Identifying Area of Rooftops in Denver, CO, for Solar Policy

Project Proposal

Problem Statement

Climate change as a crisis that decisive action. Now. A huge part of that solution involves shifting our productions of energy away from fossil fuels and towards renewable sources of energy. Paul Hawken's famous Project Drawdown lists rooftop solar as a key strategy of meeting our ambitious climate goals. Their analysis suggests that rooftop solar can grow from 0.4% of global electricity generation to 7% by 2050, and "that growth can avoid 24.6 gigatons of emissions."

Rooftop solar is an increasingly appealing option for those looking to not only contribute to building a better world, but those hoping to save money in the long term through reduced energy bills and net metering. Policies exist to incentivize the application of rooftop solar, including several in Denver: City and County of Denver Elevation Loans, Solar* Rewards through Xcel, and the RENU Loan through the Colorado Energy Office.

But how would rooftop solar would it take to essentially "run" Colorado, the state where the sun shines 300 days a year? What about just single-family residential electricity needs?

In true engineering fashion, we can begin to estimate this answer by taking it to the extreme: if every rooftop in Denver was outfitted with PVs, how much electricity could we produce?

Since this is a data science project and not an engineering project, our process will be simplified. We'll use a standard rate of electricity produced per area of PV on a rooftop. One (very basic) rule of thumb is to say that every square foot of roof space generates 15 watts of solar energy (based on a solar panel with 15.8% efficiency).

According to the US Energy Information Administration, Coloradans use an average of 706 kilowatt hours in their homes each month. According to the U.S. Census, there are 320,545 houses in Denver County. With some basic math, we can estimate that the residential sector of Denver County uses roughly 225 GW of solar energy.

Therefore, we're looking to identify roughly 1.5 x 10^10 square feet of rooftops in Denver, or 538 square miles.

Solution

Given satellite imagery of Denver, we can train a convolutional neural network to correctly identify rooftop area.

- Use OSMnx Python package to extract building footprints as a shapely file
- Overlay building footprints with images given projection and lat lon coordinates (stored in file name)
- Use tool from Tony Szedlak to clean up the building footprints
- Finish labeling the images
- Build a data generator and a NN in TensorFlow
- Train the NN

Data

The City and County of Denver generously provides open source data via https://www.denvergov.org/opendata. For this project, I am using their 2004 dataset of satellite imagery of Denver County. These data consist of almost 2,500 JP2 files.

From the Open Data website:

> Color, 6 inch resolution aerial photography for the City and County of Denver acquired in 2004. True-ortho correction is applied to all structures 4 stories and above in designated areas including all bridges and highway overpasses.
>
> Coverage are includes a 359 square mile area encompassing the City and County of Denver, City of Glendale, City of Littleton, and the Denver Water Service Area. This project does not include DIA.
>
> Spatial reference is NAD 1983 HARN StatePlane Colorado Central FIPS 0502.

For the benefit of the analysis, I converted these JP2 files into JPEGs using XnConverter for Linux.

Deliverables

Code with my exploratory data analysis, image processing, and neural net; along with a written report/narrative.

Exploratory Data Analysis Results

Each image has 2,640 rows and columns and three channels (RGB) and there are a total of 2,553 images in the data set. The max value in an RGB image is 255 (and a min of 0).

Type of the image : <class 'numpy.ndarray'>
Shape of the image : (2640, 2640, 3)
Image Height 2640
Image Width 2640
Dimension of Image 3

We can plot the channels against each other and look for any kind of correlation. Below I plot the first chanel (0) against the third channel (2) for the first image in the sest (Figure 1) and the fifth image (Figure 2. The histogram shows that there are a lot of pixels have an equal value of R and B (grey).

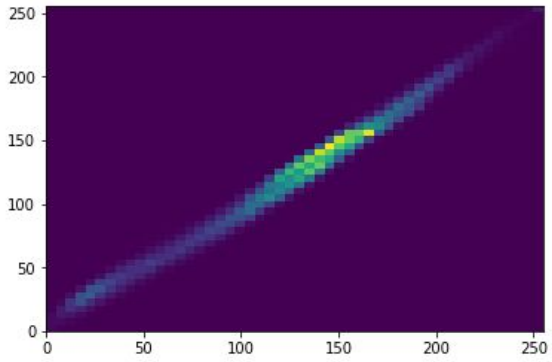

Figure 1. Histogram of R and B values in the first image

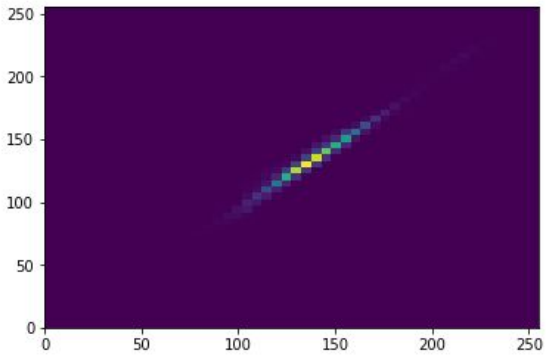

Figure 2. Histogram of R and B values in the fifth image

We can compare channels (in this case, the third channel) between the two images. The histogram below shows that the most common shared values lie in the middle... again, grey.

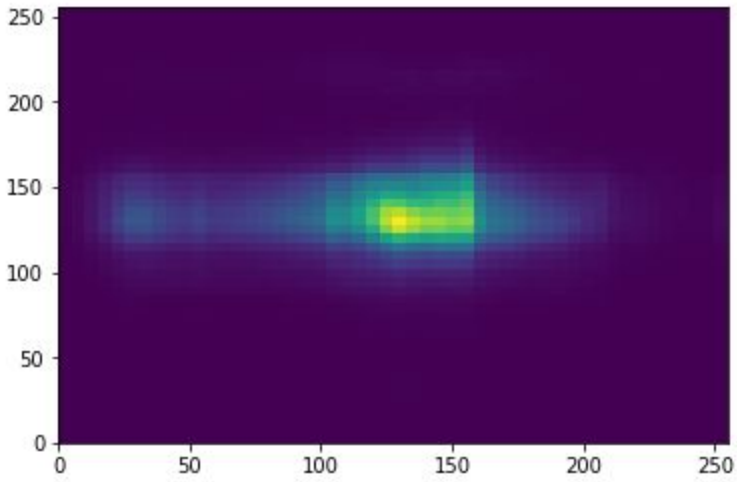

Figure 3. The comparison of the B channel between the 1st and 5th image.

We can experiment with compressing an image by applying a quick k-means clustering on the image with a k of 5. This gives us an impression of what kinds of features we might be able to extract from the image set.

From the result below, we see a dark purple for the roads, yellow for the green spaces, and blues and greens for the built environment.

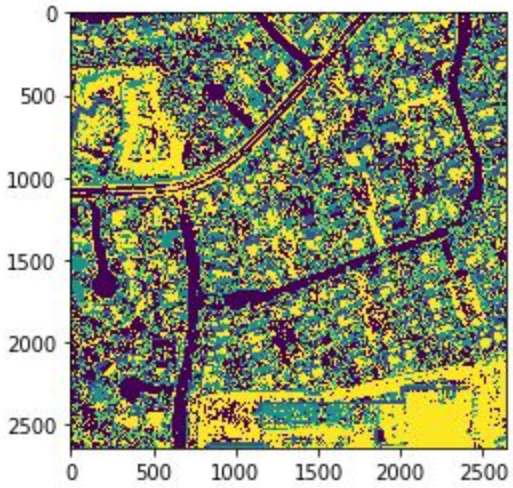

Figure 4 K-means clustering with k of 5 on the first image.

We can plot the histogram of the color values for each channel in the image. All three channels seem to correlate strongly with each other, indicating a lot of grey values.

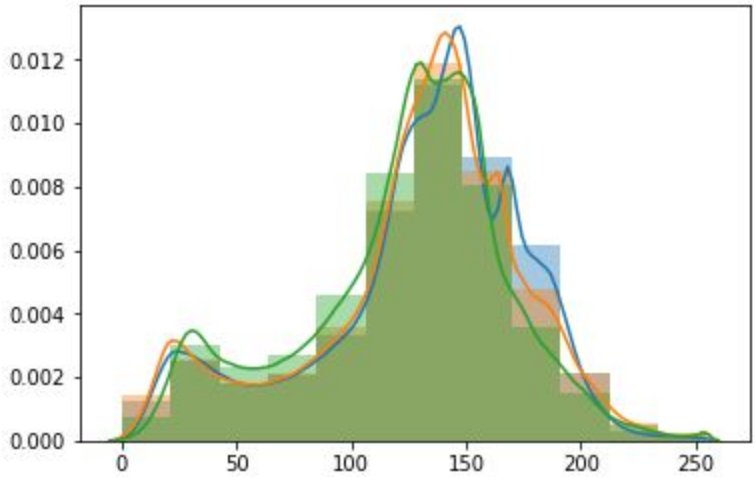

Figure 5. Histogram of 3 color channels for the first image

We can play more with the RGB channels and split the image accordingly.

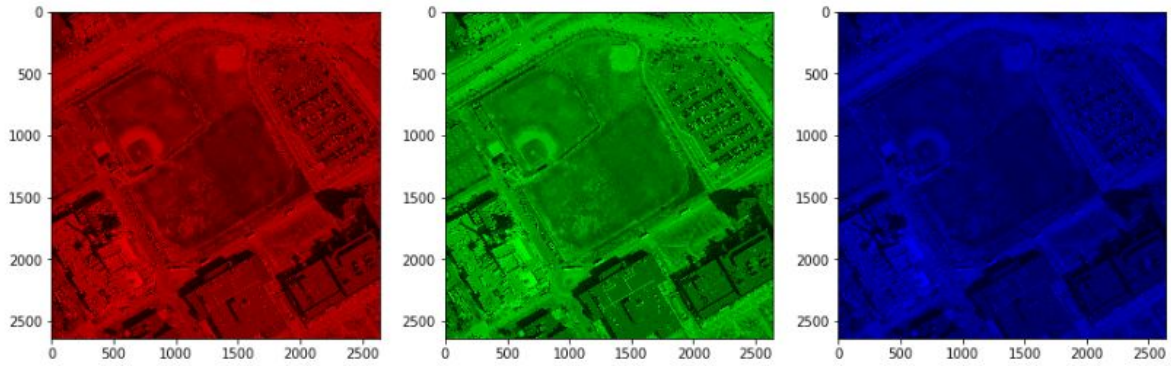

Figure 6. The split R,G, and B channels for a random image (1002 image).

Now we can get into image convolution. First, we can “blur” the image with a convolution matrix that averages the nearby pixels,

Figure 7.The 1006th image convolved by 9, 15, 30, and 60

Figures 8-11 show varying convolution filters applied to the same image. We see that throughout these filters, we're starting to segment out important features. In this image, we see the outline in the parking lot, the buildings, and the road.

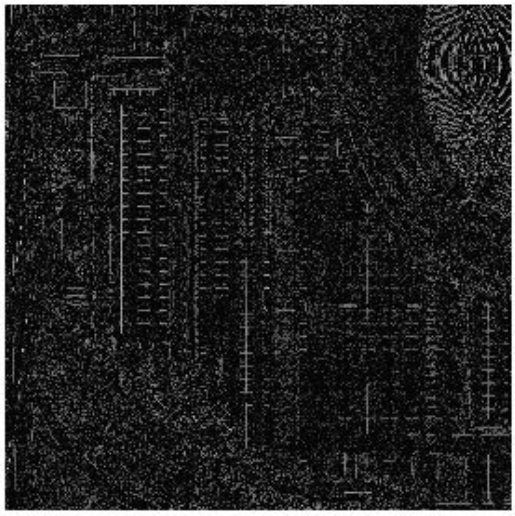

Figure 8. Outline kernel

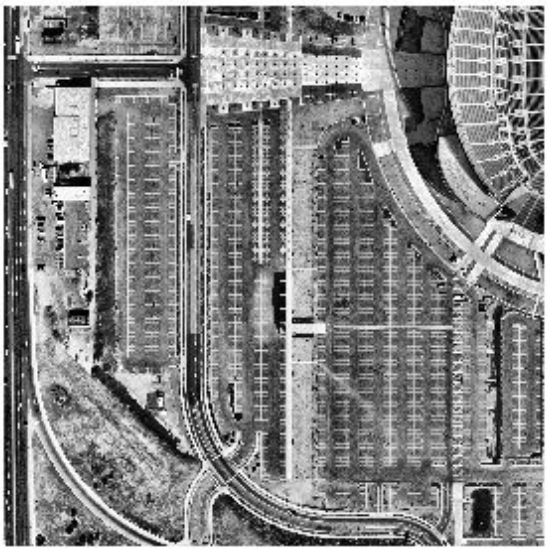

Figure 9. Gaussian window

Sobel 10. Sobel kernel

Figure 11.Gaussian Filter

Conclusions

I am so lucky to be working with such high quality images. With a 6 inch resolution, our EDA shows that even basic image manipulation can generate building outlines. These images are mostly grey, and I don't expect color to be a big deciding factor in determine what is and what is not a rooftop. We could convert these images to black and white.

References

1.Novel Approach for Rooftop Detection Using Support Vector Machine
https://www.hindawi.com/journals/isrn/2013/819768/

(2) Automatic Rooftop Detection Using a Two-stage Classification
http://ijssst.info/Vol-15/No-4/data/4923a285.pdf