geo engine

Geo Engine: Ein dynamischer Data Cube für interaktive Analysen und Verarbeitungs-Pipelines

Dr. Johannes Drönner

Philipps Universität Marburg

Gefördert durch:

Bundesministerium für Wirtschaft und Klimaschutz

aufgrund eines Beschlusses des Deutschen Bundestages

eXIST Existenzgründungen aus der Wissenschaft

ESF Europäischer Sozialfonds für Deutschland

EUROPÄISCHE UNION

Zusammen. Zukunft. Gestalten.

About Geo Engine GmbH

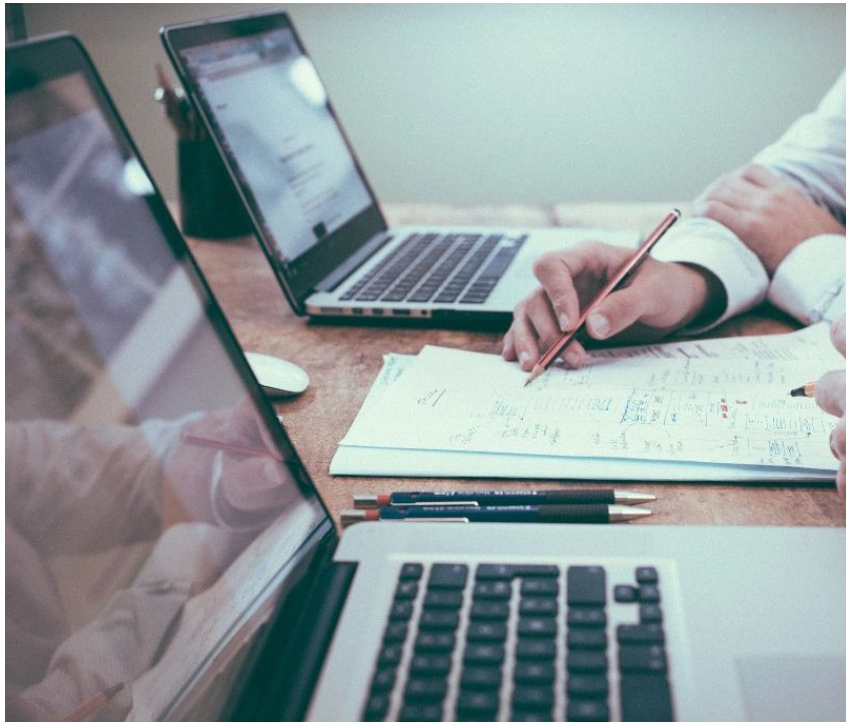
www.geoengine.de

Background

- Start-up of the University of Marburg, Germany
- EXIST Research Transfer funding
- Research in computer science, (web) GIS, biodiversity and remote sensing

Services

- Development of data portals
- Data access via browser, OGC-APIs and Jupyter Notebooks
- Data products as SaaS



Team

Christian Beilschmidt

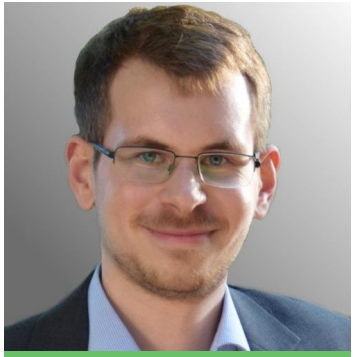
Dr. rer. nat. computer science

Software Development, Machine Learning, Visual Analytics

Johannes Drönner

Dr. rer. nat. computer science
M. Sc. Geography

Product Development, Use Cases, Deep learning



Michael Mattig

M. Sc. Computer Science

Data Integration, Processes, Infrastructure

Philip Schweitzer

M. Sc. Business Administration

Business Development, Finance, Sales

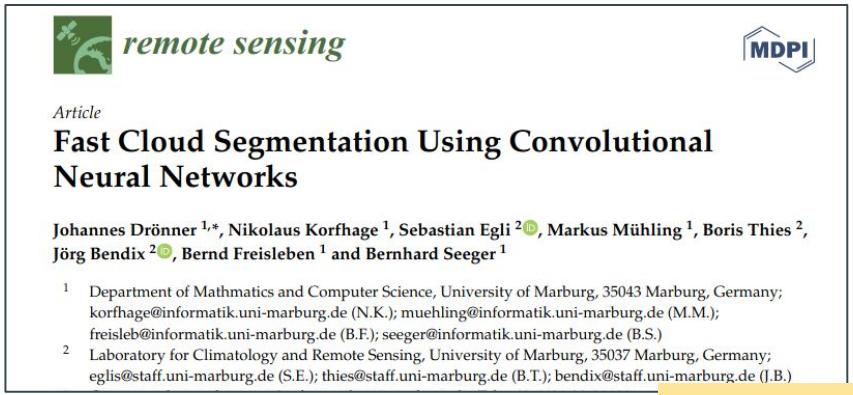
Bernhard Seeger

Prof. Computer Science

Technical Mentor, Outreach

Research background

Remote sensing / deep learning



remote sensing

MDPI

Article

Fast Cloud Segmentation Using Convolutional Neural Networks

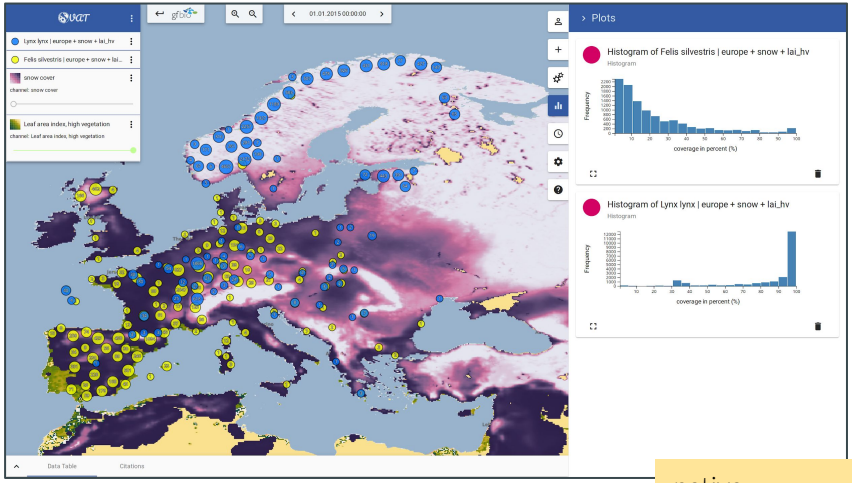
Johannes Drönner [1,*], Nikolaus Korfhage [1], Sebastian Egli [2], Markus Mühling [1], Boris Thies [2], Jörg Bendix [2], Bernd Freisleben [1] and Bernhard Seeger [1]

1 Department of Mathmatics and Computer Science, University of Marburg, 35043 Marburg, Germany; korfhage@informatik.uni-marburg.de (N.K.); muehling@informatik.uni-marburg.de (M.M.); freisleb@informatik.uni-marburg.de (B.F.); seeger@informatik.uni-marburg.de (B.S.)

2 Laboratory for Climatology and Remote Sensing, University of Marburg, 35037 Marburg, Germany; eglis@staff.uni-marburg.de (S.E.); thies@staff.uni-marburg.de (B.T.); bendix@staff.uni-marburg.de (J.B.)

on >200,000 MSG scenes

(Web) GIS / visualization / biodiversity

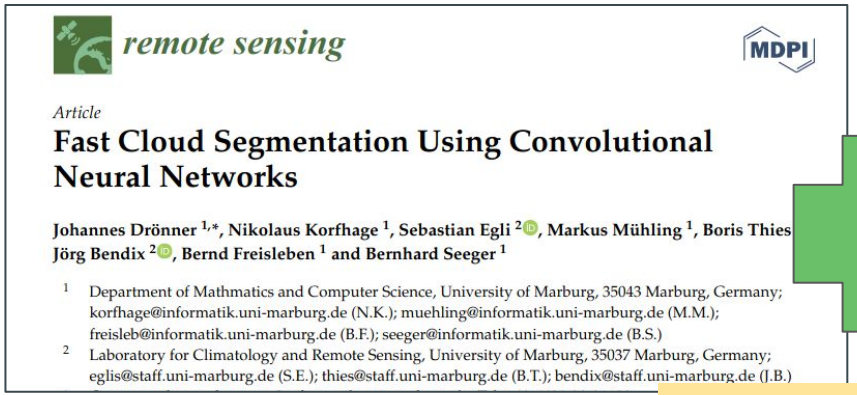


native time-series

Research background

Remote sensing / deep learning

(Web) GIS / visualization / biodiversity



remote sensing

MDPI

Article

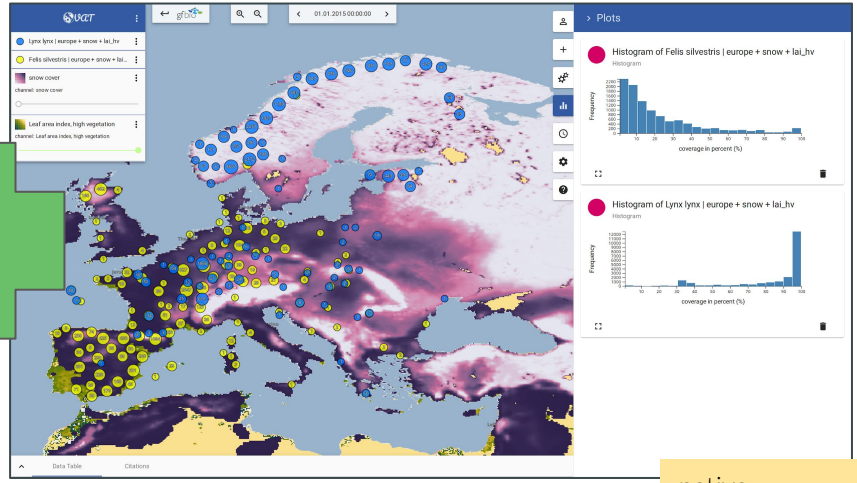
Fast Cloud Segmentation Using Convolutional Neural Networks

Johannes Drönner [1,*], Nikolaus Korfhage [1], Sebastian Egli [2], Markus Mühling [1], Boris Thies Jörg Bendix [2], Bernd Freisleben [1] and Bernhard Seeger [1]

1 Department of Mathmatics and Computer Science, University of Marburg, 35043 Marburg, Germany; korfhage@informatik.uni-marburg.de (N.K.); muehling@informatik.uni-marburg.de (M.M.); freisleb@informatik.uni-marburg.de (B.F.); seeger@informatik.uni-marburg.de (B.S.)

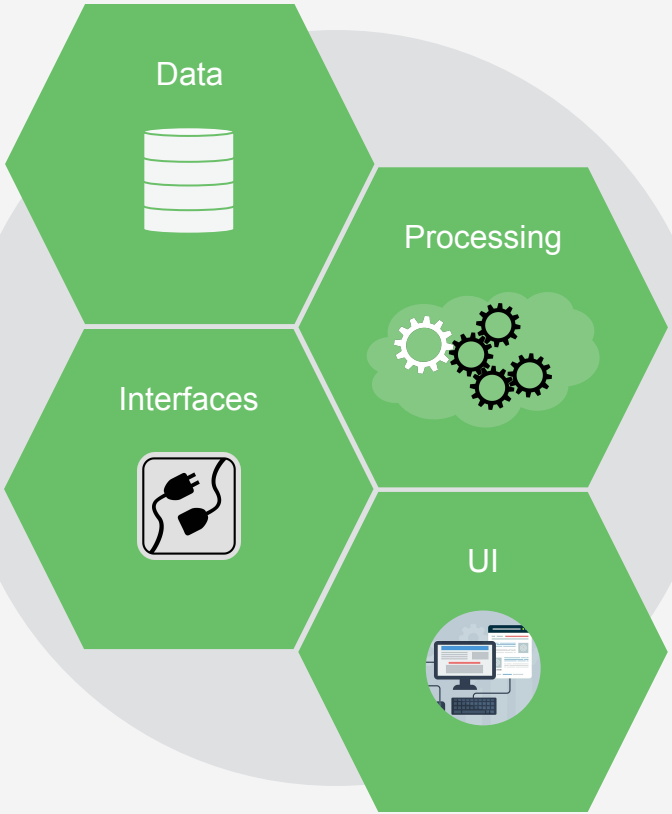
2 Laboratory for Climatology and Remote Sensing, University of Marburg, 35037 Marburg, Germany; eglis@staff.uni-marburg.de (S.E.); thies@staff.uni-marburg.de (B.T.); bendix@staff.uni-marburg.de (J.B.)

on >200,000 MSG scenes

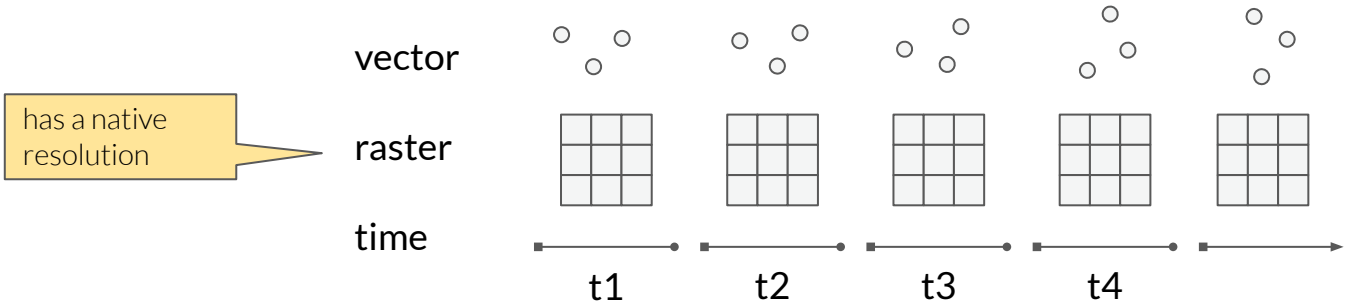


native time-series

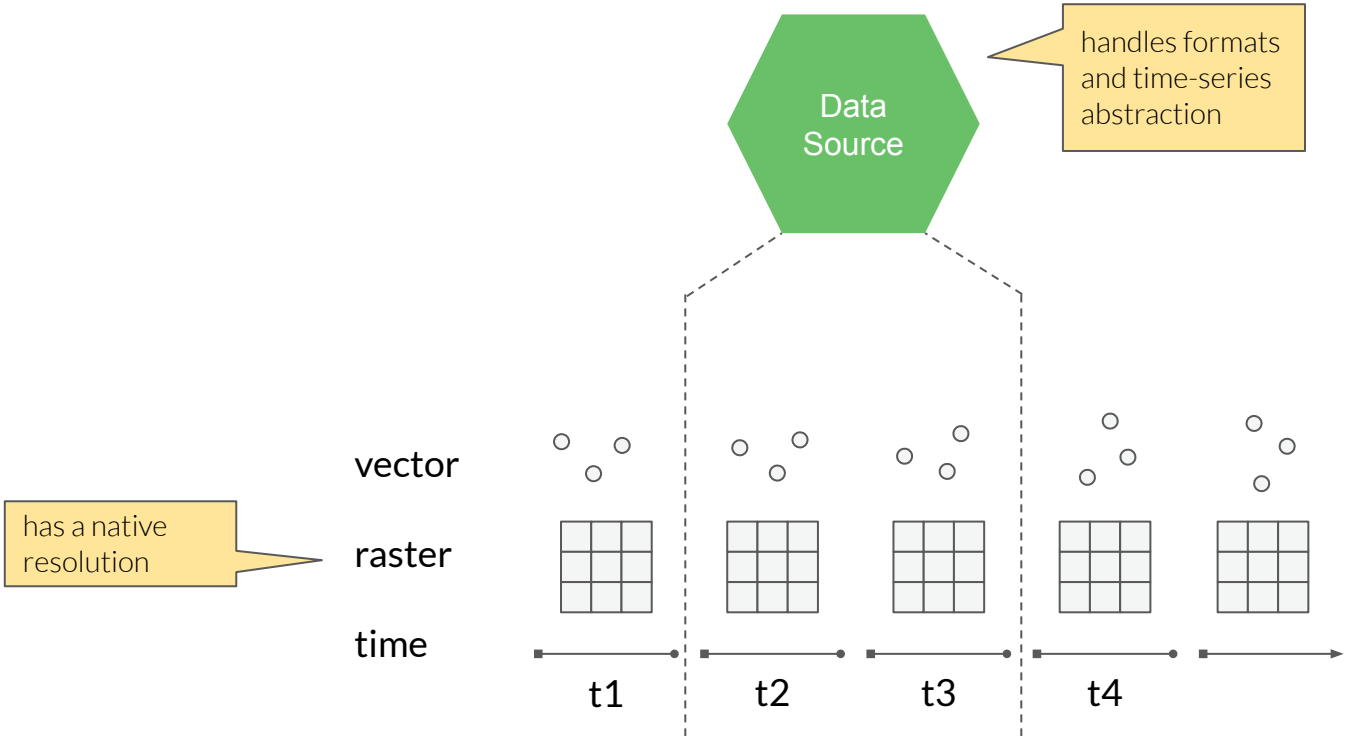
geo engine



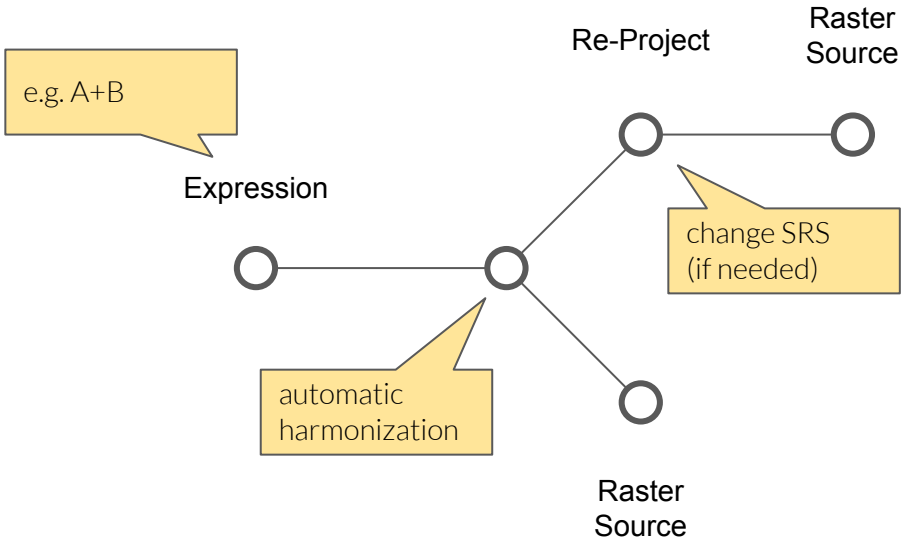
Concept 1: All Datasets are Time-Series



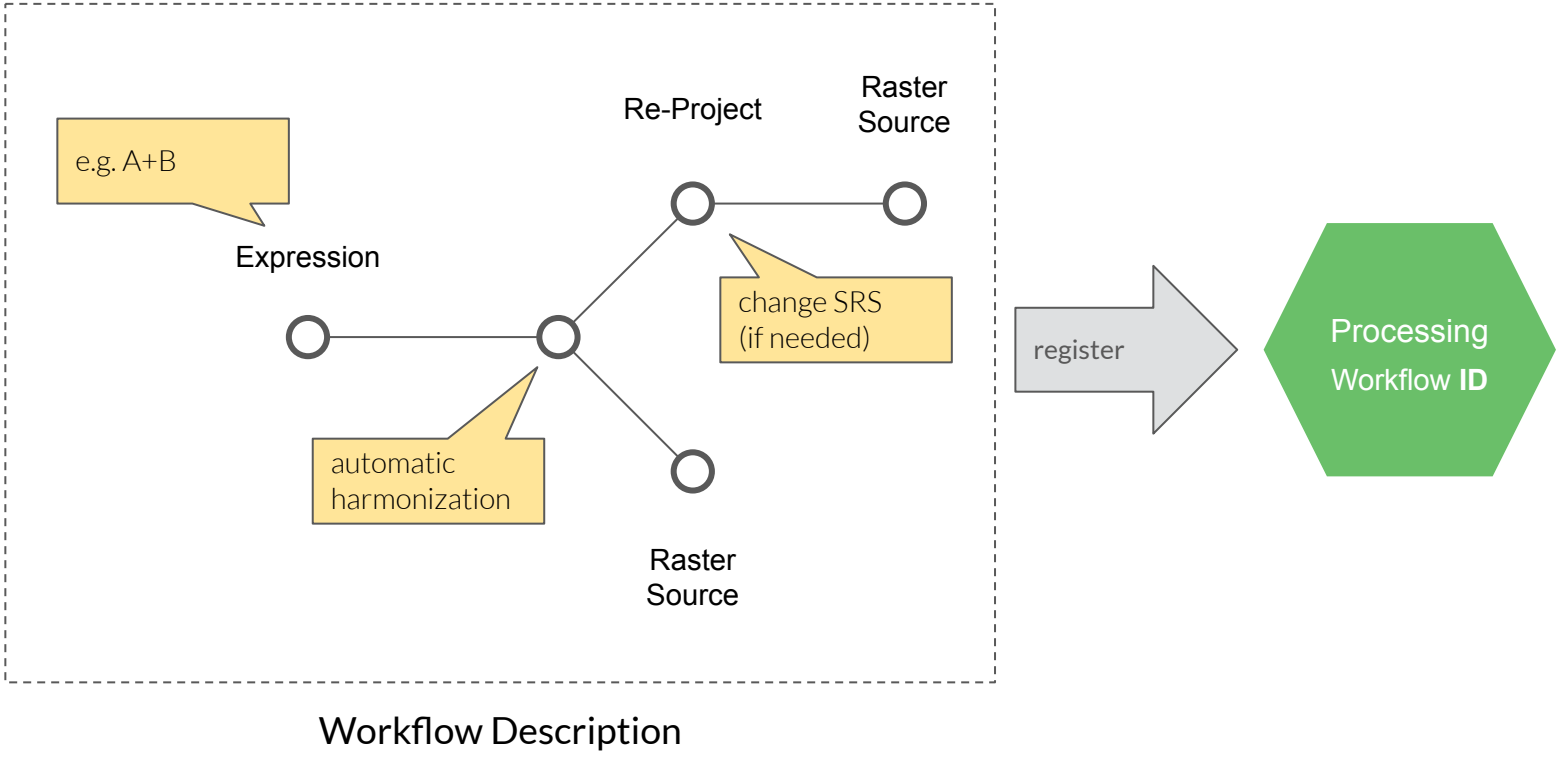
Concept 1: All Datasets are Time-Series



Concept 2: Workflows

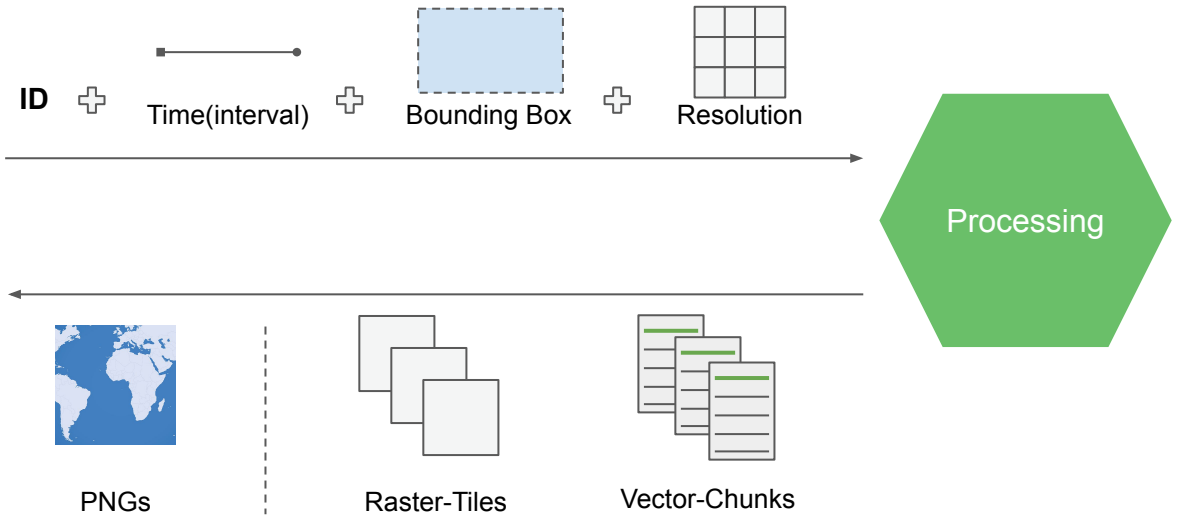


Concept 2: Workflows

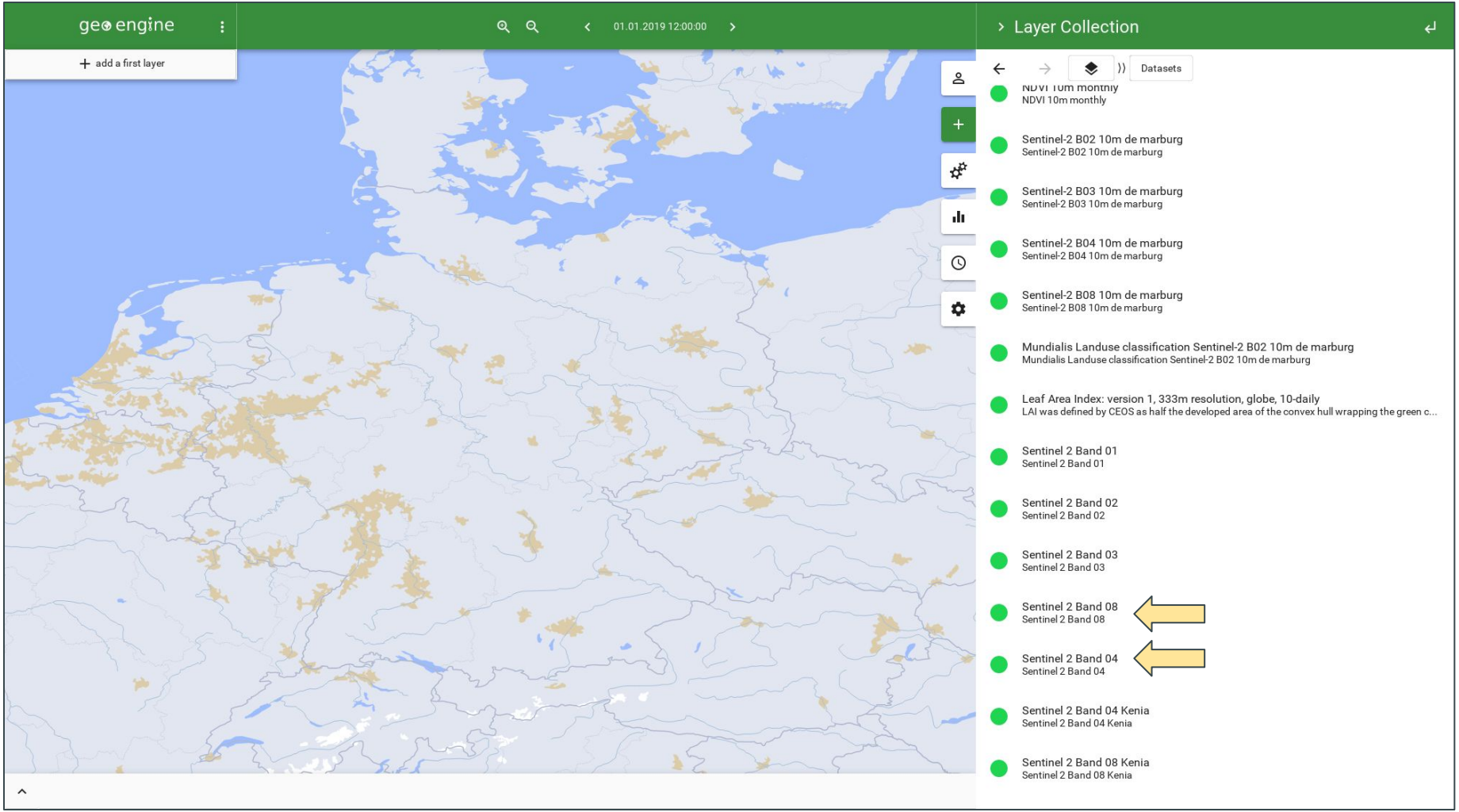


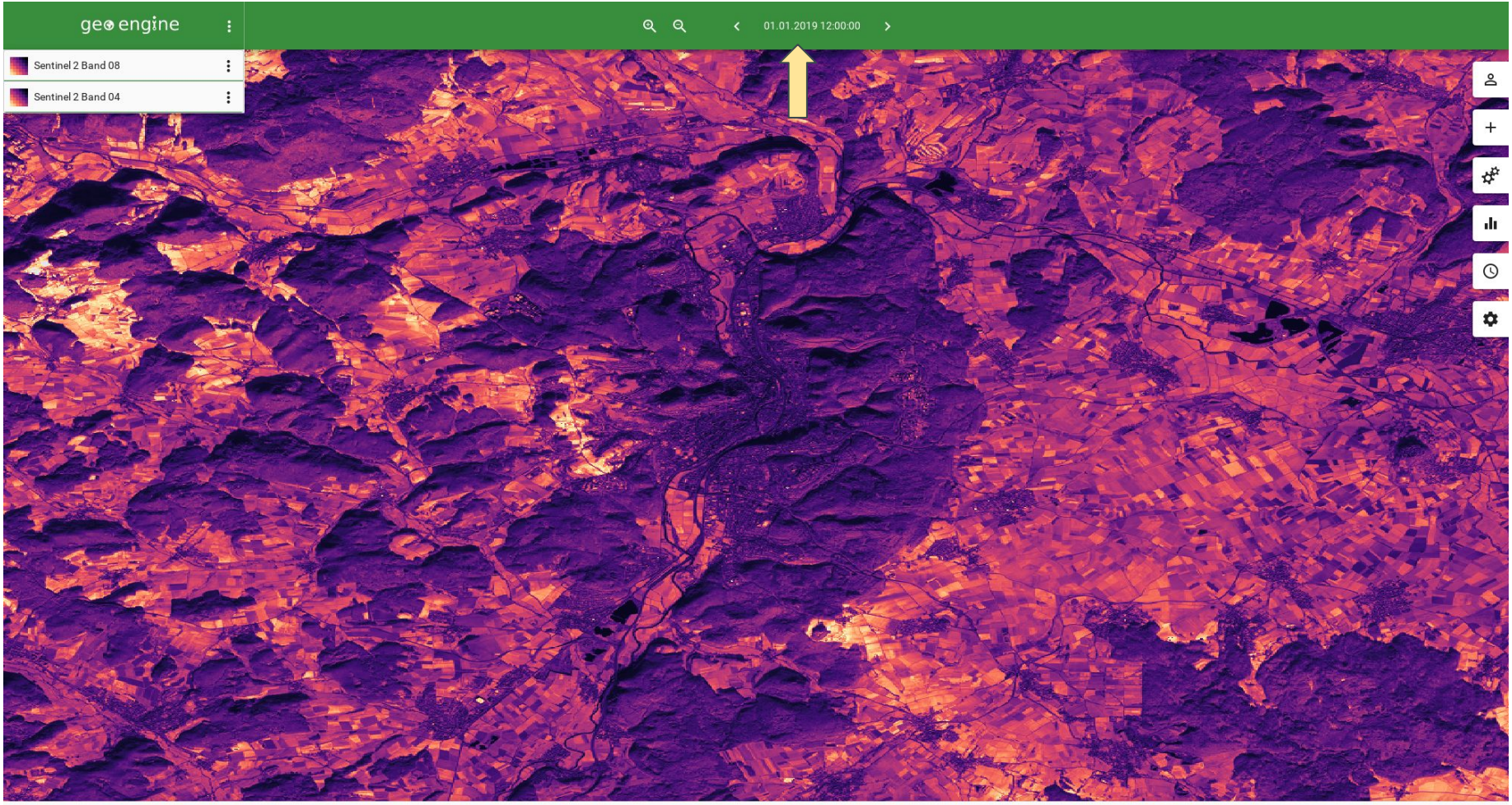
Workflow Description

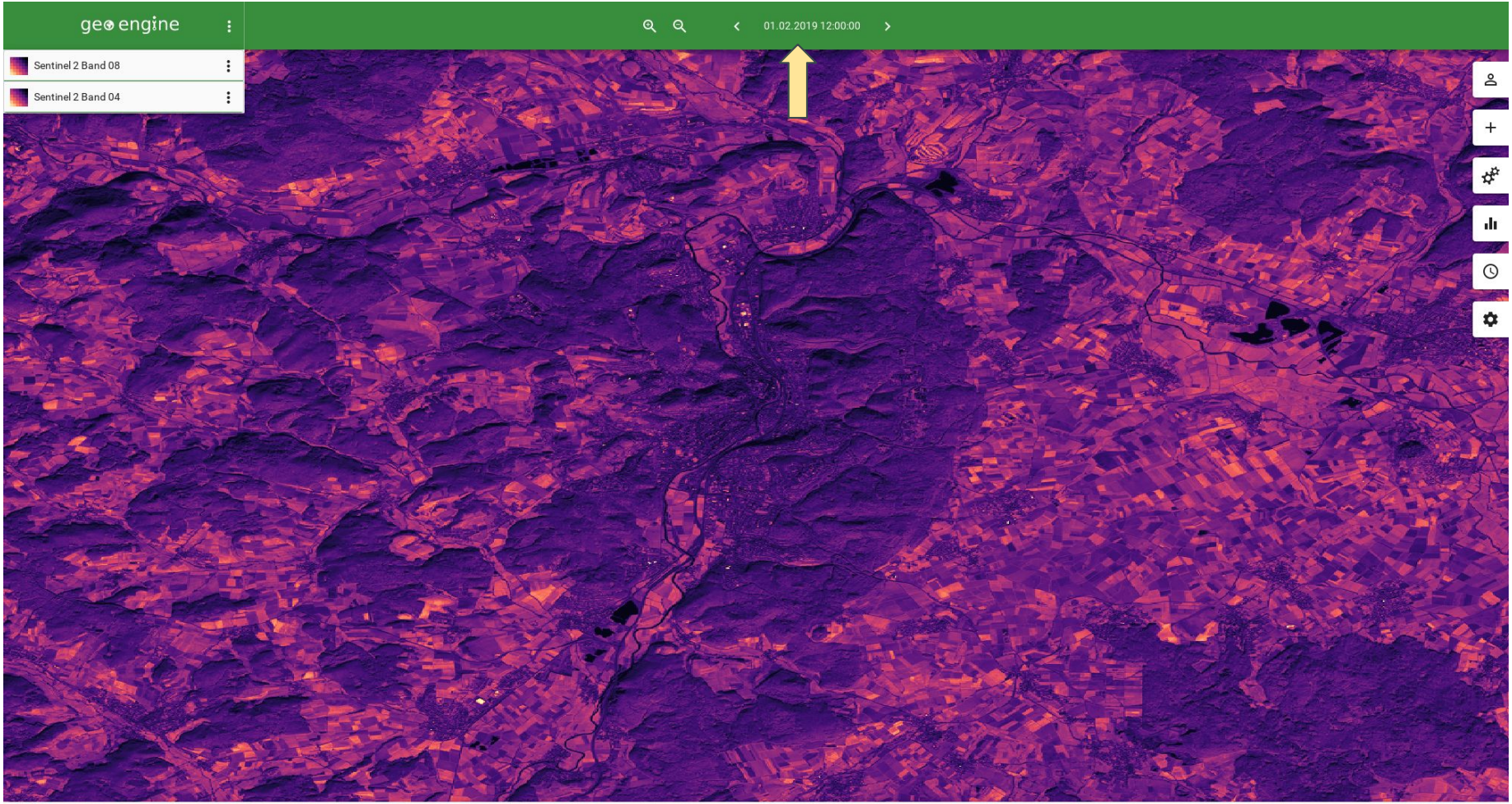
Concept 3: Query like a Data Cube

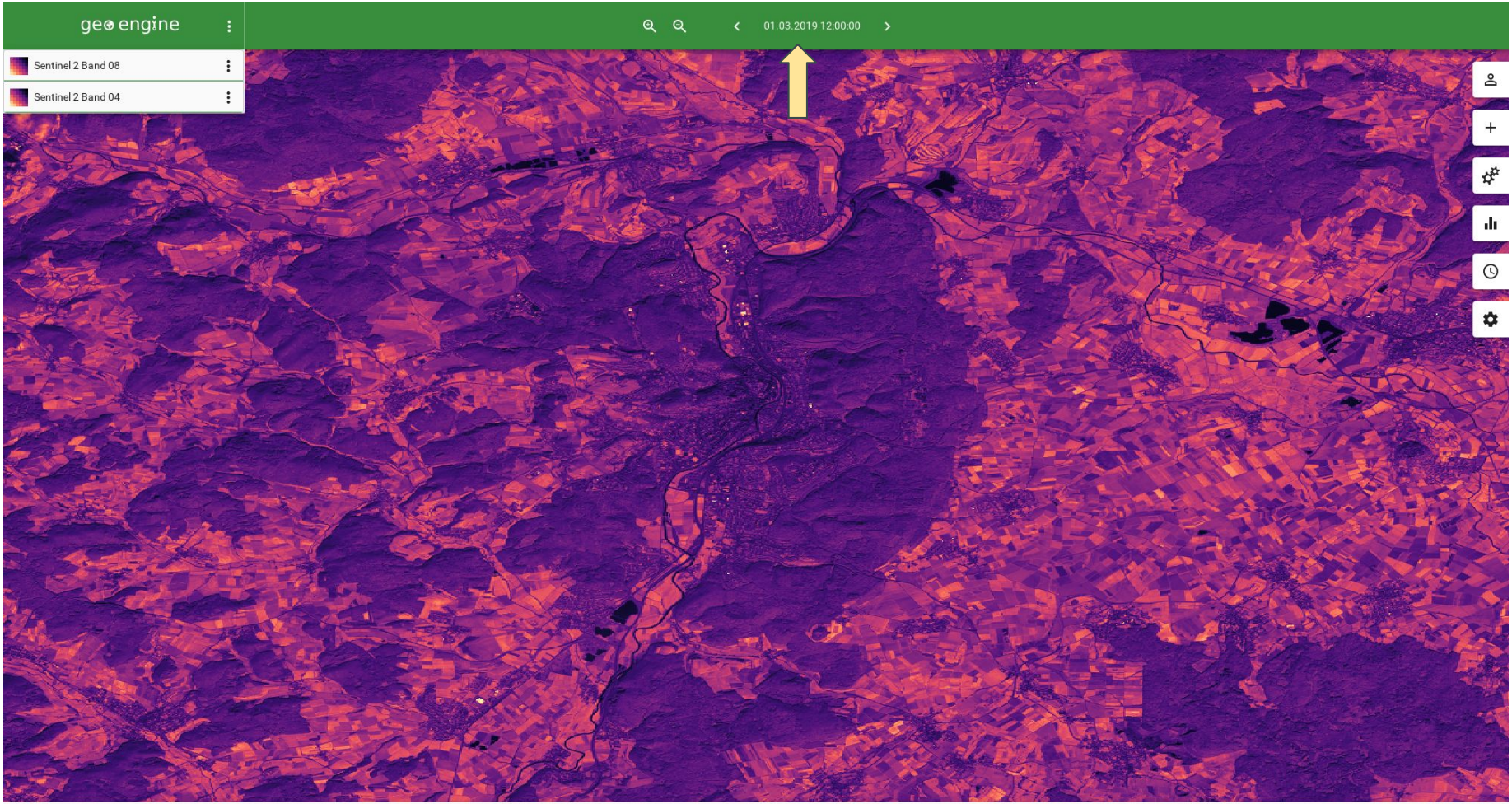


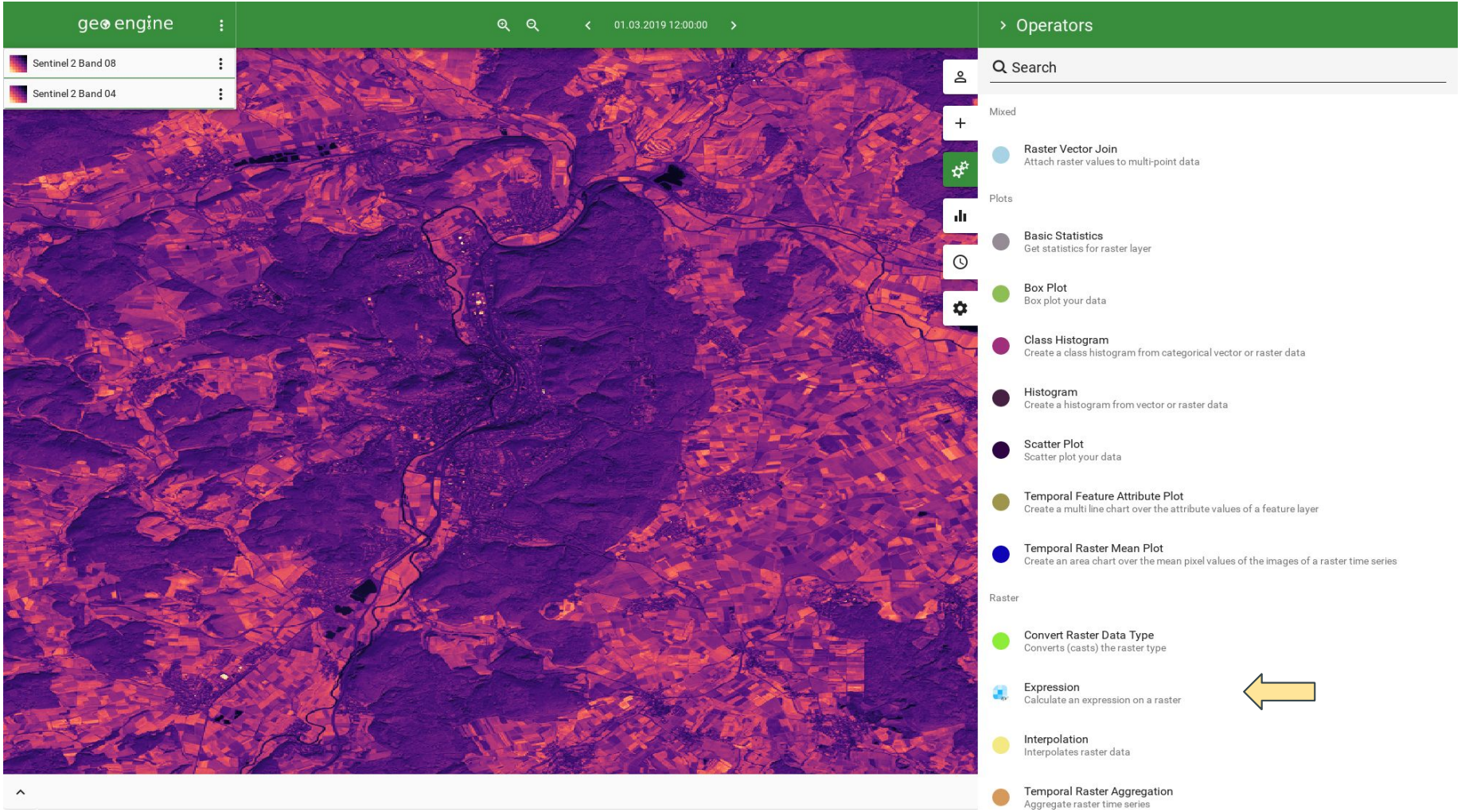
Flipbook Demo

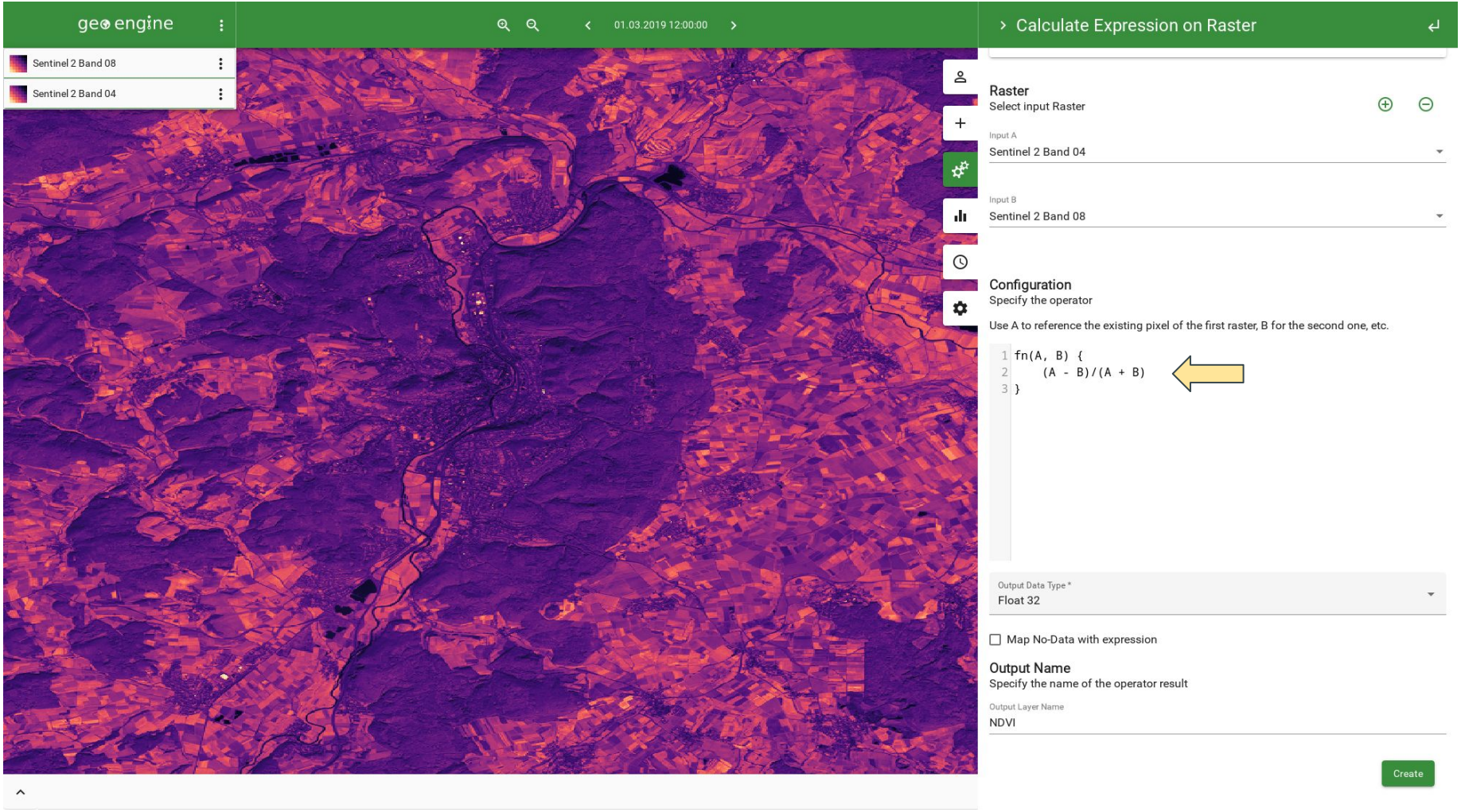

geo engine
add a first layer
01.01.2019 12:00:00
Layer Collection
Datasets
NDVI 10m monthly
NDVI 10m monthly
Sentinel-2 B02 10m de marburg
Sentinel-2 B02 10m de marburg
Sentinel-2 B03 10m de marburg
Sentinel-2 B03 10m de marburg
Sentinel-2 B04 10m de marburg
Sentinel-2 B04 10m de marburg
Sentinel-2 B08 10m de marburg
Sentinel-2 B08 10m de marburg
Mundialis Landuse classification Sentinel-2 B02 10m de marburg
Mundialis Landuse classification Sentinel-2 B02 10m de marburg
Leaf Area Index: version 1, 333m resolution, globe, 10-daily
LAI was defined by CEOS as half the developed area of the convex hull wrapping the green c...
Sentinel 2 Band 01
Sentinel 2 Band 01
Sentinel 2 Band 02
Sentinel 2 Band 02
Sentinel 2 Band 03
Sentinel 2 Band 03
Sentinel 2 Band 08
Sentinel 2 Band 08
Sentinel 2 Band 04
Sentinel 2 Band 04
Sentinel 2 Band 04 Kenia
Sentinel 2 Band 04 Kenia
Sentinel 2 Band 08 Kenia
Sentinel 2 Band 08 Kenia

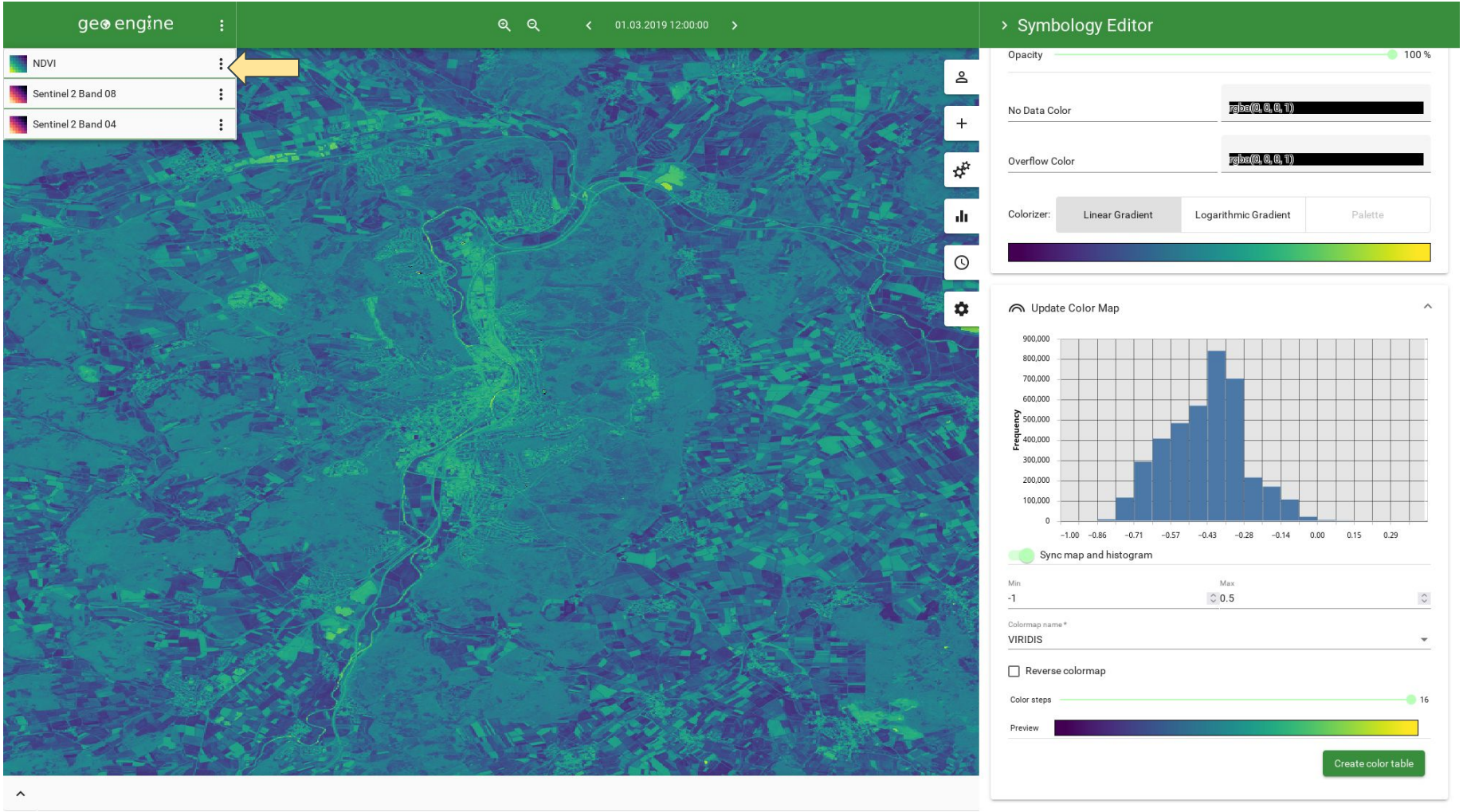

geo engine
Sentinel 2 Band 08
Sentinel 2 Band 04
01.01.2019 12:00:00

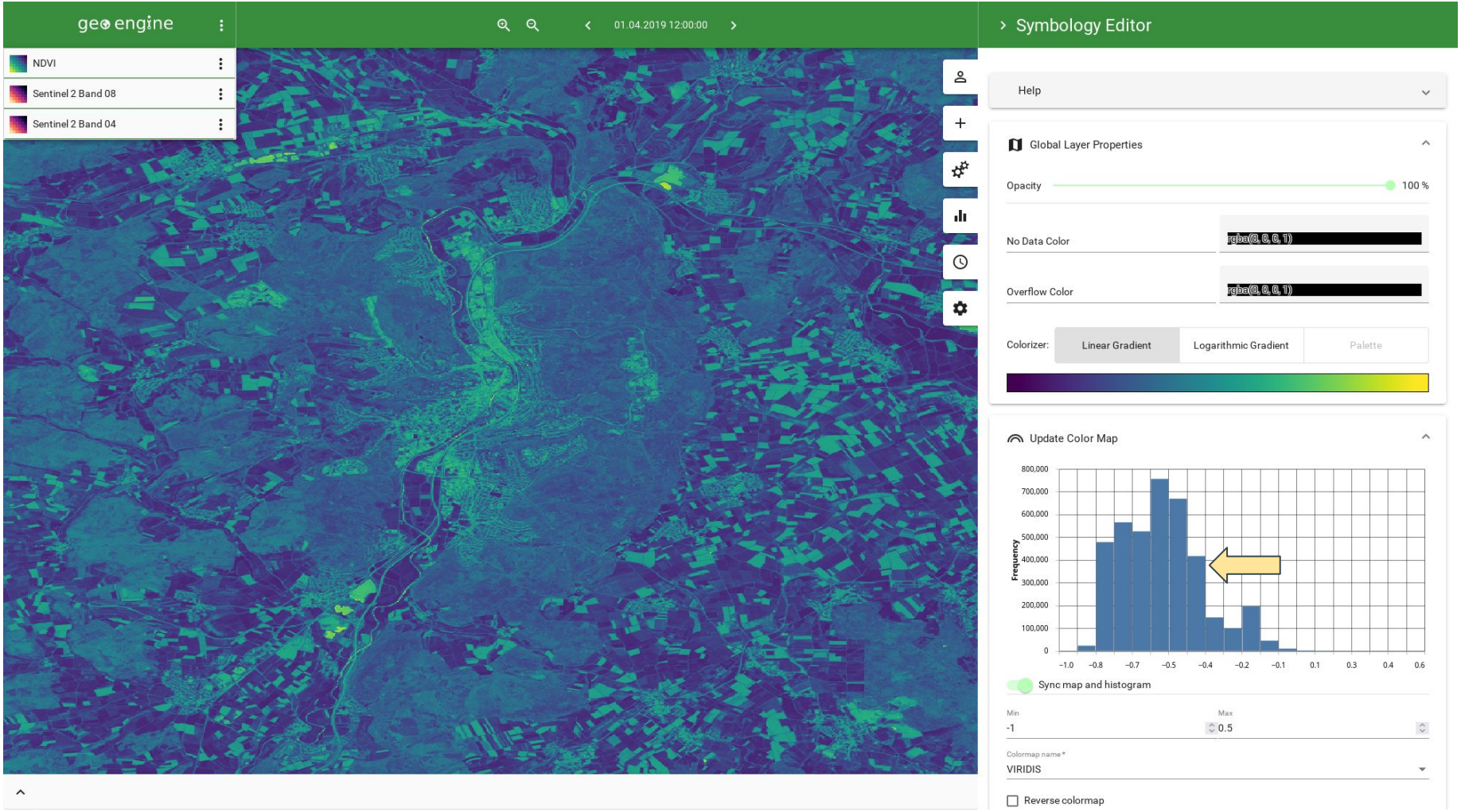

geo engine
Sentinel 2 Band 08
Sentinel 2 Band 04
01.02.2019 12:00:00

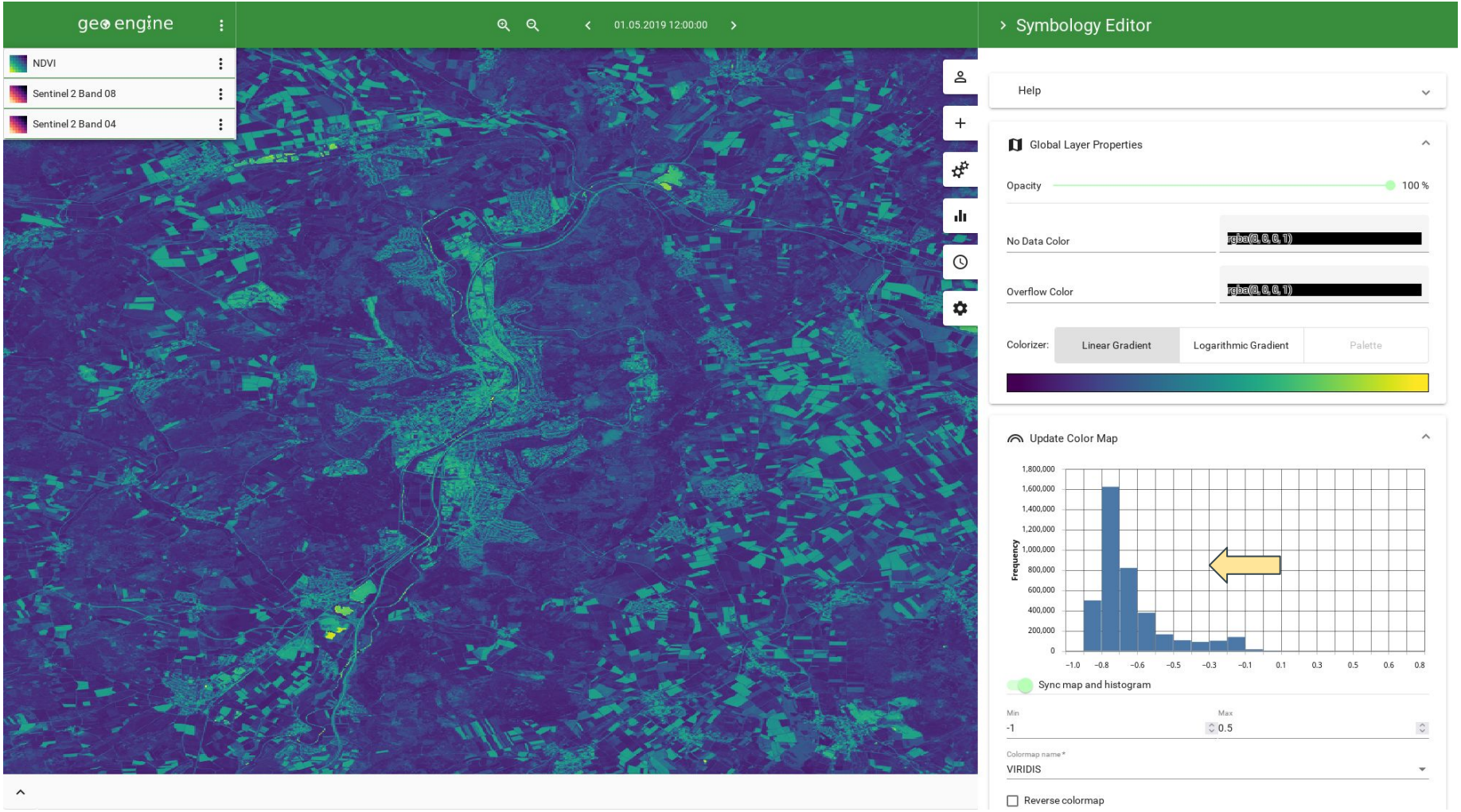

geo engine
01.03.2019 12:00:00
Sentinel 2 Band 08
Sentinel 2 Band 04

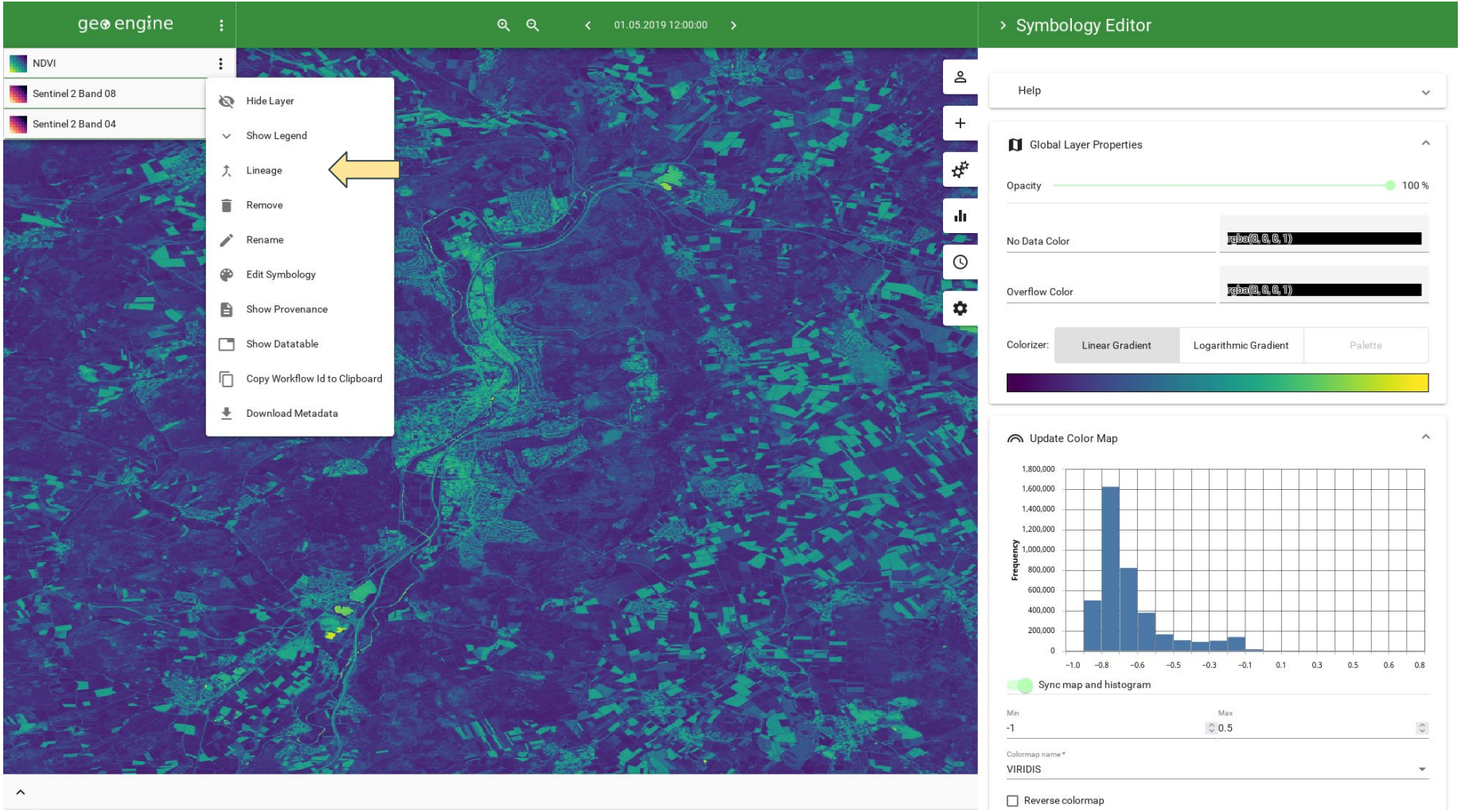

geo engine
01.03.2019 12:00:00
Sentinel 2 Band 08
Sentinel 2 Band 04
Operators
Search
Mixed
Raster Vector Join
Attach raster values to multi-point data
Plots
Basic Statistics
Get statistics for raster layer
Box Plot
Box plot your data
Class Histogram
Create a class histogram from categorical vector or raster data
Histogram
Create a histogram from vector or raster data
Scatter Plot
Scatter plot your data
Temporal Feature Attribute Plot
Create a multi line chart over the attribute values of a feature layer
Temporal Raster Mean Plot
Create an area chart over the mean pixel values of the images of a raster time series
Raster
Convert Raster Data Type
Converts (casts) the raster type
Expression
Calculate an expression on a raster
Interpolation
Interpolates raster data
Temporal Raster Aggregation
Aggregate raster time series

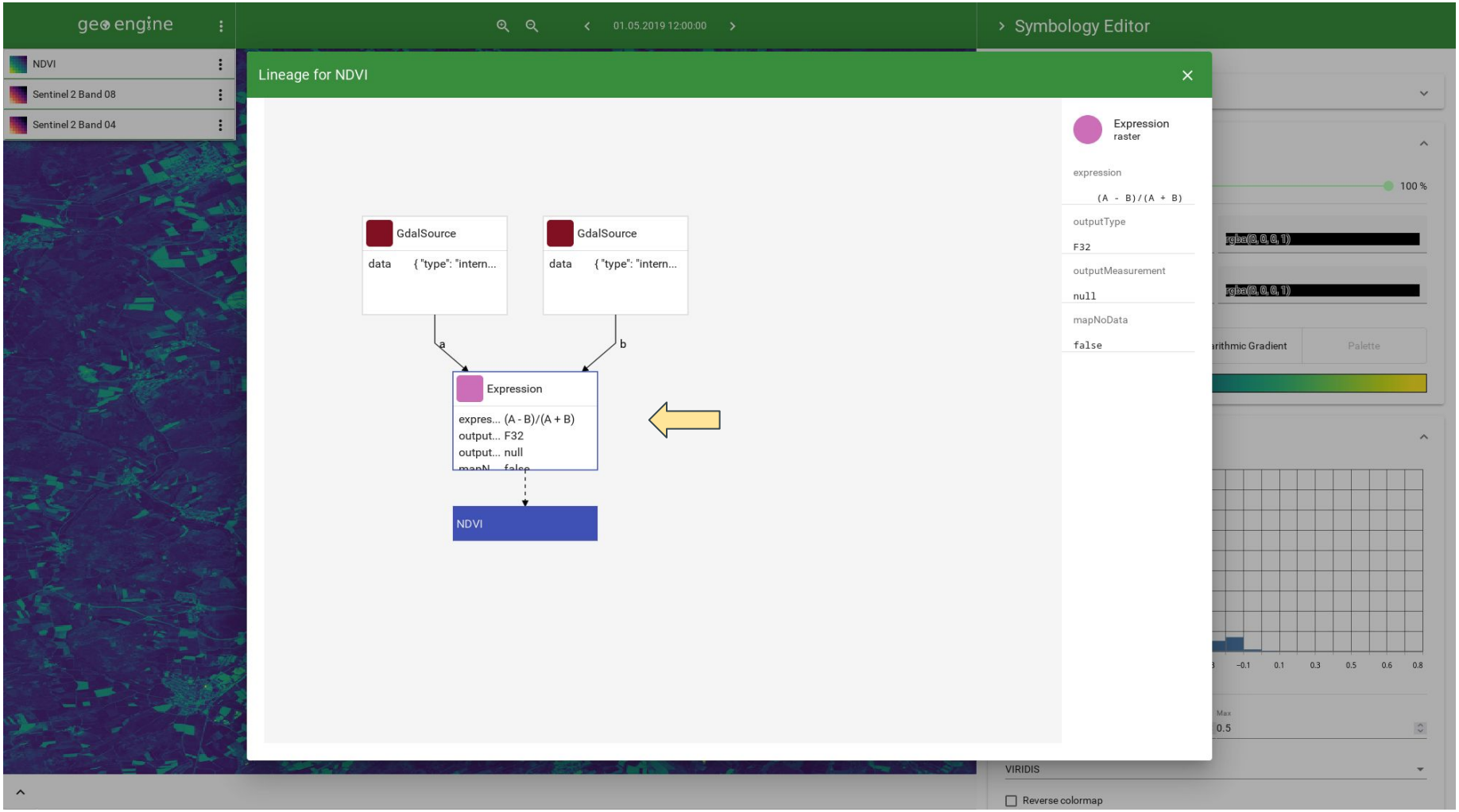

geo engine
01.03.2019 12:00:00
Sentinel 2 Band 08
Sentinel 2 Band 04
Calculate Expression on Raster
Raster
Select input Raster
Input A
Sentinel 2 Band 04
Input B
Sentinel 2 Band 08
Configuration
Specify the operator
Use A to reference the existing pixel of the first raster, B for the second one, etc.
1 fn(A, B) {
2 (A - B)/(A + B)
3 }
Output Data Type *
Float 32
Map No-Data with expression
Output Name
Specify the name of the operator result
Output Layer Name
NDVI
Create

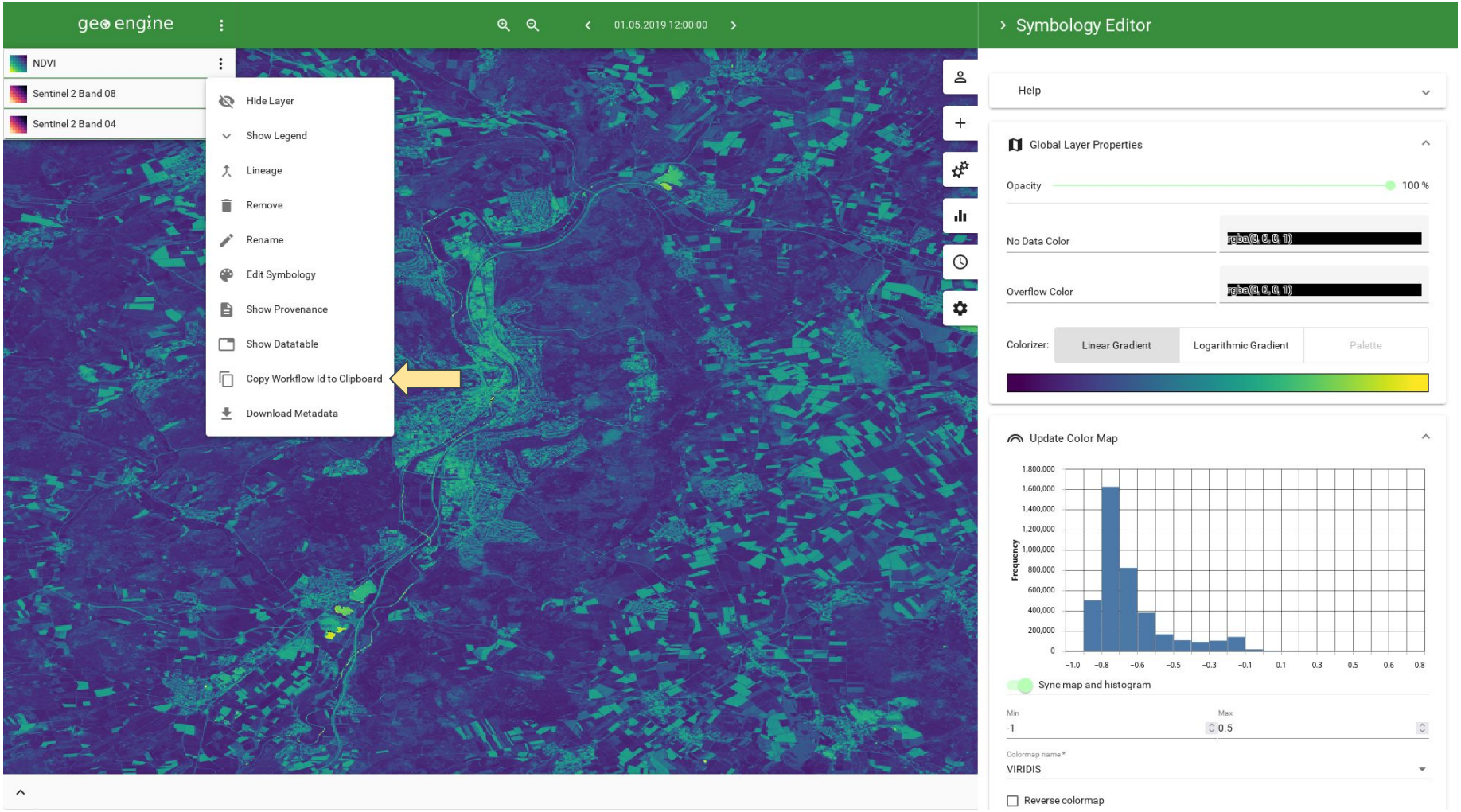

geo engine
01.03.2019 12:00:00
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04
Symbology Editor
Opacity
100 %
No Data Color
rgba(0, 0, 0, 1)
Overflow Color
rgba(0, 0, 0, 1)
Colorizer:
Linear Gradient
Logarithmic Gradient
Palette
Update Color Map
900,000
800,000
700,000
600,000
500,000
400,000
300,000
200,000
100,000
0
Frequency
−1.00
−0.86
−0.71
−0.57
−0.43
−0.28
−0.14
0.00
0.15
0.29
Sync map and histogram
Min
-1
Max
0.5
Colormap name *
VIRIDIS
Reverse colormap
Color steps
16
Preview
Create color table


geo engine
01.04.2019 12:00:00
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04
Symbology Editor
Help
Global Layer Properties
Opacity
100 %
No Data Color
rgba(0, 0, 0, 1)
Overflow Color
rgba(0, 0, 0, 1)
Colorizer:
Linear Gradient
Logarithmic Gradient
Palette
Update Color Map
Frequency
800,000
700,000
600,000
500,000
400,000
300,000
200,000
100,000
0
−1.0
−0.8
−0.7
−0.5
−0.4
−0.2
−0.1
0.1
0.3
0.4
0.6
Sync map and histogram
Min
-1
Max
0.5
Colormap name *
VIRIDIS
Reverse colormap


geo engine
01.05.2019 12:00:00
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04
Symbology Editor
Help
Global Layer Properties
Opacity
100 %
No Data Color
rgba(0, 0, 0, 1)
Overflow Color
rgba(0, 0, 0, 1)
Colorizer:
Linear Gradient
Logarithmic Gradient
Palette
Update Color Map
Frequency
1,800,000
1,600,000
1,400,000
1,200,000
1,000,000
800,000
600,000
400,000
200,000
0
−1.0
−0.8
−0.6
−0.5
−0.3
−0.1
0.1
0.3
0.5
0.6
0.8
Sync map and histogram
Min
-1
Max
0.5
Colormap name *
VIRIDIS
Reverse colormap


geo engine
01.05.2019 12:00:00
Symbology Editor
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04
Hide Layer
Show Legend
Lineage
Remove
Rename
Edit Symbology
Show Provenance
Show Datatable
Copy Workflow Id to Clipboard
Download Metadata
Help
Global Layer Properties
Opacity
100 %
No Data Color
rgba(0, 0, 0, 1)
Overflow Color
rgba(0, 0, 0, 1)
Colorizer:
Linear Gradient
Logarithmic Gradient
Palette
Update Color Map
Frequency
1,800,000
1,600,000
1,400,000
1,200,000
1,000,000
800,000
600,000
400,000
200,000
0
−1.0
−0.8
−0.6
−0.5
−0.3
−0.1
0.1
0.3
0.5
0.6
0.8
Sync map and histogram
Min
-1
Max
0.5
Colormap name *
VIRIDIS
Reverse colormap


geo engine
01.05.2019 12:00:00
Symbology Editor
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04
Lineage for NDVI
GdalSource
data { "type": "intern...
GdalSource
data { "type": "intern...
a
b
Expression
expres... (A - B)/(A + B)
output... F32
output... null
NDVI
Expression
raster
expression
(A - B)/(A + B)
outputType
F32
outputMeasurement
null
mapNoData
false
100 %
Palette
VIRIDIS
Reverse colormap


geo engine
01.05.2019 12:00:00
Symbology Editor
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04
Hide Layer
Show Legend
Lineage
Remove
Rename
Edit Symbology
Show Provenance
Show Datatable
Copy Workflow Id to Clipboard
Download Metadata
Help
Global Layer Properties
Opacity
100 %
No Data Color
rgba(0, 0, 0, 1)
Overflow Color
rgba(0, 0, 0, 1)
Colorizer:
Linear Gradient
Logarithmic Gradient
Palette
Update Color Map
Frequency
1,800,000
1,600,000
1,400,000
1,200,000
1,000,000
800,000
600,000
400,000
200,000
0
−1.0
−0.8
−0.6
−0.5
−0.3
−0.1
0.1
0.3
0.5
0.6
0.8
Sync map and histogram
Min
-1
Max
0.5
Colormap name *
VIRIDIS
Reverse colormap

File Edit View Run Kernel Tabs Settings Help

Launcher | ndvi_inverse_points.ipynb

Markdown

```
[ ]: import geoengine as ge

*[6]: ge.initialize("https://demo.app.geoengine.io/api",token="889599ce-e020-4a69-a196-e464eb0d5dbe")

[7]: session = ge.get_session()
     session
```

•••

Use the pre-defined workflow

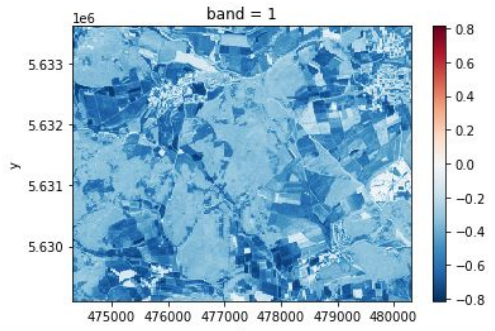
```
[8]: ndvi_workflow = ge.workflow_by_id("6a21e4af-a711-5e6d-aa95-43bf56518153")

[9]: ndvi_workflow.get_result_descriptor()

[9]: Data type:         F32
     Spatial Reference: EPSG:4326
     Measurement:       unitless
```

Request data from the virtual layer

```
[10]: time_start = datetime.strptime(
          '2019-02-01T12:00:00.000Z', "%Y-%m-%dT%H:%M:%S.%f%z")
      time_end = datetime.strptime(
          '2019-03-01T12:00:00.000Z', "%Y-%m-%dT%H:%M:%S.%f%z")

      query_rectangle = ge.QueryRectangle(
              [474304.8,5629111.5, 480289.9,5633621.2], # uniwald
              [time_start, time_end],
              srs= "EPSG:32632",
              resolution=[20,20]
              )

[12]: ndvi_array = ndvi_workflow.get_xarray(query_rectangle)
      display(ndvi_array)
```

•••

```
[13]: ndvi_array.plot()

[13]: <matplotlib.collections.QuadMesh at 0x7f9e1e4df670>
```

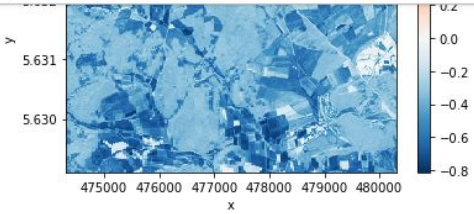


Simple 0 6 Python 3 (ipykernel) | Idle

File Edit View Run Kernel Tabs Settings Help

Launcher | ndvi_inverse_points.ipynb

Code



Extend the workflow

[14]:
```
workflow_spec = {
    "type": "Raster",
    "operator": {
        "type": "Expression",
        "params": {
            "expression": "(-1.0* A)", # this is the NDVI formula
            "outputType": "F32",
            "mapNoData": False
        },
        "sources": {
            "a": ndvi_workflow.workflow_definition()["operator"]
        }
    }
}
inverse_workflow = ge.register_workflow(workflow_spec)
inverse_workflow
```

[14]: ee9eb3b4-12a8-511b-b11b-6d40dff80efb

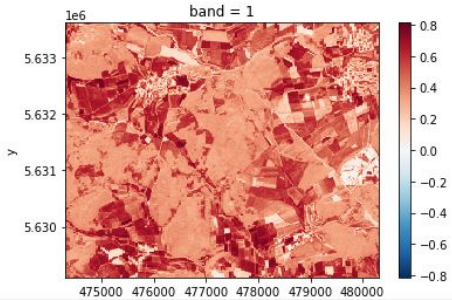
Request inverse data

[15]:
```
inverse_array = inverse_workflow.get_xarray(query_rectangle)
```

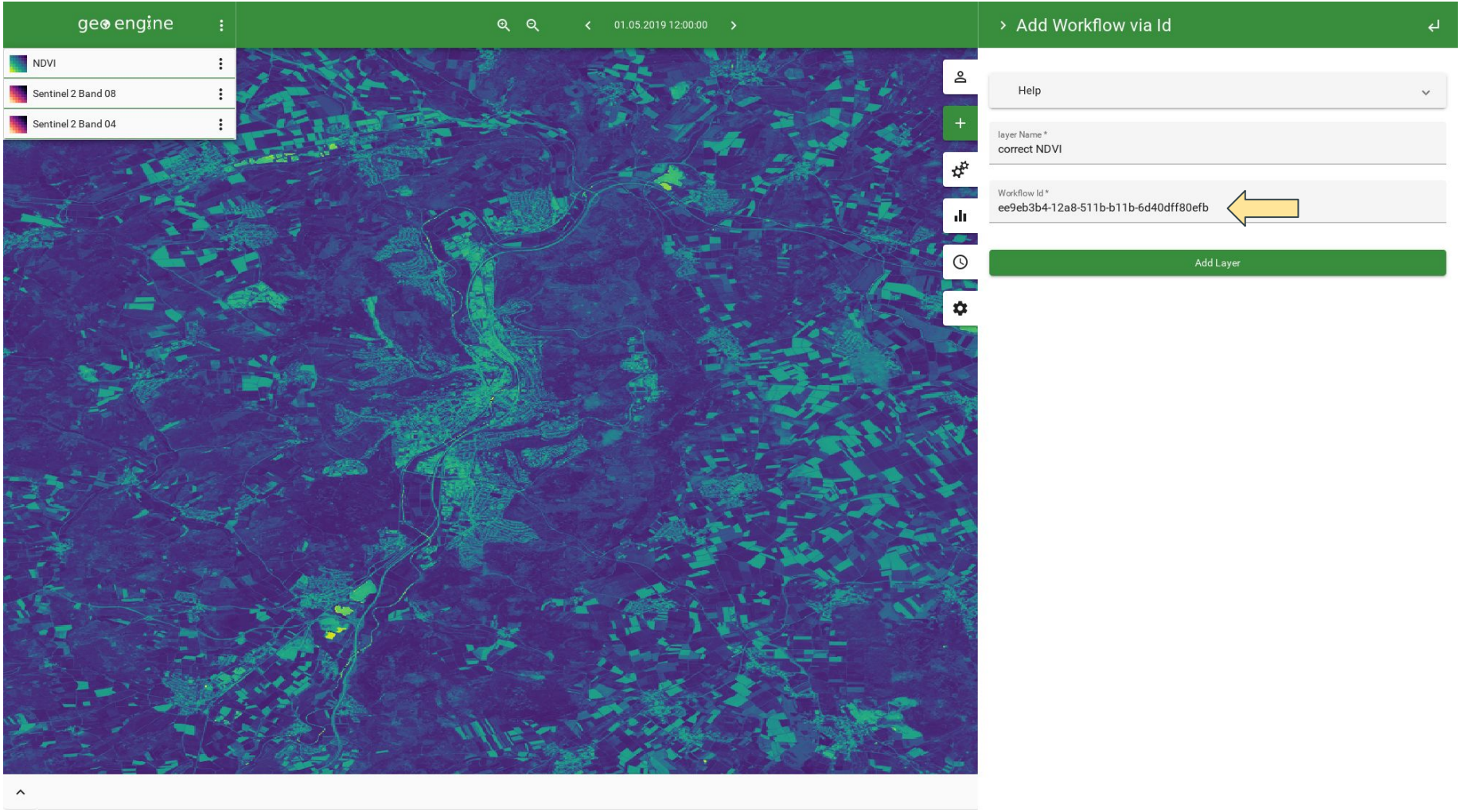
Query the NDVI raster

[16]:
```
inverse_array.plot()
```

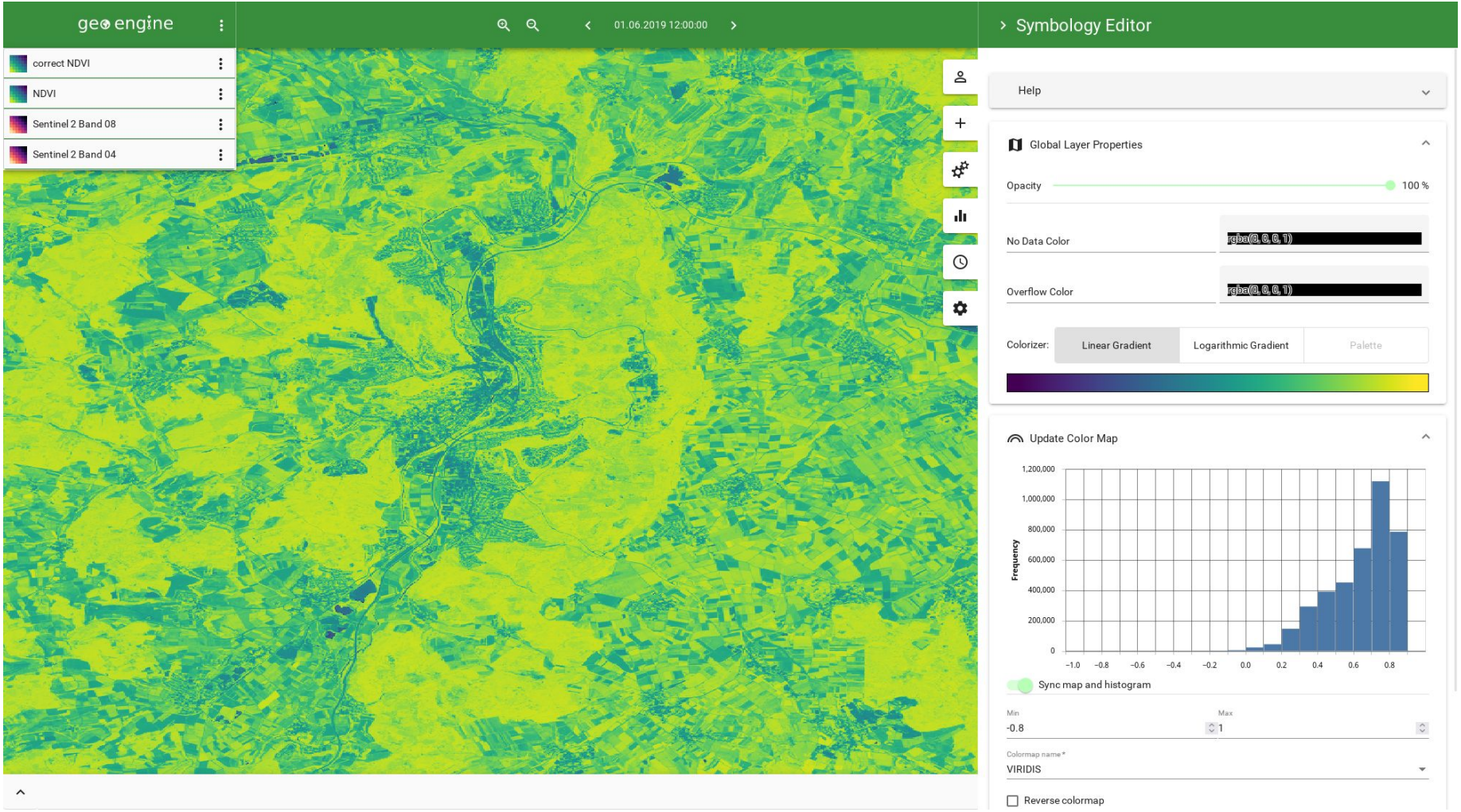
[16]: <matplotlib.collections.QuadMesh at 0x7f9e1625da20>

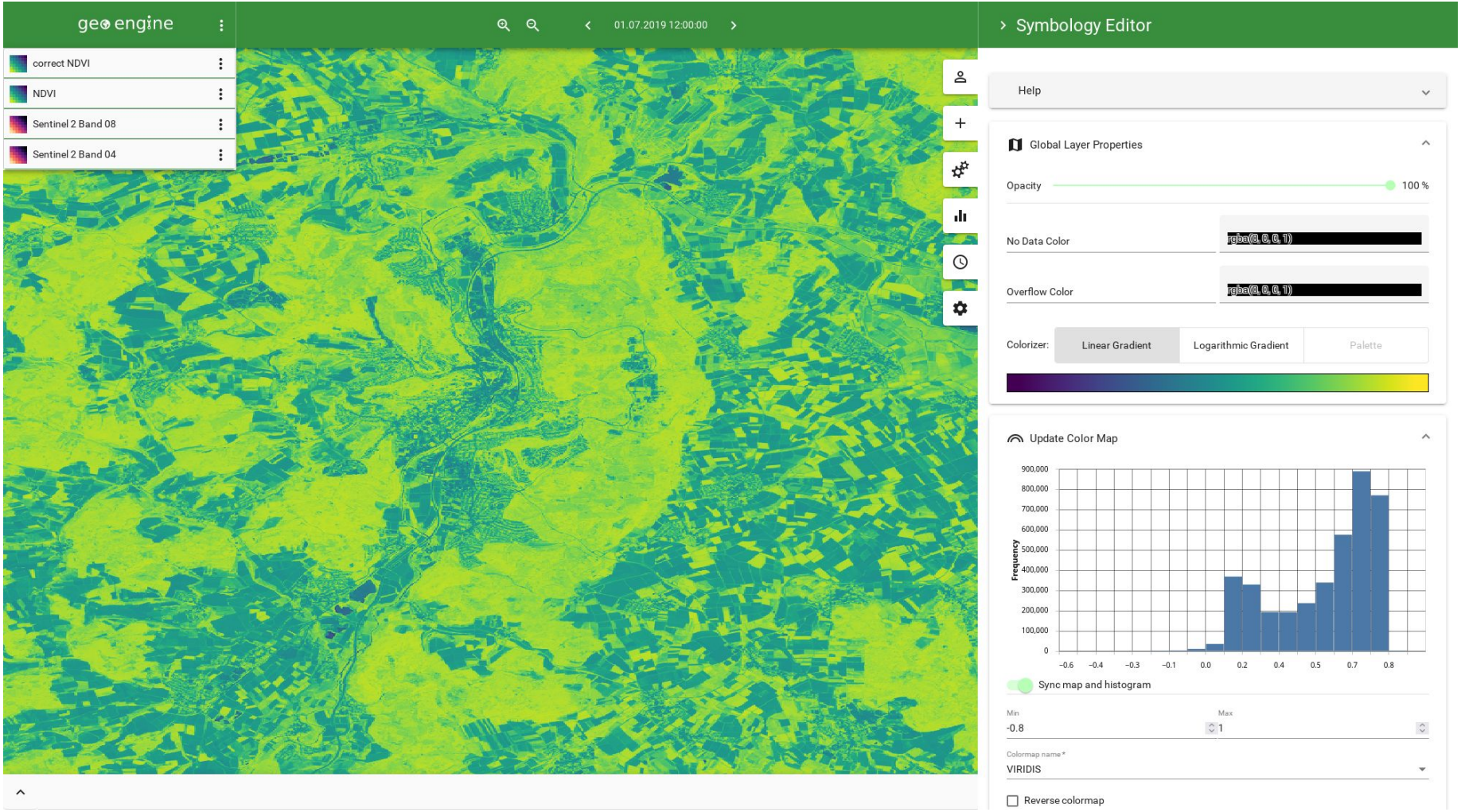


Simple 0 6 Python 3 (ipykernel) | Idle

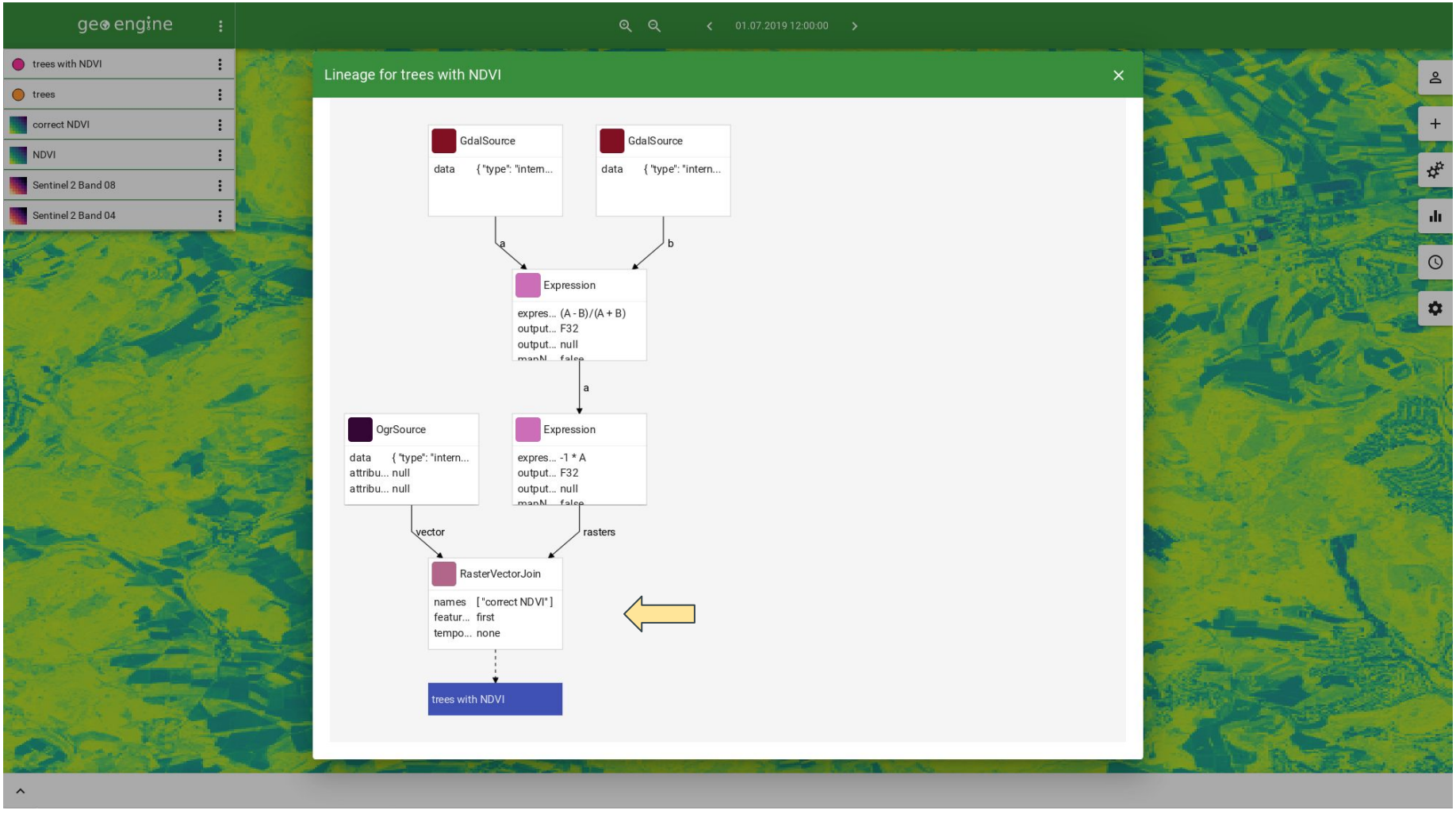

geo engine
01.05.2019 12:00:00
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04
Add Workflow via Id
Help
layer Name *
correct NDVI
Workflow Id *
ee9eb3b4-12a8-511b-b11b-6d40dff80efb
Add Layer

geo engine
01.05.2019 12:00:00
correct NDVI
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04
Symbology Editor
Opacity
100 %
No Data Color
rgba(0, 0, 0, 1)
Overflow Color
rgba(0, 0, 0, 1)
Colorizer:
Linear Gradient
Logarithmic Gradient
Palette
Update Color Map
Frequency
1,800,000
1,600,000
1,400,000
1,200,000
1,000,000
800,000
600,000
400,000
200,000
0
-0.8
-0.6
-0.5
-0.3
-0.1
0.1
0.3
0.5
0.6
0.8
1.0
Sync map and histogram
Min
-0.8
Max
1
Colormap name *
VIRIDIS
Reverse colormap
Color steps
16
Preview
Create color table


geo engine
01.06.2019 12:00:00
correct NDVI
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04
Symbology Editor
Help
Global Layer Properties
Opacity
100 %
No Data Color
rgba(0, 0, 0, 1)
Overflow Color
rgba(0, 0, 0, 1)
Colorizer:
Linear Gradient
Logarithmic Gradient
Palette
Update Color Map
1,200,000
1,000,000
800,000
600,000
400,000
200,000
0
Frequency
−1.0
−0.8
−0.6
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8
Sync map and histogram
Min
-0.8
Max
1
Colormap name *
VIRIDIS
Reverse colormap


geo engine
01.07.2019 12:00:00
correct NDVI
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04
Symbology Editor
Help
Global Layer Properties
Opacity
100 %
No Data Color
rgba(0, 0, 0, 1)
Overflow Color
rgba(0, 0, 0, 1)
Colorizer:
Linear Gradient
Logarithmic Gradient
Palette
Update Color Map
Frequency
900,000
800,000
700,000
600,000
500,000
400,000
300,000
200,000
100,000
0
−0.6
−0.4
−0.3
−0.1
0.0
0.2
0.4
0.5
0.7
0.8
Sync map and histogram
Min
-0.8
Max
1
Colormap name *
VIRIDIS
Reverse colormap

geo engine
01.07.2019 12:00:00
trees with NDVI
trees
correct NDVI
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04


geo engine
01.07.2019 12:00:00
trees with NDVI
trees
correct NDVI
NDVI
Sentinel 2 Band 08
Sentinel 2 Band 04
Lineage for trees with NDVI
GdalSource
data { "type": "intern...
GdalSource
data { "type": "intern...
a
b
Expression
expres... (A - B)/(A + B)
output... F32
output... null
a
OgrSource
data { "type": "intern...
attribu... null
attribu... null
Expression
expres... -1 * A
output... F32
output... null
vector
rasters
RasterVectorJoin
names ["correct NDVI"]
featur... first
tempo... none
trees with NDVI

File Edit View Run Kernel Tabs Settings Help

Launcher | ndvi_inverse_points.ipynb

Code | Python 3 (ipykernel)

•••

[17]:
```
points_with_ndvi_workflow = ge.workflow_by_id("e9cceb15-9b37-5ba3-b218-3224d437acfb")
```

[18]:
```
points_with_ndvi_workflow.get_result_descriptor()
```

•••

[19]:
```
point_gdf = points_with_ndvi_workflow.get_dataframe(query_rectangle)
point_gdf
```

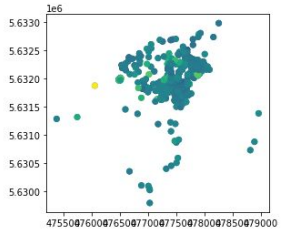
[19]:

	geometry	easting	creator_id	remarks	geomeasurement_id	northing	correct ndvi	status	tree_id	species_id	start	end
0	POINT (477491.050 5632265.790)	477491.050	jk	Plakettenverlust nach Fällung	7	5632265.790	0.397089	NaN	b00001	4269	2019-02-01 00:00:00+00:00	2019-03-01 00:00:00+00:00
1	POINT (477574.220 5632274.470)	477574.220	jk	None	7	5632274.470	0.481585	0.0	b00002	2357	2019-02-01 00:00:00+00:00	2019-03-01 00:00:00+00:00
2	POINT (477490.930 5632266.280)	477490.930	jk	Plakettenverlust nach Fällung	7	5632266.280	0.397089	NaN	b00003	4269	2019-02-01 00:00:00+00:00	2019-03-01 00:00:00+00:00
3	POINT (477535.150 5631810.850)	477535.150	dg	None	1	5631810.850	0.275136	0.0	b00004	2357	2019-02-01 00:00:00+00:00	2019-03-01 00:00:00+00:00
4	POINT (477493.470 5632266.100)	477493.470	jk	Plakettenverlust nach Fällung	7	5632266.100	0.397089	NaN	b00006	4269	2019-02-01 00:00:00+00:00	2019-03-01 00:00:00+00:00
...	...	...	...	...	...	...	...	...	...	...	...	...
2628	POINT (477871.838 5632209.359)	477871.838	dg	RTK Float	5	5632209.359	0.326692	0.0	b00954	2357	2019-03-01 00:00:00+00:00	2019-04-01 00:00:00+00:00
2629	POINT (477871.489 5632206.837)	477871.489	dg	RTK Float	5	5632206.837	0.326692	0.0	b00955	2357	2019-03-01 00:00:00+00:00	2019-04-01 00:00:00+00:00
2630	POINT (477872.012 5632203.727)	477872.012	dg	RTK Float	5	5632203.727	0.326692	0.0	b00956	4680	2019-03-01 00:00:00+00:00	2019-04-01 00:00:00+00:00
2631	POINT (477869.828 5632204.047)	477869.828	dg	RTK Float	5	5632204.047	0.326692	0.0	b00957	2357	2019-03-01 00:00:00+00:00	2019-04-01 00:00:00+00:00
2632	POINT (477869.246 5632199.116)	477869.246	dg	RTK Float	5	5632199.116	0.326692	0.0	b00958	2357	2019-03-01 00:00:00+00:00	2019-04-01 00:00:00+00:00

2633 rows × 12 columns

[21]:
```
point_gdf.plot("correct ndvi")
```

[21]: <AxesSubplot:>



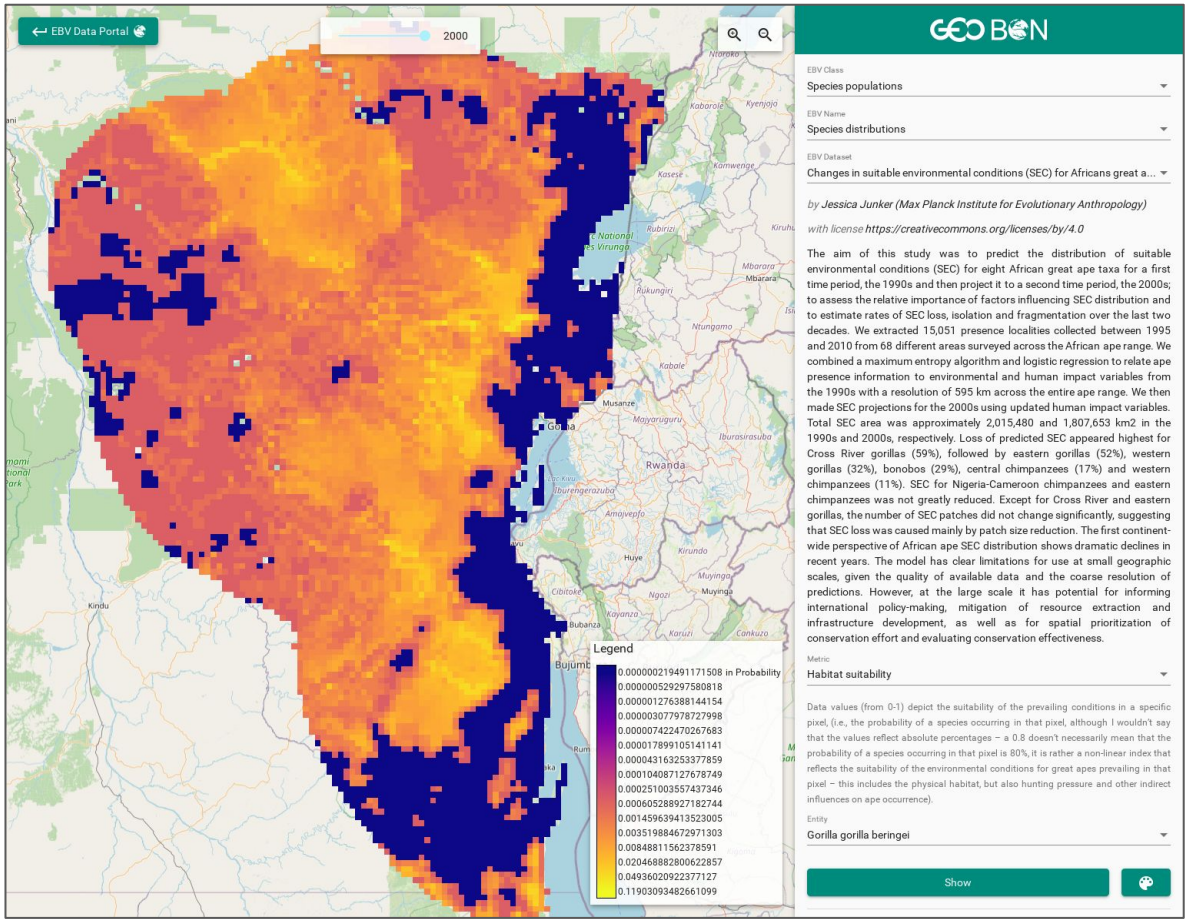
Simple 0 6 Python 3 (ipykernel) | Idle Mode: Command Ln 1, Col 1 ndvi_inverse_points.ipynb

DEMO

Custom Apps

GEO BON

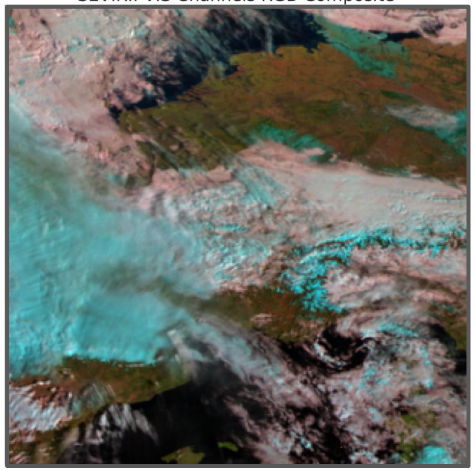
EBV Analyzer

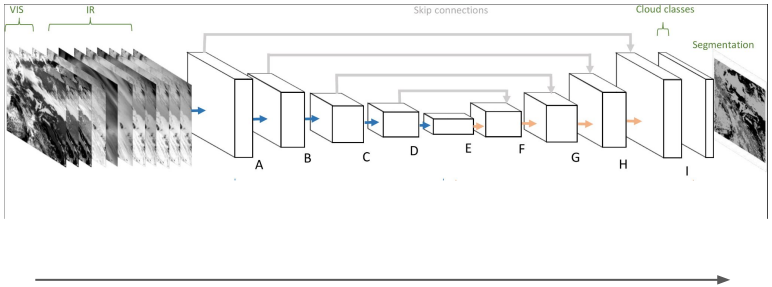


Outlook: AI Integration

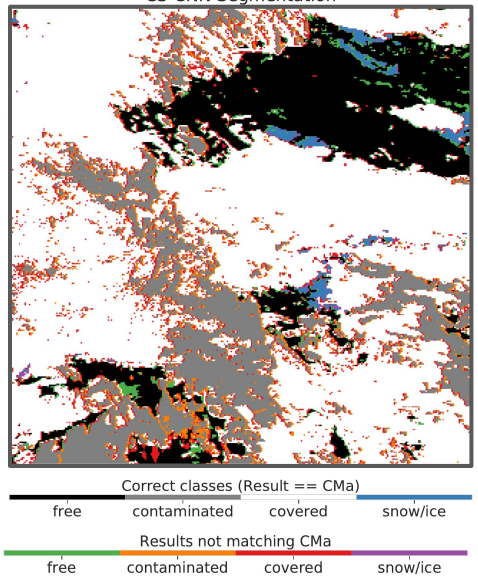
remote sensing
MDPI
Article
Fast Cloud Segmentation Using Convolutional Neural Networks
Johannes Drönner [1,*], Nikolaus Korfhage [1], Sebastian Egli [2], Markus Mühling [1], Boris Thies [2], Jörg Bendix [2], Bernd Freisleben [1] and Bernhard Seeger [1]

SEVIRI: VIS Channels RGB-Composite



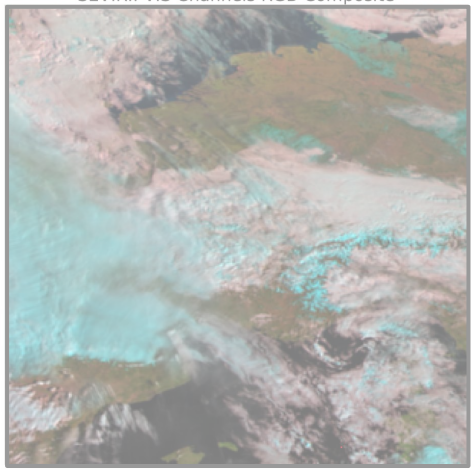


CS-CNN Segmentation



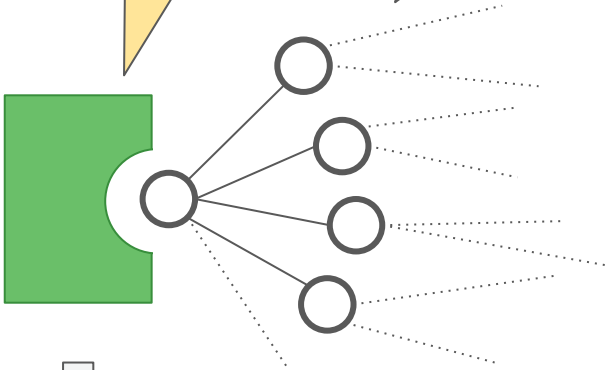
Outlook: AI Integration

SEVIRI: VIS Channels RGB-Composite



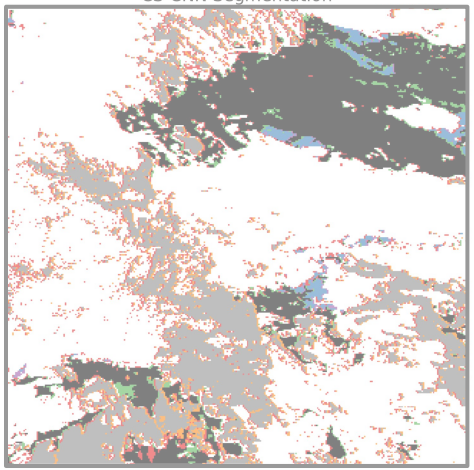
AI "Trainer"

Data preparation workflow



AI Operator

CS-CNN Segmentation



Correct classes (Result == CMa)

free contaminated covered snow/ice

Results not matching CMa

free contaminated covered snow/ice

Summary & Outlook

Summary

- Geo Engine as a geoprocessing system
  - Processing engine is open-source
  - Time series processing
  - Automatic harmonization
  - Reusable workflows
- Work in different environments
  - User Interface
  - Jupyter Notebook
- Flexible system for applications
  - Data portals (interactive)
  - Processing (long-running tasks)

Outlook

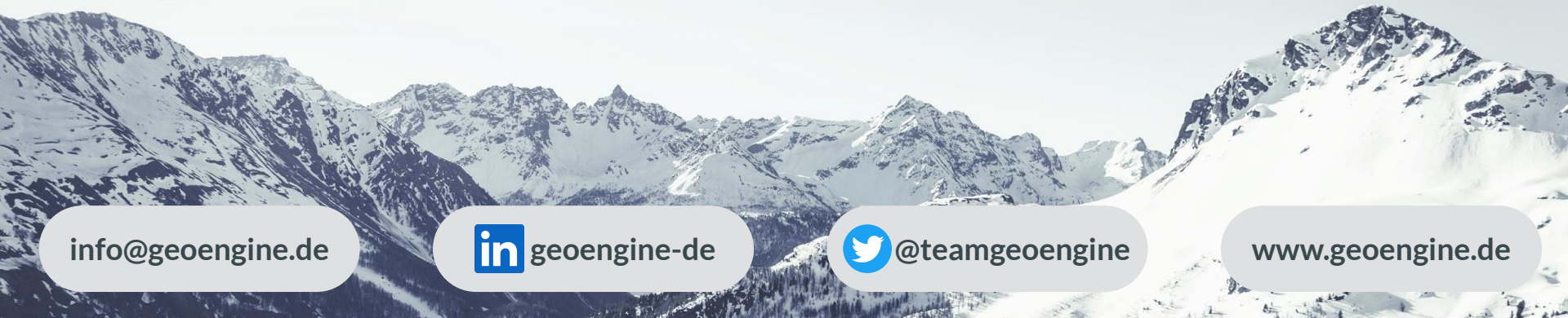
- Integration of AI workflows (RF, CNNs)
- Build use case gallery
- Hosted public demo instance

We are looking for:

- Co-development & Co-innovation
- Projects & Consortiums
- …

Get in touch.

geo engine



info@geoengine.de

geoengine-de

@teamgeoengine

www.geoengine.de