

# EnMABox 3

## Bridging Imaging Spectroscopy and GIS with Free and Open Source Software

Sebastian van der Linden<sup>1</sup>  
Andreas Janz<sup>2</sup>  
Benjamin Jakimow<sup>2</sup>  
Fabian Thiel<sup>1</sup>  
Akpona Okujeni<sup>2</sup>  
Patrick Hostert<sup>2</sup>  
<sup>1</sup>Universität Greifswald  
<sup>2</sup>Humboldt-Universität zu Berlin



Supported by:



on the basis of a decision  
by the German Bundestag

The slide collection is provided free of charge under a Creative Commons Attribution-ShareAlike 4.0 International License [CC BY-SA 4.0](https://creativecommons.org/licenses/by-sa/4.0/)

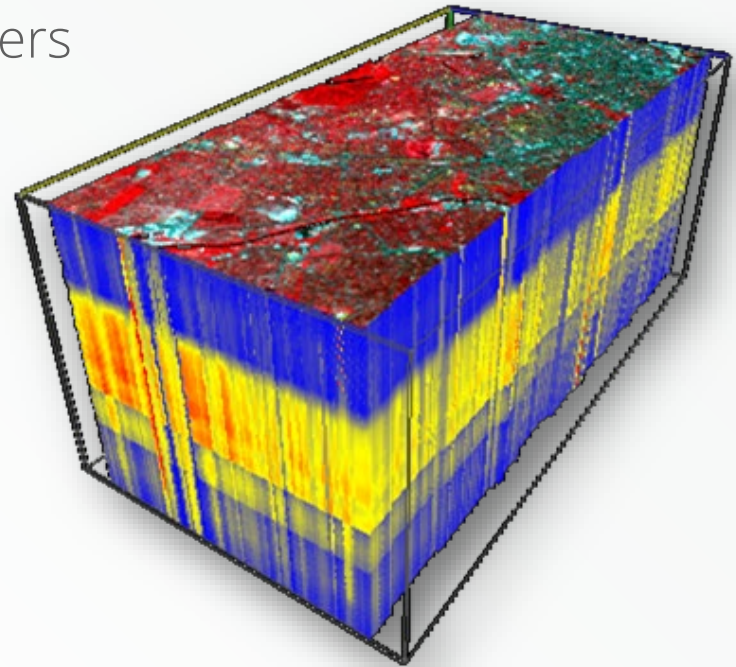


UNIVERSITÄT GREIFSWALD  
Wissen lockt. Seit 1456



# EnMABox – Motivation and Aims

- ❖ Offer a free and open source environment for visualizing and analyzing EnMAP data
- ❖ Increase the number of EnMAP data users
- ❖ Integrate full GIS functionality with advanced image/spectral processing
- ❖ Suite of application-oriented advanced Workflows (Vegetation, Geology)
- ❖ Foster the availability and exchange of state-of-the-art approaches for the analysis of imaging spectroscopy data and spectral libraries





# EnMAP-Box – Imaging Spectroscopy for QGIS

## EnMAP-Box GUI and Algorithm Provider



fid	name
62	65 sand (playground) 2
63	45 sorghum
64	66 sugarcane 1
65	47 sugarcane 2
66	43 sunflower
67	73 tarran (sports ground)
68	75 water 2
69	74 water1
70	16 white roof material (...)
71	17 white roof material (...)
72	18 white roof material (...)
73	19 white roof material (...)

- Vector creation
- Vector general
- Vector geometry
- Vector overlay
- Vector selection
- Vector table
- Vector tiles
- EnMAP-Box**
  - Accuracy Assessment
  - Auxiliary
  - Classification
  - Clustering
  - Convolution, Morphology and Filtering
  - Create Raster
  - Create Sample
  - Import Data
  - Masking
  - Post-Processing
  - Random
  - Regression
  - Resampling and Subsetting
  - Testdata
  - Transformation

- ▶ EnMAP-Box
  - ▶ Accuracy Assessment
  - ▶ Auxiliary
  - ▶ Classification
  - ▶ Clustering
  - ▶ Convolution, Morphology and Filtering
  - ▶ Create Raster
  - ▶ Create Sample
  - ▶ Import Data
  - ▶ Masking
  - ▶ Post-Processing
  - ▶ Random
  - ▶ Regression
  - ▶ Resampling and Subsetting
  - ▶ Testdata
  - ▶ Transformation



# EnMABox – Create Sensor Synergies

❖ Comfortably import multiple sensors

The screenshot displays the EnMAP-Box 3 (3.8) software interface. The main window shows a menu bar with 'Project', 'View', 'Tools', 'Applications', and 'Help'. The 'Add Data Source' menu is open, showing options like 'Add Product', 'Load Example Data', and 'Package Installer'. The 'Add Product' sub-menu is expanded, listing various data sources including 'Sentinel-2 L2A'. The 'Import Sentinel-2 L2A product' dialog box is the central focus, showing a 'Band list' with 12 items, each with a checkbox and resolution information. The 'Parameters' tab is active, and the 'Log' tab is also visible. The dialog box includes instructions for preparing a spectral raster layer, metadata file, and output raster layer. The 'Run' button is highlighted in blue.

**Import Sentinel-2 L2A product**

Prepare a [spectral raster layer](#) from the given product. [Wavelength](#) information is set and data is scaled into the 0 to 10000 range.

**Metadata file**

The MTD\_MSIL2A.xml metadata file associated with the product.

**Band list**

Bands to be stacked together. Defaults to all 10m and 20m bands ordered by [center wavelength](#). Note that the destination pixel size matches the smallest/finest pixel size over all selected bands.

**Output raster layer**

Raster file destination.

0% Cancel

Run Close Help

Run as Batch Process...



# EnMA-Box – Create Sensor Synergies

## Processing Toolbox



PRISMA

### EnMAP-Box

#### \*Import data

- Import PRISMA L1 product
- Import PRISMA L2B product
- Import PRISMA L2C product
- Import PRISMA L2D product

#### \*Spectral resampling

- Spectral resampling (to PRISMA)

The screenshot displays the QGIS Data Views panel on the left and a grid of 8 map thumbnails on the right.

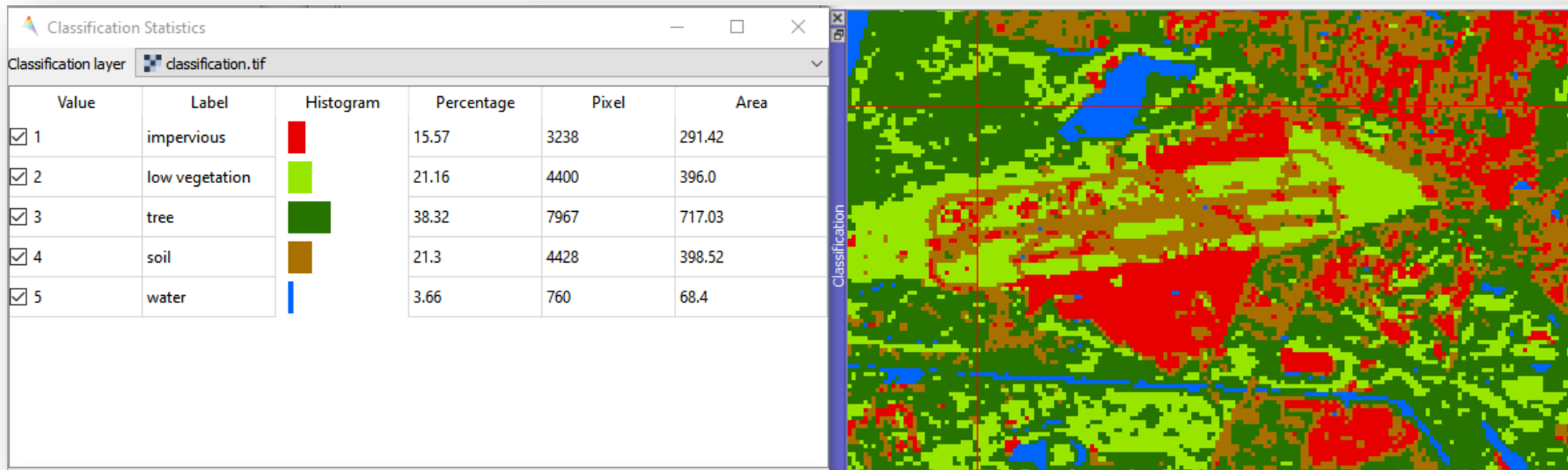
**Data Views Panel:**

- Filter:** (Empty)
- Property:**
  - Map #1
    - prismaL2D\_SPECTRAL.tif
      - Band 001: Band 1 (402.49423 Nanometers) (Gray)
      - Band 002: Band 2 (411.36902 Nanometers)
      - Band 003: Band 3 (419.42004 Nanometers)
  - Map #2
    - prismaL2D\_PAN.vrt
      - Band 1: Panchromatic
      - 36.854
      - 13.762
  - Map #3
    - prismaL2D\_SPECTRAL\_GEOLOCATION.vrt
      - Band 1: Longitude
      - Band 2: Latitude
  - Map #4
    - prismaL2D\_SPECTRAL\_GEOMETRIC.vrt
      - Band 1: Observing Angle
      - Band 2: Relative Azimuth Angle
      - Band 3: Solar Zenith Angle
  - Map #5
    - prismaL2D\_SPECTRAL\_ERROR.tif
      - Band 001: Pixel Error Band 1 (402.49423 Nanometers) (Gray)
      - 1
      - 0
  - Map #6
    - prismaL2D\_PAN\_GEOLOCATION.vrt
      - Band 1: Longitude
      - Band 2: Latitude
  - Map #7
    - prismaL2D\_PAN\_ERROR.vrt
      - Band 1: PAN Band Pixel Error
      - 1
      - 0
  - Map #8

**Map Grid:**

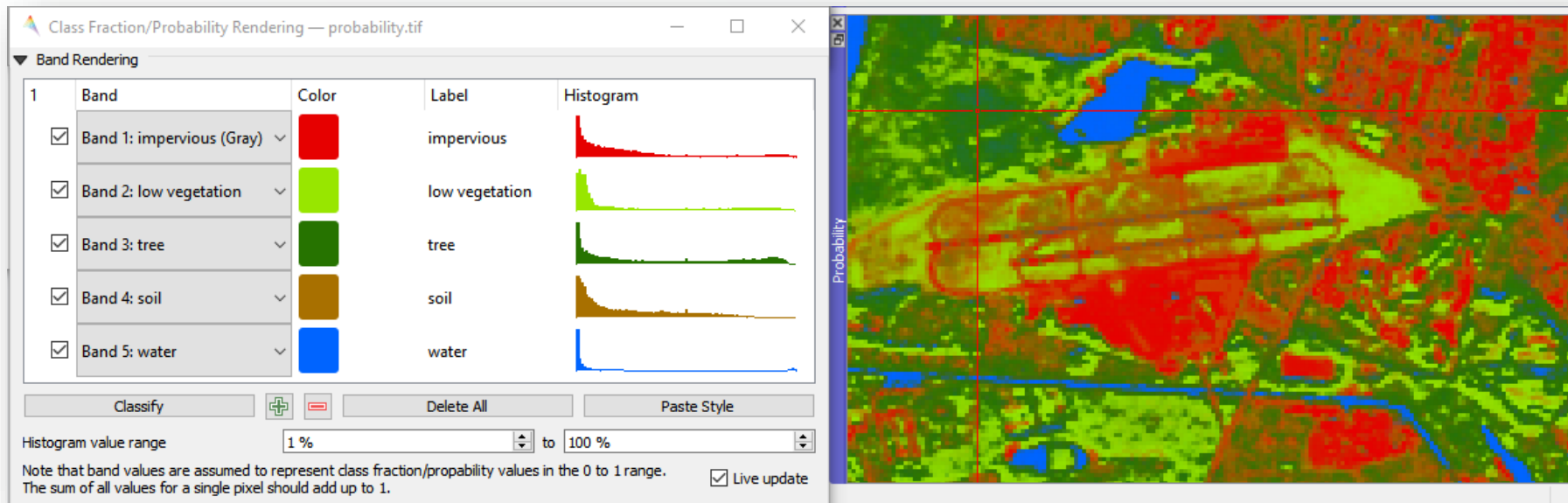
- Map #1: Original PRISMA L2D Spectral data (3 bands).
- Map #2: Panchromatic (PAN) data.
- Map #3: Spectral data with geographic location (Longitude/Latitude).
- Map #4: Spectral data with geometric information (Observing Angle, Relative Azimuth Angle, Solar Zenith Angle).
- Map #5: Spectral data with pixel error for Band 1.
- Map #6: Panchromatic (PAN) data with geographic location.
- Map #7: Panchromatic (PAN) data with pixel error.
- Map #8: A color-coded visualization, likely a composite of the spectral bands.

- ❖ General tools with increased functionality



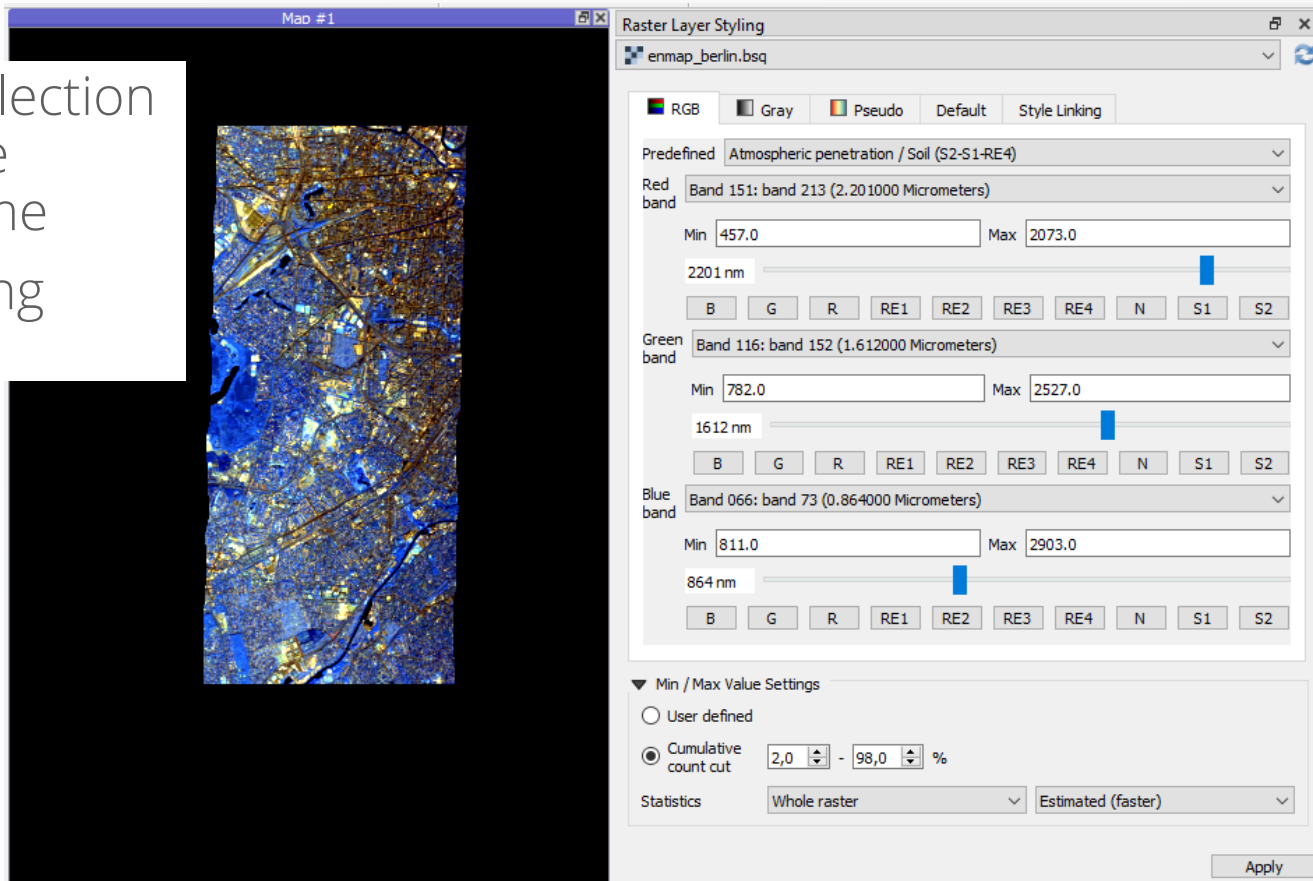
# EnMABox – Class Fractions/Quant. Variables

- ❖ General tools optimized for quantitative results from EnMAP products



# EnMABox – Raster Layer Styling

- ❖ Easy band selection along favorite sensor scheme
- ❖ Spectral linking



Map #1

Raster Layer Styling

enmap\_berlin.bsq

RGB Gray Pseudo Default Style Linking

Predefined Atmospheric penetration / Soil (S2-S1-RE4)

Red band Band 151: band 213 (2.201000 Micrometers)

Min 457.0 Max 2073.0

2201 nm

B G R RE1 RE2 RE3 RE4 N S1 S2

Green band Band 116: band 152 (1.612000 Micrometers)

Min 782.0 Max 2527.0

1612 nm

B G R RE1 RE2 RE3 RE4 N S1 S2

Blue band Band 066: band 73 (0.864000 Micrometers)

Min 811.0 Max 2903.0

864 nm

B G R RE1 RE2 RE3 RE4 N S1 S2

Min / Max Value Settings

User defined

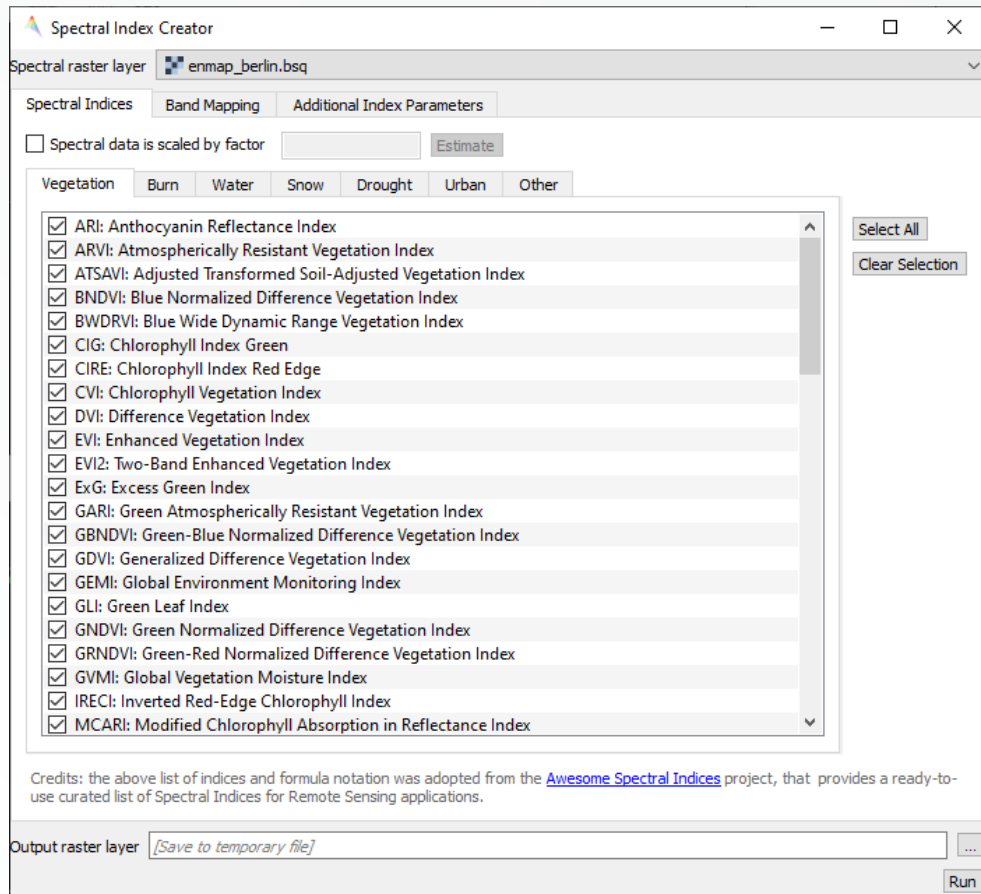
Cumulative count cut 2,0 - 98,0 %

Statistics Whole raster Estimated (faster)

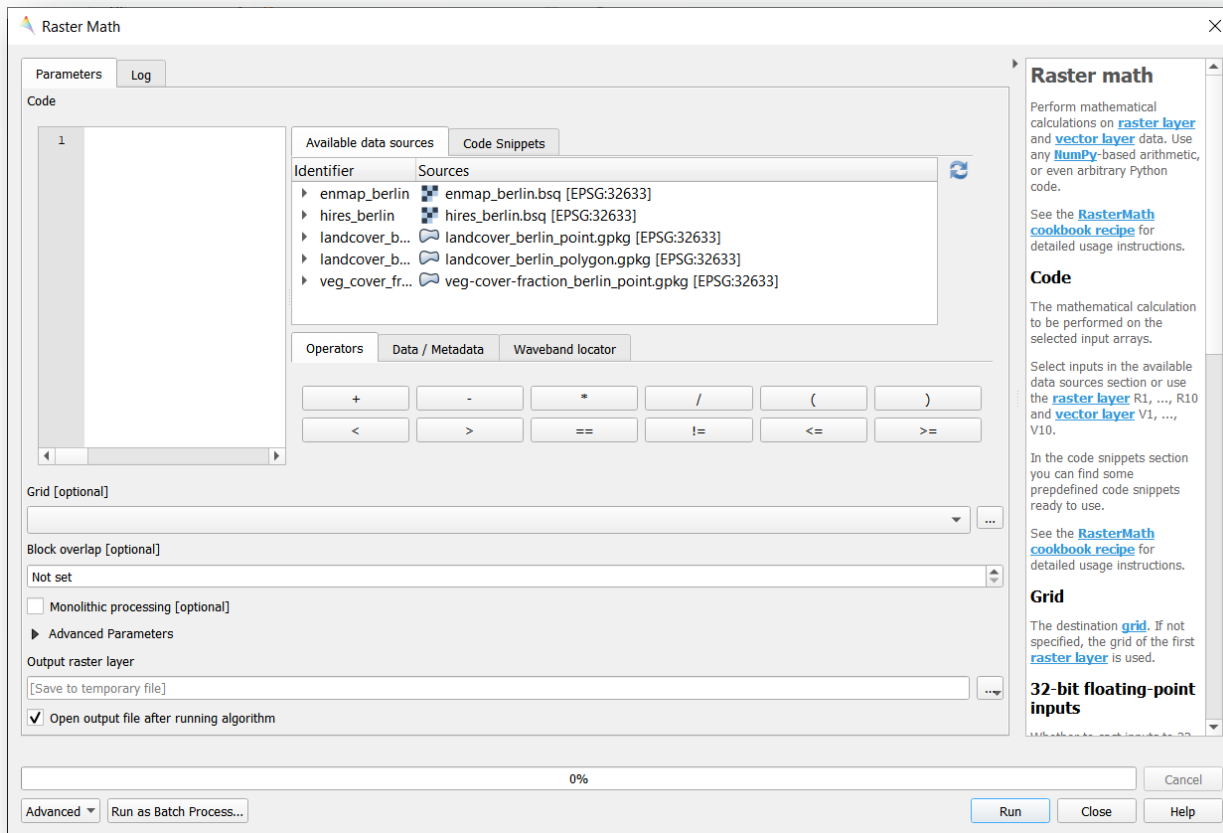
Apply



- ❖ Extensive index list taken from *AwesomeSpectral Indices* project.



❖ Easy numpy scripting with data IO fully handled



The screenshot shows the 'Raster Math' application window. It features a 'Parameters' tab and a 'Log' tab. The main area is divided into several sections:

- Code:** A text area containing the number '1'.
- Available data sources:** A table listing various data sources with their identifiers and sources.
 

Identifier	Sources
▶ enmap_berlin	enmap_berlin.bsq [EPSG:32633]
▶ hires_berlin	hires_berlin.bsq [EPSG:32633]
▶ landcover_b...	landcover_berlin_point.gpkg [EPSG:32633]
▶ landcover_b...	landcover_berlin_polygon.gpkg [EPSG:32633]
▶ veg_cover_fr...	veg-cover-fraction_berlin_point.gpkg [EPSG:32633]
- Operators:** A set of buttons for mathematical operations: +, -, \*, /, (, ), <, >, ==, !=, <=, >=.
- Grid [optional]:** A dropdown menu.
- Block overlap [optional]:** A dropdown menu set to 'Not set'.
- Monolithic processing [optional]:** A checkbox that is currently unchecked.
- Advanced Parameters:** A section with a dropdown menu.
- Output raster layer:** A dropdown menu set to '[Save to temporary file]'.
- Open output file after running algorithm:** A checked checkbox.

At the bottom, there is a progress bar showing 0%, a 'Run as Batch Process...' button, and 'Run', 'Close', and 'Help' buttons.

**Raster math**

Perform mathematical calculations on [raster layer](#) and [vector layer](#) data. Use any [NumPy](#)-based arithmetic, or even arbitrary Python code.

See the [RasterMath cookbook recipe](#) for detailed usage instructions.

**Code**

The mathematical calculation to be performed on the selected input arrays.

Select inputs in the available data sources section or use the [raster layer](#) R1, ..., R10 and [vector layer](#) V1, ..., V10.

In the code snippets section you can find some predefined code snippets ready to use.

See the [RasterMath cookbook recipe](#) for detailed usage instructions.

**Grid**

The destination [grid](#). If not specified, the grid of the first [raster layer](#) is used.

**32-bit floating-point inputs**

# EnMAP-Box – Advanced Spectral Library Tools

Project View Tools Applications Help  
 Data Sources  
 Filter  
 Source Value  
 > Raster Data (2)  
 > Vector Data (5)  
 > enmap\_srf\_library.gpkg  
 > landcover\_berlin\_point.gpkg  
 Spectral Profile Sources  
 Name Value  
 > SpecLib  
 > Point  
 > srf name 'cropland'  
 > srf class 'vegetation'  
 > EnMAP  
 > Color #f2bd13  
 > Source enmap\_berlin.bi  
 > Sampling Single Profile  
 > Scaling  
 > Offset 0.0 + 1.0 \* y  
 > 0.0  
 > 1.0  
 > Scale  
 > EnMAP3x3Mean enmap\_berlin.bi  
 > Color #f75a07c  
 > Source enmap\_berlin.bi  
 > Sampling Kernel  
 > Aggregation Mean  
 move pixel : CTRL + <Arrow>  
 Save profile: CTRL + S  
 show / hide  
 Plot Plot Settings Form View Table  
 multiple profile visualizations  
 X Axis [Wavelength [nm]]  
 Y Axis [Radiance]  
 Name Value  
 Model  
 > Center Pixel 3x3Mean  
 > Field EnMAP3x3Mean  
 > Label "name"-' 3x3 mean'  
 > Color #fb9a99  
 > Style  
 > Filter "class"='vegetation'  

name	class	EnMAP	EnMAP3x3Mea
1 forest	Profile	BLOB	Profile
2 forest	Profile	BLOB	Profile
3 forest	Profile	BLOB	Profile
4 cropland	Profile	BLOB	Profile
5 cropland	Profile	BLOB	Profile
6 cropland	Profile	BLOB	Profile

Snap to pixel center  
 search settings  
 define values for each field  
 scale profiles  
 overlay color  
 sampling  
 Enhanced Profile Source Panel  
 Enhanced Plot Settings  
 Individual / generated labels and colors  
 filter profile by attributes  
 multiple profile fields

# GEE Time Series Explorer



The screenshot displays the GEE Time Series Explorer interface. At the top left, the 'GEE Temporal Profile Viewer' window shows a graph of a time series for a specific location (14.26526, 53.46563). The x-axis represents time from 2019 to 2021, and the y-axis represents a value from 0.1 to 0.4. A blue shaded region highlights the period from 2020-04-03 to 2021-01-06. Below the graph is a map showing the geographic location. The main interface includes a legend with various indices such as SR\_B1 through SR\_B7, ST\_AEROSOL, ST\_ATRAN, ST\_CDIST, ST\_DRAD, ST\_EMIS, ST\_EMSD, ST\_TRAD, ST\_URAD, QA\_PIXEL, QA\_RADSAT, ARVI, EVI, NDMI, SARVI, SAVI, NBR, MNDWI, NDWI, NDBI, and NDSI. On the right, there are 'Pixel Quality filter' and 'Cirrus Confidence' panels with checkboxes for different filter options. At the bottom, there are checkboxes for 'GEE Image', 'ID', 'Date', and 'Bands'.

Presentation on  
**GEE Time Series Explorer**  
 by Andreas Janz  
 Thursday, 26th, 2:00 pm  
 Santiago de Chile

## EnMAP Cookbook

### ▢ HUB Datacube Cookbook

#### ▢ General

#### ▢ Raster dataset

Open a raster dataset from file

Open a raster dataset from GDAL dataset

Close a raster dataset

Get raster metadata

Set raster metadata

Get and set no data value

Get raster band

Read raster data

Write raster data

Loop through all raster bands

Convert a vector to a raster

Clip a raster with a vector

Calculate zonal statistics

Create raster from array

Create memory raster

Replace no data value of raster with new value

#### ▢ Raster band dataset

VectorData

#### ▢ Grid

#### ▢ Extent

#### ▢ RasterDriver

#### ▢ Projection

#### ▢ MapViewer

## HUB Workflow Cookbook

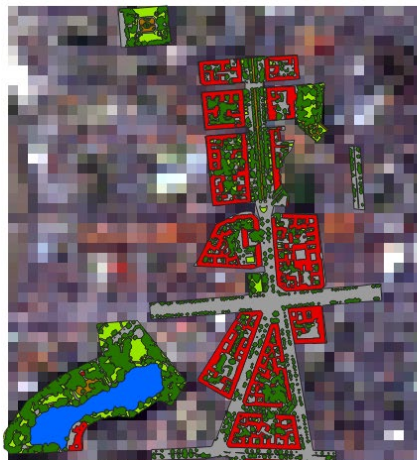
### Clip a raster with a vector

Clip a raster with the extent from a vector.

```
import enmapboxtestdata
from hubdc.core import *

rasterDataset = openRasterDataset(filename=enmapboxtestdata.enmap)
vectorDataset = openVectorDataset(filename=enmapboxtestdata.landcover_polygons)
grid = rasterDataset.grid().clip(extent=vectorDataset.extent())
clipped = rasterDataset.translate(grid=grid, filename='raster.tif', driver=GTiffDriver())
```

Note that the result raster grid is snapped to the original raster grid to prevent subpixel shifts. Because of this, some vector geometries may slightly lap over the grid borders.





# EnMABox – Conclusions

- ❖ Free and open-source EO toolbox with extensive GIS functionality
- ❖ Good solution for processing imaging spectroscopy data from EnMAP, PRISMA or other imaging spectrometers
- ❖ GUI-based functionality for many advanced imaging spectroscopy applications
- ❖ New approach for working with spectral libraries
- ❖ Good solution also for the work with multi-spectral data
- ❖ Good example how FOSS plugins for QGIS offer excellent possibilities for EO data analysis



# Contact

- ❖ For detailed information, installation, application tutorials have a look at <https://enmap-box.readthedocs.io/en/latest/>
- ❖ Or write us: [enmapbox@enmap.org](mailto:enmapbox@enmap.org)

