Wi-Fi fingerprinting based on collaborative confidence level training

Hao Jing[1]*, James Pinchin[2], Chris Hill[1], Terry Moore[1]

1 Nottingham Geospatial Institute, University of Nottingham, NG7 2TU, UK

2 Horizon Digital Economy Research, University of Nottingham, NG7 2TU, UK

*Email: lgxhj2@nottingham.ac.uk

Abstract:

Wi-Fi fingerprinting has been a popular indoor positioning technique with the advantage that infrastructures are readily available in most urban areas. However wireless signals are prone to fluctuation and noise, introducing errors in the final positioning result. This paper proposes a new fingerprint training method where a number of users train collaboratively and a confidence factor is generated for each fingerprint. Fingerprinting is carried out where potential fingerprints are extracted based on the confidence factor. Positioning accuracy improves by 40% when the new fingerprinting method is implemented and maximum error is reduced by 35%.

Keywords:

Indoor positioning, Wi-Fi fingerprinting, collaborative positioning

1 Introduction

With the advancement in positioning as well as mobile technologies, location based services (LBS) are no longer just trendy fantasies. LBS applications are expanding from military and government sectors rapidly into commercial and civil applications. Therefore, the fundamental requirement for positioning and navigation is becoming more demanding as solutions are required in more complicated environments, where traditional positioning methods such as Global Navigation Satellite Systems (GNSS) fails. This is known as the indoor positioning problem and numerous methods have been explored over the years to improve positioning performance in such environments [1,2]. Wireless network signal based positioning, such as Wi-Fi fingerprinting, have become widely applied in indoor positioning due to high availability of Wi-Fi signals in urban environments [3,4].

Yet Wi-Fi positioning is far from the perfect solution. Wi-Fi networks are not positioning dedicated systems thus signals can be unstable, and sometimes unsuitable for positioning. Hence accuracy and robustness cannot be guaranteed [5]. The complete process of Wi-Fi fingerprinting is achieved in two phases, the training phase which must be carried out first to collect received signal strength (RSS) measurements and the positioning phase to obtain positions based on the fingerprints [6,7]. The positioning performance of fingerprinting relies on the applied positioning algorithm as well as the accuracy and details of the fingerprint database. Therefore, in order to achieve accurate positioning, a detailed database is required. This relies on carefully chosen training points across the building as well as sufficient access points (AP) that covers the area of interest, as more AP will give more information on the variation of signals when in different locations. However, training can be very time consuming. Yet it must be retrained and updated whenever the internal building structure or AP locations change [8,9]. Moreover, although many buildings have been setup with dedicated dense wireless network enabling high accuracy positioning, but most indoor environment still lack such coverage.

To reduce the time and human labour required for database training, Wi-Fi simultaneous localisation and mapping (SLAM) had been applied to enable a quicker way of learning the signal pattern around a new environment and allows the system to navigate in a new environment. However inertial measurements and building information is required, and building information may not always be accessible [10-12].

A basic requirement in fingerprinting is that the positions of the fingerprints must be accurate and their RSS measurement should be up-to-date. Studies have looked into the possibility of reducing training effort by reducing training points and training time [13]. Authors in [14,15] looks into autonomous crowdsourcing method for training and updating the Wi-Fi database. For crowd-sourced database, the accuracy of estimated positions of the training data is essential. Collaborative positioning improves positioning accuracy and reliability by applying network constraints on user's positioning measurements. Nearby users may form local networks where relative constraints can be applied to adjust and share each other's measurements [16,17]. Authors in [18,19] improve fingerprinting performance by allowing the user to interact with the system to label locations and changes. However this requires active collaboration with the user who may not be willing or could potentially make mistakes.

This paper looks into reducing the training effort by introducing a collaborative Wi-Fi fingerprint database training (cFPDB) approach, which achieves quicker and more reliable training. Gaussian Process (GP) regression is applied to generate fingerprints for the entire database from a small amount of training data and a confidence factor is produced for each fingerprint indicating how reliable it is. On the other hand, this solution especially addresses Wi-Fi positioning problems in locations where dedicated network is unavailable and are covered only by very sparse APs. With very few APs, users may not be able to observe enough signal variation patterns for accurate positioning. An adaptive collaborative fingerprinting algorithm (WARCP) based on the concept of collaborative positioning is also introduced which provides the location reference for fingerprints as well as knowledge on the expected relationship between Wi-Fi measurements collected by nearby users. Positioning flexibility is also improved as users have the option of performing inertial navigation alone, with collaborative ranging aiding or Wi-Fi fingerprint aiding based on available sensors and number of users.

This paper firstly introduces the collaborative Wi-Fi fingerprint training method and an analysis on training data is presented to understand how much data is required for generating a reliable database. WARCP is then discussed to achieve positioning based on the collaboratively trained database and ranging constraint between users. Simulations are carried out based on the proposed algorithms and discussed in Section 4. Both training and positioning results are analysed for efficient and reliable Wi-Fi fingerprint training and positioning.

2 Collaborative Wi-Fi database training

2.1 Wi-Fi fingerprinting

Wireless network based positioning relies on measuring the signal strength of the received signals. Wireless signal strength will attenuate as it travels from the transmitter (i.e. Wi-Fi APs) to the receiver based on the signal path loss model [3],

$$P_{RX}(d) = P_{d0} - 10n \log_{10} d + a \cdot WAF + \varepsilon \tag{1}$$

where P_{d0} is the RSS in dB at a reference distance, usually 1m, away from the AP, n is the space loss factor which varies in different environments, WAF is the wall attenuation factor and a is the number of obstructions in between the receiver and AP, ε is a zero mean Gaussian distributed noise. Positions can be obtained based on computing the change of signal strength from each AP to the receiver. However, wireless signals are quite noisy due to interference and obstructions inside buildings. Therefore the actual observation $\tilde{P}$ and the expected $P_{RX}(d)$ from Eq.1 can differ up to 20dB. Wi-Fi fingerprinting overcomes this problem by taking advantage of signal disruptions in complicated environments. Although signals are easily disturbed and measurement error ε is hard to predict, but as long as the building structure remains unchanged, the disturbance reflected in the signal strength will remain alike in the same location. Therefore, the RSS measurements from each AP form a pattern that reflects a specific location, known as fingerprints.

The first step of fingerprinting is the training phase, where a number of locations, known as training points (TPs), are selected within the area of interest and the RSS from all APs are measured at each TP. These are stored into a database as one fingerprint. If the RSS are carefully measured, APs are well spread out and the structure of the building is complicated enough, each fingerprint should be unique referring to one specific location in the building. During the positioning phase, the user measures the current RSS and compares it to the fingerprints in the database. Usually, the mean location of k fingerprints with the smallest difference to the current RSS, known as the k-nearest neighbours (kNN), is returned as the estimated position [20].

The biggest problem with fingerprinting is that the training process requires a huge amount of human labour, especially in large complex buildings. This increases the possibility of human error and time cost. Moreover, the database needs to be retrained and updated each time the infrastructure changes to maintain an up-to-date database for accurate positioning.

2.2 Database training

Assuming that the RSS of a location is correlated to the RSS of a nearby TP based on Eq.1, GP is applied to enable faster and more efficient training, which makes the database easier to maintain and update.

For accurate database training, the selected TPs should cover the entire area of interest and RSS should be collected over a period of time on each TP to fully reflect the variance and stability of the signal from each AP. Each fingerprint is typically structured as $\{(x_n, y_n)|RSS_{n1}, \sigma_{n1}, AP_1 \cdots, RSS_{nm}, \sigma_{nm}, AP_m\}$, where (x_n, y_n) is the position of the *n*th fingerprint, RSS_{nm} is the mean RSS and σ_{nm} is the standard deviation of the *m*th AP at the *n*th TP, AP_m is the unique identification of the AP, usually the MAC address. The uniqueness of the fingerprint is enhanced by the number of APs found and the amount of RSS collected.

As positioning is achieved by comparing RSS to the fingerprints, more fingerprints would mean more detailed database, which potentially results in better positioning. The most straightforward way of increasing fingerprints would be increasing TPs. However, it is almost impossible to cover the entire floor plan with TPs due to the required amount of work. Therefore, the entire area is usually divided into evenly distributed grids and a TP is placed at the centre point of each grid, assuming that the RSS is the same within each grid. Typical grid sizes are 1m×1m, 2m×2m [21]. Smaller grids ensure a more detailed database, although it will be more time consuming and laborious.

Based on the path loss model, we can see that the signal strength at each TP is correlated to its distance to the AP. In locations with fewer obstructions, the signals behave according to the model with a small noise. Hence less TPs are required as the RSS can be predicated from RSS at nearby TPs based on Eq.1. GP is applied to predict the RSS of fingerprints that are near to but not on TPs. In areas where training data has already been collected, GP increases the density of the fingerprints without increasing the number of TP. If $(x, z)$are samples drawn from a noisy process [22],

$$z_i = f(x_i) + \varepsilon \tag{2}$$

where each x_i is an input sample and z_i is the target or observation value, ε is assumed to be a zero mean normally distributed noise. Gaussian process estimates the posterior distributions over the functions f from the training data which is specified by a mean function $m(x)$ and a covariance function, or kernel $k(x, x')$, which describes the correlation between two input values x_p and x_q. The squared exponential kernel is applied here,

$$k(x_p, x_q) = \sigma_f^2 \exp\left(-\frac{1}{2\ell}\left|x_p - x_q\right|^2\right) + \sigma_n^2 \tag{3}$$

where σ_f^2 is the signal variance, ℓ is the length scale that describes the strength of correlation over a distance, σ_n^2 is the Gaussian observation noise. The RSS measurements and the locations of TPs are input into the system to train for the hyperparameters $\theta = \langle \sigma_n^2, \ell, \sigma_f^2 \rangle$, which define the predication functions. The predication process is then carried out based on the predicative distribution

$$p(z_*|x_*, X, Z) = \int p(z_*|f(x_*))p(f(x_*)|x_*, X, Z)\,\mathrm{d}f(x_*) \tag{4}$$

The locations of the TPs are input as X while the RSS measured at the TPs are the target values Z. The desired fingerprints cover the building by 1m grids at locations x_*, and the RSS of the desired fingerprints z_* is predicted based on the trained predication functions.

To understand the required density and location setup of training data for generating accurate fingerprint database, different training methods are compared. A Toshiba laptop is used throughout the trials in this paper for consistency, whose wireless adapter is Intel® Centrino® Advanced-N 6200. Four APs are located on Floor A of Nottingham Geospatial Building (NGB), each transmitting signals on both 2.4GHz and 5GHz frequencies. As signal characteristics are different, thus the signals from different frequencies will be treated separately. A full database consists eight MAC address groups, each denoted as AP1a (2.4GHz), AP1b (5GHz), AP2a, AP2b, AP3a, AP3b, AP4a and AP4b respectively. Locations of the APs are indicated in Figure 1. Figure 2 shows the spread of the RSS for 2.4GHz and 5GHz signals during 30 minutes.

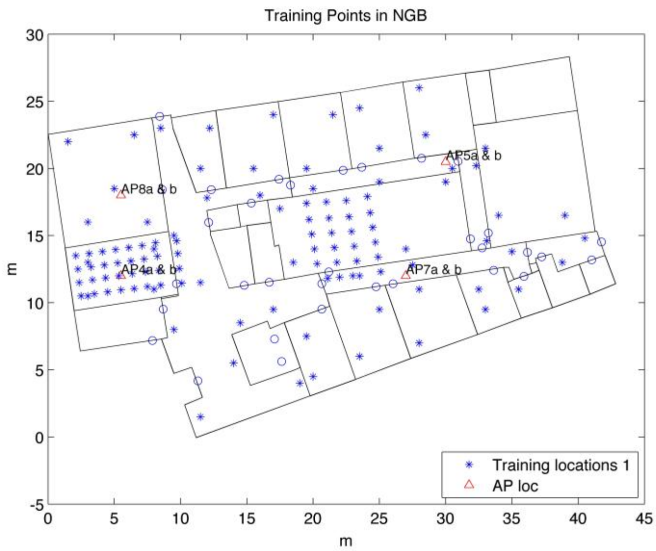


Figure 1 TPs selected for full fingerprint database

56 TPs are selected to cover the entire accessible area in NGB Floor A to establish the ground truth for the fingerprint database, i.e. the best possible database from conventional training. On average, two TPs are located inside a small office and four to six TPs are located in large rooms. The laptop is placed at each location to collect the Wi-Fi RSS data for around thirty minutes until at least 100 vectors from each of the four APs have been collected. The mean and standard deviation of all the collected RSS from all APs at each TP is obtained and sorted into the training input vector. GP is then applied based on the training data to increase the fingerprint density to 1m×1m. The resulting fingerprints are stored into a database, denoted as sDB. The training data were collected while the receiver was static and placed over the TP to obtain more stable information of the signal. Training for the 56 TPs takes around 37 hours in total.

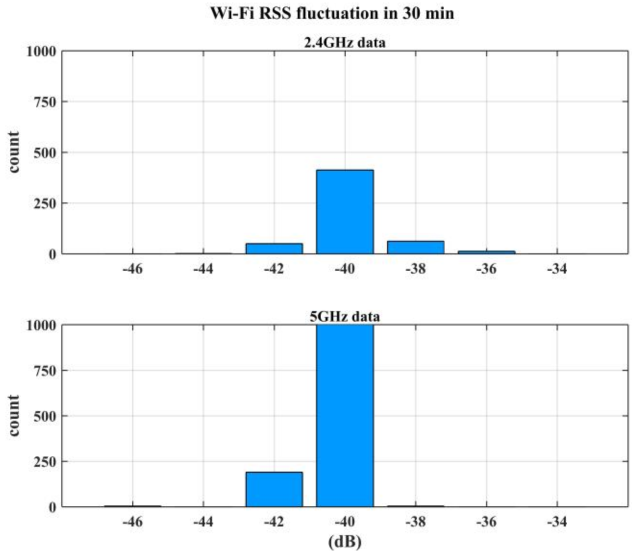


Figure 2 Wi-Fi RSS fluctuation in 30 minutes

Two rooms are further selected to compare the database quality of different TP densities. Another 24 and 32 TPs are selected respectively in two rooms, R1 and R2, so that the TP density in the rooms are 1m×1m. A local database is generated for each room based on this set of TPs. R1 is a small meeting room with no obstructions and simple furnishing. R2 is a heavily obstructed store room with metal shelves and electronic equipment. ΔRSS is the difference between the RSS of fingerprints from the two different databases at the same location. The difference for each AP is listed in Table 1.

A larger ΔRSS is seen in R2 which is the heavily obstructed room. Therefore, signals are noisier and less predictable in such places. Hence more TPs are required to generate better database. However, the difference for 5GHz signal is smaller. This is due to that it is less able to penetrate obstructions and the signal pattern for different locations are more unique.

Table 1 Mean ΔRSS between fingerprints generated from different TP density (dB)

	AP1		AP2		AP3		AP4	
	a	b	a	b	a	b	a	b
R1	2.65	2.12	3.19	2.78	1.77	3.34	8.92	2.97
R2	10.94	3.77	8.00	7.65	17.68	12.62	8.16	5.89

2.3 Collaborative training

As the indoor wireless environment can alter caused by changes in the wireless hardware, the building structure or even furnishings. Therefore, each fingerprint database must be maintained and updated. Re-training can still be laborious work even when GP is applied. Collaborative database training (cFPDB) is proposed here to save time and also enhance database quality.

cFPDB fundamentally relies on collaborative positioning between a number of mobile users to estimate the reference positions of the TPs and the relationship between the training data. First of all, the basic collaborative positioning algorithm in cFPDB is introduced. Collaborative positioning constrains the measurement error of users by applying a relative ranging constraint. The basic navigation is achieved from inertial measurements and propagated forward based on the dead reckoning model at each step,

$$\begin{bmatrix} \hat{x}_k \\ \hat{y}_k \end{bmatrix} = \begin{bmatrix} \hat{x}_{k-1} + \hat{s}_{(k|k-1)} \cos \hat{\theta}_{(k|k-1)} \\ \hat{y}_{k-1} + \hat{s}_{(k|k-1)} \sin \hat{\theta}_{(k|k-1)} \end{bmatrix} \tag{5}$$

where $[\hat{x}_k, \hat{y}_k]$ is the user position at time k, $\hat{s}_{(k|k-1)}$ is the estimated step length between time $k-1$ and k, $\hat{\theta}_{(k|k-1)}$ is the heading estimation during the step. Low-cost inertial system measurements tend to drift badly after initialisation [23,24]. Therefore, the inertial measurement errors need to be constrained by external measurements, e.g. relative ranging measurements between users that can be obtained from high precision wireless units such as Ultra-wideband sensors [17]. This estimated position serves as the reference location of the measured RSS, i.e. the location of the TPs during dynamic data collection.

Collaborative training is carried out during the collaborative positioning process and this means that only one RSS is collected at a specific location and no knowledge of the signal fluctuation pattern could be obtained initially. While the signal strength could vary up to 20dB or even more at any single location when the equipment is static, this may increase further when the receiver is moving. In conventional training, the fluctuation pattern is captured by extending the training time over hours or even days. In cFPDB, extracting signal features obtained by different users passing previous TPs during different periods helps to capture this pattern.

The ranging measurement r between the users builds a link between the collected training data. Two thresholds, the separation threshold δ_{sep} and integration threshold δ_{int}, is defined to identify three different kinds of relationships between the data. If r is above δ_{sep}, it would be regarded that the users were not in the same area of interest. Their training data will be stored separately and used to generate individual databases. If r is within δ_{sep} but above δ_{int}, the training data would be considered to be within the same area of interest and used to generate the same database. As it is almost impossible for users to pass the exact same locations during collaborative training, TP is expanded into training areas (TA). Any training data that are within a range of δ_{int} would be regarded to reflect the signal pattern for the same TA and adjusted to form fingerprints with knowledge of the signal variation for the TA. Figure 3 shows an indication of the relationship between a TA and fingerprint location. The RSS of the fingerprint is assumed to represent the RSS for the entire TA. These thresholds are set according to the expected correlation between fingerprints, which can be affected by the environment and the stableness of the RSS in nearby training locations. This will not differ much in the same building, hence the same threshold can be applied throughout.

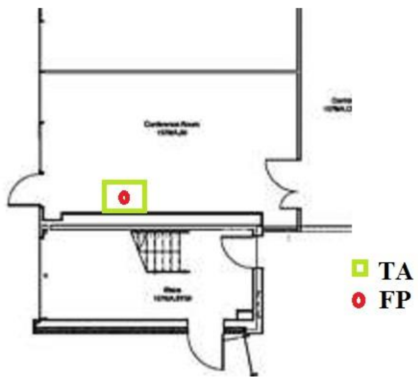
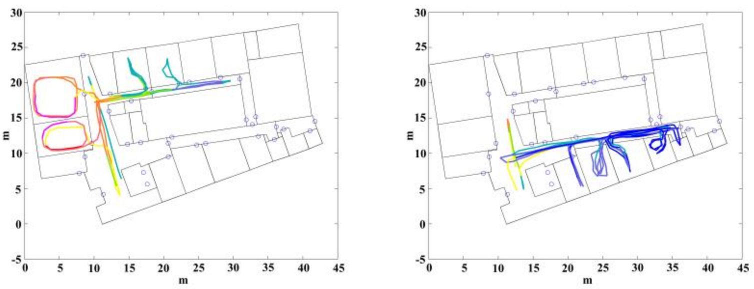
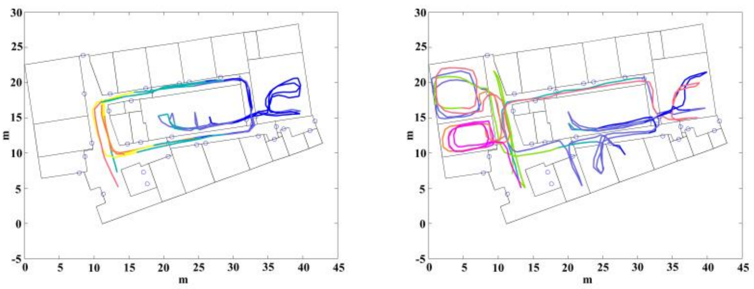


Figure 3 A fingerprint representing the training area (Green grid indicating a training area, red circle is a fingerprint in the TA)

To build the collaborative database dynamically, four different trajectories, denoted as T1, T2, T3 and T4, of varying length and locations within NGB Floor A are chosen where training data will be collected during the collaborative positioning phase. Each user follows one of the different routes and collects training data at each step, shown in Figure 4, where blue indicates low RSS and red indicates high RSS values. The data collected along different trajectories are combined to generate collaborative fingerprint databases using GP, denoted as cDB. Collaborative training greatly extends the training data coverage and increases the amount of data for each TA. cDB fingerprints are generated from more sufficient data and longer time span. Hence captures the RSS fluctuation and environment disturbances.



(a) Collaborative training data of T1 (b) Collaborative training data of T2



(c) Collaborative training data of T3 (d) Collaborative training data of T4

Figure 4 Collaborative training data from AP1a

The RSS of the training data along the dynamic trajectories are compared to the RSS of the TPs of sDB by extracting those that are within a certain distance to the static TPs and measuring the ΔRSS. The RSS difference for distances from 1m up to 4m is listed in Table 2, where ΔRSS indicates the mean difference, Std dev. indicates the standard deviation of the RSS difference. Signal acquisition is less stable while the receiver is moving thus it can be anticipated that the RSS of the dynamic training data are noisier and differs to that of the static TP RSS. From the data listed in Table 2, we can see that even though the ΔRSS between training data is almost 10dB even at 1m, but stays within 15dB up to 3m, which is actually within the RSS fluctuation range. While the difference increases when the distance is 4m, the standard deviation actually drops. This indicates that correlation fails between the two points and their RSS is steadily different as they are too far apart. According to the ΔRSS here, the integration threshold can be set to 2m or 3m depending on the environment, i.e. whether it is more like R1 or R2.

Table 2 Δ*RSS* between dynamic and static TPs (dB)

	1m	2m	3m	4m
ΔRSS	9.85	12.55	13.39	19.36
Std dev.	10.61	10.49	15.91	8.58

Three different cDBs are generated and their fingerprints are compared to those of sDB. cDB1 is generated from the training data along T1 and T2; cDB2 is generated from T1 and T3; cDB3 is generated from T1,T2,T3 and T4. Figure 5 plots the mean RSS difference between the fingerprints of each AP in cDB and sDB. As more data is used to generate the database, the fingerprint RSS of the cDB approaches that of the sDB.

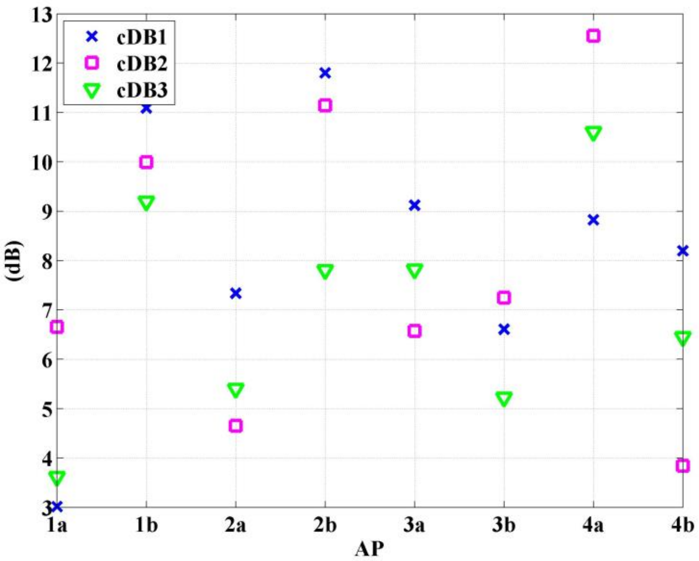


Figure 5 ΔRSS between cDB and sDB (dB)

3 Fingerprinting based on confidence level

3.1 Fingerprint confidence factor

As dynamic training data contain large signal variances, they should be treated appropriately when applied to generate databases. During the cFPDB process, the system keeps track of all previously and currently collected training data by storing them along the timeline. When new training data is picked up at a TA that has been trained previously, the mean of all RSS from all history data is used as the RSS to generate the fingerprint in GP. The standard deviation of RSS is computed to generate a confidence factor for the fingerprint at the location. The confidence factor vector consists of two values, i.e. the training data difference level Δ_{sgn} and the confidence level η_{CF}.

The difference level Δ_{sgn} is updated by measuring the sign of the ΔRSS between the new RSS at the TA and the RSS of the fingerprint representing the TA. A positive Δ_{sgn} indicates that the RSS is increasing and negative Δ_{sgn} indicates decreasing RSS. The confidence level η_{CF} for each fingerprint in the entire building is generated from the training data standard deviation, which indicates how much signal strength variance to expect at each specific fingerprint location. Smaller η_{CF} means higher confidence with the current RSS of the fingerprint stored in the database. There would not be enough training data to compute the standard deviation of fingerprints at the beginning of a training process. Hence it is given a low η_{CF}, assuming that the collected data is trustworthy. The confidence level only decreases when the amount of training data accumulates and shows obvious fluctuation.

If the Δ_{sgn} for a location is always positive or negative, it would be assumed that the RSS is constantly increasing or decreasing, indicating a possibility of permanent change in the Wi-Fi properties at the location. If the confidence level goes over a given threshold under such conditions, the RSS of the old fingerprint will be replaced with new RSS data. η_{CF} for the fingerprint is then reset to the initial value which represents a high confidence level. If Δ_{sgn} changes randomly, the collected RSS is assumed to be within the signal strength random fluctuation range. In such cases, the signal fluctuation range is reflected by η_{CF}. In general, the difference level Δ_{sgn} keeps track of the direction of change of the fingerprint along the time axis while the confidence level η_{CF} reflects the expected signal fluctuation at the location, giving the user an updated knowledge of how trustworthy the fingerprints are. The procedure of generating the confidence factor is shown in Figure 6.

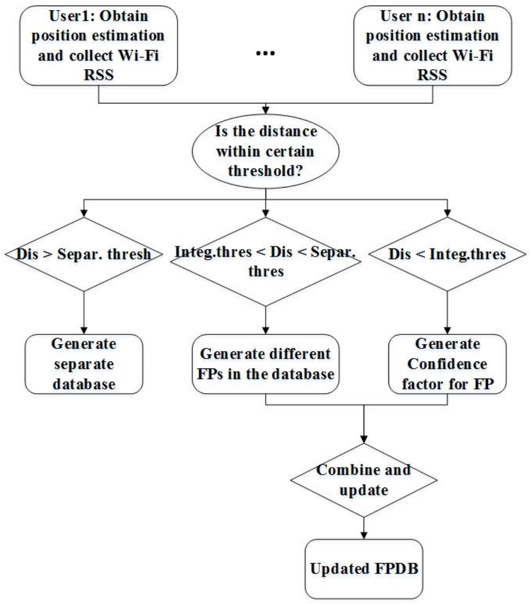


Figure 6 Flowchart for collaborative fingerprint database training

3.2 Collaborative Wi-Fi fingerprinting

In conventional fingerprinting, potential fingerprints are usually found by defining a variance boundary τ_{FP} first. If RSS_P is collected at an unknown location P and RSS_{FP} are the fingerprints from the database, any fingerprints from the database that fit within

$$RSS_P - \tau_{FP} < RSS_{FP} < RSS_P + \tau_{FP} \quad (6)$$

are extracted as potential fingerprints. However, deciding the value of τ_{FP} can be difficult. If the given τ_{FP} is underestimated, there is a possibility that no potential fingerprints will be extracted if either RSS_P or RSS_{FP} is noisy. Yet if τ_{FP} is overestimated, too many potential fingerprints may be found, introducing large location ambiguities.

Fingerprints generated from the cFPDB process take the form of $\{(x_i, y_i)|\mathrm{AP}_1, (\mathrm{RSS}_1, \eta_{CF_1}, \Delta_{sgn}) \cdots, \mathrm{AP}_\mathrm{n}, (\mathrm{RSS}_n, \eta_{CF_n}, \Delta_{sgn})\}$. The confidence factor generated during the training process is used here to help decide the value of τ_{FP}, as below

$$\tau_{FP} = a \cdot \eta_{CF} \tag{7}$$

where a is a coefficient defining the relationship between the two values. It is adjusted from 1.5 to 3 until potential fingerprints are found. From examining the trial data, it has been found that we might choose $a = 1.5$ in open areas and $a = 3$ in heavily obstructed areas.

As the database training is carried out during a collaborative positioning phase, the collaborative measurements can also be applied in fingerprinting when available. Hence the Wi-Fi adaptive collaborative fingerprinting algorithm (WARCP) is proposed. The steps of WARCP are given as below:

1. Each user propagates based on the DR model as in Eq.5;
2. At each step, user i takes a set of Wi-Fi RSS measurement RSS_P^i from each AP, if more than one user is found, ranging measurements r_{ij} are also obtained between user i and j;
3. RSS_P^i is stored to update the database; fingerprinting is then performed by considering both the confidence factor and the distance between the potential fingerprints following Eq.6. When M and N potential fingerprints are found for user i and j , the distance between pairs of potential fingerprints are measured,

$$dist_{FP} = \sqrt{(x_{FP_m} - x_{FP_n})^2 + (y_{FP_m} - y_{FP_n})^2}\ ,(m = 1,2,\cdots, M; n = 1,2,\cdots, N) \tag{8}$$

Fingerprints that obey Eq.9 will remain as potential fingerprints,

$$|dist_{FP} - r_{ij}| \leq \varepsilon_{range} \tag{9}$$

Where ε_{range} is defined based on the expected noise of the ranging measurement.

4. Position estimations are obtained from the weighted average of the remaining potential fingerprint positions.

Fingerprinting reliability is improved here as potential fingerprints are selected according to Eq.6 where τ_{FP} changes adaptively. Therefore, a fingerprint with high confidence level, i.e. small η_{CF}, would also be given a small τ_{FP}. It would not be chosen as a potential fingerprint unless its RSS_{FP} is reliable and close to RSS_P. If a fingerprint's confidence level is low, its possibility of being selected as potential fingerprint is increased as the range of $RSS_{FP} \pm \tau_{FP}$ is larger. This is to decrease its possibility of being discarded when it differs from RSS_P due to fluctuation, but its location is actually close to the true location.

4 Simulations and trials

4.1 Dynamic training

To examine how data is integrated to update the database, the training data of T4 is collected in two rounds. The first round in the building is part 1(P1) and the second round part 2 (P2). The training data of P1 and P2 are used to generate two individual databases, P1-DB and P2-DB. The combination of P1 and P2 is used to train for another database, T4-DB. The difference in dB for the fingerprints of each database and the fingerprints of sDB is measured and plotted in Figure 7.

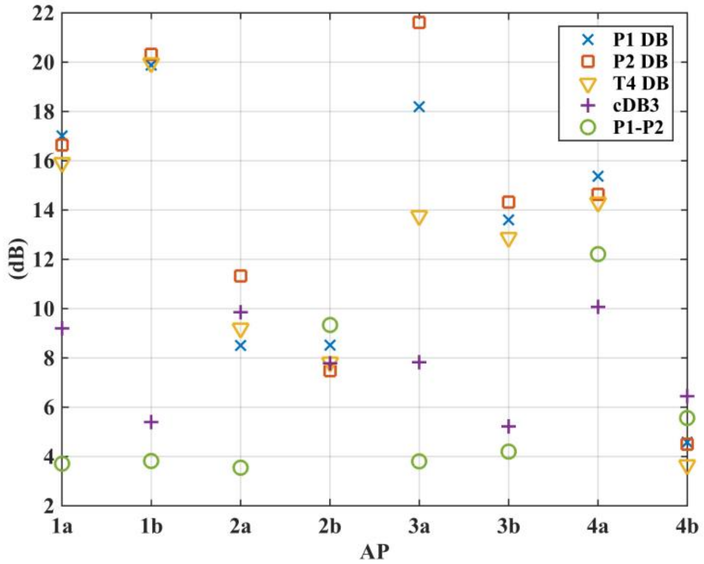
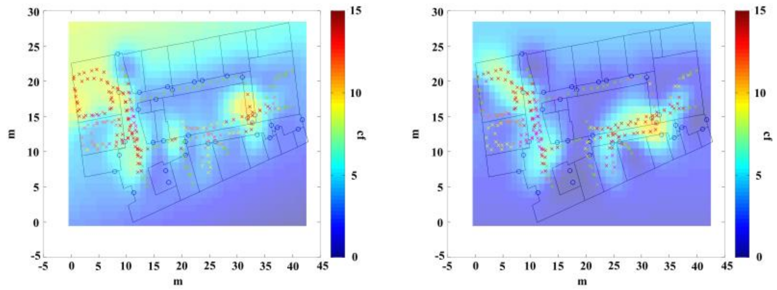


Figure 7 ΔRSS between T4-DB and sDB

In most cases, the difference between the dynamic trained database and sDB is reduced when P1 and P2 data is combined together. The difference is further reduced when more data is integrated with T4 to form cDB3. The difference between the fingerprints generated from P1 and P2 is indicated by green circles. However, there are still instances when the difference of RSS is continuously different from each other, resulting in a large difference from sDB, e.g. AP2b. Another instance is AP1b, where the difference between P1 and P2 is not very large, but because both are very different from sDB, their combined data still results in a large bias, as indicated by the yellow ∇. During the training process itself, it is hard to decide which data is biased or not. Hence we can only record the variance of all collected data and indicate its likelihood of being at a certain signal strength level.

To build up the collaborative database cDB3, new training data is stored and compared to old data iteratively. Each time a new data is collected at a TA where data has been collected previously, the variance of the signal strength is measured and applied to generate the confidence factor as described in Section 3.1. Figure 8 plots the confidence level of AP3 for Floor A that is derived by updating the database from T4-P1 with T4-P2, T1 and T2. Blue areas indicate a small η_{CF}, i.e. high confidence in the fingerprint, and red areas vice versa.



(a) Confidence level map for AP3a (b) Confidence level map for AP3b

Figure 8 Fingerprint confidence level map

The RSS of the training data is also plotted on the map for reference. The resulting η_{CF} is higher in areas where the training data changes rapidly between each training. Furthermore, the confidence indicator for 5GHz signals is smaller in general than that of 2.4GHz signals. The RSS of the 5GHz signals remain relatively stable for different regions of the building hence the fingerprint pattern is more unique, producing lowerη_{CF}. 2.4GHz signals, on the other hand, have greater ranging distance and penetrate walls better. However, this causes noisier training data and higher η_{CF}.

Three different τ_{FP} are chosen to extract potential fingerprints based on the RSS measurements observed at a given location. Figure 9 shows the potential fingerprints extracted when different thresholds are given, $\tau_{FP} = 5$, $\tau_{FP} = 10$ or $\tau_{FP} = a \cdot \eta_{CF}$. As shown in the figure, too many fingerprints are extracted when $\tau_{FP} = 10$. Even though the fingerprints close to the true location are extracted, but those that are almost 10m away are also considered as potentials. When $\tau_{FP} = 5$, not enough fingerprints that are close to the true location are found. When τ_{FP} is set adaptively according to η_{CF}, the potential fingerprints are more suitable as all extracted fingerprints are located near the true position. Table 3 lists the average distance from selected potential fingerprints to the true location throughout a whole trajectory when given different τ_{FP} and also comparing the results for different frequencies. While 5GHz signal fingerprints are slightly closer to the true location, the best result is still achieved when both frequencies are used.

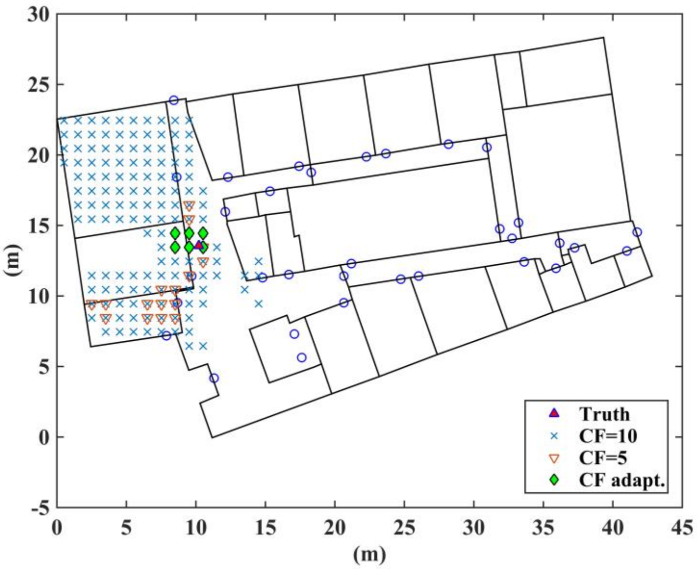


Figure 9 Potential fingerprints extracted based on different τ_{FP}

Table 3 Fingerprinting error for different τ_{FP} (m)

τ_{FP}	5dB	10dB	$a \cdot \eta_{CF}$		
			Dual	2.4GHz	5GHz
Error	16.48	15.51	9.07	11.37	9.69

4.2 Collaborative Fingerprint positioning

To evaluate the performance of the fingerprinting method based on the improved fingerprint database, an indoor positioning trial is carried out in NGB with two rovers starting from the same point, indicated by the red star in Figure 10. Both rovers wear a foot-mounted IMU to obtain inertial measurements and carry a laptop to collect the Wi-Fi RSS. Relative ranging measurements are simulated based on the indoor performance of UWB units so that the mean is the true distance with a standard deviation of 3m. To enhance the effectiveness of the constraint provided by relative ranging, both rovers start at the same place but travel in different directions so they do not follow each other.

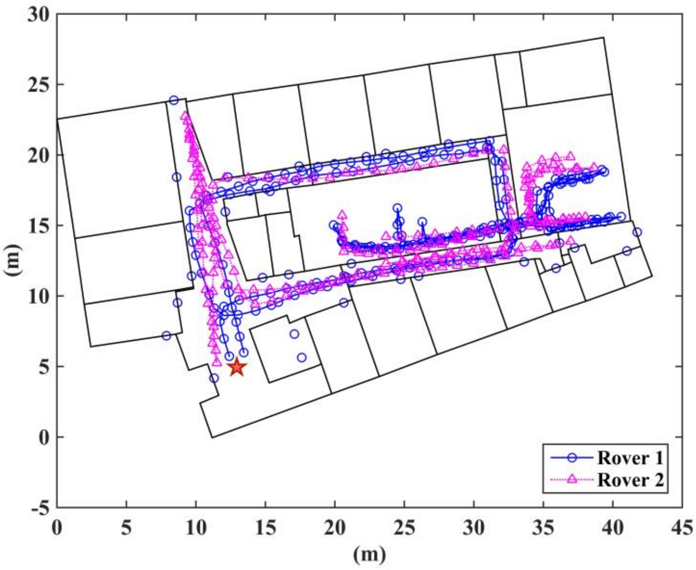
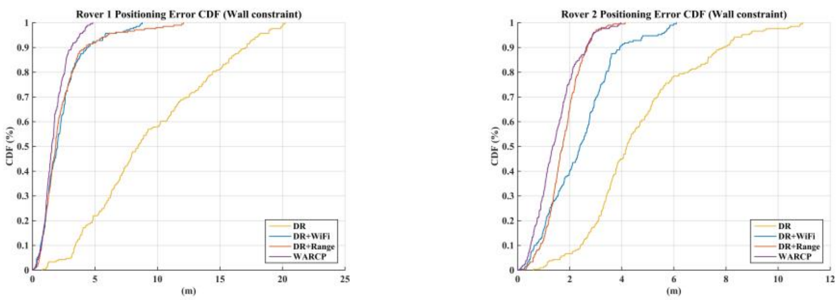


Figure 10 Trajectory for Rover 1 and Rover 2

The collected data is used to compare the performance of different positioning algorithms by processing the data using the data from different sensors each time respectively. DR indicates positioning achieved from just inertial measurements. DR+Wi-Fi indicates the result of integrating DR and fingerprinting with confidence factor. DR+range indicates the result achieved by correcting inertial measurements with ranging measurements. WARCP indicates the result of integrating fingerprinting with ranging and confidence level as introduced in Section 3.2. The cumulative distribution functions (cdf) of the positioning errors of the two rovers are plotted in Figure 11.



(a) Positioning error cdf of Rover 1 (b) Positioning error cdf of Rover 2

Figure 11 Positioning error cdf for each different algorithm

4.3 Results and analysis

The mean positioning error of each algorithm for both rovers is listed in Table 4, where DR/Wi-Fi indicates results of DR integrated with conventional Wi-Fi fingerprinting, DR/Wi-Fi (cf) indicates results of DR integrated with improved Wi-Fi fingerprinting with confidence factor. Although four APs used for Wi-Fi positioning here seems like a very sparse network for fingerprinting, but the realistic situation is that inside most buildings, the number of APs are only setup to ensure network coverage and not fitted to meet the density requirement for fingerprinting. Results indicate that the accuracy of fingerprinting with confidence factor improves by 36% for Rover 1 and 50% for Rover 2 compared to conventional fingerprinting with very few available APs. The performance of integrating DR with improved Wi-Fi fingerprinting is similar to that of DR integrated with ranging. While Wi-Fi fingerprinting gave slightly better results for Rover 1, ranging results were better for Rover 2. Wi-Fi fingerprinting results are not given alone because with only four APs, the performance is very unstable. Although accuracy can be quite good during some periods, but the robustness is too low for comparison. Furthermore, as Table 4 shows, integrating improved Wi-Fi with DR enhance accuracy compared to conventional Wi-Fi with DR. This clearly shows the improvements on fingerprinting with the improved method.

Table 4 Positioning error of different algorithms (m)

	DR/Wi-Fi	DR/Wi-Fi(cf)	DR/Ranging	WARCP
Rover 1	3.67	2.32	2.35	1.76
Rover 2	5.05	2.40	1.76	1.47

The integration of DR, fingerprinting and ranging, i.e. WARCP, improves positioning accuracy further as the mean error is reduced by around 10% compared to DR/Wi-Fi and DR/ranging integration method. Although the average performance difference between the three integrations is not immense, but the maximum error reduces by 35% when WARCP is implemented. As the accuracy level when integrating DR with Wi-Fi and ranging was around the same level, thus the improvement seen here is once more improvement on fingerprinting. While this study focus on the overall improvement of positioning accuracy when integrating low accuracy inertial measurement with Wi-Fi fingerprinting and collaborative ranging, this system shows more freedom of choosing the appropriate positioning algorithm based on what sensor and measurement is available.

As the indoor environment is complicated and prone to change, the WARCP algorithm allows users start its own navigation by using only the inertial measurement from a mobile device (i.e. DR). As more nearby users are found, DR/Ranging can be applied to enhance positioning accuracy by constraining the inertial drift. While collaborative positioning is performed, users can help to train for a Wi-Fi fingerprint database through the collaborative training process which can be stored on to a central server and shared to all local users. When users lose the relative constraint from nearby users, collaborative positioning can no longer be applied, but fingerprinting can be performed when the database is available, which achieves almost the same accuracy level as collaborative positioning. When both Wi-Fi signals and relative ranging measurements are available to the user, the collaborative fingerprinting phase of WARCP can be applied to obtain positioning estimation, which is the most accurate.

WARCP allows the system to search for different choice of integration when different signals and measurements are available. Integrating Wi-Fi and relative ranging with inertial navigation not only enhance the positioning accuracy, but most importantly, the positioning robustness is improved as positioning estimation can be obtained even in changing environments where signals are intermittently available.

5 Conclusions

Wi-Fi fingerprinting is a popular method for indoor positioning as Wi-Fi signals are widely available in most urban areas and infrastructures are already well established. However, Wi-Fi signals are not positioning dedicated, hence suffer instability and disturbance from the changing environment and obstructions, which can cause instability in positioning accuracy. Inertial measurements from mobile devices are useful to indoor positioning users as they are available regardless of the environment. But due to the large heading drift of low-cost inertial sensors, errors must be constrained by external measurements to achieve reasonable positioning. This paper presents an improved Wi-Fi fingerprinting method for both phases of fingerprinting which is fundamentally based on indoor pedestrian inertial navigation but also enhance inertial navigation performance, as described in cFPDB and WARCP.

During the training phase, cFPDB is applied where the RSS measurements from a number of mobile users are collected during different periods and different locations. Measurements are either sorted into training data for different TAs depending on the distance between the location of the data, where fingerprints are generated and updated based on the data of each TA. When updated with new measurements, the fingerprints in the database are given a confidence factor which indicates both the long-term change direction of the RSS and the expected short-term signal fluctuation at the location of the fingerprint. The positioning procedure is then carried out using WARCP, which allows the system to navigate using available measurement flexibly. Potential fingerprints are extracted based on the confidence factor associated with each fingerprint and then further selected by ranging measurements when available.

With the proposed methods, fingerprint database can be setup during the positioning phase when users enter an environment without prior database. Previous databases can also be updated by gathering information from surrounding users. Both history data and new data are applied to update the database so users not only know the current RSS of the fingerprints but also have an idea of how much signal variance to expect at each location. Therefore, during the positioning phase, fingerprints are selected based on whether the current measurement lies within the RSS range of the fingerprint.

Fingerprint based positioning is improved by 40% compared to conventional fingerprinting when the confidence factor is considered. By applying WARCP, which includes ranging measurement constraint, positioning error is further reduced, especially the maximum error which is reduced by 35%. The application of integrating ranging measurements with fingerprinting during training and positioning gives the user more freedom of choosing positioning algorithms based on what information is available. The training effort of fingerprint database is also greatly reduced as training data can be obtained from crowdsourcing.

Reference

[1] D. Niculescu, B. Nath, Position and orientation in ad hoc networks, Ad Hoc Networks 2 (2) (2004) 133–151. doi: 10.1016/S1570-8705(03)00051-9

[2] R. Harle, A Survey of Indoor Inertial Positioning Systems for Pedestrians, IEEE Communications Surveys and Tutorials 15 (3) (2013) 1281–1293. doi:10.1109/SURV.2012.121912.00075.

[3] F. Evennou, F. Marx, Advanced Integration of WiFi and Inertial Navigation Systems for Indoor Mobile Positioning, EURASIP Journal on Advances in Signal Processing 2006 (2006) 1–12. doi:10.1155/ASP/2006/86706.

[4] E. Mok, B. Cheung, An Improved Neural Network Training Algorithm for Wi-Fi Fingerprinting Positioning, ISPRS International Journal of GeoInformation 2 (3) (2013) 854–868. doi:10.3390/ijgi2030854.

[5] K. Kaemarungsi, P. Krishnamurthy, Analysis of WLANs received signal strength indication for indoor location fingerprinting, Pervasive and Mobile Computing 8 (2) (2012) 292–316. doi:10.1016/j.pmcj.2011.09.003.

[6] E. Mok, G. Retscher, Location determination using WiFi fingerprinting versus WiFi trilateration, Journal of Location Based Services 1 (2) (2007) 145–159. doi:10.1080/17489720701781905.

[7] H. Leppakoski, S. Tikkinen, J. Takala, Optimizing radio map for WLAN fingerprinting, in: Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS), 14-15 October 2010, IEEE, Kirkkonummi, 2010, pp. 1–8. doi:10.1109/UPINLBS.2010.5654332.

[8] S. Bak, S. Jeon, Y.-J. Suh, C. Yu, D. Han, Characteristics of a large-scale WiFi radiomap and their implications in indoor localization, IEEE, 2013, pp. 1–5. doi:10.1109/NOF.2013.6724506.

[9] B. Viel, M. Asplund, Why is Fingerprint-based indoor localization still so hard?, in: Pervasive Computing and Communications Workshops (PERCOM Workshops), 2014 IEEE International Conference on, 24-28 March 2014, Budapest, 2014.

[10] B. Ferris, D. Hahnel, D. Fox, Gaussian Processes for Signal Strength-Based Location Estimation, in: In Proc. of Robotics Science and Systems, 2006, Philadelphia, Pennsylvania, 2006.

[11] R. Faragher, R. Harle, SmartSLAM - An Efficient Smartphone Indoor Positioning System Exploiting Machine Learning and Opportunistic Sensing, in: ION GNSS+ 2013, Proceedings of the 26th International Technical Meeting of The Satellite Division of the Institute of Navigation, 16-20 September 2013, ION, Nashville, TN, 2013, pp. 1006–1019.

[12] P. Robertson, M. G. Puyol, M. Angermann, Collaborative Pedestrian Mapping of Buildings Using Inertial Sensors and FootSLAM, in: Proceedings of the 24th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2011), September 20 - 23, 2011, Portland, OR, 2011, pp. 1366 – 1377.

[13] J. Krumm, J. Platt. Minimizing Calibration Effort for An Indoor 802.11 Device Location Measurement System, Microsoft Research Technical Report, November, 2003.

[14] Y. Zhuang, Z. Syed, J. Georgy, N. El-Sheimy, Autonomous smartphone-based WiFi positioning system by using access points localization and crowdsourcing, Pervasive and Mobile Computing. doi:10.1016/j.pmcj.2015.02.001.

[15] Y. Zhuang, Z. Shen, Z. Syed, J. Georgy, H. Syed, N. El-Sheimy. Autonomous WLAN Heading and Position for Smartphones, in: Position, Location and Navigation Symposium - PLANS 2014, 5-8 May 2014, IEEE/ION, pp.1113-1121. doi: 10.1109/PLANS.2014.6851481.

[16] L.-W. Chan, J.-R. Chiang, Y.-C. Chen, C.-N. Ke, J. Hsu, H.-H. Chu, Collaborative Localization: Enhancing WiFi-Based Position Estimation with Neighborhood Links in Clusters, in: Pervasive Computing: Proceedings of 4th International Conference, Pervasive, Dublin, Ireland, 7-10 May 2006, Vol. 3968, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 50–66.

[17] H. Jing, J. Pinchin, C. Hill, T. Moore, A novel weighting approach to collaborative indoor positioning, in: Proceeding of European Navigation Conference (ENC-GNSS 2014), Rotterdam, 2014.

[18] J.-G. Park, B. Charrow, D. Curtis, J. Battat, E. Minkov, J. Hicks, S. Teller, J. Ledlie. Growing An Organic Indoor Location System, in: Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services (MobiSys '10), June 2010. San Francisco, CA, pp. 271-284. doi: 10.1145/1814433.1814461.

[19] P. Bolliger, K. Partridge, M. Chu, M. Langheinrich. Improving Location Fingerprinting Through Motion Detection and Asynchronous Interval Labeling, in: Location and Context Awareness, 4th International Symposium, LoCA 2009, Tokyo, Japan, May 7-8, 2009, Springer Berlin Heidelberg, pp. 37-51. doi: 10.1007/978-3-642-01721-6_3.

[20] B. Shin, J. H. Lee, T. Lee, H. S. Kim, Enhanced weighted K-nearest neighbor algorithm for indoor Wi-Fi positioning systems, in: Computing Technology and Information Management (ICCM), 2012 8th International Conference on, Vol. 2, 2012, pp. 574–577.

[21] W. Liu, Y. Chen, Y. Xiong, L. Sun, H. Zhu, Optimization of Sampling Cell Size for Fingerprint Positioning, International Journal of Distributed Sensor Networks 2014 (2014) 1–6. doi:10.1155/2014/273801.

[22] C. Rasmussen, C. Williams, Gaussian processes for machine learning, Adaptive Computation and Machine Learning, MIT Press, Cambridge, Mass, 2006.

[23] F. Weimann, G. Abwerzger, B. Hofmann-Wellenhof, A Pedestrian Navigation System for Urban and Indoor Environments, in: Proceedings of the 20th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2007), September 25 - 28, 2007, ION, Fort Worth, TX, 2007, pp. 1380 – 1389.

[24] K. Abdulrahim, C. Hide, T. Moore, C. Hill, Aiding Low Cost Inertial Navigation with Building Heading for Pedestrian Navigation, Journal of Navigation 64 (02) (2011) 219–233. doi:10.1017/S0373463310000573

Tables

Table 1 Mean ΔRSS between fingerprints generated from different TP density (dB)

	AP1		AP2		AP3		AP4	
	a	b	a	b	a	b	a	b
R1	2.65	2.12	3.19	2.78	1.77	3.34	8.92	2.97
R2	10.94	3.77	8.00	7.65	17.68	12.62	8.16	5.89

Table 2 ΔRSS between dynamic and static TPs (dB)

	1m	2m	3m	4m
ΔRSS	9.85	12.55	13.39	19.36
Std dev.	10.61	10.49	15.91	8.58

Table 3 Fingerprinting error for different τ_{FP} (m)

τ_{FP}	5dB	10dB	$a \cdot \eta_{CF}$		
			Dual	2.4GHz	5GHz
Error	16.48	15.51	9.07	11.37	9.69

Table 4 Positioning error of different algorithms (m)

	DR/Wi-Fi	DR/Wi-Fi(cf)	DR/Ranging	WARCP
Rover 1	3.67	2.32	2.35	1.76
Rover 2	5.05	2.40	1.76	1.47