green
     # Objects      Area[pixels]    % of all objects  % of total FGarea
           9           529626          5.69620        64.227297
-------------------------------------------------------------------------
-------------------------------------------------------------------------
Sum of all classes:
     # Objects      Area[pixels]    % of all objects  % of total FGarea
         158           824612          100.000        100.00000

 Median Patch Size:          907
Average Patch Size:          5219.06
Standard Deviation:          16724.0

Largest object:    146318
```



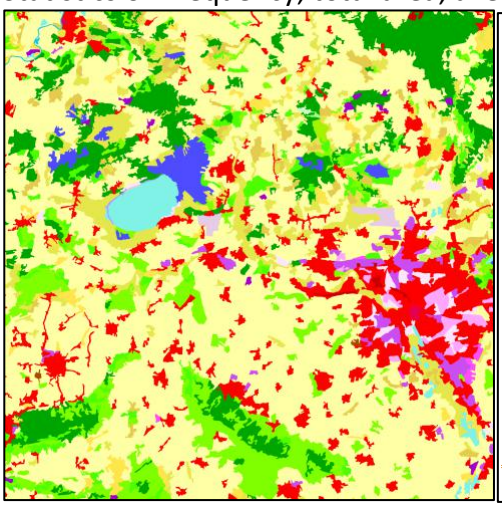
The chart above shows the statistics and the map of the six size classes color-coded in (black-103, red-33, yellow-65, orange-1, brown-9, green-17). Moving the mouse cursor in the GTB viewport will show the ID and area in pixels for each foreground object.

Parcellation (← click for product sheet)

Question: how many individual classes do we have? How many objects are in each class? What is the area coverage, average patch size and degree of parcellation in each class?

How: load a (landcover) map and run Parcellation.

Result:
Statistics on frequency, total area, average patch size, and degree of parcellation in each class.



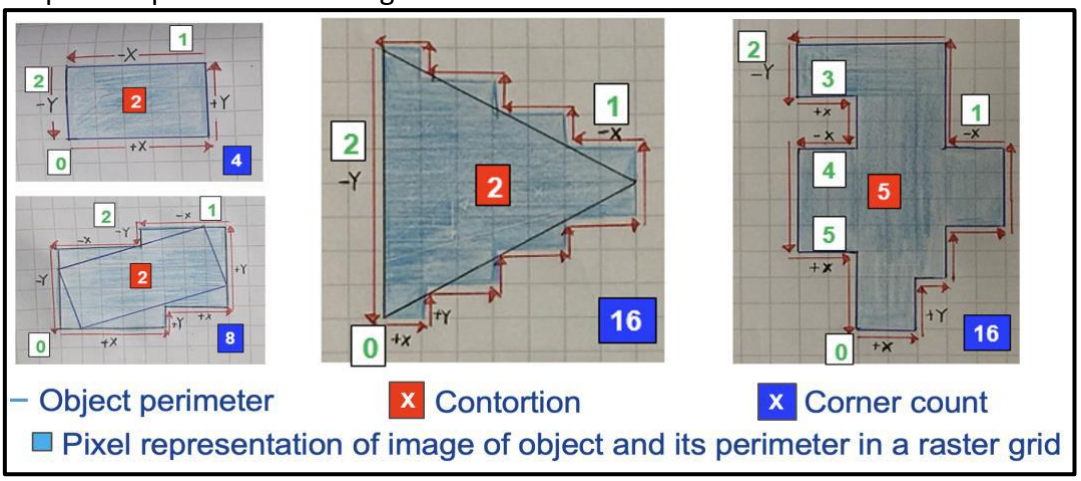
Class	Value	Count	Area[pixels]	APS	AWAPS	AWAPS/data	DIVISION	PARC[%]
1	1	2	2792	1396.0000	1998.3600	1.3948	0.2843	2.4207
2	2	201	432930	2153.8800	24287.2000	2628.6600	0.9439	20.8507
3	3	35	56446	1612.7400	3655.9400	51.5909	0.9352	19.8106
4	4	5	6131	1226.2000	1335.9000	2.0476	0.7821	11.0293
5	5	1	418	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

The map above has 26 individual classes. The table lists the map value, occurrence frequency, total area, average patch size, area-weighted average patch size, degree of division and parcellation for each class as well as for the entire image (bottom line of the table).

Contortion

Question: how complex is the shape of each foreground object (2 byte)? Can we delineate regular-shaped objects under human influence (houses, agricultural fields) from complex-shaped natural objects?

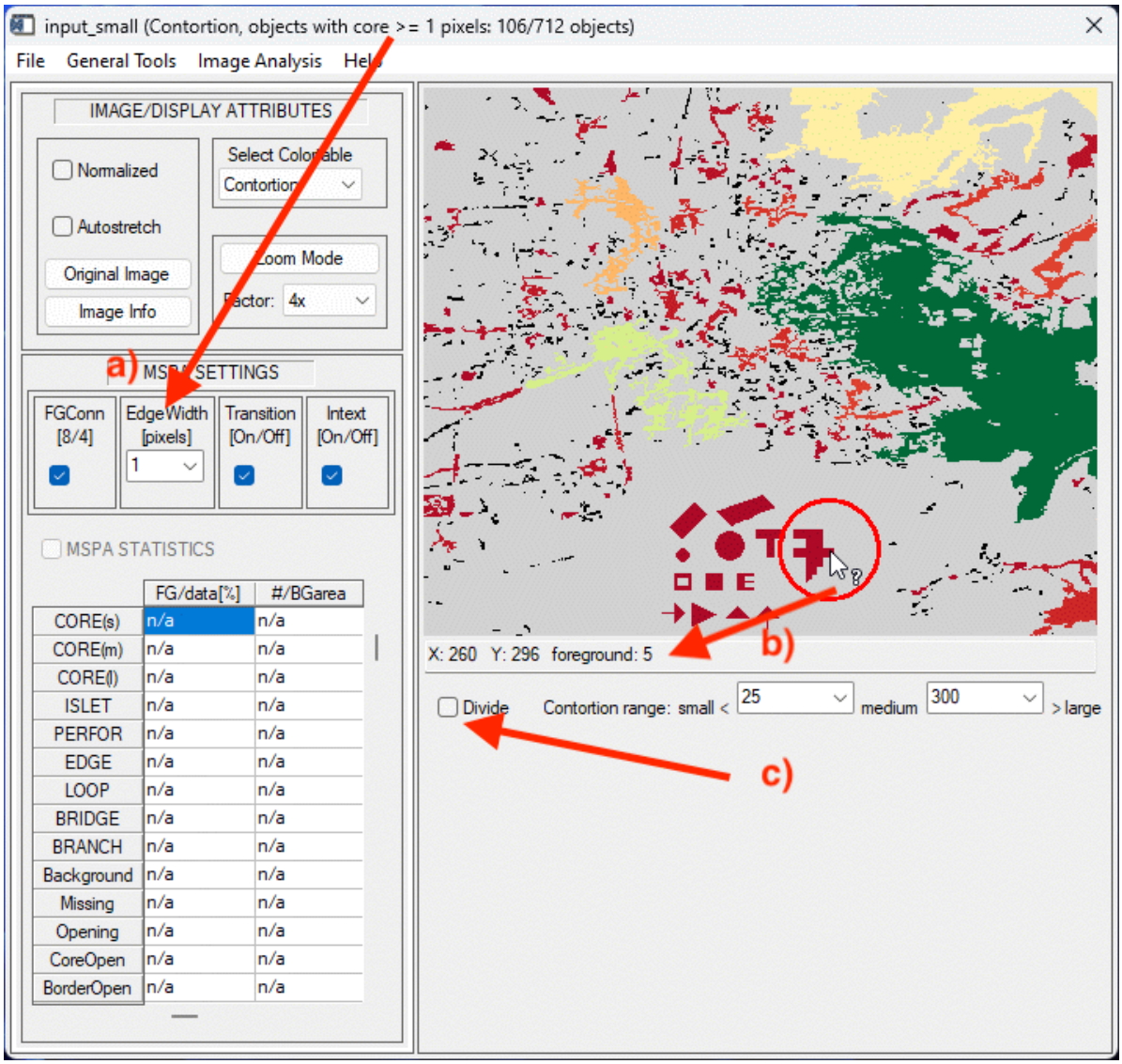
Contortion measures the complexity of each object perimeter by counting directional changes in x/y direction. Simple shaped perimeters, like a triangle, square, or circle, have a low contortion value while natural objects have a more complex shape and hence a higher contortion value.



The chart above illustrates the concept of counting directional changes in x/y direction. In contrast to corner counting, Contortion provides constant results when rotating or increasing the size of the object.

How: load a map with foreground objects (2 byte) and run Contortion.

Result:
A map showing the degree in contortion (perimeter complexity) for each foreground object with a core area. Note that small objects without core area, like lines and isolated pixels, do not need to be analyzed because they always have the minimum contortion value of 2. They are marked in black color and excluded to speed up the analysis.



The chart above shows contortion for objects with varying degree in shape complexity.

a) Use the MSPA parameter 2 *EdgeWidth* drop-down menu to constrain the analysis to objects equal or larger to a minimum core area. The tile bar shows how many objects of the total number of objects will be analyzed, here 106 out of 712. Smaller, excluded objects are marked in black color.
b) Move the mouse cursor over an object of interest and read out the contortion value below the viewport, for example the contortion value of the object indicated is 5.
c) Activate the *Divide* switch to group the objects in a contortion range of small, medium, and large within the user-selected thresholds.