International Cartographic Association
Association Cartographique Internationale

Leveraging High-Performance Computing to Visualize Big Data on the Web – an Illustrative Case Study

Loeffler, S.

Polar Geospatial Center, University of Minnesota (USA)

Keywords: Big data, web mapping, high performance computing, visualization

As greater collection capabilities, machine learning, and expanding storage options make larger and larger datasets possible, the challenges of visualizing these massive datasets become a central challenge to creating useable and sharable research, as well as unlocking scientific insights. In order to improve data analysis and accessibility, the same high-performance computing (HPC) resources often used to create large datasets can and should be leveraged to produce visualizations of the derived data for exploration, outreach, and quality control purposes.

High resolution satellite imagery, combined with machine learning to identify features, is one of many methods that can create datasets with features numbering in the billions. The data processing pipeline for the method described here was designed around a dataset derived from high-resolution (0.5m) commercial satellite imagery. The dataset includes well over 10 billion polygons, each with associated attributes, far more than can easily be visualized in desktop GIS software simultaneously, let alone streamed over the web. In order to streamline the dataset while maintaining interactivity with individual features, a vector tile approach was deemed appropriate. With the size of the original dataset being over two terabytes and feature counts in the billions, desktop computing power was not practical for creating derived vector tiles. HPC resources were needed to process such large data into formats and structures compatible with current web mapping technologies.

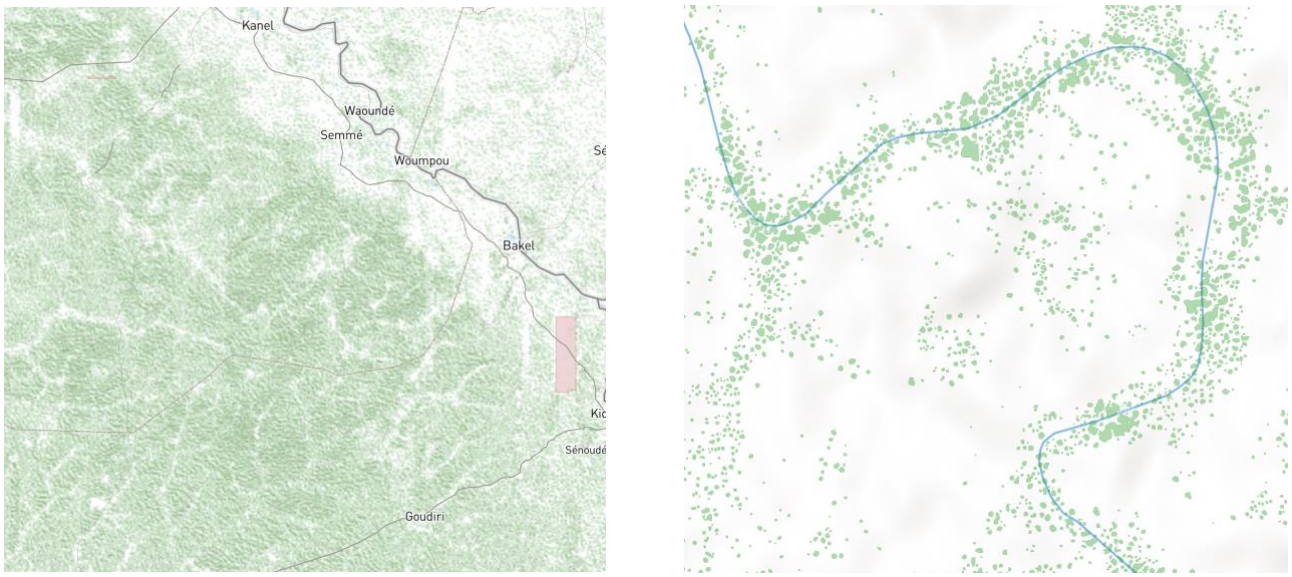

Figure 1. Left: Low zoom level screenshot showing summarized overview points representing overall polygon density. Right: High zoom level screenshot showing individual polygon features.

The open source tool Tippecanoe (github.com/mapbox/tippecanoe) is a powerful library for creating vector tile sets from vector data, with many options for controlling simplification of features and reduction of feature counts at lower zoom levels while maintaining important visual trends. The tool is compatible with Linux and so can be installed on most HPC systems, allowing much larger datasets than would be possible on standard desktop hardware to be processed. All processing described below was completed using the Minnesota Supercomputing Institute's Mangi Cluster at the University of Minnesota.

Tippecanoe can read vector datasets in either GeoJSON or Geobuf format, so the first step in creating this processing pipeline was converting the thousands of individual GeoPackage files to GeoJSON using the open source GDAL library's ogr2ogr command. Next, these GeoJSON data were compressed into geobuf format (github.com/mapbox/geobuf), a compact binary encoding for vector data leading to far more manageable file sizes and memory usage. Since individual polygon vertices are only fully resolvable at very high zoom levels in the final map, centroids were calculated for each

https://doi.org/10.5194/ica-abs-3-184-2021

polygon to create a duplicate dataset to be displayed at lower zoom levels, saving processing power and disk space both in tile generation and during client-side rendering.

In order to maintain reasonably sized vector tiles, every zoom level below the highest level necessary to display each individual feature must contain only a subset of the original features. Tippecanoe provides many options to configure this reduction in data, and after many comparative experiments, it was found that a feature reduction strategy that drops a fraction of features at each lower zoom level, as opposed to other options that drop features preferentially from high density regions, was optimal. This strategy allowed the overall trends of the data to be maintained, with areas of higher density showing up more prominently at low zoom levels.

Final tile sets are formatted as .mbtiles files (an open format for storying vector tiles based on SQLite databases) and served using an open source mbtiles server library (github.com/consbio/mbtileserver) hosted on University of Minnesota servers. These tiles are consumed and styled by a Mapbox GL-JS map running in a React.js web application.

Interactivity with the features in the final web map was an important goal of this visualization. Since all features are maintained at the highest zoom level of the polygon vector tile set, interacting with individual features was easily implemented in the web map. However, displaying summary statistics at lower zoom levels required summing feature attributes (e.g. polygon area) into the features not dropped during the zoom-based simplification routine of Tippecanoe. To solve this, at each subsequently lower zoom level, applicable feature attributes are summed into remaining nearby features, allowing users to query areas on the map for summarized statistics of the selected region's data, without having to query and sum millions to billions of individual polygons in the browser.

Supplemental data, such as the coverage footprints of imagery used for feature identification are added to the map as another layer and highlighted on selection. Links to previews of the imagery used for identification of features let users interrogate the machine learning model's results against the source imagery. Additionally, areas of no data are indicated to avoid confusion between sparse results and areas not covered by the dataset.

Cartographically, the map emphasizes the data by using muted base maps (with the exception of a satellite option) and a simplistic interface of map popups and screen overlays, combined with a mixture of hover and click interaction. A subtle hillshade is included to allow the user to connect terrain morphology to dataset patterns. Overview points at low zoom levels are blurred slightly to visually account for the features dropped during vector tile creation. Since the overall density of the dataset is maintained through the tile creation process, this results in a more accurate overview of the dataset's coverage than the individual remaining points themselves could achieve at low zoom levels.

Practically, the map has been used by the original dataset team to discover bugs, such as overlapping and double counting of data in certain locations where imagery datasets overlapped, as well as for general exploration and hypothesis generation, as there is no other way to quickly explore a dataset of this size dynamically. Once published, the map will accompany the data and serve as a first point of exploration for users interested in using the results of the study, from on-the-ground decision makers to the general public.

Expanding the use of HPC systems to include not only dataset creation and processing, but also routines explicitly designed to get large datasets into easily explorable web-based products has enormous applications for advancing and communicating the science made possible by these systems.

Abstracts of the International Cartographic Association, 3, 2021.
30th International Cartographic Conference (ICC 2021), 14–18 December 2021, Florence, Italy.
https://doi.org/10.5194/ica-abs-3-184-2021 | © Author(s) 2021. CC BY 4.0 License.