

## **Project: The calculation of housing density from satellite images**

### **How to use it:**

The ways in which different scripts are executed are:

- In a PYTHON nucleus: a call to the basic method
- In line of command: via execution mode

#### **1. Downloading satellite images**

##### **a. Bing Satellite Maps**

Directory name: "download\_satellite\_images-Bing-downloader"

<b>Scripts</b>	<b>Details</b>	<b>Arguments</b>	<b>Execution</b>	<b>Main method</b>
main.py	Downloads images	bot_long bot_lat top_long top_lat	python main.py top_lat top_long bot_lat bot_long Example: For The Set, python main.py 5.311335 -4.034104 5.341868 -4.004153	No
stitchTiles.py	Helps reunite pieces of images	No	Python stitchTiles.py	No

**Trailer:** Images should not exceed the size of 20,512 X 20.512.

Requires: opencv-python (pip install opencv-python)

Numpy (basic package)

##### **b. Google Satellite Maps**

Directory name: "download\_satellite\_images- GoogleMaps-downloader"

<b>Scripts</b>	<b>Details</b>	<b>Arguments</b>	<b>Execution</b>
download_tiles_for_py3.py	Downloads images	lon_start lat_stop lon_stop lat_start zoom- level	python download_tiles_for_py3.py top_lat top_long, bot_lat bot_long zoom  Example for the plateau: python download_tiles_for_py3.py -4.0279855151 5.3319522642 -4.0249599833 5.3350461774 19

**Note:** This script downloads and builds the image of the area of interest.

##### **c. OSM satellite Maps**

Répertoire : « download\_satellite\_images\OSM-tiles-downloader »

Scripts	Details	Arguments	Execution
tiles_to_tiff.py	Downloads images	lon_min lat_min lon_max lat_max zoom-level	python tiles_to_tiff.py lon_min lat_min lon_max lat_max zoom-level  Example for the plateau: python tiles_to_tiff.py 5.3319522642 -4.0279855151 5.3350461774 -4.0249599833 16

## 2. Image processing

Provides treatments for images in raster or non-raster format.

### a. Image projection

Directory: "Image-processing-image\_reproject"

Scripts	Details	Arguments	Execution
reproject_raster.py	Allows automatic re-projecting in the destination <b>crs</b> without compression of a tiff image	<i>tif_to_tiff</i> <i>path_to_new_tiff</i> <i>CRS_dst</i> (EPSG:32630)	python reproject_raster.py tif_to_tiff path_to_new_tiff CRS_dst

**Note:** This scenerisgenerally dependent on the resolution of the size of the image. For high-resolution images, consider the **gdalwarptool**, a gdal extension:

```
gdalwarp -co compress=JPEG -co PHOTOMETRIC=YCBCR -co TILED=YES -co "BLOCKXSIZE=256" -co "BLOCKYSIZE=256" -s_srs crs_source -t_srs crs_dst 'src_tif' 'dst_tif'
```

src.tif: The path absolu of the ancienne image

dst.tif: the absolu**path**of the new image

crs\_source: The source of the image to be reprojected, see the metadata of the image

crs\_dst: The destination crs

### b. Image extraction

Directory: "Image-processing-image-clip"

Scripts	Details	Arguments	Execution
image_clip.py	Allows the image to be extracted by a Geoljson file by geometry	path_to_tif: Tiff file path geoljson_url: geoljson file path new_tif_url: the path to the new image	python image_clip.py path_to_tif geoljson_url new_tif_url  Example: Python image_clip.py tmp.tif gdf. Geoljson clip_tmp.tif
image_clip_gdf.py	Allows the corresponding images to be extracted by several geometries	path_to_tif: Tiff file path geoljson_url: geoljson file path	python image_clip.py path_to_tif geoljson_url tiles_dir  Example:

		_dir_dir tiles: path or save the pieces	Python image_clip_gdf.py gdf_multiple. GeoJson tmp.tif tiles
--	--	---	--

### c. Cutting

Directory: "project-details-Image-processing-image-clip"

Scripts	Details	Arguments	Execution
split_raster.py	Cuts a raster file according to the specified size and stores the pieces	path_to_tif path_to_save_tiles tile_size_by_256	python split_raster.py path_to_tif path_to_save_tiles tile_size_by_256  Exemple: python split_raster.py clip_tmp.tif tiles/tiles 256

### d. Reuniting of raster images

Scripts	Details	Arguments	Execution
merge_raster.py	Brings together the pieces of images to form a single image	dir_of_tiles image_path	python merge_raster.py <dir_of_tiles> <image_path>  Exemple : python merge_raster.py tiles clip_tmp.tif

**Note:** The images to be collected must be in the "tiff" format. tiff

### e. Polygon image formatting

Directory: "Image-processing-image\_to\_polygons"

Scripts	Details	Arguments	Execution
image_to_polygons.py	Transforms a raster image to polygonnes of the same crs as the image	tif_url: Path or url of the raster file geojson_url: way from file to exit	python image_to_polygons.py tif_url geojson_url  python image_to_polygons.py '/content/tiff/inference_cocody_reproj.tif' f'/content/exports/buildigns_COCODY.geojson'

### f. Extracting forms of homes

#### i. By the canny method

Directory: "Image-processing canny"

Scripts	Details	Arguments	Execution
canny_methd.py	Extracts shapes corresponding to homes	path_to_image path_to_result path_to_result2	Python canny_methd.py path_to_images path_to_result path_to_result1  Exemple: python canny_methd.py tiles tiles_res tiles_res_img

**Note:** Only for ".tif" extension s files.

Adjust selection conditions based on which images to process

## ii. By the Pymeanshift method

Directory: "Image-processing-watershed\_meanshift"

Scripts	Details	Arguments	Execution
meanshift.py	Extracts shapes corresponding to homes	path_to_image path_to_result path_to_result2	Python meanshift.py path_to_images path_to_result path_to_result1  Example: python meanshift.py tiles tiles_res tiles_res_img

## 3. Scratching POI

Directory: "project-details-POI\_crawler-GoogleMapsStreet-crawler"

Scripts	Details	Arguments	Execution
GoogleMapCrawler.py	Scratches Gogle Street Maps to get the POIs you want	path_to_keywords: Txt file path of wordslés filter_keyword search criteria	python GoogleMapCrawler.py fichier_txt_poi criterion
merge.py	Combine the collected Poi to form a single file.	No	Python merge.py

**Note:** It is necessary to create a "data" directory to control POIs according to keywords. The "merge.py" script releases the "merged\_frame.csv" file, which contains all the peas.

## 4. The modèles

### a. Detection of inhabited areas

Directory: "classification-buildingNoBuilding"

Scripts	Details	Arguments	Execution
buildingNoBuilding.py	Classifies images in size 256x256	path_to_images: Path of Images ext: extension of images	python buildingnobuilding.py path_to_images ext  Example: python buildingnobuilding.py /.tif images

**Note:** Images may not be 256x256 in size.

### b. Segmentation of homes

Directory: models-segmentation-fastai"

Scripts	Details	Arguments	Execution
inference_to_images.py	Lets make the prediction of images. On the way out, we get pictures	path_to_images:Path of Images path_to_results:way to the result	pythoninference_to_geojson.py path_to_images path_to_results  Example: python inference_to_images.py /images /results
inference_to_geojson.py	Lets make the prediction of images. On the way out, we get GeoJson files	path_to_images:Path of Images path_to_results:way to the result	python inference_to_geojson.py /images /results  Exemple: python inference_to_images.py /images /results

**Graphic: Images must be marquee : 256x256.**

The "inference\_to\_geojson.py" scripts can produce errors on jupyter

## 5. Estimation

### a. Azimuth

Scripts	Details	Arguments	Execution
azimuth_estimation.py	Estimate the azimuth	Long: longitude Lat: latitude date_start: goal date date_end: end date spatial_resolution: spatial resolution max_elevation: maximum elevation path_to_results: path of the fichier csv from angles	python azimuth_estimation.py long lat date_start date_end spatial_resolution max_elevation path_to_results  Example: python azimuth_estimation.py -3.9837119579315186 5.301122665405273 "2008-01-01 08:00:00" "2008-01-01 18:00:00" 2.5 az.csv

### b. The size of the dwellings

Scripts	Details	Arguments	Execution
height_estimation.py	Estimates the tail	img_dir: Image Directory ext: Azimuth file extension: azimuth angle	python height_estimation.py img_dir ext azimuth  Exemple: %run -i height_estimation.py tiles ".tif" "178.3282145189649"

### c. Household counts

Scripts	Details	Arguments	Execution
household_estimation.py	Estimates households	img_dir: Image Directory ext: Azimuth file extension: azimuth angle	python height_estimation.py img_dir ext azimuth  Exemple: %run -i height_estimation.py tiles ".tif" "178.3282145189649"

--	--	--	--

## Some Useful Commands

These commands will be useful on colab..

!gsutil cp R chemin\_sourthis chemin\_destination :P our copy a folder

!gsutil cp chemin\_du\_fichier chemin\_de\_destination: To copy a file

!zip -r nom\_du\_fichier.zip chemin\_du\_fichier: To zip a file

! unzip file.zip: To unzip

! unrar x file.zip: To unzip

%run -i python\_script\_.py: allows you to run a script in jupyter a notebook.

To download directly a shared file from a drive

```
# Install PyDrive
!pip install PyDrive

#Import modules

#Ex link : https://drive.google.com/file/d/1c7Ffo1Go1dtUpKcSWxdbdVyW4dfhEoUp/view?usp=sharing

# Get the id from the link 1c7Ffo1Go1dtUpKcSWxdbdVyW4dfhEoUp

!gdown --id id_from_url
```

## Links

[Link](#) 1: Find the box of coordinates long, lat min, max

[Link](#) 2: For unsupervised classification and segmentation

[Link](#) 3: For images of Cocody and Bingerville

[Link](#) 4: For segmentation model

[Link](#) 5: For the classification model