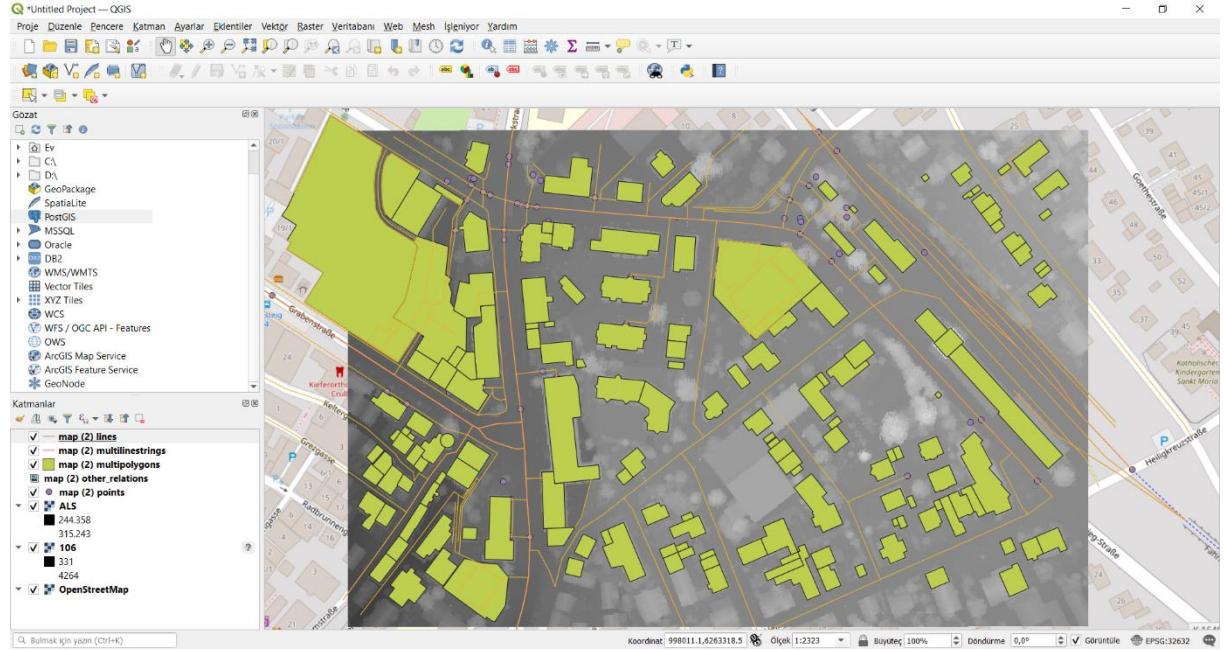


Photogrammetric Image Analysis

Midterm Project Report:

- 1) I opened the OSM and click the export link (<https://www.openstreetmap.org/export#map=18/48.93216/8.96213&layers=O>) after that I wrote to required longitude and latitude of our polygon. And I downloaded the vector data. I visualized on the QGIS with our lidar dsm data.



- 2) Answers;

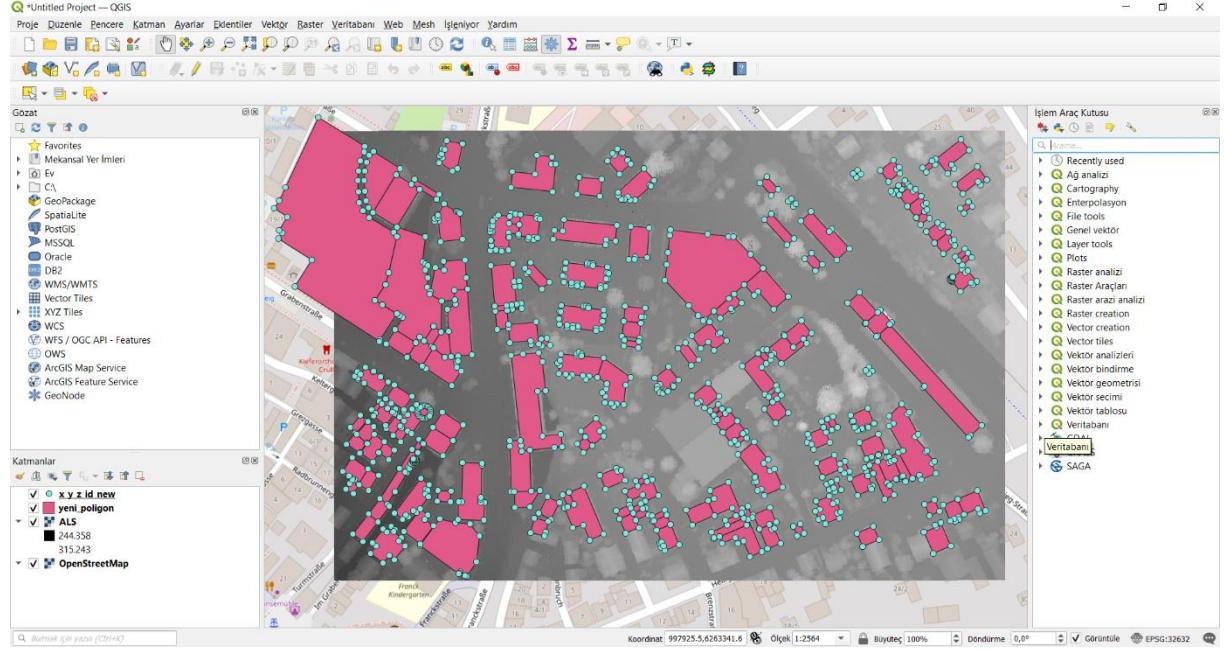
2.1) Firstly I checked vector data's (map(2) multipolygons) coordinate system. After that I clicked order, vector > Geometry Settings > Extract vertices to get vertex points. I should have seen zonal value in the vertex points' attribute table. So all vertex points and polygons must be the same coordinate reference system with ALS raster data (EPSG:32632 - WGS 84 / UTM zone 32N). (figure 2.1)

2.2) After that I clicked order, raster analysis > zonal statistics for our polygons. (figure 2.2)

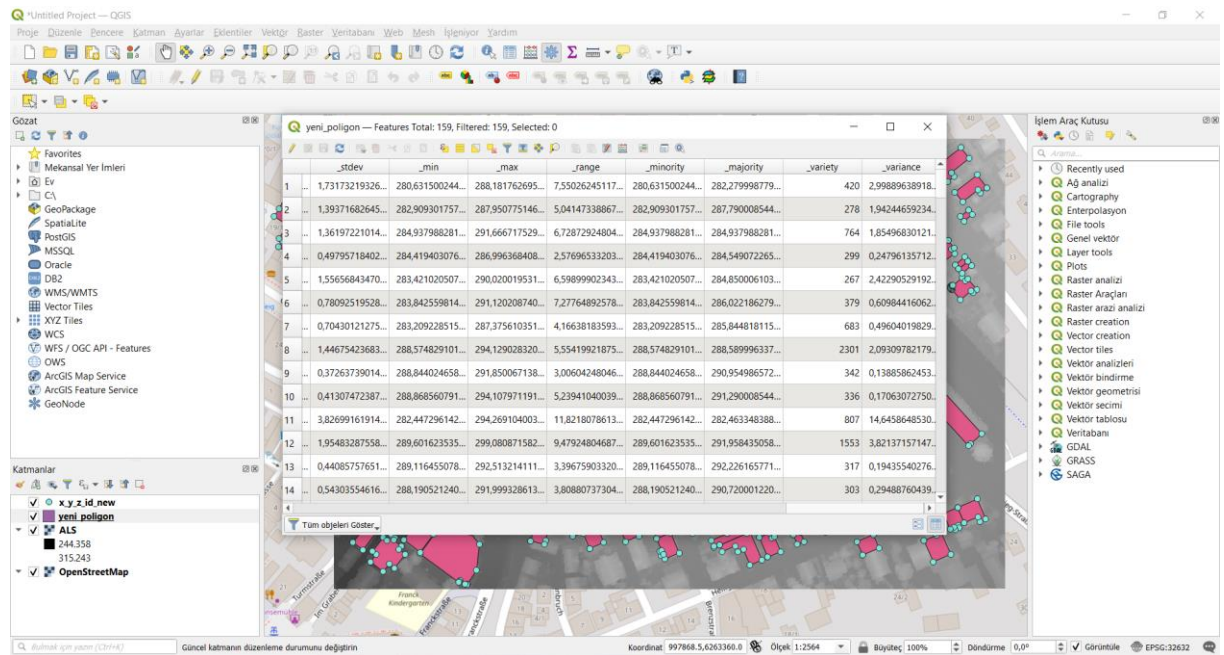
2.3) I clicked order, plugins > manage and install plugins > install point sampling tool. In this way, I can use point sampling tool. I clicked order, plugins > analyses > points sampling tool > select point's osm_id, name, Als band 1. So in this way finally I attained vertex point's z values .(figure 2.3)

2.4) I wanted to x and y coordinates so I clicked order, vector > Geometry Settings > Add geometry feature.(x_coord,y_coord in the figure 2.3)

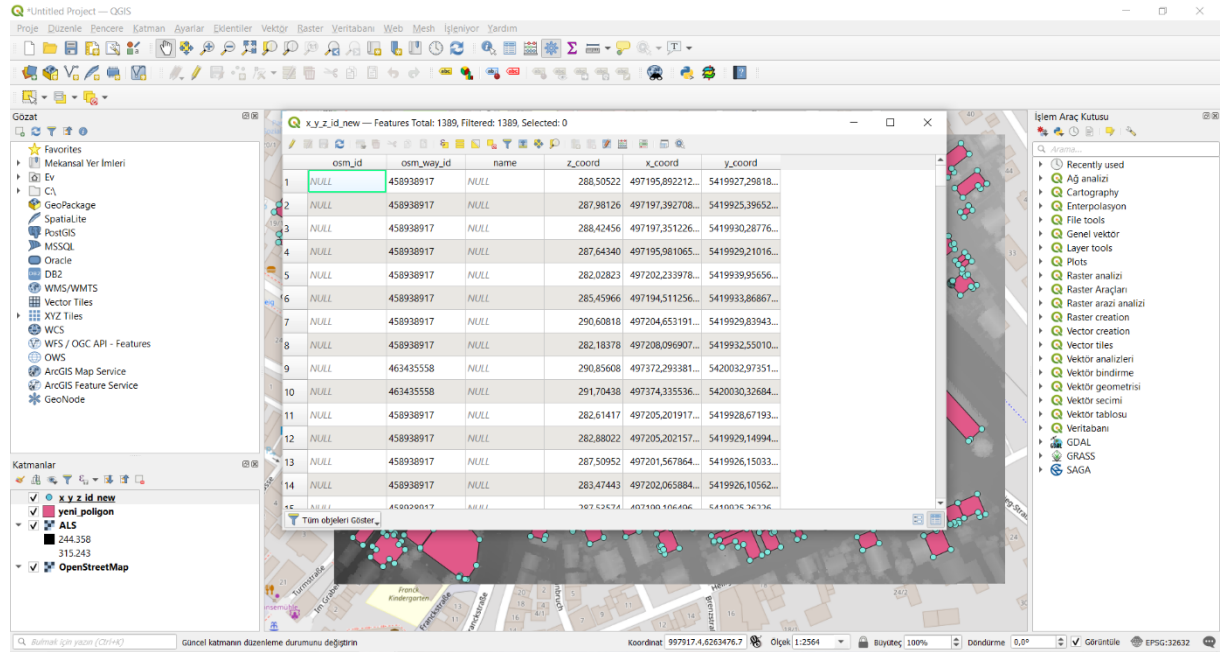
(Figure 2.1)



(Figure 2.2 shown new columns thanks to zonal statistic)

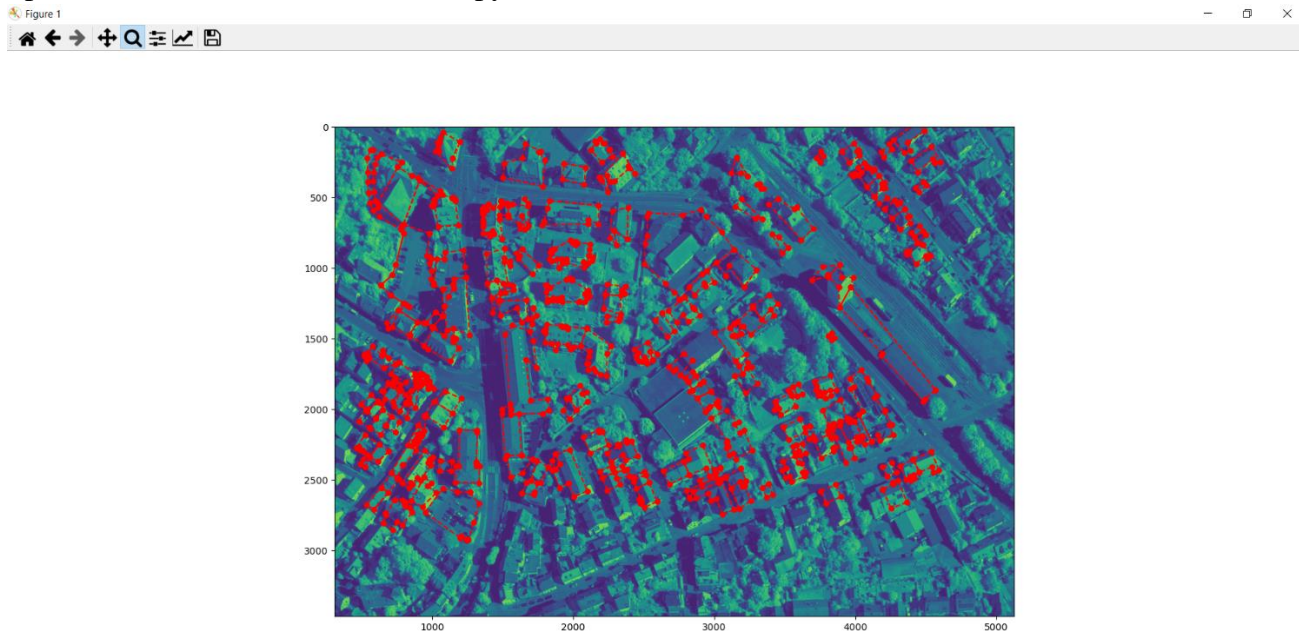


(Figure 2.3 z value,x and y coordinates of vertex points)



3) My answer is in the third_fourth_answers.py file. I created projection matrix with homogenous coordinates and all operations explained as command line. Also world coordinates save in the x_y_z_id. .xlsx excel file to use in the code.

4) I plotted in the third_fourth_answers.py file. Zoom like this..



5) Please comment/discuss/explain the problems/reasons/causes you face in the output building boundaries projected on image 106.jp2.

5.1) Firstly, as we move away from the principle point, we lose orthogonality. That is why I see not only the roofs of buildings, but also the facades of buildings. Therefore, we have seen that our vector points differ from the roof corner positions of the buildings.

5.2) When I connected the x, y, and z coordinates for each excel file's row (line) in the image, it tried to connect the last connecting or plotting point for each polygon towards the starting point of another polygon. In short, the polygon assignment of our excel file according to the coordinates in the row (line) order caused some problems. So, I put condition about the polygon. This condition is `osm_way_id` (buildings id number). In this way all points are not mixed with each other and all polygons matched and plotted their points.

5.3) When I downloaded osm buildings vector data, some buildings were outside the lidar data in the image. So, I couldn't get z information of the vertex points. Unfortunately, I had null value of these Z coordinates in the excel file. And so, it created some problems when connecting image 106.jp2 and vector data information of the buildings.

5.4) Our lidar image is up to date, but osm vector data is out of date. Therefore, in our 106.jp2 image, we see that some buildings do not have vector drawings.

5.5) Finally, I had some problems about the image format(.jp2). Rasterio library can open but opencv or another image library are not support this format. So, I used matplotlib to see the vector drawings on the 106.jp2. But the 106.jp2 image cannot open their colors, it used the default matplotlib graphic colors to visualize the image. When I converted .jp2 to .jpg, image appears darker than the original.