Building Footprint Area Extraction using Low-Medium Resolution Satellite Imagery

Final Project Documentation

Authors:

Anna Kiefer, Joshua Delgadillo, Erik Bruenner, Daniel Tisdale, and Connor Sexton

Advisors:

Dr. Franz Kurfess (Cal Poly), Jingzhe Wu (World Bank), Angie Arevalo (World Bank)

Affiliation:

California Polytechnic State University Computer Science Department

5 December 2021

Abstract

In partnership with the World Bank, the Safer Schools team used medium-high resolution satellite/aerial imagery from Esri and Google Maps tile servers and machine learning to estimate building footprints. Specifically, the arcGIS deep learning model was used to estimate the building footprints of at-risk schools in Kyrgyzstan. This machine learning model uses feature extraction to identify the individual school structures in an area of interest.

The building footprints extracted are being used in combination with the known latitude/longitude locations of schools and an existing World Bank economic model to estimate the investments needed to improve the schools' infrastructure. The model resulted in an accuracy of 59% in determining the building footprints when compared to the hand- measured building footprints for each school. The footprint data output from the model will be useful to estimate the size of the buildings in order to improve schools' infrastructure.

Relevance

This project is relevant to the class' focus on intelligent agents and artificial intelligence as a whole. In this project, we'll be using a neural network that will learn to extract building footprints from satellite data, which puts our topic in the domain of machine learning. Our implementation will be tested and applied in the real-world, which suits the research driven aspect of this course. Furthermore, we are continuing to research an area our student-peers explored in the Spring.

Introduction

In partnership with the World Bank, our team used medium resolution satellite imagery and machine learning to estimate the building footprints of schools in the country of Kyrgyzstan. This project is part of a larger initiative at The World Bank called the Global Program for Safer Schools. This initiative aims at boosting investments to improve the safety and resiliency of schools at-risk from natural hazards. Acquiring these building footprints manually would be difficult, as the schools are hard to access geographically and sending assessors and engineers on-the-ground is costly and time-consuming. Therefore, using machine learning to gather this data is cheaper, faster, and more scalable.

The machine learning model we used performed feature extraction to identify the individual school structures to predict the schools' building footprints and square meters. The extracted polygons, along with structural data (number of stories, building material) and other data will be used to help target the buildings most in-need of repair and potential economic investment.

Background and Related Work

As mentioned, this has been an ongoing World Bank and CalPoly DxHub joint effort. Previous teams built a baseline ML model, based on InceptionResnetV2, using images of building facades to predict building features and to expand a deep learning classification system to capture key features of the Global Library of School Infrastructure (GLOSI) structural typologies (see [1] for the results from last quarter).

However, this team worked on a different facet of predicting building features, namely, through the use of satellite imagery. Broken down, this project has three primary areas of interests: the collection of non-commercial satellite imagery, and the training of or the adaption of a model for estimating building footprints for variable satellite resolution quality.

Potential Datasets of Non-Commercial Satellite Imagery:

We will be analyzing available satellite imagery providers to find the highest resolution free imagery available for Kyrgyzstan. Some of these satellite providers include Google Earth, BlackSky, Planet, Digital Globe, and NASA. Thus far, the highest resolution, non-commercial imagery we have access to is 10m per pixel. The Awesome Satellite Imagery Datasets github repository [2] has links to lots of different high resolution satellite/aerial images ranging from 0.3m - 5m per pixel. These datasets have been used previously for training multiple computer vision and deep learning models including ones that identify buildings.

The World Bank will provide images of 100 school buildings in Kyrgyzstan. There is also other metadata provided (e.g. latitude/longitude). This dataset will be useful for potentially training a model and in validating our implementation since the dataset includes the area of each building's footprint.

Models for Estimating Building Footprints & Making Them Resolution Invariant:

There are existing open source building footprint estimation machine learning models, including Esri's Building Footprint Extraction model [3], Microsoft Azure building footprint extraction model [4]. After acquiring the satellite imagery, we will use these models to try to extract the building footprints

Software and technologies used may include PyTorch and/or Tensorflow (if training our own model) and spatial libraries (geopandas, rasterio, shapely). We will be using AWS compute resources to process the data. Depending on the quality of the existing satellite imagery, we may need to perform some preprocessing on the dataset (converting the image to grayscale, image augmentation, downscale, etc).

The majority of current research on feature extraction utilizes exclusively high-resolution satellite imagery, thus most of the research will be helpful with respect to architecture strategy, :

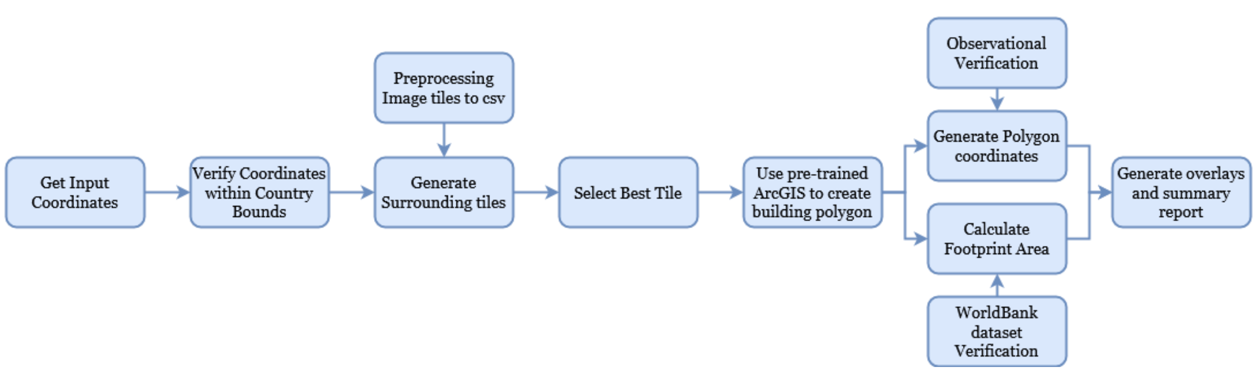
- [5] Uses satellite images to identify buildings at a pixel-perfect scale. The results are promising, but not ideal. Deep learning was used as a means to accomplish semantic segmentation.

- [6] Used U-net and a larger sample of data to one-up other building footprint extraction methods. Improvements range from 1.1% to 12%, but the authors included sources of their data. Basically, a better version of the first method (still uses deep learning semantic segmentation)

- [7] proposed a simple GAN (CWGAN) and a more complex GAN (CWGAN-GP), both of which performed well. CWGAN almost outperformed Deep Learning methods, and CWGAN-GP outperformed them outright. CWGAN-GP adds a grade penalty, which is why it performed better than its counterpart.

- In [8], the topic of upscaling a satellite image was brought up to help improve the efficiency when entered into a model. The paper fuses the datasets Sentinel-1 and Sentinel-2, both at a resolution of 10m, and upscales them to effectively 2.5m resolution. The two algorithms they used to perform this task were Bicubic and Nearest Neighbor. This allowed for better results when trying to map building footprints.

System Design



Implementation

1. **Coordinate input**: Python program takes in a set of longitude and latitude coordinates as well as a school id to identify the school
2. **Verify Coordinates:** The program checks each coordinate to make sure they are inside the country of kyrgyzstan
3. **Generate Surround tiles:** Once the coordinates are verified, they are converted to an x and y value in order to correspond to specific satellite image tiles. In order to encompass large schools, the boarding tiles are generated so that a 3x3 of satellite image tiles can be loaded. The python code then uses the arcGISonline tile server to access ESRI's satellite image data and download the corresponding tiles. The tiles are concatenated into a single image tile that contains the school(s) inside the image.
4. **Select Best Tile:** The x and y tile points are recalculated without being rounded in order to approximate the center of the image. Once this is done, the code then zooms in on the longitude and latitude center of the image so the school takes up a majority of the image. This image is then saved as a .tiff file and is ready to be imported into arcGIS Pro.
5. **Use pre-trained ArcGIS to create building polygon:** Using built in functionality of ArcGIS with Jupyter notebooks, we intend to create accompanying notebooks which interact with the GUI of ArcGIS Pro in order to maximize functionality and usability.
6. **Generate Polygon/Calculate Footprint Area:** ArcGIS outputs an object with readable results with minimal modification in order to make results easily understood and interpretable to humans.
7. **Generate Overlays and Summary Report:** Takes the outputs of the ArcGIS area and the calculated area based on the footprint boxes and then compares both of them to the actual footprint area sizes. We then can generate a report of how accurate the arcGIS model vs. how accurate our calculations are based on the bounding box placed by the arcGIS model. We will also add functionality to generate reports of multiple coordinates and create a folder/json file with all the information

Technologies, Tools, Languages, Development Environments

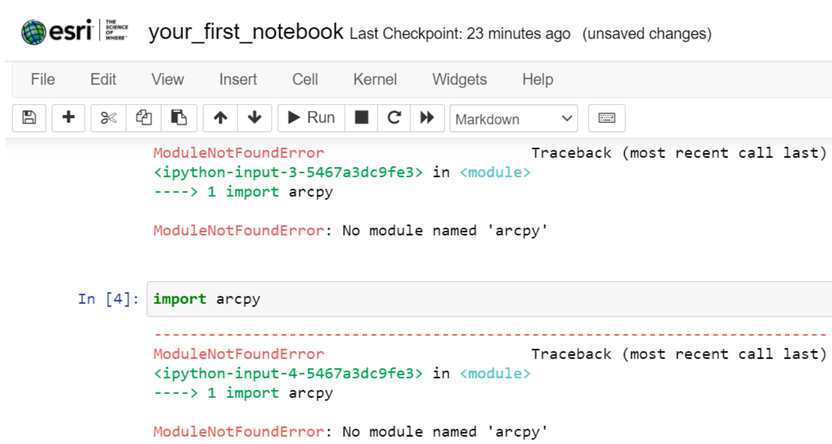
ArcGIS: ArcGIS is the toolset that we are using to analyze school footprint aerial imagery. It is a family of client software, server software, and online geographic information system (GIS) services developed and maintained by Esri. We've become familiar with its functionality and have met with Russ White (Cal Poly's Data and GIS specialist) to learn more about how to use this tool and help with using ArcGIS's python API.

Conda/Python: Conda is an open-sourced package and environment manager for Python. ArcGIS uses Conda for all their packages, as thus, we are leveraging conda's powerful libraries and ease of use to build a comprehensive programming pipeline.

ArcPy, jupyter notebook, AWS: ArcPy is ArcGIS's Python API. It supports the deep learning tools necessary for this project. ArcGIS has a built-in functionality to run jupyter notebooks with the ArcPy environment available to them. This environment can also be cloned and modified by a user (though this did not work in practice). There is support for running ArcGIS and jupyter notebooks in AWS, but this also did not work when we tried to use it).

Obstacles and Implementation Issues

AWS/Docker



Pictured above is a screenshot of ESRI's custom docker (see ESRI's installation guide here). Docker is an encapsulated environment with the jupyter and python environment preconfigured. Docker provides an OS level virtualization to deliver this software in containers. Essentially, we have a Docker container running within our AWS EC2 instance container. It however, does not work, and there is very little documentation available to help troubleshoot. Because of this, we are forced to abandon the goal of running our project on AWS.

ArcGIS API: Working with the API has been a difficult endeavor. We tried several avenues including two official Ersi-sponsored methods of installing the ArcPy library onto our devices. It seems that the installation was broken a few months back. As pictured above, we encountered errors with importing ArcPy in many different development environments. We tried pip, docker, conda, cloning an environment directly from ArcGIS, and notebooks. None of these methods worked out and we settled on using notebooks built-in to ArcGIS with the environment pre-loaded.

Satellite image licensing: We were able to find a good satellite image tile server that we could use for getting images. The resolution was really good and worked well with our model. The issue is that the server ended up being a legacy outdated server that was being reused for the google maps API. It was against google's terms and services to access the server without an API key and we were not allowed to use their images. This led us to look into other satellite images we could use that were high quality and we were allowed to use. The few we found were Sentinel-2 images, but the issue is that the resolution for Kyrgyzstan was around 20m so nothing was visible. Another option that we are still keeping open is Bing maps. They allow students and non-profits to use their API and access their satellite imagery. We ended up deciding to stick with the ESRI images and access them through the arcGIS tile server. This was the best option because the resolution is around 0.5m and requires minimal adjustments to the already written code.

YoLo: We have tried running a pre-trained object detection network in hopes it would help with finding the building in the collection of tiles in order to obtain more tight zooms of the buildings. However, it most likely did not have many buildings in its training set, so we have been looking at more specialized bounding box methods that are specifically designed for detecting buildings through satellite imagery such as spacenet

Testing and Validation

With our approach of using a pretrained model for footprint extraction, all data we test on can be used as validation data. We have a list of 138 schools and their footprint area (sq. meters) provided by the World Bank. Since our project hopes to extract both the area and shape of the buildings' footprints, we can validate the areas we generate using a script to compare with official measurements made by the World Bank. However, for shape information, due to lack of comparison data, we validate the area shapes through observation rather than objective measurements.

Mappings of Features to Requirements

Features:

F1: Accept input coordinates of a school with variable precision

F2: Draw a polygon with estimated dimensions of the building footprint

F3: Devise a method to compute the area in square meters given the building footprint

F4: Output a comprehensive report of building footprint ready to be used for further risk assessment

F5: Maintain professional standards of privacy and documentation

Requirements:

R1: The program will take in valid input coordinates and corresponding satellite and aerial-view images of schools

R2: The program will draw a bounding box of reasonable size around each building coordinate, balancing model performance and accuracy

R3: The program will estimate and display the dimensions of each side of the buildings as a generated polygon

- **R3.1:** The program will utilize a readily available pre-trained model or will use a clear network architecture which is trained on an open-sourced and available dataset
- **R3.2:** Program will calculate footprint area (sq. m) and coordinates of corners of polygon

R4: The program will generate a comprehensive report, including:

- **R4.1:** Display footprint shape overlaying surrounding satellite imagery
- **R4.2:** Report footprint area (sq. m) and coordinates of corners of polygon
- **R4.3:** Polygon shape mask file

R5: We will be in compliance with the NDA signed with the World Bank while maintaining readable code with future facing documentation.

Usage of Evaluation Criteria

Verify Coordinates:

In order to verify the coordinates are actually in Kyrgyzstan we took a shapefile of Kyrgyzstan and verified each coordinate lies inside the country. We also verify there are no duplicates of coordinates in order to save time on processing if there are. These do not include the schools that have multiple buildings in the same location. A 100% accuracy was reported validating that every coordinate lies inside of Kyrgyzstan and that there are no duplicate coordinates.

Footprint Area:
The current dataset that includes longitudes and latitudes also includes the area in square meters. The arcGIS footprint model calculates an area in square meters of the detected school based on the generated polygon. The sum of the quotient of the actual area and the arcGIS generated area based on the bounding box gives the accuracy of the building detection model. In addition to the overall accuracy, the accuracies based on size of the building and absolute difference are recorded. These values help paint a picture on how well the model is doing and to find the point in which the model is the most accurate in determining the correct building footprints values.

In addition, the polygons extracted by the model output a confidence percentage for each polygon drawn. The confidence values were taken and averaged in order to create an overall confidence for the building footprint detection model.

Footprint Shape:
Due to the lack of validation data and resources to annotate polygons, we perform simple visual inspections to verify the polygons generated are consistent with the actual building shapes. Provided enough time, we may categorize the shapes as passes or fails and report the accuracy, precision, recall, and f1 scores found.

NDA/Privacy:
Our signed NDA's have been followed. We maintain a private Github with all information presented to people outside the team receiving prior approval by the World Bank team. Additionally, we keep the coordinates of the school hidden for all public presentations.

Documentation:
By creating a comprehensive pipeline, the usage of the project remains simple. And our project interacting with GUI changes and error correction should be easily accessible for end users to run with very little knowledge of the program. Our backend scripts maintain consistent naming conventions, comment styles, and provide concise and easily understood purpose statements.

Overall Model Accuracy
The accuracy predicted by the ArcGIS building extraction model was **71.8%**.

Square Meter Comparison Accuracy
When comparing the square meters provided to the model's predicted square meters, the accuracy was **58.7%**.

Model Runtime

Average Times per location	
Process	**Time (s)**
Tile Selection	1.138
Identifying Footprint (Deep Learning Model)	15.137
Regularizing Results	1.210
Combined	**17.485**

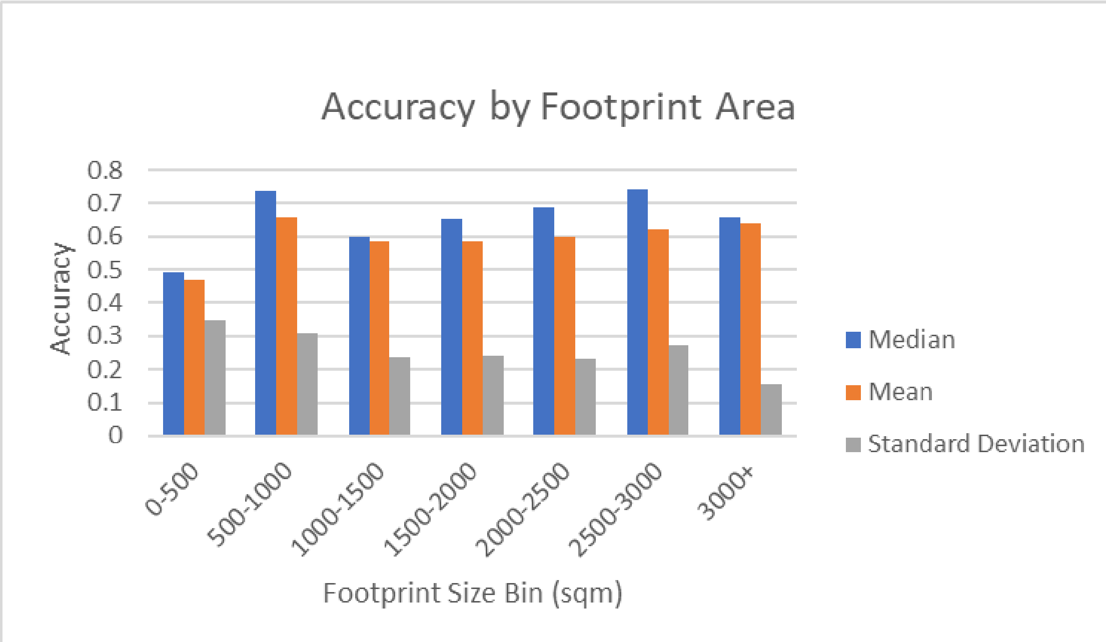
Results Obtained Running the Model on an 8-core Cpu

Taking around 40 minutes to run all 138 buildings

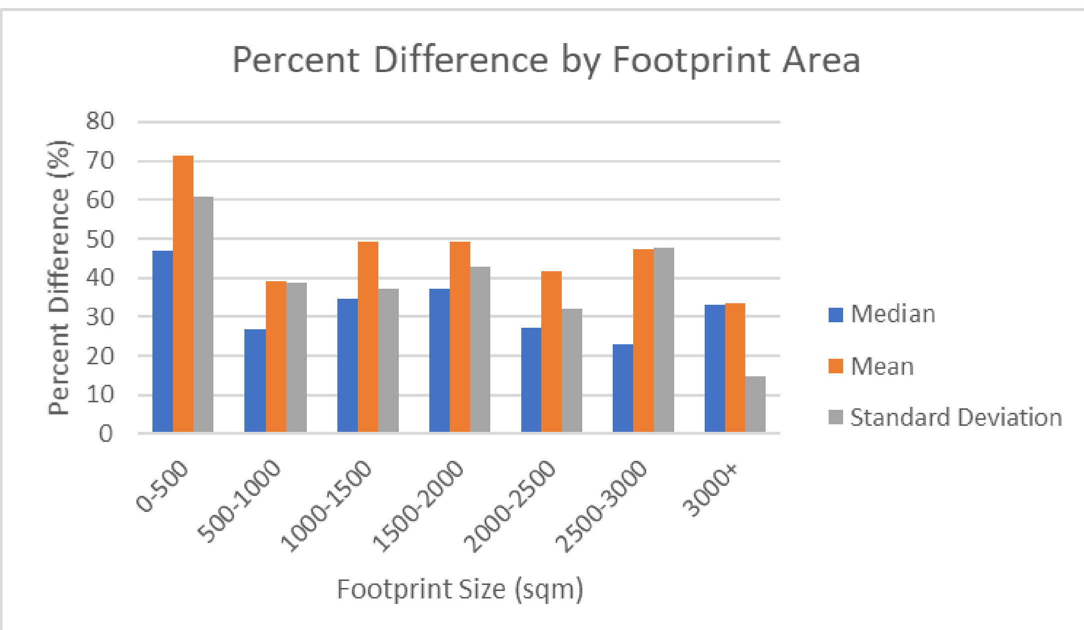
Square Meter Binning

To further evaluate our model's accuracy, we binned the buildings by predicted square meter (the square meter from the data provided by The World Bank) and calculated the accuracy of each bin.

The graph below shows that the model has similar performance on all footprint size bins over 500 square meters. The buildings below 500 square meters had both the lowest mean and median accuracy. We predict this is the case because there are fewer pixels for a smaller building; therefore, the model has less data to use and is more likely to predict incorrectly.



We also binned the percent difference by footprint area, again in increments of 500 sq m. Similar to the first graph, we can see that the 0-500 sq meter group performed the worst, with all other groups performing similarly.



Future Work

There are some areas of this project that could be good for continued development. For example, labelling a Kyrgyzstan dataset and training on similar buildings at the same resolution could help model performance. Additionally, the model's output could be compared using more varied satellite or aerial imagery. For example, running the model against densely vs sparsely populated areas could give an indication of performance and answer questions around the model's shortcomings. This same goes for running the model against images from other parts of the world.

The World Bank's ultimate goal is to identify high-risk areas and target those schools for economic investment. Therefore, it may be useful to combine the building polygons with other building data (infrastructure, number of stories) and weather data (risk of earthquake or inclement weather) to create a risk metric for each school. Obviously this would require additional data gathering.

The model's performance is another area that could be further improved. Configuring ArcGIS to run using AWS or another cloud compute engine would increase the compute resources and likely speed up model runtime as the current values were obtained from running the model on a regular computer off it's CPU.

Conclusion

In partnership with the World Bank, the Safer Schools team used medium-high resolution satellite/aerial imagery from Esri and Google Maps tile servers in combination with the arcGIS building footprints model in order to extract the area from a school in Kyrgyzstan. The model was designed to be able to run in a single process through the use of notebooks in ArcGIS Pro to simplify its use. After running 140 separate satellite images of schools the model predicted the area of the building footprint with a 59% with respect to the building's hand measured area. The output data of the model shows a trend that larger area buildings tend to have a higher accuracy due to the fact that smaller buildings are harder to correctly identify. The footprint data output from the model will be useful to estimate the size of the buildings in order to improve the schools' infrastructure.

References

[1] Rodriguez, A., Watkins, T., Popal, P., et al., 2021, Mar. School Safety Project Final Documentation. Cal Poly.

[2] https://github.com/chrieke/awesome-satellite-imagery-datasets

[3] https://storymaps.arcgis.com/stories/69fb21b744204d75a1f7146602a0b479

[4] https://azure.microsoft.com/en-us/blog/how-to-extract-building-footprints-from-satellite-images-using-deep-learning/

[5] Shackelford, A.K., Davis, C.H. and Wang, X., 2004, September. Automated 2-D building footprint extraction from high-resolution satellite multispectral imagery. In *IGARSS 2004. 2004 IEEE International Geoscience and Remote Sensing Symposium* (Vol. 3, pp. 1996-1999). IEEE.

[6] Li, W., He, C., Fang, J., Zheng, J., Fu, H. and Yu, L., 2019. Semantic segmentation-based building footprint extraction using very high-resolution satellite images and multi-source GIS data. *Remote Sensing*, *11*(4), p.403.

[7] Shi, Y., Li, Q. and Zhu, X.X., 2019, May. BFGAN—building footprint extraction from satellite images. In *2019 Joint Urban Remote Sensing Event (JURSE)* (pp. 1-4). IEEE.

[8] C. Ayala, R. Sesma, C. Aranda, and M. Galar, “A deep learning approach to an enhanced building footprint and road detection in high-resolution satellite imagery,” *Remote Sensing*, vol. 13, no. 16, p. 3135, 2021.

Useful Links

1. Deep Learning Model
2. Deep Learning Libraries