AndroCon: An Android Phone-based Sensor for Ambient, Human Activity and Layout Sensing using Fine-Grained GPS Information

SOHAM NAG, Center for Excellence in Cyber Systems and Information Assurance, Indian Institute of Technology Delhi, India

SMRUTI R. SARANGI, Computer Science and Engineering, Indian Institute of Technology Delhi, India

Ambient sensing, human activity recognition and indoor floor mapping are three classical context detection problems that hackers try to crack using a variety of sources of side information. Other than overt signals such as microphones and cameras, covert channels such as WiFi, Bluetooth, assisted and coarse-grained Global Positioning System (GPS) signals have been exploited to date. Fine-grained or precise GPS information that is ubiquitously available to almost all Android devices as of today, has not been studied as a side channel for these problems. We present a novel set of attacks that exploit the 9 parameters that precise GPS provides to infer such contexts with an accuracy that often exceeds 99%. Specifically, this work reports a longitudinal study conducted over a period of more than a year in a 40,000 sq. km geographical region, where data was passively collected in all kinds of settings including open stadia, underground metros, flights, cruise ships, office spaces, dormitories and high-altitude locations. We propose a novel method, *AndroCon*, that combines linear discriminant analysis, unscented Kalman filtering, gradient boosting and random forest based learning to create a highly accurate context sensor. To the best of our knowledge, this study provides the most comprehensive characterization of all the information that we get from precise GPS signals, finds the parameters of interest, and uses them effectively to match the quality of results of other methods that rely on stronger channels such as WiFi. For instance, *AndroCon* can detect motions such as hand waving in front of the phone, which is known to be a very challenging use case.

CCS Concepts: • **Security and privacy → Web application security**; *Domain-specific security and privacy architectures*; Pseudonymity, anonymity and untraceability.

Additional Key Words and Phrases: Precise GPS, Floor Mapping, Ambient Sensing, Human Activity Recognition, Covert Channels

ACM Reference Format:
Soham Nag and Smruti R. Sarangi. 2025. AndroCon: An Android Phone-based Sensor for Ambient, Human Activity and Layout Sensing using Fine-Grained GPS Information. 1, 1 (October 2025), 37 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

A user's ambience and activities are a vital part of their personal space (defined as their context), and should remain *private*, unless they decide to selectively disclose them. Sometimes this information can leak out as an unanticipated consequence of doing some other activity like monitoring the strength of WiFi signals [24, 31, 45, 58]. Based on the leaked signal's characteristics, a malicious app can infer the user's immediate environment or their activity, collectively referred to as their *context*. Similar information leaks have been shown to happen with mobile phone signals[46],

Authors' addresses: Soham Nag, Center for Excellence in Cyber Systems and Information Assurance, Indian Institute of Technology Delhi, New Delhi, India, sohamnag97@gmail.com; Smruti R. Sarangi, Computer Science and Engineering, Indian Institute of Technology Delhi, New Delhi, India, srsarangi@cse.iitd.ac.in.

Manuscript submitted to ACM

Bluetooth[26], coarse-grained GPS data that just provide location information[83], acoustic information [43], ultrasound waves[42] mm waves [74] and even smart garments [44]. Additionally, optical wireless communication (OWC) methods, including Light Fidelity (Li-Fi)[39] and Visible Light Communication (VLC)[40, 41] have also been used for retrieving similar forms of context. In this paper, we shall confine our attention to the information obtained from ubiquitously available GPS signals (multiple satellites) and show their potential for revealing all kinds of context-related private information.

There are two kinds of GPS information that are provided by Android phones[1] (as of version 14): coarse (accuracy at the level of city blocks) and fine (accuracy within meters). The latter is more accurate and is primarily used for providing location-based services [19, 49, 73, 76]. This includes navigation, local search, traffic alerts, home delivery and ride sharing. The market for such applications is currently 50 billion USD and is anticipated to soar to 400 billion USD by 2030 (CAGR of 24.6%) [33, 57].

An app with fine location permissions can be compromised *unbeknownst to the user*. It can obviously misuse location information [73]. However, it turns out that it **can do much more** because modern GPS chips typically provide 32 parameters (per satellite) to applications such that they can fine tune their location estimates. To the best of our knowledge, this is among the first works that conclusively show that these parameters such as the Doppler shift, uncertainty figures or signal-to-noise ratio can be used to create a strong information-leaking channel.

We can, for instance, find out whether the user is in a closed space or open space, is the place crowded, is the user sitting, sleeping or standing, moving quickly or slowly, underground or overground, inside a flight, close to a lift or close to a staircase, or even just moving their hand in front of the phone's camera (refer to Section 6 for the full list). There is no need to take the help of any other kind of sensor such as the camera, accelerometer, microphone or even consider the GPS coordinates. Our claim is that semi-processed GPS data (collected in pure receive-only mode) is sufficient. In this paper, we shall advance various arguments to convince the reader that this is indeed possible. We shall make theoretical arguments, show the results of simulations, show real measurement data using GPS sensors and finally show extensive evaluation results using multiple phones collected over a period of 1 year, across a 40,000 sq. km geographical region.

Let us try to understand why work in this space is so sparse even though this information has been available since the release of Android 7 (August 2016). The only genre of work that we could locate is in the area of correcting the location information [71]. This is anyway the original purpose of making this data available. It turns out that the computed discrepancy between the location estimate before and after processing the additional parameters can be used to make some broad inferences [18, 75, 83]: indoor, outdoor, semi-indoor and semi-outdoor. The accuracy for even detecting such simple scenarios is limited to about 91% (in prior work), whereas our accuracies hover around 99% for far more complex scenarios. Early approaches primarily rely on encoder-decoder networks. We, on the other hand, show that we can use far simpler ML algorithms, which will also allow them to run surreptitiously. There are a few other problems in such related work: reliance on too many features, limited training data and reduced noise tolerance.

We perform detailed correlation studies, semi-informal analysis based on EM-wave physics and narrow down the list of features to 9 (out of 32). Next, we employ a nonlinear unscented Kalman filter for removing noise, which turned out to be a crucial design decision in hindsight. It preserves the essential signal variations that our algorithm needs [32]. Subsequently, we apply Linear Discriminant Analysis (LDA) for feature reduction and identify linear combinations of semi-processed parameters that effectively distinguish between different ambient contexts. The resulting data is then fed into a simple, explainable ML model for context detection. Additionally, we detect indoor layouts (not just positions) using this framework along with considering GPS coordinates.

This paper makes the following contributions:

❶ To the best of our knowledge, this is among the first studies that characterize in-depth the potential of semi-processed GPS signals for ambient sensing, barring a few works that use this information to *correct* the GPS coordinates and make simple inferences based on the computed discrepancies.

❷ We show how apps using GPS sensing can covertly capture semi-processed GPS data without consent and utilize it to discern the user's ambience (99.6%) and activities (87%) across *diverse* settings, effectively jeopardizing their privacy. At the moment, this vulnerability affects *over 90%* of Android users.

❸ We propose effective solutions to the three main impediments to realizing high accuracies in all kinds of these tasks: feature selection, noise removal and underfitting due to limited training data.

❹ We infer indoor floor maps (error margin of 4 meters), using semi-processed GPS data and user trajectories, without needing access to other ambient information leaking sources such as cameras.

The organization of this paper is as follows: section 2 provides the necessary background information. In section 3, we discuss the semi-processed GPS parameters that we use and their effectiveness in classifying activities and environments followed by characterization and validation of semi-processed data readings in section 4. The architecture of the model is discussed in section 5 and its performance is evaluated in section 6. The overview and evaluation of the layout estimation algorithm are presented in section 7. We finally review the relevant literature in section 9 and conclude in section 10. Some additional results can be found in Appendix A; however, reading it is optional. The paper is self-contained.

2 BACKGROUND

2.1 Overview of GPS

The GPS was established by the US Department of Defense in 1973. It is a member of the Global Navigation Satellite System (GNSS) constellation. GPS, in 2024, has 31 satellites that orbit the Earth in six planes of rotation. All of them are in the medium earth orbit (20,000 kms above sea level [10]). These satellites transmit navigation signals, which contain precise information about their position, velocity and current time (PVT).

Let us explain the physics. Every GPS satellite has an atomic clock that very accurately maintains time. The uncertainty is less than 1 part in 10^{16}. There are delays induced due to the ionosphere, signal interference, multipath effects, etc. Hence, modern receivers need to apply many different correction factors. There is thus a need to also transmit and compute many other parameters (described in Section 3) as well such that receivers can correct the received data. These include the pseudorange along with the SNR, phase shifts, Doppler shifts and details about the satellite and the constellation (see Table 1). A GPS message also contains other parameters about the receiver's clock (other than the time itself). There are four unknowns here: the clock skew (between the receiver device and the satellite), x, y and z coordinates of the GPS device. Hence, we need four equations, and as a result we need data from at least 4 satellites. In practice, we need data from many more satellites for accuracy enhancement.

2.1.1 GPS Data Processing Pipeline. Figure 1 shows the GPS processing pipeline at the receiver. Code and carrier tracking loops monitor all incoming GPS signals, demodulate and identify them. They identify the satellite for a given signal, which aids in calculating the *pseudorange* – apparent distance between the satellite and the receiver [70].

Modern GPS chipsets allow an application to tap information at any point in the GPS processing pipeline. In general, applications do not tap the information available at the intermediate stages because they don't find it useful. We decided to tap 9 of these *semi-processed* parameters (refer to Table 1, reasons mentioned in Section 3.2). We broadly use five types of parameters that can be classified into the following buckets: received signal power, carrier phase shift, multipath interference, signal-to-noise ratio (SNR) and the Doppler shift (due to motion). These parameters can be used to extract

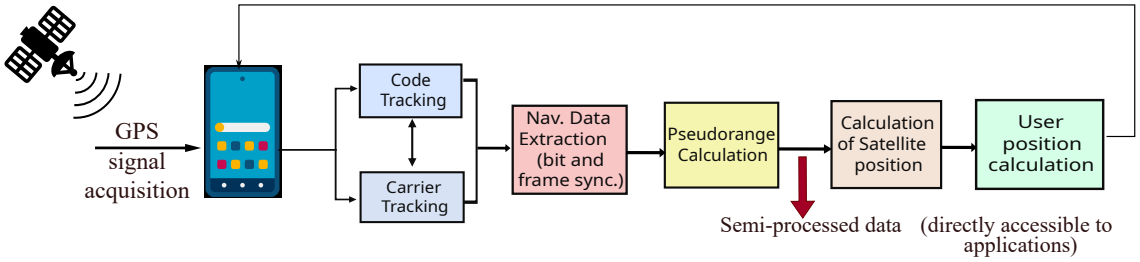


Fig. 1. GPS pipeline: From signal acquisition to position calculation

information about the surroundings because they are influenced by it. For instance, if there are a lot of objects around the receiver, the multi-path interference and SNR will be high.

Table 1. List of semi-processed GPS signal parameters

Field	Notations	Description	Unit
PseudorangeRate	PR	Pseudorange rate at the associated timestamp.	m/s
PseudorangeRateUncertainty	PRU	Pseudorange's rate uncertainty (1-σ).	m/s
ReceivedSvTimeUncertainty	RecSvTU	Error estimate (1-σ) for the received GNSS time.	ns
AccumulatedDeltaRange	ADRng	Accumulated delta range since the last channel reset.	m
AccumulatedDeltaRangeUncertainty	ADRngU	Uncertainty of the accumulated delta range (1-σ).	m
CN0	CN0	Carrier-to-noise density in the range [0,63].	dB-Hz
BasebandCn0DbHz	BbCN0	Baseband carrier-to-noise density (added in API level 30).	dB-Hz
AgcDb	Agc	Incoming signal power.	dB
State	State	Integer representing satellite sync state, with each bit indicating a specific measurement status.	-

2.1.2 Location Accuracy Permissions in Android. Android has *coarse* and *fine* location access permissions. Starting from Android 12 (released: Oct' 2021), users have the freedom to choose. However, the trade-offs are quite stark, and the choice is quite clear. The former relies on assisted GPS (cellphone towers or WiFi networks) and its inaccuracies can be as high as 50-100m. The latter uses satellites to provide an accuracy [20] of up to a few meters.

2.2 Unscented Kalman Filter

The Unscented Kalman Filter (UKF) [32] is an advanced unsupervised technique used for state estimation and noise filtering in *nonlinear* systems [35]. It predicts the future state of a nonlinear system and updates this prediction continually using incoming "noisy" measurements. Unlike the traditional Kalman Filter, which assumes linearity, UKF employs a deterministic sampling approach to generate sample points around the mean, which are then passed as inputs to nonlinear functions. This method accurately captures the mean and covariance, enhancing its effectiveness in complex applications. The UKF is widely used in fields such as robotics and navigation, where precise state estimation in the presence of noise is crucial. To the best of our knowledge, the application of the UKF to this type of precise GPS data for context sensing is a novel approach.

Consider a nonlinear state-space model at time t:

$$\begin{aligned} x_t &= f(x_{t-1}) + w_t \\ z_t &= h(x_t) + v_t \end{aligned} \tag{1}$$

x_t is the state vector and z_t is the corresponding noisy measurement at time t; $f(\cdot)$ is the state transition function; $h(\cdot)$ is the observation function; $w_t \sim \mathcal{N}(0, Q)$ is the Gaussian noise representing the uncertainty in the model's state transition with covariance Q; $v_t \sim \mathcal{N}(0, R)$ is the measurement noise representing the uncertainty in the measurements with covariance R. The Unscented transform approximates the state distribution using a set of sigma points. These points are passed as inputs to the the nonlinear functions f and h (resp.).

❶ *Sigma Point Generation*

Sigma points $\chi_{t-1}^{(i)}$ are generated around the mean $\hat{x}_{t-1}$:

$$\begin{aligned} \chi_{t-1}^{(0)} &= \hat{x}_{t-1} \\ \chi_{t-1}^{(i)} &= \hat{x}_{t-1} + \sqrt{(L+\lambda)P_{t-1}}, \quad i = 1, \ldots, L \\ \chi_{t-1}^{(i+L)} &= \hat{x}_{t-1} - \sqrt{(L+\lambda)P_{t-1}}, \quad i = 1, \ldots, L \end{aligned} \tag{2}$$

P_{t-1} is the error covariance matrix; L is the dimension of the state vector, and λ is a scaling parameter describing the spread of the sigma points around $\hat{x}_{t-1}$. Note that in the first case, we are adding a scaled version of the error, and in the second case we are subtracting it. We set $\lambda = 10^{-6}L - L$. L is approximately 70,000 (varies per iteration).

❷ *Prediction Step*

The sigma points are next passed to the state transition function to generate a new set of samples.

$$\chi_{t|t-1}^{(i)} = f(\chi_{t-1}^{(i)}) \quad (i = 0, 1, 2, \ldots, 2L) \tag{3}$$

The predicted mean and covariance are computed as:

$$\hat{x}_{t|t-1} = \sum_{i=0}^{2L} W_m^{(i)} \chi_{t|t-1}^{(i)} \tag{4}$$

$$\begin{aligned} P_{t|t-1} &= \sum_{i=0}^{2L} W_c^{(i)} \left[\chi_{t|t-1}^{(i)} - \hat{x}_{t|t-1}\right] \times \left[\chi_{t|t-1}^{(i)} - \hat{x}_{t|t-1}\right]^T + Q \\ &\begin{cases} W_m^{(i)} = 1 - \frac{1}{\lambda^2} & i = 0, \ldots, 2L \\ W_c^{(i)} = \frac{1}{2L\lambda^2} & i = 0, \ldots, 2L \end{cases} \end{aligned} \tag{5}$$

$W_m^{(i)}$ and $W_c^{(i)}$ are the weights for the mean and covariance, respectively.

❸ *Update Step*

The predicted measurements are obtained by passing the sigma points to the observation function (h).

$$\gamma_{t|t-1}^{(i)} = h(\chi_{t|t-1}^{(i)}) \tag{6}$$

The predicted measurement's mean and covariance are as follows:

$$\hat{z}_{t|t-1} = \sum_{i=0}^{2L} W_m^{(i)} \gamma_{t|t-1}^{(i)}$$
$$S_t = \sum_{i=0}^{2L} W_c^{(i)} \left[\gamma_{t|t-1}^{(i)} - \hat{z}_{t|t-1}\right] \left[\gamma_{t|t-1}^{(i)} - \hat{z}_{k|t-1}\right]^T + R \tag{7}$$
$$C_t = \sum_{i=0}^{2L} W_c^{(i)} \left[\chi_{t|t-1}^{(i)} - \hat{x}_{t|t-1}\right] \left[\gamma_{t|t-1}^{(i)} - \hat{z}_{t|t-1}\right]^T$$

C_t is the cross-covariance matrix that represents the covariance between the state and the sigma points. S_t is the residual covariance matrix that includes the measurement noise covariance. The Kalman gain K_t and the updated state and covariance are as follows:

$$K_t = C_t S_t^{-1}$$
$$\hat{x}_t = \hat{x}_{t|t-1} + K_t(z_t - \hat{z}_{t|t-1}) \tag{8}$$
$$P_t = P_{t|t-1} - K_t S_t K_t^T$$

We set all the elements in the matrices P, Q and R to 0.1. We keep repeating this process from the sigma point generation step onwards for the subsequent inputs until all the data is processed.

2.3 Linear Discriminant Analysis (LDA)

Any kind of learning that involves a lot of parameters tends to be inefficient unless we have a large number of training examples. Hence, dimensionality reduction needs to be done. We use LDA, which is a widely used supervised-learning technique for dimensionality reduction in pattern-classification applications [59, 64, 65]. It projects a dataset into a lower-dimensional space that enhances *class separability* through a linear combination of features. This is done to address the curse of dimensionality and reduce computational costs [68]. Unlike a cognate technique, Principal Component Analysis (PCA), which identifies the directions of maximum variance, LDA creates a transformation that maximizes the separation between different classes – maximize the in-class variance and minimize the cross-class variance. The linear transformation matrix used here is created by solving an eigenvalue problem. We have 9 classes. We use the *lsqr* solver with auto shrinkage and a $toI = 10^{-5}$.

3 THE VALUE OF SEMI-PROCESSED GPS DATA

In this section, we begin by examining the way semi-processed GPS data is retrieved on an Android platform. Subsequently, we explain all the 9 GPS parameters we consider, and finally explain how ambient sensing can be achieved.

3.1 Accessing Semi-Processed GPS Measurements

Prior to Android 7, developers accessed location details via the `android.gsm.location` API, which provided basic satellite information like C/N0 (carrier-to-noise ratio), azimuth and elevation, along with the fundamental National Marine Electronics Association (NMEA) sentences containing the PVT information [77]. Multipath effects and signal interference greatly degrade the positioning accuracy by altering the computed distance between satellites and the user. GPS receivers attempt to correct these errors, but since these measurements aren't directly accessible to users, they rely on often inadequate receiver-embedded correction models [22]. To foster the development of improved correction models for higher accuracy, Google made the raw and semi-processed GPS data public on Android phones. This information

comprises the semi-processed GPS data, which is available to developers via the `android.location.GnssMeasurements` API within the `android.location` package (starting from Android 7 [48]).

3.2 Semi-Processed GPS Parameters

Please keep referring to Table 1 while reading this section. We specifically choose 9 out of the 32 semi-processed GPS parameters because these parameters showed the least variance across phones (same time, same location). The rest either produced very volatile and fluctuating values that were heavily dependent on the receiver's clock's characteristics or were empty fields.

❶ ***Doppler Shift*** :

The Doppler shift (PR) indicates a change in frequency due to the relative motion between the signal source and the receiver. Given that the satellites are moving and the receiver itself may be non-stationary, a Doppler shift is expected. A positive shift indicates that the satellite is moving towards the receiver (increasing the perceived signal frequency), and vice versa.

Multipath phenomena, where signals reflect off surfaces before reaching the receiver affect Doppler shifts. In open areas with minimal multipath effects, the PR measurements remain stable, while in crowded or indoor areas, they can fluctuate significantly due to multipath effects.

Similarly, the parameter PRU quantifies the uncertainty in PR measurements, which varies with environmental conditions. In crowded or mobile environments, multipath effects and irregular Doppler measurements increase this uncertainty, resulting in high PRU values. In open areas with minimal multipath effects, the uncertainty is lower, leading to low PRU values. This analysis underscores the utility of PR and PRU in inferring both the state of motion and the environmental context of individuals based on semi-processed GPS measurements.

❷ ***Carrier Phase***:

The parameter ADRng quantifies the accumulated error in the distance between the satellite and the receiver and is calculated by measuring the phase shift (carrier phase). It mainly captures minor differences in the signal delay with respect to historical values.

In open areas with minimal multipath interference, ADRng values are stable. ADRng and its associated uncertainty (ADRngU) can detect subtle movements that are associated with the number of people in the room or a hand moving in front of the mobile device.

❸ ***Signal-to-Noise Ratio***: CN0, or Carrier-to-Noise (C/N) density in dB-Hz, is a crucial metric for evaluating the GNSS signal quality. It typically ranges from 10 to 50 dB-Hz with a potential spectrum extending from 0 to 63 dB-Hz in exceptional circumstances [6]. Variances in CN0 allow differentiation between the following environmental conditions.

In congested or enclosed areas with signal interference, CN0 levels decrease, indicating heightened noise in the GPS signal and suggesting the individual's presence in such environments. Conversely, open-sky scenarios with minimal interference increase CN0 levels. Similarly, BbCN0 represents the C/N of the signal at the baseband, obtained after demodulating the GPS signal received by the antenna. This value is typically slightly lower than the C/N measured at the antenna port (CN0) [6]. Our experiments demonstrated a strong correlation (=0.98) between CN0 and BbCN0, indicating similar behavior in identifying the user's context.

❹ ***Received Signal Strength (RSS)***: The Agc (RSS) refers to the power of the incoming signal. Negative or low Agc values indicate potential interference or presence of large obstacles [6]. We shall see that the value of Agc in combination with other parameters helps differentiate between user activity – motion/rest – because the signal strength at rest is

slightly higher. Agc values can distinguish between different ambient conditions such as crowded urban areas or open spaces that are near mobile towers.

❺ ***Multipath Interference***: The *State* field indicates the current synchronization state for each satellite signal. It can assume a value that is either 0 or a combination of different synchronization states [6]. When the *State* is STATE_MSEC_AMBIGUOUS (value: 16), it implies millisecond-level ambiguity in the GPS measurement's tracking state due to multipath effects [6] or obstacles/movement in the immediate vicinity.

RecSvTU represents the error estimate (1-σ) of the received GPS time of a particular *SV* (satellite vehicle) – it is typically influenced by different sources of interference. As it quantifies uncertainty, its effectiveness in distinguishing user activity and the immediate environment is similar to PRU And ADRngU.

4 STUDY OF SEMI-PROCESSED GPS SIGNALS' CHARACTERISTICS

This section aims to establish a correlation between quintessential EM wave parameters, such as the RSS and Doppler shift, and the corresponding semi-processed GPS parameters. Subsequently, we examine the correlation among semi-processed parameters to reinforce the idea that they can be *fused* to achieve improved classification outcomes.

4.1 Real-World Data Collection

Semi-processed GPS data was logged using the GnssLogger [5] Android application on six different Android phones (refer to Table 2). Note that phones that use the Samsung Snapdragon chipset do not allow the user to retrieve the ADRng measurements [8]. We deliberately used two phones with the same version (Redmi Note 9 Pro Max) to characterize variations across two phones of the same model. There was some variation even in this case. Therefore, noise filtering is required to ensure that our technique works across phones in a robust manner.

Table 2. Details of the phones

Model Name	Android version	Chipset	Year
Redmi Note 9 Pro Max	Android 11	*Snapdragon 720G*	2020
Redmi Note K20 Pro	Android 12	*Snapdragon 855*	2020
Redmi Note 9 Pro Max	Android 11	*Snapdragon 720G*	2021
OnePlus Nord CE2	Android 13	*MediaTek Dimensity 900*	2022
Galaxy A54	Android 14	*Exynos 1380*	2023
Nothing Phone 3	Android 15	*Snapdragon 8 Gen 3*	2025

4.2 Datasets: Synthetic (Kaggle) and Real-World

Along with the data that we collect using these phones, we use open-source datasets as well: a Raw GNSS dataset [67] and the Kaggle GNSS dataset [27]. This is needed to ensure that similar effects are also being seen in those datasets and our noise filtering techniques work. The Raw GNSS dataset includes 40 traces collected using the Xiaomi Mi8 and Google Pixel 5 smartphones, with each trace comprising approximately ten minutes of raw GNSS data per point, recorded consistently over multiple days across open, semi-open, and covered environments. For this study, we have only considered the Xiaomi Mi8 data [66], as it was logged using the GNSSLogger application, which aligns with our data collection framework and captures all required parameters (see Table 1). On the other hand, the Google Pixel 5 data, logged in the RINEX [34] format, lacked the necessary raw parameters and was therefore excluded. The Kaggle

GNSS dataset consists of 39 traces collected using the Pixel 4, Pixel 4 XL and Xiaomi Mi8 phones (resp.). This dataset primarily contains measurements collected during motion (driving).

The open-source datasets, while valuable, were limited in their representation of diverse user activities and environmental conditions. To simulate diverse user activities and environments, we generated supplementary data at different sites using our 6 phones. The settings include an open ground, flights, indoors, inside a metro, in a stadium (crowded and empty), while standing, sitting, lying down and waving a hand in front of the mobile device. The data collection process lasted approximately 45-150 minutes at each site on an average.

The ionosphere's electron density variations cause fluctuations in GPS signals [21], resulting in measurement variability. Furthermore, diverse weather conditions can introduce additional fluctuations. Additionally, there are GPS hardware-specific variations as well along with variations between two phones of the same age and model (same time, same location). To simulate real-world settings and mitigate these effects, semi-processed data was collected three times over the course of a year under various atmospheric conditions and at different times of the day. We tried to minimize the experimental noise by ensuring that the ambient setting/activity was the same. Of course, there were variations due to the weather, position of satellites, etc. We did not try to achieve consistency in these parameters because it is very hard to do so.

4.3 Calibration with Real Sensors: Signal Power

Researchers have primarily used the RSS, the Doppler shift and SNR for designing human activity recognition (HAR) and ambient sensing systems for WiFi, Bluetooth and cellular tower signals [28, 54, 60]. We, of course, use many more parameters.

However, we stick to these three parameters for the purpose of characterization and calibration of our setup. We map these parameters to the corresponding semi-processed GPS parameters. The GPS signal inherently captures the Agc (RSS or signal power). We used an RF Explorer Spectrum Analyzer (Model B34J7ML7J58J9MD6) [9], which records the signal RSS. We set the RF Explorer to the L1 GPS band frequency range (1575.42 MHz) [2]. We logged the GPS's RSS using the measurement device (RF Explorer) and simultaneously, we logged the Agc. We found a roughly linear relationship between them (refer to Figure 2).

4.4 Correlation Results across the GPS Parameters: Feature Selection and LDA

Next, we consider all our data, i.e., our traces and the open-sourced GNSS traces. The trends are roughly similar. We plot the correlation for all pairs of GPS signals (results shown in Figure 3). CN0 exhibits a strong correlation with BbCN0 (0.98). Hence, only one of them is required. We choose CN0. The correlation of CN0 with other signals is as follows: ADRngU (0.68), PRU (0.48), and RecSvTU (-0.53). PR shows a correlation of 0.63 and 0.54 with ADRng and RecSvTU, respectively. This means that we have a lot of pairs of parameters that have reasonably high correlations. This motivates the use of a feature reduction algorithm such as LDA.

5 DESIGN OF THE INFERENCE PIPELINE

We shall use the same ML pipeline to classify both the ambience as well as human activity. Figure 4 shows our proposed framework.

We first collect 8 features from each phone. Recall that we had discarded BbCN0. Then, we eliminate the nonlinear noise using UKF. Next, we use StandardScaler [12] to normalize the data given that different fields have different units and scales. Subsequently, we perform feature reduction using the LDA algorithm such that we can effectively classify

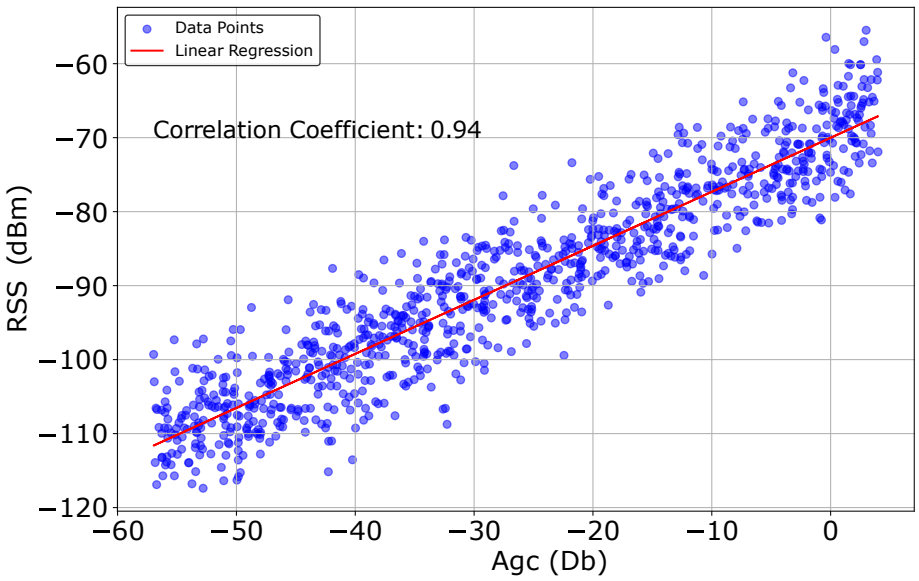


Fig. 2. Correlation between Agc and RSS

the features into C classes. Here, the output is a vector with $C - 1$ elements (dimensions reduced). Then we use an ML-based classifier to map the reduced set of features to a class label. In the subsequent evaluation section, we shall evaluate different types of ML models.

6 AMBIENT SENSING AND HUMAN ACTIVITY RECOGNITION

We group ambient sensing and HAR into one section because the techniques that are used are similar.

6.1 Evaluation Setup

6.1.1 ML Models. We evaluated six machine learning algorithms: Random Forest (RF), K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Decision Tree (DT), Naive Bayes (NB) and Gradient Boosting (GB). Given that we were getting a high accuracy, we did not use more sophisticated CNN-based models. These algorithms were selected based on their established efficacy in prior research [31, 69]. Additionally, their relatively lightweight nature allows for efficient execution on client devices while maintaining interpretability. To ensure the model's generalizability to new data and avoid overfitting, extensive empirical analyses were conducted to determine the optimal hyperparameters for each model using GridSearchCV [38] (refer to Table 3).

Notably, for SVM, we employed the one-versus-one (OVO) strategy to handle class imbalance, decomposing the multi-class classification problem into multiple binary classification tasks [17]. To mitigate overfitting, we utilized ten-fold cross-validation. The dataset was randomly split into 80% for training and 20% for testing. The data collection duration for classifying a setting ranged from 2 to 4 minutes, requiring a minimum of 1,000 samples.

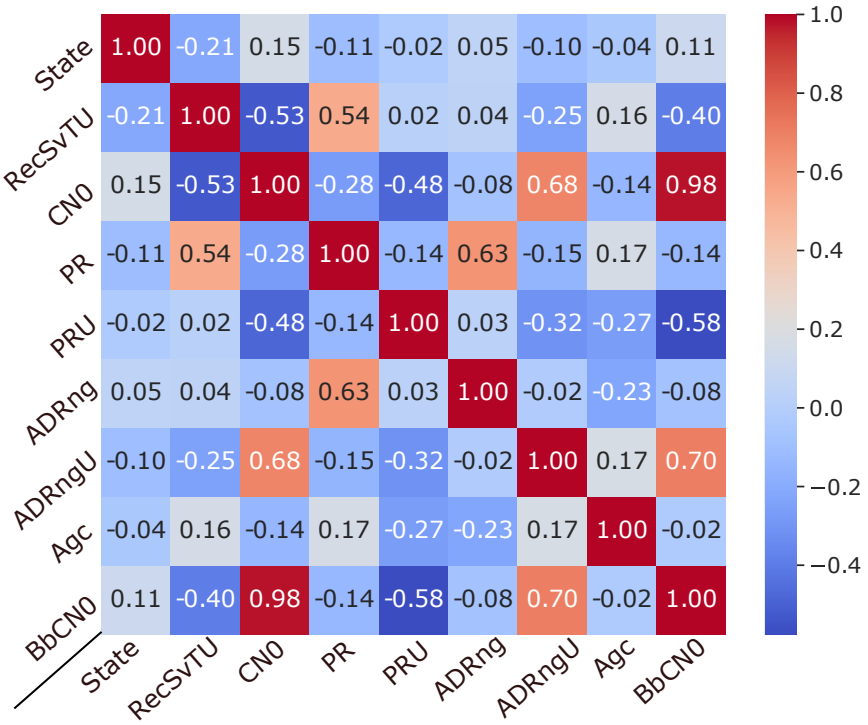


Fig. 3. Correlation of various semi-processed GPS parameters

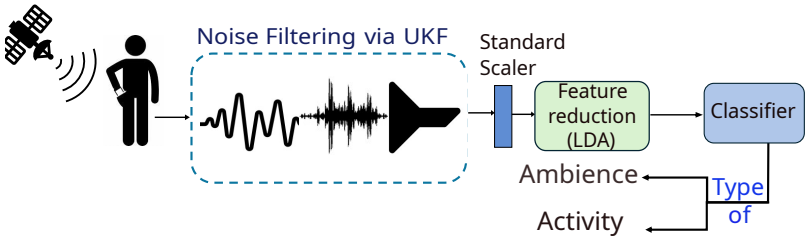


Fig. 4. Flow diagram of our ML pipeline

Table 3. Hyperparameters used for each classifier

Classifier	Hyperparameters
Random Forest (RF)	n_estimators: 100, max_depth: 10, min_samples_split: 10, min_samples_leaf: 4, bootstrap: True
Support Vector Machine (SVM)	C: 110, kernel: rbf, tol: 0.001, break_ties: True
K-Nearest Neighbors (KNN)	n_neighbors: 10, weights: distance, algorithm: auto, leaf_size: 30
Decision Tree (DT)	max_depth: 20, min_samples_split: 20
Naive Bayes (NB)	var_smoothing: 1e-9, fit_prior: True, class_prior: None
Gradient Boosting Classifier (GB)	n_estimators: 100, learning_rate: 0.01, max_depth: 10, min_samples_split: 10, min_samples_leaf: 4

6.1.2 Details of the Platform. The learning models were trained on the Google Collaboratory cloud platform (https://colab.research.google.com/). The cloud-based system was equipped with a dual-core Intel® Xeon® CPU running at 2.20GHz, 13GB of RAM, and a disk capacity of 107.72 GB. The hyperparameter tuning task was slow and time consuming. Hence, we used a NVIDIA® Tesla® T4 GPU equipped with 15GB of dedicated RAM.

6.2 Ambient Sensing

6.2.1 Data collection. For each ambient setting, over 100K samples were collected in various locations such as a dormitory floor, stadium, bustling market and underground metro tunnel. The data collection involved 20 volunteers who were graduate and undergraduate students studying in a university. They were aged between 19 and 28. The volunteers were first rigorously trained on how to take measurements. Furthermore, to ensure the generality of the dataset, the volunteers were requested to vary the settings (within limits). For example, one group put the mobile phones in their shirt pockets, some kept it in their pant pockets and some in their purses.

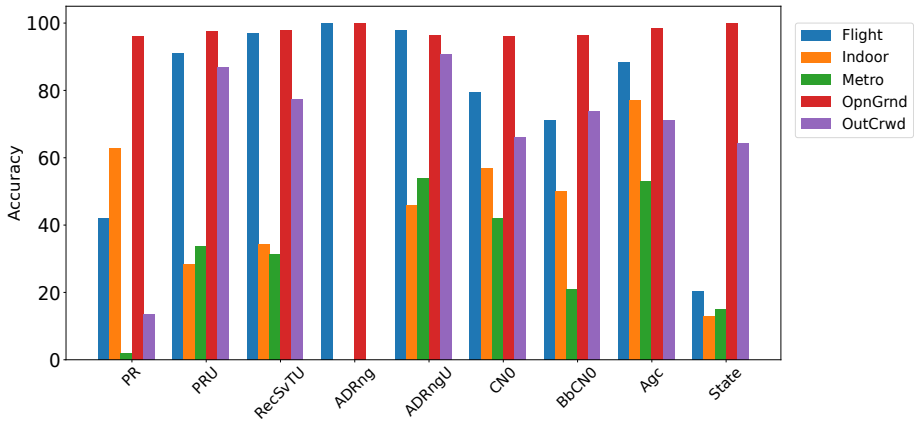


Fig. 5. Ambient sensing accuracy for each semi-processed GPS parameter

6.2.2 Class Labels. Five user environments were defined: flight[1], indoor, metro tunnel, open ground and outdoor crowded area. The objective of this work is to use GPS semi-processed data in isolation to classify the ambient. For the open space and outdoor crowded area settings, samples were collected in an open-air stadium. We considered two cases: empty and filled (with people). Additionally, we also included the open, semi-open, and covered environments from the open-source GNSS dataset [67], treating the open environment as analogous to our open ground setting. We will discuss the performance of our models on the open-source dataset separately in Section 6.2.7.

6.2.3 Results with Parameters Considered Individually. We evaluated the effectiveness of each of the semi-processed GPS parameters (after applying UKF) for characterizing the ambient environment (see Figure 5).

In the interest of space, only the most accurate results for each parameter (across all models) have been shown. Agc classifies the ambient with an average accuracy of 86.7% while ADRng performs the worst with an accuracy of 40.6%.

[1]All the signals were collected in passive, receive-only mode. There was no signal transmission. This is allowed as per rules.

All other parameters have an accuracy range between 57.9-82.1%. This accuracy is on the lower side because we are considering the parameters *individually*. We shall later combine the parameters to achieve much better results.

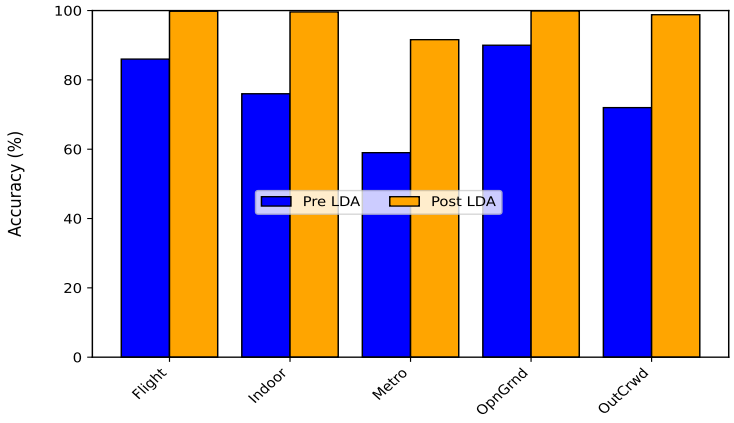


Fig. 6. Pre vs. Post LDA accuracy for ambient sensing

6.2.4 Impact of LDA on Accuracy. We conducted a comparative study to analyze the accuracy before and after applying LDA for ambient classification. While individual GPS parameters offer some discriminative ability, they also introduce noise and redundancy, leading to lower accuracy. LDA improves feature separability by projecting the parameters into a lower-dimensional space that maximizes class distinction. As shown in Figure 6, without LDA, the models struggled with stability, particularly in challenging environments such as metro tunnels. After applying LDA, accuracy increased to over 90% across all settings, with an average improvement of 22.14%, underscoring LDA's effectiveness in enhancing accuracy and its essential role in our pipeline.

6.2.5 Results with Combined Parameters. The accuracy of the models for each setting after all the steps (UKF + scaling + LDA) is shown in Table 4. Although SVM accurately predicted the results for three settings, it failed to classify the Indoor and Metro environments with misclassification rates of 57.6% and 79.3%, respectively. Similarly, KNN achieved over 90% accuracy for all the settings other than Metro, where it had a misclassification rate of 43.4%. The NB (Naive Bayes) model also underperformed for Metro with a misclassification rate of 32.4%. The Metro setting is hard to classify because several parameters such as ADRng and State are not available. In contrast, RF (Random Forest) and GB (Gradient Boosting) exhibited the best performance with accuracies exceeding 90% for all the settings. We will discuss the reasons for this trend in Section 7.4.

Table 4. Accuracy of ambient classifier models: GB and RF are the best. The results are averaged across all the phones and volunteers.

Classifier	Flight	Indoor	Metro	OpnGrnd	OutCrwd
GB	99.8	99.6	91.6	99.9	98.8
RF	99.5	99.3	88.0	99.8	98.8
DT	99.9	99.2	91.6	99.8	98.8
KNN	97.5	92.8	56.7	98.0	89.7
NB	79.1	89.7	67.6	97.6	90.2
SVM	96.4	42.4	20.7	97.9	84.9

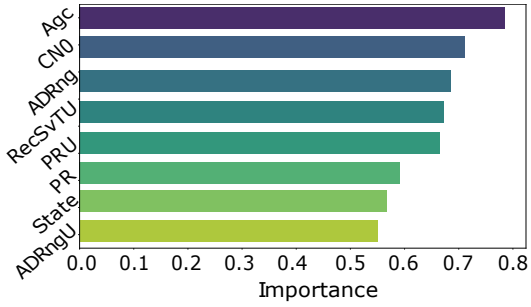
6.2.6 Characterization of Indoor Environments. Additionally, we further characterized the indoor environment based on population density. We consider two settings: nobody is present in a small (12 ft x 8 ft) room and two people are present. The GB model accurately classified both scenarios with an accuracy of 99.25%.

6.2.7 Evaluation on synthetic Dataset. We evaluated the efficacy of our models on the open-source GNSS dataset [67], encompassing open, semi-open, and covered environments, as shown in Table 5. These findings are consistent with those discussed in Section 6.2.5. The SVM exhibited the poorest performance, demonstrating inconsistency; it achieved 95.9% accuracy in open environments but faltered in semi-open and covered settings, with misclassification rates reaching as high as 51.3%. GB emerged as the top performer, attaining an average accuracy of 98.1%. Other classifiers, including KNN, DT, NB, and RF, also performed robustly, achieving accuracies above 93% with average accuracies ranging from 93.3% to 97.5%. These results underscore the capability of *AndroCon* to effectively classify diverse environments, affirming its robustness across varied scenarios.

Table 5. Accuracy of ambient classifier models on Open-Source GNSS Dataset

Classifier	Open	Semi-Open	Covered
GB	99.8	96.5	98.2
RF	99.6	95.2	97.8
DT	99.6	94.3	96.5
KNN	96.8	94.0	88.6
NB	96.5	86.1	87.4
SVM	95.9	48.7	52.3

Fig. 7. Feature importance test for ambient classification (GB model)



6.2.8 Relative Importance of Features. We further evaluated the importance of each feature. This involves randomly perturbing the values of each parameter and measuring the resulting increase in the model's prediction error (refer to [4] for the definition of the *importance* metric). A feature is deemed *important* if perturbing its values leads to a significant increase in the error, indicating that the model relied on that feature for accurate predictions. Agc has the highest importance score of 0.78, while all other features have importance scores ranging from 0.55 to 0.75 (see Figure 7), demonstrating their substantial contributions to classification.

6.3 Human Activity Recognition (HAR)

6.3.1 Class Labels. Prior work [11, 28, 51, 60, 78] has extensively explored various HAR systems. They cover both static activities (sitting, lying and standing) and dynamic activities (such as walking, running or traveling in a vehicle).

The latter can be discerned on the basis of the speed or velocity. The Android platform's `getSpeed` API [7] in the `android.location.Location` package facilitates this. In our case, we shall use Doppler shifts [7] (captured by the PR attribute) to distinguish dynamic activities, obtained from the Kaggle GNSS dataset [27] in terms of vehicular motion.

Distinguishing static activities cannot be done on the basis of Doppler shifts. We wish to classify static activities into four classes: sitting, standing, lying down and hand waving (mobile phone not held with the moving hand). In case of hand waving, the device was mounted on a wall stand and hand movements were performed in its close proximity (within 2m).

6.3.2 Results with Individual GPS Parameters. The results are consistent with previous observations (Figure 19 in the appendix). Agc achieves the highest accuracy of 77.2%, while PR performs the worst with a high misclassification rate of 59.8%. The accuracy for other parameters ranges from 52.5-71.5%. This reflects the inability of individual parameters in classifying activities. Hence, we combine the parameters to achieve better results.

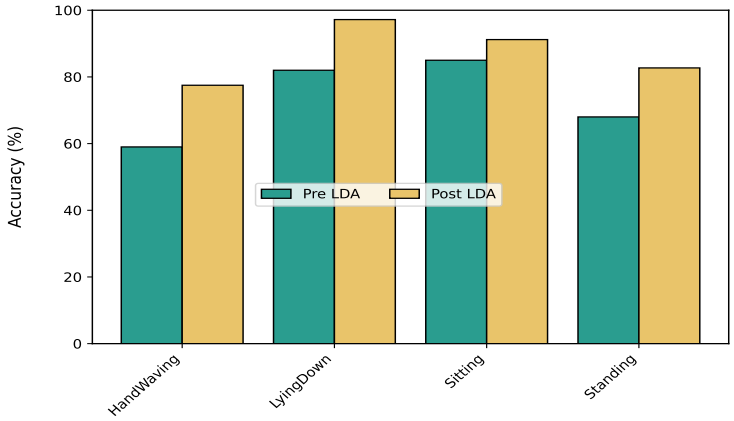


Fig. 8. Pre vs. Post LDA accuracy for HAR

6.3.3 Impact of LDA on Accuracy. Similar to ambient classification, we conducted an experiment to evaluate the efficacy of LDA in HAR. As shown in Figure 8, without LDA, the model exhibited low accuracies, particularly for activities such as hand waving, where accuracy dropped to as low as 59%. Other activities also showed comparable results. However, after applying LDA, we observed a significant improvement, with an average accuracy increase of approximately 14%, underscoring the importance of LDA in enhancing the performance of our system.

6.3.4 Results with Combined Parameters. Table 6 compares model performance using all the parameters. SVM achieves 93.4% accuracy for lying down but has a an accuracy of $\leq$ 60% for other activities. KNN accurately classifies hand waving with an accuracy of 88.4% but has an average accuracy of 74.4% for other activities. NB identifies the lying down posture with $\geq$ 80% accuracy but performs poorly (about 67% on average) for the other three activities. RF, DT, and GB are the best – they achieve a minimum accuracy of 80%. The peak accuracy is 97.2% (RF). Note that the use case of hand waving is particularly very difficult. Recently proposed sophisticated schemes that use array radars [63] achieve an accuracy in a similar range (80-88%).

Consequently, RF was selected as the model of choice for this activity.

Table 6. Accuracy of activity classifier models: RF is the best. The results are averaged across all the phones and volunteers.

Classifier	HandWaving	LyingDown	Sitting	Standing
RF	77.5	97.2	91.2	82.7
DT	80.7	84.9	85.1	87.9
GB	82.4	96.9	86.7	82.8
KNN	71.8	88.4	68.9	56.9
NB	61.4	80.7	77.8	63.2
SVM	60.3	93.4	38.3	44.8

6.3.5 Evaluation on synthetic Dataset. We assessed the performance of our models on the Kaggle GNSS dataset [27]: distinguishing vehicular motion from vehicle at rest, as shown in Table 7. RF delivered the highest performance, achieving an average accuracy of 95.1% across both activities, while SVM and NB showed the weakest results, with average accuracies of 68.6% and 69.0%, respectively. Other classifiers, including DT, GB, and KNN, exhibited strong performance, with average accuracies ranging from 75.0% to 93.7%. A significant accuracy gap is observed, with models classifying moving vehicle scenarios with higher accuracies (up to 98.5%) compared to vehicle at rest, where accuracies drop by approximately 8.8% on average, due to overlap with other static activities in the dataset, posing static activity classification challenges. Despite these difficulties, *AndroCon's* robust pipeline achieved accuracies exceeding 90%, outperforming results reported in related works.

Table 7. Accuracy of activity classifier models on the Kaggle GNSS Dataset: covering dynamic activities

Classifier	Moving Vehicle	Vehicle at Rest
RF	98.5	91.7
DT	97.3	90.1
GB	91.8	79.4
KNN	78.6	71.3
NB	70.2	67.8
SVM	71.4	65.7

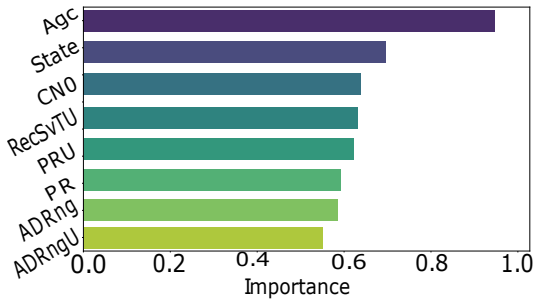


Fig. 9. Feature importance test for human activity recognition

6.3.6 Relative Importance of Features. Additionally, we evaluated the importance of each feature in HAR when they are fused together. The feature Agc has the highest importance score of 0.88, while all other features have importance scores in the range of 0.5 to 0.65, which means that they are also nonetheless quite important (see Figure 9). This further strengthens our claim – using parameters in combination improves accuracy.

6.4 Assessment of the Robustness

We conduct several experiments to demonstrate the robustness of our method, *AndroCon*. We consider different environmental conditions, variations in the satellite vehicle ID (SvID) density, different test:train splits, sparse fingerprinting techniques and accuracy variability across different phones.

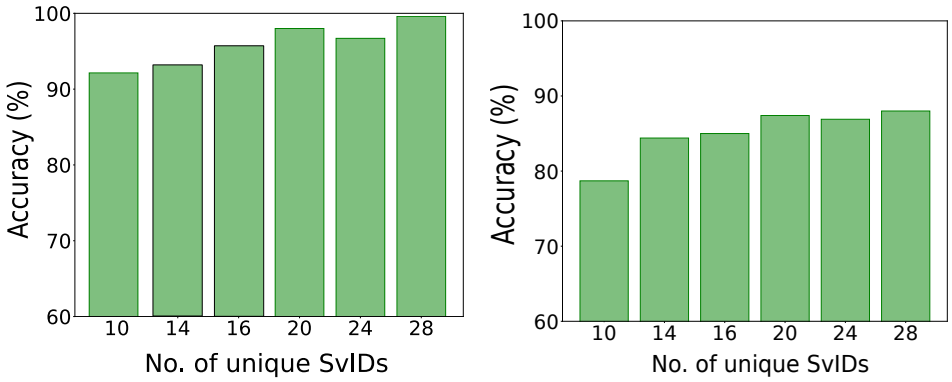


Fig. 10. Accuracy results: varying the number of SvIDs for **(a)** ambience classification and **(b)** activity classification

6.4.1 Variation in the Number of Unique SvIDs. Figure 10 shows the impact of varying the number of SvIDs on the prediction accuracy. A subset of SvIDs was randomly selected from the logged data and only semi-processed GPS data corresponding to these SvIDs was used for evaluation. Care was taken to maintain the integrity of the dataset by ensuring that SvIDs representing a significant portion of the logged data were not removed. The dataset was becoming really sparse in such cases.

The maximum number of unique SvIDs observed during the experiment was 28. The results show that even with a SvID density as low as 50% of the total number of unique SvIDs (14 unique SvIDs) an accuracy of 92.1% (10(a)) and 84.4% (10(b)) in ambient and activity classification, respectively, can be achieved.

6.4.2 Different Train:Test Splits. Figure 11 shows the accuracies when we vary the train:test ratio. For example, a ratio of 80:20 indicates that the training data is 80% of the overall dataset and the remaining 20% of the data is used for testing purposes.

Even when the proportion of test data is as high as 50%, the average accuracy of ambient classification is 94.2% (11(a)), and 85.4% (11(b)) for activity classification, respectively. We can attribute this high degree of resilience to the UKF's noise removal and feature preservation capabilities. **When we remove the UKF phase, the accuracy drops by more than 50%**.

6.4.3 Unseen Events. The accuracy of *AndroCon* was further assessed by experiments performed by absolutely untrained volunteers using another set of Android phones in different settings. It is important to note that up till now, we used a set of 5 phones and the same settings, even though we varied the following parameters: time of the year, time of the day and weather conditions. There was invariably some inherent noise. For example, in a crowded area, the composition of the crowd varied. Such factors cannot be controlled.

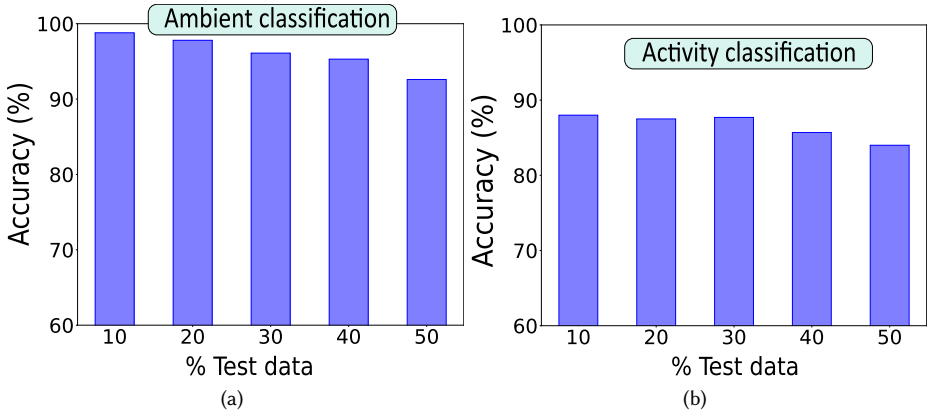


Fig. 11. Accuracy with different level of train:test ratio for **(a)** ambience classification and **(b)** activity classification

In this experiment, we vary everything else other than the inherent nature of the setting. For example, we collected measurements at locations that were roughly 1000 kms away from our original location. One location had an altitude of 3.5 kms. We also tested on a cruise ship, which we categorized as an open space.

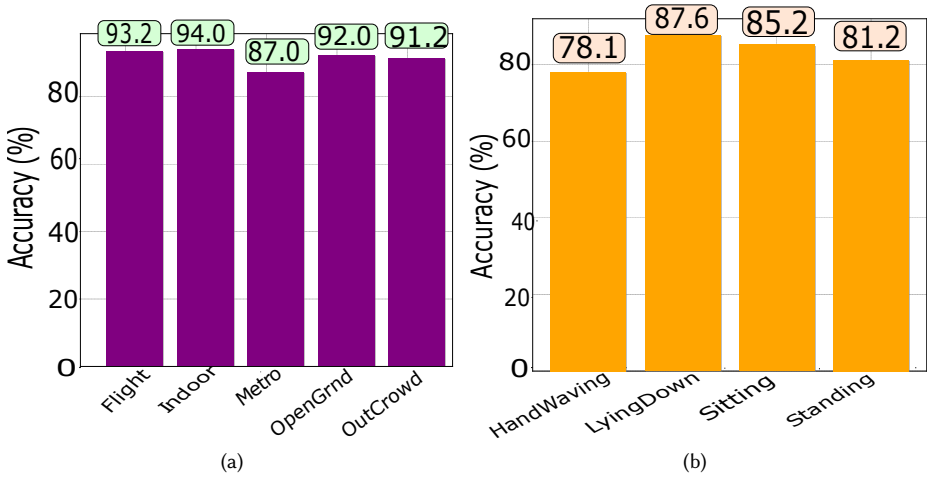


Fig. 12. Accuracy of events performed by untrained volunteers at new locations for **(a)** ambience classification and **(b)** activity classification

Figure 12 shows the results. While a decrease in accuracy was observed, it remained consistently above 91.5% 12(a) for ambient recognition and 83% 12(b) for activity classification, respectively. The best-performing technique was still GB for ambient classification and RF for human activity recognition. This reduction in precision was mainly due to variations in multipath effects and the significantly different experimental setups. Moreover, these volunteers did not go through our orientation process. There is a possibility that our orientation process may have sub-consciously created

some degree of homogeneity and bias, which we sought to remove by doing such experiments. Note that we tried our best to avoid such biases. With this fresh set of volunteers, it was not possible for such biases to creep in.

Table 8. Ambient sensing accuracy variability across mobile devices with and without UKF+LDA

Device Model (Year)	Raw Data		Post UKF+LDA Data	
	Accuracy (%)	Accuracy Variability (%)	Accuracy (%)	Accuracy Variability (%)
Redmi Note 9 Pro Max (2020)	58.7	±8.6	94.7	±3.8
Redmi Note K20 Pro (2020)	61.5	±9.0	95.2	±3.4
Redmi Note 9 Pro Max (2021)	66.3	±8.1	95.2	±4.0
OnePlus Nord CE2 (2022)	73.8	±7.0	98.3	±4.2
Galaxy A54 (2023)	82.2	±5.1	99.2	±3.0
Nothing Phone 3 (2025)	84.1	±4.8	99.4	±2.7

6.4.4 Variability across Different Phones. We studied the accuracy variability across various mobile devices under similar conditions, as presented in Table 8. In the absence of UKF+LDA processing (unfiltered), a statistically significant 8% accuracy difference was observed between two Redmi Note 9 Pro Max units (purchased in 2020 and 2021, respectively), with accuracies declining to as low as 59%. Under identical settings, a 9% variation emerged for the same model across different days, attributable to weather fluctuations, which escalated to 25% across distinct phone models (Redmi Note 9 Pro Max and Nothing Phone 3) at the same location. In our setup, the Nothing Phone 3 (2025) exhibited the highest accuracy, whereas older models demonstrated notably inferior performance. Post-processing with UKF+LDA markedly reduced these disparities: the maximum difference decreased to 3% for data collected simultaneously at the same location and to 4.2% across different days (up to months apart), with absolute accuracy reaching up to 99%. For brevity, we focus here on ambient sensing; similar trends were observed for HAR (see Table 15 in the appendix).

7 FINDING THE INDOOR FLOOR LAYOUT

Our goal is to create an indoor floor map using semi-processed GPS data and identify features such as rooms, lifts, staircases, etc. Prior work [80] has used WiFi RSSI (received signal strength indicator), user activities (measured via accelerometers or pressure sensors) in conjunction with graph optimization techniques to construct the indoor map. These approaches do not work for us given that the RSSI proves to be quite a feeble variable in the case of GPS signals. For WiFi, the source of the signal is nearby, hence, its strength variation carries much more information. However, for us the source is very far away and the variation observed is minimal. It is often affected by noise to a much greater degree. This had necessitated the need for using the UKF and LDA algorithms in the first place. We nevertheless found RSS (Agc) to be a good starting point for our study. We then build on it since it is clearly inadequate for our purpose if used exclusively.

7.1 Study of GPS RSS Patterns

To study GPS RSSI variations across different areas of a large indoor area, we conducted an experiment in a dormitory corridor. It measured 32.3 m × 25.6 m (refer to Figure 13(a)). Our methodology involved collecting raw GPS signals from 10 volunteers traversing the corridor with phones in their pockets. To ensure data authenticity, participants walked naturally without specific instructions, capturing realistic signal statistics.

Figure 13(b) depicts the averaged variation in the GPS AGC (RSS) parameter across the dormitory. The results highlight significant differences in RSS readings at various locations within the site, suggesting the potential to infer

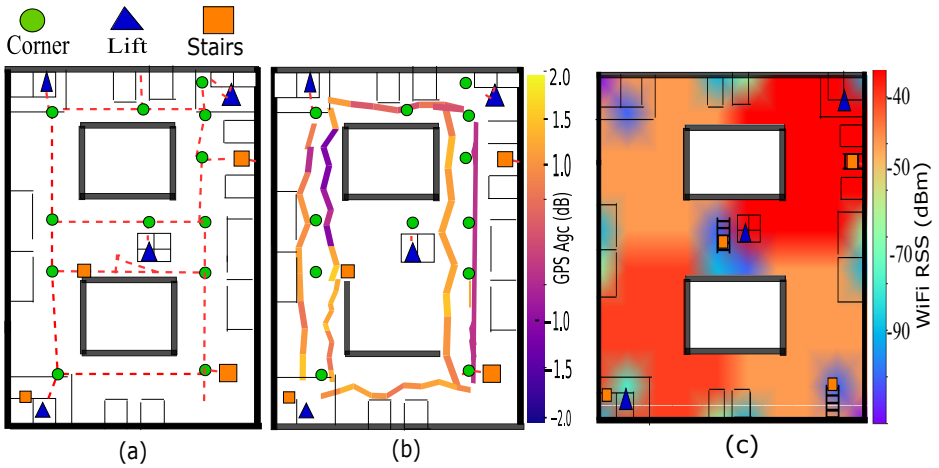


Fig. 13. (a) Layout of a dormitory floor (the site), (b) GPS RSS (Agc) signal variation. Major signal drop is observed near the lifts, (c) WiFi Signal strength heatmap

the building's layout. Notably, a consistent and substantial signal drop was observed near the lifts, attributed to electromagnetic shielding effects.

To further contextualize these findings and align with methodologies in related work [29, 30, 61, 80], we conducted a supplementary study. Since no GPS-based heatmap simulators are available and civilian GPS operates at a frequency close to 2.4 GHz WiFi, we used NetSpot, a commercial WiFi signal strength mapping tool, to approximate spatial variations in GPS RSS. We generated a heatmap illustrating the relative GPS RSS levels across the site (see Figure 13(c)), under the assumption of ideal, zero-noise conditions. To simulate a GPS environment, we positioned two access points outside the site, specifically in the open areas to the northeast and southwest. The differences in the RSS readings remains consistent with our previous observations. The data was scaled to align with real-world GPS signal intensity, which was measured and validated using the RF Explorer Spectrum Analyzer at key points of interest, including entrances, elevators, stairs, and corners. strong correlation of 0.83 was observed, reflecting the alignment between the modeled and actual signal intensity. It is important to note that NetSpot was not intended to directly replicate GPS signal characteristics but rather to model spatial variability in signal strength of a similar nature, enabling comparison with related studies within a controlled indoor environment. **Our study exclusively uses raw GPS signal values**.

The combined results from our empirical study confirm that GPS signal variations can effectively identify *points of interest*, such as elevators, rooms, and staircases. However, the data alone does not reveal specific features at these points of interest. Even with WiFi signals, accelerometer and pressure sensor data are required [80]. We are replacing it with semi-processed GPS data and the GPS location information (not used up till now).

7.2 Trajectory Creation Methodology: Use Location Data

7.2.1 Activity Landmarking. Our aim is to identify floor *landmarks* such as pathways, stairs, elevators, rooms and obstructions based on user activity like walking across the site and GPS-based location information. We will not use data from any other sensor.

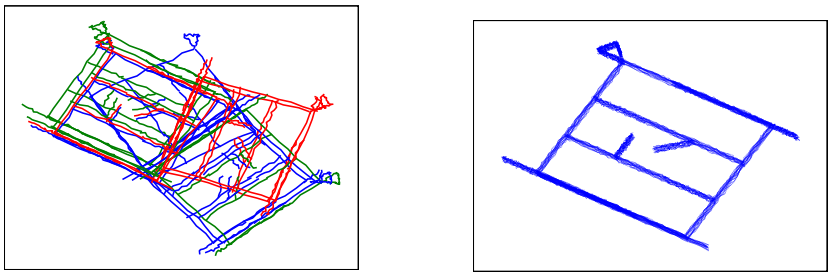
For example, users typically walk straight along pathways, turn when they encounter obstructions, and engage in different activities like climbing stairs or entering elevators. These activities can be captured using our HAR framework. Notably, elevators act as Faraday cages [3], and they block GPS signals. We shall rely on multiple users to collect this data – semi-processed GPS parameters, HAR fingerprints and trajectory information. Then we shall merge these results to **identify the landmarks**.

7.2.2 Trajectory Alignment. We collected the walking trajectories of our volunteers. It is necessary to employ a trajectory *alignment* algorithm (akin to [80]) because different people will produce different trajectories. Let us briefly describe the algorithm proposed by Zhou et al. [80]. Their algorithm involves using a transformation matrix, which it creates for adjusting the curvilinear paths (translation and rotation operations). This matrix is based on "activity landmarks" that are common to the different trajectories. Trajectory coordinates are classified as either activity landmark coordinates (ALC) or non-activity landmark coordinates (NALC). When merging trajectories, we focus on common ALCs and calculate their relative coordinates to determine the necessary translations and rotations. This iterative process ensures a more accurate and consistent overall map by aligning each new path to the existing map. The advantage of aligning trajectories is that we create a robust skeleton of the internal map. A single observation is prone to a lot of noise especially when using GPS signals for precise location.

After aligning the trajectories with themselves and with a virtual coordinate system, we apply graph optimization techniques (refer to [80]) to further refine the locations of the ALC and NALC trajectory points. We use the Levenberg-Marquardt algorithm [56] for optimization. The result is a map of the site with the ALC points (specifically) indicated.

7.3 Evaluation of the Accuracy of Classifying ALC Points

7.3.1 Data collection. 10 volunteers participated in the data logging process. The data was collected using the Gnss-Logger Android application, the same tool employed for ambience/HAR classification. Each volunteer carried a device with the GnssLogger app installed and traversed the floor, and sometimes used the services at the landmarks. To ensure robustness and mirror real-time scenarios, participants were requested to start and end their journeys at random points on the site.



(a) (b)

Fig. 14. Result of the indoor floor mapping experiments: **(a)** Raw trajectories, **(b)** After trajectory alignment. Extended results that show the trajectories after graph optimization are shown in Figure 21 in the appendix.

7.3.2 Results of the Trajectory Creation Algorithm. Figure 14 illustrates the outcome of our trajectory mapping process (setting: dorm-room corridor). The activity landmark trajectory data is shown in Figure 14(a). Given the initial random alignment of the trajectories, we realign them to eliminate inconsistencies as shown in Figure 14(b). The resultant map contains noise, which we remove using graph optimization to obtain an optimized floor map with reduced errors (see Figure 21 in the appendix). The final step involves aligning the generated map with the ALC information, where we get a near-ideal layout with the points of interest clearly shown (similar to Figure 13(a)). Note that we add the real layout in the background only for the purpose of better visualization.

7.3.3 Accuracy Assessment.

Layout Shape. To evaluate the quality of the generated map, we use the following metrics [61, 79, 80]:

Graph Discrepancy Metric (GDM): This measures the differences between the landmark points of the generated map and the real map (ground truth) using Euclidean distance. A smaller GDM indicates a closer match between the generated map and the real map.

Shape Discrepancy Metric (SDM): This assesses the differences in the overall shape of the generated map as compared to the real map. It involves uniformly sampling points along lines connecting landmarks in both the maps and measuring the distances between corresponding sampling points. A smaller SDM indicates a greater similarity in shape between the two maps.

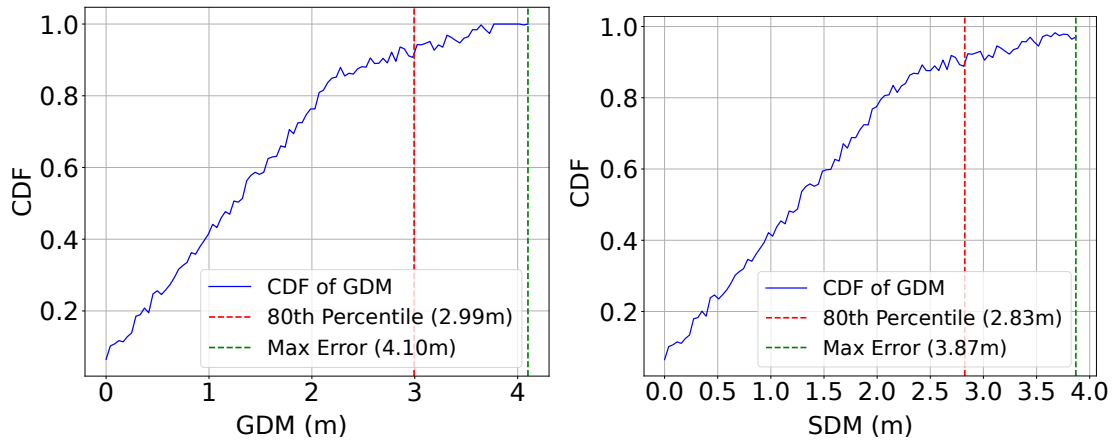


Fig. 15. CDF of **(a)** GDM and **(b)** SDM

Figure 15(a) shows the cumulative distribution function (CDF) plot for the GDM. The maximum error observed is 4.10m, and the 90^{th} percentile error is 3.42m. Similarly, for SDM, the maximum error is 3.41m, and the 90^{th} percentile error is 2.94m (refer to Figure 15(b)). Our results align closely with related work [79, 80], affirming the efficacy *AndroCon* in constructing floor maps.

Landmark Detection Accuracy: We utilized our GPS-based HAR system to accurately detect landmarks and categorized them as stairs, rooms, lifts and empty corners. Table 9 shows the accuracy of the models (RF, DT, GB, KNN, NB, SVM) for four landmarks: lift, stairs, rooms and corners. RF shows a strong performance with an average accuracy of 90.15%. DT and GB also perform well, with average accuracies of 87.5% and 87.7%, respectively. KNN has a moderate

accuracy of 81%. NB performs adequately with an average accuracy of 70.8%, but has a notably lower accuracy for lifts (61.4%).

7.4 Comparison of ML Models: All Settings

KNN is very sensitive to noise, outliers and dataset imbalance, which explains its low accuracy for all settings. Consider SVM that has the lowest average accuracy of 50.2% for layout mapping and underperforms even otherwise for our use cases. It is known to be inefficient with noisy data in multi-class labeling settings. It exhibits particularly poor accuracies for rooms (54.7%) and corners (34.3%). NB assumes feature independence, which is not the case in our settings. It has issues with continuous variables and assumes informative priors, which also don't hold in our case. Decision trees (DT) have issues with overfitting, high noise sensitivity and poor extrapolation capability. That leaves RF and GB, which are fast, and do not have many of the aforementioned problems. One of the reasons is that they rely on randomization and have more regularization.

Table 9. Accuracy of landmark classification models

Classifier	Lift	Stairs	Rooms	Corners
RF	98.5	96.2	86.2	79.7
DT	96.7	93.1	77.9	82.3
GB	96.6	93.0	78.3	82.8
KNN	91.8	84.4	72.9	74.9
NB	61.4	80.7	77.8	63.2
SVM	61.6	57.8	54.7	34.3

8 DISCUSSION

Our research uncovers a previously unexplored dimension: the ability to transform raw, semi-processed GPS data into a high-fidelity context sensor. The results are not incremental but reveal a new class of information leakage with high accuracy. Existing literature has extensively explored terrestrial RF signals, such as WiFi and Bluetooth, for HAR and ambient sensing. Our findings demonstrate that these principles extend to low-power, non-terrestrial GPS signals, opening new avenues for navigation data usage. Achieving over 99% accuracy across diverse and challenging settings—ranging from underground metros to airborne flights—demonstrates that subtle modulations in satellite signals encode a rich narrative of the surrounding environment, including fine-grained indoor context such as detecting human presence in a room and distinguishing between crowded versus empty outdoor spaces. We extend this to precise HAR, with *AndroCon* achieving over 97% accuracy in classifying postures such as lying, sitting, and standing, and detecting subtle gestures like hand waving—a difficult HAR task—with performance comparable to recent radar-based systems [63]. Central to such precision is our processing pipeline, which fuses UKF with LDA to yield stable, class-separable signals; robust across devices and environments.

In addition to ambient sensing and HAR, we also studied indoor mapping with semi-processed GPS. Our framework builds on the graph-optimization and trajectory-alignment concepts pioneered by Zhou et al. [80]; however, it introduces a key innovation. Their approach required complex fusion of multiple sensors—including WiFi, gyroscopes, and magnetometers—to detect landmarks such as walls, rooms, lifts, and turns, *AndroCon* replaces this multi-sensor setup with semi-processed GPS. This is motivated by our HAR system observations: users exhibit predictable activity patterns at points of interest, typically walking straight along pathways, turning at obstructions, or performing activities like

climbing stairs or entering elevators. These behaviors are well captured by GPS, allowing landmarks to be reliably inferred. Using this approach, *AndroCon* achieves comparable mapping precision, with a map reconstruction error of 3.9 m and an average landmark detection accuracy of 90.15%. This capability to reconstruct indoor layouts with a single, ubiquitous sensor is a significant advancement in the state of the art.

Ultimately, these high-precision results expose a critical privacy vulnerability in the Android ecosystem, arising from unrestricted access to semi-processed GPS data via the `GnssMeasurements` API [48]. This side-channel operates under the standard *fine*-location permission, allowing any application granted this access—ostensibly for legitimate location-based services—to covertly exploit the data. Such exploitation enables inference of a user's environment, their daily activities, and reconstruction of private space; creating a threat far greater than the sum of its parts. In effect, it constructs a detailed and invasive profile of a user's life, exposing over 90% [8] of Android users to this novel form of surveillance.

8.1 Limitations & Future Work

Our work relies on the Android `GNSSMeasurements` API, which provides access to semi-processed GPS measurements without explicitly informing users about associated privacy and side-channel risks. In contrast, iOS does not explicitly expose comparable raw or semi-processed GNSS data to applications. Consequently, conducting similar studies on iOS would require alternative approaches, such as developing new APIs, reverse engineering, or hardware instrumentation. Therefore, the findings and implications presented here are specific to the Android ecosystem. Investigating analogous side-channels on other platforms remains an important avenue for future research.

The proposed method assumes access to semi-processed GPS parameters on devices running Android 7 or later, alongside fine-location permissions. However, device-level restrictions, custom ROMs, or user-configured privacy settings may reduce parameter availability or sampling frequency, potentially affecting *AndroCon*'s performance.

Our evaluation focuses exclusively on GPS L1 signals, without fusion of data from other GNSS constellations. While integrating data from multiple GNSS constellations could enhance robustness, it also introduces added complexity and privacy considerations.

Although the dataset used in this study is extensive—encompassing diverse devices, participants, and environmental conditions—dynamic factors pose ongoing challenges. Rapid ionospheric variations, satellite geometry shifts, and weather fluctuations can influence GNSS signal characteristics and impact performance. Large-scale, long-term deployment would therefore require regular dataset updates and periodic model retraining to adapt to such changes. Currently, our classifiers are trained offline and do not update in real time; integrating online learning and scheduled retraining is a promising direction for future work.

The effectiveness of trajectory alignment and indoor map reconstruction depends on the availability of sufficiently dense and diverse crowdsourced user trajectories. In spatially sparse settings, supplementary data collection strategies may be necessary to ensure consistent results.

9 RELATED WORK

Given that there is an overlap in prior work on activity and ambience recognition, we combine them and discuss the combined related work in Section 9.1. Subsequently, we discuss the related work on indoor layout detection in Section 9.2.

For both event recognition and indoor floor mapping, imaging-based solutions[36, 55] captured using cameras are highly effective; however, they raise significant privacy concerns. This has limited their widespread adoption, and thus

we don't consider them in this section. We limit ourselves to methods that are primarily based EM waves (in some form) like WiFi, GPS, Bluetooth, 5G, etc.

9.1 Ambient Sensing and HAR

Year	Work	Approach	WiFi	WiFi Probe	IMU	Cell Tower	GPS	RFID	Camera	Optical
2012	Iodetector [82]	Multi-sensor (Light + Cell + Magnetic)	X	X	X	✓	X	X	X	X
2015	PAWS [28]	RSS-Based	✓	X	X	X	X	X	X	X
2016	Wang et al. [72]	Wireless signal CSI patterns	✓	X	X	X	X	X	X	X
2017	Gao et al. [25]	Wireless signal CSI patterns	✓	X	X	X	X	X	X	X
2020	Muaaz et al. [50]	WiFi CSI + IMU sensors	✓	X	✓	X	X	X	X	X
2020	Bui et al. [16]	Processed GPS signals	X	X	X	X	✓	X	X	X
2020	Bhat et al. [15]	WiFi and RSS-based	✓	X	X	X	X	X	X	X
2020	Shuaeib et al. [62]	RFID-Based	X	X	X	X	X	✓	X	X
2021	Ramirez et al. [55]	Video-based trajectory	X	X	X	X	X	X	✓	X
2021	Kouhini et al. [39]	Li-Fi	X	X	X	X	X	X	X	✓
2021	Sekiguchi et al. [60]	GPS coordinates + Other wireless signals	✓	X	X	✓	✓	X	X	X
2023	Fuada et al. [23]	Passive WiFi probe	X	✓	X	X	X	X	X	X
2024	Li et al. [41]	VLC	X	X	X	X	X	X	X	✓
2024	Zhu et al. [83]	Processed GPS signals	X	X	X	X	✓	X	X	X
2024	***AndroCon***	Semi-processed GPS signal parameters	X	X	X	X	✓	X	X	X

Table 10. Comparison of HAR/ambient classification approaches

Year	Work	Approach	SLAM	WiFi	Smartphone sensors	IMU	GPS
2012	Alzantot et al. [13]	User trajectories	X	X	✓	X	X
2013	Shen et al. [61]	WiFi RSS alignment	X	✓	X	X	X
2014	Philipp et al. [52]	Pedestrian movement+crowd-sourced structural (external) information of buildings	X	X	X	✓	X
2015	Zhou et al. [79]	Link-node approach	X	✓	✓	X	X
2016	Gu et al. [29]	User trajectory +WiFi+Bluetooth	X	✓	✓	X	X
2018	Zhou et al. [80]	Link-node approach+GO	X	✓	✓	X	X
2020	Karam et al. [37]	IMU + LiDAR	✓	X	X	✓	X
2023	Girolami et al. [26]	Bluetooth- based localization	✓	X	X	X	X
2023	Liu et al. [46]	5G-signal based positioning	✓	X	X	X	X
2024	**AndroCon**	GPS Semi-processed data + user trajectory	X	X	X	X	✓

Table 11. Comparison of indoor floor mapping approaches

We present a brief comparison of the related work on ambience/HAR (*events*) recognition in Table 10. One of the earliest works in pure RSS-based solutions is PAWS [28], which utilizes ambient WiFi signals to create RSSI fingerprints. Bhat et al. [15] refined existing techniques to develop a recognition system using the RSS of a single communication channel. Shuaeib et al. [62] introduced an RFID-based indoor HAR system that uses RSS from passive RFID tags to track activity in real-time by mapping the analyzed data to reference datasets. Such solutions rely heavily on the availability of good-quality WiFi signals, which may not always be feasible especially in outdoor settings. GPS, on the other hand, is ubiquitously available. More recently, Fuada et al. [23] leverage the WiFi probe requests from smartphones for indoor crowd monitoring, capturing RSSI, MAC addresses, and timestamps via portable nodes, and applying an UKF for noise reduction. This requires additional infrastructure and is limited by MAC randomization [14] in modern smartphones. In contrast, *AndroCon* relies solely on semi-processed GPS signals, enabling finer-grained classification beyond indoor settings.

Let us now look at works that use RSS and the channel state information (CSI). Wang et al. [72] introduced a deep learning technique, a sparse autoencoder (SAE), which recognizes events using WiFi CSI (channel state info.) signals. Gao et al. [25] developed another CSI-based system for event classification using deep learning, where they converted CSI measurements into radio images, extracted features, and then applied an SAE network for better accuracy. Muaaz et al. [50] proposed WiWeHAR, a multimodal HAR system that combines WiFi CSI data and wearable inertial measurement unit (IMU) sensor data. These systems demand precise feature engineering, efficient signal processing and noise removal.

Researchers have also explored multi-sensor frameworks for ambient sensing- IODetector from Zhou et al. [82], employing an integration of light sensors, cellular signal strength, and magnetometers to distinguish between indoor, outdoor, and semi-outdoor environments with approximately 90% accuracy, though it requires multiple sensors, adjustments and is limited to specific scenarios. In contrast, our work, AndroCon, utilizes only semi-processed GPS signals—a widely available resource—eliminating the need for additional hardware while achieving a superior 99.6% accuracy across diverse settings, including crowd density estimation in open spaces, metro stations, and conditions during flights. Zhou et al. [81] subsequently adapted IODetector for commercial deployment in China's food delivery sector, scaling it to nationwide implementation and demonstrating tangible business benefits; underscoring the practical efficacy of ambient sensing technologies. This accomplishment further strengthens the case for AndroCon's potential adoption in industry, offering a streamlined, GPS-only sensing solution characterized by simplicity and precision.

Beyond RF-based methods, OWC has also been explored for localization and HAR. Kouhini et al. [39] employed active tracking by estimating distances using the Time of Flight of LED signals transmitted from ceiling-mounted sources to a user's device. Li et al. [41] extended this to passive, device-free sensing, using ceiling-mounted transmitters and receivers to detect human activities via machine learning on reflected light disturbances. However, these approaches rely on specialized infrastructure, controlled environments, and line-of-sight conditions, making real-world deployment challenging. Importantly, *AndroCon* circumvents these limitations by exploiting *only* ubiquitous GPS signals, highlighting its robustness across diverse, real-world environments.

Prior studies have also attempted event classification using GPS and cellular signals. Sekiguchi et al. [60] categorize events based on GPS coordinates along with cell tower and WiFi signal data. This approach, however, is limited by its reliance on a stable location fix and the availability of heterogeneous signals (cellular and WiFi), confining its use to well-connected environments. Our approach does not use GPS coordinates or any other kind of location information. We solely focus on semi-processed GPS signal parameters, which are almost always available. We do not use any WiFi or assisted-GPS signals (signals from cellphone towers). This makes our work a more self-contained sensing modality. Bui et al. [16] utilized the magnitude of the received GPS signal for distinguishing between indoor and outdoor environments. Zhu et al. [83] classify ambient conditions based on visible satellites, GNSS distribution, C/N0 and multipath effects. All these studies utilize processed signal data, which, being devoid of context-rich signal–environment interactions, yields lower accuracy - at most 92.72% in the best case. On the other hand, our work leverages raw, semi-processed GPS data, a methodological choice that allows *AndroCon* to achieve a much higher accuracy that often exceeds 99%. Additionally, the range of settings considered in prior work is extremely restricted. Our work considers a very diverse set of settings (similar to prior work using stronger signals like WiFi and radars).

9.2 Indoor Layout Mapping

We show a comparison of related work in Table 11. Most approaches utilize Simultaneous Localization and Mapping (SLAM), a computer vision technique designed to localize a robot in an unknown environment while concurrently

generating a map. Conventional SLAM techniques typically depend on visual cues such as landmarks, camera-detected obstacles, sonar data or laser-range sensors (Kamar et al. [37]) [36, 47, 53]. However, these approaches incur high costs and can possibly be quite intrusive in terms of privacy.

Alzantot et al. [13] propose to use crowd-sourced user trajectories to construct indoor maps. However, their method was susceptible to errors due to the non-alignment of trajectories. Addressing the issue, Shen et al. [61] presented an indoor pathway mapping system using WiFi RSS fingerprints to align trajectories. Gu et al. [29, 30] further improved the performance by employing Bluetooth and WiFi data for trajectory alignment, but sparse WiFi access point deployment posed practical challenges. Philipp et al. [52] propose Mapgenie – generating maps from pedestrian movement traces and utilizing external building data. They propose an elaborate grammatical model to interpret pedestrian movement patterns and additionally use IMUs. Zhou et al. [79] use a link-node approach (landmarks as node and pathways and links). This method utilizes smartphone sensors and WiFi RSS fingerprinting, complemented by geometric scaling methods. However, challenges arise in accurately representing curved features within the maps. Zhou et al. [80] address these challenges through graph optimization methods.

Our approach *AndroCon* builds on the work of Zhou et al. [80] that requires WiFi RSS and MAC address fingerprinting, gyroscope and compass readings for trajectory alignment and landmark identification. *AndroCon*, on the other hand, does not need any of these sensor readings and only relies on semi-processed GPS data and GPS coordinates.

10 CONCLUSION

The fact that semi-processed GPS data can be used to sense the ambient, recognize human activity and infer floor layouts was hitherto unknown. We successfully showed that all of the above can be achieved, and that too with a high accuracy that ranges from roughly 99.5% in controlled conditions to 87% in absolutely uncontrolled conditions. We conducted an extensive set of experiments with tens of volunteers, many phones of different makes and brands, diverse set of scenarios and hundreds of sample points. We collected data for a year across a large geographical area – some of the collection points were 1000 kms away from the place where the research was primarily carried out. We also collected data on flights, cruise ships and high-altitude locations. Our results are thus quite robust.

Furthermore, our approach can construct floor maps with a maximum error margin of 4.1m as compared to the ground truth. We can classify points of interest within an indoor layout such as elevators, stairs, corridors, empty corners and rooms with a roughly 90.15% accuracy, while just relying on GPS data. Currently, Android does not address this vulnerability, which leaves approximately 90% [8] of users exposed.

RESPONSIBLE DISCLOSURE: We reported the identified vulnerability and use-cases, excluding the floor mapping use-case to the Android security team.

REFERENCES

[1] 2021. What's the difference between precise and approximate location in Android 12? https://www.androidcentral.com/whats-difference-between-precise-and-approximate-location-android-12
[2] 2023. https://www.nist.gov/pml/time-and-frequency-division/popular-links/time-frequency-z/time-and-frequency-z-g
[3] 2024. Faraday cage. https://en.wikipedia.org/wiki/Faraday_cage
[4] 2024. Feature importances with a forest of trees. https://scikit-learn.org/stable/auto_examples/ensemble/plot_forest_importances.html
[5] 2024. GnssLogger App. https://play.google.com/store/apps/details?id=com.google.android.apps.location.gps.gnsslogge
[6] 2024. GnssMeasurement. Retrieved January, 2024 from https://developer.android.com/reference/android/location/GnssMeasurement
[7] 2024. Location. Retrieved January, 2024 from https://developer.android.com/reference/android/location/Location
[8] 2024. Raw GNSS Measurements. Retrieved January, 2024 from https://developer.android.com/develop/sensors-and-location/sensors/gnss
[9] 2025. RF Explorer. https://j3.rf-explorer.com/

[10] Federal Aviation Administration. 2022. Satellite Navigation - Global Positioning System (GPS). Retrieved January, 2024 from https://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/navservices/gnss/gps

[11] Nadeem Ahmed, Jahir Ibna Rafiq, and Md Rashedul Islam. 2020. Enhanced human activity recognition based on smartphone sensor data using hybrid feature selection model. *Sensors* 20, 1 (2020), 317.

[12] Md Manjurul Ahsan, MA Parvez Mahmud, Pritom Kumar Saha, Kishor Datta Gupta, and Zahed Siddique. 2021. Effect of data scaling methods on machine learning algorithms and model performance. *Technologies* 9, 3 (2021), 52.

[13] Moustafa Alzantot and Moustafa Youssef. 2012. Crowdinside: Automatic construction of indoor floorplans. In *Proceedings of the 20th international conference on advances in geographic information systems*. 99–108.

[14] Android. 2025. MAC randomization behavior. Retrieved July, 2025 from https://source.android.com/docs/core/connect/wifi-mac-randomization-behavior

[15] Sameer Ahmad Bhat, Abolfazl Mehbodniya, Ahmed Elsayed Alwakeel, Julian Webber, and Khalid Al-Begain. 2020. Human motion patterns recognition based on RSS and support vector machines. In *2020 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 1–6.

[16] Van Bui, Nam Tuan Le, Thanh Luan Vu, Van Hoa Nguyen, and Yeong Min Jang. 2020. GPS-based indoor/outdoor detection scheme using machine learning techniques. *Applied Sciences* 10, 2 (2020), 500.

[17] Suthipong Daengduang and Peerapon Vateekul. 2016. Enhancing accuracy of multi-label classification by applying one-vs-one support vector machine. In *2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*. IEEE, 1–6.

[18] Zhiqiang Dai, Chunlei Zhai, Fang Li, Weixiang Chen, Xiangwei Zhu, and Yanming Feng. 2022. Deep-learning-based scenario recognition with GNSS measurements on smartphones. *IEEE Sensors Journal* 23, 4 (2022), 3776–3786.

[19] Kostas Drakonakis, Panagiotis Ilia, Sotiris Ioannidis, and Jason Polakis. 2019. Please forget where I was last summer: The privacy risks of public location (meta) data. In *Proceedings of the 26th Network and Distributed System Security Symposium*.

[20] Nabil M Drawil, Haitham M Amar, and Otman A Basir. 2012. GPS localization accuracy classification: A context-based approach. *IEEE Transactions on Intelligent Transportation Systems* 14, 1 (2012), 262–273.

[21] Smita Dubey, Rashmi Wahi, and AK Gwal. 2006. Ionospheric effects on GPS positioning. *Advances in Space Research* 38, 11 (2006), 2478–2484.

[22] Mia Filić and Renato Filjar. 2018. Smartphone GNSS positioning performance improvements through utilisation of Google Location API. In *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. IEEE, 0458–0461.

[23] Syifaul Fuada, Trio Adiono, Prasetiyo, Harthian Widhanto, Shorful Islam, and Tri Chandra Pamungkas. 2023. A Consumer Product of Wi-Fi Tracker System using RSSI-based Distance for Indoor Crowd Monitoring. *International Journal of Advanced Computer Science and Applications* 14, 5 (2023). https://doi.org/10.14569/IJACSA.2023.0140555

[24] Jun Gao, Dongze Wu, Feng Yin, Qinglei Kong, Lexi Xu, and Shuguang Cui. 2023. MetaLoc: Learning to Learn Wireless Localization. *IEEE Journal on Selected Areas in Communications* 41, 12 (2023), 3831–3847.

[25] Qinhua Gao, Jie Wang, Xiaorui Ma, Xueyan Feng, and Hongyu Wang. 2017. CSI-based device-free wireless localization and activity recognition using radio image features. *IEEE Transactions on Vehicular Technology* 66, 11 (2017), 10346–10356.

[26] Michele Girolami, Francesco Furfari, Paolo Barsocchi, and Fabio Mavilia. 2023. A Bluetooth 5.1 dataset based on angle of arrival and RSS for indoor localization. *IEEE Access* (2023).

[27] Google. 2024. Google Smartphone Decimeter Challenge 2023-2024. Retrieved May, 2024 from https://www.kaggle.com/competitions/smartphone-decimeter-2023/

[28] Yu Gu, Fuji Ren, and Jie Li. 2015. Paws: Passive human activity recognition based on wifi ambient signals. *IEEE Internet of Things Journal* 3, 5 (2015), 796–805.

[29] Yang Gu, Qian Song, Ming Ma, Yanghuan Li, and Zhimin Zhou. 2016. Using iBeacons for trajectory initialization and calibration in foot-mounted inertial pedestrian positioning systems. In *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 1–7.

[30] Yang Gu, Caifa Zhou, Andreas Wieser, and Zhimin Zhou. 2017. WiFi based trajectory alignment, calibration and crowdsourced site survey using smart phones and foot-mounted IMUs. In *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 1–6.

[31] Linlin Guo, Lei Wang, Chuang Lin, Jialin Liu, Bingxian Lu, Jian Fang, Zhonghao Liu, Zeyang Shan, Jingwen Yang, and Silu Guo. 2019. Wiar: A public dataset for wifi-based activity recognition. *IEEE Access* 7 (2019), 154935–154945.

[32] Gaoge Hu, Bingbing Gao, Yongmin Zhong, and Chengfan Gu. 2020. Unscented kalman filter with process noise covariance estimation for vehicular ins/gps integration system. *Information Fusion* 64 (2020), 194–204.

[33] Magdalena Jackiewicz. Accessed on: April 16 2024. Location-based app development: a complete 2022 guide for CEOs. https://www.rst.software/blog/location-based-app-development-a-complete-2022-guide-for-ceos

[34] Volker Janssen. 2024. Understanding the RINEX format. In *Proceedings of Association of Public Authority Surveyors Conference (APAS2024)*. 3–18.

[35] Simon J Julier and Jeffrey K Uhlmann. 1997. New extension of the Kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI*, Vol. 3068. Spie, 182–193.

[36] Zhizhong Kang, Juntao Yang, Zhou Yang, and Sai Cheng. 2020. A review of techniques for 3d reconstruction of indoor environments. *ISPRS International Journal of Geo-Information* 9, 5 (2020), 330.

[37] Samer Karam, G Vosselman, and VV Lehtola. 2020. Strategies to integrate IMU and LiDAR SLAM for indoor mapping. In *XXIVth ISPRS Congress 2020*. Copernicus, 223–230.

[38] Weiheng Kong, Lili He, and Hailong Wang. 2021. Exploratory data analysis of human activity recognition based on smart phone. *IEEE Access* 9 (2021), 73355–73364.

[39] Sepideh Mohammadi Kouhini, Ziyan Ma, Christoph Kottke, Sreelal Maravanchery Mana, Ronald Freund, and Volker Jungnickel. 2021. Lifi based positioning for indoor scenarios. In *2021 17th International Symposium on Wireless Communication Systems (ISWCS)*. IEEE, 1–5.

[40] Ambuj Kumar, Albena Mihovska, Sofoklis Kyriazakos, and Ramjee Prasad. 2014. Visible light communications (VLC) for ambient assisted living. *Wireless personal communications* 78, 3 (2014), 1699–1717.

[41] Jiarong Li, Changshuo Ge, Chihan Xu, Junhao Gong, Weihua Gui, Chaobo Zhang, Xiaojun Liang, and Wenbo Ding. 2024. Vlocsense: Integrated vlc system for indoor passive localization and human sensing. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*. 2022–2027.

[42] Jie Lian, Changlai Du, Jiadong Lou, Li Chen, and Xu Yuan. 2024. EchoSensor: Fine-grained Ultrasonic Sensing for Smart Home Intrusion Detection. *ACM Trans. Sens. Networks* 20, 1 (2024), 10:1–10:24. https://doi.org/10.1145/3615658

[43] Jie Lian, Xu Yuan, Jiadong Lou, Li Chen, Hao Wang, and Nian-Feng Tzeng. 2024. Room-scale Location Trace Tracking via Continuous Acoustic Waves. *ACM Trans. Sens. Networks* 20, 3 (2024), 61:1–61:23. https://doi.org/10.1145/3649136

[44] Qi Lin, Shuhua Peng, Yuezhong Wu, Jun Liu, Hong Jia, Wen Hu, Mahbub Hassan, Aruna Seneviratne, and Chun Hui Wang. 2023. Subject-adaptive Loose-fitting Smart Garment Platform for Human Activity Recognition. *ACM Trans. Sens. Networks* 19, 4 (2023), 84:1–84:23.

[45] Jian Liu, Hongbo Liu, Yingying Chen, Yan Wang, and Chen Wang. 2019. Wireless sensing for human activity: A survey. *IEEE Communications Surveys & Tutorials* 22, 3 (2019), 1629–1645.

[46] Zhaoliang Liu, Liang Chen, Xin Zhou, Zhenhang Jiao, Guangyi Guo, and Ruizhi Chen. 2023. Machine learning for time-of-arrival estimation with 5G signals in indoor positioning. *IEEE Internet of Things Journal* 10, 11 (2023), 9782–9795.

[47] Chris Xiaoxuan Lu, Stefano Rosa, Peijun Zhao, Bing Wang, Changhao Chen, John A Stankovic, Niki Trigoni, and Andrew Markham. 2020. See through smoke: robust indoor mapping with low-cost mmwave radar. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*. 14–27.

[48] Steve Malkos. 2016. Google to provide raw GNSS measurements - GPS World. https://www.gpsworld.com/google-to-provide-raw-gnss-measurements/

[49] Mappitall. Accessed on: April 16 2024. How Location-Based Services Function and Why They Are Essential. https://www.linkedin.com/pulse/how-location-based-services-function-why-essential-mappitall-vulbf/

[50] Muhammad Muaaz, Ali Chelli, Ahmed Abdelmonem Abdelgawwad, Andreu Català Mallofré, and Matthias Pätzold. 2020. WiWeHAR: Multimodal human activity recognition using Wi-Fi and wearable sensing modalities. *IEEE access* 8 (2020), 164453–164470.

[51] Ohoud Nafea, Wadood Abdul, Ghulam Muhammad, and Mansour Alsulaiman. 2021. Sensor-based human activity recognition with spatio-temporal deep learning. *Sensors* 21, 6 (2021), 2141.

[52] Damian Philipp, Patrick Baier, Christoph Dibak, Frank Dürr, Kurt Rothermel, Susanne Becker, Michael Peter, and Dieter Fritsch. 2014. Mapgenie: Grammar-enhanced indoor map construction from crowd-sourced data. In *2014 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 139–147.

[53] Darius Plikynas, Arūnas Žvironas, Andrius Budrionis, and Marius Gudauskis. 2020. Indoor navigation systems for visually impaired persons: Mapping the features of existing technologies to user needs. *Sensors* 20, 3 (2020), 636.

[54] Qifan Pu, Sidhant Gupta, Shyamnath Gollakota, and Shwetak Patel. 2013. Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th annual international conference on Mobile computing & networking*. 27–38.

[55] Heilym Ramirez, Sergio A Velastin, Ignacio Meza, Ernesto Fabregas, Dimitrios Makris, and Gonzalo Farias. 2021. Fall detection and activity recognition using human skeleton features. *Ieee Access* 9 (2021), 33532–33542.

[56] Ananth Ranganathan. 2004. The levenberg-marquardt algorithm. *Tutoral on LM algorithm* 11, 1 (2004), 101–110.

[57] Allied Market Research. Accessed on: April 16 2024. Location Based Services Market Size, Share, Competitive Landscape and Trend Analysis Report by Component, by Technology, by Application, by Industry Vertical : Global Opportunity Analysis and Industry Forecast, 2021-2031. https://www.alliedmarketresearch.com/location-based-services-market

[58] Piotr Sapiezynski, Arkadiusz Stopczynski, Radu Gatej, and Sune Lehmann. 2015. Tracking human mobility using wifi signals. *PloS one* 10, 7 (2015), e0130824.

[59] Alireza Sarveniazi. 2014. An actual survey of dimensionality reduction. *American Journal of Computational Mathematics* 4, 02 (2014), 55–72.

[60] Ryoichi Sekiguchi, Kenji Abe, suzuki shogo, Masayasu Kumano, Daisuke Asakura, Ryo Okabe, Takeru Kariya, and Masaki Kawakatsu. 2021. Phased human activity recognition based on GPS. In *Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers*. 396–400.

[61] Guobin Shen, Zhuo Chen, Peichao Zhang, Thomas Moscibroda, and Yongguang Zhang. 2013. {Walkie-Markie}: indoor pathway mapping made easy. In *10th USENIX symposium on networked systems design and implementation (NSDI 13)*. 85–98.

[62] Wafa Shuaieb, George Oguntala, Ali AlAbdullah, Huthaifa Obeidat, Rameez Asif, Raed A Abd-Alhameed, Mohammed S Bin-Melha, and Chakib Kara-Zaïtri. 2020. RFID RSS fingerprinting system for wearable human activity recognition. *Future Internet* 12, 2 (2020), 33.

[63] Arthur Sluÿters, Sébastien Lambot, Jean Vanderdonckt, and Radu-Daniel Vatavu. 2023. RadarSense: Accurate recognition of mid-air hand gestures with radar sensing and few training examples. *ACM Transactions on Interactive Intelligent Systems* 13, 3 (2023), 1–45.

[64] Masashi Sugiyama. 2006. Local fisher discriminant analysis for supervised dimensionality reduction. In *Proceedings of the 23rd international conference on Machine learning*. 905–912.

[65] Masashi Sugiyama. 2007. Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis. *Journal of machine learning research* 8, 5 (2007).

[66] J Tomaštík and M Varga. 2020. Raw GNSS data, Xiaomi Mi 8, 10 minutes, 3 points, 40 days, Mendeley Data.

[67] Julián Tomaštík, Matej Varga, and Tim Everett. 2024. Raw GNSS data collected using smartphones and low-cost receiver under optimal and sub-optimal conditions. *Data in Brief* 53 (2024), 110148.

[68] Michel Verleysen and Damien François. 2005. The curse of dimensionality in data mining and time series prediction. In *International work-conference on artificial neural networks*. Springer, 758–770.

[69] B Vidya and P Sasikumar. 2022. Wearable multi-sensor data fusion approach for human activity recognition using machine learning algorithms. *Sensors and Actuators A: Physical* 341 (2022), 113557.

[70] Bac Nghia Vu and Miloš Andrle. 2014. The code and carrier tracking loops for GPS signal. In *Proceedings of the 16th International Conference on Mechatronics-Mechatronika 2014*. IEEE, 569–574.

[71] Jin Wang and Jun Luo. 2021. No Perfect Outdoors: Towards a Deep Profiling of GNSS-Based Location Contexts. *Future Internet* 14, 1 (2021), 7.

[72] Jie Wang, Xiao Zhang, Qinhua Gao, Hao Yue, and Hongyu Wang. 2016. Device-free wireless localization and activity recognition: A deep learning approach. *IEEE Transactions on Vehicular Technology* 66, 7 (2016), 6258–6267.

[73] Shengling Wang, Qin Hu, Yunchuan Sun, and Jianhui Huang. 2018. Privacy preservation in location-based services. *IEEE Communications Magazine* 56, 3 (2018), 134–140.

[74] Yingxiao Wu, Haocheng Ni, Changlin Mao, Jianping Han, and Wenyao Xu. 2024. Non-intrusive Human Vital Sign Detection Using mmWave Sensing Technologies: A Review. *ACM Trans. Sens. Networks* 20, 1 (2024), 16:1–16:36.

[75] Yan Xia, Shuguo Pan, Wang Gao, Baoguo Yu, Xingli Gan, Yue Zhao, and Qing Zhao. 2020. Recurrent neural network based scenario recognition with Multi-constellation GNSS measurements on a smartphone. *Measurement* 153 (2020), 107420. https://doi.org/10.1016/j.measurement.2019.107420

[76] Heng-Li Yang and Shiang-Lin Lin. 2018. User adoption of location-based service. In *2018 Tenth International Conference on Advanced Computational Intelligence (ICACI)*. 51–56.

[77] Farzaneh Zangenehnejad and Yang Gao. 2021. GNSS smartphones positioning: Advances, challenges, opportunities, and future perspectives. *Satellite navigation* 2 (2021), 1–23.

[78] Yue Zheng, Yi Zhang, Kun Qian, Guidong Zhang, Yunhao Liu, Chenshu Wu, and Zheng Yang. 2019. Zero-effort cross-domain gesture recognition with Wi-Fi. In *Proceedings of the 17th annual international conference on mobile systems, applications, and services*. 313–325.

[79] Baoding Zhou, Qingquan Li, Qingzhou Mao, Wei Tu, Xing Zhang, and Long Chen. 2015. ALIMC: Activity landmark-based indoor mapping via crowdsourcing. *IEEE Transactions on Intelligent Transportation Systems* 16, 5 (2015), 2774–2785.

[80] Baoding Zhou, Qingquan Li, Guanxun Zhai, Qingzhou Mao, Jun Yang, Wei Tu, Weixing Xue, and Long Chen. 2018. A graph optimization-based indoor map construction method via crowdsourcing. *IEEE Access* 6 (2018), 33692–33701.

[81] Pengfei Zhou, Yi Ding, Yang Li, Mo Li, Guobin Shen, and Tian He. 2022. Experience: Adopting indoor outdoor detection in on-demand food delivery business. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*. 94–105.

[82] Pengfei Zhou, Yuanqing Zheng, Zhenjiang Li, Mo Li, and Guobin Shen. 2012. Iodetector: A generic service for indoor outdoor detection. In *Proceedings of the 10th acm conference on embedded network sensor systems*. 113–126.

[83] Feng Zhu, Kegan Luo, Xianlu Tao, and Xiaohong Zhang. 2024. Deep Learning Based Vehicle-Mounted Environmental Context Awareness via GNSS Signal. *IEEE Transactions on Intelligent Transportation Systems* (2024).

A APPENDIX

A.1 Glossary

Table 12 lists the abbreviations used in this paper along with their expanded forms.

Table 12. List of abbreviations used in this paper.

Abbreviation	Expanded Form
ADRng	Accumulated Delta Range
ADRngU	Accumulated Delta Range Uncertainty
Agc	Automatic Gain Control (incoming signal power)
BbCN0	Baseband Carrier-to-Noise Density
CN0	Carrier-to-Noise Density
DT	Decision Tree
GB	Gradient Boosting
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HAR	Human Activity Recognition
KNN	K-Nearest Neighbor
LDA	Linear Discriminant Analysis
NB	Naive Bayes
NMEA	National Marine Electronics Association
OVO	One-Versus-One
PCA	Principal Component Analysis
PR	Pseudorange Rate
PRU	Pseudorange Rate Uncertainty
PVT	Position, Velocity, and Time
RecSvTU	Received SV (Satellite Vehicle) Time Uncertainty
RF	Random Forest
RINEX	Receiver Independent Exchange Format
RSS	Received Signal Strength
SNR	Signal-to-Noise Ratio
SVM	Support Vector Machine
SvID	Satellite Vehicle ID
UKF	Unscented Kalman Filter

A.2 Semi-processed GPS Data Parameters: Mindmap

A brief taxonomy of the chosen GPS parameters is shown in Figure 16. We broadly use five types of parameters that can be classified into the following buckets: received signal power, carrier phase shift, multipath interference, SNR and the Doppler shift (due to satellite motion). These parameters can be used to extract information about the surroundings because they are influenced by it. For instance, if there are a lot of objects around the receiver, the multi-path interference and SNR will be high.

Fig. 16. Representation of the semi-processed GPS signal parameters that we use. RSS stands for "Received Signal Strength"

A.3 Ambient Classification

Figure 17 illustrates the accuracy of the selected models under each experimental configuration, after going through a pre-processing pipeline consisting of UKF, normalization through scaling, LDA, and parameter fusion. The figure shows the confusion matrices.

Table 13 shows the same data for each setting using several popular ML metrics: accuracy, sensitivity, specificity and the F-score.

Figure 18 shows that GB achieves an accuracy of 99.25% in further classifying indoor environments based on population density.

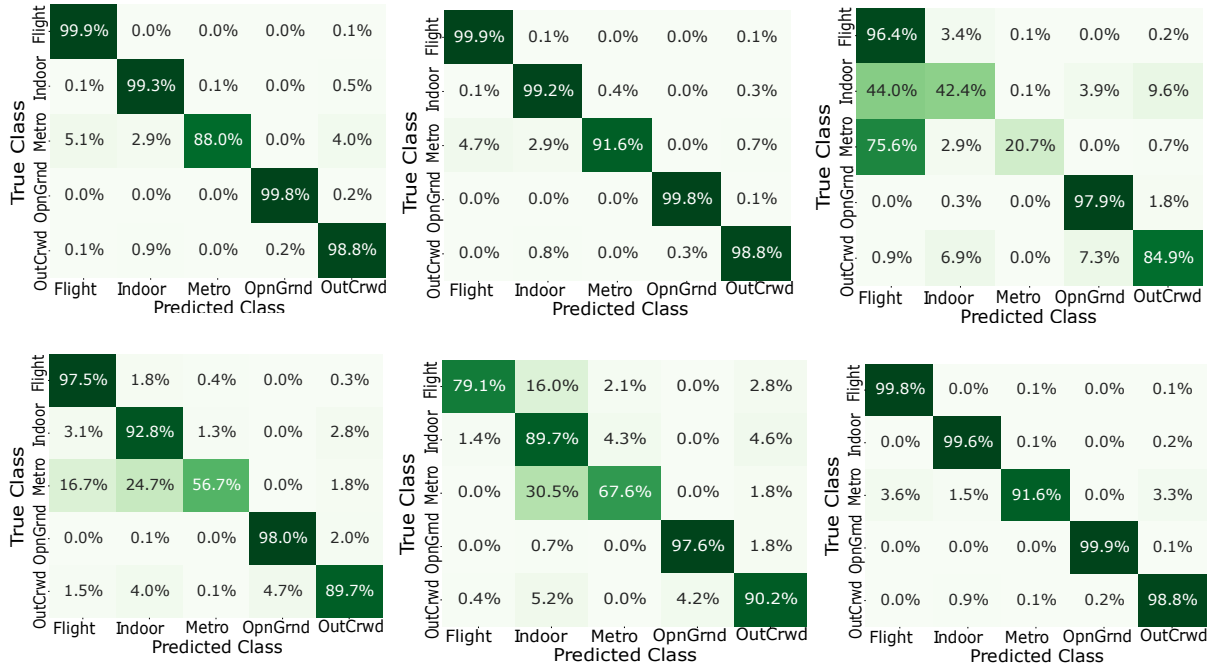


Fig. 17. Ambient classification confusion matrices: **(a)** RF, **(b)** DT, **(c)** SVM, **(d)** KNN, **(e)** NB, and **(f)** GB

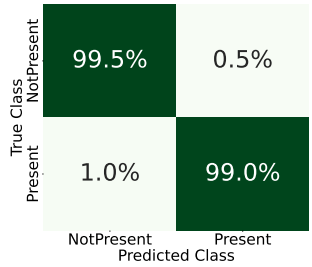


Fig. 18. Confusion matrix of indoor environments with different population densities

A.4 Activity Classification

Figure 19 shows the utility of each GPS parameter in classifying the activity. We can clearly see State, Agc, CN0, BbCN0 and ADRng to be important parameters. However, in isolation their accuracy hovers in the low 80-85% range. Hence, fusing the parameters is essential.

Figure 20 compares the accuracies of different models using the pre-processed data and fused parameters. Table 14 shows the same data along with other ML metrics: sensitivity, specificity and the F-score. We can observe that SVM performs the worst (accuracy = 57.4%) and RF performs the best (accuracy = 87%).

Table 13. Performance metrics of ambient classifier models: GB and RF are the best. The results are averaged across all the phones and volunteers.

Classifier	Ambience	Acc	Sen	Spe	F-score
GB	Flight	99.8	99.8	99.9	99.8
	Indoor	99.6	99.6	99.8	99.4
	Metro	91.6	91.6	99.9	93.9
	OpnGrnd	99.9	99.9	100.0	99.9
	OutCrwd	98.8	98.8	99.8	98.7
RF	Flight	99.5	99.9	99.9	99.8
	Indoor	99.3	99.3	99.8	99.2
	Metro	88	88.0	100.0	93.3
	OpnGrnd	99.8	99.8	100.0	99.9
	OutCrwd	98.8	98.8	99.7	98.4
DT	Flight	99.9	99.9	99.9	99.8
	Indoor	99.2	99.2	99.8	99.1
	Metro	91.6	91.6	99.9	93.2
	OpnGrnd	99.8	99.8	99.9	99.9
	OutCrwd	98.8	98.8	99.9	98.9
KNN	Flight	97.5	97.5	98.7	97.2
	Indoor	92.8	92.8	98.4	92.4
	Metro	56.7	56.7	99.7	62.2
	OpnGrnd	98	98.0	99.0	98.3
	OutCrwd	89.7	89.7	98.4	89.1
NB	Flight	79.1	79.1	99.6	87.8
	Indoor	89.7	89.7	93.0	79.6
	Metro	67.6	67.6	98.7	51.0
	OpnGrnd	97.6	97.6	99.1	98.2
	OutCrwd	90.2	90.2	97.4	86.0
SVM	Flight	96.4	96.4	88.4	85.4
	Indoor	42.4	42.4	97.7	55.0
	Metro	20.7	20.7	100.0	33.7
	OpnGrnd	97.9	97.9	97.4	97.2
	OutCrwd	84.9	84.9	97.3	82.9

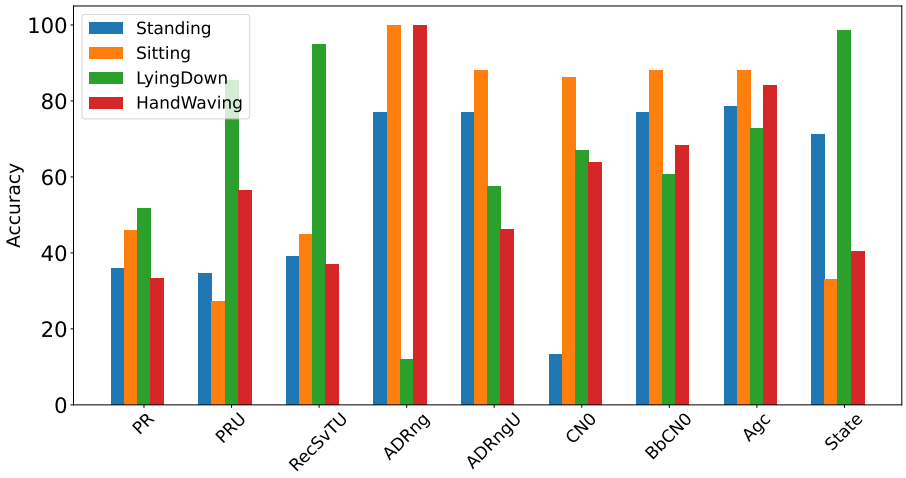


Fig. 19. Activity classification accuracy for each semi-processed GPS parameter

A.4.1 HAR Accuracy Across Mobile Devices. Table 15 shows HAR accuracy across devices under consistent conditions. Raw accuracies ranged from 47.8% (Redmi Note K20 Pro, 2020; ±9.3% variance across same model) to 74% (Nothing

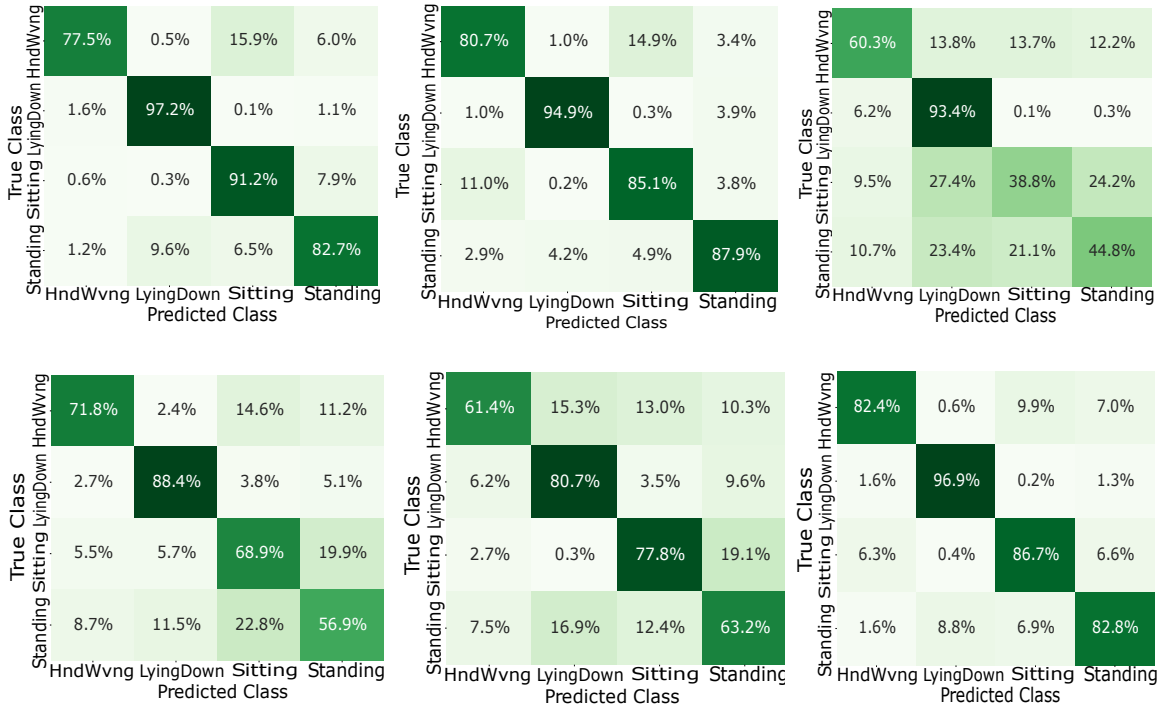


Fig. 20. Activity classification confusion matrix: **(a)** RF, **(b)** DT, **(c)** SVM, **(d)** KNN, **(e)** NB, and **(f)** GB

Phone 3, 2025; ±4.1% variance), indicating uneven performance. Post-UKF+LDA, accuracies increased to 84.5–88%, with variance reduced to ±3.2–4.6%, improving stability across diffferent mobile devices.

Table 14. Performance metrics of activity classifier models

Classifier	Activity	Acc	Sen	Spe	F-score
RF	HandWaving	77.5	77.5	98.9	85.6
	LyingDown	97.2	97.2	96.5	92.6
	Sitting	91.2	91.2	92.3	86.4
	Standing	82.7	82.7	94.7	83.8
DT	HandWaving	80.7	80.7	94.7	81.7
	LyingDown	84.9	94.9	98.2	94.2
	Sitting	85.1	85.1	93.2	84.0
	Standing	87.9	87.9	96.3	88.8
GB	HandWaving	82.4	82.4	96.7	85.4
	LyingDown	96.9	96.9	96.7	92.8
	Sitting	86.7	86.7	94.1	85.9
	Standing	82.8	82.8	94.8	84.0
KNN	HandWaving	71.8	71.8	94.2	75.4
	LyingDown	88.4	88.4	93.4	83.3
	Sitting	68.9	68.9	85.6	66.8
	Standing	56.9	56.9	87.3	59.3
NB	HandWaving	61.4	61.4	94.6	68.8
	LyingDown	80.7	80.7	89.5	73.8
	Sitting	77.8	77.8	90.1	76.4
	Standing	63.2	63.2	86.6	63.1
SVM	HandWaving	60.3	60.3	91.0	63.8
	LyingDown	93.4	93.4	78.1	68.7
	Sitting	38.3	38.8	87.7	45.4
	Standing	44.8	44.8	86.8	49.4

Table 15. HAR accuracy variability across mobile devices with and without UKF+LDA

Device Model (Year)	Raw Data		Post UKF+LDA Data	
	Accuracy (%)	Accuracy Variability (%)	Accuracy (%)	Accuracy Variability (%)
Redmi Note 9 Pro Max (2020)	47.8	±9.2	84.5	±4.4
Redmi Note K20 Pro (2020)	51.2	±9.3	85.3	±3.9
Redmi Note 9 Pro Max (2021)	52.1	±8.4	84.9	±3.7
OnePlus Nord CE2 (2022)	67.6	±7.8	85.3	±4.6
Galaxy A54 (2023)	72.1	±4.3	87.3	±3.5
Nothing Phone 3 (2025)	74.0	±4.1	88.2	±3.1

A.5 Indoor Layout Mapping

Figure 21 shows the extended results for indoor floor mapping. Figure 21(a) shows the raw trajectories that need to be rotated and aligned. Figure 21(b) shows the results after alignment. Subsequently, some graph-based optimization is performed to reduce the noise and create a fine trajectory (results in Figure 21(c)). Finally, the trajectory is rotated and aligned. The points of interest (based on the Agc) are also demarcated. Figure 21(d) shows the final output of the trajectory with the points of interest clearly marked. Note that the layout is shown in the background to enhance readability. It is not available to our algorithm at this stage.

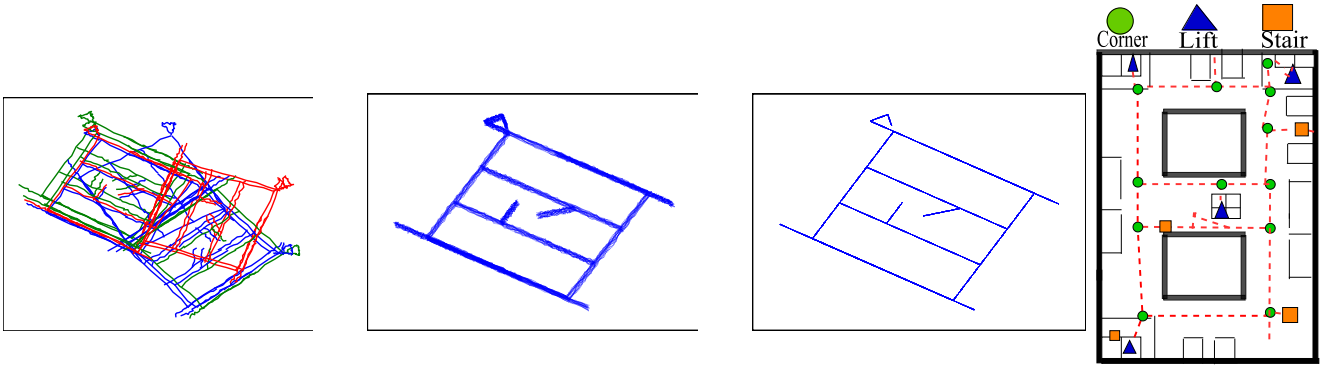


Fig. 21. Result of the indoor floor mapping experiments: **(a)** Raw trajectories, **(b)** After trajectory alignment, **(c)** After graph optimization, and **(d)** Final result after aligning with a coordinate system