THE GEO PROJECT
GEOSPATIAL EDUCATION & OUTREACH

www.geoproject.msstate.edu

GEO TUTORIAL

#QGIS

Dealing with Coastal Flooding series, part 2:

SPATIAL PREDICATES: PREPARING RESIDENTIAL DATASET

Krzysztof Raczynski
Kate Grala
John Cartwright

Geosystems Research Institute
Mississippi State University

MAY 2025

This work was supported through funding by the National Oceanic and Atmospheric Administration Regional Geospatial Modeling Grant, Award # NA19NOS4730207.

Geosystems Research Institute, Mississippi State University, Box 9627, Mississippi State, MS 39762-9652

The Geospatial Education and Outreach Project (GEO Project) is a collaborative effort among the Geosystems Research Institute (GRI), the Northern Gulf Institute (a NOAA Cooperative Institute), and the Mississippi State University Extension Service. The purpose of the project is to serve as the primary source for geospatial education and technical information for Mississippi.

The GEO Project provides training and technical assistance in the use, application, and implementation of geographic information systems (GIS), remote sensing, and global positioning systems for the geospatial community of Mississippi. The purpose of the GEO Tutorial series is to support educational project activities and enhance geospatial workshops offered by the GEO Project. Each tutorial provides practical solutions and instructions to solve a particular GIS challenge.

SPATIAL PREDICATES: PREPARING RESIDENTIAL DATASET

Krzysztof Raczynski [1, 2, 4, 5, 6, 8] chrisr@gri.msstate.edu
Kate Grala [3, 7] kgrala@gri.msstate.edu
John Cartwright [7, 9, 10, 11] johnc@gri.msstate.edu

Geosystems Research Institute
Mississippi State University

CRediT: 1: Conceptualization; 2: Methodology; 3: Verification; 4: Resources; 5: Data Curation; 6: Writing - Original Draft; 7: Writing - Review; 8: Visualization; 9: Supervision; 10: Project administration; 11: Funding acquisition

REQUIRED RESOURCES

– QGIS 3+

FEATURED DATA SOURCES

– Click here to access dataset used in this tutorial (127.6 MB).

OVERVIEW

Coastal areas across the United States face increasing challenges from changing water levels, which can lead to more frequent flooding and infrastructure strain. In communities like Bay St. Louis, Mississippi, rising water can make roads impassable, damage property, and disrupt daily life—posing serious concerns for homeowners and local economies.

As part of a planning team, your role is to assess how changing sea levels may impact the safety, infrastructure, and long-term growth of this Gulf Coast community. The focus is on protecting property, ensuring economic stability, and strengthening community resilience. This is the theme of the *Dealing with Coastal Flooding* tutorial series, which includes the following topics:

– Part 1: Creating Raster DEM from LiDAR Data
– **Part 2: Spatial Predicates: Preparing Residential Data**
– Part 3A: Using Unsupervised Machine Learning for Land Use Land Cover Classification
– Part 3B: Using Supervised Machine Learning for Land Use Land Cover Classification
– Part 4: Hydrologic Raster Preparation: Resampling and Burning Stream Network
– Part 5: Generating Flooding Extent with Raster Calculator
– Part 6: Calculating Spatial Statistics of Inundated Areas
– Part 7: Creating 3D Maps of Flooding Projections
– Part 8: 3D Map Animations
– Part 9: Creating and Animating Timeseries

In the previous part, we processed LiDAR data to create a digital elevation model (DEM) and a digital surface model (DSM) for our project. This tutorial covers spatial predicates as we prepare residential data for analysis. Make sure to check the remaining tutorials in the series to learn more about the entire analysis process.

DATA

In this tutorial, we will utilize two datasets that contain features related to buildings. We will cover how to get the data manually in the *Preparing Building Data* section, but you can also use the Featured Data Sources link above to download the complete dataset. Remember that the .zip archive with data must be extracted before use. If you wish to go through all the steps in this tutorial, only add the DEM file provided with the tutorial data to your project. Building layers saved in geopackages are provided for you if you do not wish to go through Preparing Building Data section, or if there are problems with downloading the features through the used plugin.

TOPOLOGY

Topology is an area of geospatial science that deals with relationships between geographic features. Spatial relations describe how two or more geometric features relate to each other in space. Simply put, when we have two features, A and B, they determine whether for example A touches B, crosses B, or contains B. The *Dimensionally Extended 9-Intersection Model* (*DE-9IM*), which constitutes a topological standard, provides us with standard ways the features can relate to each other, called ***spatial predicates***. Below are the definitions of each predicate. Don't worry if some of the definitions appear confusing due to their proximity to one another. When you read through the definitions, look at Fig. 1 and check the brackets next to each definition; they will tell you which of the relations in the picture represent the given predicate:

- *Equals* – features are geometrically the same [no features on the figure are the same];
- *Disjoint* – features are separate and have no common points [G is outside all other features];
- *Intersects* – features share at least one point [A and B, A and C, A and D, A and E, A and F, D and E intersect];
- *Touches* (also *Meets*) – features have at least one common edge point, but their interiors are separate [A and B, E and D touch];
- *Contains* – feature is located inside the interior of other feature and has common at least one interior point (describes the perspective of the outer feature) [A contains D, A contains F];
- *Within* – similarly to contains, describes relation when one feature is located inside the other, but describing the opposite perspective [D is within A, F is within A];
- *Overlap* – features share a common area but are not completely contained by each other [A and C overlap]
- *Crosses* – feature crosses another feature if some interior points are common, and it has a lower dimension (line-polygon relation) [E crosses A].

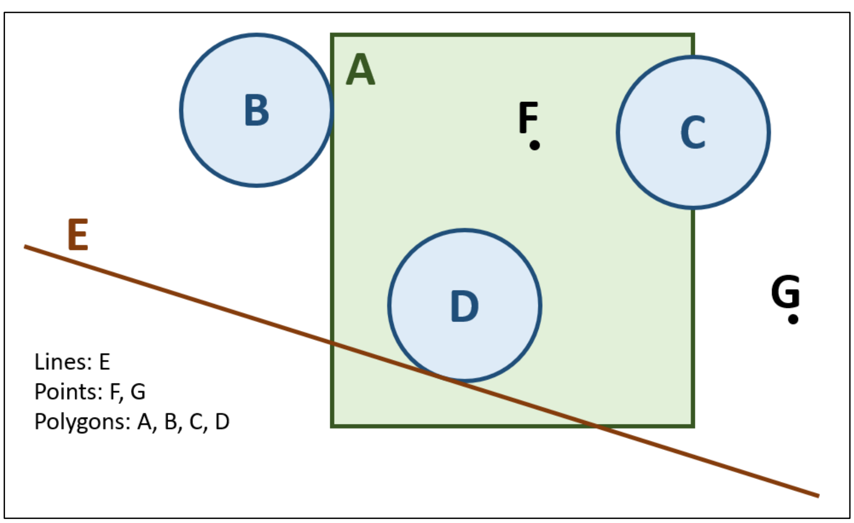

Fig. 1. *Different spatial relations between features.*

PREPARING BUILDING DATA

Now that we know what spatial predicates are, we can start preparing residential information for our project. There are two main ways to obtain building data for our analysis, and we will use both. The first one involves downloading the official building footprint dataset from the *Mississippi Automated Resource Information System* (MARIS). Click to download: [Building Footprints data](). Once you download the dataset, remove the polygons outside the study area. In such a case you should end up with 9,171 polygon features.

The second method entails utilizing a QGIS plugin, which enables you to quickly extract any data from the *OpenStreetMap* dataset at no cost. To install the plugin, click on the *Plugins* menu and choose *Manage and Install Plugins*. In the *Plugins* window, switch the tab to *All* if not opened by default, and search for *QuickOSM*. Select the plugin from the list and proceed to install it. After successfully installing the plugin, close the window and open the newly installed *QuickOSM* by clicking its icon on the toolbar. In the plugin window (Fig. 2), write ***building*** in the *key* cell, then adjust the extent setting to ***layer extent*** and select ***DEM_BayStLouis***. Expand *advanced* options and turn off ***points*** and ***both line*** types, leaving only ***multipolygons***. Click *Run query*[1].

After successfully adding the building layer to the map, close the plugin window. Note that the buildings extracted from OSM have 13,736 features, which seems like a more precise dataset. It's important to keep in mind that OSM, being a community-maintained datasource, is not official and may contain errors or outdated information. Let's organize the coordinate reference system of the layer to match our project. Right-click on the layer in the *Layers* panel and select *Export*, then *Save Features As… option*. Choose the path to save the file, then change CRS to ***NAD83(2011) / Mississippi East (ftUS)*** (EPSG code: 6507).

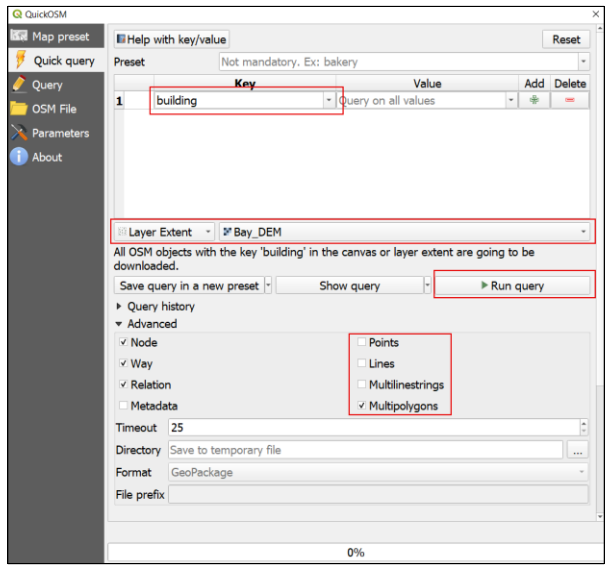

Fig. 2. *QuickOSM plugin for QGIS allows for fast feature download from OSM database.*

If you closely examine the differences between the two datasets, you will notice that both sets contain some missing elements, some overlapping sections, and sometimes polygons with different geometries. We will now create a dataset of common buildings that will join both datasets, and by doing so, we will practice using spatial predicates.

First, let's filter out some of the unnecessary data in the *OSM building dataset*. There are a number of small polygons that represent structures such as garages, apartments, or sheds. Small area structures are unlikely to serve as residential or commercial buildings, thus they can be removed. The smallest houses in the United States have around 150-300 ft^2 (13.9-27.9 m^2). In our scenario, we will apply a 200 ft^2 criterion (18.6 m^2). Additionally, we will apply a second filter that will keep the cases of small objects if there is a house number assigned in the address, which might indicate that even though the feature is of a small area, it might just be a tiny house. Let's first calculate the area for each feature. Open the layer *attribute table* and click on the *Field Calculator* icon. Create a new field with the *name* ***area*** and a ***decimal*** *type*. In the *expression,* write:

```
$area
```

Confirm by clicking *OK*. Save the layer edits and turn off the edit mode.

[1] If you experience *Bad request OverpassAPI* error, verify that your DEM file has reference system assigned. If you will see a question mark next to DEM layer in the Layers panel, right-click on the layer and select Properties, then under Source assign Coordinate Reference System to NAD83(2011) / Mississippi East (ftUS) EPSG: 6507. If you still experience the error, you can change tab in the QuickOSM plugin window to Parameters and change different API from the list.

To limit the features in the layer, without modifying it or creating a new layer, we can use the *Feature Filter Query Builder*. To open it, right-click on the OSM buildings layer and select *Properties*, then switch the active tab to *Source*. You will see the *Provider Feature Filter* and *Query Builder* button at the bottom of this property. Click the button. Here, we construct an expression to apply to the feature filter. To keep the objects larger than 200 ft^2 or with house numbers, write the following expression in the *Provider Specific Filter Expression* field (note the attribute *addr:housenumber* might have a bit different name, depending on the plugin version; you can verify this in the *fields* view of the *Querry Builder* or in the *properties* of the layer, in *Fields* tab):

```
"area" >= 18.5 OR "addr:housenumber" IS NOT NULL
```

The number of features on our OSM layer has been reduced to about 12.5 thousand.

Now, let's deal with the overlapping features. In both layers, there are features that describe the same object, but their shapes typically differ slightly. Let's combine both data sources to create a shape, which will be a combination of overlapping features. To accomplish this, we will utilize the *Extract by Location* tool found in the *Vector Selection* toolbox.

If you don't have the *Processing Toolbox* visible, navigate to the *Processing* menu, then select *Toolbox*, or use the key combination [CTRL] + [ALT] + [T]. Select the *Extract by location* tool.

First, set the *extract features from* to the ***OSM buildings*** layer. We will compare these features to MARIS buildings and select only the features that have common areas. Let's consider different options for spatial predicates. Out of the list: *touch*, *contain*, *equal*, and *are within* are too narrow to include all the cases we want to consider; therefore, the final choice is between *intersect* and *overlap*. If you recall, the *overlap* predicate requires that features share a common area, while *intersect* requires only a common point. In our case, this will come down to the question if we want to include the features that touch each other. If not, use *overlap* (resulting in 7,341 features); if yes, use ***intersects*** (resulting in 7,411 features). You can choose either approach, but in this tutorial, we will continue with ***intersect***. Finally, set the *by comparing to the features from* to the ***MARIS buildings*** layer. Click *Run*.

Once the algorithm finishes, go back to the *Parameters* tab (you were switched to the *Log* settings) and switch the places of the ***OSM*** and ***MARIS buildings*** layers (Fig. 3). These operations will result in two extracted layers, one representing a common structure from the *OSM layer* and the other from the *MARIS layer*. You may think the number of features will be the same but remember that sometimes two or more features in one layer overlap with one feature in the other layer.

Let's now merge these two layers into one. You can do this by either selecting the *Vector* menu, then *Geoprocessing Tools*, and selecting *Union* (select the two layers as *input* and *overlay input*, then *run* the

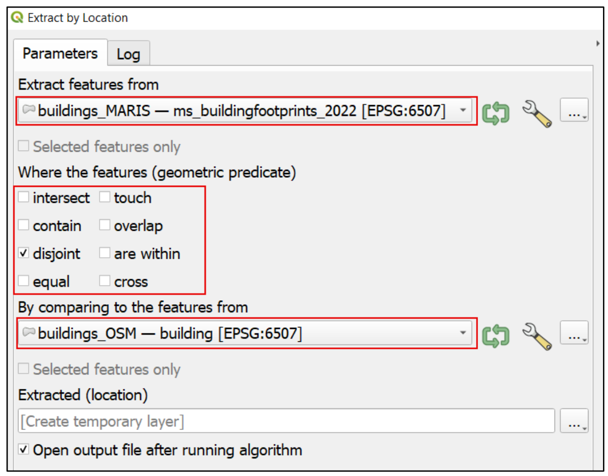

Fig. 3. Using spatial predicates allows to quickly organize data between two layers based on their spatial relation.

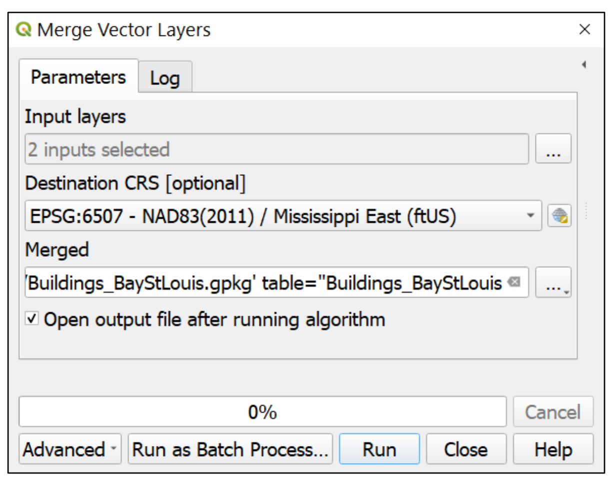

Fig. 4. Using Merge Vector Layers tool allows to quickly combine multiple layers into one.

algorithm), or by using the *Merge Vector Layers* tool available in the *Processing Toolbox* under *Vector General* (Fig. 4, select both layers as *input layers* and click *Run*).

The last step is to combine the intersecting features to update building shapes. To do so, click on the *Vector* menu, then in the *Geoprocessing Tools* choose *Dissolve*. As the *input layer*, set the recently ***merged layer***, then under *advanced parameters*, ensure the ***keep disjoint features separate*** option is selected. If you don't use this setting, the result will be one multipart feature, and you will have to break it into single parts. *Run* the tool which will create a new layer called ***Dissolved***. We have now successfully dealt with all common features in both layers.

In the next step, we will grab all the remaining features from both building layers that were not transferred so far. Right-click on the *Extract by location* tool (*Processing Toolbox*, *Vector selection*) and select *Execute as batch process* (if you don't know about batch processing, check our tutorial [Batching GIS Tasks: a Way To Speed Up Repetitive Procedures]). Set two rows and use the original layers with ***MARIS buildings*** in one and ***OSM buildings*** (Fig. 5) in the other as *extract features from*. In the *geometric predicate* set only ***disjoint*** (a quick recall: *disjoint* finds features that do not have common points). Finally, set *by comparing to* as ***Dissolved*** layer. Set the output names (e.g., as *Maris_extracted* and *OSM_extracted*), check *load layers on completion*, and run the tool.

Now that we have handled all the buildings in both datasets, it's time to merge the three layers into a final one. Select *Merge Vector Layers* from the *Vector General* toolbox. In the *input layers*, choose ***both*** the ***extracted*** layers as well as the ***Dissolved*** one. Set the destination CRS to ***EPSG: 6507*** (same as before), then save the buildings to a new geopackage or shapefile (called e.g., ***All_buildings***). The newly created layer should have about 14 thousand features.

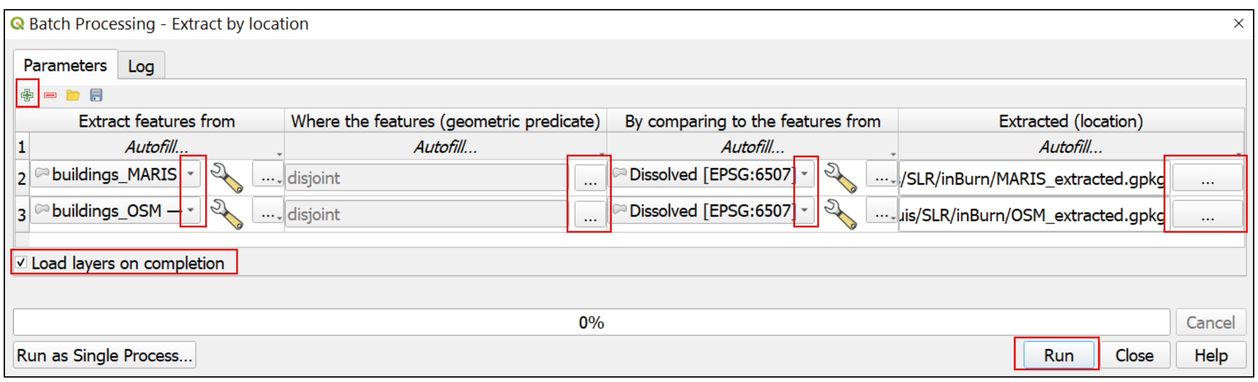

Fig. 5. *Batching Extract by location tool allows to run the same tool in multiple inputs at the same time.*

EXERCISE

Now that you know about different predicate types, try to assign the residents' data to all residential buildings. To do so, use the point data layer ***building_tags*** (available in **Featured Data Sources**). This layer contains 853-point features that indicate buildings of commercial use within our study area (Fig. 6). Additionally, let's make a further assumption that the houses of areas lower than 300 ft^2 are of the not-permanent residential type. Use this information to validate which of the buildings are of residential, non-permanent residential, and non-residential types. Use the information from the *US Census Bureau* that indicates there are 2.08 persons per household on average in the area. The average house size in the area is about 1950 ft^2. Once the residential buildings are identified, use the census data to make an estimate of how many people live in each house based on its area (you can check our [Calling Algorithms From Field Calculator In QGIS] and [Unlock a Hidden Potential of One-Line expressions in QGIS] tutorials to learn how to make all the calculations in the attribute table).

SOLUTION

We can use the *Select by Location* tool to select the buildings that are of commercial type and then update the category information for these buildings directly in the attribute table. To do so, open the *Select by Location* tool in the *Vector Selection* tab of the *Processing Toolbox*. Choose features from ***BayStLouis buildings*** layer (if you followed the naming in this tutorial, this will be *All_buildings* layer) that intersect with the ***commercial buildings*** layer (*building_tags* layer from the exercise input) and select *Run*. In the location bar (bottom of the QGIS window), you should see *823 features selected on layer…*. Now open the attribute table for the buildings layer and then the *field calculator* . Create a new field called ***Category*** of ***text*** *type* and ***12*** characters in *length*. Ensure the *only update 823 selected features* option is selected. In the *expression* write:

```
'commercial'
```

This allows marking the 823 selected objects with a *commercial* attribute. Now click on the *Invert selection* tool , which should switch the selection from 823 features to a bit over 13 thousand features. Open the field calculator again and verify that *only 13,255*[2] *features* are selected, and the previous selection is not indicated. Switch to *update existing field* and select the ***Category*** attribute. Here the *expression* will be indicating that if the building is smaller than 300 ft^2 (27.87 m^2), it will be categorized as *nonpermanent*; otherwise, it will be marked as *residential*:

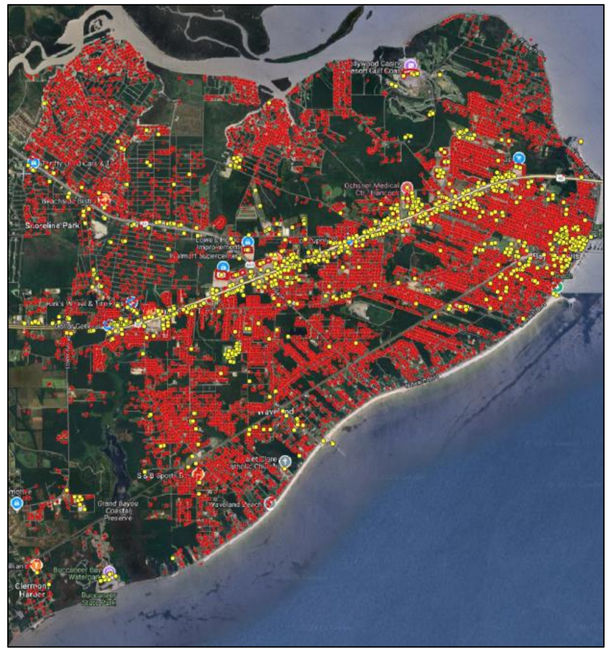

Fig. 6. *Point features over the buildings used for commercial purposes in the studied area.*

```
if ($area < 28, 'nonpermanent', 'residential')
```

Now that we have classified the buildings, we need to add residents' data to it. Since we are only interested in the residential buildings, we can use the *Query Builder* to temporarily filter the data:

```
"Category" = 'residential'
```

and then compute the number of residents' in each house:

```
round($area * 10.76 * 0.00107, 0)
```

In the above formula, we are calculating the area of each building using ***$area***, which produces area in m^2. By multiplying it by ***10.76***, we are reprojecting the unit to ft^2. Finally, the number ***0.00107*** is the average number of residents per square foot of building, which we calculated from census data using averages (2.08 residents / 1950 ft^2). We use the ***round*** function to round the number to an integer (0 indicates no decimal places). This method rounds numbers mathematically, which means that small houses might end up with 0 residents (Fig. 7). Another approach would be to use the ***ceil*** function that rounds to the next integer, meaning we would always have at least one resident. With such an approach, however, we end up with a total of about 30 thousand residents, while *round* gives us about 23 thousand. Census data indicates that Bay St. Louis has 10,511 permanent residents, Waveland has 6,990 residents, and 4,058 residents live in Shoreline Park. This gives the

[2] This number might vary slightly.

total of 21,559 residents in the studied area; therefore, the *round* estimate is a better fit but remember that this is still just an estimate.

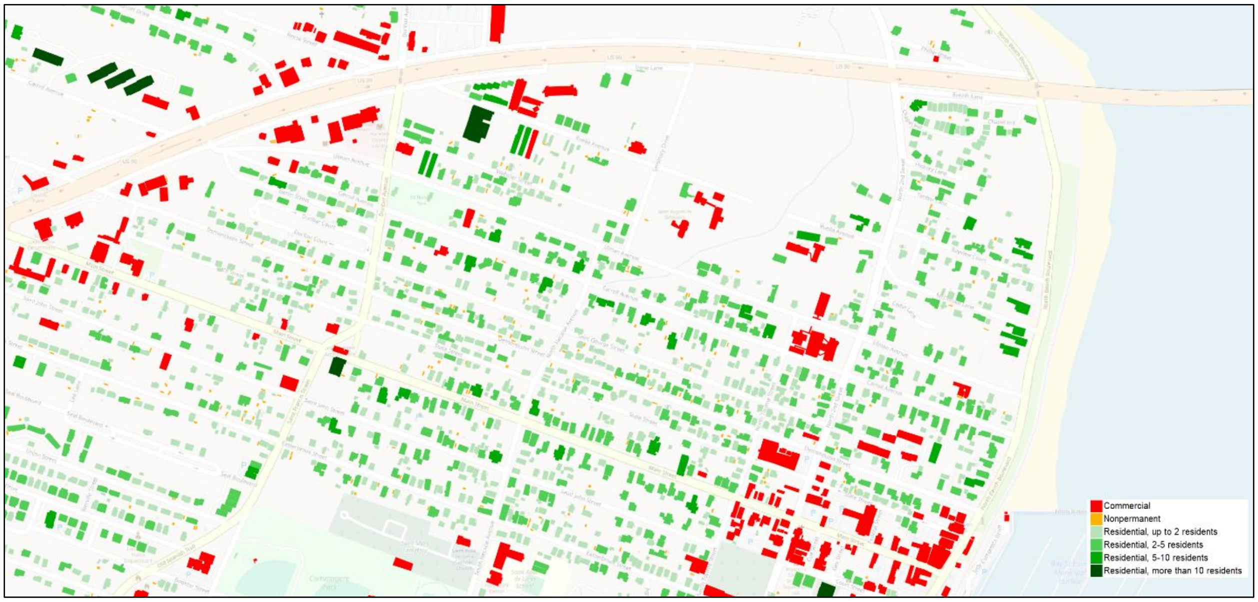

Fig. 7. *Preview of part of the result with buildings classification and estimated number of residents in residential buildings.*

CONCLUSION

This concludes our GEO Tutorial, where you learned about spatial predicates and used them to prepare a residential dataset for further analysis. If you're interested in expanding your knowledge and exploring similar topics, please check out the remaining tutorials in this series.