Homework 3
Satellite/Aerial Image Retrieval

2019/04/06

Jinchen Jiang

Zhongzhu Ding

Content

Objective

Knowledge Area

Approach

Result

Reference

Objective

INPUT

Given latitude and longitude bounding box

Download aerial imagery from Bing maps tile system automatically

TASKS

Cut the image

Based on the latitude and longitude

Knowledge Area

Bing Maps Tile System

Bing Maps Tile System utilizing addressing scheme, projections, addressing scheme of the map tiles.

Knowledge Area

Map Projection

Aim

Make the map seamless, make the map line up more properly.

Mercator Projection

It's a **conformal projection**, which means that it preserves the shape of relatively small objects. This is especially important when showing aerial imagery, because we want to avoid distorting the shape of buildings. Square buildings should appear square, not rectangular.

It's a **cylindrical projection**, which means that north and south are always straight up and down, and west and east are always straight left and right.

Mercator Projection

Knowledge Area

Ground Resolution and Map Scale

Ground Resolution

The **ground resolution** indicates the distance on the ground that's represented by a single pixel in the map.

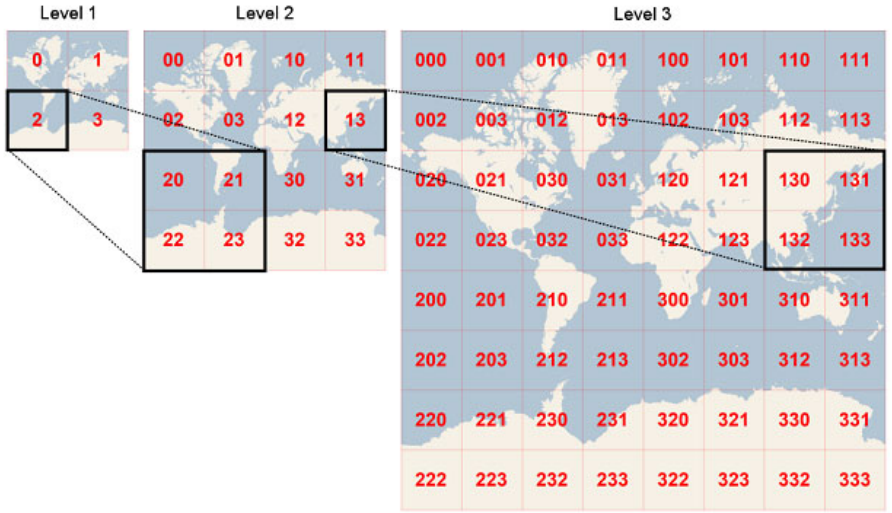
map width = map height = $256 * 2^{level}$ pixels

ground resolution = cos(latitude * pi/180) * earth circumference / map width

Map Scale

The map scale indicates the ratio between map distance and ground distance, when measured in the same units.

map scale = 1 : ground resolution * screen dpi / 0.0254 meters/inch



Resolution Conversion

Knowledge Area

Pixel Coordinates

The pixel at the upper-left corner of the map always has pixel coordinates (0, 0). The pixel at the lower-right corner of the map has pixel coordinates (width-1, height-1), or referring to the equations in the previous section, $(256 * 2^{level}–1, 256 * 2^{level}–1)$.

sinLatitude = sin(latitude * pi/180)

pixelX = ((longitude + 180) / 360) * 256 * 2^{level}

pixelY = (0.5 – log((1 + sinLatitude) / (1 – sinLatitude)) / (4 * pi)) * 256 * 2^{level}

Knowledge Area

Tile Coordinates and Quadkeys

To optimize the performance of map retrieval and display, the rendered map is cut into tiles of 256 x 256 pixels each. As the number of pixels differs at each level of detail, so does the number of tiles.

map width = map height = 2^{level} tiles

To optimize the indexing and storage of tiles, the two-dimensional tile XY coordinates are combined into one-dimensional strings called quadtree keys, or “quadkeys” for short. Each quadkey uniquely identifies a single tile at a particular level of detail, and it can be used as an key in common database B-tree indexes. To convert tile coordinates into a quadkey, the bits of the Y and X coordinates are interleaved, and the result is interpreted as a base-4 number (with leading zeros maintained) and converted into a string.

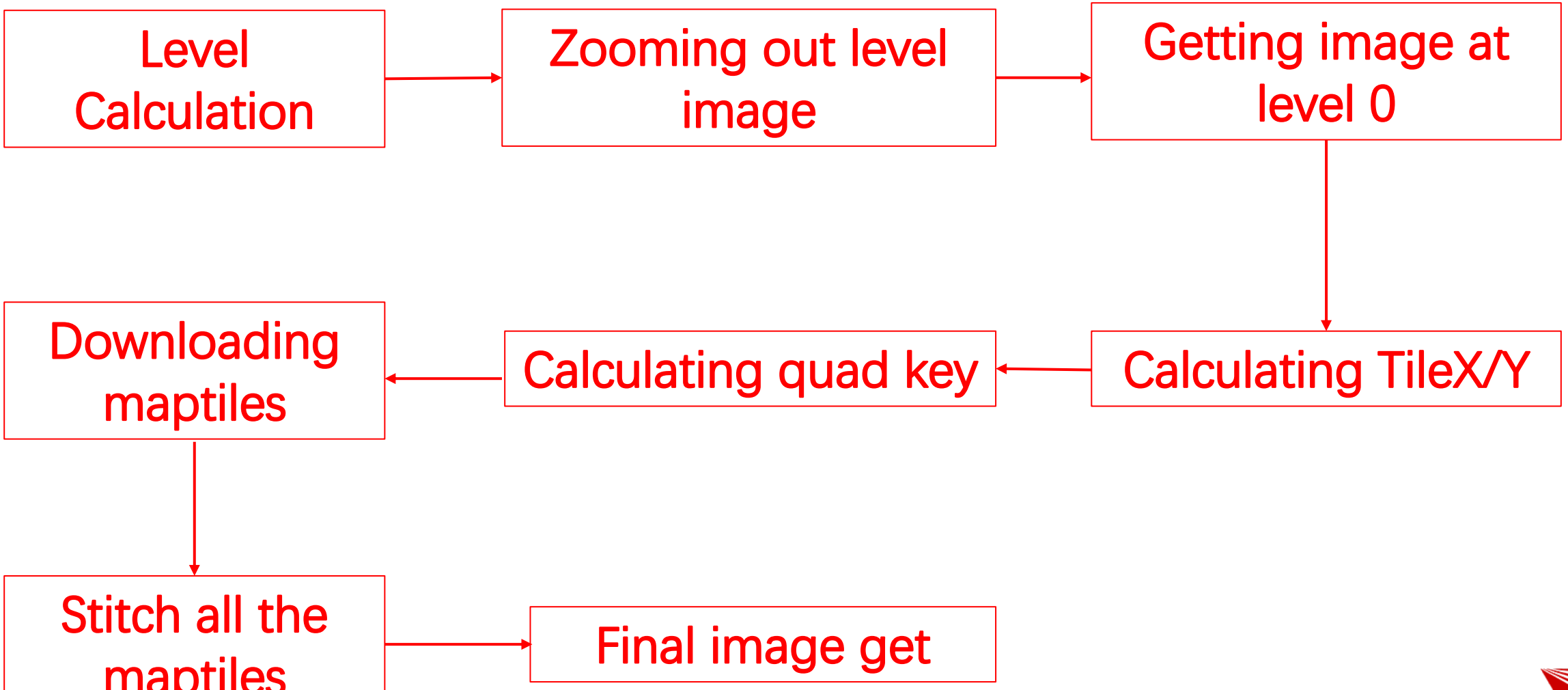
Knowledge Area

Tile Coordinates and Quadkeys

Finally, quadkeys provide a one-dimensional index key that usually preserves the proximity of tiles in XY space. In other words, two tiles that have nearby XY coordinates usually have quadkeys that are relatively close together. This is important for optimizing database performance, because neighboring tiles are usually requested in groups, and it's desirable to keep those tiles on the same disk blocks, in order to minimize the number of disk reads.

Each tile is given XY coordinates ranging from (0, 0) in the upper left to (2^{level}–1, 2^{level}–1) in the lower right.

Approach


Level Calculation
Zooming out level image
Getting image at level 0
Calculating TileX/Y
Calculating quad key
Downloading maptiles
Stitch all the maptiles
Final image get

Run the Program

Aerial.py:

python Aerial.py <lat1,lon1,lat2,lon2>(no space)

lat1,lon1,lat2,lon2:

-the longitude and latitude of the bounding box.

-left top corner :lat1,lon1

-right bottom corner: lat2,lon2

output:

- image.png
  cutImage.png

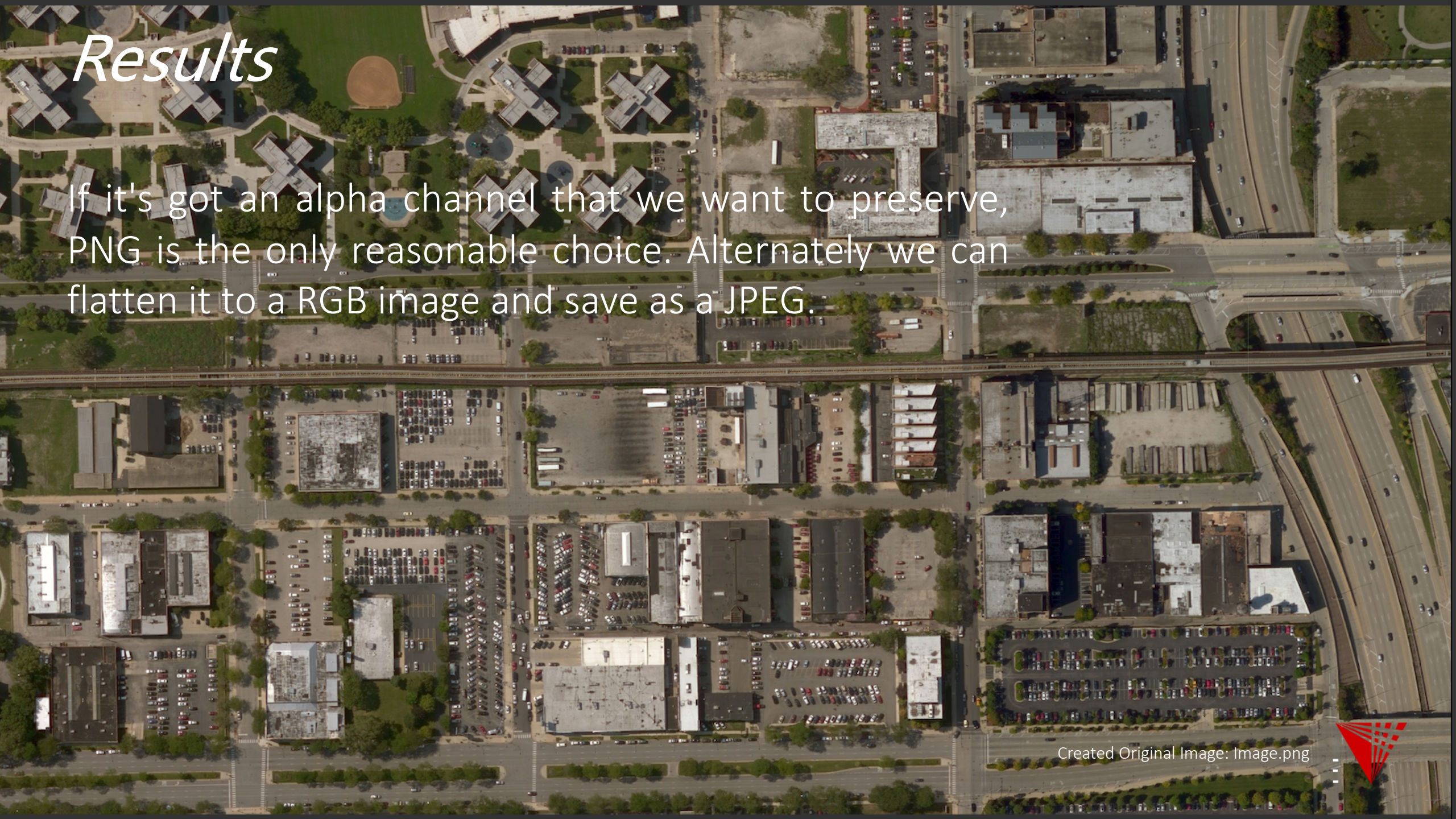
Example: python Aerial.py 41.847934,-87.62847,41.839658,-87.623400

```
dhcp19:CS513 Geospacial Visualization zhongzhuding$ python Aerial.py 41.847934,-
87.62847,41.839658,-87.623400
The longitude and latitude is
<41.847934, -87.62847, 41.839658, -87.6234>
The bounded box level is 20.
Image created.
68877056 99800064 68877205 99800212 68880985 99808495
Cut image created.
dhcp19:CS513 Geospacial Visualization zhongzhuding$
```

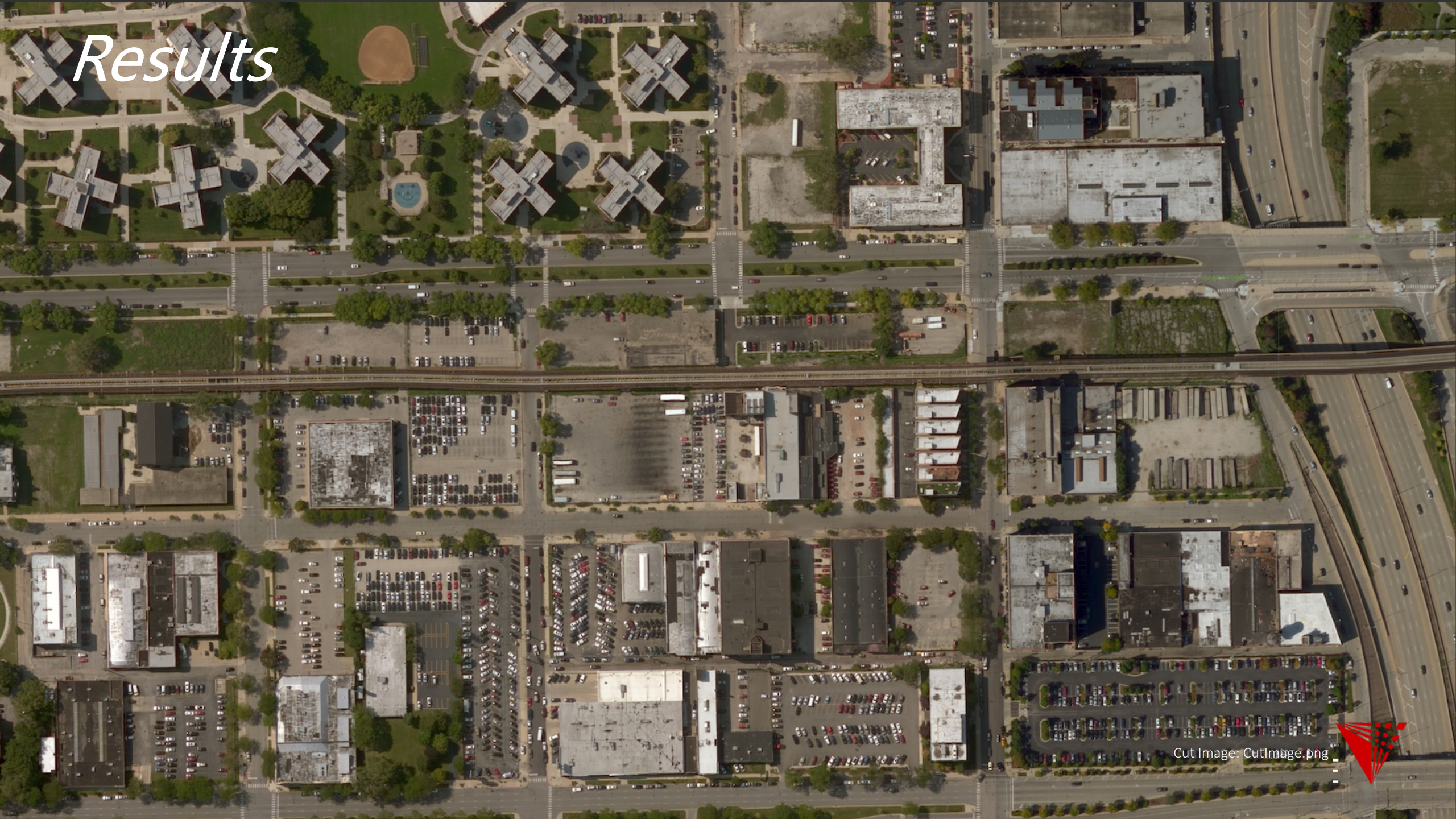
Program in Progress

Results

If it's got an alpha channel that we want to preserve, PNG is the only reasonable choice. Alternately we can flatten it to a RGB image and save as a JPEG.



Results



Reference

1. Bing Map Tile System “https://msdn.microsoft.com/en-us/library/bb259689.aspx”
2. https://docs.microsoft.com/en-us/bingmaps/articles/bing-maps-tile-system