Simon Fraser University

Analyzing BC's Hiking Trails

Group Project

Yang Liu, Ronney Lok, Sina MohammadiNiyaki

CMPT 353

Greg Baker

December 4th, 2023

Introduction

The goal of the OpenStreetMap (OSM) project is to produce an editable, comprehensive map of the world. That said, contributions to OSM are welcome from anyone. Contributors upload and modify map data using their resources, and satellite images. The end product is an extensive worldwide map that is used for a range of purposes.

As an open source map, OSM has its issues. Because of the different methods and tools used by participants, the quality of the data varies. Some locations may be precisely documented, while other data is not as accurate. These inconsistencies are noticeable when OSM data is applied for specific tasks such as route planning or geographic feature analysis.

The purpose of this project is to apply smoothing techniques to OSM data in order to eliminate anomalies and improve overall precision. This project will describe the problem of data noise in OSM, the smoothing approaches chosen to solve the problem, and the benefits of smoothing algorithms. Using the results of the observation, it will be able to help us determine what situations benefit most from specific smoothing algorithms.

This project is inspired from the fact that over thousands of people get lost per year while hiking in the wilderness according to an article by Hiker's University, a website dedicated to educating people on how to hike prepared. While many of the reasons are due to inclement weather and dangerous wildlife, a large percentage are from navigation issues. Our analysis seeks to lessen the impact of the latter.

Data Noise in OpenStreetMap

Our project confronts the challenge of inconsistent data quality within the OpenStreetMap (OSM) landscape. Our research on OSM's database has shown that the amount of information offered varies by location. While many regions have features that have been thoroughly recorded, others have inaccurate and insufficient data. This variation comes from the techniques of gathering and submitting volunteers information; they contribute to the database, there is a lot of data that needs to be adjusted to reach the quality of reliability required for complex uses like precise navigation and geographic analysis. OSM data's varied quality could be troublesome for the systems and services that depend on the data from the map. In order to trust the data from OSM, there needs to be some improvement on the data.

Data Used

The data that we used in our project was obtained from OpenStreetMap. We used two different datasets, the first one is a GPX trail of the Garibaldi hike. This trail was chosen because of the location's close proximity to us and many residents of Vancouver are familiar with this place and have hiked here before.
[https://www.openstreetmap.org/user/Alan%20Trick/traces/2509002]

The second dataset that we chose is the Upper Falls trail in Golden Ears Park. This is another well known hiking trail from the Greater Vancouver region. [https://www.openstreetmap.org/user/Alan%20Trick/traces/2945730]

The data is reliable as it is continuously improved upon by community members of the OSM site. In addition, we ensured the trail was popular enough such that there are enough members that would care enough to maintain any changes and errors in the GPX data.

These two GPX files that we chose to use were smoothed already so to simulate the roughness that it would originally come with when mapped, we have created a Python script that takes in the GPX file and “roughens" it by deleting every other point in the file. This allows a fairer comparison among the different data sets in the next step.

Planning the Refinement Through Smoothing Techniques

Oftentimes, sensor noise can be an issue in GPS systems. In fact GPS’ are only accurate up to about 5m. To combat this, smoothing techniques can be applied to improve the accuracy.

We are looking into a number of smoothing techniques in an attempt to enhance OSM data quality. We anticipate the benefits and drawbacks of each method taking into account the unique challenges faced by GPX data.

1. Kalman Smoothing
2. Loess Smoothing
3. SMA
4. Gaussian Smoothing
5. Savitzky-Golay

The five techniques mentioned above are common smoothing techniques applied on GPX data. We will be employing these algorithms to smoothen out our GPX data. At the end, we will compare the results of the smoothing algorithms to find out which works best for our case: hiking trails.

For our report, we will be using five different smoothing techniques against two GPX files to ensure comprehensive coverage.

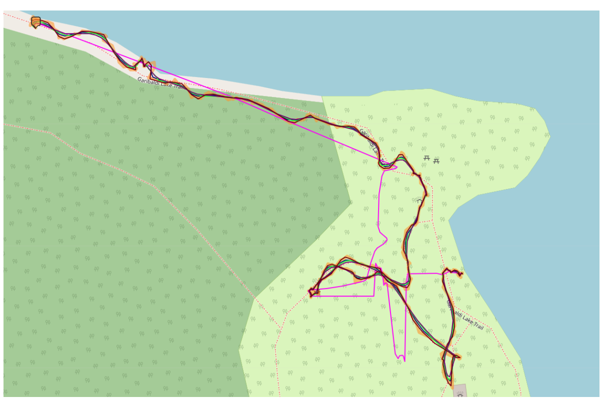
Observations

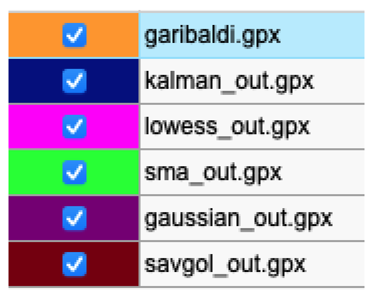
Garibaldi Trail

Haversine distance: 530.70 m

Algorithms	Calculated Distance	Difference
Kalman	441.72 m	88.98 m
Loess	376.27 m	154.43 m
SMA	434.91 m	95.79 m
Gaussian	414.55 m	116.15 m
Savitzky-Golay	487.54 m	43.16 m

<u>GPX Result</u>





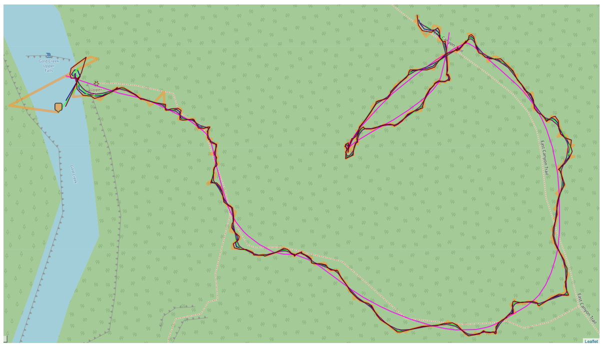
Upper Falls Trail

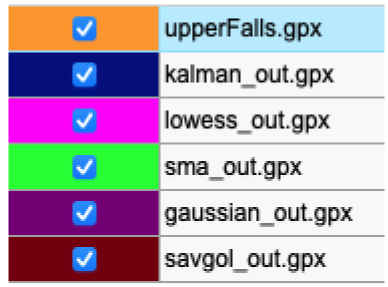
Haversine distance: 916.83 m

Algorithms	Calculated Distance	Difference
Kalman	728.46 m	188.37 m
Loess	591.53 m	325.30 m
SMA	718.49 m	198.34 m

Gaussian	672.16 m	244.67 m
Savitzky-Golay	821.95 m	94.88 m

<u>GPX Result</u>





Discussion and Result

The analysis of the Garibaldi and Upper Falls data files employing the five smoothing algorithms reveals significant insights into the application and efficacy of these methods for hiking trail data refinement in OSM.

Effectiveness of Smoothing Algorithms

- **Kalman Filter:** This algorithm showed a moderate reduction in the noise of the GPX data. While it provided a general distance reduction. The amount of reduction seems to vary across different trails, which may indicate its sensitivity to the initial state and noise parameters. Potentially, in order to get a better result with Kalman Filter, fine-tuning parameters depending on the specific data is helpful.
- **LOESS (Locally Estimated Scatterplot Smoothing):** It offered the most drastic reduction in distance, which suggests it may be too aggressive for trails with lots of natural turns and may not preserve the trail's true shape. LOESS is best suited for data that is less noisy or for applications where a smoother overall trend is more important than detailed accuracy.
- **SMA (Simple Moving Average):** Its results were similar to the Kalman filter, indicating a consistent smoothing effect without overfitting the data. SMA's consistent results suggest it might be a reliable choice for trails with a balance of straight sections and turns. However, it might not be the best for trails with frequent sharp turns or highly irregular paths due to its inherent lagging nature.
- **Gaussian Smoothing:** This method balanced detail retention with noise reduction well, suggesting it might be a good fit for trails with varied terrain.

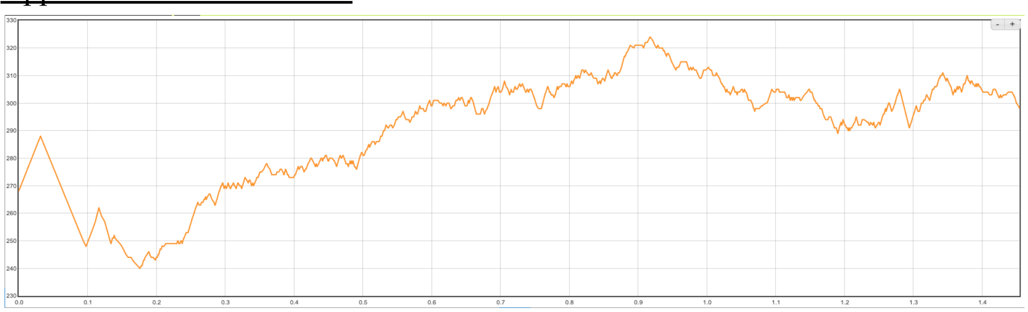
- **Savitzky-Golay:** The least reduction in distance might indicate this method preserves more detail, making it suitable for rugged trails where navigation precision is crucial. Savitzky-Golay filter's lesser reduction in distance does not necessarily mean it is preserving more noise. It could be maintaining the essential features of the trail's geometry, which is crucial for accurate representation and navigation..

Influence of Elevation on Smoothing Algorithms

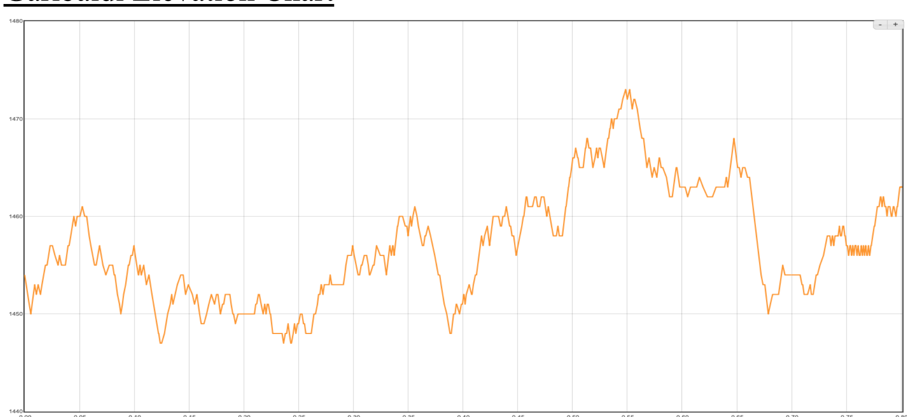
Elevation data is crucial in trail mapping, as it affects a trail's difficulty and safety. Algorithm performance varies with elevation changes, as demonstrated by the Garibaldi and Upper Falls data.

When mapping trails, the choice of a smoothing algorithm matters a lot for showing elevation. The Kalman Filter can smooth out hills and valleys too much, which might hide steep parts of the trail. LOESS might make a trail seem flatter than it is by over-smoothing the elevation data. SMA tends to average out the heights and could miss small but important changes in the trail's slope. On the other hand, Gaussian Smoothing is more flexible and can keep important parts of the trail's shape. The Savitzky-Golay filter is good at keeping the trail's actual ups and downs, which is important for accurate maps. It's important to choose the right algorithm that best matches the real ups and downs of a trail.

UpperFalls Elevation Chart



Garibaldi Elevation Chart



Analyzing the Best Fit

- The LOWESS algorithm's tendency to produce the shortest distances may not always align with the actual trail paths, potentially leading to data that is too smooth for accurate navigation.
- The Savitzky-Golay filter, while minimally altering the trail distance, seems to maintain a high level of detail, which could be more beneficial for rugged, feature-rich trails.
- The Gaussian and SMA filters offer a compromise, smoothing noise while possibly retaining essential trail characteristics. A combination of these methods could potentially yield the best results, with Gaussian smoothing adjusting for trail variability and SMA providing trend smoothing.

Limitations and considerations for the future

Problems encountered

- Finding a reliable website for GPX data
  - Although there are many GPX trails online, it was difficult finding data that was both open source and reliable. Many were also too smooth, which didn't make sense for our use case.
- Insufficient data and data quality issues
  - Some data was incomplete or had sections missing. It was important for us to find data that was complete and also reliable which was not easy to do considering anyone can just upload data to OSM.
- Overfitting and underfitting
  - Since we compared five different algorithms, it was important to understand each algorithm's parameters in order to not underfit or overfit our data.

Future plans

- Selecting different algorithms
  - We can explore other options as there are other interesting algorithms that exist for smoothening GPX data such as the Butterworth Filter, Bezier Curve, and the Douglas-Peucker Algorithm.
- Trying a combination of algorithms for specific sections
  - Use a piecewise approach to employ specific algorithms that work best in the specific sections of the trail to take advantage of each algorithms strength
- Cross validate our smoothing algorithms with other sources
  - Compare with other sources online that have smoothed out trails such as AllTrails.

In retrospect

- Experiment more with the features of each algorithm and fine tuning them
  - We should have spent more time understanding and mastering the depths of each algorithm's parameters to ensure we are getting the best output that we can.
- Creating our own GPX file from hiking a trail that we all enjoy
  - This would make the project more interesting and realistic. We could have also covered trails that are not available on OSM to add to the community.

Conclusion

In conclusion, this data science project helped us determine which of the smoothing algorithms worked best for hiking trails. Although it was clear, we should avoid the LOWESS algorithm due its tendency to over smooth, it was more situation dependent for the other algorithms. Our results found that the remaining algorithms would work best if they were used in combination. For example, we should employ Savitsky-Golay when specific sections are feature rich as it is more susceptible to noise. Overall, this project helped us realize that it was not about determining the best smoothing algorithm for hiking trails, but rather understand how different algorithms can work together in creating the best smoothed out trail.

Accomplishment Statement - Yang

Created a function to “roughen” the GPX file by deleting every other point in the file and added logic to output the resulting files in their respective locations.

Organized the file structure of the codebase, cleaned the repository, created the readme instructions and refactored the code.

Authored and proofread the first half of the report. This includes the Introduction, Data Noise in OpenStreetMap, and Data Used.

Accomplishment Statement - Ronney

Applied smoothing algorithms in Python to Hiking Trail GPX data to lessen the amount of noise in the output data.

Authored an analysis research report involving GPX data and compiling it to research needs.

Cleaned GPX data using Pandas to filter out unneeded data that would cause additional overhead and slow down speeds.

Accomplishment Statement - Sina

Implemented multiple smoothing algorithms in Python for comparison purposes.

Conducted analysis to assess the performance of different smoothing algorithms on hiking trail data.

Aimed to identify the optimal smoothing method for hiking trails, accounting for the unique challenges presented by hiking trail files.