Finding Sasquatch

or how to fake a moon landing

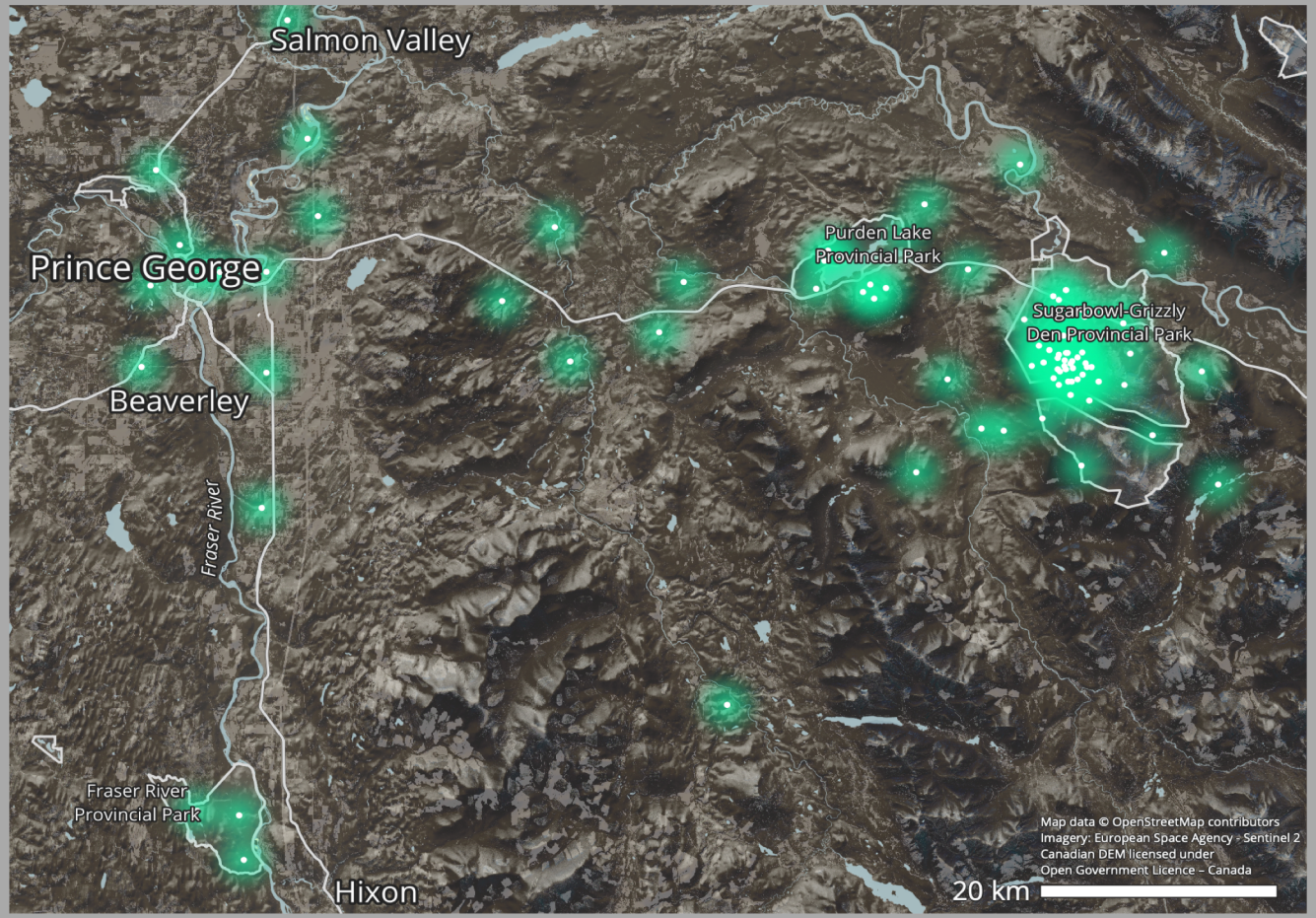
James Banting
Sparkgeo

@jr_banting
james@sparkgeo.com

geo

Sasquatch /ˈsæskwɑtʃ/

1. (cryptozoology) A large hairy humanoid creature of western North America; a Bigfoot.


Salmon Valley
Prince George
Beaverley
Fraser River
Purden Lake
Provincial Park
Sugarbowl-Grizzly
Den Provincial Park
Fraser River
Provincial Park
Hixon
Map data © OpenStreetMap contributors
Imagery: European Space Agency - Sentinel 2
Canadian DEM licensed under
Open Government Licence – Canada
20 km

geo

POST https://sat-api.developmentseed.org/stac/search/ Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

none form-data x-www-form-urlencoded raw binary GraphQL BETA JSON (application/json) Beautify

```
1  {
2      "intersects": {
3          "type": "Polygon",
4          "coordinates": [
5              [...]
27         ]
28     },
29     "collections": [
30         "sentinel-2-l1c",
31         "landsat-8-l1"
32     ],
33     "time": "2019/2019-04",
34     "query": {
35         "eo:cloud_cover": {
36             "lte": 0.4
37         }
38     }
39 }
```

Response

SpatioTemporal Asset Catalog

Defined metadata for geospatial catalogs and assets

- Focus on search and discovery
- Simple and extensible

Static catalogs

- Crawlable

Dynamic APIs

- Searchable on arbitrary fields

www.stacspec.org

POST https://sat-api.developmentseed.org/stac/search/ Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code Comments (0)

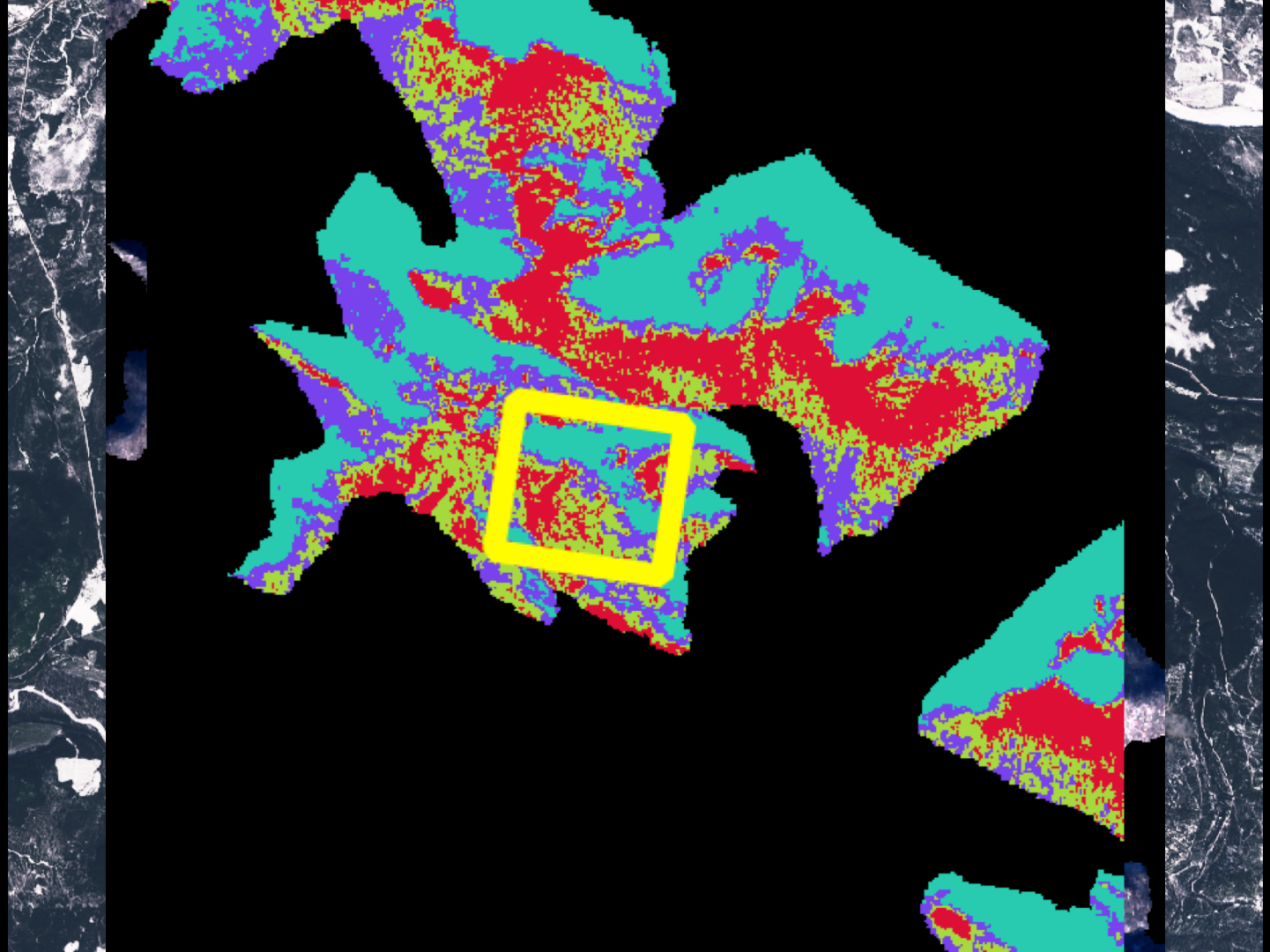
none form-data x-www-form-urlencoded raw binary GraphQL BETA JSON (application/json) Beautify

```
1  {
2      "intersects": {
3          "type": "Polygon",
4          "coordinates": [
5              [...]
27         ]
28     },
29     "collections": [
30         "sentinel-2-l1c",
31         "landsat-8-l1"
32     ],
33     "time": "2019/2019-04",
34     "query": {
35         "eo:cloud_cover": {
36             "lte": 0.4
37         }
38     }
39 }
```

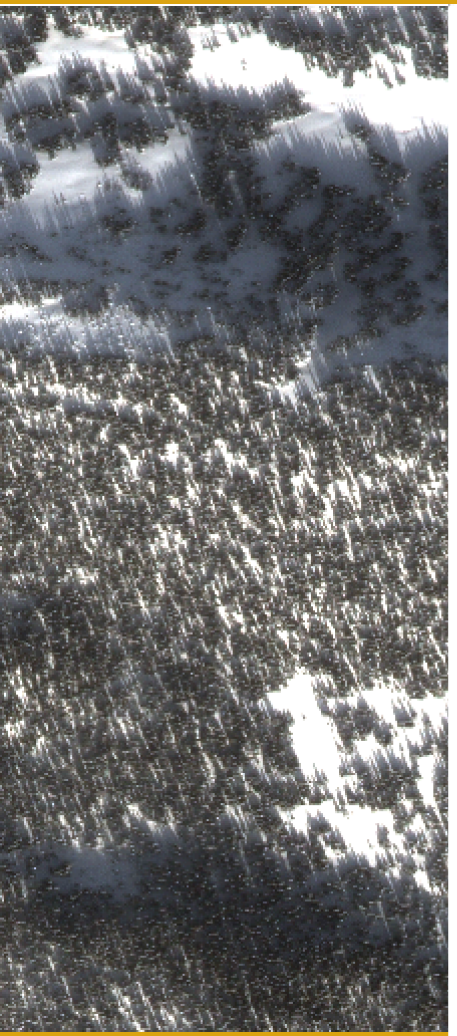
Response

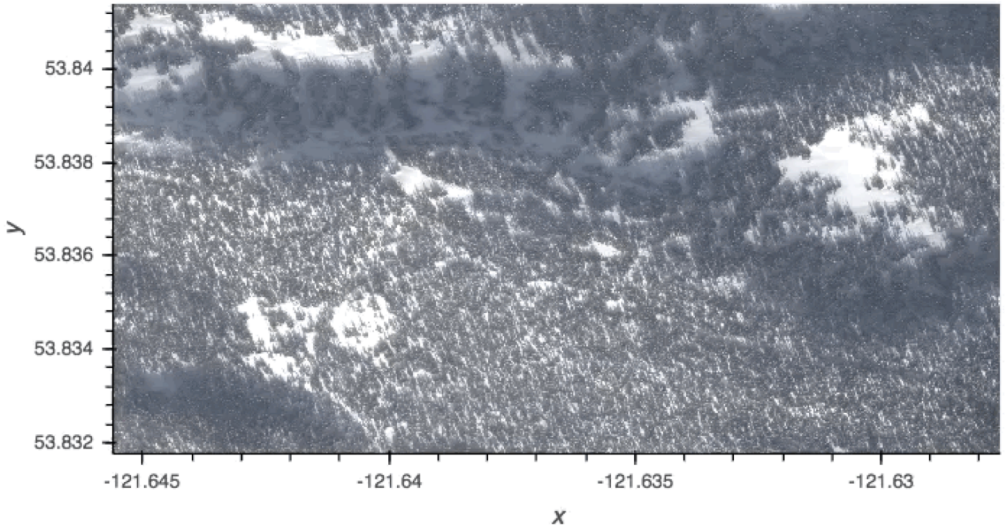
geo

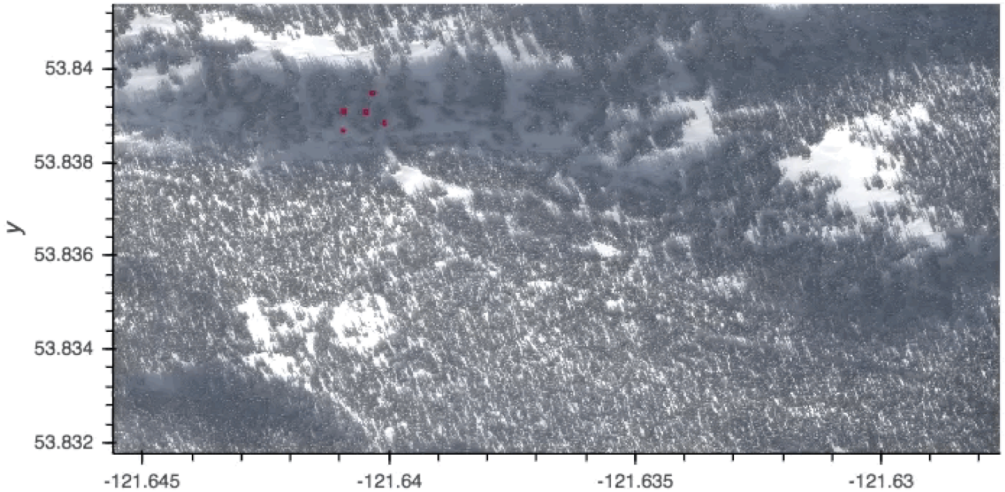
www.cogeo.org/map

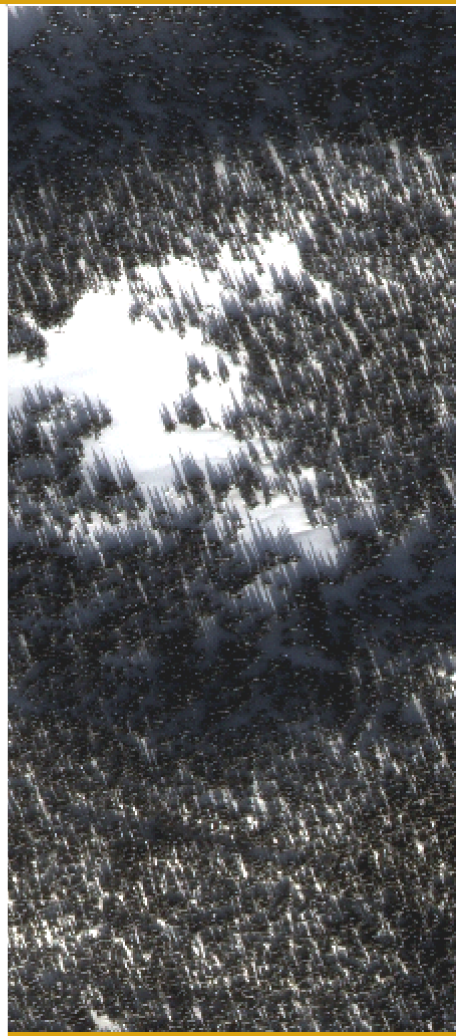


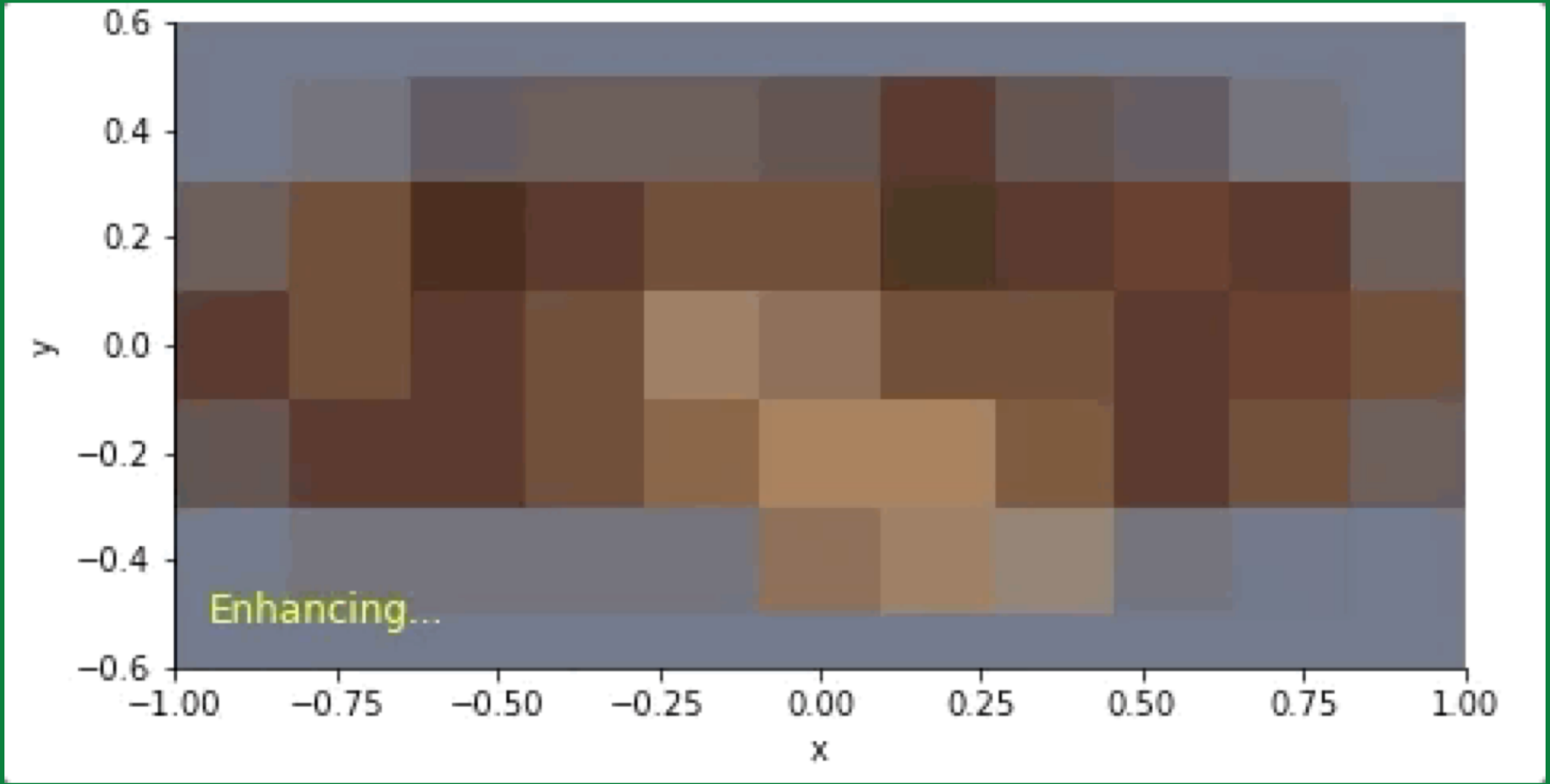
geo




53.84
53.838
53.836
53.834
53.832
y
-121.645
-121.64
-121.635
-121.63
x


53.84
53.838
53.836
53.834
53.832
y
-121.645
-121.64
-121.635
-121.63




0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
y
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
x
Enhancing...

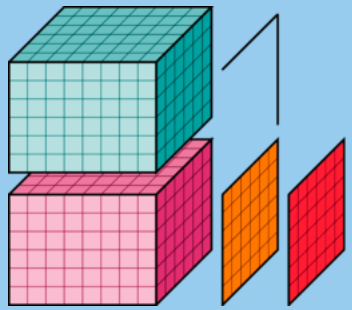
geo

geo

geo

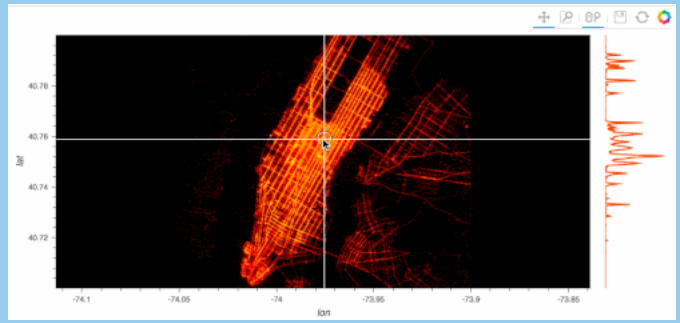
Tools for processing

xarray – Works great for n-dimensional data



xarray

Holoview – Easily visualize geospatial data



geo

File Edit View Insert Cell Kernel Help

Trusted Python 3

Import libraries and set extensions

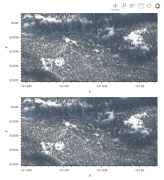
```
In [1]:
from PIL import Image
import matplotlib as mpl
import numpy as np
import os
import random
import rasterio
import warnings

%r.extension('bubah')
warnings.filterwarnings('ignore')
```

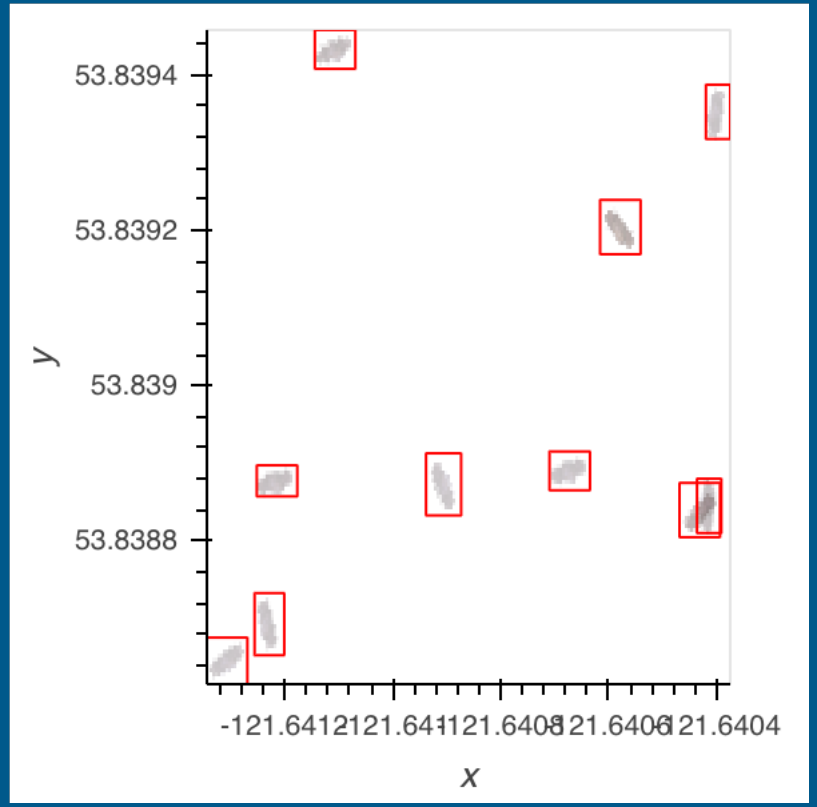
Resample

```
In [2]:
input_img = './sampatch.png'
img = Image.open(input_img)

factor = 2
resized = img.resize(
    (
        int(img.width/factor),
        int(img.height/factor)
    ),
    resample = Image.BICUBIC
)
resized.save('./sampatch_small_2.png')
```

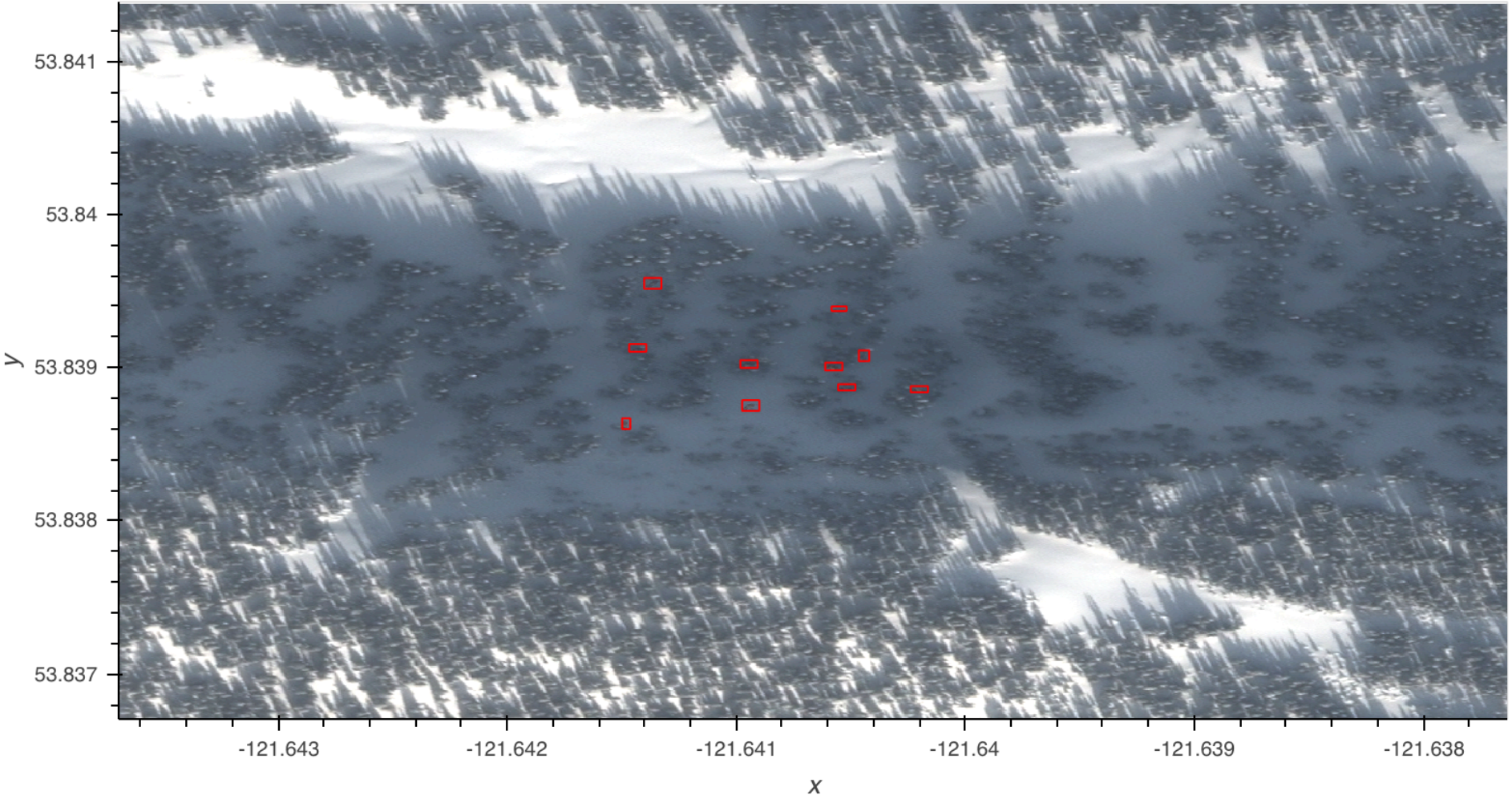
```
In [3]:
img = Image.open('./sampatch_small_2.png')

sampatches = []
boxes = []

num_sampatches = 10

for ind in range(num_sampatches):

    rotated = img.rotate(random.randint(0, 360), expand=True)

    # cropping of image + buffer
    x0, y0 = (
        -121.64 - (random.random() * 0.0013),
        53.8386 + (random.random() * 0.0013)
    )

    darkness = random.randint(50, 70)/100.0

    X = np.array(rotated)
    X[:,:,:1] = X[:,:,:1] * darkness
    X[:,:,:2] = X[:,:,:2] * darkness
    X[:,:,:3] = X[:,:,:3]
    X[:,:,:4] = X[:,:,:4] * darkness/2

    rgbs = r.RGB(
        X,
        bounds=
            x0=(rotated.width/2),
            y0=(rotated.height/2),
            x1=(rotated.width/2),
            y1=(rotated.height/2)
        )
    )
    rgbs.bounds = r.BoundingBox(
        point=(
            x0=(rotated.width/2)/2000000, y0=(rotated.height/2)/2000000,
            x1=(rotated.width/2)/2000000, y1=(rotated.height/2)/2000000
        )
    )

    sampatches.append(rgbs)

    bnds = rgbs.bounds
    cur = bnds.left()
    w = cur[1] - cur[0]
    h = cur[3] - cur[2]
    box = r.Box(x0, y0, w, h).opts(color='red')

    boxes.append(box)

r.Overlay(sampatches + boxes)
```

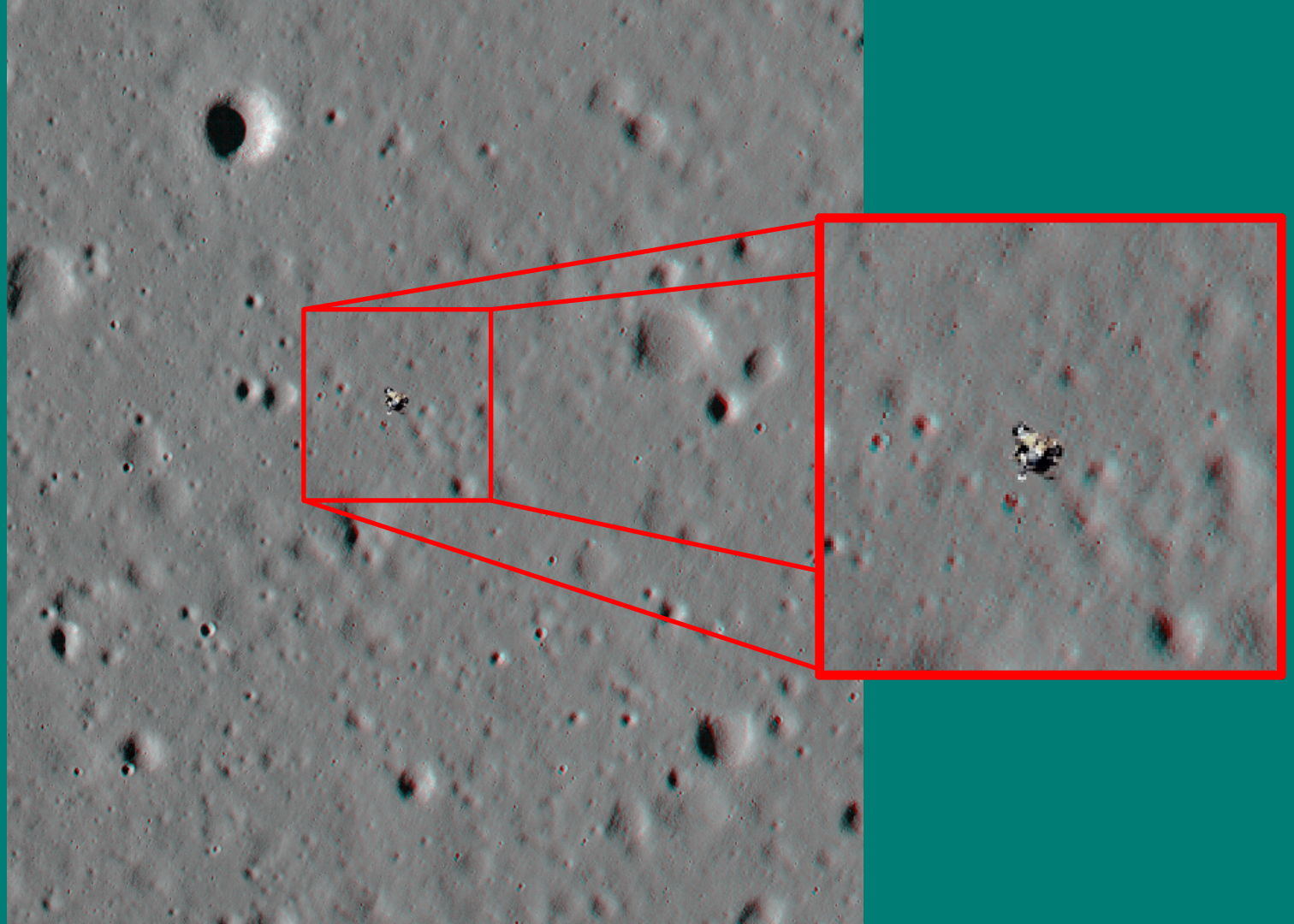
```
In [4]:
ash_img = './potasiy_obm_ancient.tif'
sing = rasterio.open_rasterio(ash_img)

def scale_band(data):
    return np.clip(data/np.quantile(data, 0.99), 0, 1*255).astype('uint8')

rgb = r.RGB(
    (
        sing[ 'r' ],
        sing[ 'g' ],
        scale_band(sing[1].data),
        scale_band(sing[1].data),
        scale_band(sing[1].data),
        np.ones(sing[ 'r' ].data.shape).astype('uint8' )*255
    )
    vdims=[sing[ 'RGBA' ]
)
.opts(
    frame_width = 500,
    width = int(sing.shape[ 1/2 ]),
    height = int(sing.shape[ 1/2 ])
)

hv.Layout([hv.Overlay([rgb] + sampatches), hv.Overlay([rgb] + sampatches + boxes)].cols(1))
```





geo


53.841
53.84
53.839
53.838
53.837
y
-121.643
-121.642
-121.641
-121.64
-121.639
-121.638
x



geo

geo

james@sparkgeo.com
@jr_banting