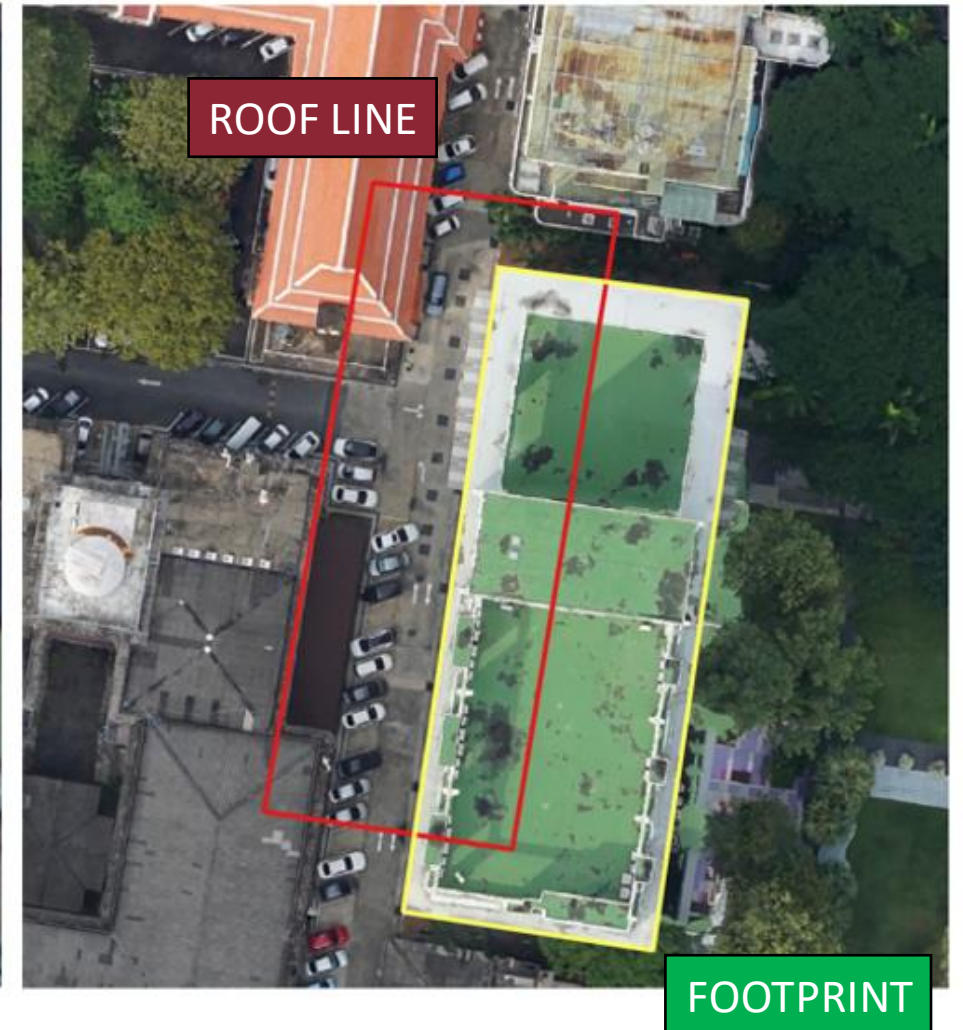
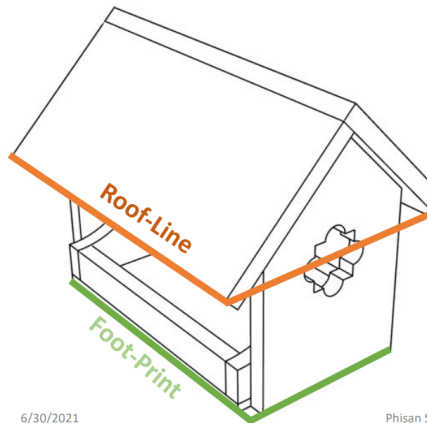


BUILDING FOOTPRINT

*2108373 Advanced Photogrammetry
Semester 2/2023*

*Thepchai Srinoi
Department of Survey Engineering
Chulalongkorn University*







Microsoft Building Footprint

ใน กทม. 1.1 ล้านกลุ่มอาคาร



Google Open Buildings V2

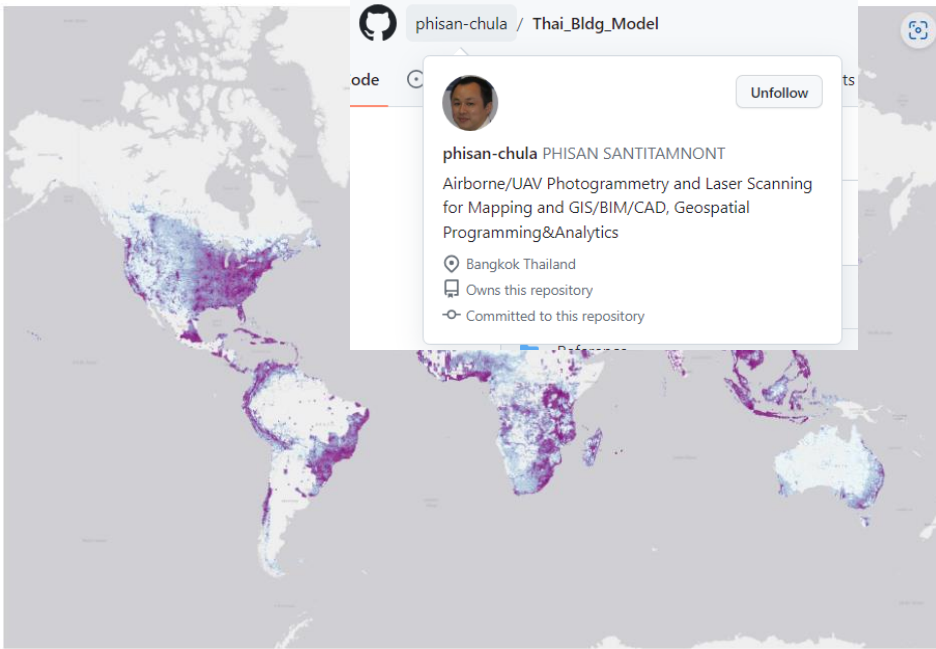
ใน กทม. 2.4 ล้านกลุ่มอาคาร

(Confidence ≥ 0.7) อยู่ที่ 1.6 ล้านกลุ่มอาคาร

<https://github.com/microsoft/GlobalMLBuildingFootprints>

README.md

Regions included



phisan-chula / Thai_Bldg_Model


phisan-chula PHISAN SANTITAMNONT
Airborne/UAV Photogrammetry and Laser Scanning for Mapping and GIS/BIM/CAD, Geospatial Programming&Analytics
Bangkok Thailand
Owns this repository
Committed to this repository

You can download the layer above as GeoJSON [here](#).

```
{
  "type": "FeatureCollection",
  "crs": { "type": "name", "properties": { "name": "urn:ogc:def:crs:OGC:1.3:CRS84" } },
  "features": [
    { "type": "Feature", "properties": { "quadkey": "3022233233", "building_count": 63461 }, "geometry": { "type": "Polygon", "coordinates": [ [ [ -85.25390625, 41.112468789180852 ], [ -85.25390625, 40.979898069620127 ], [ -85.078125, 41.112468789180852 ], [ -85.25390625, 41.112468789180852 ] ] ] ] },
    { "type": "Feature", "properties": { "quadkey": "2301023203", "building_count": 22092 }, "geometry": { "type": "Polygon", "coordinates": [ [ [ -122.16796875, 37.020098201368114 ], [ -122.16796875, 36.879620605026759 ], [ -121.9921875, 37.020098201368114 ], [ -122.16796875, 37.020098201368114 ] ] ] ] },
    { "type": "Feature", "properties": { "quadkey": "2311231222", "building_count": 64052 }, "geometry": { "type": "Polygon", "coordinates": [ [ [ -97.03125, 33.284619968887682 ], [ -97.03125, 33.137551192346145 ], [ -96.85546875, 33.284619968887682 ], [ -97.03125, 33.284619968887682 ] ] ] ] },
    { "type": "Feature", "properties": { "quadkey": "3022233233", "building_count": 63461 }, "geometry": { "type": "Polygon", "coordinates": [ [ [ -85.25390625, 41.112468789180852 ], [ -85.25390625, 40.979898069620127 ], [ -85.078125, 41.112468789180852 ], [ -85.25390625, 41.112468789180852 ] ] ] ] }
  ]
}
```

https://github.com/phisan-chula/Thai_Bldg_Model

phisan-chula/Thai_Bldg_Model: A research on Thai 3D buiding model



#Research on Thai Building Modelling

| | Name | AreaKm2 | NumBldg | Size | LINK |
|---|--------------------|---------|-----------|----------|----------------------|
| 0 | Amnat Charoen | 3,371.2 | 183,873 | 76.9 MB | link |
| 1 | Ang Thong | 982.7 | 112,731 | 46.6 MB | link |
| 2 | Bangkok Metropolis | 1,611.2 | 1,101,690 | 548.3 MB | link |
| 3 | Buang Kan | 4,204.9 | 149,759 | 57.2 MB | link |

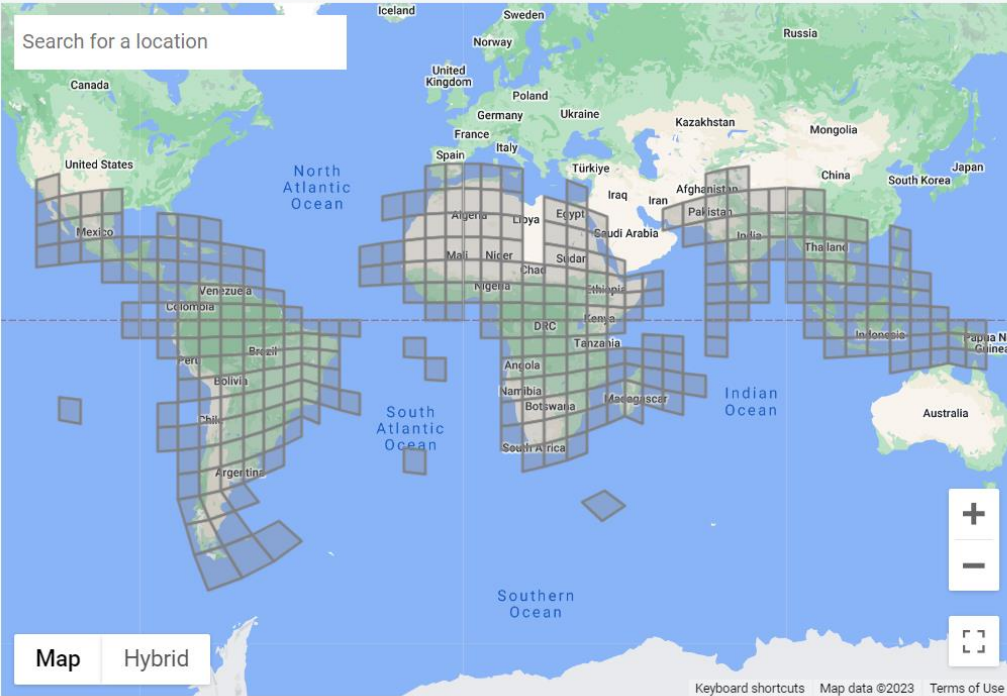
Geopackage
(Footprint and Centroid)

Open Buildings

https://sites.research.google/open-buildings/#download

Download from the map

To manually download polygons data for a specific cell, click on the map below.



Download polygons or points data for a specific country or region

This [Colab notebook](#) shows how data can be downloaded for a specific country or region.

Open Buildings - download region polygons or points. - Colaboratory

https://colab.research.google.com/github/google-research/google-research/blob/master/building_detection/open_buildings_download_region_...

Open Buildings - download region polygons or points.

File Edit View Insert Runtime Tools Help

+ Code + Text Copy to Drive

[] # limitations under the License

▼ Step 1. Prepare a compressed CSV file using [Open Buildings](#) data [takes 1-15 minutes depending on the region]

First, select a region from either the [Natural Earth low res](#) (fastest), [Natural Earth high res](#) or [World Bank high res](#) shapefiles:

region_border_source: Natural Earth (Low Res 110m)

region: [GHA](#) (Ghana)

or specify an area of interest in [WKT format](#) (assumes crs="EPSG:4326"); this [tool](#) might be useful.

your_own_wkt_polygon: " Insert text here "

Select type of data to download here:

data_type: polygons

[Show code](#)

Wicket

Wicket is a lightweight JavaScript library that reads and writes Well-Known Text (WKT) strings. It can also be extended to parse and to create geometric objects from various mapping frameworks, such as Leaflet, the ESRI ArcGIS JavaScript API, and the Google Maps API.

POLYGON((100.52208747092041 13.743220086564374,100.54706420126709 13.743220086564374,100.54706420126709 13.724793847040369,100.52208747092041 13.724793847040369,100.52208747092041 13.743220086564374))

Format for URLs Clear Map Map It!

Home Contact Me "Fork me on GitHub"

Design © 2012-2013 K. Arthur Endrey
Wicket is released under the GPL v3

segment-geospatial

Segmenting remote sensing imagery with text prompts and the Segment Anything Model (SAM)

This notebook shows how to generate object masks from text prompts with the Segment Anything Model (SAM).

Make sure you use GPU runtime for this notebook. For Google Colab, go to Runtime -> Change runtime type and select GPU as the hardware accelerator.

segment-geospatial

Home

Installation

Usage

Contributing

FAQ

Changelog

Report Issues

Examples

Satellite

Automatic mask generator

Automatic mask generator hq

text_prompts.ipynb - Colaboratory

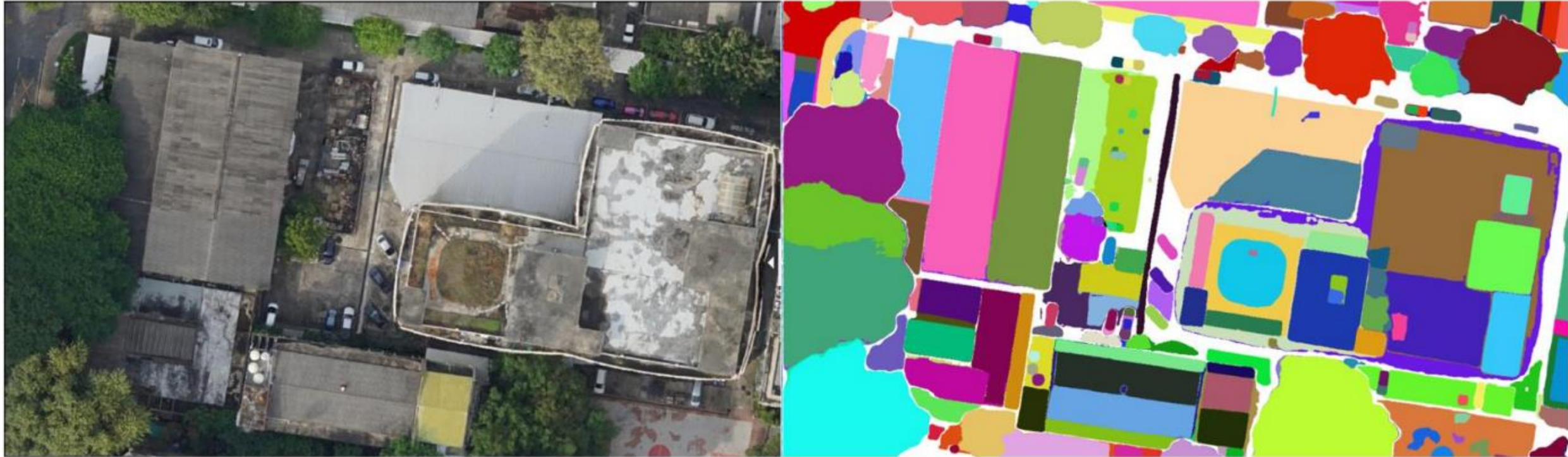
Segmenting remote sensing imagery with text prompts and the Segment Anything Model (SAM)

This notebook shows how to generate object masks from text prompts with the Segment Anything Model (SAM).

Make sure you use GPU runtime for this notebook. For Google Colab, go to Runtime -> Change runtime type and select GPU as the hardware accelerator.

```
chula = '/content/chula_engineering.tiff'
text_prompt = "building"
bt = 0.24
tt = 0.2
lang_sam.predict(chula, text_prompt, box_threshold=bt, text_threshold=tt)
```





True Orthophoto



Microsoft



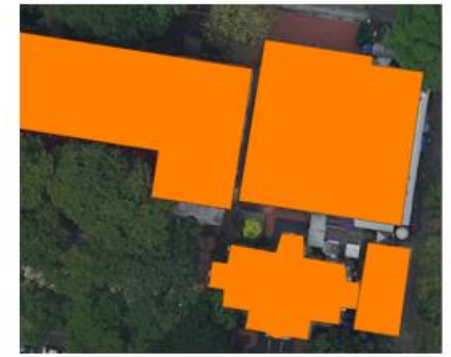
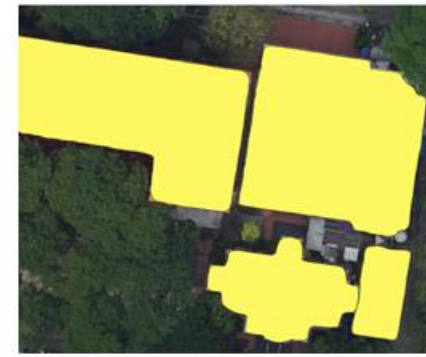
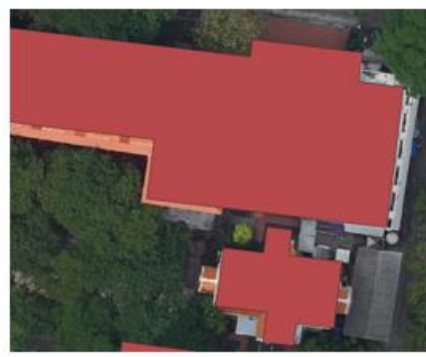
Google

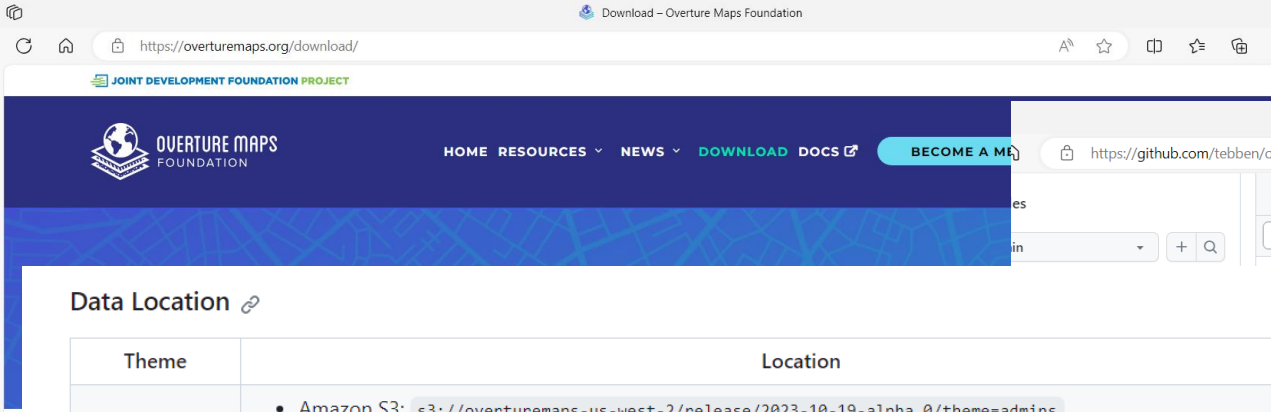


GeoSAM



Delineation





Download - Overture Maps Foundation

https://overturemaps.org/download/

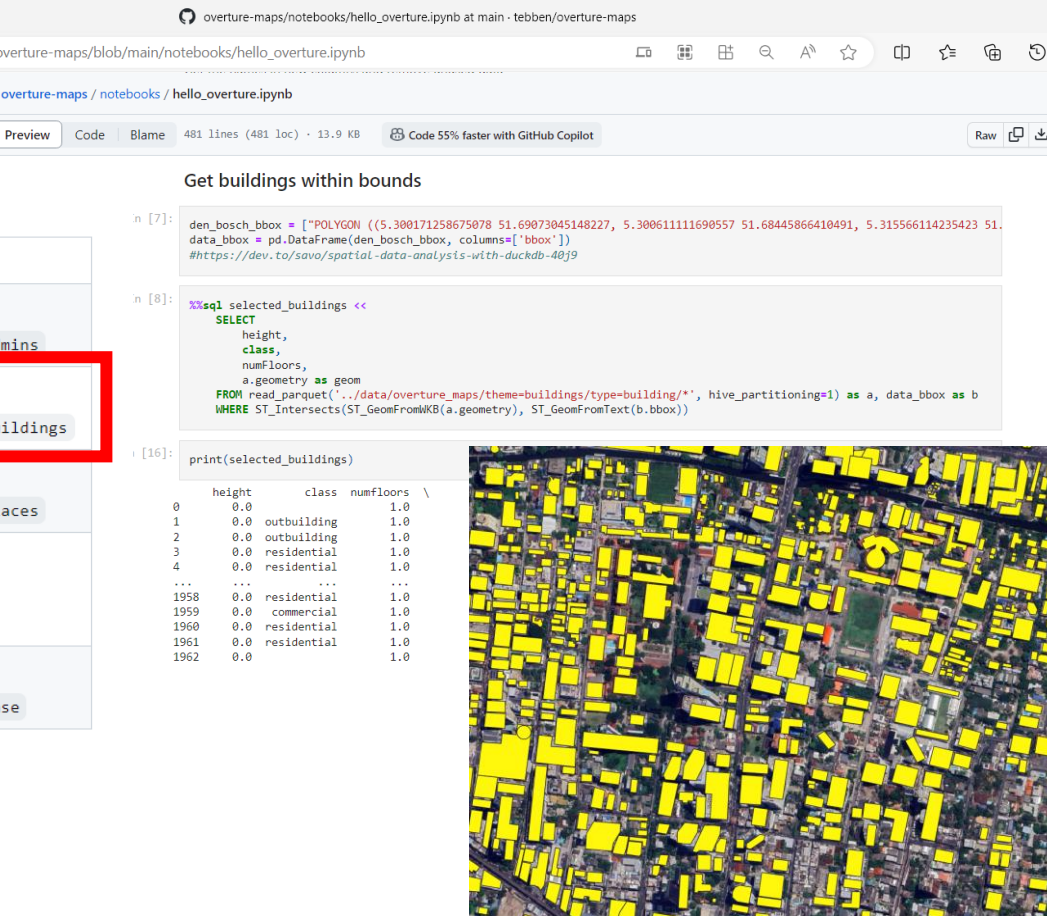
JOINT DEVELOPMENT FOUNDATION PROJECT

OVERTURE MAPS FOUNDATION

HOME RESOURCES NEWS DOWNLOAD DOCS BECOME A MEMBER

Data Location

| Theme | Location |
|------------------|--|
| Admins | <ul style="list-style-type: none"> Amazon S3: <code>s3://overturemaps-us-west-2/release/2023-10-19-alpha.0/theme=admins</code> Microsoft Azure: <code>https://overturemapswestus2.blob.core.windows.net/release/2023-10-19-alpha.0/theme=admins</code> |
| Buildings | <ul style="list-style-type: none"> Amazon S3: <code>s3://overturemaps-us-west-2/release/2023-10-19-alpha.0/theme=buildings</code> Microsoft Azure: <code>https://overturemapswestus2.blob.core.windows.net/release/2023-10-19-alpha.0/theme=buildings</code> |
| Places | <ul style="list-style-type: none"> Amazon S3: <code>s3://overturemaps-us-west-2/release/2023-10-19-alpha.0/theme=places</code> Microsoft Azure: <code>https://overturemapswestus2.blob.core.windows.net/release/2023-10-19-alpha.0/theme=places</code> |
| Transportation | <ul style="list-style-type: none"> Amazon S3: <code>s3://overturemaps-us-west-2/release/2023-10-19-alpha.0/theme=transportation</code> Microsoft Azure: <code>https://overturemapswestus2.blob.core.windows.net/release/2023-10-19-alpha.0/theme=transportation</code> |
| Base | <ul style="list-style-type: none"> Amazon S3: <code>s3://overturemaps-us-west-2/release/2023-10-19-alpha.0/theme=base</code> Microsoft Azure: <code>https://overturemapswestus2.blob.core.windows.net/release/2023-10-19-alpha.0/theme=base</code> |



overture-maps/notebooks/hello_overture.ipynb at main · tebben/overture-maps

https://github.com/tebben/overture-maps/blob/main/notebooks/hello_overture.ipynb

overture-maps / notebooks / hello_overture.ipynb

Preview Code Blame 481 lines (481 loc) · 13.9 KB Code 55% faster with GitHub Copilot Raw

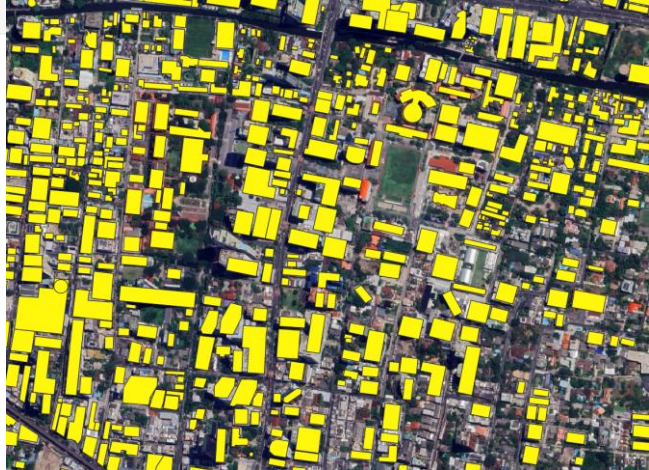
Get buildings within bounds

```

n [7]: den_bosch_bbox = ["POLYGON ((5.300171258675078 51.69073045148227, 5.300611111690557 51.68445866410491, 5.315566114235423 51.68445866410491, 5.315566114235423 51.69073045148227, 5.300171258675078 51.69073045148227))"]
data_bbox = pd.DataFrame(den_bosch_bbox, columns=['bbox'])
#https://dev.to/savo/spatial-data-analysis-with-duckdb-40j9

n [8]: %sql selected_buildings <<
SELECT
    height,
    class,
    numFloors,
    a.geometry as geom
FROM read_parquet('../data/overture_maps/theme=buildings/type=building/*', hive_partitioning=1) as a, data_bbox as b
WHERE ST_Intersects(ST_GeomFromWKB(a.geometry), ST_GeomFromText(b.bbox))

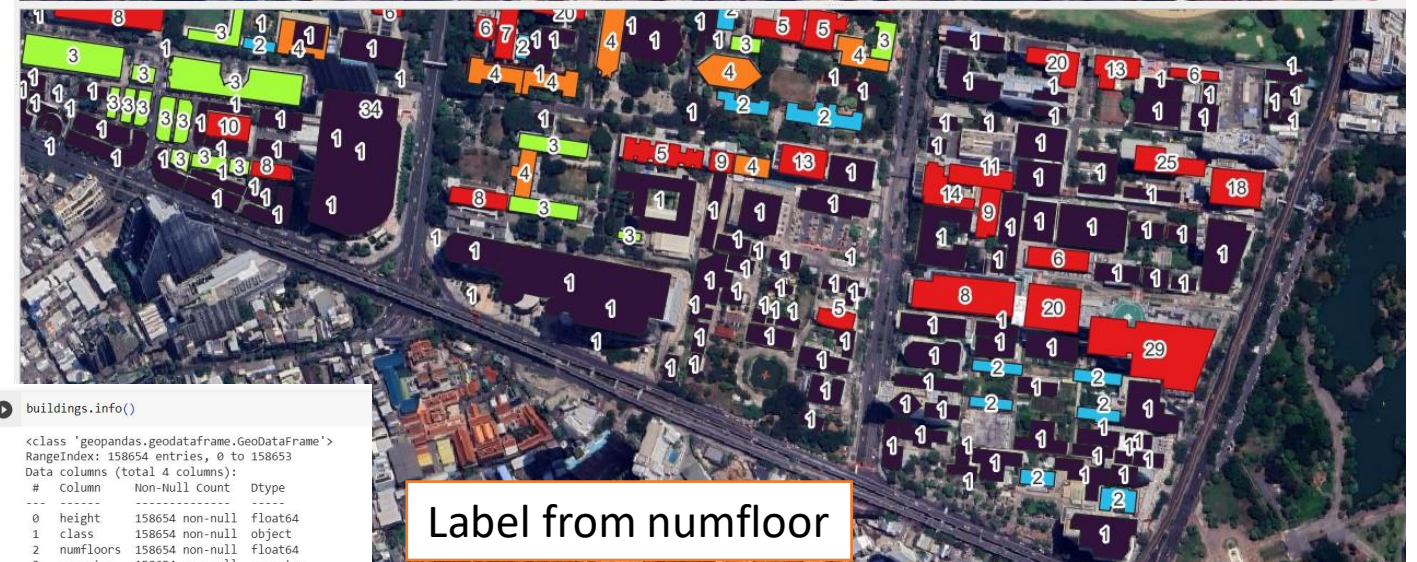
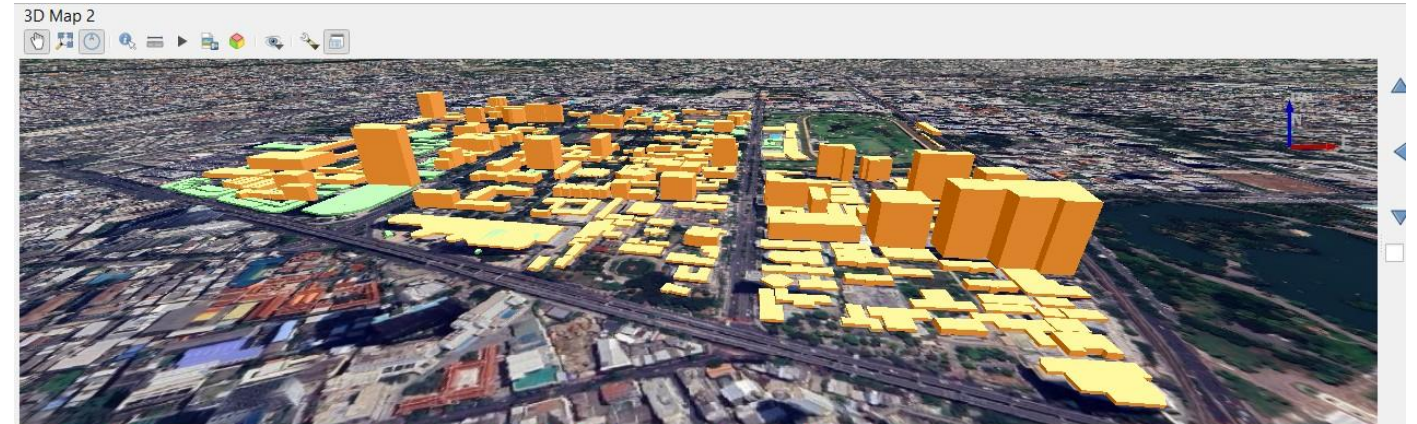
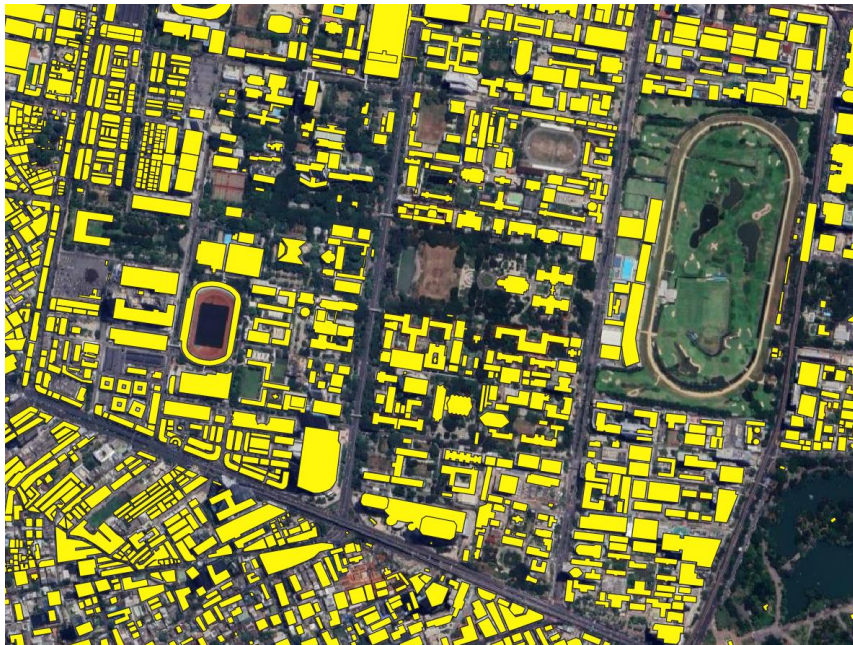
n [16]: print(selected_buildings)
      height  class  numfloors \
0      0.0      0.0          1.0
1      0.0  outbuilding  1.0
2      0.0  outbuilding  1.0
3      0.0  residential  1.0
4      0.0  residential  1.0
...      ...      ...      ...
1958    0.0  residential  1.0
1959    0.0  commercial  1.0
1960    0.0  residential  1.0
1961    0.0  residential  1.0
1962    0.0  residential  1.0
    
```



OVERTURE MAP in Bangkok

| | height | numfloors |
|-------|---------------|---------------|
| count | 158654.000000 | 158654.000000 |
| mean | 0.140081 | 1.154210 |
| std | 3.363749 | 1.701671 |
| min | 0.000000 | 0.000000 |
| 25% | 0.000000 | 1.000000 |
| 50% | 0.000000 | 1.000000 |
| 75% | 0.000000 | 1.000000 |
| max | 235.000000 | 70.000000 |

| class | |
|---------------------------|--------|
| residential | 114631 |
| commercial | 40574 |
| education | 1629 |
| industrial | 848 |
| religious | 337 |
| medical | 217 |
| transportation | 160 |
| civic | 103 |
| service | 47 |
| outbuilding | 37 |
| entertainment | 28 |
| agricultural | 26 |
| | 17 |
| Name: count, dtype: int64 | |



Label from numfloor

```
buildings.info()
<class 'geopandas.geodataframe.GeoDataFrame'>
RangeIndex: 158654 entries, 0 to 158653
Data columns (total 4 columns):
#   Column      Non-Null Count  Dtype
---  -
0    height      158654 non-null  float64
1    class       158654 non-null  object
2    numfloors   158654 non-null  float64
3    geometry    158654 non-null  geometry
dtypes: float64(2), geometry(1), object(1)
memory usage: 4.8+ MB
```


THE END

*2108373 Advanced Photogrammetry
Semester 2/2023*

*Thepchai Srinoi
Department of Survey Engineering
Chulalongkorn University*

