Identifying Houses in Areas without Addresses via Satellite Imagery and Mask-RCNN

Bob Adams
July 2023

Photo Credit: NASA - ISS030-E-254053 (22 April 2012)

Problem Statement

Though common throughout most western countries and localities, unified building address systems are not universally available. From remote communities in Africa and Ireland to Carmel by the Sea in California ***descriptive addresses*** are common and often require local knowledge to navigate. Guiding deliveries and guests to unaddressed locations often requires providing detailed directions or providing local help, though providing a GPS location via a 'pin' in a mapping software is common in some areas.

Goal

This project seeks to leverage publicly available satellite and aerial imagery of unaddressed areas to identify buildings, roof types and materials and provide users with a map narrowing down likely destinations. In cases where the town/city designation, paired with the roof description is unique, the goal is to return a geolocation of of the building they are looking for. Otherwise, this project seeks to reduce the number of wrong turns taken in unfamiliar, unaddressed areas.

The initial scope of this project is to create a proof of concept, covering all key subjects

Penelope's House - Manatades
via corfuhomefinders

Manatades, Corfu
via Google Maps

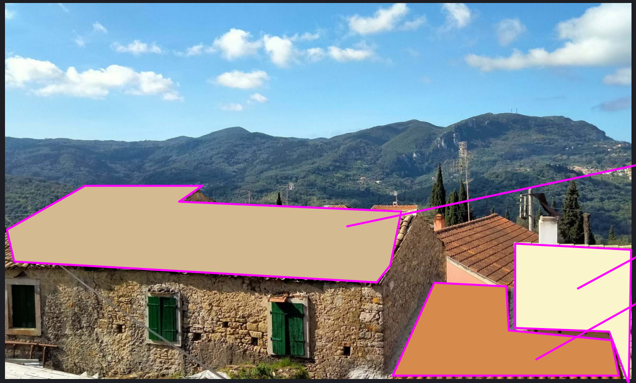
003-1040559   1250 003-77156.8   1760 0009-14563.7   73273

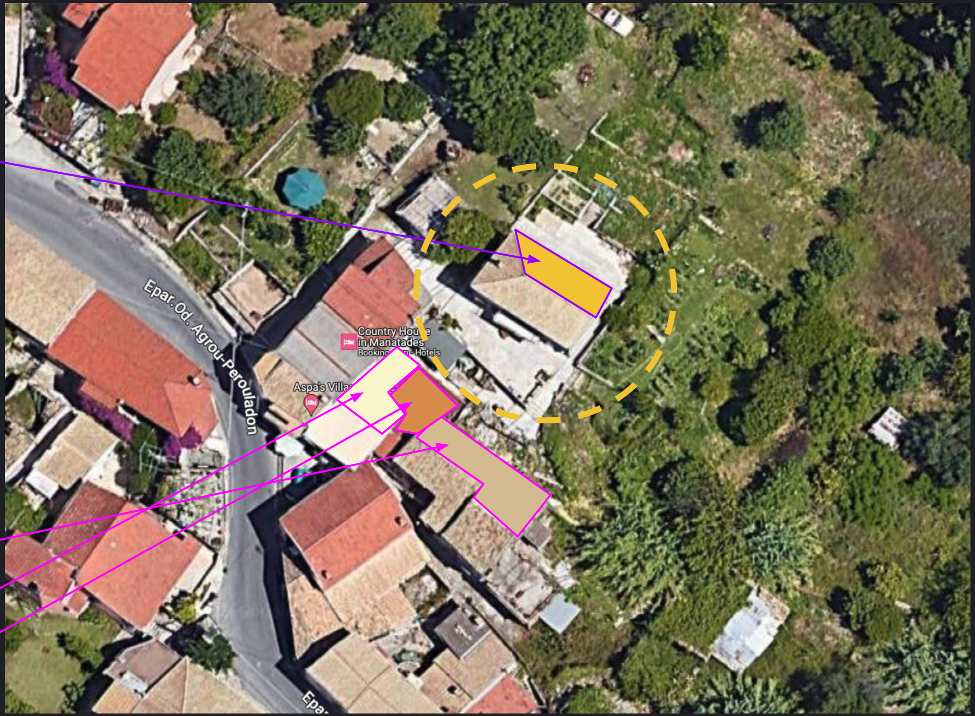
Penelope's House - Manatades
via corfuhomefinders

Manatades, Corfu
via Google Maps

003-1040559 1250 003-77156.8 1760 0009-14563.7 73273

Penelope's House - Manatades
via corfuhomefinders





Manatades, Corfu
via Google Maps

003-1040559   1250 003-77156.8   1760 0009-14563.7   73273

Project Overview

1. Boundaries

Define Geographic Boundaries for Image Collection

[Greek Government]

2. Images

Source Satellite Imagery from Target Area

[Google Earth Engine]

3. Model Selection

Select a model to predict building locations

[Mask-RCNN]

4. Training Data

Source Training Data to Enable Building Detection

[SpacenetV2]

5. Modeling

Train Selected Model for Building Detection & Assess

[Google Colab]

6. Identify Buildings

Extract Locations and Roof Colors from Buildings

[Geolocation & Colors]

003-1040559 1250 003-77156.8 1760 0009-14563.7 73273

T

1

How many pixels of imagery are available on Google Earth?

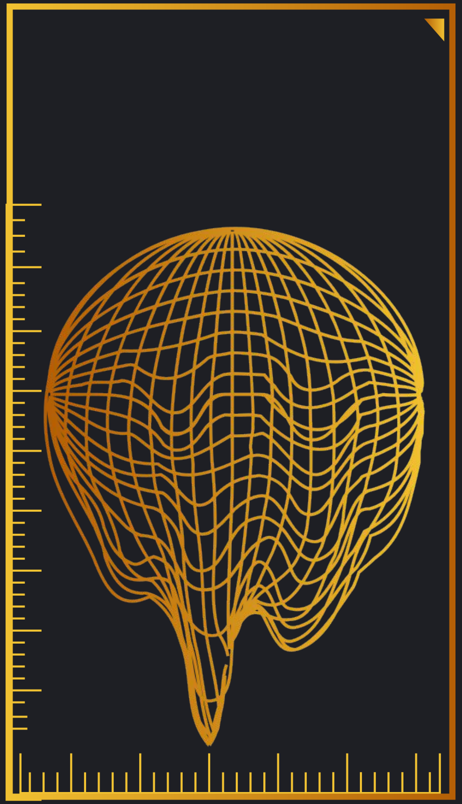
003-1040559 1250 003-77156.8 1760 0009-14563.7 73273

https://www.theatlantic.com/technology/archive/2016/06/google-maps-gets-a-satellite-makeover-mosaic-700-trillion/488939/

700,000,000,000,000+

700 Trillion, per The Atlantic 2016
2.1 Petabytes of 8-bit pixels

How many pixels of imagery are available on Google Earth?

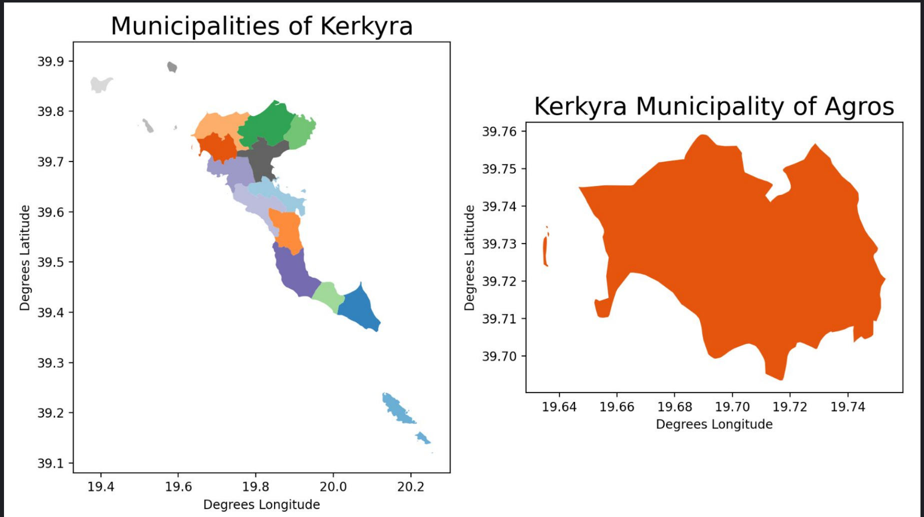


Setting scope-appropriate boundaries is key!

003-1040559 1250 003-77156.8 1760 0009-14563.7 73273

https://www.theatlantic.com/technology/archive/2016/06/google-maps-gets-a-satellite-makeover-mosaic-700-trillion/488939/

Geographic Boundaries

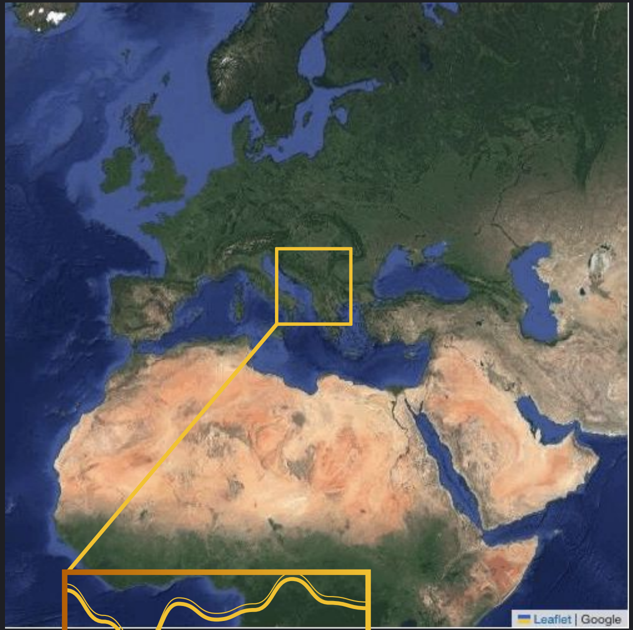


- Coastline geojson objects sourced from the Greek Government
- Corfu identified by first closing the objects, then sorting objects by area
- Municipality Boundaries Sourced and processed similarly

003-1040559 1250 003-77156.8 1760 0009-14563.7 73273

T

2



02.

Image Sourcing

Extracting Image Tiles from Google Earth Engine

Considerations

Capturing relevant, high quality images

Leaflet Layer

Google Earth Engine has access to a wide range of visualizations (leaflets).

Select a layer with high clarity coverage at deep zoom level for quality.

Esri Imagery

Google Satellite Imagery

Image Size

Balance:

- Model Requirements $(2^{n}_{x,y})$
- Image Count
- Objects per Image

Grid Patterns

Consider:

- Image Count
- Partial Building Capture
- Boundary Considerations

Offset Grids, Center Inside

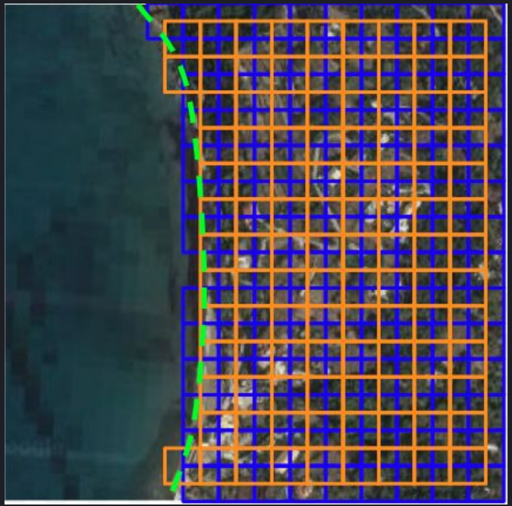


Image Sourcing

T

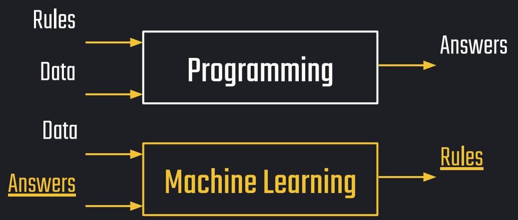
2

Images are captured with ½ vertical & horizontal offset

Samples of 1800 Images Captured for Agros, Corfu | Imagery Credit: Google Satellite

Model Design and Training

Machine Learning and Model Training Overview



"This is what every possible house looks like. If you see something that looks exactly like this, it's a house!"

"Here are 10,000 labeled pictures of houses (and not houses). Figure out what makes an image a house."

Is this a picture of a Frog?
Because Machine Learning Models are only capable of making (accurate) predictions on classes they have been trained on, models must be trained to *solve specific tasks using relevant data*.

To identify houses from aerial photos, any model used will need to be trained on labeled images of buildings from altitude.

But we don't need to build every model from scratch!

Image Citation: Deep Learning with Python by Francois Chollet

T

3

003-1040559 1250 003-77156.8 1760 0009-14563.7 73273

Neural Networks

Layers of Logic and Where's Waldo

How do you go about finding Waldo on a beach full of people?

1. Start at a corner
2. Look for key features (glasses, hat) in small sections
3. Remember the basics about that section (swimmer in the water)
4. Move across the image and repeat

> Convolutional Neural Networks follow the same process

What if you don't find Waldo on the first pass? <u>You might have to start over.</u>

> Here is where Convolutional Neural Networks have the advantage.

CNNs pass the key features to following 'layers' (the next pass looks at a simplified image). They can repeat this process, building more complex representations of the input image (the beach scene) to find the answer.

The middle layers are 'hidden' ... we only really care where Waldo is!

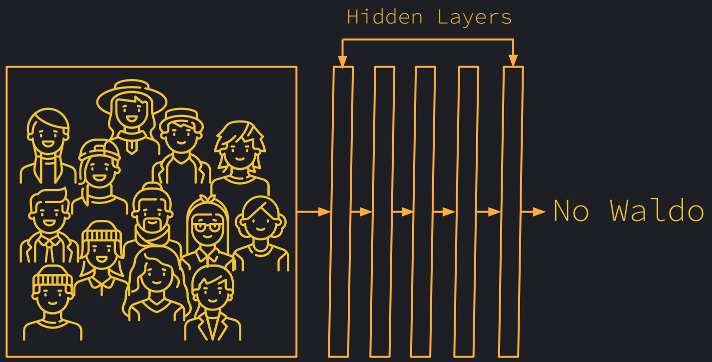
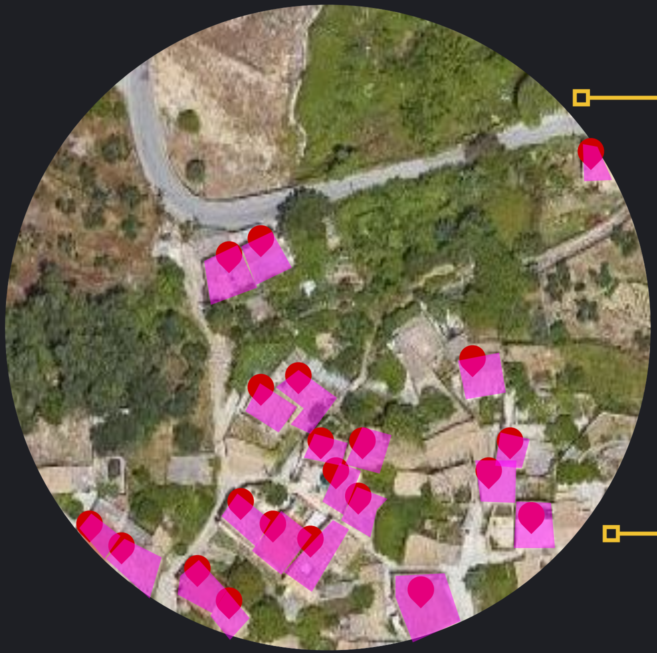


Image Source: https://www.hiclipart.com/free-transparent-background-png-clipart-ahrop

Model Selection

Selecting a pre-trained machine learning model for image evaluation



T

3

RGB Images

Convolutional Neural Networks perform well on images

Instance Segmentation

Identify Multiple Buildings per Image

Object Masks

Identify pixels belonging to each identified building to enable color extraction

003-1040559 1250 003-77156.8 1760 0009-14563.7 73273

Mask-RCNN

Pre-Trained Image Segmentation and Object Detection Neural Network
Constructed by Matterport on a Feature Pyramid Network and Resnet 101 Backbone

The model is designed to detect multiple objects per class (ex. car) from input images and generate bounding boxes and object masks with high accuracy.

Mask-RCNN is available to download and retrain for specific use cases

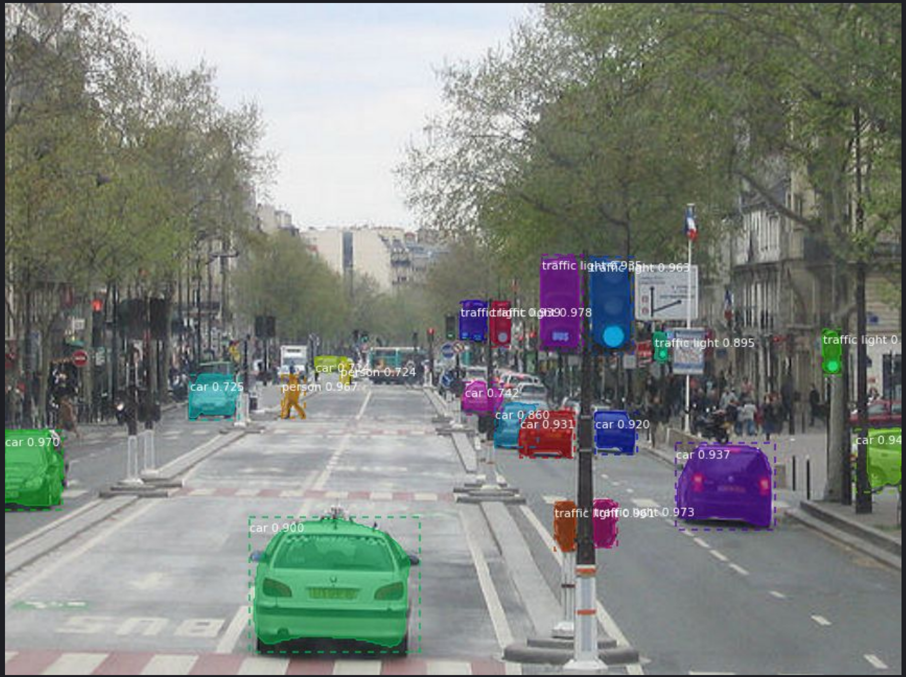


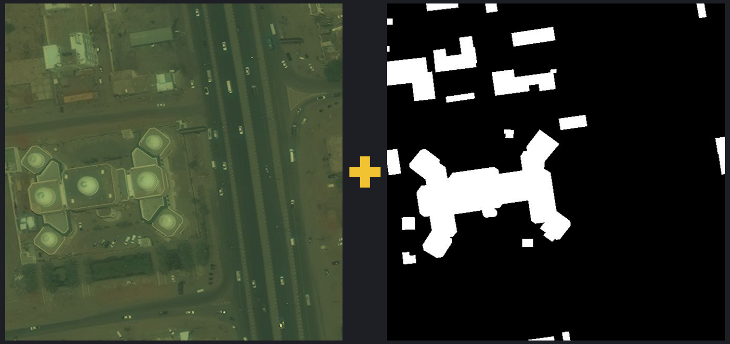
Photo Credit: Matterport Mask-RCNN via Github

Training Data - SpacenetV2

High Quality, Labeled Building Images to Train the Model Input Layers

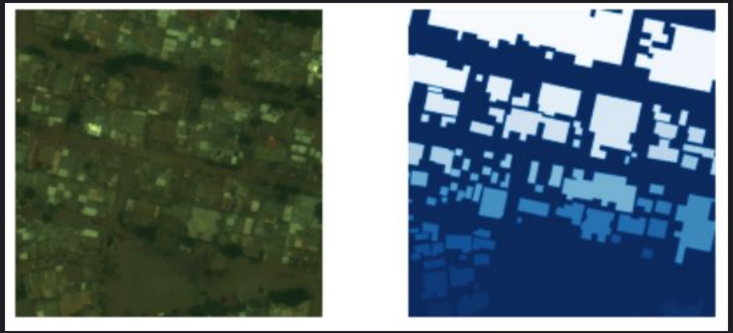
Aerial Images

Thousands of available images from Khartoum, Las Vegas, Paris, Singapore



Individually Labeled Buildings

Individual building outlines in geojson format



Significant data preparation is required when using this dataset, including file structure and naming, image filetype updates and individual mask extraction (blue image on far right). See Readme for details.

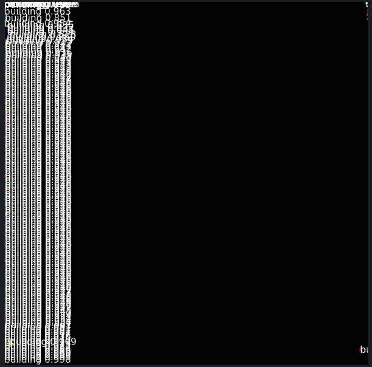
Modeling with Mask-RCNN

Expected Output

Bounding Boxes and Masks which approximate the known outputs on the left (SpacenetV2 data)

Actual Output

Zero-dimension (or none found) masks and bounding boxes, generally matching horizontal or vertical lines.



Modelling Parameters:

- 160 Images, 18 Epochs, 100 steps/Epoch
- Initial Weights: Imagenet
- Training ‘Heads’ weights only
- Leveraging model branch updated for compatibility to Tensorflow 2.0+
- Execution on Tensorflow 2.12 via Colab

Results:

- 0% mAP Score

Drivers:

- Mask-RCNN have been upgraded to work with Tensorflow 2.x… but running on Tensorflow versions 2.6+ results in a compatibility issue with the weight loading process.

  Result: The top layers are retrained, while the hidden layers are incorrectly configured.

Modeling with Mask-RCNN

Modelling Next Steps:

- Instantiate and run Mask-RCNN as defined in this project in an environment built to run Tensorflow 2.4 and matching Keras 2.4.3 libraries
- This has been shown on other projects to return results as expected, including on aerial imagery.
- As of July 2023, versions of Tensorflow prior to 2.8 are unavailable in Google Colab - an offline environment with at least one GPU is required.

T

5

Model Output and Extension

Extending Mask-RCNN Outputs to Identify and Categorize Buildings - Unit Tested

Mask-RCNN

Class Info

Identify Buildings

Could be extended to identify sub-classes of buildings

Bounding Boxes

BUILDING01

Building Locations

1. Center of Bounding Box
2. Transposed to Center of Image
3. Combined with Geolocation Data from Image >> (Latitude, Longitude)

Object Masks

Roof Colors

1. Isolate building mask pixels
2. Convert image pixels outside mask to black
3. Calculate average Red, Green, Blue of the remaining image

Conclusions

A service enabling users to filter local buildings by roof color using available satellite imagery is possible!

Recommendations

1. Run Mask-RCNN on a compatible version of Tensorflow in a dedicated environment
2. Train the model on the full SpacenetV2 dataset (all cities) for improved performance
3. Consider transfer learning opportunities with roof structure type datasets (available from loosgagnet and in the repository)
4. Construct UI to feed geolocations back into a map for end users.

003-1040559 1250 003-77156.8 1760 0009-14563.7 73273

THANKS

Do you have any questions?

rradams@umich.edu
+1 734-355-0977
bobadams1.github.io