Enhancing Smartphone GNSS Accuracy with Kalman Filter Integration

Xiang Gao
University of California San Diego
9500 Gilman Drive, La Jolla, CA 92093-0021
x9gao@ucsd.edu

Abstract

Urban signal obstructions and technical challenges like interference and environmental factors often lead to inaccuracies in smartphone GNSS positioning. Current accuracy of 3-5 meters [1] falls short for lane-specific navigation, resulting in missed exits and inaccurate arrival times. This project aims to enhance smartphone GNSS accuracy to decimeter or 1-meter levels using Kalman Filter (KF) [2] and Kalman Smoothing (KS) [3]. We compare these methods with a Weighted Linear Regression approach [4]. Results show significant improvements in positioning accuracy, with KS outperforming KF, and KF surpassing Weighted Linear Regression, making them ideal for precise positioning in complex urban environments.

1. Introduction

Global Navigation Satellite Systems (GNSS) [5] have evolved into a multi-system, globally integrated framework, becoming essential for modern navigation and positioning. GNSS applications range from everyday use in in-car navigation to specialized fields like urban planning, precision agriculture, disaster relief, and autonomous driving. However, high-precision positioning faces challenges such as climate changes, multipath effects, complex environments, and signal interference. This paper focuses on enhancing smartphone GNSS positioning, which currently achieves only 3-5 meters accuracy. While useful, this can cause "position jumps," resulting in unstable and unreliable experiences in many applications. [1]

To enhance GNSS positioning accuracy, we employ robust estimation methods, focusing on the Kalman Filter and its variants. Our aim is to achieve smartphone GNSS accuracy at the decimeter or even centimeter level, thereby improving geospatial information granularity for advanced navigation methods.Initially, we use Weighted Least Squares (WLS) for preliminary processing. WLS assigns different weights to observations based on their reliability, reducing the impact of noise and generating a reliable covariance matrix.

After obtaining preliminary results from WLS, we further optimize these estimates using the Kalman Filter. This recursive algorithm combines WLS-processed GNSS measurements with a dynamic system model to estimate the receiver's most likely state at each time step, effectively correcting errors and significantly improving position accuracy. This method is crucial for real-time navigation as it allows for continuous updates, adapting to environmental changes and measurement conditions, thereby enhancing positioning reliability.

To further enhance accuracy, we apply Kalman Smoothing, an extension of the Kalman Filter. Kalman Smoothing utilizes current, past, and future measurements to improve positioning accuracy. It integrates noisy pseudorange measurements with high-precision Doppler shift data to smooth pseudoranges, refine state estimates, and reduce errors. Unlike traditional carrier phase smoothing, which often suffers from degradation and cycle slips in urban environments, Kalman Smoothing is more resilient to these issues. This method is particularly suitable for high-precision GNSS positioning, especially where real-time data is limited or post-processing is required, and it has gained wide attention for its ability to leverage larger datasets to enhance estimation accuracy.

In this project, we systematically evaluate the three aforementioned methods. First, WLS is applied to remove noise and generate a covariance matrix. Next, a Kalman Filter is constructed based on WLS results to achieve precise real-time position estimates. Subsequently, Kalman Smoothing is used to further refine the Kalman Filter results by integrating past and future measurement data to fine-tune state estimates. Finally, the effectiveness of these three methods is assessed using Root Mean Square Error (RMSE) [6] and Percentile-Error Evaluation (PEE) [7], with errors visualized in relation to ground truth values on a map for intuitive analysis.

2. Literature Review

2.1. Global Navigation Satellite Systems (GNSS)

The development of GNSS has evolved from early satellite navigation systems to a comprehensive, multi-system framework essential for modern navigation and positioning. The first system, TRANSIT, developed by the U.S. in the 1960s, led to the NAVSTAR GPS project in 1973, with global coverage achieved by 1993 through a network of 24 satellites [8]. Concurrently, the Soviet Union developed GLONASS, achieving initial deployment by 1995 [9]. Significant advancements in the 2000s included modernization of GPS and GLONASS, the initiation of Europe's Galileo system [10], and the development of China's BeiDou system, offering global service by 2020 [11]. Today, GNSS comprises interoperable systems like GPS, GLONASS, Galileo, and BeiDou, enhancing accuracy and reliability. These advancements support applications from everyday navigation to specialized fields such as precision agriculture and autonomous vehicles, underscoring GNSS's crucial role in modern society.

2.2. Adaptive Filtering in GNSS

Adaptive filtering is a critical technique in signal processing, providing real-time adaptability to changing signal characteristics. Prominent algorithms include Least Mean Squares (LMS) [12], Normalized LMS (NLMS) [13], and Recursive Least Squares (RLS) [14], each offering varying balances of convergence speed and computational complexity. Adaptive filters are extensively used in noise cancellation, echo suppression, and system identification due to their dynamic adjustment capabilities.

In GNSS applications, adaptive filtering addresses challenges such as multipath effects, signal interference, and dynamic environmental conditions. Multipath mitigation is achieved by dynamically filtering out reflected signal components, enhancing positioning accuracy. Interference suppression is managed through adaptive identification and exclusion of unwanted signals, ensuring cleaner data. Moreover, adaptive filters adjust to varying environments, maintaining reliable GNSS performance. Additionally, adaptive filtering facilitates sensor fusion, where algorithms like the Kalman Filter integrate GNSS data with inputs from inertial measurement units (IMUs) [15] and other sensors, yielding more accurate and robust positioning. These applications underscore the significance of adaptive filtering in achieving high-precision GNSS solutions, crucial for advanced navigation, autonomous systems, and other high-stakes applications.

2.3. RTKLIB and GNSS Positioning Techniques

RTKLIB [16] is an open-source GNSS processing software library developed by Japanese scientist Tomoji Takasu, designed for both real-time kinematic (RTK) [17] and post-processing kinematic (PPK) [18] high-precision positioning. It supports multiple GNSS systems, including GPS, GLONASS, Galileo, and BeiDou, thus enhancing positioning accuracy and reliability through diverse satellite data. RTKLIB offers various positioning modes such as Single Point Positioning (SPP) [19], Differential GNSS (DGNSS) [20], RTK, and Precise Point Positioning (PPP) [21], catering to a wide range of application needs.

SPP, the simplest method, is suitable for low-precision applications, relying solely on satellite signals. DGNSS improves accuracy by correcting receiver data with reference station observations, mitigating atmospheric and clock errors. RTK, ideal for dynamic and real-time applications, uses carrier phase differences between the receiver and reference stations to achieve centimeter-level accuracy. PPP delivers high precision globally by leveraging precise satellite orbit and clock data without requiring real-time communication with reference stations. RTKLIB offers flexibility with its support for multiple data formats and real-time data streams via serial and network interfaces. Its high-precision algorithms, including Kalman filtering, enhance stability and accuracy. As an open-source tool, RTKLIB is highly customizable, with extensive configuration options to tailor algorithms and processes. The software includes user-friendly interfaces like RTKPOST for post-processing and RTKNAVI for real-time navigation.

Widely used in research and commercial applications, RTKLIB is essential in fields such as autonomous driving, precision agriculture, GIS, and surveying, providing robust tools for high-precision GNSS positioning.

2.4. Accumulated Doppler Range (ADR) data

ADR data is crucial for high-precision GNSS positioning, offering centimeter-level accuracy through carrier phase measurements [22]. By measuring the continuous phase of GNSS satellite signals, ADR provides greater precision than traditional pseudorange measurements. Over time, these measurements mitigate random errors and noise, enhancing positioning stability and reliability. ADR is fundamental to advanced GNSS techniques like Real-Time Kinematic (RTK) and Precise Point Positioning (PPP), which utilize differential corrections and precise satellite data, respectively, for superior accuracy.

Despite its advantages, ADR processing faces chal-

lenges such as cycle slips and multipath effects, necessitating sophisticated detection and correction algorithms. ADR applications include autonomous vehicle navigation, precision agriculture, and geophysical monitoring, offering high-precision geospatial measurements essential for topographic surveys, cadastral mapping, and infrastructure development. Realizing the full potential of ADR requires high-quality GNSS receivers capable of precise carrier phase tracking and robust signal processing to address environmental and technical challenges.

2.5. Factor Graph Optimization (FGO)

Factor Graph Optimization (FGO) is a robust framework extensively used in robotics, computer vision, and GNSS positioning to solve complex estimation problems [23]. It represents variables and their interdependencies through a bipartite graph comprising variable nodes and factor nodes, which correspond to the system's constraints or measurements. This representation allows for the efficient formulation and solution of nonlinear least squares optimization problems.

FGO's strength lies in its ability to integrate diverse information sources, enhancing the accuracy and robustness of the estimated variables. In robotics, FGO is central to Simultaneous Localization and Mapping (SLAM) [24], enabling precise trajectory estimation and environmental mapping. In GNSS, FGO amalgamates measurements like pseudorange and carrier phase data to improve positioning precision.

The optimization process typically employs iterative nonlinear optimization algorithms such as Gauss-Newton, Levenberg-Marquardt, and incremental smoothing and mapping (ISAM), which iteratively refine variable estimates to minimize overall error. The flexibility and scalability of factor graphs make them ideal for handling large-scale, real-world problems, providing a powerful tool for high-precision applications.

3. Methods

3.1. Weighted Least Square (WLS)

Weighted Least Squares (WLS) is an extension of the ordinary least squares (OLS) [25] method that accounts for varying degrees of reliability in the data points by assigning different weights to them. This technique minimizes the weighted sum of squared residuals, leading to more accurate parameter estimates when the observations have non-constant variances. Normally, WLS includes 4 main components: Objective Function, Weight Matrix, Normal Equations, Choosing Weights. The objective of WLS is to minimize the following weighted sum of squared residuals:

$$\min_{\beta} \sum_{i=1}^{n} w_i (y_i - X_i \beta)^2 \tag{1}$$

where:

- w_i is the weight for the i-th observation.
- y_i is the observed value.
- X_i is the row vector of predictor variables for the i-th observation.
- β is the vector of parameters to be estimated.

The weights are represented in a diagonal matrix W:

$$W = \text{diag}(w_1, w_2, \ldots, w_n) \tag{2}$$

The WLS estimates are obtained by solving the weighted normal equations:

$$\hat{\beta} = (X^T W X)^{-1} X^T W y \tag{3}$$

where: X is the matrix of predictor variables and y is the vector of observed values.

Weights w_i are typically chosen as the inverse of the variance of the measurement errors:

$$w_i = \frac{1}{\sigma_i^2} \tag{4}$$

where: σ_i^2 is the variance of the i-th measurement error.

Implement WLS in a GNSS positioning algorithm to enhance accuracy by weighting measurements according to their reliability.

3.2. Kalman Filter (KF)

KF is a recursive algorithm used for estimating the state of a dynamic system from a series of incomplete and noisy measurements. Named after Rudolf E. Kalman, who published the foundational paper in 1960, the KF is widely used in various fields, including signal processing, control systems, and GNSS positioning. Normally, WLS includes following main steps:

- **State Transition Model:**The system is represented in a state-space model, comprising a state transition model and a measurement model.

  $$\mathbf{x}_k = \mathbf{F}_k \mathbf{x}_{k-1} + \mathbf{B}_k \mathbf{u}_k + \mathbf{w}_k \tag{5}$$

  where: $\mathbf{x}_k$ is the state vector at time k. $\mathbf{F}_k$ is the state transition matrix. $\mathbf{B}_k$ is the control input matrix. $\mathbf{u}_k$ is the control input vector. $\mathbf{w}_k$ is the process noise vector, assumed to be Gaussian with covariance $\mathbf{Q}_k$.

- **Measurement Model:**

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \tag{6}$$

where: $\mathbf{z}_k$ is the measurement vector at time k. $\mathbf{H}_k$ is the measurement matrix. $\mathbf{v}_k$ is the measurement noise vector, assumed to be Gaussian with covariance $\mathbf{R}_k$.

Kalman Filter Equations include two stepsPrediction Step and Update Step.

- **Prediction Step:**

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_k \hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}_k \mathbf{u}_k \tag{7}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^T + \mathbf{Q}_k \tag{8}$$

where: $\hat{\mathbf{x}}_{k|k-1}$ is the predicted state estimate and $\mathbf{P}_{k|k-1}$ is the predicted error covariance.

- **Update Step:**

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \tag{9}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1}) \tag{10}$$

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1} \tag{11}$$

where: $\mathbf{K}_k$ is the Kalman gain, $\hat{\mathbf{x}}_{k|k}$ is the updated state estimate, and $\mathbf{P}_{k|k}$ is the updated error covariance.

In GNSS applications, Kalman Filtering is used to improve the accuracy and reliability of position estimates by integrating GNSS measurements with dynamic models of the receiver's motion. The GNSS observation model involves pseudorange or carrier phase measurements, which can be incorporated into the measurement model of the Kalman Filter.

KF method has many advantages in GNSS:

- **Real-Time Processing:** The recursive nature of the KF makes it suitable for real-time GNSS applications.
- **Error Mitigation:** Effectively reduces the impact of measurement noise and improves the accuracy of position estimates.
- **Adaptive:** Can be extended to non-linear models using the Extended Kalman Filter (EKF) [26] or the Unscented Kalman Filter (UKF) [27], making it versatile for various GNSS scenarios.

3.3. Kalman Smoothing (KS)

KS is an extension of the Kalman Filter designed to improve the accuracy of state estimates by considering not only past and present measurements but also future ones. It provides smoothed estimates of the state vector by incorporating all available data over a given time period, resulting in more accurate and reliable estimates compared to filtering alone. The two most common KS techniques are the Rauch-Tung-Striebel (RTS) [28] smoother and the fixed-interval smoother [29]. We introduce and use the RTS mainly.

1. **Kalman Filter Recap:** This part is same with the KF steps.
2. **Kalman Smoothing:**
   - **Forward Pass:**The first step involves running the standard KF forward in time to compute the filtered estimates $\hat{\mathbf{x}}_{k|k}$ and the error covariance $\mathbf{P}_{k|k}$.
   - **Backward Pass:**The second step involves running a backward pass to refine these estimates by incorporating future measurements.
3. **Rauch-Tung-Striebel (RTS) Smoother:** The RTS smoother is a popular fixed-interval smoothing algorithm that uses the results from the forward pass (KF) and performs a backward pass to produce smoothed estimates. RTS Smoothing Equations:

$$\mathbf{G}_k = \mathbf{P}_{k|k} \mathbf{F}_{k+1}^T \mathbf{P}_{k+1|k}^{-1} \tag{12}$$

$$\hat{\mathbf{x}}_{k|N} = \hat{\mathbf{x}}_{k|k} + \mathbf{G}_k (\hat{\mathbf{x}}_{k+1|N} - \hat{\mathbf{x}}_{k+1|k}) \tag{13}$$

$$\mathbf{P}_{k|N} = \mathbf{P}_{k|k} + \mathbf{G}_k (\mathbf{P}_{k+1|N} - \mathbf{P}_{k+1|k}) \mathbf{G}_k^T \tag{14}$$

where:

- $\hat{\mathbf{x}}_{k|N}$ is the smoothed state estimate at time k.
- $\mathbf{P}_{k|N}$ is the smoothed error covariance.
- $\mathbf{G}_k$ is the smoothing gain.

In GNSS applications, KS is used to enhance the accuracy of position estimates by considering the entire sequence of measurements over a time interval, rather than just relying on current and past measurements.

KS method has many advantages in GNSS:

- **Enhanced Accuracy:** By considering future measurements, KS provides more accurate and reliable position estimates than filtering alone.
- **Noise Reduction:** Reduces measurement noise impact, yielding smoother position estimates.
- **Post-Processing:** Particularly useful in scenarios where real-time processing is not critical, allowing for post-processing to achieve higher accuracy.

4. Experiments

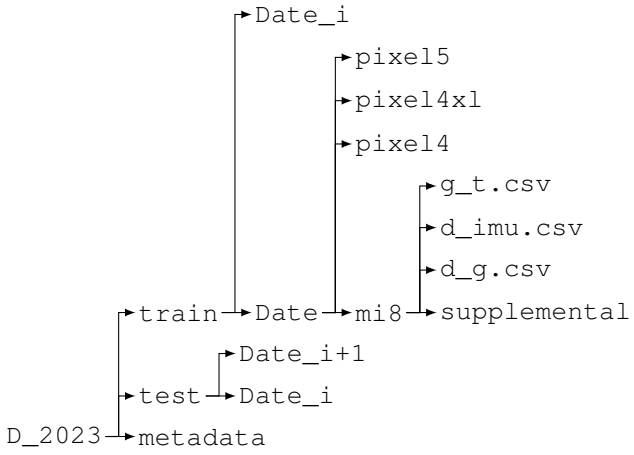
4.1. Dataset and Preprocessing

4.1.1 Dataset Description

The dataset for this project originates from the **Smartphone Decimeter Challenge 2023** [30], facilitated by Google's data research team. It includes data from four Android smartphones (Pixel 4, Pixel 4 XL, Pixel 5, Mi 8), presenting challenges such as noise and variations from different satellite configurations. The dataset comprises GNSS logs, ground truth CSV files, and IMU readings.

GNSS logs provide raw GPS positions in Earth-Centered, Earth-Fixed (ECEF) coordinates, which require conversion to latitude and longitude for practical application. Ground truth CSV files contain accurate location data for evaluating the solutions' accuracy. IMU readings, including accelerometer and gyroscope data, offer supplementary data for positioning, calibration, and error compensation, enhancing robustness.

Spanning from May 2020 to December 2023, this comprehensive dataset includes positioning data sampled by various smartphones. The data is stored in device_gnss.csv, device_imu.csv, and ground_truth.csv files, with positional information in ECEF coordinates. This detailed dataset provides a rich foundation for addressing challenges and advancing GNSS positioning accuracy.



The provided .csv files contain 58 columns capturing various aspects such as time, position, satellite information, frequency, and associated uncertainties. These columns are crucial for understanding the dataset. As shown in Figure 1, the path data is notably noisy and exhibits low accuracy. Key columns in this dataset include:

- `svid`: Denotes the satellite index.
- `utcTimeMillis`: Represents the current time in GNSS clock timestamp.
- `SvPosition[X/Y/Z] EcefMeters`: Indicates the satellite's position in ECEF coordinates.
- `SvVelocity[X/Y/Z] EcefMeters PerSecond`: Describes the satellite's velocity in ECEF coordinates.
- `Pseudorange`: The measured pseudorange.
- `UncertaintyMeters`: The error estimate of the position.
- `PseudorangeRate`: The rate of change of the pseudorange.
- `UncertaintyMetersPerSecond`: The error estimate of the velocity.

4.1.2 Data Preprocessing

Raw data was initially extracted from the GNSS database, including measurements from multiple satellites. Identifying and removing outliers and corrupted data points was essential to ensure dataset reliability and prevent skewed results.

The data was then organized by satellite source, accounting for variations across GNSS systems. Key information, such as position (latitude, longitude, altitude), sampling time, and velocity, was extracted through GNSS protocol analysis.

1. **Data Retrieval and Cleaning:**
   - GNSS was extracted for analysis.
   - Outliers and corrupted data were removed to ensure data integrity and quality.
2. **Data Grouping:**
   - Data was grouped by satellite source to facilitate efficient GNSS protocol analysis.
3. **GNSS Protocol Analysis:**
   - Key positional information (latitude, longitude, altitude), sampling times, and velocities were extracted.
   - Earth-Centered Earth-Fixed (ECEF) coordinates were converted to geographic coordinates (latitude and longitude) [31].

This systematic approach ensures the accuracy and reliability of the data used for subsequent processing, establishing a robust foundation for high-precision GNSS positioning analysis.

4.2. Method Formulation

4.2.1 WLS Estimation

In GNSS positioning, Weighted Least Squares (WLS) is used to estimate position and velocity parameters. Accurate initial values are crucial for WLS convergence and error estimation, so a preliminary normal Least Squares (LS) [32] method is first applied.

The WLS step produces:

- x_{wls} **and** v_{wls}**:** Matrices representing the estimated position and velocity (longitude, latitude, altitude).
- cov_x **and** cov_v**:** Covariance matrices indicating the uncertainty of each sample.

GNSS positioning depends on pseudorange measurements from multiple satellites, which vary in accuracy due to satellite geometry, signal quality, and atmospheric conditions. The basic observation model for these measurements accounts for these variations, refining the position estimates.

This systematic WLS approach, based on initial LS estimates, ensures accurate and reliable GNSS positioning by addressing measurement variabilities.

We could build the basic observation model for GNSS pseudorange measurements is:

$$\rho_i = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} + c\delta t + \epsilon_i \tag{15}$$

where: ρ_i is the observed pseudorange. (x, y, z) is the receiver's position. (x_i, y_i, z_i) are the satellite's coordinates. c is the speed of light. δt is the receiver clock bias. ϵ_i is the measurement noise.

Rewrite the observation model in matrix form:

$$\rho = \mathbf{H}\mathbf{p} + \mathbf{e} \tag{16}$$

where: ρ is the vector of observed pseudoranges. $\mathbf{H}$ is the design matrix relating satellite positions to the receiver position. $\mathbf{p}$ is the vector of unknowns (receiver position and clock bias). $\mathbf{e}$ is the vector of measurement errors.

Construct the weight matrix W based on the variances of the pseudorange measurements:

$$W = \text{diag}\left(\frac{1}{\sigma_1^2}, \frac{1}{\sigma_2^2}, \ldots, \frac{1}{\sigma_n^2}\right) \tag{17}$$

Apply the weighted normal equations to solve for the unknowns:

$$\hat{\mathbf{p}} = (\mathbf{H}^T\mathbf{W}\mathbf{H})^{-1}\mathbf{H}^T\mathbf{W}\rho \tag{18}$$

4.2.2 WLS+KF Estimation

Following WLS estimation, Kalman Filtering was applied to refine position and velocity estimates. The variable correspondences are detailed below.

State Vector: Typically includes position, velocity, and possibly other parameters such as clock bias.

$$\mathbf{x}_k = \begin{bmatrix} x_k & y_k & z_k & v_{x,k} & v_{y,k} & v_{z,k} & \delta t_k \end{bmatrix}^T \tag{19}$$

State Transition Model: Reflects the motion dynamics of the receiver, often modeled as constant velocity or constant acceleration.

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{I} & \Delta t \cdot \mathbf{I} & 0 \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{20}$$

Measurement Model: Relates the GNSS measurements (e.g., pseudoranges) to the state vector.

Prediction Step: Uses the state transition model to predict the next state and error covariance.

Update Step: Incorporates the GNSS measurements to update the state estimate and error covariance.

4.2.3 WLS+KS Estimation

Following KF, KS was applied. The forward step of smoothing mirrors standard KF, predicting state estimates and updating them with new measurements for refined positioning results. The observation model remains consistent, using pseudorange or carrier phase measurements.

Implementing Kalman Smoothing:

- **Forward Pass:** Execute the standard Kalman Filter to obtain filtered estimates $\hat{\mathbf{x}}_{k|k}$ and $\mathbf{P}_{k|k}$.
- **Backward Pass:** Use the RTS smoothing equations to refine these estimates by incorporating future measurements.

4.3. Evaluation and Visualization

4.3.1 Distance Errors

The error, defined as the absolute distance between estimated positions and the ground truth, is illustrated for each sample across different models in the accompanying figure. The average error, calculated as the mean of all sample errors, is presented in Figure 1 below.

Figure 1 compares error values across the Original Measurement, WLS, KF, and KS models. The Original Measurement exhibits the highest error variability, with significant spikes indicating substantial deviations from the ground truth. WLS partially mitigates these errors, refining initial estimates. KF consistently reduces errors by integrating dynamic modeling and measurement

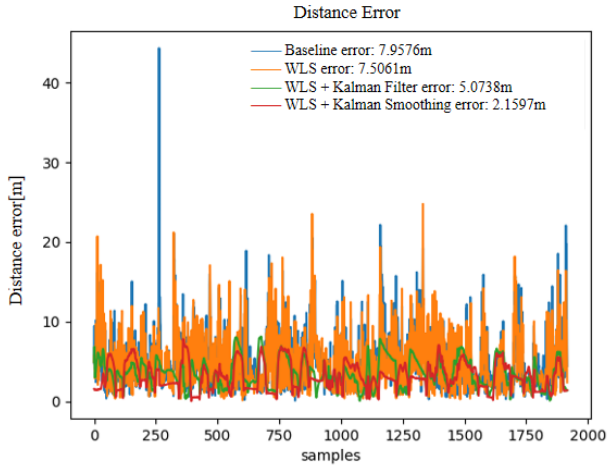


Figure 1: Distance Error

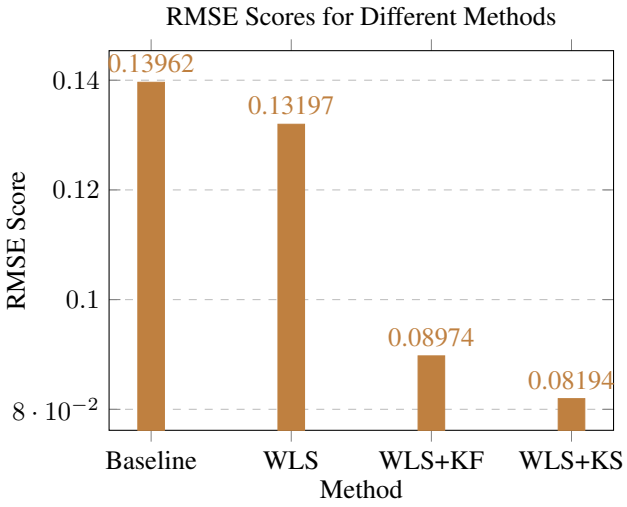


Figure 2: RMSE Scores for Different Methods

updates. KS achieves the most significant error reduction by leveraging both past and future measurements for the most accurate position estimates. Figure 1 quantifies these improvements, showcasing the progressive enhancement in positioning accuracy through advanced filtering and smoothing techniques.

4.3.2 Evaluation Scores

1. **RMSE Score:**

   The Root Mean Square Error (RMSE) provides a measure of the average deviation between predicted values and actual values, assuming errors follow a normal distribution. The formula for RMSE is:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{21}$$

   where y_i is the actual value, $\hat{y}_i$ is the predicted value, and n is the number of samples. RMSE is particularly sensitive to large deviations due to squaring all errors, giving more weight to larger errors.

   The RMSE scores obtained shows in Figure 2:

   These results indicate that applying WLS significantly reduces the error compared to the baseline. As shown in Figure 2, further improvements are seen with the KF, and the best performance is achieved with KS.

2. **Percentile-Error Evaluation Score (PEE):**

   The PEE Score calculates the mean of the 50th and 95th percentile distance errors, providing insights into the distribution characteristics and specific error positions. The formula for percentile error evaluation includes: Distance Error Calculation, Percentile Errors (50th & 95th Percentile Error) and Average Error Calculation, Overall Mean Error.

$$a = \sin^2\left(\frac{\Delta\varphi}{2}\right) + \cos(\varphi_1)\cdot\cos(\varphi_2)\cdot\sin^2\left(\frac{\Delta\lambda}{2}\right) \tag{22}$$

$$c = 2\cdot\text{atan2}\left(\sqrt{a}, \sqrt{1-a}\right) \tag{23}$$

$$d = R\cdot c \tag{24}$$

   where: $\Delta\varphi$ is the difference in latitude. $\Delta\lambda$ is the difference in longitude. φ_1 and φ_2 are the latitudes. R is the Earth's radius. d is the distance error.

$$\text{Average Error} = \frac{\text{50th P-Error} + \text{95th P-Error}}{2} \tag{25}$$

$$\text{Overall Mean Error} = \frac{\sum \text{Average Errors}}{\text{Number of Devices}} \tag{26}$$

The results and Figure 3 demonstrate a significant error reduction with advanced filtering techniques, with KS achieving the lowest errors, indicating enhanced positioning accuracy and reliability.

Evaluation metrics show the effectiveness of WLS, KF, and KS in improving GNSS positioning accuracy. Both RMSE and Percentile-Error scores decrease progressively with these techniques. KS provides the most precise estimates, effectively reducing both average and percentile errors. This approach, using WLS for initial

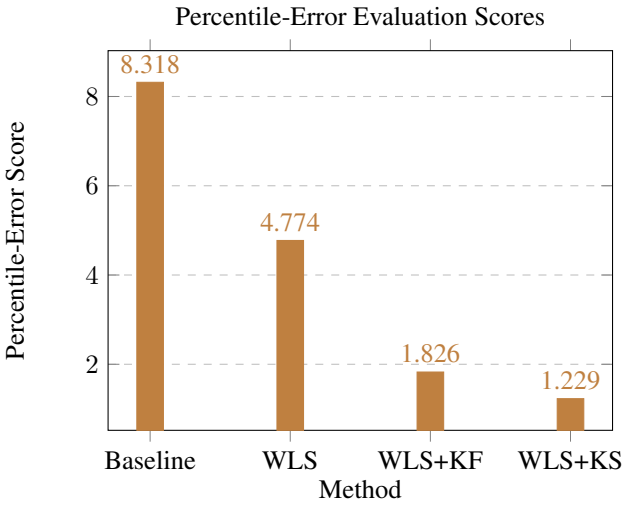


Figure 3: PEE Scores for Different Methods

refinement and advanced Kalman techniques for optimization, substantially improves GNSS positioning performance.

4.3.3 Path Visualization

Using Folium, Figure 4 visualized the estimated positions on a map, comparing the paths derived from each method against the ground truth. Despite consistent performance improvements across methods, including KS, none achieved an average error below 1 meter. Thus, Percentile-Error Evaluation offers a more nuanced assessment. Our results, with errors between 1 and 2 meters, are competitive and align with the standards observed among other participants.

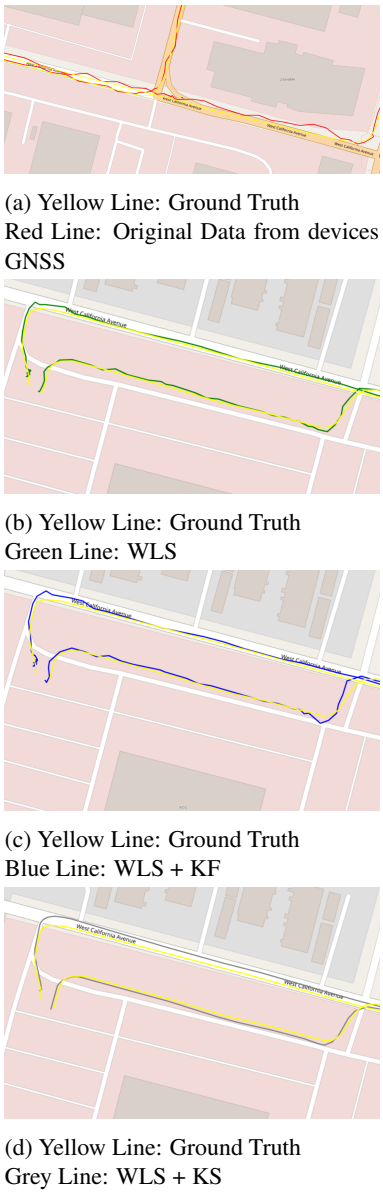
5. Conclusion

This project validated the effectiveness of Weighted Least Squares (WLS), Kalman Filtering (KF), and Kalman Smoothing (KS) in enhancing GNSS positioning accuracy. Each technique offers unique benefits and trade-offs.

WLS provides foundational estimates and convenient covariance computation but lacks dynamic updates, resulting in limited performance improvement.

KF significantly enhances positioning accuracy with real-time data processing, suitable for navigation and other real-time applications. However, it relies solely on current and past measurements, which may not fully eliminate errors.

KS achieves the highest accuracy by utilizing both past and future data, ideal for post-processing applications. Its primary limitation is the requirement for off-time processing and access to the entire dataset, making it unsuitable for real-time use.



(a) Yellow Line: Ground Truth
Red Line: Original Data from devices GNSS

(b) Yellow Line: Ground Truth
Green Line: WLS

(c) Yellow Line: Ground Truth
Blue Line: WLS + KF

(d) Yellow Line: Ground Truth
Grey Line: WLS + KS

Figure 4: Comparison of Different GNSS Positioning Techniques

In summary, WLS is essential for initial parameter estimation, KF is optimal for real-time applications with notable accuracy improvements, and KS excels in post-processing scenarios, delivering the highest precision. Designers must consider the trade-offs between performance, complexity, and real-time requirements when developing GNSS systems.

References

[1] Guoyu Michael Fu, Mohammed Khider, and Frank Van Diggelen. Android raw gnss measurement datasets

for precise positioning. In *Proceedings of the 33rd international technical meeting of the satellite division of the Institute of Navigation (ION GNSS+ 2020)*, pages 1925–1937, 2020.

[2] Arthur Gelb et al. *Applied optimal estimation*. MIT press, 1974.

[3] Daniel Alazard. Introduction to kalman filtering. *SUPAERO*, 2005.

[4] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

[5] Per K Enge. The global positioning system: Signals, measurements, and performance. *International Journal of Wireless Information Networks*, 1:83–105, 1994.

[6] Timothy O Hodson. Root mean square error (rmse) or mean absolute error (mae): When to use them or not. *Geoscientific Model Development Discussions*, 2022:1–10, 2022.

[7] George A Barnard. Statistical inference. *Journal of the Royal Statistical Society. Series B (Methodological)*, 11(2):115–149, 1949.

[8] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and James Collins. *Global positioning system: theory and practice*. Springer Science & Business Media, 2012.

[9] Sergey Revnivykh, Alexey Bolkunov, Alexander Serdyukov, and Oliver Montenbruck. Glonass. *Springer Handbook of Global Navigation Satellite Systems*, pages 219–245, 2017.

[10] John L Heilbron. *Galileo*. OUP Oxford, 2010.

[11] Yuanxi Yang, Weiguang Gao, Shuren Guo, Yue Mao, and Yufei Yang. Introduction to beidou-3 navigation satellite system. *Navigation*, 66(1):7–18, 2019.

[12] Bernard Widrow, Marcian E Hoff, et al. Adaptive switching circuits. In *IRE WESCON convention record*, volume 4, pages 96–104. New York, 1960.

[13] Paulo SR Diniz et al. *Adaptive filtering*, volume 4. Springer, 1997.

[14] John Cioffi and Thomas Kailath. Fast, recursive-least-squares transversal filters for adaptive filtering. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(2):304–337, 1984.

[15] George Schmidt and George Schmidt. Gps/ins technology trends for military systems. In *Guidance, Navigation, and Control Conference*, page 3826, 1998.

[16] Tomoji Takasu and Akio Yasuda. Development of the low-cost rtk-gps receiver with an open source program package rtklib. In *International symposium on GPS/GNSS*, volume 1, pages 1–6. International Convention Center Jeju Korea Seogwipo-si, Republic of Korea, 2009.

[17] Gemunu Senadeera Gurusinghe, Takashi Nakatsuji, Yoichi Azuta, Prakash Ranjitkar, and Yordphol Tanaboriboon. Multiple car-following data with real-time kinematic global positioning system. *Transportation Research Record*, 1802(1):166–180, 2002.

[18] Xiaohong Zhang, Yuxi Zhang, and Feng Zhu. A method of improving ambiguity fixing rate for post-processing kinematic gnss data. *Satellite Navigation*, 1:1–13, 2020.

[19] Antonio Angrisano, Salvatore Gaglione, and Ciro Gioia. Performance assessment of gps/glonass single point positioning in an urban environment. *Acta Geodaetica et Geophysica*, 48:149–161, 2013.

[20] Chris Rizos, Volker Janssen, Craig Roberts, and Th Grinter. Precise point positioning: Is the era of differential gnss positioning drawing to an end? 2012.

[21] Jan Kouba, François Lahaye, and Pierre Tétreault. Precise point positioning. *Springer handbook of global navigation satellite systems*, pages 723–751, 2017.

[22] Yoji Goto, Tetsuo Iwata, Katsuhisa Yamashina, Hironori Susaki, and Osamu Arai. Estimation of high precision carrier-phase by accumulated-delta-range (adr) data for reduction of the multipath problem and evaluation by 2d-music. In *Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003)*, pages 1908–1913, 2003.

[23] Weisong Wen, Tim Pfeifer, Xiwei Bai, and Li-Ta Hsu. Factor graph optimization for gnss/ins integration: A comparison with the extended kalman filter. *NAVIGATION: Journal of the Institute of Navigation*, 68(2):315–331, 2021.

[24] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 13(2):99–110, 2006.

[25] Clara Dismuke and Richard Lindrooth. Ordinary least squares. *Methods and designs for outcomes research*, 93(1):93–104, 2006.

[26] Keisuke Fujii. Extended kalman filter. *Refernce Manual*, 14:41, 2013.

[27] Eric A Wan and Rudolph Van Der Merwe. The unscented kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 adaptive systems for signal processing, communications, and control symposium (Cat. No. 00EX373)*, pages 153–158. Ieee, 2000.

[28] Patrick Nima Raanes. On the ensemble rauch-tung-striebel smoother and its equivalence to the ensemble kalman smoother. *Quarterly Journal of the Royal Meteorological Society*, 142(696):1259–1264, 2016.

[29] Joseph E Wall Jr, Alan S Willsky, and Nils R Sandell Jr. On the fixed-interval smoothing problem. *Stochastics*, 5(1-2):1–41, 1981.

[30] Guoyu (Michael) Fu, Mohammed Khider, Vivek Gulati, and Frank van Diggelen. Google smartphone decimeter challenge. `https://kaggle.com/competitions/smartphone-decimeter-2023/data`, September 2023. Retrieved [Date Retrieved].

[31] Yifeng Zhou, Henry Leung, and Martin Blanchette. Sensor alignment with earth-centered earth-fixed (ecef) coordinate system. *IEEE Transactions on Aerospace and Electronic systems*, 35(2):410–418, 1999.

[32] Steven J Miller. The method of least squares. *Mathematics Department Brown University*, 8(1), 2006.