

Analyzing Google Smartphone Decimeter Challenge 2023-2024 Data on a Pixel 6 Pro: A Comparison with Rtklibexplorer

Shilin Shen¹, Nobuaki Kubo²

¹Graduate School of Marine Science and Technology, Tokyo University of Marine Science and Technology, Tokyo, Japan

²Faculty of Marine Technology, Tokyo University of Marine Science and Technology, Tokyo, Japan

Abstract

This paper presents an analysis of GNSS data from the Google Smartphone Decimeter Challenge 2023-2024 using a Pixel 6 Pro smartphone. The study compares the results obtained using rtklibexplorer and a proprietary program developed in our laboratory focusing on real-time forward analysis applicable to practical scenarios. Initially, a program script was constructed following detailed instructions from Tim Everett, the developer of rtklibexplorer, achieving a public leaderboard score of 1.807. Through further optimizations, including adjustments to GNSS parameters such as the signal-to-noise ratio (SNR) mask and elevation ratio, the score was improved to 1.545, securing a highest ranking of 13th place. Building on insights from the challenge, this study further refines our laboratory's software for real-world GNSS applications. By processing data from the Pixel 6 Pro using its L1 frequency and forward techniques, we compared our results with those obtained using rtklibexplorer. Our program demonstrated comparable performance, with horizontal 2DRMS values ranging from 3m to 5m, and in some instances, even surpassing the accuracy of rtklibexplorer. These findings highlight the significant research value and potential of our laboratory's program for enhancing GNSS positioning accuracy in practical applications.

Keywords

GNSS, Google Smartphone Decimeter Challenge, rtklibexplorer, positioning accuracy, real-time analysis, forward processing, GNSS optimization

1. Introduction

This paper presents the results of an analysis using GNSS (Global Navigation Satellite System) data from the Google Smartphone Decimeter Challenge 2023-2024 (GSDC)[1]. The challenge has been completed, and we utilized the GNSS data provided as training data. The software used in this analysis includes rtklibexplorer[2] and a program developed in our laboratory. Rtklibexplorer, a popular GNSS analysis tool available on GitHub, is a modified version of RTKLIB[3] optimized for moving platforms and challenging environments. Similarly, our lab's software has been designed to address these challenging conditions, particularly through the use of loosely-coupled GNSS and velocity integration.

The rtklibexplorer software was employed by the author to participate in the challenge, achieving a score of approximately 2 meters and a ranking of 50th without specific optimizations. Despite the performance gap between the top scores and that of rtklibexplorer, it serves as a valuable benchmark. This paper aims to demonstrate the performance of forward analysis and explore strategies for improving positioning accuracy using the software developed in our laboratory.

2. Participation and Performance in the Google Smartphone Decimeter Challenge

The GSDC 2023-2024 focused on enhancing high-precision GNSS positioning using data collected from smartphones[4]. Hosted on Kaggle, the competition encouraged participants to develop solutions

Proceedings of the Work-in-Progress Papers at the 14th International Conference on Indoor Positioning and Indoor Navigation (IPIN-WiP 2024)

✉ m235029@edu.kaiyodai.ac.jp (S. Shen); nkubo@kaiyodai.ac.jp (N. Kubo)



© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

achieving decimeter-level accuracy.

2.1. Scoring and Leaderboards

Submissions were evaluated based on the mean of the 50th and 95th percentile distance errors across various smartphone models. The competition included two phases: a public leaderboard for interim rankings and a private leaderboard for final rankings, with prizes and recognition based on the private leaderboard results[5].

2.2. Data Collection

GSDC utilized smartphones with advanced GNSS capabilities to collect raw GNSS measurements and high-precision ground truth data across diverse environments in the United States, such as open areas, urban canyons, and highways. The smartphones used included:

- Google Pixel Series (e.g. Pixel 6 Pro)
- Xiaomi Mi 8
- Samsung Models (e.g. Samsung S22 Ultra)

The dataset was divided into training and test sets, with the training set used for model development and the test set for final submissions and leaderboard ranking.

3. Score Using Rtklibexplorer

During our participation in GSDC, we followed the guidance and programs provided by rtklibexplorer to obtain initial positioning results. Building on this foundation, we made further optimizations and improvements, ultimately enhancing the public leaderboard score from 1.807 to 1.545 for the test dataset. Although the improvement in the public score was modest, the progress in the private score underscores the feasibility and effectiveness of our optimizations. Below is a detailed account of our participation process:

3.1. Initial Setup

Based on rtklibexplorer's "Getting Started with RTKLIB: 2023" article for the Google Smartphone Decimeter Challenge[6], Tim Everett, the developer of rtklibexplorer, provides a detailed competition notebook outlining four key steps to enhance positioning accuracy:

1. **Pre-processing Data:** Preprocessing to convert raw data into readable rover data, resulting in `gnss_log.obs` and `gnss_log.trc` files from the initial `gnss_log.txt` file.



 gnss_log.obs	2024/6/22 5:40
 gnss_log.trc	2024/6/22 5:40
 gnss_log.txt	2023/9/12 20:13

Figure 1: Converted Files

2. **Initial Processing with RTKLIB:** Retrieving base observation and satellite navigation files. For the 2023 competition, the primary base observations are SLAC and VDCY, obtainable using the `get_base_data.py` script in the Python folder of the notebook.
3. **Post-processing and Smoothing:** Applying Kalman filtering and other smoothing techniques to refine initial estimates and improve accuracy.
4. **Automating with Python:** Using Python scripts to automate the entire processing workflow, ensuring efficient and rapid data handling.

In the 2023-2024 Challenge, the test dataset comprised 40 data sets. Using Tim Everett's program, an initial public score of 1.807 was achieved.



Figure 2: Score in Public Leaderboard on Initial Setup

3.2. Enhancing Accuracy

At this stage, our score ranked around 40th place. To achieve a higher ranking, we further optimized the GNSS parameters in the configuration file to better align with the positioning accuracy requirements of the competition data. Since the data lacked L2 frequency information, we focused on the L1 and L5 frequencies. The key adjustments made were as follows:

1. **Signal-to-Noise Ratio (SNR) Mask:** We adjusted the SNR mask from

```
snrmask_L1=28, 28, 28, 28, 28, 28, 28, 28, 28,  
snrmask_L5=20, 20, 20, 20, 20, 20, 20, 20, 20
```

to

```
snrmask_L1=28, 28, 29, 29, 30, 30, 31, 31, 32,  
snrmask_L5=20, 20, 21, 21, 22, 22, 23, 23, 24.
```

The SNR mask filters out satellite signals that do not meet the minimum SNR requirements, ensuring that only signals with sufficient quality are used, thereby improving positioning accuracy.

2. **Elevation Ratio (Eratio):** We modified the elevation ratio (Eratio) from

```
stats-eratio1 = 200
```

to

```
stats-eratio1 = 100.
```

The Eratio filters out satellite signals that are too close to the horizon, which are more prone to errors due to atmospheric interference and multipath effects. This adjustment further enhanced the accuracy of our GNSS solutions.

Through these parameter adjustments, we achieved a best public score of 1.545, with a highest ranking of 13th place.

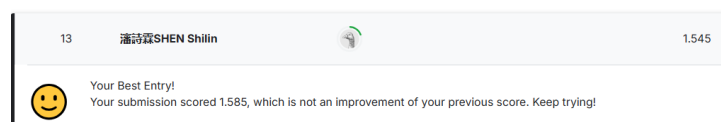


Figure 3: Best Score and Highest Ranking

4. Results and Comparison

4.1. New Attempts After the Challenge

In real-world applications, smartphone positioning is typically performed in real-time. Although the GSDC permits a combination of pre-processing and post-processing (integrating forward and backward processing) to achieve highly accurate positioning, practical scenarios generally rely on forward processing for real-time measurements. Based on this, the present study aims to optimize smartphone positioning algorithms using GSDC data under real-world conditions.

Following participation in the 2023-2024 GSDC, this research leverages GSDC datasets to refine our lab's existing algorithms, with the objective of benchmarking our results against those from the rtklibexplorer project. Our laboratory's methodology involves using pseudorange and Doppler residuals to exclude satellites with significant errors, followed by Kalman filtering for state estimation, which includes the smoothing of both position and velocity measurements. By systematically adjusting the filtering thresholds, we aim to achieve optimal performance. This approach is expected to yield critical insights into the efficacy of our methodologies and contribute valuable knowledge to the GNSS research community.

The optimization process for improving positioning accuracy is systematically divided into four critical steps, each addressing specific aspects of positioning errors to refine the overall solution. These steps are detailed below.

4.1.1. Differential GNSS

The first step involves obtaining an initial solution through Differential GNSS (DGNSS). Our proprietary program provides a foundational positioning estimate. Figures 4, 5, and 6 illustrate the initial pseudorange residuals, Doppler residuals, and horizontal error results, respectively, observed during this stage. These residuals are computed using the least squares method, where velocity vectors are derived by minimizing the sum of squared pseudorange residuals. This step establishes the baseline performance of the system.

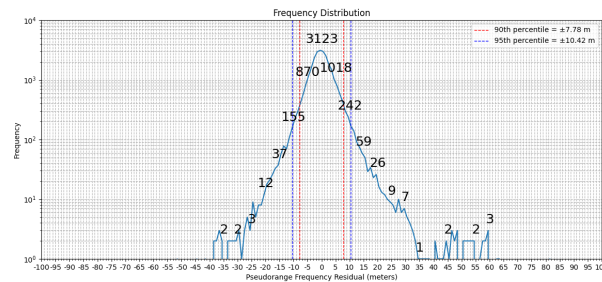


Figure 4: Original Pseudorange Residuals

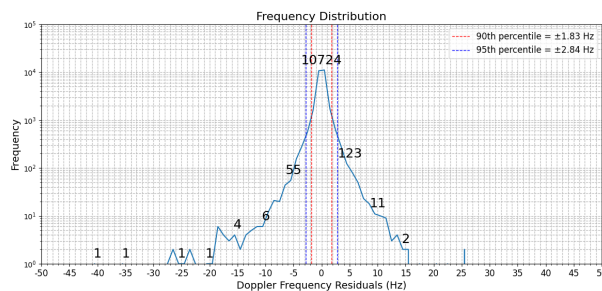


Figure 5: Original Doppler Residuals

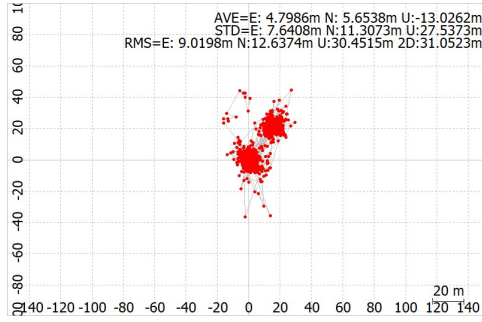


Figure 6: Initial Horizontal Error Result

4.1.2. Mitigation of Suspect Satellites Using Pseudorange Residuals

In this part, we focus on enhancing positioning accuracy by identifying and eliminating suspect satellites based on pseudorange residuals. GNSS signals are susceptible to interference and multipath effects, often indicated by significant pseudorange residuals, which can introduce systematic errors into the positioning solution.

To mitigate these errors, we first increase the precision of pseudorange residual calculations. The pseudorange residual is computed using the following mathematical expression:

$$r_i = \sqrt{(SVx_i - x_0)^2 + (SVy_i - y_0)^2 + (SVz_i - z_0)^2} \quad (1)$$

where SVx_i , SVy_i , SVz_i are the satellite coordinates for satellite i , while x_0 , y_0 , z_0 are the initial estimated receiver coordinates.

The pseudorange residual R_i is then computed as:

$$R_i = P_i + \Delta t + C - r_i \quad (2)$$

where P_i is the measured pseudorange, Δt is the satellite clock correction, C is any additional correction factor.

By utilizing this method, we can identify and exclude satellites with large pseudorange residuals. Figure 7 shows the corrected pseudorange residuals after removing the suspect satellites.

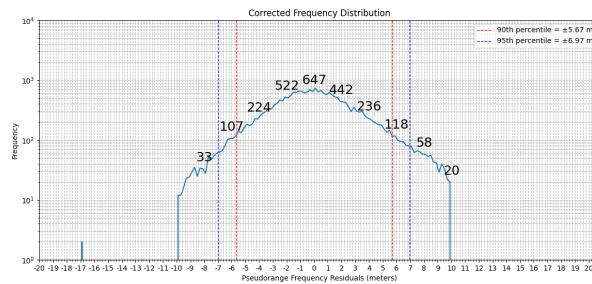


Figure 7: Corrected Pseudorange Residuals

4.1.3. Mitigation of Suspect Satellites Using Doppler Residuals

This part addresses the enhancement of positioning accuracy by focusing on Doppler residuals. Significant Doppler residuals can indicate errors in the GNSS signals, which may degrade the positioning solution.

The Doppler residual for each satellite is calculated by comparing the measured Doppler shift with the calculated Doppler shift. The calculated Doppler shift ($\text{Doppler}_{\text{calculated}}$) is determined using the following formula:

$$\text{Doppler}_{\text{calculated}} = \frac{f_c}{c} \times (\mathbf{v}_{\text{sat}} - \mathbf{v}_{\text{recv}}) \cdot \hat{\mathbf{r}} \quad (3)$$

where f_c is the carrier frequency of the signal, c is the speed of light, \mathbf{v}_{sat} is the velocity vector of the satellite, \mathbf{v}_{recv} is the velocity vector of the receiver, $\hat{\mathbf{r}}$ is the unit vector pointing from the receiver to the satellite.

The Doppler residual is then calculated as:

$$\text{Doppler}_{\text{residual}} = \text{Doppler}_{\text{measured}} - \text{Doppler}_{\text{calculated}} \quad (4)$$

where $\text{Doppler}_{\text{measured}}$ is the observed Doppler shift from the satellite.

This process allows us to identify and exclude satellites with large Doppler residuals, thus improving the accuracy of the positioning solution. The corrected Doppler residuals are shown in Figure 8.

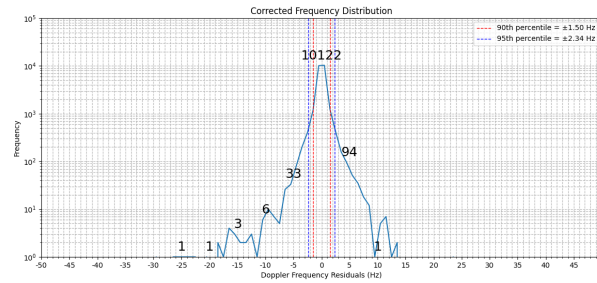


Figure 8: Corrected Doppler Residuals

Additionally, after enhancing the positioning accuracy through the mitigation of suspect satellites using pseudorange residuals and Doppler residuals, the updated horizontal error results are illustrated in Figure 9. This demonstrates the improvement in positioning accuracy after applying the pseudorange residuals and Doppler residuals corrections.

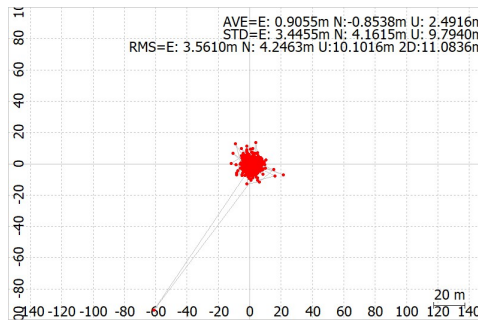


Figure 9: Horizontal Error Result after Mitigating Suspect Satellites Using pseudorange residuals and Doppler residuals

4.1.4. Incorporation of Loosely coupled Kalman Filter

The third step involves applying a Loosely coupled Kalman Filter to the positioning data. This technique is crucial for smoothing and refining estimated positions by filtering out noise and correcting errors that may have accumulated during data collection. The Kalman Filter is particularly effective in dynamic environments, offering a method to continuously improve the accuracy and stability of the positioning solution[7].

The final horizontal error result after implementing the aforementioned three steps was 5.45 meters, indicating a significant improvement in positioning accuracy.

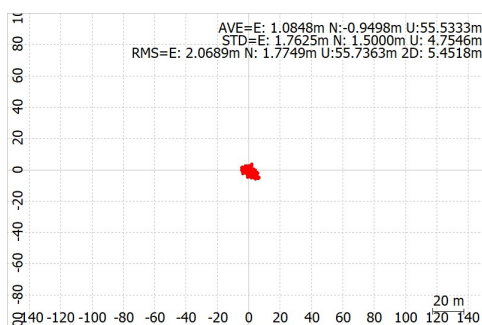


Figure 10: Horizontal Error Result after the Three Steps

4.1.5. Optimization Through Threshold Selection

The final step is selecting appropriate thresholds, essential for balancing sensitivity and specificity in detecting and correcting errors. Systematically adjusting and comparing different threshold values allows us to identify the settings that yield the best performance in terms of accuracy and reliability. This ensures that the refined positioning solution is both accurate and robust across various operational conditions.

In the subsequent analysis, we used data obtained from the Pixel 6 Pro model in the training dataset of the Google Smartphone Decimeter Challenge. The dataset contains 14 routes with Pixel 6 Pro measurement data. We processed this data using only L1 frequency and forward techniques, comparing our laboratory program results with those obtained using Tim Everett’s rtklibexplorer program.

First, we obtained processing results from both rtklibexplorer and our laboratory’s program. Since the training dataset includes ground truth data (span_log.nmea), we were able to compare the Google Score values of the output position files, as shown in Table 1.

Table 1
Google Score Comparison of Results

File	Our Score	Rtklibexplorer Score
2021-12-07-19-22-us-ca-lax-d	2.51	2.57
2021-12-07-22-21-us-ca-lax-g	2.82	2.07
2021-12-08-17-22-us-ca-lax-a	2.62	1.04
2021-12-08-18-52-us-ca-lax-b	3.80	2.74
2021-12-08-20-28-us-ca-lax-c	2.36	1.69
2021-12-09-17-06-us-ca-lax-e	3.26	1.83
2022-01-11-18-48-us-ca-mtv-n	6.50	3.46
2022-04-01-18-22-us-ca-lax-t	6.23	24.78
2022-05-13-20-57-us-ca-mtv-pe1	2.44	1.25
2023-03-08-21-34-us-ca-mtv-u	2.97	1.93
2023-09-06-00-01-us-ca-routen	3.76	3.34
2023-09-06-18-47-us-ca	6.62	2.62
2023-09-07-19-33-us-ca	2.48	2.45
2023-09-07-22-47-us-ca-routebc2	2.78	2.14
Final Score	3.65	3.85

In addition to the Google Score comparison, we also compared the total number of epochs processed by both programs. Table 2 below presents the original number of epochs and the total positioning results from our laboratory’s program versus rtklibexplorer.

Table 2
Google Score Comparison of Results

File	Original Number of Epochs	Our Total Positioning Results	Rtklibexplorer Total Positioning Results
2021-12-07-19-22-us-ca-lax-d	1760	1748	1723
2021-12-07-22-21-us-ca-lax-g	1433	1428	1432
2021-12-08-17-22-us-ca-lax-a	1441	1430	1432
2021-12-08-18-52-us-ca-lax-b	1417	1414	1379
2021-12-08-20-28-us-ca-lax-c	922	911	918
2021-12-09-17-06-us-ca-lax-e	1100	1092	1090
2022-01-11-18-48-us-ca-mtv-n	943	938	934
2022-04-01-18-22-us-ca-lax-t	1465	1421	242
2022-05-13-20-57-us-ca-mtv-pe1	2161	2174	2147
2023-03-08-21-34-us-ca-mtv-u	1101	1096	1078
2023-09-06-00-01-us-ca-routen	1649	1644	1506
2023-09-06-18-47-us-ca	1485	1429	1393
2023-09-07-19-33-us-ca	1088	1077	1060
2023-09-07-22-47-us-ca-routebc2	2066	2061	2026

The table shows that the results from our laboratory’s program are generally comparable to those obtained using rtklibexplorer, demonstrating the research value and potential of our approach. In some cases, our program even outperforms rtklibexplorer.

5. Summary, Conclusion, and Future Works

5.1. Summary

This study utilized GNSS data from the Google Smartphone Decimeter Challenge 2023-2024 to evaluate and enhance smartphone positioning accuracy. The primary tools employed were rtklibexplorer and a proprietary program developed in our laboratory. The focus was on real-time forward analysis, which is more relevant for everyday smartphone use. Optimizations were made by adjusting GNSS parameters in the configuration files, such as the signal-to-noise ratio (SNR) mask and elevation ratio (Eratio), to improve performance.

5.2. Conclusion

The comparison between rtklibexplorer and our laboratory’s program demonstrated that our program could achieve comparable accuracy levels. Through parameter adjustments, the initial public leaderboard score was improved from 1.807 to 1.545, with the highest ranking reaching 13th place. This improvement indicates the feasibility and effectiveness of the optimizations implemented. Post-competition comparisons revealed that, in certain scenarios and environments, our program outperformed rtklibexplorer, suggesting that our lab’s program has significant research value and potential for real-world GNSS applications, on par with rtklibexplorer.

5.3. Future Works

Future work will focus on several key areas:

- **Incorporating Carrier Phase Measurements:** As carrier phase measurements were not utilized in this analysis, future efforts will focus on integrating these measurements into our program to enhance accuracy.
- **Incorporating L5 Measurements:** Given that L5 measurements are generally more accurate than L1, incorporating L5 data into our program will be a priority.

- Improving Pseudorange Residual Detection: We have used a standard pseudorange residual check in this study. Future work will involve developing more sophisticated algorithms to detect large pseudorange residuals more accurately.
- Achieving 2.0 m Horizontal Errors in 2DRMS: Our ultimate goal is to achieve horizontal errors of 2.0 meters in 2DRMS. This will require the development of new methods to reach this target.

Acknowledgments

We would like to thank the organizers of the Google Smartphone Decimeter Challenge for providing the dataset used in this study. This research was supported by the Tokyo University of Marine Science and Technology.

References

- [1] Google smartphone decimeter challenge 2023-2024, <https://www.kaggle.com/competitions/smartphone-decimeter-2023>, 2023. Accessed: 2024-08-27.
- [2] RTKLIB Explorer, Rtklib explorer, <https://github.com/rtklibexplorer/RTKLIB>, 2024. Accessed: 2024-08-27.
- [3] Rtklib, <https://github.com/tomojitakasu/RTKLIB>, 2024. Accessed: 2024-08-27.
- [4] C. Loh, Google decimeter challenge explanation, <https://www.chadloh.com/google-decimeter-challenge>, 2022. Accessed: 2024-08-27.
- [5] Smartphone decimeter competition, <https://www.ion.org/gnss/smartphone-decimeter-challenge.cfm>, 2024. Accessed: 2024-08-27.
- [6] T. Everett, Getting started with rtklib: 2023, <https://www.kaggle.com/code/timeverett/getting-started-with-rtklib-2023>, 2023. Accessed: 2024-08-27.
- [7] K. N. E.-M. A. Fredeluces E, Ozeki T, Modified rtk-gnss for challenging environments, *Sensors* 24 (2024) 2712. doi:10.3390/s24092712.