Smartphone positioning optimization based on GNSS/IMU integration and AI-Based stop detection

2355029 SHEN SHILIN

SUPERVISOR: PROF. NOBUAKI KUBO

Overview

- Background
- GNSS-Based Positioning
- IMU-Based Optimization and Integration Method
- AI-Based IMU Stop Detection Method
- Stop Detection Method Comparison
- Experimental Results and Evaluation

Background

Motivation:

Participation in the Google Smartphone Decimeter Challenge 2023–2024

Research Goal:

Develop smartphone positioning program by myself

Post processing

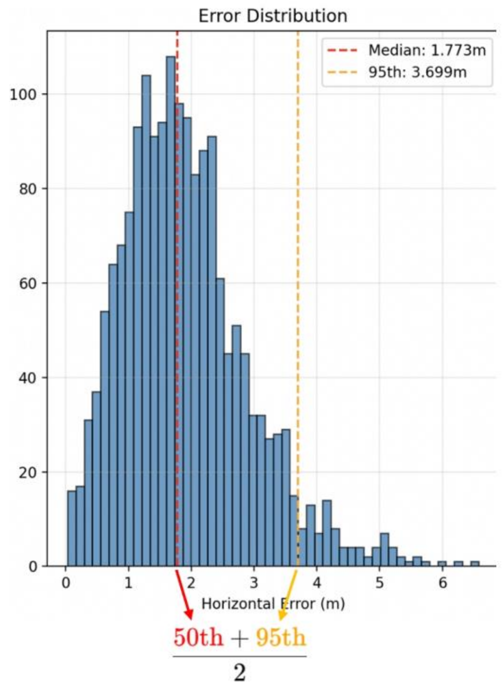
Google Smartphone Decimeter Challenge 2023-2024

Overview Data Code Models Discussion Leaderboard Rules

13 濬詩霖SHEN Shilin

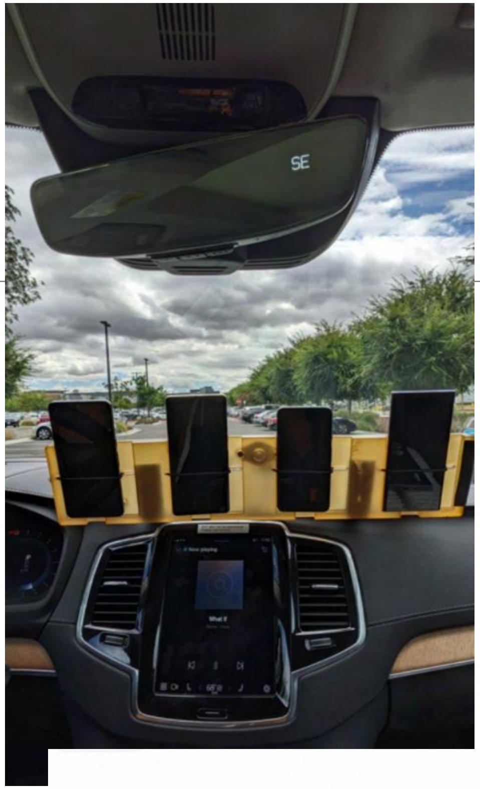
Horizontal positioning errors

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



Data Collection

For data collection, GSDC used a car and fixed several smartphones near the front windshield using a special holder. To make sure all phones were in the same position each time, using 3D-printed parts and a robot arm to measure and check smartphone location carefully.



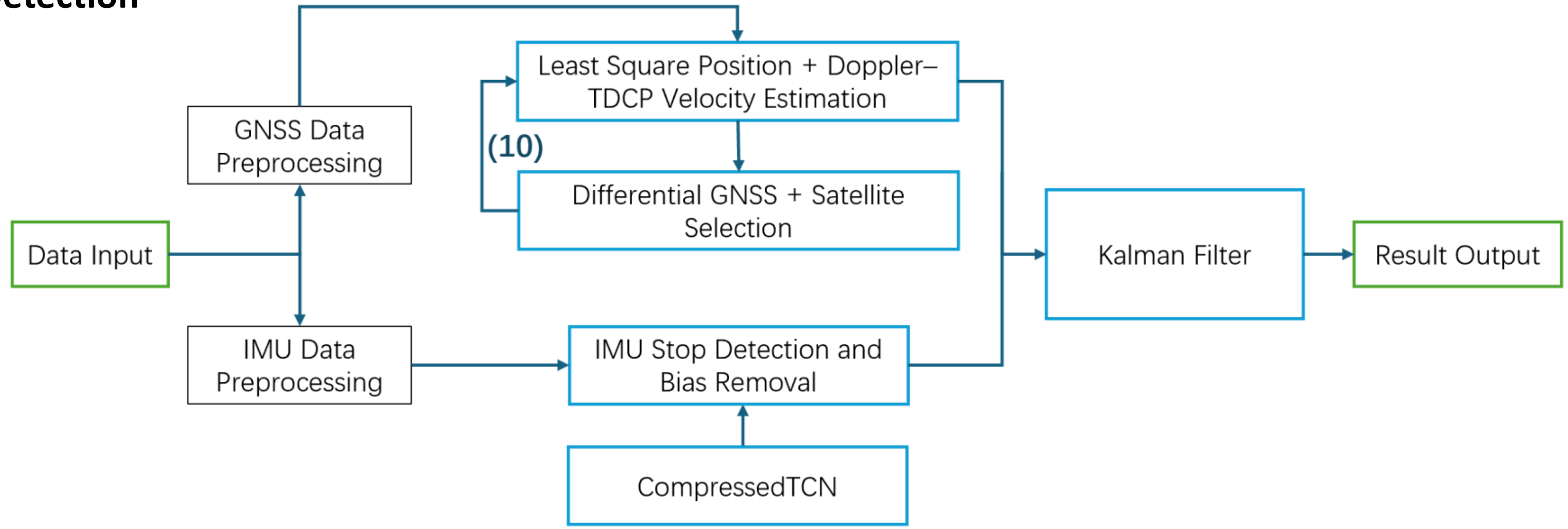
Smartphone Types:
Three major brands were used for data collection:

Brand	Models
Google	Pixel Series
Xiaomi	Mi 8 Series
Samsung	S Series

Pixel 6 pro

Technical Objectives

1. Multi-Sensor Fusion
2. Differential GNSS
3. Carrier-Phase Differencing (TDCP)
4. Kalman Filter Design
5. IMU Denoising with Butterworth Low-Pass Filter
6. AI-Based Stop Detection



Performance Objective & Theoretical Contribution

Performance Objective

Reduce the average 2D positioning error from over 30 meters to less than 7 meters (2DRMS) on the GSDC dataset, and ensure the system works in real time for various urban scenarios.

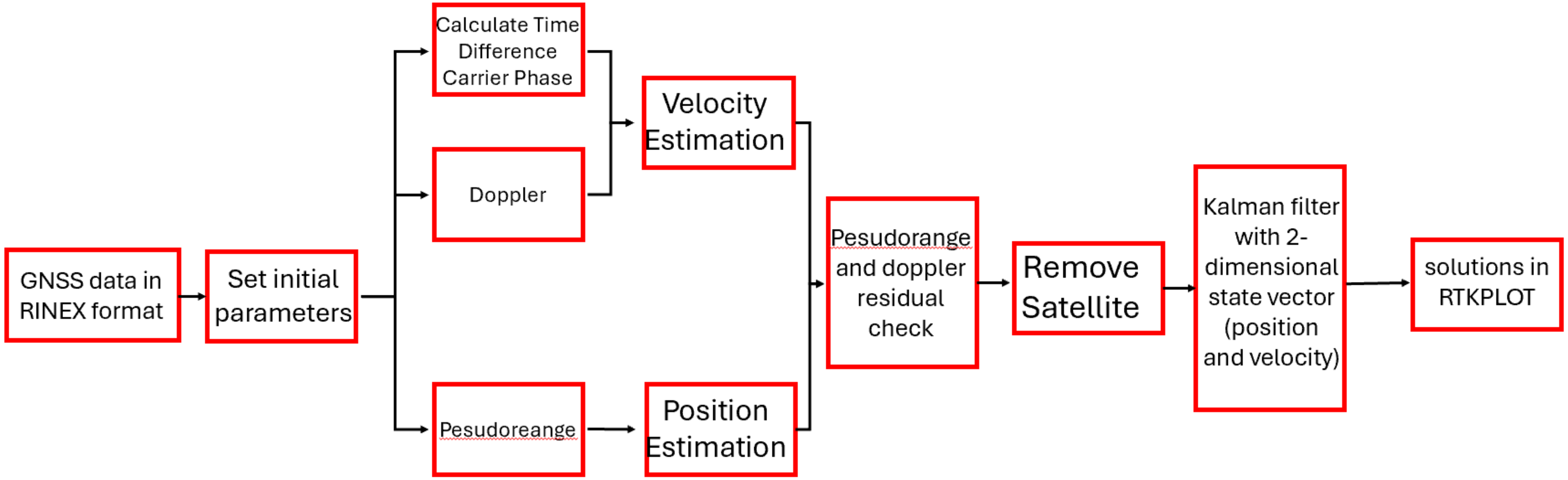
Theoretical Contribution

Provide theoretical support and real-world experiments to push forward the use of AI-enabled GNSS/IMU systems for smartphones.

GNSS-Based Positioning

- GNSS Data Preprocessing
- Differential GNSS Algorithms for Position Estimation
- Doppler and TDCP Velocity Fusion Strategy for Velocity Estimation
- Suspect Satellite Removal
- GNSS position-velocity integration Positioning Algorithm

GNSS positioning framework



Differential GNSS Algorithms for Position Estimation

Differential GNSS Processing Framework

Data Input
Base Station + Rover Observations

→

SPP Processing
Single Point Positioning for Base

→

Corrections
Calculate Differential Terms

→

DGNSS Solution
Enhanced Rover Position

Input Data Sources

- Base Station Observations (P^{base})
- Rover Observations (P^{rov})
- Navigation Messages
- Clock & Orbital Parameters

Processing Steps

Initialize
X_0, Y_0, Z_0, clock bias

↓

Compute Residuals
$r = P - \text{corrections} - \rho$

↓

Form Jacobian
A matrix ($n \times 4$)

↓

Weighted Least Squares
$\Delta x = (A^T P A)^{-1} A^T P r$

↓

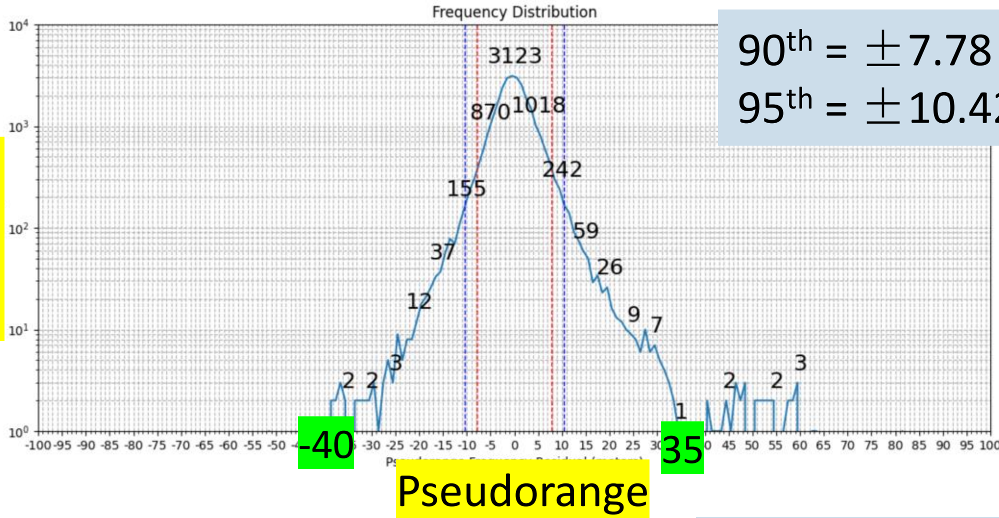
Update State
$X \leftarrow X + \Delta X$

⟹ **Check** if $||\Delta x|| < 0.001$ m iteration finish

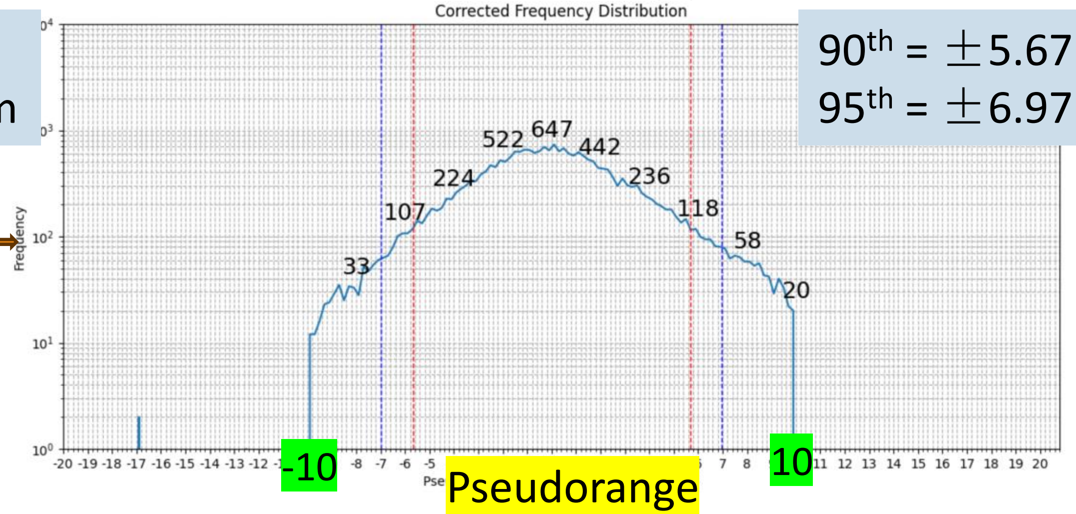
Suspect Satellite Removal

Mitigation of Suspect Satellites Using Pseudorange Residuals $r_i^{rov} = P_i^{rov} + SV\_corrtime_i + Correction_i - \rho_i^{rov}$

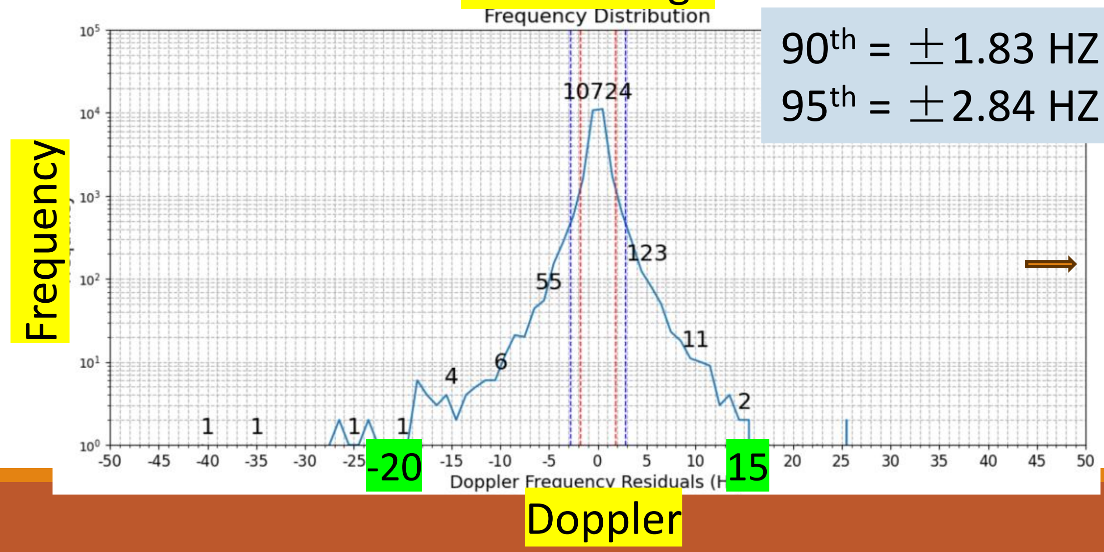
Mitigation of Suspect Satellites Using Doppler Residuals $Residual_i^{doppler} = Dp1_i - \left(\frac{f_c}{c}\right) \cdot [(\vec{v}_s - \vec{v}_r) \cdot \vec{e}_i]$



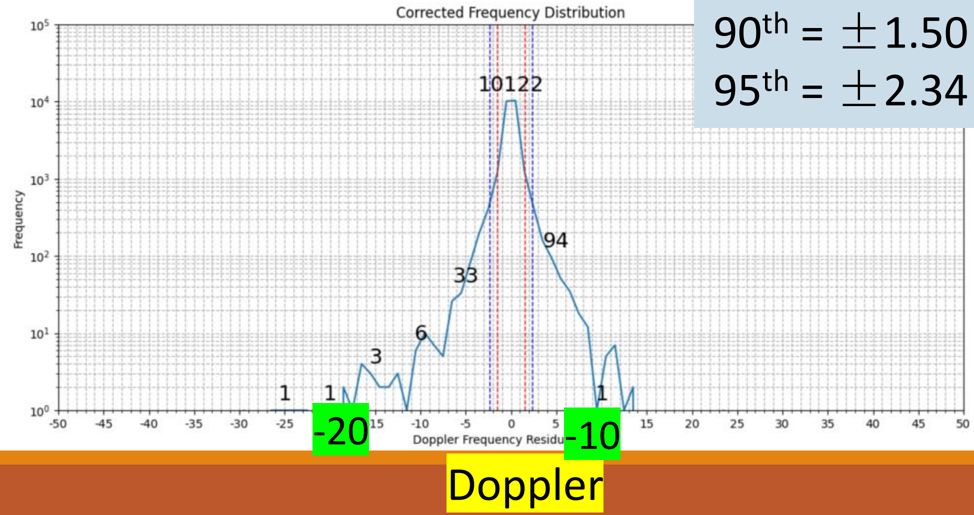
90th = ±7.78 m
95th = ±10.42 m



90th = ±5.67 m
95th = ±6.97 m



90th = ±1.83 HZ
95th = ±2.84 HZ



90th = ±1.50 HZ
95th = ±2.34 HZ

① Doppler and TDCP for Velocity Estimation

Velocity Fusion Strategy

Gross Error Threshold (5.0 m/s):

$$\left| \hat{v}_{lat}^{TDCP} - \frac{\hat{v}_{lat}^{Doppler} + v_{lat}^{pre}}{2} \right| < 5.0$$

$$\left| \hat{v}_{lon}^{TDCP} - \frac{\hat{v}_{lon}^{Doppler} + v_{lon}^{pre}}{2} \right| < 5.0$$

Significance Threshold (1.0 m/s):

$$\left| \hat{v}_{lat}^{TDCP} - \frac{\hat{v}_{lat}^{Doppler} + v_{lat}^{pre}}{2} \right| > 1.0$$

$$\left| \hat{v}_{lon}^{TDCP} - \frac{\hat{v}_{lon}^{Doppler} + v_{lon}^{pre}}{2} \right| > 1.0$$

$$TDCP_i = \frac{CP_i(t_2) - CP_i(t_1)}{\Delta t}$$

Velocity Decision

$$\mathbf{v}_{final} = \begin{cases} \mathbf{v}_{TDCP}, & \text{if TDCP passes both threshold tests} \\ \mathbf{v}_{Doppler}, & \text{otherwise} \end{cases}$$

② Mitigation of Suspect Satellites

③ GNSS position-velocity integration Positioning Algorithm

GNSS signals sometimes are missing. If we can detect stop by IMU the result could be improved

Kalman Filter

observation equation

$$\mathbf{x}_k = \begin{bmatrix} x_k \\ y_k \\ v_{x,k} \\ v_{y,k} \end{bmatrix} \qquad \mathbf{z}_k = \begin{bmatrix} x_k^{\text{obs}} \\ y_k^{\text{obs}} \\ v_{x,k}^{\text{obs}} \\ v_{y,k}^{\text{obs}} \end{bmatrix} = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k$$

Prediction step:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1}$$
$$\mathbf{P}_{k|k-1} = \mathbf{F}\mathbf{P}_{k-1|k-1}\mathbf{F}^\top + \mathbf{Q}$$

Update step:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}^\top \left(\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^\top + \mathbf{R}\right)^{-1}$$
$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\left(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1}\right)$$
$$\mathbf{P}_{k|k} = \left(\mathbf{I} - \mathbf{K}_k\mathbf{H}\right)\mathbf{P}_{k|k-1}$$

Positioning Result

(Lat,Lon,alt)

IMU-Based Optimization and Integration Method

Data Source and Characteristics

Synchronization and Resampling

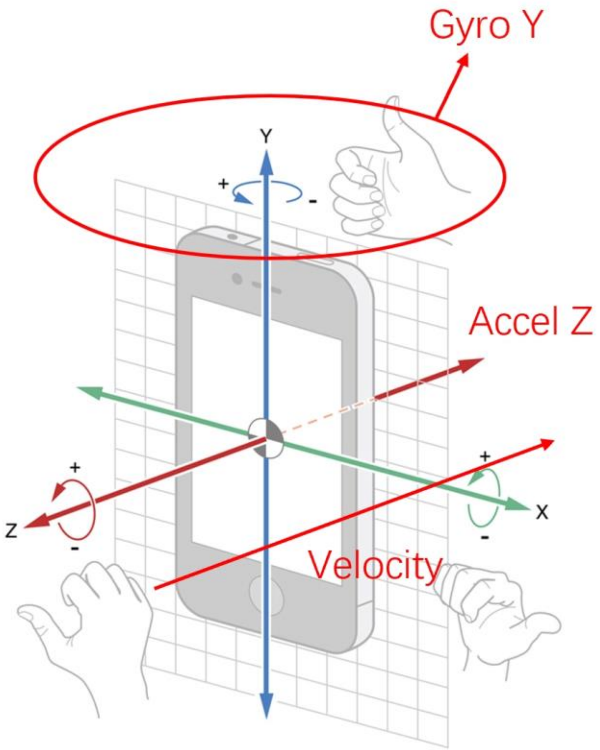
Low-Pass Filter

Threshold-Based Stop Detection Algorithm

Data Source and Characteristics

- device_imu.csv ⟶ Raw IMU values

- **utcTimeMillis**: The timestamp of each measurement expressed in milliseconds since the Unix epoch (January 1, 1970, UTC);
- **MessageType**: An identifier indicating the sensor type, distinguishing between accelerometer data (`UncalAccel`) and gyroscope data (`UncalGyro`);
- **MeasurementX, MeasurementY, MeasurementZ**: The raw measurement values along the device coordinate frame's X, Y, and Z axes, provided in units of m/s^2 for accelerometers and rad/s for gyroscopes.



Synchronization and Resampling

1. Sliding Window Averaging Resampling
2. Linear Interpolation for Missing Data

Raw IMU Data Structure

utcTimeMillis	AccelerateX	AccelerateY	AccelerateZ	GyroscopeX	GyroscopeY	GyroscopeZ
1638989539917	-0.14416756	9.672985	-1.9513469			
1638989539922				-0.0074831	-0.00503964	-0.002138
1638989539926	-0.10169496	9.6173525	-1.7934206			
1638989539935	-0.13938192	9.559925	-1.8173488			
1638989539940				-0.0009163	-0.00106901	-0.001069
1638989539945	-0.13459627	9.592826	-1.8837496			
1638989539954	-0.12981063	9.602397	-1.8604196			
1638989539958				-0.0038179	-0.00122173	-0.0007636
1638989539963	-0.12023934	9.559925	-1.7934206			
1638989539972	-0.09212367	9.626924	-1.8077775			
1638989539977				0.00580322	0	0.00045815
1638989539981	-0.15792629	9.653843	-2.012962			
1638989539990	-0.08793623	9.669994	-1.841277			
1638989539995				-0.0056505	-0.0009163	0.00045815
1638989540000	-0.11066805	9.531809	-1.783251			
1638989540009	-0.12981063	9.5455675	-1.7886349			
1638989540013				0.00610865	0.000458149	-0.0006109
1638989540018	-0.15373886	9.64487	-1.9411774			
1638989540027	-0.10169496	9.65504	-1.8891335			

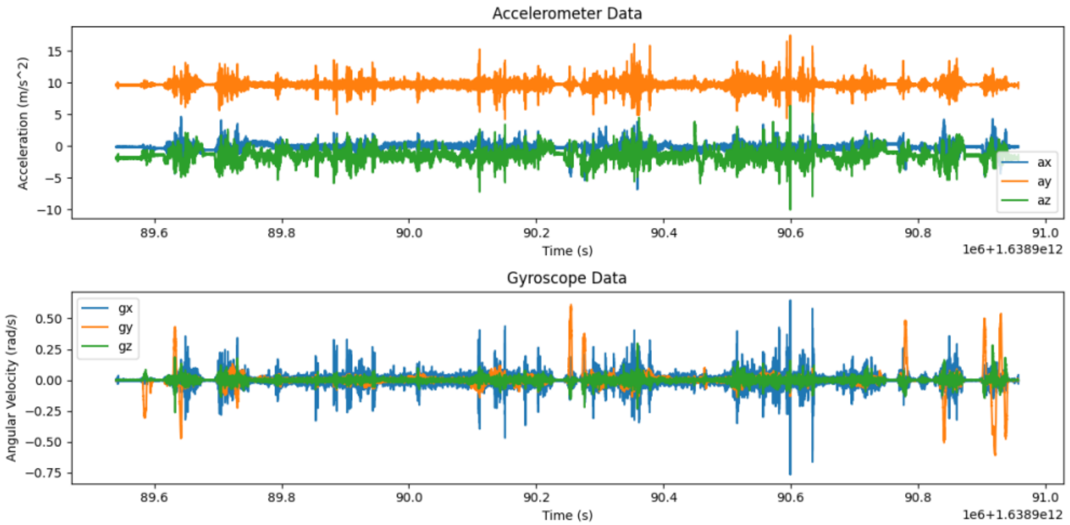
After Synchronization and Resampling (100HZ)

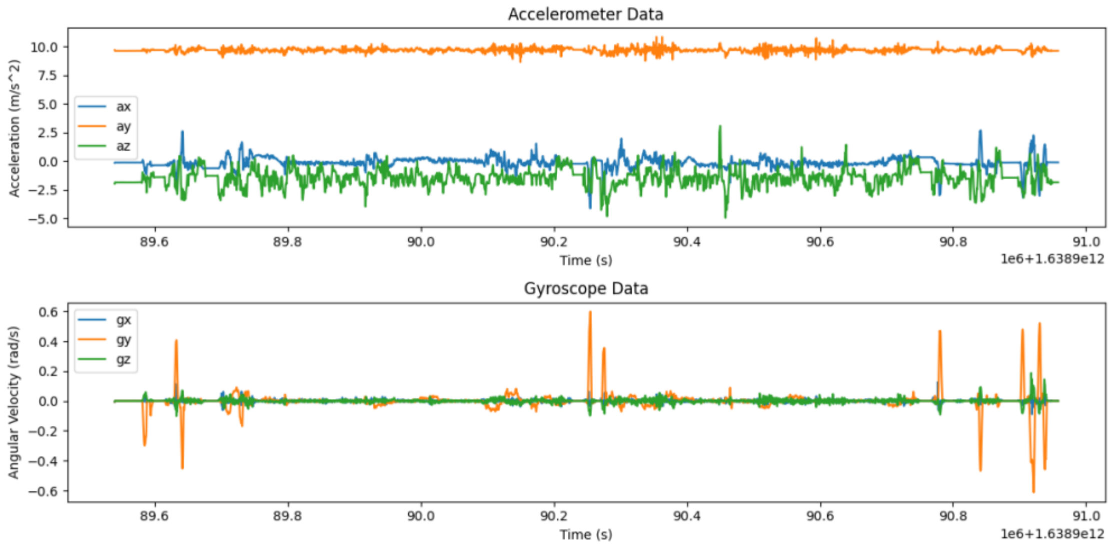
utcTimeMillis	AccelerateX	AccelerateY	AccelerateZ	GyroscopeX	GyroscopeY	GyroscopeZ
1638989539910	-0.14416756	9.672985	-1.9513469	-0.0074831	-0.00503964	-0.002138
1638989539920	-0.10169496	9.6173525	-1.7934206	-0.0074831	-0.00503964	-0.002138
1638989539930	-0.13938192	9.559925	-1.8173488	-0.0041997	-0.00305433	-0.0016035
1638989539940	-0.13459627	9.592826	-1.8837496	-0.0009163	-0.00106901	-0.001069
1638989539950	-0.12981063	9.602397	-1.8604196	-0.0038179	-0.00122173	-0.0007636
1638989539960	-0.12023934	9.559925	-1.7934206	0.00099266	-0.00061087	-0.0001527
1638989539970	-0.09212367	9.626924	-1.8077775	0.00580322	0	0.00045815
1638989539980	-0.15792629	9.653843	-2.012962	7.64E-05	-0.00045815	0.00045815
1638989539990	-0.08793623	9.669994	-1.841277	-0.0056505	-0.0009163	0.00045815
1638989540000	-0.12023934	9.53868825	-1.785943	0.00022907	-0.00022907	-7.64E-05
1638989540010	-0.15373886	9.64487	-1.9411774	0.00610865	0.000458149	-0.0006109
1638989540020	-0.10169496	9.65504	-1.8891335	0.00114537	-7.64E-05	7.64E-05
1638989540030	-0.07716853	9.564711	-1.8311075	-0.0038179	-0.00061087	0.00076358
1638989540040	-0.10169496	9.598209	-1.8364913	-0.0013744	-0.00038179	0.00022907
1638989540050	-0.10169496	9.640682	-1.9124635	0.00106901	-0.00015272	-0.0003054
1638989540060	-0.10169496	9.602996	-1.7886349	-0.0007636	-0.00061087	-0.0004581
1638989540070	-0.11126626	9.593424	-1.7886349	0.00015272	-0.00053451	-0.0001527
1638989540080	-0.14416756	9.644272	-1.9842482	0.00106901	-0.00045815	0.00015272
1638989540090	-0.12562318	9.659226	-1.9274186	-0.0006109	-0.00022907	-0.0003818

Low-Pass Filter

Butterworth low-pass filter

$$H(s) = \frac{1}{\sqrt{1 + \left(\frac{s}{\omega_c}\right)^{2n}}}$$



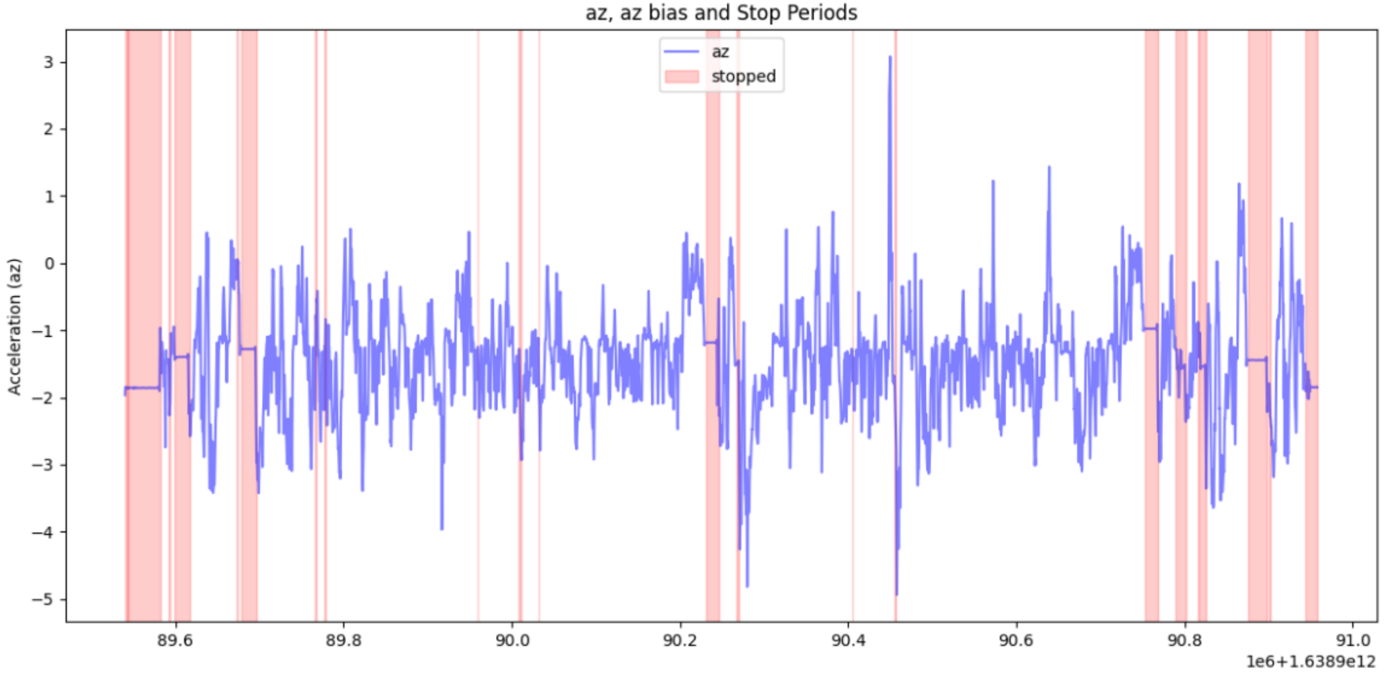


Threshold-Based Stop Detection Algorithm

The algorithm defines adaptive thresholds based on the minimum observed standard deviation values. A time point t is considered a potential stop if all of the following conditions are satisfied:

$$\sigma_a(t) < \min(\sigma_a) + 0.01, \quad \sigma_\omega(t) < \min(\sigma_\omega) + 0.01, \quad \|\boldsymbol{\omega}(t)\| < 0.05$$

$$\text{is\_stopped}(t) = \begin{cases} 1, & \text{if all conditions are satisfied} \\ 0, & \text{otherwise} \end{cases}$$



Threshold-Based Stop detection performance

File	Accuracy (%)	Precision (%)	Recall (%)
2021-12-07-19-22-us-ca-lax-d	98.24	95.32	87.63
2021-12-07-22-21-us-ca-lax-g	96.03	99.48	87.50
2021-12-08-17-22-us-ca-lax-a	97.16	85.57	75.45
2021-12-08-18-52-us-ca-lax-b	98.03	100.00	79.71
2021-12-08-20-28-us-ca-lax-c	97.18	99.32	85.47
2021-12-09-17-06-us-ca-lax-e	98.46	100.00	88.67
2022-01-11-18-48-us-ca-mtv-n	95.76	99.51	83.82
2022-04-01-18-22-us-ca-lax-t	97.89	91.07	66.23
2022-05-13-20-57-us-ca-mtv-pe1	97.13	94.67	72.45
2023-03-08-21-34-us-ca-mtv-u	98.09	100.00	83.20
2023-09-06-00-01-us-ca-routen	97.64	100.00	91.06
2023-09-06-18-47-us-ca	98.92	100.00	79.22
2023-09-07-19-33-us-ca	96.69	85.71	82.76
2023-09-07-22-47-us-ca-routebc2	98.36	98.37	92.80
Total Average	**97.54**	**96.36**	**82.57**

Improve?

AI-Based IMU Stop Detection Method

- Data Source and Label Generation
- Model Architecture & Residual Dilated Convolution & Receptive Field
- Training Result and Stop Performance

Data Source and Label Generation

The ground truth trajectories are provided at a frequency of 1 Hz. Serve as the label y.

Labeling rule: If position is nearly unchanged for at least 5 seconds, it's labeled as "stop"; otherwise, "move".

A	B	C	D	E	F
MessageTy	Provider	LatitudeDegrees	LongitudeDegrees	AltitudeMe	SpeedMps
Fix	GT	34.1759672	-118.4652808	178.3346	0.001414
Fix	GT	34.1759672	-118.4652808	178.3346	0.001414
Fix	GT	34.1759672	-118.4652808	[illegible]	[illegible]
Fix	GT	34.1759672	-118.4652808	178.3336	0.002237
Fix	GT	34.1759672	-118.4652808	178.3336	0.002235
Fix	GT	34.1759671	-118.4652808	178.3326	0.003604
Fix	GT	34.1759671	-118.4652808	178.3306	0
Fix	GT	34.1759654	-118.4652807	178.3246	0.520642
Fix	GT	34.175958	-118.4652784	178.3236	1.134147
Fix	GT	34.175946	-118.465272	178.3086	1.682416
Fix	GT	34.1759299	-118.4652622	178.2876	2.337498
Fix	GT	34.1759103	-118.4652497	178.2806	2.53507
Fix	GT	34.1758888	-118.4652372	178.2196	2.848074
Fix	GT	34.1758615	-118.4652244	178.1956	3.42567

>= 5 second

Feature Design:

Six channels from the IMU: AccelerateX, AccelerateY, AccelerateZ and GyroscopeX, GyroscopeY, GyroscopeZ.

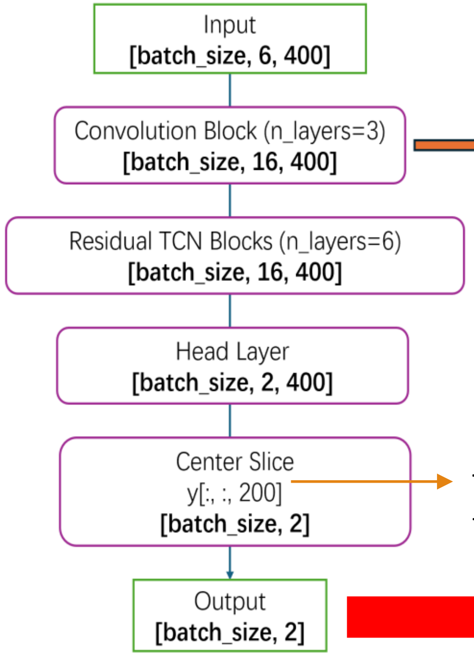
Frequency of 100 Hz. Serve as the input tensor x.

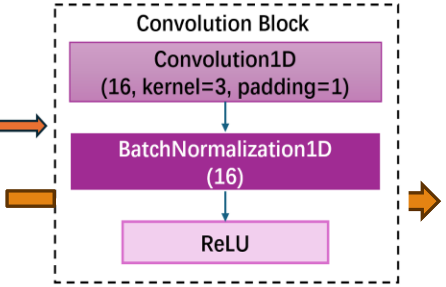
Model Architecture & Residual Dilated Convolution& Receptive Field

Python 3.12. PyTorch

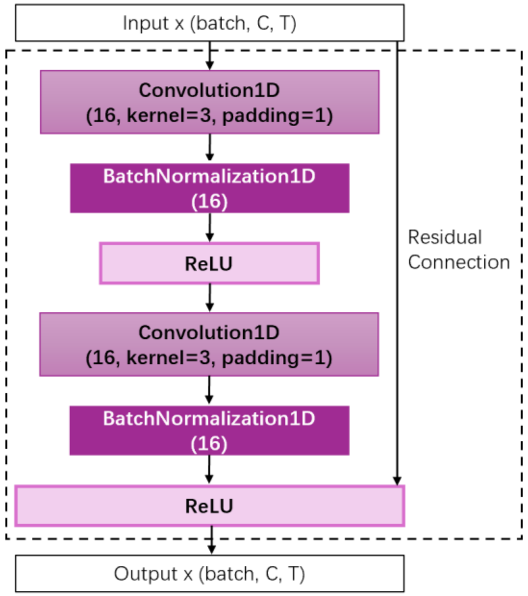
CompressedTCN model

Temporal Convolutional Network





Residual Block (dilation=2^n, padding= 2^n)



The window is centered at each 1Hz ground truth timestamp

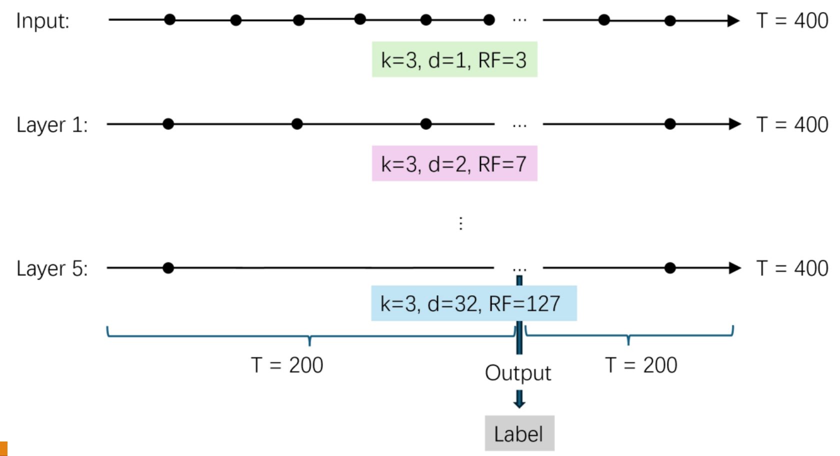
Final output result is classification ‘stop’ or ‘move’

Receptive Field Design

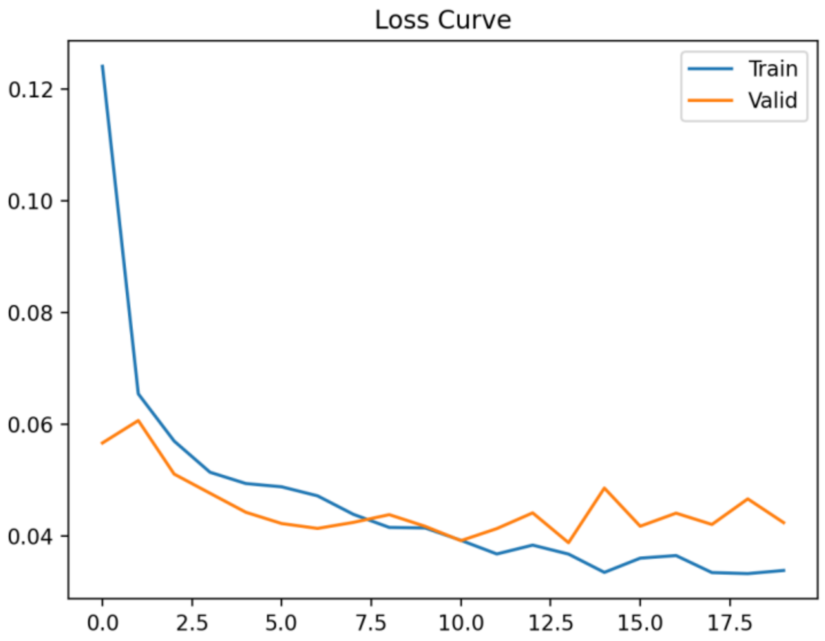
Receptive field calculation for each layer in CompressedTCN

Layer Index	Dilation Rate (d)	Receptive Field (RF)
0	1	3
1	2	7
2	4	15
3	8	31
4	16	63
5	32	127

Dilated convolutions and receptive field



Training Result and Stop Performance



Epoch	Train Loss	Train Accuracy	Valid Loss	Valid Accuracy
20	0.0338	0.9878	0.0424	0.9850

Classification Report on Validation Set

Label	Precision	Recall	F1-score	Support
0 (Static)	0.9950	0.9874	0.9912	3409
1 (Moving)	0.9311	0.9716	0.9509	598
Accuracy		0.9850		4007
Macro Avg	0.9630	0.9795	0.9710	4007
Weighted Avg	0.9854	0.9850	0.9852	4007

AI-based Stop detection performance

Trajectory	**AI-based**		
	Accuracy (%)	Precision (%)	Recall (%)
2021-12-07-19-22-us-ca-lax-d	98.92	93.87	97.07
2021-12-07-22-21-us-ca-lax-g	98.67	97.60	98.25
2021-12-08-17-22-us-ca-lax-a	97.92	84.25	91.45
2021-12-08-18-52-us-ca-lax-b	98.66	92.95	94.77
2021-12-08-20-28-us-ca-lax-c	98.37	95.95	95.40
2021-12-09-17-06-us-ca-lax-e	99.18	98.65	95.42
2022-01-11-18-48-us-ca-mtv-n	98.94	97.66	98.43
2022-04-01-18-22-us-ca-lax-t	99.32	93.90	93.90
2022-05-13-20-57-us-ca-mtv-pe1	98.20	96.84	84.79
2023-03-08-21-34-us-ca-mtv-u	99.46	96.83	98.39
2023-09-06-00-01-us-ca-routen	98.97	97.37	98.89
2023-09-06-18-47-us-ca	99.73	95.45	100.00
2023-09-07-19-33-us-ca	99.08	95.31	96.83
2023-09-07-22-47-us-ca-routebc2	99.27	97.99	98.24
Total Average	**98.80**	**94.92**	**95.51**

Stop Detection Method Comparison

Trajectory	Traditional			AI-based		
	Accuracy (%)	Precision (%)	Recall (%)	Accuracy (%)	Precision (%)	Recall (%)
2021-12-07-19-22-us-ca-lax-d	98.24	95.32	87.63	98.92	93.87	97.07
2021-12-07-22-21-us-ca-lax-g	96.03	99.48	87.50	98.67	97.60	98.25
2021-12-08-17-22-us-ca-lax-a	97.16	85.57	75.45	97.92	84.25	91.45
2021-12-08-18-52-us-ca-lax-b	98.03	100.00	79.71	98.66	92.95	94.77
2021-12-08-20-28-us-ca-lax-c	97.18	99.32	85.47	98.37	95.95	95.40
2021-12-09-17-06-us-ca-lax-e	98.46	100.00	88.67	99.18	98.65	95.42
2022-01-11-18-48-us-ca-mtv-n	95.76	99.51	83.82	98.94	97.66	98.43
2022-04-01-18-22-us-ca-lax-t	97.89	91.07	66.23	99.32	93.90	93.90
2022-05-13-20-57-us-ca-mtv-pe1	97.13	94.67	72.45	98.20	96.84	84.79
2023-03-08-21-34-us-ca-mtv-u	98.09	100.00	83.20	99.46	96.83	98.39
2023-09-06-00-01-us-ca-routen	97.64	100.00	91.06	98.97	97.37	98.89
2023-09-06-18-47-us-ca	98.92	100.00	79.22	99.73	95.45	100.00
2023-09-07-19-33-us-ca	96.69	85.71	82.76	99.08	95.31	96.83
2023-09-07-22-47-us-ca-routebc2	98.36	98.37	92.80	99.27	97.99	98.24
Total Average	**97.54**	**96.36**	**82.57**	**98.80**	**94.92**	**95.51**

Experimental Results and Evaluation

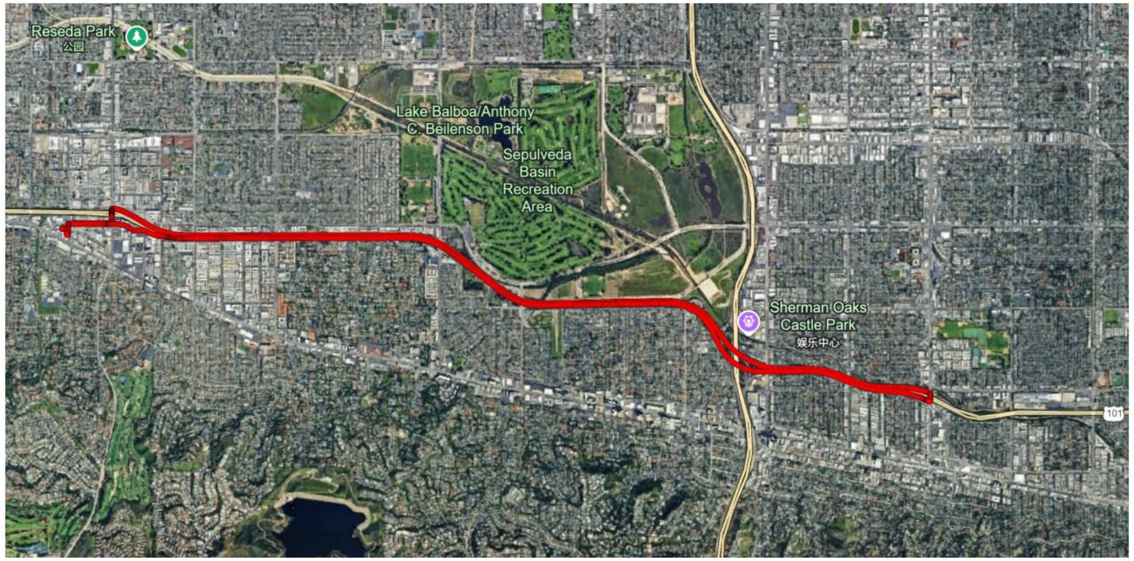
Data: 14 different trajectories in the training dataset

representative trajectory: 2021-12-08-18-52-us-ca-lax-b

Program Setting

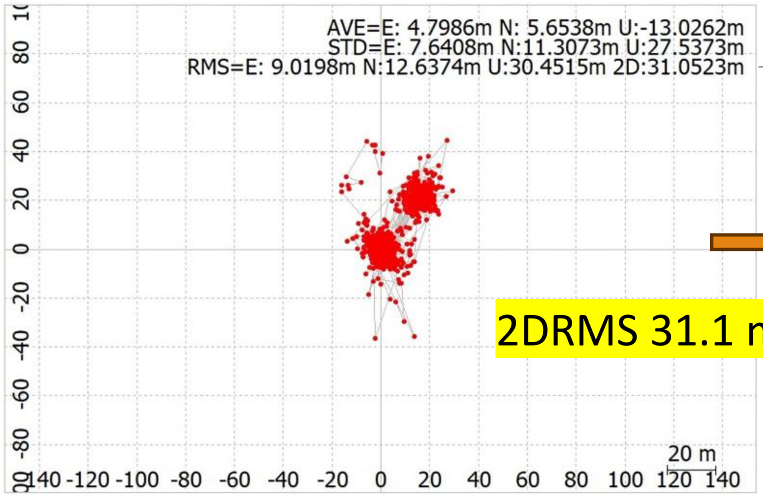
```
Mask_angle 10.0
Rov_obs train\2021-12-07-19-22-us-ca-lax-d\pixel6pro\supplemental\gnss_log.obs
Ref_obs train\2021-12-07-19-22-us-ca-lax-d\vdcy3410.obs
Nav_file train\2021-12-07-19-22-us-ca-lax-d\BRDM00DLR_S_20213410000_01D_MN.nav
POSreflat 34.17856659
POSreflon -118.22000501
POSrefhgt 318.230
Code_noise 0.025
Carrier_noise 0.005
Iteration 2900
Threshold_cn 28.0
RTK_DGNSS 1
Ratio_limit 2.0
GPS 1
QZSS 1
GALILEO 1
BEIDOU 0
GLONASS 1
GPSWEEK 2187
```

The trajectory located in the San Fernando Valley area of Los Angeles, California, USA.
The route begins near Highway 101.

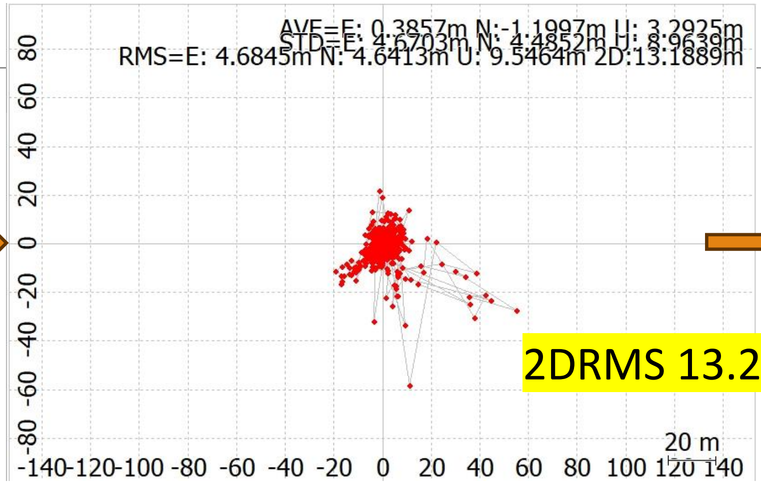


Positioning results showing 2DRMS distribution

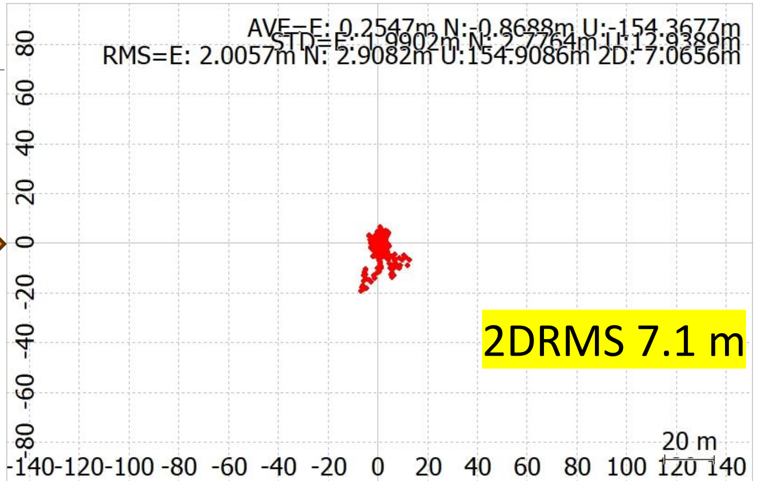
Initial DGNSS



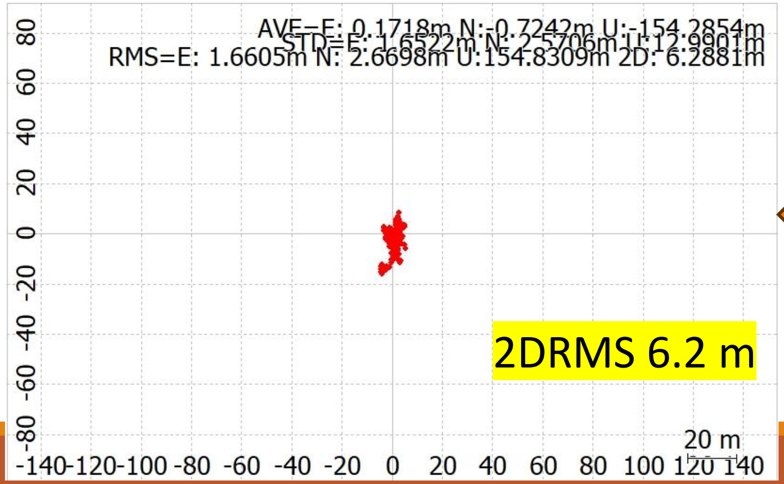
DGNSS + Remove suspect satellites



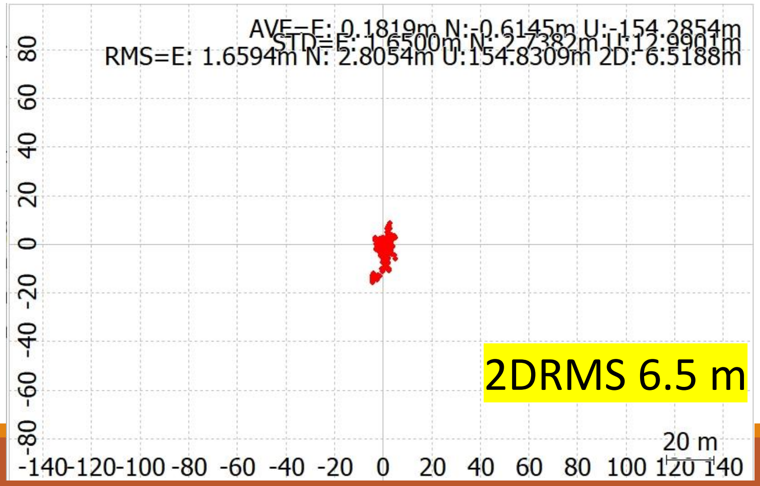
Kalman Filter



Kalman Filter + AI-Based Stop detection



Kalman Filter + Threshold-Based Stop detection



14 trajectories Horizontal positioning errors

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

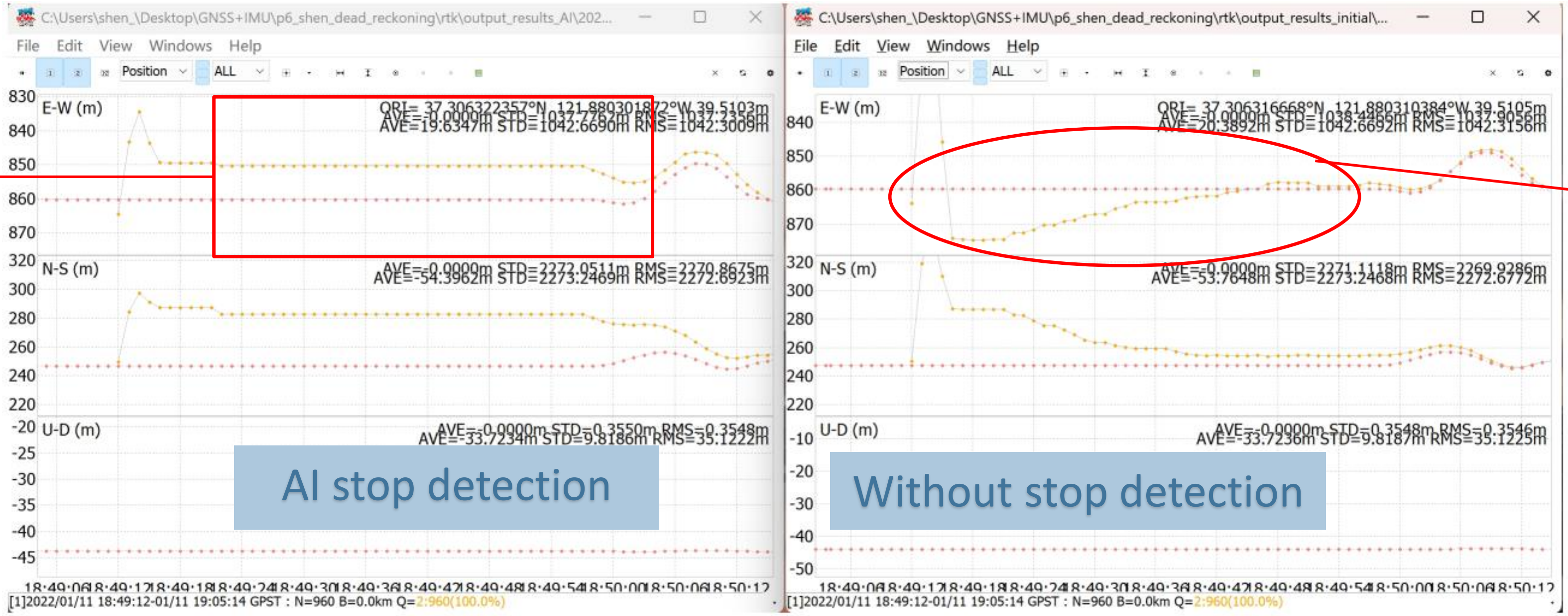
3.880

File	GNSS Only (m)	GNSS+IMU Traditional (m)	GNSS+IMU AI-Base (m)
2021-12-07-19-22-us-ca-lax-d	2.982	3.036	2.81
2021-12-07-22-21-us-ca-lax-g	3.75	3.207	2.764
2021-12-08-17-22-us-ca-lax-a	3.113	3.357	3.202
2021-12-08-18-52-us-ca-lax-b	4.38	4.354	4.385
2021-12-08-20-28-us-ca-lax-c	2.942	2.811	2.813
2021-12-09-17-06-us-ca-lax-e	3.376	3.287	3.092
2022-01-11-18-48-us-ca-mtv-n	5.706	21.588	19.415
2022-04-01-18-22-us-ca-lax-t	6.345	6.374	6.411
2022-05-13-20-57-us-ca-mtv-pe1	2.425	2.399	2.389
2023-03-08-21-34-us-ca-mtv-u	3.472	3.267	3.294
2023-09-06-00-01-us-ca-routen	4.059	3.376	3.12
2023-09-06-18-47-us-ca	4.188	4.198	4.177
2023-09-07-19-33-us-ca	2.917	2.885	2.813
2023-09-07-22-47-us-ca-routebc2	3.088	2.884	2.736
Average (excluding 2022-01-11-18-48-us-ca-mtv-n):	3.618	3.495	3.385

Failure Case Analysis

Initial DGNSS positioning contain a large offset.

AI-based stop detection freezes the position at starting and prevents Kalman filter from adjusting the positioning.



Without stop detection, the Kalman filter gradually corrects the position using continuous GNSS velocity updates.

Conclusion

This study I build up a simple positioning system for smartphones by myself, shows my exploration and learning in the GNSS field.

- On the GNSS side, DGNSS, remove suspect satellites, and combine TDCP and doppler velocity estimation to get position and velocity. Kalman filter achieves GNSS position-velocity integration.
- On the IMU side, synchronization and resampling, Butterworth low-pass filter remove noise. At first, I use a Threshold-Based Stop detection. After, an AI model CompressedTCN helps make the recall more reliable.

Tests on real data show that these methods together can reduce positioning errors to 2-4 meters.

Limitations

1. Sensor Bias
2. Limited Dataset Scope
3. Kalman Filter Architecture
4. Lack of IMU data integrate in Model
5. Assumption of Spherical Earth
6. Positive Stop Detection

Future Work

1. Modeling sensor IMU bias
2. Using more different datasets
3. improving the of AI models through domain adaptation and semi-supervised learning
4. exploring EKF and FGO integration methods
5. implementing IMU GNSS integration
6. replacing hard stop constrained with soft unconstrained baseline estimates

Thank You!