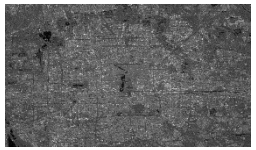
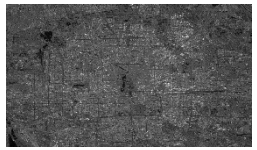
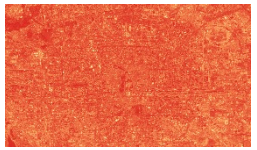
Sentinel Imagery on Google Earth Engine


Sentinel-1-VV


Sentinel-1-VH


Sentinel-2-RGB


Sentinel-2-NIR

1.1 Data export

ee.batch.Export.table.toCloudStorage (fileFormat= "TFRecord")

TFrecords on Google AI Platform

```
dataset = tf.data.TFRecordDataset(exportedFilePath)
featuresDict = {
  'B2': tf.io.FixedLenFeature(shape=[1], dtype=tf.float32),
  'B3': tf.io.FixedLenFeature(shape=[1], dtype=tf.float32),
  'B4': tf.io.FixedLenFeature(shape=[1], dtype=tf.float32),
  'B5': tf.io.FixedLenFeature(shape=[1], dtype=tf.float32),
  'B6': tf.io.FixedLenFeature(shape=[1], dtype=tf.float32),
  'B7': tf.io.FixedLenFeature(shape=[1], dtype=tf.float32),
  'landcover': tf.io.FixedLenFeature(shape=[1], dtype=tf.float32)
}
parsedDataset = dataset.map(lambda example: tf.io.parse_single_example(example, featuresDict))
```

2.2 Model Loading

ee.Model.fromAiPlatformPredictor

- projectName
- modelName
- version
- **inputTileSize**
- **inputOverlapSize**
- **proj, fixInputProj**
- outputBands

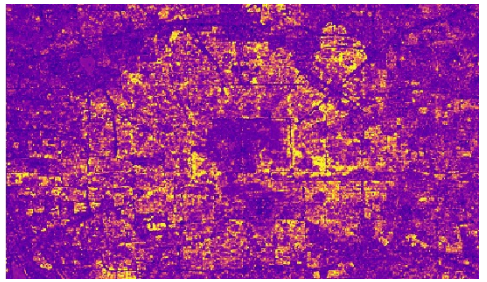
Multiple Input
Multiple Output
?

2.3 Prediction

ee.Model.predictImage(ee.Image)

Predicted results on Google Earth Engine


average building footprint


average building height

2.1 EEification

earthengine model prepare

GEE	AI Platform
Input name	Input array
Output name	Output band

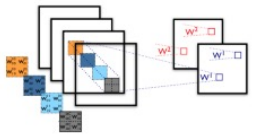
1.2 Training

tensorflow.keras.Model.fit()

ML models on Google AI Platform


Random Forest

eXtreme Gradient Boosting


Convolutional Neural Networks (CNN)

Prediction with pretrained CNN models from SHAFTS on Google Cloud Infrastructures

➢ Workflow for global 3D building information mapping

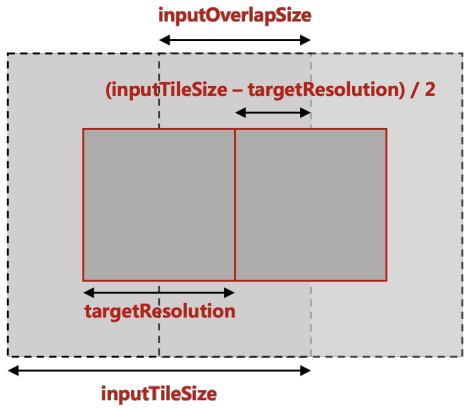
1. Google Earth Engine
- a. Sentinel-1/2 image
- b. SRTM data

region selection

1. Google Earth Engine
- a. High Resolution Settlement Layer

Note: The HRSL dataset includes the information of human population at a resolution of 30 m.

image patching



3.2 Local machine
- a. saved Tensorflow models

3.1 Google Cloud Storage
- a. saved Tensorflow models

model loading

2. Google Cloud Storage
- a. patched TFRecord datasets

dataset loading

4. Compute Engine
- a. Tensorflow model instances

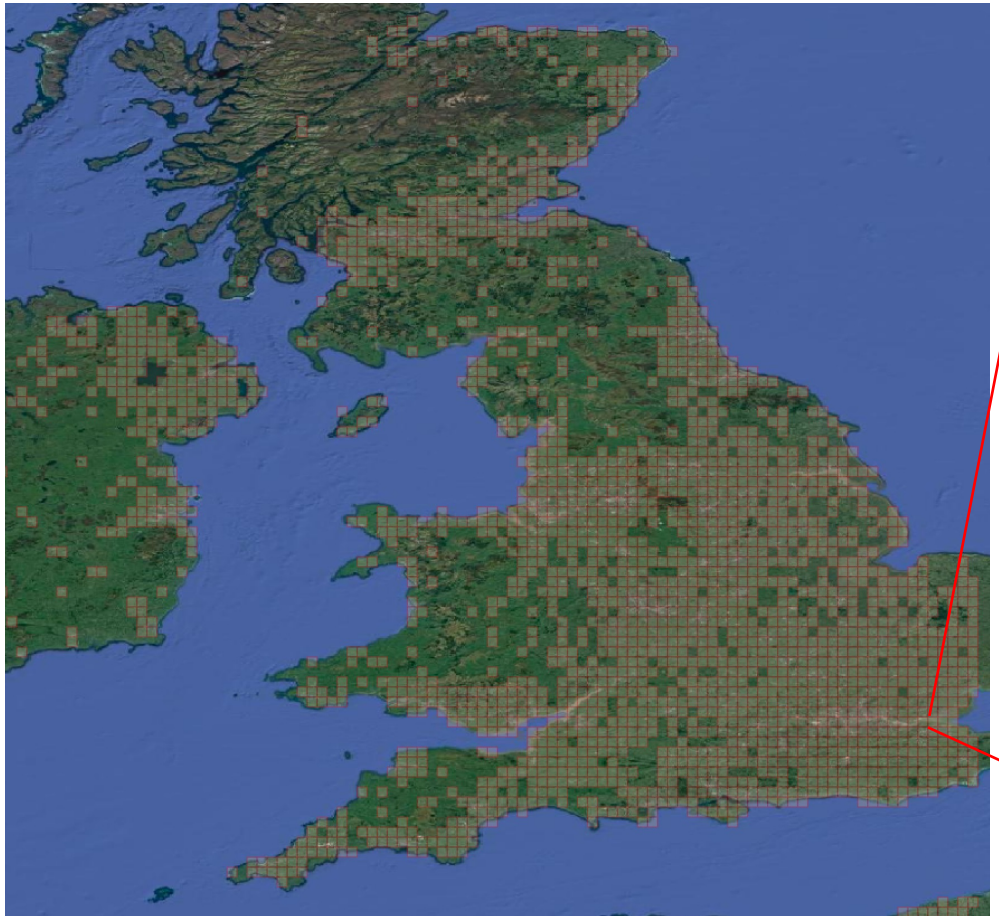
Note: The `Compute Engine` here can be a Google Virtual Machine, a Google Colab notebook or a local machine.

model predicting

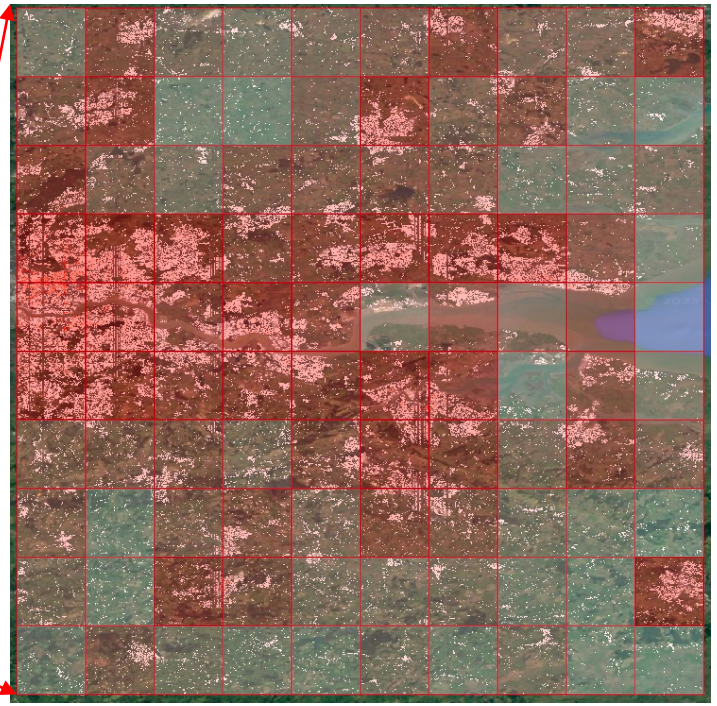
5. Compute Engine
- a. Predicted building height and footprint images

Prediction with pretrained CNN models from SHAFTS on Google Cloud Infrastructures

➢ Workflow for global 3D building information mapping



Regions of Interests
0.9º × 0.9º

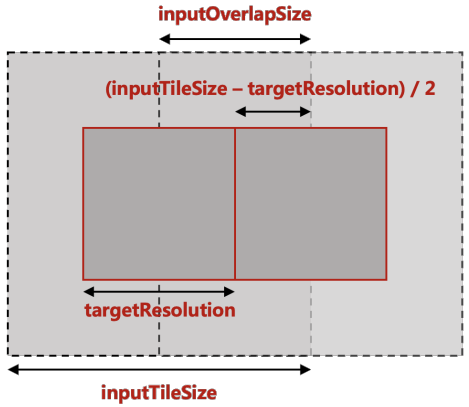


Sample areas
0.09º × 0.09º

Input patches
0.0009º × 0.0009º

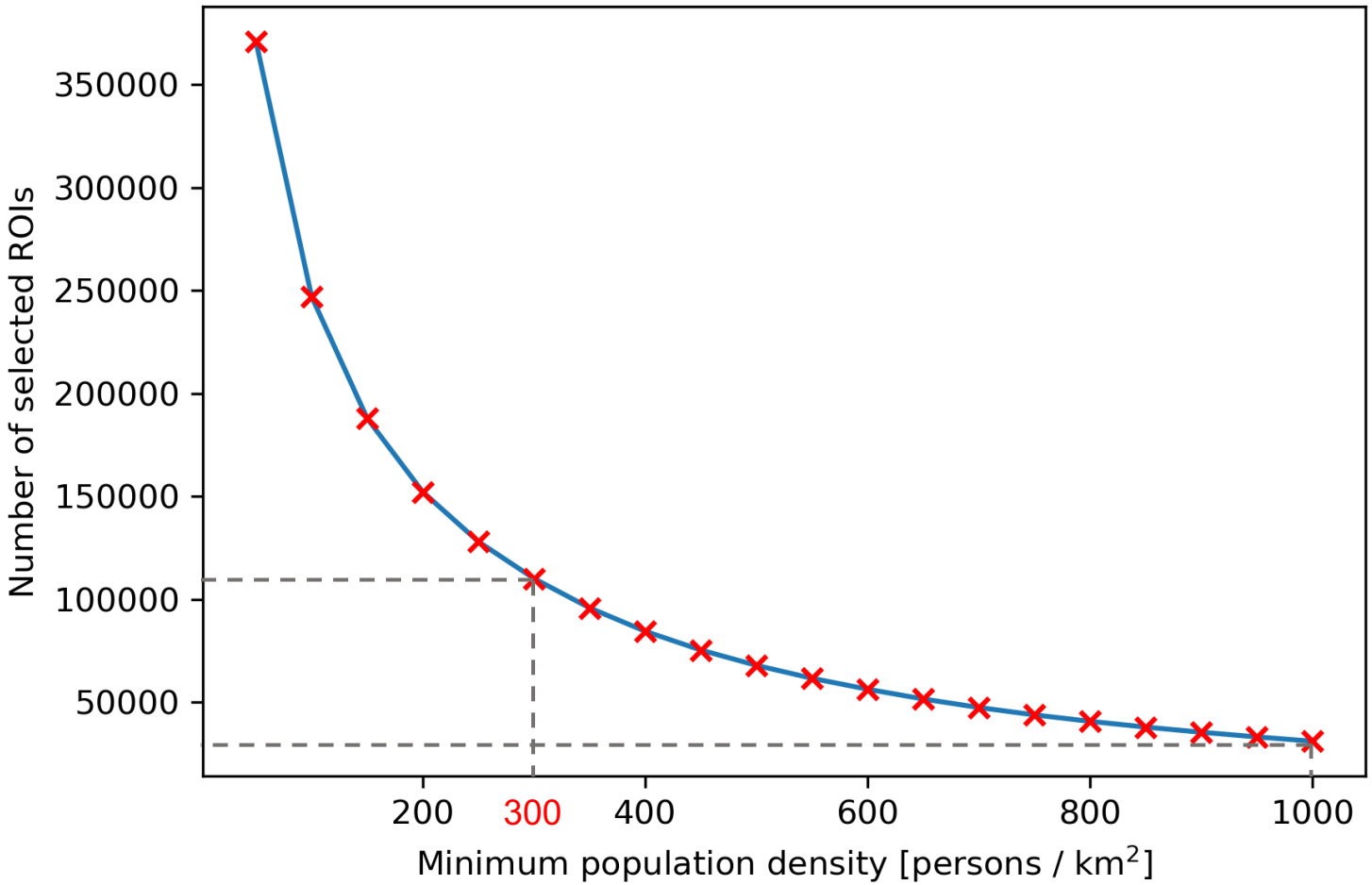
SHATFS
prediction

100 × 100 = 10000 patches
for every sample area



Prediction with pretrained CNN models from SHAFTS on Google Cloud Infrastructures

➢ Workflow for global 3D building information mapping



- **Test the time of CNN inference on UCL's Research Computing Service**
- **If we want to finish our task in one week, then we need:**

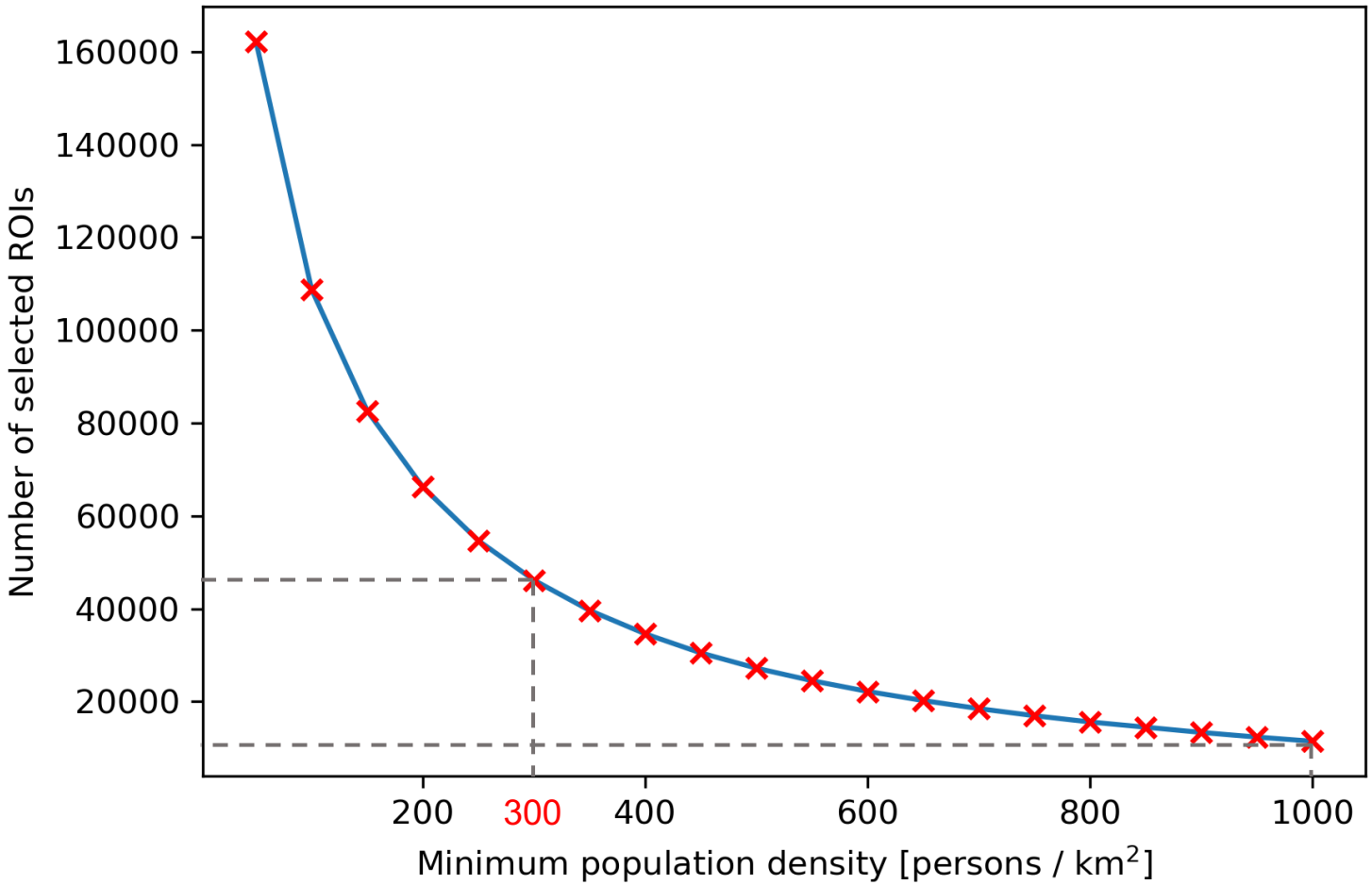
~ 20 s / ROI

- **300 persons / km^2: ~10^5 ROIs (0.9º × 0.9º)**
- **1000 persons / km^2: ~3×10^4 ROIs (0.9º × 0.9º)**

Reference population data: https://sedac.ciesin.columbia.edu/data/set/gpw-v4-population-density-adjusted-to-2015-unwpp-country-totals-rev11

Prediction with pretrained CNN models from SHAFTS on Google Cloud Infrastructures

➢ Workflow for global 3D building information mapping



- **Test the time of CNN inference on UCL's Research Computing Service**
- **If we want to finish our task in one week, then we need:**

~ 60 s / ROI

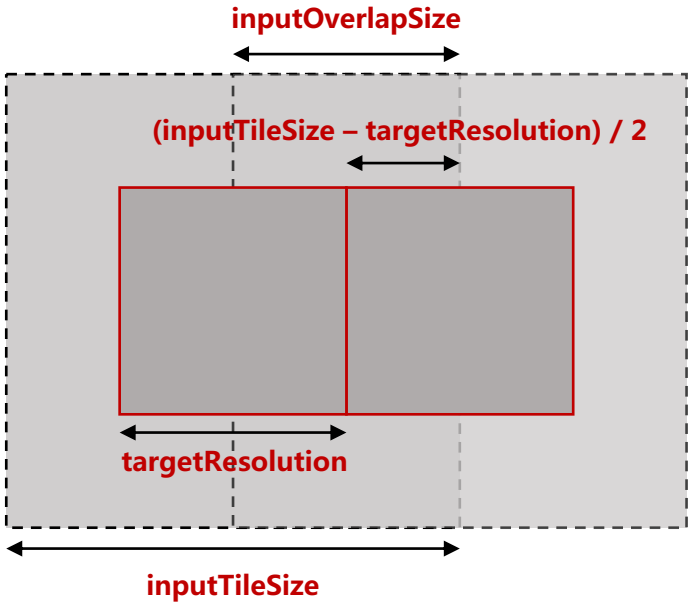
- **300 persons / km^2: ~5×10^4 ROIs (0.9º × 0.9º)**
- **1000 persons / km^2: ~1×10^4 ROIs (0.9º × 0.9º)**

Reference population data: https://ciesin.columbia.edu/data/hrsl/#data

➢ Code Optimization

- **Optimization for exporting datasets from Google Earth Engine**
  - **Select the Sentinel-1 dataset in the natural scale (`S1_GRD_FLOAT` dataset)**
  - **Improve the efficiency of aggregation analysis:**
    - Reduce the kernel size for, e.g., focalMin, reduceNeighborhood
    - Constrain the range of proper images by increasing `S2_CLOUD_PROBABILITY` threshold
    - Replace the `median` with the `mean`
  - **Fine-tune the queueing of concurrent tasks**
    - Number of concurrent tasks ↑ ⇒ check tasks' status ↑, parallel execution of tasks ↓
  - **Parallel submitting tasks by Python's multiprocessing module**
    - Tasks are blocked and cause nearly 100% memory usage of the GeoServer

- **Optimization for inference with TensorFlow's CNN model**
  - **Use multi-task learning-based models to predict building footprint and height simultaneously**
  - **Increase the number of workers, `num_parallel_calls`, for parallel loading of remote datasets**


inputOverlapSize
(inputTileSize – targetResolution) / 2
targetResolution
inputTileSize

Resources