Chula
Chulalongkorn University

# Open Remote Sensing I

# LAB : Introduction to Google Earth Engine

2108-421 Modern Integrated Surveying Technology
Semester 2/2022

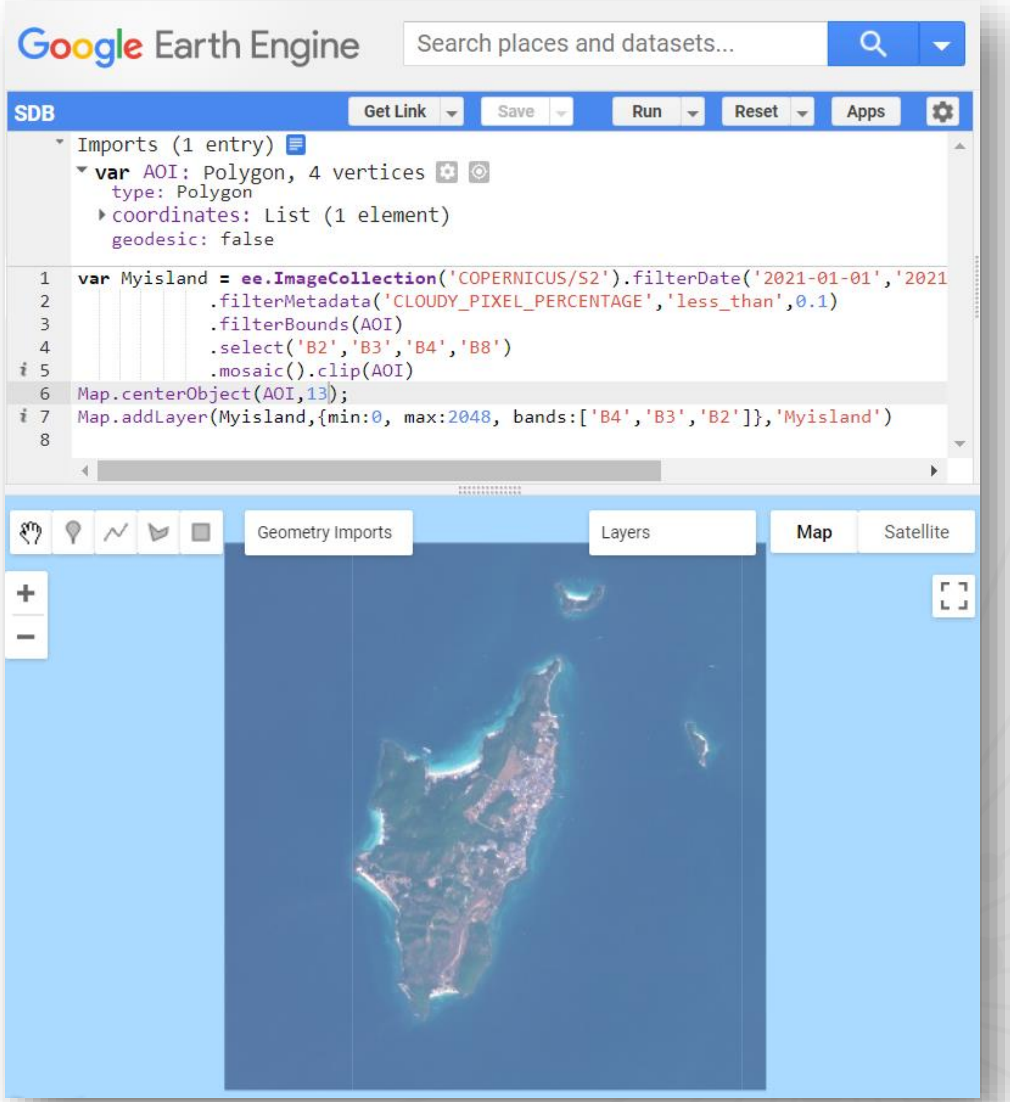
Thepchai Srinoi

(thepchaisrinoi@gmail.com)

Department of Survey Engineering Chulalongkorn University

Bangkok Thailand

- Google Earth Engine เป็นเครื่องมือในการสืบค้นภาพถ่ายทางดาวเทียมและข้อมูลภูมิสารสนเทศอันมหาศาล นอกจากนี้ยังสามารถประมวลผล วิเคราะห์ และสร้างเป็นภูมิสารสนเทศใหม่ๆ ได้ผ่านทางระบบออนไลน์




Google Earth Engine with Java programming

https://code.earthengine.google.com/

สำหรับ python programming สามารถทำได้ผ่าน Google Colab

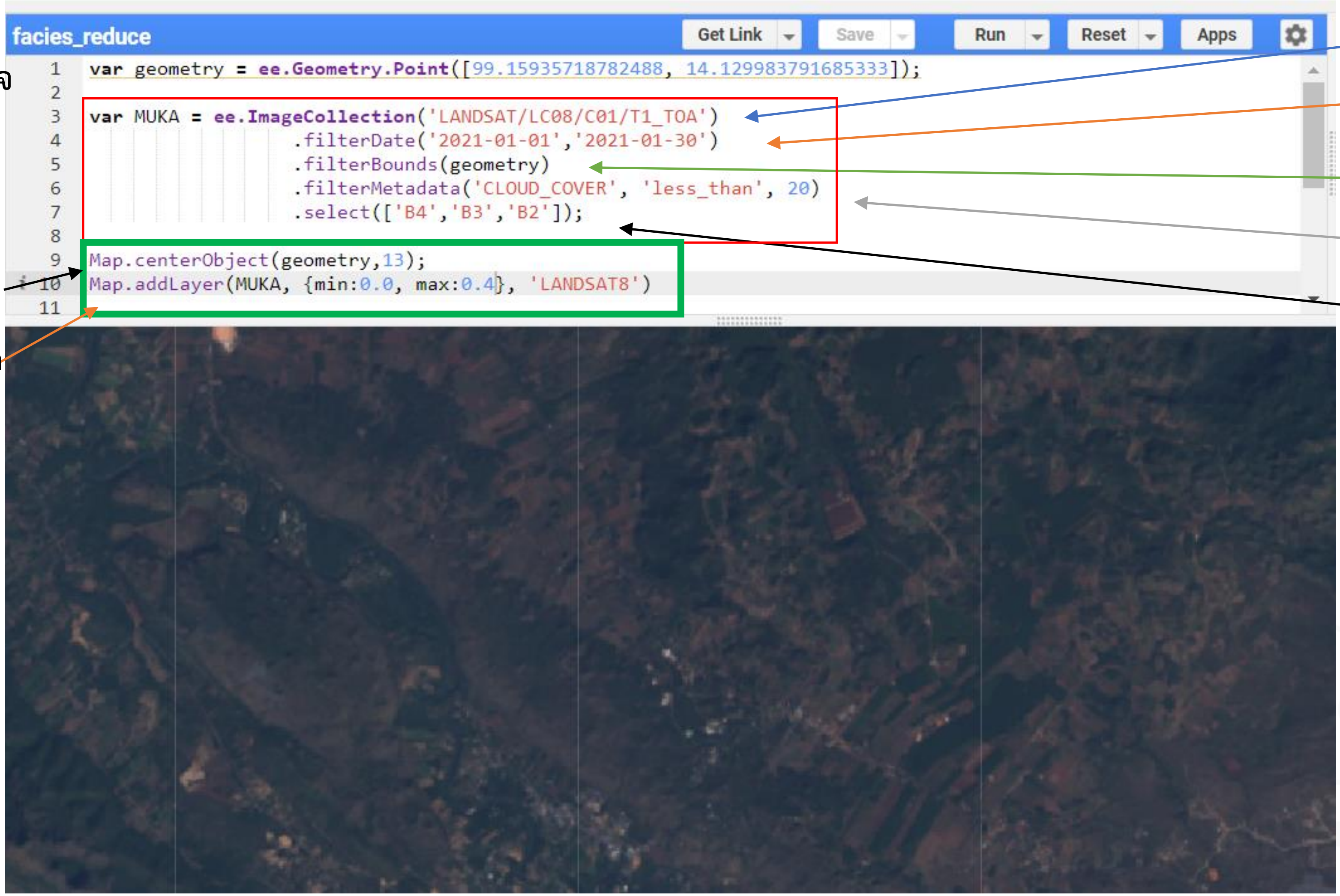
https://colab.research.google.com/

Chula
Chulalongkorn University

บริเวณที่สนใจ

กำหนดศูนย์กลางภาพ

กำหนดแสดงชั้นข้อมูล




ค้นจากชื่อดาวเทียมและระบบภาพ

ค้นวันที่ทำการถ่ายภาพ

ค้นครอบคลุมบริเวณที่สนใจ

ค้นจากสัดส่วนเมฆบนท้องฟ้า

เลือก R G B band : True Color

LANDSAT-8

VISUALIZATION

TRUE COLOR

COMPOSITE

Chula
Chulalongkorn University

# Harmonized Sentinel-2 MSI: MultiSpectral Instrument, Level-2A

Description | Bands | Image Properties | Terms of Use

**Image Properties**

| Name | Type | Description |
|---|---|---|
| AOT_RETRIEVAL_ACCURACY | DOUBLE | Accuracy of Aerosol Optical thickness model |
| CLOUDY_PIXEL_PERCENTAGE | DOUBLE | Granule-specific cloudy pixel percentage taken from the original metadata |
| CLOUD_COVERAGE_ASSESSMENT | DOUBLE | Cloudy pixel percentage for the whole archive that contains this granule. Taken from the original metadata |
| CLOUDY_SHADOW_PERCENTAGE | DOUBLE | Percentage of pixels classified as cloud shadow |
| DARK_FEATURES_PERCENTAGE | DOUBLE | Percentage of pixels classified as dark features or shadows |


```
var image = ee.ImageCollection('LANDSAT/LC08/C01/T1_TOA')
                  .filterDate('2021-01-01','2021-01-30')
                  .filterBounds(POI)
                  .filterMetadata('CLOUD_COVER', 'less_than', 40)
                  .select(['B2','B3','B4','B5','B6','B7']);

Map.centerObject(POI,11);
Map.addLayer(image, {min:0.0, max:0.4, bands:['B4','B3','B2']}, 'LANDSAT8 432')
```

# USGS Landsat 9 Level 2, Collection 2, Tier 1

Description | Bands | Image Properties | Terms of Use

**Image Properties**

| Name | Type | Description |
|---|---|---|
| ALGORITHM_SOURCE_SURFACE_REFLECTANCE | STRING | Name and version of the surface reflectance algorithm. |
| ALGORITHM_SOURCE_SURFACE_TEMPERATURE | STRING | Name and version of the surface temperature algorithm. |
| CLOUD_COVER | DOUBLE | Percentage cloud cover (0-100), -1 = not calculated. |
| CLOUD_COVER_LAND | DOUBLE | Percentage cloud cover over land (0-100), -1 = not calculated. |
| COLLECTION_CATEGORY | STRING | Scene collection category, "T1" or "T2". |

```
var image = ee.ImageCollection('LANDSAT/LC08/C01/T1_TOA')
                  .filterDate('2021-01-01','2021-01-30')
                  .filterBounds(POI)
                  .filterMetadata('CLOUD_COVER', 'less_than', 10)
                  .select(['B2','B3','B4','B5','B6','B7']);

Map.centerObject(POI,11);
Map.addLayer(image, {min:0.0, max:0.4, bands:['B4','B3','B2']}, 'LANDSAT8 432')
```

# LANDSAT-8

# VISUALIZATION

## TRUE AND FALSE COLOR COMPOSITE

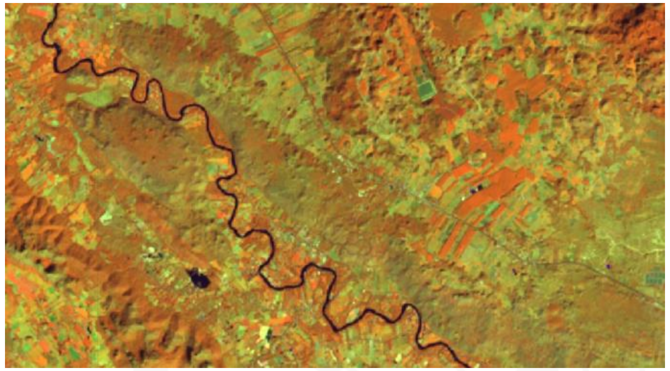
```
var geometry = ee.Geometry.Point([99.15935718782488, 14.129983791685333]);

var MUKA = ee.ImageCollection('LANDSAT/LC08/C01/T1_TOA')
                  .filterDate('2021-01-01','2021-01-30')
                  .filterBounds(geometry)
                  .filterMetadata('CLOUD_COVER', 'less_than', 20)

Map.centerObject(geometry,13);
Map.addLayer(MUKA, {min:0.0, max:0.4, bands:['B4','B3','B2']}, 'LANDSAT8 432')
Map.addLayer(MUKA, {min:0.0, max:0.4, bands:['B5','B6','B4']}, 'LANDSAT8 564')
Map.addLayer(MUKA, {min:0.0, max:0.4, bands:['B5','B4','B3']}, 'LANDSAT8 543')
```

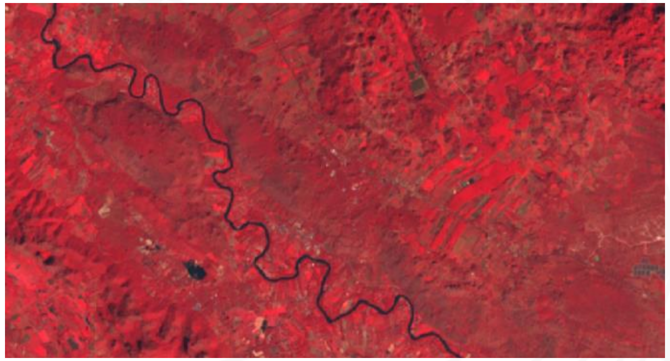
ชื่อตัวแปรเก็บภาพ

ตัวแปรบอกการมองเห็น

Min Max เป็นไปตามค่า Digital Number

ชื่อชั้นข้อมูล





# Google Earth Engine : NDVI

**NDVI_Pakkret ***   Get Link   Save   Run   Reset   Apps

```
Imports (1 entry)
 var geometry: Point (100.51, 13.91)
// Image Collecting
var myimage = ee.ImageCollection('COPERNICUS/S2_SR')
                .filterDate('2022-1-1','2022-1-10')
                .filterBounds(geometry)
                .mean()
// Processing vegetable index
var B8_img = (myimage.select(['B8']))
var B4_img = (myimage.select(['B4']))
var vi = B8_img.subtract(B4_img)
var ai = B8_img.add(B4_img)
var ndvi = vi.divide(ai)
// Visualization
Map.centerObject(geometry,14)

var VIZ = {min:0, max:2000, bands:['B4','B3','B2']} ;
Map.addLayer(myimage,VIZ,'S2_T')

var viz = {min:-1, max:1, palette:['red','orange','yellow','green']} ;
Map.addLayer(ndvi,viz,'ndvi')
```

**SENTINEL-2 L2A BOTTOM OF ATMOSPHERE**

$$NDVI = \frac{\rho_{NIR} - \rho_{red}}{\rho_{NIR} + \rho_{red}}$$

NDVI values range from -1 to 1

BAND MATH ลูกทุ่งๆ


แบบนี้ก็ได้ ยาวหน่อย

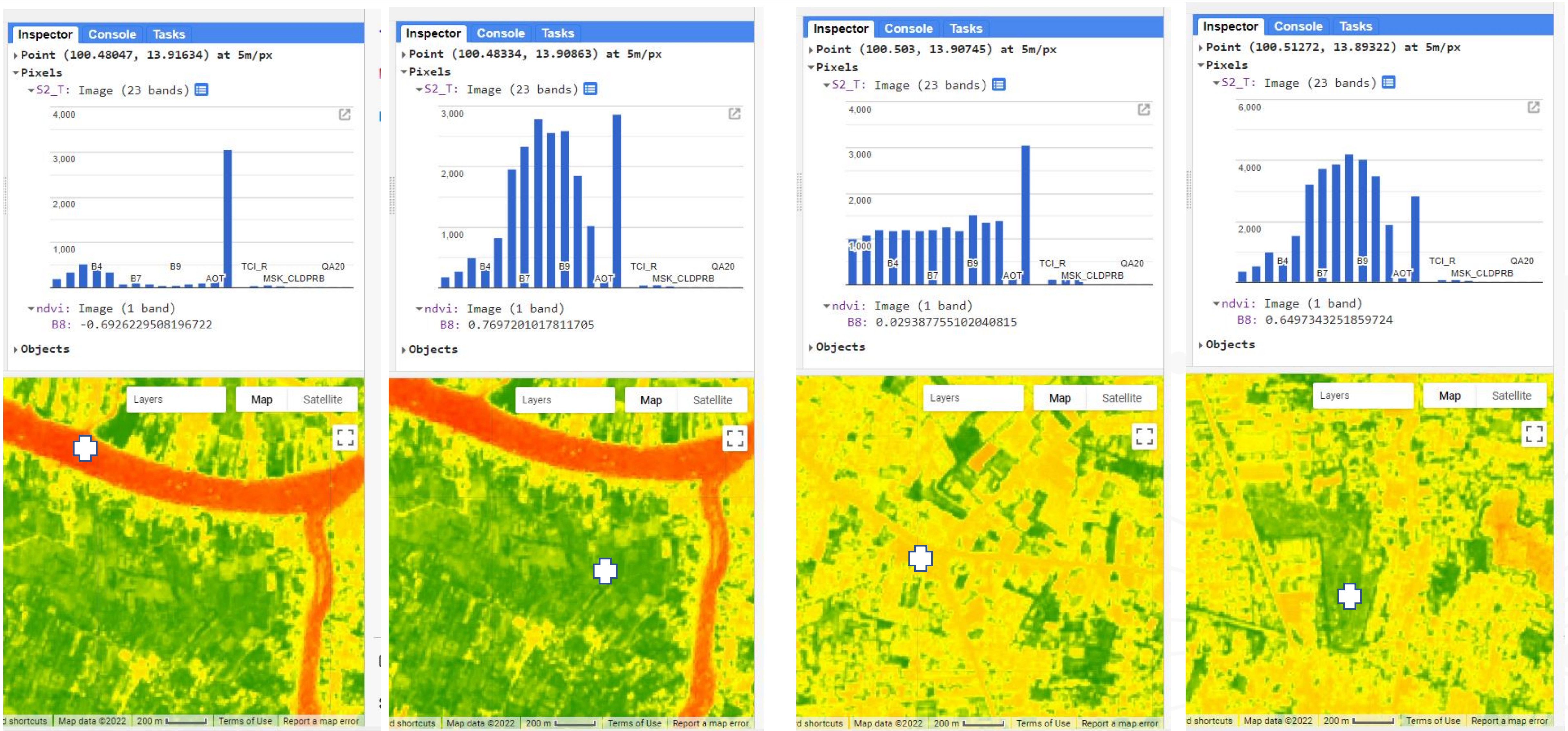
```
// Compute the Normalized Difference Vegetation Index (NDVI).
var nir = image.select('B5');
var red = image.select('B4');
var ndvi = nir.subtract(red).divide(nir.add(red)).rename('NDVI');
```


บรรทัดเดียวแบบคนฉลาด

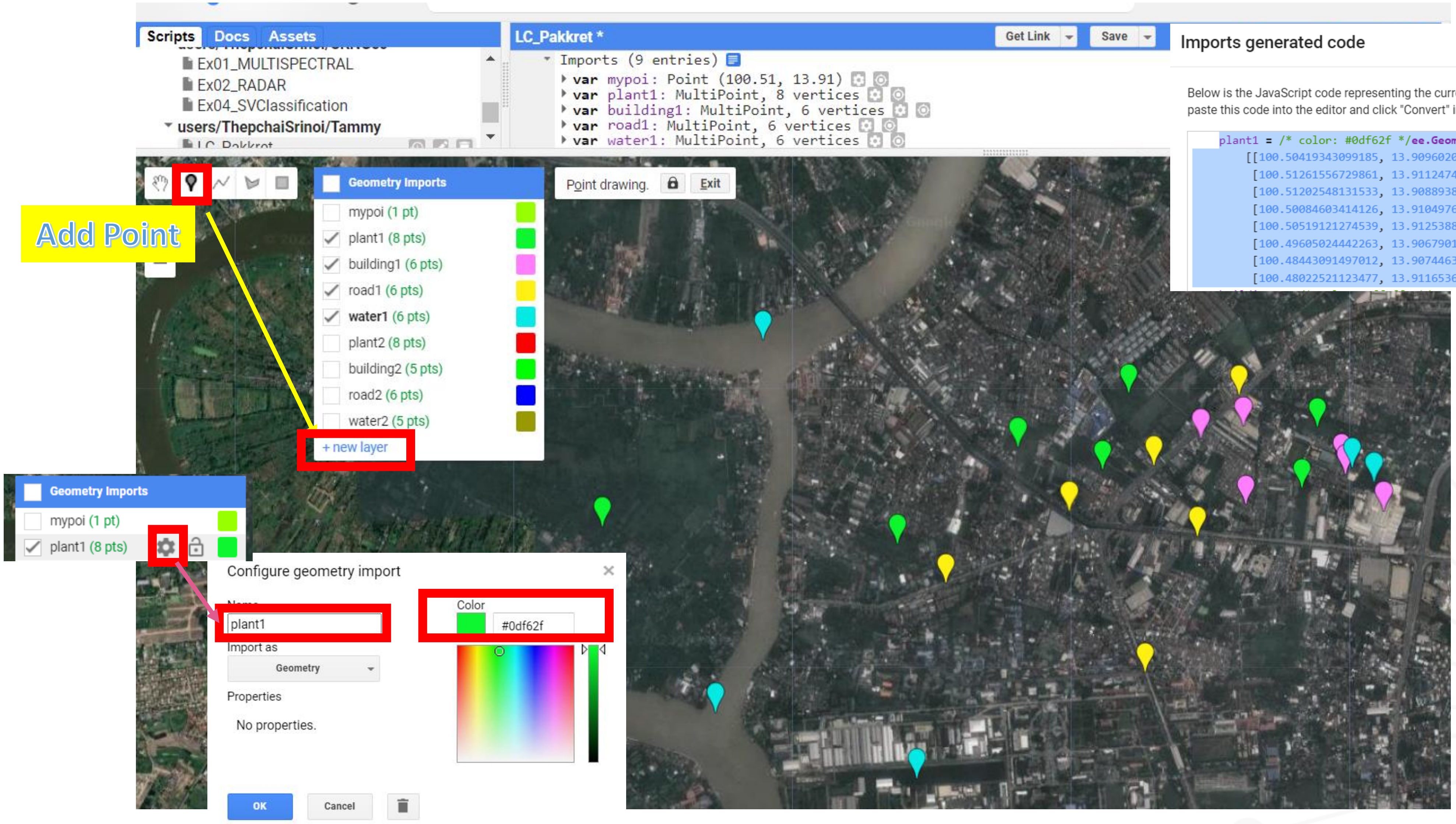
```
var ndvi = image.normalizedDifference(['B5', 'B4']).rename('NDVI');
```

# Google Earth Engine : Spectral Reflectance Inspection

Chula
Chulalongkorn University



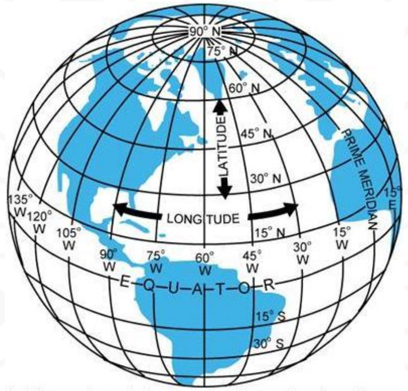
# Google Earth Engine : Add Geometry (Point)




X – Longitude

Y – Latitude

EPSG : 4326



# Google Earth Engine : Image Classification (Supervised Classification)

Chula
Chulalongkorn University

LC_Pakkret *   Get Link   Save   Run   Reset   Apps

```
var plant1: MultiPoint, 8 vertices
var building1: MultiPoint, 6 vertices
var road1: MultiPoint, 6 vertices
var water1: MultiPoint, 6 vertices
var plant2: MultiPoint, 8 vertices
var building2: MultiPoint, 5 vertices
var road2: MultiPoint, 6 vertices
var water2: MultiPoint, 5 vertices
// Supervised Classification - Machine Learning
//Satellite Image Data
var image = ee.ImageCollection('COPERNICUS/S2_SR')
                 .filterDate('2022-01-01', '2022-01-10')
                 .filterBounds(mypoi)
                 .mean()

var bands = ['B2','B3','B4','B8','B12'];

//Training Data
var trainPOI = ee.FeatureCollection([ee.Feature(plant1, {'class': 0}),
                      ee.Feature(building1, {'class': 1}),
                      ee.Feature(road1, {'class': 2}),
                      ee.Feature(water1, {'class': 3}),
                      ]);
var testPOI = ee.FeatureCollection([ee.Feature(plant2, {'class': 0}),
                      ee.Feature(building2, {'class': 1}),
                      ee.Feature(road2, {'class': 2}),
                      ee.Feature(water2, {'class': 3}),
                      ]);

var training = image.sampleRegions({collection: trainPOI,properties: ['class'],scale:10});
var validation = image.sampleRegions({collection: testPOI,properties: ['class'],scale:10});

//Classifier train your training data.
// Classification and Regression Trees (CART) classifier
var trained = ee.Classifier.smileCart().train(training, 'class', bands);

//Classifier classify your area.
var classified = image.select(bands).classify(trained);

Map.centerObject(mypoi, 13);
Map.addLayer(image, {bands: ['B4', 'B3', 'B2'], max: 4096}, 'truecolor');
Map.addLayer(classified,
          {min: 0, max: 3, palette: ['green','yellow','red','blue']},
          'classification');
```

ดึงภาพ

กำหนดข้อมูล TRAIN

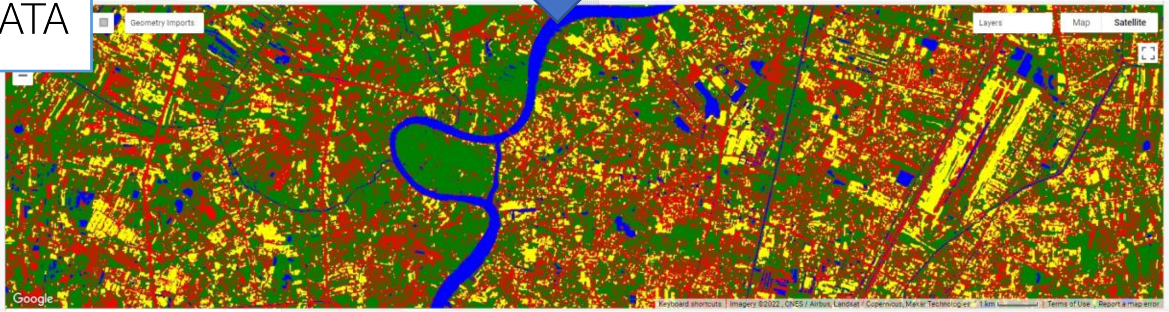
กำหนดข้อมูล TEST

ดึงค่าการสะท้อนเข้าจุด

TRAIN THE DATA

LET’S GO, SOLO!!


เลือกบริเวณ Training data
นำค่าการสะท้อนเข้า data
(feature … class)
สอนตัวจำแนก
สั่งตัวจำแนกจัดการทั้งภาพ !!!!!



# Google Earth Engine : Welcome to SAR HANAKAAAA

Google Earth Engine | Search places and datasets... | seniorproject2022

Scripts | Docs | Assets

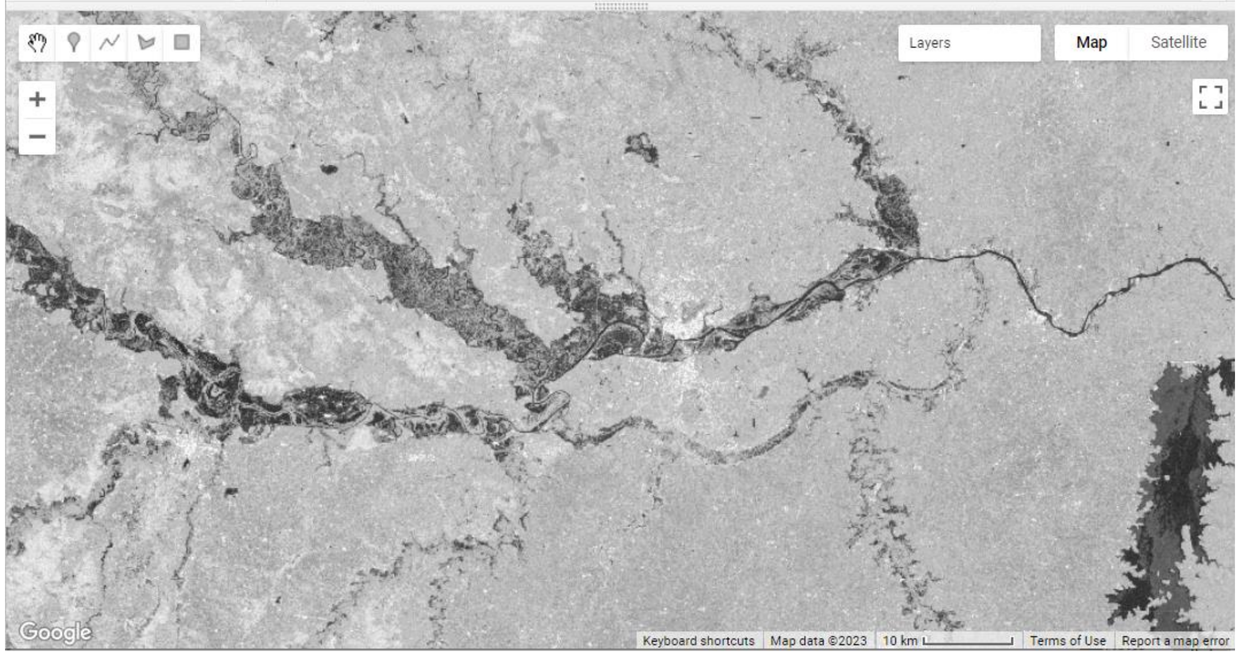
- LTE Explain
- SENTINEL
- SV CLASSIFICAT...

Thaiflood * | Get Link | Save | Run | Reset | Apps

```
// POINT OF INTEREST (POI)
var POI = ee.Geometry.Point([104.79165, 15.23193]);
// LOADING A SENTINEL-1
var pawan = ee.ImageCollection('COPERNICUS/S1_GRD')
                  .filterBounds(POI)
                  .filter(ee.Filter.listContains('transmitterReceiverPolarisation', 'VV'))
                  .filter(ee.Filter.eq('instrumentMode', 'IW'))
                  .filter(ee.Filter.eq('orbitProperties_pass', 'DESCENDING'))
                  .filterDate('2022-10-1', '2022-10-10')
                  .select('VV')

var vizParams = {min:-30, max:0};
Map.setCenter(104.79165, 15.23193, 10);
Map.addLayer(pawan, vizParams, 'Sentinel-1 VV Before flood');
```

**Table 2.** Average backscatter (dB) in different water features

| Water Features | Backscatter (dB) | | | |
|---|---|---|---|---|
| | HH | HV | VH | VV |
| Flood Water | −8 to −12 | −15 to −24 | −15 to −24 | −6 to −15 |
| River Water | −16 to −30 | −24 to −36 | −24 to −36 | −19 to −32 |
| Tank Water | −13 to −26 | −22 to −40 | −22 to −40 | −16 to −28 |
| Oxbow Lake | −16 to −24 | −21 to −32 | −21 to −32 | −24 to −32 |
| Partially Submerged Features | −18 to −30 | −24 to −34 | −24 to −34 | −8 to −18 |

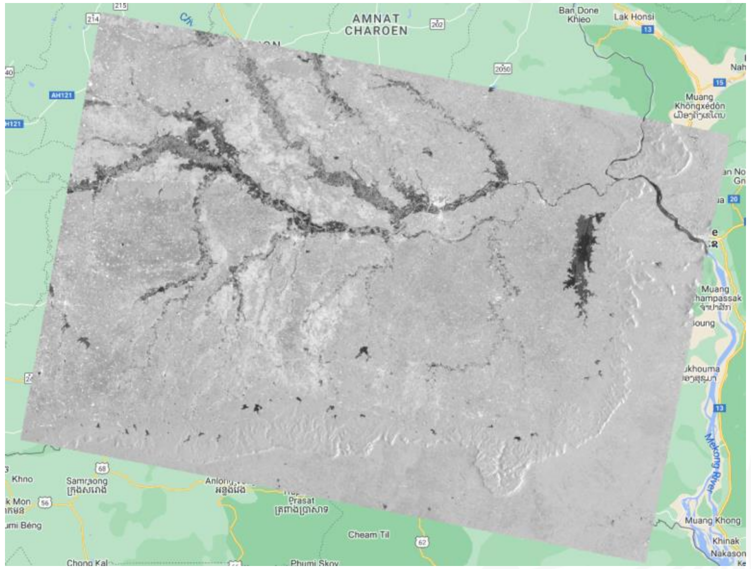



## Metadata and Filtering

To create a homogeneous subset of Sentinel-1 data, it will usually be necessary to filter the collection using metadata properties. The common metadata fields used for filtering include these properties:

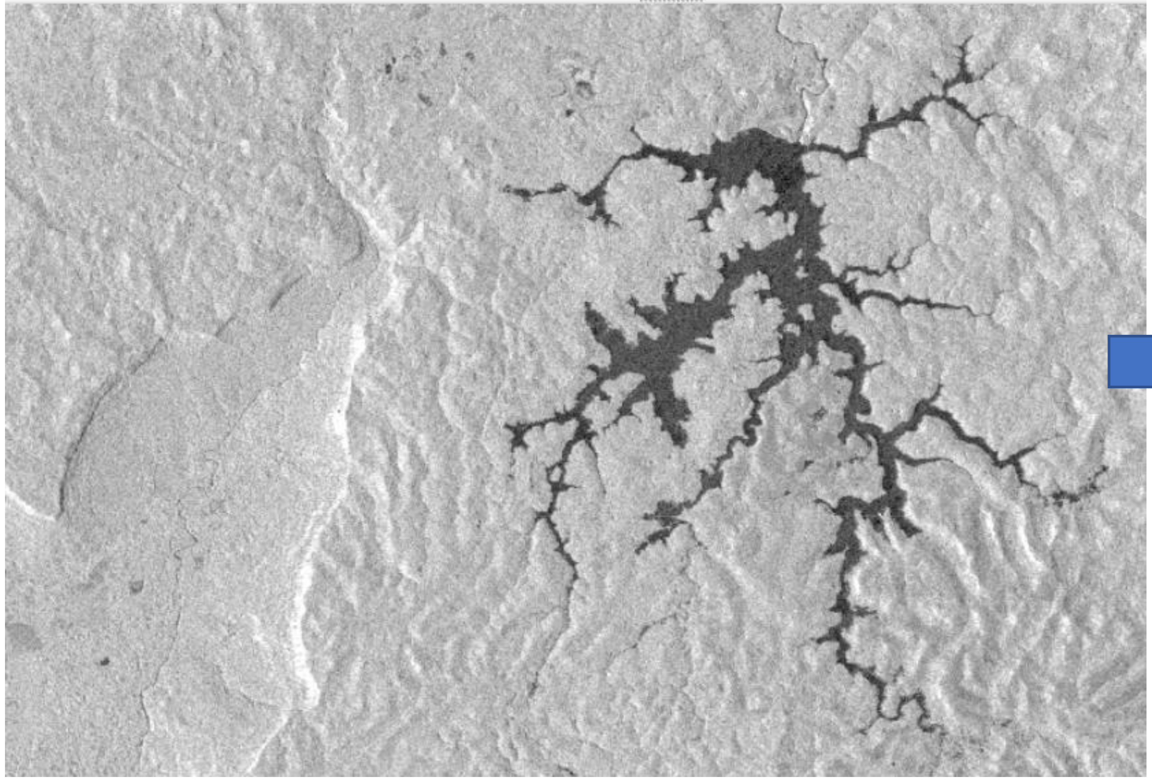
1. `transmitterReceiverPolarisation` : ['VV'], ['HH'], ['VV', 'VH'], or ['HH', 'HV']
2. `instrumentMode` : 'IW' (Interferometric Wide Swath), 'EW' (Extra Wide Swath) or 'SM' (Strip Map). See this reference for details.
3. `orbitProperties_pass` : 'ASCENDING' or 'DESCENDING'
4. `resolution_meters` : 10, 25 or 40
5. `resolution` : 'M' (medium) or 'H' (high). See this reference for details.

ภาพเต็มที่ให้มา

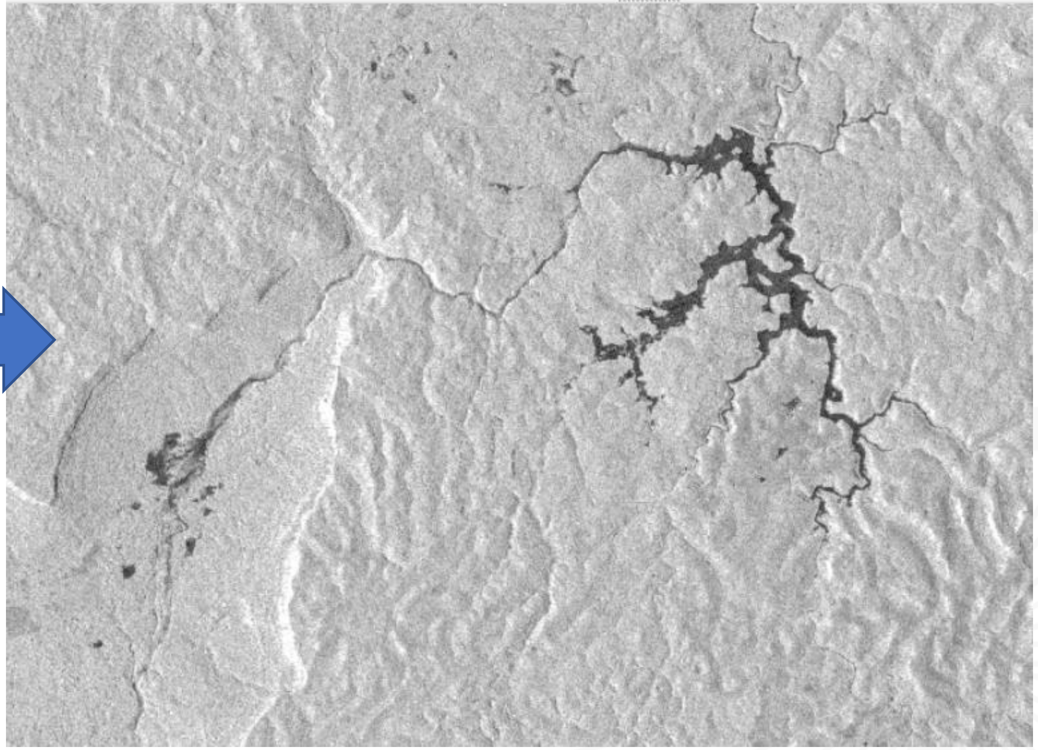


the Xe Namnoy Xe Pian Dam in the Attapeu province of Laos

```
var before = ee.ImageCollection('COPERNICUS/S1_GRD')
               .filterBounds(POI)
               .filter(ee.Filter.listContains('transmitterReceiverPolarisation', 'VV'))
               .filter(ee.Filter.eq('instrumentMode', 'IW'))
               .filter(ee.Filter.eq('orbitProperties_pass', 'ASCENDING'))
               .filterDate('2018-7-10', '2018-7-15')
               .select('VV')
```



```
// LOADING A SENTINEL-1
var after = ee.ImageCollection('COPERNICUS/S1_GRD')
               .filterBounds(POI)
               .filter(ee.Filter.listContains('transmitterReceiverPolarisation', 'VV'))
               .filter(ee.Filter.eq('instrumentMode', 'IW'))
               .filter(ee.Filter.eq('orbitProperties_pass', 'ASCENDING'))
               .filterDate('2018-7-25', '2018-7-30')
               .select('VV')
```



Chula
Chulalongkorn University


Thepchai Srinoi

(thepchaisrinoi@gmail.com)

Department of Survey Engineering Chulalongkorn University

Bangkok Thailand