Bisecting K-Means Based Fingerprint Indoor Localization

Yuxing Chen[1(✉)], Wei Liu[1], Haojie Zhao[1], Shuling Cao[1], Shasha Fu[1], and Dingde Jiang[2]

[1] State Key Labs of ISN, Xidian University,
Xi'an 710071, Shaanxi, People's Republic of China
{yxchen_2,hjzhao,slcao_cn,ssfu}@stu.xidian.edu.cn,
liuweixd@mail.xidian.edu.cn
[2] School of Astronautics and Aeronautic,
University of Electronic Science and Technology of China, Chengdu 611731, China
jiangdd99@sina.com

Abstract. This paper presents a fingerprint indoor localization system based on Bisecting k-means (BKM). Compared to k-means, BKM is a more robust clustering algorithm. Specifically, BKM based indoor localization consists of two stages: offline stage and online positioning stage. In the offline stage, BKM is used to divide all the reference points (RPs) into k clusters. A series of experiments have been made to show that our system can greatly improve localization accuracy.

Keywords: Fingerprint · Bisecting K-means · WiFi · Indoor localization

1 Introduction

With the rapid development of wireless communication technologies and the Internet industry, the demand for LBS (location-based services) is also growing. LBS has developed rapidly and received extensive attention and has been widely used in social networks, advertising services, travel, shopping, public safety services, and emergency assistance [1].

In terms of its application scenario, localization can be distinguished into indoor localization and outdoor localization. GPS is a commonly used outdoor wireless positioning technology, and has been relatively mature. Owing to the fact that GPS signal becomes weak after passing through the building, the satellite positioning cannot give reliable position information [2]. Therefore, traditional outdoor positioning technology cannot be used in indoor environments [3].

The financial support of the program of Key Industry Innovation Chain of Shaanxi Province, China (2017ZDCXL-GY-04-02), of the program of Xi'an Science and Technology Plan (201805029YD7CG13(5)), Shaanxi, China, of National S&T Major Project (No. 2016ZX03001022-003), China, and of Key R&D Program - The Industry Project of Shaanxi (Grant No. 2018GY-017) are gratefully acknowledged.

H. Song et al. (Eds.): SIMUtools 2019, LNICST 295, pp. 1–12, 2019.
https://doi.org/10.1007/978-3-030-32216-8_1

Consequently, wireless indoor positioning technology emerged. The commonly used wireless indoor positioning technologies include: ultrasonic positioning technology, ultra-wideband positioning technology, Bluetooth technology, and WiFi technology. Among them, WiFi is one of the most commonly used wireless communication technologies that covers a wider area, and has the advantages of easy-to-install, low cost, and relatively stable. A variety of terminal devices such as mobile phones, computers, and pads support WiFi communication, so WiFi indoor positioning technology is portable.

Fingerprint-based localization has become one of the most attractive and promising techniques due to its performance of high accuracy and stability [4–6]. The core idea of fingerprinting positioning is to map the location information that is difficult to measure to the characteristics of the radio signal that are easy to measure [7].

In [8], a system based on database partition and Euclidean distance-weighted pearson correlation coefficient is proposed, and this system is the combination of fingerprint database and machine learning. Support Vector Machine (SVM) is also an efficient algorithm that makes a great improvement in localization [9]. A mixture Gaussian distribution model can be used to minimize the error of the measured RSSI data and neural network plays the role in excavating the relationship between RSSI data and the position [10].

In this paper, we propose an improved fingerprinting localization algorithm based on the localization method in [11]. This system consists of two stages: offline stage and online localization stage. In the offline stage, a fingerprint database or a radio map is constructed that stores the relationship between Received Signal Strength Indicator (RSSI) data and Reference Points (RPs). BKM is adopted to divide all the RPs into clusters based on the fingerprint database [12]. In the online stage, RSSI data collected at test points are matched to the database to infer the concrete position.

The rest of our paper is organized as follows. In Sect. 2, a description about the system architecture is presented. In Sect. 3, the concrete approach we adopt in our system is presented. The experiment and result are illustrated in detail in Sect. 4. Finally, we list out our conclusion and look for the future.

2 System Model

In the indoor localization area, multiple Access Point (AP) signal can be detected at each location. With a mobile terminal equipped with wireless network card, we can record the AP MAC address and RSSI data. The AP MAC address and corresponding RSSI data at each location constitute a fingerprint. We can represent and determine a concrete location if we collect adequate fingerprints. Fingerprint indoor localization system is composed of two stages: offline stage and online stage. Offline stage is a procedure that maps locations to fingerprint. We process the raw RSSI data collected at each RP, and build a fingerprint database. In online stage, we sample online RSSI data. Then we compare and match the online data with the fingerprint database to determine the specific location. Figure 1 shows the architecture of our system.

In indoor localization environment, we receive RSSI data at positions from different APs. To make better use of existing APs in the building, we do not install additional APs. We pick out efficient APs, and classify positions into several clusters, then we build a decision tree for each cluster in offline stage. In online positioning stage, we lump the test point with the cluster whose cluster center is nearest to the test point. And then we use decision tree to determine its concrete position.

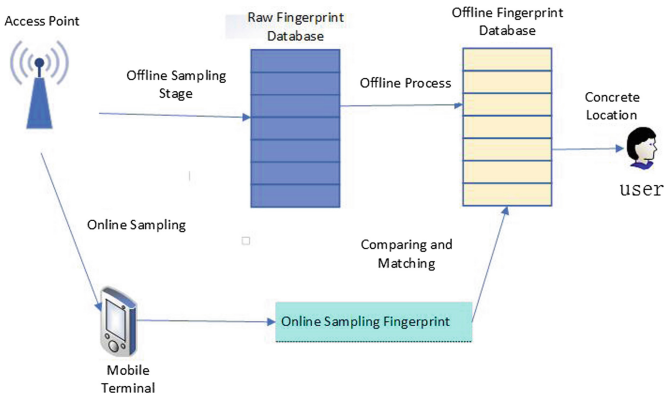

Fig. 1. System architecture.

3 BKM-Based Indoor Localization Approach

Assuming that there are N RPs in the indoor environment and each RP can receive signals from part of APs. The fingerprint database stores the coordinate of each location and the RSSI data from APs, which can be marked as:

$$D_i = \{i, AP_j, \mathrm{RSSI}_{ij}\} \tag{1}$$

In formula (1), i represents the i_{th} RP in the localization area, $(AP_j, \mathrm{RSSI}_{ij})$ means that AP_j can be detected at the i_{th} location, and the RSSI data received at the i_{th} position from AP_j. Each D_i is a sampling fingerprint. Fingerprints can be stored in a database that maps RSSI data to positions.

In order to guarantee the precision of localization, we need to sample a large number of RSSI data at each location. It is clear that more data we have, the longer time it takes to fix the location. So it is necessary for us to take measures to shorten localization time. In this paper, we adopt three methods to reduce computation and space complexity: AP selection, clustering, and decision tree [11,13].

When row RSSI data are collected at each position, we need to select some APs which will increase the localization accuracy. Then RSSI data collected

from these APs are stored as fingerprint database. According to the similarity between different RPs in the database, we divide the RPs into several clusters by clustering method. To guarantee the localization accuracy in a fine grain, we need to build a decision tree for each cluster. Each leaf node of a tree represents a position in the environment. Figure 2 is the offline stage process.

3.1 *AP* Selection

Due to multi-path effect and signal reflection, signal may suffer from path loss, attenuation, and time delay in indoor environment. RSSI data collected from long-distance APs are greatly interfered by noise. And these RSSI data may lead to the decrease in localization accuracy. Furthermore, with more APs, we need to handle more data, which will affect the real-time behavior. Thus, the process of selecting APs is an important approach and guarantee for improving positioning accuracy, and reducing the complexity of positioning algorithms. We adopt maximum information gain to select APs [11].

We take the resolution capability as the evaluation standard, and choose the APs with the strongest resolution capability. That is, after the formulas below being calculated, we select m APs with the maximum information gains and constitute a N by m dimensional fingerprint information database.

For each position G, we can treat RSSI data from AP_i as a feature. In this system, a certain position G_j can be expressed as $(G_j^1, G_j^2, ...G_j^m)$ by those features. And G_j^i indicates the average signal strength from AP_i collected over a period of time as a feature of G_j. If AP_k cannot be detected in G_j, then we set G_j^k the default value -90 dBm.

Once we determine the localization area, we get to know the entropy of the environment. We can get the information entropy with the formula (2) [11]:

$$H(G) = -\sum_{j=1}^{n} P(G_j) \log P(G_j) \tag{2}$$

When we get the RSSI data from AP_i, the entropy will change into [11]:

$$H(G/AP_i) = -\sum_{v} \sum_{j=1}^{n} P(G_j, AP_i = v) \log P(G_j/AP_i = v) \tag{3}$$

Then the variation of the information entropy, which indicates the loss of position uncertainty, or information gain is [11]:

$$\text{InfoGain}(AP_i) = H(G) - H(G/AP_i) \tag{4}$$

In the above formulas, $P(G_j)$ is the prior probability of position G_j. When all RPs are uniformly distribued, $P(G_j)$ is a constant $1/n$; $P(G_j, AP_i = v)$ is the joint probability under condition when RSSI data collected from AP_i is v at the position G_j, and $P(G_j/AP_i = v)$ is the conditional probability of location G_j when RSSI data collected from AP_i is v.

3.2 BKM Based Clustering

A cluster is the convergence of points in the test space [12]. The distance between any two points of the same cluster is less than the distance between any two points of different clusters; the cluster can be described as a space with a relatively high density. In fact, clustering is an unsupervised classification and usually does not require the use of training data for learning. We adopt BKM as the clustering method in this paper [12].

BKM can be regarded as an optimization version of k-means. The difference between BKM and k-means is that BKM reaches a global optimum, while k-means just reaches a local optimum. K-means can be greatly affected by the initial cluster centers and may lead to a nonideal division, which leads to a local optimum. Moreover, because it operates less similarity computing, BKM can accelerate the execution speed of clustering.

This algorithm first takes all the points as a cluster, and then divides all the points into two clusters. When partitioning, one point is chosen randomly as the first initial cluster center. Then the point which is farthest from the set center is selected as the second initial cluster center. The next procedure is to perform K-means algorithm to divide points into two clusters. Then one of the clusters that has the maximum value of the Sum of Squares for Error (SSE) is selected to continue the partition. The partitioning process is repeated until k clusters are formed for the points.

In n-dimensional Euclidean space, SSE can be calculated by the formula below

$$\mathrm{SSE} = \sum_{i=1}^{k} \sum_{\boldsymbol{p} \in C_i} \mathrm{dist}(\boldsymbol{p}, \boldsymbol{c_i})^2 = \sum_{i=1}^{k} \sum_{\boldsymbol{p} \in C_i} (\boldsymbol{p} - \boldsymbol{c_i})^2 \tag{5}$$

$$\boldsymbol{c_i} = \frac{\sum\limits_{\boldsymbol{p} \in C_i} \boldsymbol{p}}{n_{c_i}} \tag{6}$$

In formulas above, C_i is the i_{th} cluster, $\boldsymbol{p}$ is the point in cluster C_i, n_{c_i} is the element number in cluster C_i, $\mathrm{dist}(\boldsymbol{p}, \boldsymbol{c_i})$ is the Euclidean distance between $\boldsymbol{p}$ and $\boldsymbol{c_i}$. The i_{th} cluster center $\boldsymbol{c_i}$ is updated after each iteration.

The specific implementation process of the clustering algorithm is described in Algorithm 1.

3.3 Decision Tree Algorithm

After constructing k clusters, we can just estimate the approximate position of the user roughly. If we construct a cluster for each position, the amount of calculation will be very large and it will be very time-consuming, seriously affect the real-time positioning. So we need to use other algorithms to locate precisely. The decision tree can classify samples with a series of attributes and has the

Algorithm 1. BKM algorithm for finding k clustering

Require: The mean RSSI vectors p for all RPs G;
Require: The number of clusters k;
Ensure: k clusters: C_1, C_2,... C_k
  $m \leftarrow 1$
  $C_m \leftarrow G$
  while $m < k$ **do**
    Compute SSE_m
    $j^* = \arg\max_j(SSE_1, SSE_2, ...\mathrm{SS}E_j, ...SSE_m)$
    Use k-means algorithm to split C_j^* into two clusters: C_j^*, C_{m+1}
    $m \leftarrow m + 1$
  end while

advantage of high efficiency. In this paper, we use C4.5 decision tree for precise localization [11,13].

There are two steps to build a decision tree: selecting splitting attributes and choosing splitting points. We can simplify the procedure of building a decision tree for deciding which attribution to choose and how to split data at each splitting point. Different attributes are selected at different branches. We select information gain ratio as standard to choose splitting attributes. And it is necessary to decide segmentation points for each attribute. Suppose that we have dataset $D = \{D_1, D_2, \ldots D_n\}$, in which D_i means fingerprints at position G_i. And m APs are detectable at all n positions in the cluster. We can divide D into k subsets $\{T_1, T_2, \ldots, T_k\}$. And T_i is the subset when RSSI data are collected from AP_i. Then we need to select splitting attributes with the maximum Ratio (AP_i).

Step 1. When we have already clustered the RPs, we should make a decision tree for each cluster. For a certain cluster $\mathscr{A}$, we adopt all RSSI data collected at positions in cluster $\mathscr{A}$. We adopt information entropy Info(D) to demonstrate the uncertainty of each position [11,13].

$$\mathrm{Info}(D) = -\sum_{j=1}^{n} P(G_j) * \log P(G_j) \tag{7}$$

Step 2. We can learn from information gain that the information entropy decreases when we get more information. When we get the information of M attributes for each position, the entropy becomes Info(D/APi) [11,13].

$$\mathrm{Info}(D/AP_i) = -\sum_{v} \sum_{j=1}^{n} (P(G_j, AP_i = v) \log P(G_j/AP_i = v)) \tag{8}$$

Step 3. So we can get to know that the information gain [11,13].

$$\text{Gain}(AP_i) = \text{Info}(D) - \text{Info}(D/AP_i) \tag{9}$$

Step 4. The information entropy of attribute AP_i is $H\,(AP_i)$ [13].

$$H\,(AP_i) = -\sum_{v}^{Ti} P(v) * log_2 P(v) \tag{10}$$

Step 5. The ratio of attribute AP_i is $\text{Ratio}(AP_i)$ [13].

$$\text{Ratio}(AP_i) = \frac{\text{Gain}(AP_i)}{H(AP_i)} \tag{11}$$

In the formulas, $P\,(G_j)$ is the probability of G_j in the environment. $P(G_j, AP_i = v)$ indicates the joint probability when RSSI value collected from AP_i is v at position G_j; $P(G_j/AP_i = v)$ is the conditional probability of G_j when RSSI value collected from AP_i is v; v is the element in T_j, and $P(v)$ is the probability of v in T_i.

The RSSI data are consecutive values, so we need to discretize the data first. Discretization refers to selecting some appropriate segmentation points in RSSI data and dividing them into several ranges. If some segmentation point $\mathcal{X}$ is chosen, data will be discretized into ranges (RSSI data $\leq \mathcal{X}$) and (RSSI data $\geq \mathcal{X}$). The principle of choosing segmentation points is as follows. First, we systematically arrange the RSSI data obtained at n positions from AP_i from small to large. And then we preselect one segmentation points ($l \leq m$, and we do not constitute segmentation points inside the same RSSI data subset). We compute the information gains corresponding to the condition of the preselected segmentation points. And the preselected segmentation point with the largest information gain is selected as the final segmentation point of AP_i.

3.4 Online Positioning Stage

In the actual online positioning process, when the user moves in the environment, the mobile terminal first detects the signals sent by the surrounding APs, records RSSI data, and then traversals the fingerprint database to perform matching. Once the best match is found, the location of the user is estimated as the location of the best matching fingerprint.

The online positioning process can be regarded as the inverse process of constructing clusters and building decision trees. We first calculate the distance between the user's RSSI data and each cluster center, then select the cluster with the smallest distance as the position set the user belongs to. We use decision tree in that cluster for judgment. The feature values are compared with the corresponding split point and then compared with the left subtree and right subtree. We continue this process until it reaches the leaf node.

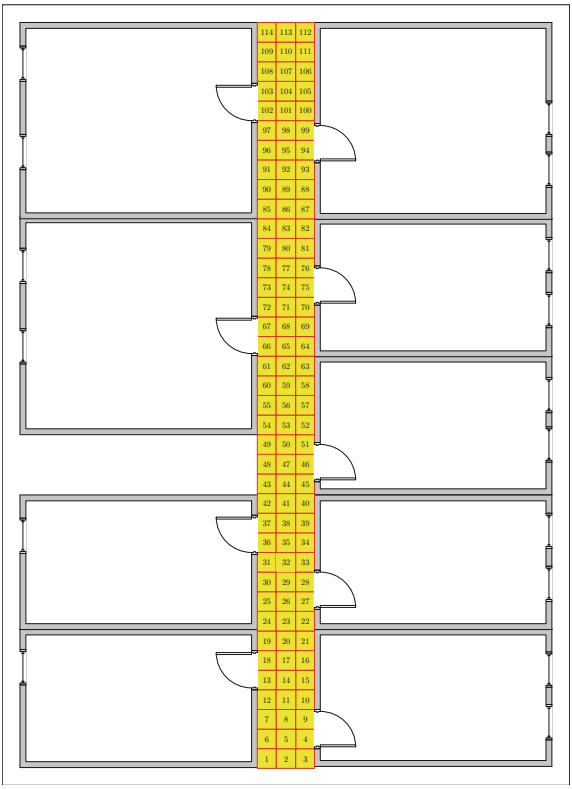

Fig. 2. Experimental environment

4 Experiment

We perform this experiment on a part of the corridor of the 4th floor of the main building. Owing to the fact that there already exist a certain number of *AP*s in the environment, we did not install any *AP*s to the environment. Figure 2 shows the schematic diagram of the corridor on the fourth floor. Each position area is 0.8 m * 0.8 m, so the interval between two measurement positions is 0.8 m. We use WirelessMon to detect *AP*s and collect RSSI data. We choose positions 1–100 to collect RSSI data. At each position, we collect 50 groups of data.

In the experiment, we first collect RSSI data at 100 positions to build a database. Then we adopt AP selection method to filter out the bad performance *AP*s. We classify the RPs using clustering method with better performance *AP*s. For each cluster, we build a decision tree to do precise localization. Finally, we test this algorithm with online test set.

Localization accuracy is the correct rate at a certain precision. Localization accuracy highly depends on the maximum allowed distance between actual position and the tested position ($d_{max\_allowed}$). The accuracy can be obtained with

formula below:

$$\text{accuracy}(d_{max\_allowed}) = \frac{\sum_{i}^{n} 1\left\{\text{dist}(r_i, t_i) \leq d_{max\_allowed}\right\}}{n} \tag{12}$$

where we set $d_{max\_allowed}$ as 0.8 m, 1.6 m, and 2.0 m in the experiment; r_i is the real location of the i_{th} point, and t_i is the test location of the i_{th} point; $\text{dist}(r_i, t_i)$ is the Euclidean distance between r_i and t_i; $1\{\}$ is the indicator function. With (12), we can get to know the accuracy under different precision range.

Since maximum information gain is used to select the APs that are less affected by the noise in the environment, different numbers of APs indicate different degrees the data in database are affected. To investigate the accuracy of AP selection method, we compare our algorithm with the one without AP selection. Furthermore, we choose different numbers of APs to compare the localization accuracy. The position accuracy of different numbers of APs are listed in Fig. 3. We set the number of APs as m. In our environment, there are 17 APs in total. The result shows that AP selection method improves the performance of localization accuracy, and the localization accuracy decreases once m is too large or too small. In Fig. 3, the lines show the localization accuracy in 0.8 m, 1.6 m, and 2 m when we set k as 5 respectively.

It can be observed that localization accuracy changes with different AP numbers. And when we select 16 APs, we gain the best localization accuracy in 0.8 m, 1.6 m and 2 m with BKM algorithm. The result shows that the localization accuracy improves with AP selection method.

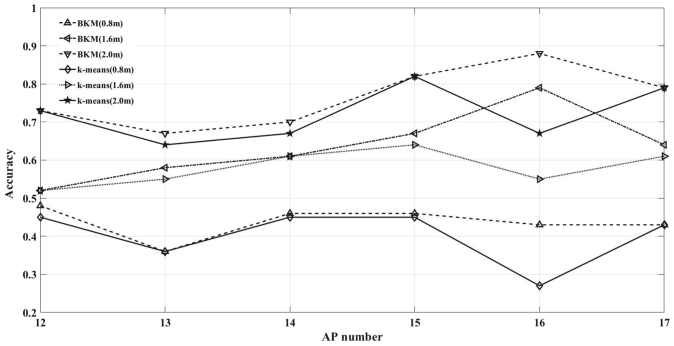

Fig. 3. Performance of AP selection

We adopt BKM to perform clustering in this paper. To compare the performance of BKM with k-means, we set the k-means as a comparision group. Figure 4 presents the results when we choose different k in clustering algorithm. And we select 16 APs during the experiment. When k is too large, the points that are Relatively relevant might be devided into different clusters. And this

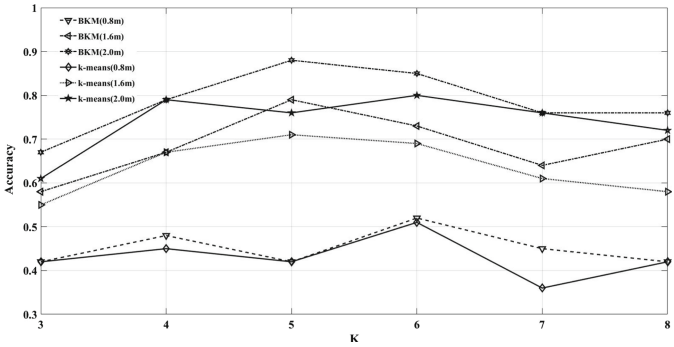

Fig. 4. Performance of clustering

decreases the divergence between different clusters. Furthermore, when k is too small, the points that are less relevant are much more likely to be devided into a same cluster, which will decrease the resolution of a cluster. Thus, the localization accuracy decreases with k being too large or too small.

Table 1 shows the comparison about localization precision between BKM and k-means. Average error is the average of localization error, which is computed using (13). Max error is the maximum localization error distance.

Table 1. Comparison of localization between BKM and k-means.

Algorithm	Average error	Max error
BKM(m)	1.51	5.60
k-means(m)	1.58	5.60

$$\text{Average error} = \sum_{i}^{n} \sum_{\text{Dist}} (\text{dist}(r_i, t_i)) \cdot P(\text{dist}(r_i, t_i)) \cdot P(r_i) \tag{13}$$

where Dist is the set composed of all distances between real locations and test locations, and $\text{dist}(r_i, t_i) \in \text{Dist}$, $\forall$ i=1, 2..., n; $P(\text{dist}(r_i, t_i))$ is the probability of $\text{dist}(r_i, t_i)$. $P(r_i)$ is the probability of location r_i. The result shows that the BKM significantly reduces localization errors. It is clear that BKM outperforms k-means.

As we can see from Figs. 3 and 4, BKM almost always performs better than k-means. The result indicates that BKM can markedly improve the localization accuracy. BKM is a better performance algorithm than k-means.

5 Conclusion

This paper adopts an efficient indoor localization which utilizes *AP* selection, clustering and decision tree. We adopt maximum information gain to filter out the *AP*s that do not perform well in lessening the uncertainty of localization. Clustering is presented to locate roughly. And decision tree is introduced to streamline the positioning. The result proves that our algorithm gains much better performance in improving positioning accuracy.

The future works is to utilize CSI (channel state information) in indoor localization, which is a finer-grained and more stable channel characteristic. We may also combine CSI with machine learning to improve localization accuracy.

References

1. Liu, H., Darabi, H., Banerjee, P., et al.: Survey of wireless indoor positioning techniques and systems. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **37**(6), 1067–1080 (2007)
2. Yassin, M., Rachid, E.: A survey of positioning techniques and location based services in wireless networks. In: IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES), pp. 1–5 (2015)
3. Basri, C., Khadimi, A.E.: Survey on indoor localization system and recent advances of WiFi fingerprinting technique. In: International Conference on Multimedia Computing and Systems (ICMCS), pp. 253–259 (2016)
4. Tang, P., Huang, Z., Lei, J.: Fingerprint localization using WLAN RSS and magnetic field with landmark detection. In: International Conference on Computational Intelligence Communication Technology (CICT), pp. 1–6 (2017)
5. Zhang, G., Zhan, X., Dan, L.: Research and improvement on indoor localization based on RSSI fingerprint database and K-nearest neighbor points. In: International Conference on Communications, Circuits and Systems (ICCCAS), pp. 68–71 (2013)
6. Lemic, F., Handziski, V., Caso, G., et al.: Enriched training database for improving the WiFi RSSI-based indoor fingerprinting performance. In: IEEE Annual Consumer Communications Networking Conference (CCNC), pp. 875–881 (2016)
7. Alshamaa, D., Mourad-Chehade, F., Honeine, P.: Localization of sensors in indoor wireless networks: an observation model using WiFi RSS. In: IFIP International Conference on New Technologies, Mobility and Security (NTMS), pp. 1–5 (2018)
8. Chen, G., Liu, Q., Wei, Y., et al.: An efficient indoor location system in WLAN based on database partition and Euclidean distance-weighted Pearson correlation coefficient. In: IEEE International Conference on Computer and Communications (ICCC), pp. 1736–1741 (2016)
9. Zhou, R., Lu, S., Chen, J., et al.: An optimized space partitioning technique to support two-layer WiFi fingerprinting. In: IEEE Wireless Communications and Networking Conference (WCNC), pp. 1–6 (2017)
10. Yang, G., Gao, F., Zhang, H.: An effective calibration method for wireless indoor positioning system with mixture Gaussian distribution model. In: IEEE International Conference on Computer and Communications (ICCC), pp. 1742–1746 (2016)
11. Chen, Y., Yang, Q., Yin, J., et al.: Power-efficient access-point selection for indoor location estimation. IEEE Trans. Knowl. Data Eng. **18**, 877–888 (2006)

12. Steinbach, M., Karypis, G., Kumar, V.: A comparison of document clustering techniques. In: KDD Workshop on Text Mining (2000)
13. Quinlan, J.R.: Learning efficient classification procedures and their application to chess end games. In: Michalski, R.S., Carbonell, J.G., Mitchell, T.M. (eds.) Machine Learning: An Artificial Intelligence Approach. SYMBOLIC, pp. 463–482. Springer, Heidelberg (1983). https://doi.org/10.1007/978-3-662-12405-5_15