# Construction and Management of Fingerprint Database with Estimated Reference Locations for WiFi Indoor Positioning Systems

Myat Hsu Aung, Hiroshi Tsutsui, and Yoshikazu Miyanaga
Graduate School of Information Science and Technology, Hokkaido University
Kita 14, Nishi 9, Kita-ku, Sapporo, Hokkaido 060-0814, Japan

**ABSTRACT**

In this paper, we propose a positioning system for indoor environments based on WiFi fingerprint methods. In fingerprint methods, the database construction is important for the position estimation performance. We created the fingerprint database by gathering pairs of media access control (MAC) addresses and received signal strength indicator (RSSI) at each estimate reference location. In this proposed method, reference devices are moving at a constant speed with a simple direction that can be estimated the location of each reference point. The multiple data sets are collected in same areas to create the fingerprint database. The location of reference points are slightly different with each other. We use the mean-shift clustering algorithm to get merged reference points from the multiple data sets. We manage to update the database using the user's input data in different time duration. We created four types of fingerprint database and evaluated the user's position estimation. The results show that the user's position can be estimated with an average accuracy error of 1.8 m.

**Keywords**: WiFi indoor positioning, Fingerprint, Received signal strength (RSS), and Mean-shift clustering

## 1. INTRODUCTION

Recently, positioning systems have become indispensable not only for navigation and tracking but also for various location-based applications. The global positioning system (GPS) is most popular and suitable technology for outdoor positioning system. However, the large error occurs in indoor environment because the GPS signal becomes weak when they penetrate the construction materials [1]–[4]. The accuracy of indoor positioning remains the greatest challenges in real-time applications. Also, a flexible and low-cost indoor positioning system is highly demanded. For indoor positioning systems (IPS), RF signals from wireless devices such as WiFi, Bluetooth, and radio frequency identification (RFID) devices can be utilized to estimate mobile device positions [5]–[8]. Among them, we focus on the most cost-effective method that is WiFi-based IPS since the WiFi coverage is getting higher due to the significantly increasing number of private or public WiFi access points in metropolitan areas.

There are various position algorithms in WiFi-based IPS such as trilateration and fingerprinting that are using measurement signal such as received signal strength (RSS), time of arrival (TOA), and time difference of arrival (TDOA) from WiFi devices [6, 9]. In our previous work [10], we proposed a fingerprint method based IPS using estimated reference locations. In case of fingerprint based IPS, the information of access points (APs) reachable from the user's location is used for estimating the user's location by comparing it with the pre-stored data in the database. Such user's information is called a fingerprint. As for the database, since we need to store the information of APs reachable from a lot of known reference locations, this database creation requires a significant cost. Focusing on the difficulty of database creation, we are trying to develop a method to create the database of reference point information which does not require precise reference point locations.

In our previous paper, the propose method when the fingerprint database is created using received signal strength indicator (RSSI) values, there are some problems in accuracy to estimate the device position. Because RSSI values fluctuate with various effects such as spatial and temporal variations of interference, hardware variations, and environmental effects such as human presences [11]. Another difficulty is that the database depends on the APs, which means new installations and replacements of APs have a significant impact on the estimation accuracy. Considering these issues, not only the database creation but also how to manage the database after the database creation is essential.

In this paper, we focus on the database creation of WiFi based IPS using estimated reference locations. We are trying to create the fingerprint database with multiple data sets from two devices that contain the different sampling points in different data collection day. The multiple data sets are merged to create the single database. In the proposed method, we use the mean-shift clustering algorithm [12] to get merged reference points from the multiple data sets. We present the accuracy evaluation of the proposed system with different types of database which are constructed by all data set of each day and each device, and all data set of both device from different data collection that includes and excludes the testing data.

## 2. PROPOSED FINGERPRINT METHOD

The proposed approach estimates user locations based on fingerprint methods utilizing WiFi signal strength measure-

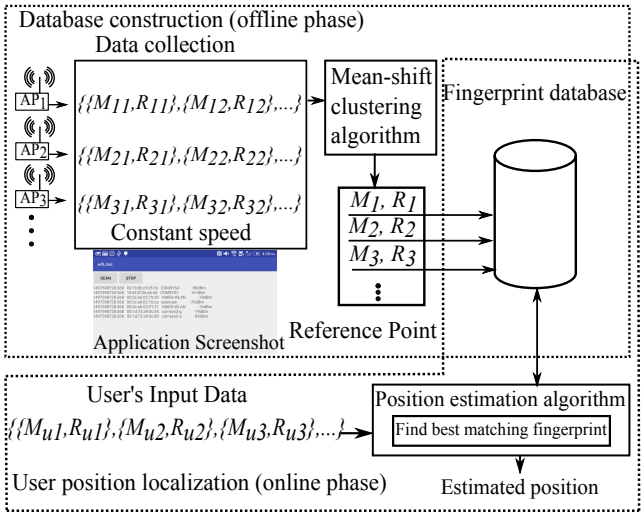



Fig. 1: The overview of the proposed system.

ment. Figure 1 shows the overview of the proposed system. The total system consists of a database construction part (offline phase) and a user position localization part (online phase). In the offline phase, RSSI values from available APs are collected at each specific position in target area. Such positions are called *reference points*. The collected RSSI values for each reference point is stored in a fingerprint database. In the online phase, the real-time position of a user is estimated using the input set of current RSSI values that are collected by the user's mobile device and compared with those in the fingerprint database using location estimation algorithm.

### Construction of fingerprint database

First using WiFi scanning Android application, the training data for database construction are collected in the target area that include pairs of media access control (MAC) addresses and RSSI values which can be obtained from reachable APs for each reference point. These MAC-RSSI pairs are gathered every specific period such as one second with walking at a constant speed to estimate the actual location of each reference point. The absolute timestamps for each pair is also collected. The locations where MAC-RSSI pairs are sampled can be regarded as reference points for each data set. As a result, a set of pairs of MAC addresses and RSSI values for each reference point is stored as a fingerprint in the database. Let the set of MAC addresses $M_i = \{M_{i1}, M_{i2}, \ldots, M_{iN_i}\}$ for reference point $i$, where the number of access points available at reference point $i$ is given by $N_i$. The RSSI values paired with $M_i$ is denoted by $R_i = \{R_{i1}, R_{i2}, \ldots, R_{iN_i}\}$. We denote the set of MAC-RSSI pairs for reference point $i$ by $P_i = \big(M_i, R_i\big)$.

In this proposed approach, multiple data sets are used to create the database that are collected same area but different sampling rate due to speed change. The multiple data sets are merged using mean-shift clustering algorithm. The list of MAC-RSSI pairs for the reference point $i$ from multiple data sets can get using mean-shift clustering algorithm. RSSI values for each merged reference point $i'$ are calculated by averaging RSSI values of the reference points in the corresponding mean-shifted cluster. Let us denote the set of MAC-RSSI pairs for merged reference point $i'$ by $P_{i'} = \big(M_{i'}, R_{i'}\big)$. The list of the set of merged reference points $P_{i'}$ is stored in the database as $P$. Note that the number of merged reference points $i'$ depends on the thresholds parameter $r$ to calculate means in the mean-shift algorithm.

### Location Estimation

In the following, we explain our location estimation approach. This approach is same as our previous paper [10]. In the user position localization part, user's mobile device samples the MAC address and the RSSI value of each AP available from the current user's position. This sample can be denoted by using the similar notation as follows,

- the set of MAC addresses at the user's position $u$: $M_u = \{M_{u1}, M_{u2}, \ldots, M_{uN_u}\}$, where $N_u$ is the number of available access points,
- the RSSI values paired with $M_u$: $R_u = \{R_{u1}, R_{u2}, \ldots, R_{uN}\}$, and,
- the set of MAC-RSSI pairs at the user's position: $P_u = \big(M_u, R_u\big)$.

This $P_u$ is the input of the position estimation algorithm and the algorithm estimates user's location by using $P_u$ and $P$. Note that $P$ is the database. The procedure of the user's position estimation is the following.

(1) Pick up reference points which include at least one MAC address of user's input and create the set of such reference points. This set is described by

$$S = \big\{i \big| M_u \cap M_i \neq \varnothing\big\}. \tag{1}$$

(2) Create the list of unique MAC addresses included the above set of reference points. which is related to the input data.

$$M' = \bigcup_{i \in S} M_i \tag{2}$$

(3) Create location vector, that is, fingerprint, for user's input and references points selected in step (1). The location vector is given as the RSSI values list for corresponding MAC address list created in step (2). If no RSSI value is available for a MAC address, the element is set $\varnothing$.

$$V_u = \{V_{uj}\},\ V_{uj} = \begin{cases} R_{uj} & \text{if } M'_j \in M_u \\ \varnothing & \text{otherwise} \end{cases} \tag{3}$$

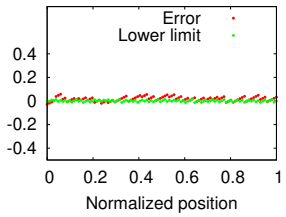
where $M'_j \in M'$. Similarly, as for selected references points, the location vectors are created by

$$V_i = \{V_{ij}\},\ V_{ij} = \begin{cases} R_{ij} & \text{if } M'_j \in M_i \\ \varnothing & \text{otherwise} \end{cases} \tag{4}$$

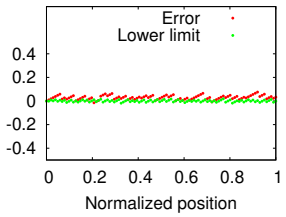
where $M'_j \in M'$ and $i \in S$.

(4) Calculate vector distances $D_{ui}$ between user's input vector $V_u$ and all reference vectors $V_i$.

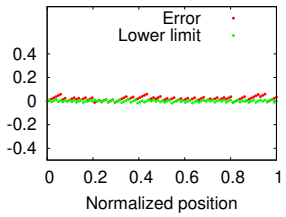
(5) Output the position $\hat{u} = \operatorname{argmin}_i D_{ui}$ as the estimated user's position.



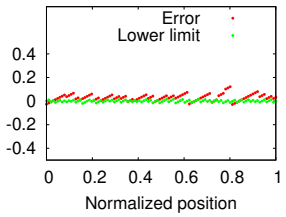

(a) Condition A.




(b) Condition B.

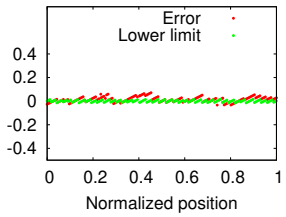



(c) Condition C.

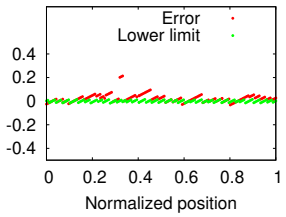



(d) Condition D.

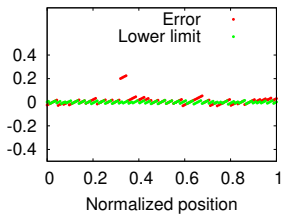
Fig. 2: Evaluation 1 results.



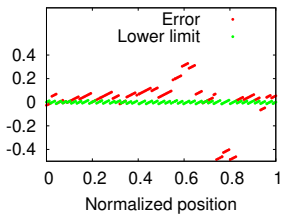

(a) Condition A.




(b) Condition B.




(c) Condition C.




(d) Condition D.

Fig. 3: Evaluation 2 results.

## 3. Experimental results

In our previous research, we developed an Android application which gathers MAC-RSSI pairs of available APs every specific period aiming database construction and location estimation algorithm development. By using this application, data were collected on the 11th floor of Graduate School of Information and Science Technology Building, Hokkaido University. In this experiment, we used two Android devices, (A) HTC One (M7) and (B) Nexus 7 (2013). The data is gathered by walking at constant slow speed (slow 1 and slow 2), a faster speed, and in the reverse direction. Considering the effect of APs which are replaced new one, we collected the information of the reachable APs in reference point in different two days. We obtained 4 data sets that are slow 1, slow 2, fast and reverse for each device and each day, resulting 16 data sets in total. Note that the shape of the walking path is a rectangle and that the starting point and the end point is same. The walking distance is about 120 m.

In the fingerprint database creation, all data sets are merged using mean-shift clustering algorithm to create the database. The number of samples are different with each other due to speed change. Therefore, timestamp normalization is required. Assuming that the number of samples is $N_k$ in an obtained data set and that the timestamp of each sample is given by $t_k$, $k = 0, 1, \ldots, N-1$, the normalized timestamp is given by,

$$L_k = \frac{t_k - t_0}{t_{N_k-1} - t_0}, k = 0, 1, \ldots, N_k - 1. \quad (5)$$

Note that $L_0 = 0$, $L_{N_k-1} = 1$, and 1 corresponds to 120 m.

$L_k$ is regarded as the estimated reference point location. The user's position is estimated by the WiFi-based positioning approach based on these estimated reference points.

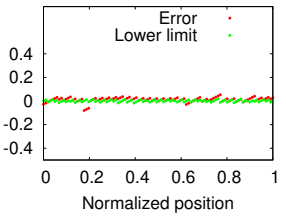
We created four types of database to test data sets. We denote these types as Conditions A, B, C, and D as listed below.

Cond. A uses all data sets including a test input data set,
Cond. B uses all data sets excluding a test input data set,
Cond. C uses data sets from same device for test, and
Cond. D uses data sets from the other device for test.

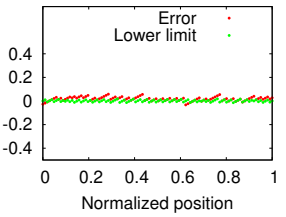
We denote three types of evaluation in Figs. 2, 3, and 4 as listed below.

Evaluation 1 uses 8 data sets including a test input data set in the day 1,
Evaluation 2 uses 8 data sets except one used as a test input data set in day 2,
Evaluation 3 uses all 16 data sets including test input data, and

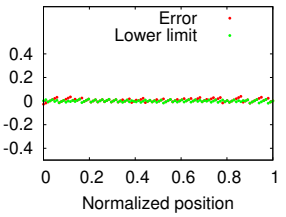
The types of the databases and their accuracy evaluation results are shown in Table I. In this table, the data set number 1 to 8



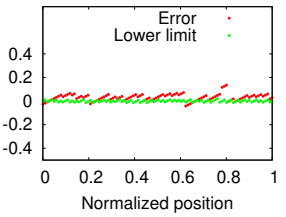

(a) Condition A.




(b) Condition B.




(c) Condition C.




(d) Condition D.

Fig. 4: Evaluation 3 results.

TABLE I: Database types and their accuracy evaluation results.

| | | | | | | | | | | | | | | | | | | Normalized error | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Evaluation | Condition | Data sets used to create the database | | | | | | | | | | | | | | | | Ave error | Max error |
| 1 | Cond. A | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | | | | | | | | | 0.018 | 0.147 |
| | Cond. B | 1 | | 3 | 4 | 5 | 6 | 7 | 8 | | | | | | | | | 0.019 | 0.217 |
| | Cond. C | 1 | 2 | 3 | 4 | | | | | | | | | | | | | 0.017 | 0.234 |
| | Cond. D | | | | | 5 | 6 | 7 | 8 | | | | | | | | | 0.033 | 0.175 |
| 2 | Cond. A | | | | | | | | | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 0.017 | 0.093 |
| | Cond. B | | | | | | | | | 9 | | 11 | 12 | 13 | 14 | 15 | 16 | 0.017 | 0.214 |
| | Cond. C | | | | | | | | | 9 | 10 | 11 | 12 | | | | | 0.015 | 0.224 |
| | Cond. D | | | | | | | | | | | | | 13 | 14 | 15 | 16 | 0.043 | 0.486 |
| 3 | Cond. A | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 0.015 | 0.106 |
| | Cond. B | 1 | | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 0.015 | 0.107 |
| | Cond. C | 1 | 2 | 3 | 4 | | | | | 9 | 10 | 11 | 12 | | | | | 0.012 | 0.108 |
| | Cond. D | | | | | 5 | 6 | 7 | 8 | | | | | 13 | 14 | 15 | 16 | 0.035 | 0.208 |

In this table to show the valuations of our evaluations, we assume that data set 2 is used for testing.
The average error values are averaged ones over all tests while the maximum error values are maximum values of all tests.

are colleced in day 1, and 9 to 16 are in day 2. The data set 2 is used for testing for all types of database for exemplifying the conditions in the table. The average error values are averaged over all tests while the maximum values are maximum values of all tests.

Figs. 2, 3, and 4 show the normalized errors of testing data set 2 for different condition of database creation. As for the mean-shift parameter of $r$, we used 0.01 of normalized timestamps considering the number of obtained reference points for each data set. Note that, in these figures, the lower limit errors due to the discrete reference points are also included.

We summarize the averaged and maximum normalized error for each condition as follows. As for Evaluation 1, the normalized error of Condition A (includes the testing data) is smaller than Condition B (excludes the testing data). The average error of Condition D is larger than Condition B due to the different device to testing data. In Evaluation 2, the database is created form the data of different day to the testing data. As we can see from the table, similar results are obtained to those of Evaluation 1. Evaluation 3 results show the accuracy where the combination of data sets from two days data collection is used to create the database. As for Conditions A and B in Evaluation 3, the average error is 0.015 in the normalized error, which corresponds to about 1.8 m. Condition B shows the large maximum error since the database does not contain the user's input data set. In Condition D, it can be seen that the user's position error is larger than the other. This is because the database is created from the different device from that for testing data. As results, the average error can be reduced due to the management of the database creation.

## 4. Conclusion

In this paper, we proposed a positioning system for indoor environments based on WiFi fingerprint method. The fingerprint databases were constructed from different data sets obtained by gathering MAC-RSSI pairs using reference devices moving at a constant speed. To get merged reference points from the multiple data sets, the mean-shift clustering algorithm is used in this paper. Estimation accuracy evaluation results show that the proposed approach can estimate user's location without any precise reference point locations. This proposed approach can be reduced the accuracy error due to the estimation accuracy evaluation results. In the previous work, the accuracy error is large because of the database are constructed using one data set. The user's position can be also estimated or compensated by using embedded inertial sensors in mobile phones, which is listed as one of our future works.

## References


[1] S. Panzieri, F. Pascucci, and G. Ulivi, "An outdoor navigation system using GPS and inertial platform," *IEEE/ASME Transactions on Mechatronics*, vol. 7, no. 2, pp. 134–142, Jun. 2002.

[2] V. Honkavirta, T. Perälä, S. Ali-Löytty, and R. Piché, "A comparative survey of WLAN location fingerprinting methods," in *Proc. Positioning, Navigation and Communication (WPNC)*, Mar. 2009, pp. 243–251.

[3] B. Molina, E. Olivares, C. E. Palau, and M. Esteve, "A multimodal fingerprint-based indoor positioning system for airports," *IEEE Access*, vol. 6, pp. 10 092–10 106, Jan. 2018.

[4] A. Solin, S. Srkk, J. Kannala, and E. Rahtu, "Terrain navigation in the magnetic landscape: Particle filtering for indoor positioning," in *2016 European Navigation Conference (ENC)*, May 2016, pp. 1–9.

[5] W. K. Zegeye, S. B. Amsalu, Y. Astatke, and F. Moazzami, "WiFi RSS fingerprinting indoor localization for mobile devices," in *Proc. UEMCON*, Oct. 2016, pp. 1–6.

[6] M. E. Rusli, M. Ali, N. Jamil, and M. M. Din, "An improved indoor positioning algorithm based on RSSI-trilateration technique for internet of things (IOT)," in *Proc. International Conference on Computer and Communication Engineering*, Jul. 2016, pp. 72–76.

[7] H. Jun-Ho and S. Kyungryong, "An indoor location-based control system using bluetooth beacons for IoT systems," *Sensors*, vol. 17, no. 12, 2017.

[8] D. A. Savochkin, "Simple approach for passive RFID-based trilateration without offline training stage," in *Proc. 2014 IEEE RFID Technology and Applications Conference (RFID-TA)*, Sep 2014, pp. 159–164.

[9] G.Retscher, "Fusion of location fingerprinting and rrilateration based on the example of differential WI-FI positioning," Sep. 2017, pp. 377–384.

[10] Myat Hsu Aung, H. Tsutsui, and Y. Miyanaga, "An accuracy evaluation of WiFi based indoor positioning system using estimated reference locations," in *Proc. International Workshop on SmartInfo-Media Systems in Asia (SISA)*, Sep. 2017, pp. 321–326.

[11] Y. Chapre, P. Mohapatra, S. Jha, and A. Seneviratne, "Received signal strength indicator and its analysis in a typical WLAN system (short paper)," in *Proc. 38th Annual IEEE Conference on Local Computer Networks*, Oct. 2013, pp. 304–307.

[12] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE TIT*, vol. 21, no. 1, pp. 32–40, Jan. 1975.