IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 19, 2019, accepted February 13, 2019, date of publication February 26, 2019, date of current version March 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2901736

Crowdsourcing Indoor Positioning by Light-Weight Automatic Fingerprint Updating via Ensemble Learning

JIANGHONG YANG, XIAOHUI ZHAO, (Member, IEEE), AND ZAN LI, (Member, IEEE)
College of Communication Engineering, Jilin University, Changchun 130012, China

Corresponding author: Zan Li (zanli@jlu.edu.cn)

This work was supported by the Natural Science Foundation of China under Grant 61801189.

ABSTRACT In recent years, Wi-Fi-based indoor positioning has attracted increasing research attention due to its ubiquitous deployment. Although extensive research has been conducted on Wi-Fi fingerprint-based positioning, especially, in complex environments and long-term deployments, the automatic adaptation of radio map has not been fully studied and the problems remain open. When the positions of some Access Points (APs) change, the traditional approach regularly conducts site surveying which is time-consuming and labor-intensive. In this paper, we propose a crowdsourcing indoor positioning approach based on ensemble learning for automatic Altered APs Identification and Fingerprints Updating, namely AAIFU. We propose an algorithm to detect and identify the altered APs in crowdsourcing data. After getting the altered APs, we use the relationship between the received signal strength values of the altered APs and the unaltered APs in the crowdsourcing data to train a prediction model used to update the radio map. We also handle the device diversity on all the processes of AAIFU. Our proposed solution is light-weight which does not rely on additional infrastructure and inertial sensors with high power consumption. The comprehensive experiments have been carried out in our teaching building to evaluate the effectiveness of AAIFU. The results show that our proposed AAIFU can effectively adapt the radio map to the movement of APs and improve positioning accuracy. Correspondingly, we achieve an average positioning accuracy of 2.6m which outperforms the fingerprinting approach with the original radio map by 63.9%.

INDEX TERMS Indoor positioning, ensemble learning, radio map updating, crowdsourcing.

I. INTRODUCTION

In the past decades, many wireless indoor positioning techniques have been developed rapidly, including Wi-Fi, RFID (Radio Frequency Identification), UWB (Ultra Wide Band), etc. Because of the ubiquitous deployment of Wi-Fi infrastructure and convenience of extracting Received Signal Strength Indicator (RSSI), indoor positioning based on Wi-Fi RSSI has attracted extensive attentions. Particularly, fingerprinting approaches based on Wi-Fi RSSI have become the most popular solution for indoor positioning due to its high positioning accuracy. Typically, Wi-Fi fingerprint-based indoor positioning has two stages [1], i.e., offline radio map construction by site surveying and online positioning by fingerprint matching. In the offline stage, we need to generate a radio map containing fingerprints at known physical locations. Each fingerprint is a vector of Received Signal Strength (RSS) values from Wi-Fi Access Points (APs) labeled by its ground truth locations. In the online positioning stage, the mobile client collects RSS values and sends the measurement results to the server in which pattern matching algorithms are adopted to match the measured RSS with the most similar fingerprints in the database and return the associated location.

To improve the accuracy of Wi-Fi fingerprint-based indoor positioning, various pattern matching algorithms have been widely investigated [2]–[5] for fingerprint matching, which are generally divided into deterministic [6]–[8] and probabilistic methods [9]–[11]. RADAR [6] as the pioneer work of indoor positioning uses the nearest neighbor algorithm as a deterministic method for fingerprint matching to find user's location from fingerprint database. HORUS [9] proposes a probabilistic method based on the signal intensity distribution

The associate editor coordinating the review of this manuscript and approving it for publication was Nafees Mansoor.

in a radio map for fingerprinting matching to improve the positioning accuracy.

In addition to the fingerprint matching, the positioning accuracy significantly depends on the freshness of a radio map. If some APs have been moved, namely altered APs, the fingerprints stored in the server will be outdated. For example, we find that the fingerprint database stored in the offline stage may not match current signal environment as time goes by and the positioning accuracy degrades gradually in our previous work [12]. To update the radio map, a common way is to regularly conduct site surveying by professionals which is however labor intensive and time consuming.

In recent years, crowdsourcing approaches [13]–[18] have been intensively investigated for constructing and updating radio maps to eliminate site surveying and positioning. On one hand, some research [19]–[21] has been conducted in trace matching algorithms to generate radio maps based on the crowdsourcing user traces. In these studies, the updated radio map can automatically be generated without site surveying by professionals. However, these trace matching algorithms relying on inertial sensors to produce user traces are usually affected by uncertain holding attitudes of smartphones and normally with high power consumption. On the other hand, some crowdsourcing approaches [1], [19] have been proposed to identify the altered APs and update the RSS of these altered APs in the original radio map. As a result, the original radio map generated from initial site surveying can be automatically updated by replacing the outdated RSS of the altered APs with their fresh RSS.

In this work, we propose a system with Altered APs Identification and Fingerprint Updating, namely AAIFU. AAIFU targets on adopting the crowdsourcing RSS data to automatically identify the altered APs and update their RSS in an original radio map generated by initial site surveying. Compared to regenerating the radio map by trace matching, AAIFU makes full use of information of the original radio map in which the RSS values of the unaltered APs remain unchanged. Additionally, AAIFU is a light-weight radio map updating solution by which the power-consuming and error-prone data collection of crowdsourcing user traces are avoided. To achieve the aforementioned goals, we propose a three-step crowdsourcing radio map updating algorithm in AAIFU comprised of fast detection of altered APs, identification of altered APs and radio map updating. We summarize the main contributions of this work as follows.

A. FAST DETECTION OF ALTERED APS

In the first step of AAIFU, we design a fast detection algorithm to detect the existence of altered APs based on the relationship among the RSS values of different APs which is predefined by a *GBDT* (Gradient Boosting Decision Tree) regression model for each AP in the original radio map. If the RSS in the crowdsourcing data does not fit the predefined regression model due to the movement of certain APs, we can accurately detect the existence of altered APs. This step is used to skip the following steps for identifying the altered APs and updating the radio map which are normally time and power consuming, if there are no any altered APs.

B. IDENTIFICATION OF ALTERED APS

If altered APs are detected, a novel crowdsourcing approach is adopted to identify the altered APs in AAIFU. In this step, we identify the altered APs based on the differences between the measured RSS values in the crowdsourcing data and the predicted values based on the predefined *GBDT* regression model in the original radio map. We counts the times when the difference is larger than a predefined threshold for each AP. A two-class clustering algorithm is adopted to recognize the altered APs from the unaltered ones. With our proposed algorithm, we can achieve a 100% identification accuracy according to our experiments.

C. RADIO MAP UPDATING

We give a radio map updating algorithm based on *GBDT* regression. We use the collected crowdsourcing data to learn the mapping function between the RSS of the altered and the unaltered APs via *GBDT* regression. Once we have learned the mapping function from the crowdsourcing data, we use it to update entire radio map, which can better reflect current spatial signal environment.

We further implement a Weighted K-Nearest Neighbor (WKNN) algorithm [6], [7] for positioning to evaluate the effectiveness of the radio map updating in AAIFU. We conduct a set of experiments in a teaching building with five altered APs. The experiment results show that AAIFU is robust to altered APs and our average positioning accuracy achieves 2.5m which significantly outperforms the fingerprinting approach with the original radio map.

The rest of the paper is organized as follows. We review the related work and summarize the problems in indoor positioning based on fingerprinting in Section II. The preliminaries of *GBDT* regression are presented in Section III. Our main contributions in AAIFU are presented in Section IV. In Section V, we present the implementation of AAIFU system. Then the experimental results and evaluation are given in Section VI. Finally, Section VII concludes the paper.

II. RELATED WORK AND PROBLEMS OF FINGERPRINTING

To update radio map for a fingerprinting-based positioning system, there are still many challenges. In this section, we review some related work and summarize the problems in this domain.

When locations of some APs change, the traditional approaches to update radio map is to regularly make site surveying which is time consuming and labor intensive. In order to reduce these burdens of surveyors, automatic radio map updating methods are required. In recent years, some research has been conducted for automatically updating radio map to the changes of surrounding environments [1], [19], [22], [23], and [24]. In these studies, some work relies on certain fixed reference anchors to estimate the RSS changes of surrounding APs and update the radio map. For example, Ni *et al.* [22] and Krishnan *et al.* [23] leverage an additional set of pre-deployed reference anchors with fixed locations to obtain fresh RSS for updating radio maps. Atia *et al.* [24] estimate the updated RSS values relying on some additional sniffers with fixed locations. However, the deployment of additional devices and fixed reference anchors is expensive and non-scalable, which hinders the inherent advantages of fingerprint-based positioning.

Crowdsensing approaches [25]–[27] have recently been adopted to position and automatically update radio maps for leveraging crowdsourcing inertial sensor data. Relying on the crowdsourcing user traces constructed by inertial sensors, trace matching algorithms are designed to establish original user walking paths and generate radio maps [20], [28], [29], [30] and [31]. AcMu in [19] locates users through smartphone inertial sensors and indoor signal markers, which collects signals from smartphones by long time monitoring. Although AcMu eliminates the need for surveyors, the holding attitudes of smartphones significantly affects the accuracy of estimated user traces based on inertial sensors. Moreover the widespread use of inertial sensors leads to high power consumption. The work of [26] requires additional receivers called reference points installed in the environment. In this approach, the prediction of user location is realized by the regression models originated from a static radio map using RSS values collected by the reference points along with signals captured by mobile devices. The deployment of additional receivers and reference points is expensive and non-scalable. Lin *et al.* [27] propose a cross-domain cluster intersection algorithm to weight each sample reliability and construct radio propagation surfaces by polynomial functions matching. However, this work assumes that each crowdsourced sample has been annotated with a location, which is hard to get and the predicted location could be incorrect if altered APs exist.

In our work, we focus on the design of a light-weight crowdsourcing radio map updating system only relying on the relationship of the collected RSS values from different APs, which avoids pre-deploying anchor nodes and power-consuming data collection of inertial sensors.

To achieve the aforementioned goal, there are three challenges to address as follows, namely accurate identification of altered APs, adaptive fingerprint updating and device diversity handling

A. ACCURATE IDENTIFICATION OF ALTERED APS

Accurate identification of the altered APs is the prerequisite for radio map updating. He *et al.* [1] propose a cluster-based detection method to detect the altered APs based on locations of the target users. Because the locations of the target users are unavailable in crowdsourcing data, this work has to first estimate the locations of the users. However, due to the fluctuation of the measured RSS values, the positioning results may not be corresponding to their ground truth locations, which will introduce deviations to judging of dense clusters and identification of the altered APs. Instead of identifying the altered APs, AcMu [19] updates the RSS values of all APs even the APs in the radio map are not altered. This approach does not well exploit the information of unaltered APs in original radio map and may degrade the accuracy of radio map when the number of altered APs is small.

Therefore, in this paper, we aim to first identify the altered APs with the crowdsourcing data in which the locations of the users are unavailable.

B. ADAPTIVE FINGERPRINT UPDATING

After identifying the altered APs, it is still challenging to accurately update the RSS values of these altered APs in a radio map to match new space environment. He *et al.* [1] propose to estimate the locations of the user relying on the original radio map first and then to update the radio map based on the estimated locations. However, as mentioned before, due to the fluctuation of the measured RSS values, the estimated locations may not be reliable. AcMu uses inertial sensor data and trace matching methods to locate users for updating the radio map [19]. However, because of the uncertain holding attitudes of smartphones, it is still very difficult to generate accurate user traces by crowdsourcing data collection. Therefore, the matched traces may not be accurate enough for updating radio map. Besides, the trace matching algorithms require long user traces but some crowdsourced traces are too short to be used to position users. As a result, it is not easy to obtain accurate location based on crowdsourcing data collection.

Therefore, we aim to avoid using inaccurate information of user locations for updating RSS of the altered APs in the radio map.

C. DEVICE DIVERSITY HANDLING

Two collocated devices may receive different signal strengths from the same AP due to difference in hardware. Li *et al.* [32] show a histogram of RSS offsets measured by two collocated phones HTC Titan and Nokia Lumia 900 at various locations. Specifically, the data is collected in a large shopping mall where the two devices are put at various locations to measure RSS from surrounding APs. The offset can lead to bad positioning results if the radio map is constructed with one device and directly used by another one [33]–[35]. One solution to this problem is to measure the offset of hardware gains between any pair of devices, which, however, is not scalable [35]. Fang *et al.* [33] evaluate the existing methods and present the effectiveness of using RSS differences between the observed APs in each fingerprint instead of using the absolute RSS. However, it cannot be used in our work since it cannot guarantee that the reference AP is unaltered AP.

Therefore, we aim to adopt a simple and efficient approach to handle device diversity, which can fit our regression model.

III. PRELIMINARIES OF GBDT REGRESSION

To solve the aforementioned problems in automatic radio map updating, regression is required to find the relationship among the RSS values of different APs used for both identification of altered APs and fingerprint updating in our system. *GBDT* regression [36]–[39] is an ensemble learning approach with high prediction accuracy and ability of processing nonlinear data, which is widely used to solve practical problems. In the work of [40]–[43], we can find that in these specific areas, compared with machine learning algorithms such as *SVM*, *KNN*, etc., ensemble learning algorithms can get better results, which present the ability of ensemble learning algorithms to solve practical problems. As an important branch of ensemble learning, *GBDT* has its unique advantages. In the work of [44], [45], we can find the applications of *GBDT* in many fields and its outstanding performance.

Generally, our aim is to learn a stable and well-behaved model with high generalization ability. This kind of model is normally hard to obtain and we often get some models which only perform well in some specific cases. However, the ensemble learning [46]–[49] is to combine several weak models in order to get a better and more comprehensive model. The potential idea of ensemble learning is that even if one weak learner gets a wrong prediction, other weak learners can correct the error in some extent. In practice, the ensemble learning tends to mitigate over-fitting and obtain good predictive performance with high generalization ability.

GBDT regression is an important branch of ensemble learning which uses decision trees as basic learners. The decision tree is easy to understand and implement, which has fast training rate and high efficiency. *GBDT* first learns a decision tree with an initial value and a predicted value can be obtained at the leaf. Then *GBDT* will learn the decision tree based on the residuals of the true value and the predicted value of the previous decision trees until the residual is sufficiently small. Finally, the predicted value of the test sample is the cumulation of the previous decision trees.

Basically, *GBDT* regression combines the weak learners into a single strong learner in an iterative way which considers additive models of the following form [36]

$$F(x) = \sum_{m=1} \gamma_m h_m(x) \tag{1}$$

where $h_m(x), m = 1, 2, \cdots$, are the basis functions usually called weak learners in the context of boosting. *GBDT* uses decision trees of fixed size as weak learners. The decision trees have abilities to make these learners valuable for boosting, namely the ability to handle data of mixed type and the ability to model complex functions. Similar to other boosting algorithms, *GBDT* builds an additive model in a forward stagewise fashion

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x) \tag{2}$$

At each stage, $h_m(x)$ is chosen to minimize a loss function L given the current model and its fit $F_{m-1}(x)$.

$$F_m(x)=F_{m-1}(x)+\arg\min_h \sum_{i=1}^{n} L(y_i, F_{m-1}(x_i) + h(x)) \tag{3}$$

To find an initial model $F_0(x)$ is a specific problem. For the least-squares regression L, the mean of the target values are chosen as $F_0(x)$. Gradient Boosting attempts to solve the minimization problem numerically via steepest descent. The steepest descent direction is the negative gradient of the loss function evaluated at the current model. In this paper, we take this approach to get

$$F_m(x) = F_{m-1}(x) - \gamma_m \sum_{i=1}^{n} \nabla_F L(y_i, F_{m-1}(x_i)) \tag{4}$$

where the step length γ_m is

$$\gamma_m = \arg\min_\gamma \sum_{i=1}^{n} L(y_i, F_{m-1}(x_i) - \gamma \frac{\partial L(y_i, F_{m-1}(x_i))}{\partial F_{m-1}(x_i)}) \tag{5}$$

Because *GBDT* regression can efficiently solve practical problems with high accuracy and high generalization ability, we identify our altered APs and update our radio map by using *GBDT* regression.

IV. ALTERED APS IDENTIFICATION AND FINGERPRINT UPDATING (AAIFU)

In general, a radio map is constructed by a list of fingerprints containing the RSS values and the corresponding locations over all sample points in the region of interest. The physical area of interest is sampled as a finite location space $L = \{l_1, l_2 \cdots, l_n\}$, where n is the total number of sample locations, and each location is represented by its coordinate $l_i = (x_i, y_i)$, $1 \leq i \leq n$. Correspondingly, the fingerprints are modeled as the signal space $F = \{f_1, f_2, \cdots, f_n\}$ where each $f_i = \{v_1^i, v_2^i, \cdots, v_p^i\}$ is the fingerprint record corresponding to its ground truth location l_i, v_j^i denotes the RSS value of the jth AP at position l_i, $1 \leq j \leq p$, and p is the total number of APs in the target location space. The fingerprints in a radio map can be defined as a matrix as

$$\begin{aligned} F &= \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix} = \begin{bmatrix} v_1^1 & v_2^1 & \cdots & v_p^1 \\ \cdots & \cdots & \ddots & \cdots \\ v_1^n & v_2^n & \cdots & v_p^n \end{bmatrix} \\ &= [V_1, V_2, \cdots, V_p] \end{aligned} \tag{6}$$

where V_j is the column vector of the jth AP value. The original radio map is denoted as RM_0 established at time point t_0, i.e., in the offline stage. If the location of the jth AP changes after construction of the original radio map, the values of V_j have to be adapted in the updated radio map.

To design an automatic radio map updating system relying on crowdsourcing data, we have to handle the problems presented in Section II and our solution has four main parts, i.e., device diversity handling, fast detection of altered APs, identification of altered APs, and radio map updating.

A. DEVICE DIVERSITY HANDLING

In our AAIFU, we develop a light-weighted scheme to handle device diversity during fingerprint matching and regression. The idea is similar to that of [34] where Expectation Maximization (EM) is used to estimate the device power level. However, the system architecture in [34] is different from ours where we need to locate mobile clients and update the radio map when there exists altered APs. Our scheme is detailed as follows.

The conventional calculation of signal space distance between a pair of fingerprints $f_i = \{v_1^i, v_2^i, \cdots, v_p^i\}$ and $f_j = \{v_1^j, v_2^j, \cdots, v_p^j\}$ is

$$\Delta_{ij} = \sum_{k=1}^{p} \left| v_k^i - v_k^j \right| \tag{7}$$

We modify this calculation by adding a constant δ in Equation (7) as $\Delta_{ij} = \sum_{k=1}^{p} \left| v_k^i - v_k^j + \delta \right|$. We adjust δ in a certain range, and the gain offset o_{ij} between two devices is estimated by

$$o_{ij} = \arg\min_{\delta} \Delta_{ij} \tag{8}$$

In the offset estimation, we simplify the transformation between a pair of devices A and B as a constant, i.e., $RSS_A = RSS_B + \delta$. Park *et al.* [35] find that the offsets of the same type of devices are typically small but the offsets of different types of devices are large. Hence, we conduct a set of preliminary experiments to obtain δ for different types of devices. Then, we can compensate δ for the crowdsourcing data collected from the corresponding types of devices. Note that we are normally able to collect the types of devices from the crowdsourcing users.

B. FAST DETECTION OF ALTERED APS

Because the altered APs do not occur frequently, we first design an approach to fast detect whether there are the altered APs in the original radio map before identifying the altered APs and updating our radio map. The fast detection of the altered APs is based on the deviation between the relationships among the RSS values from APs in the crowdsourcing data and the original radio map. In this step, the critical issue is to model the functional relationship among the RSS values from APs and predict the existence of the altered APs based on crowdsourcing data.

First, from the original radio map (Equation (6)), we learn a predictive functional relationship $\mathcal{H}_j$, $1 \leq j \leq p$, based on *GBDT* regression among the RSS values of all APs. Considering the jth AP, the functional relationship $\mathcal{H}_j$ from the APs excluding the jth AP itself as

$$V_j = \mathcal{H}_j(V_1, \cdots, V_{j-1}, V_{j+1}, \cdots, V_p) \tag{9}$$

which reflects the mapping of the RSS values received from the $p-1$ APs to the RSS value of the jth AP. It is expected that the above relationship can capture the relationship between the RSS values of the jth AP and the values of the other APs. If AP_j are altered APs in the new collected data, the predicted value obtained by the learned function $\mathcal{H}_j$ will be greatly deviated from the true value which is actually received. If AP_j are not altered APs, the predicted value will be close to the true value.

Then, given a piece of data $[v_1, v_2, \cdots, v_j, \cdots, v_p]$ from the crowdsourcing RSS data set by p APs collected at time point t, we use the learned function $\mathcal{H}_j$ to calculate the prediction RSS value $\hat{v}_j$ of the jth AP. The difference between the predicted and the measured value of each AP is

$$\Delta v_j = \left| \hat{v}_j - v_j \right| \tag{10}$$

Finally, we sort Δv of all APs in descending order and calculate the average value of the first half of the differences denoted by $\Delta \bar{v}$ for avoiding accident errors caused by fluctuation of the measured values. If $\Delta \bar{v}$ is larger than a threshold φ which is empirically set to 10, this piece of data indicates that the altered APs exists and it will be labeled. We repeat the aforementioned procedure for all data in the crowdsourcing RSS data set and count the number of labeled data. If the number of labeled data is more than 50% of the total number of the crowdsourcing data, we determine that there exists the altered APs, from which we know that the original radio map does not fit the current signal space environment and the radio map needs to be updated. Otherwise, the radio map will not be updated.

C. IDENTIFICATION OF ALTERED APS

When the existence of the altered APs are detected, we further identify these altered APs in two steps to avoid mislabeling, namely coarse identification and fine identification.

1) COARSE IDENTIFICATION

To identify the altered APs, we first detect the APs with high probability whose locations do not change. Same as the procedure in Section IV-B, for a piece of data in the crowdsourcing data set, we obtain Δv_j in Equation (10) based on $\mathcal{H}_j$ and the crowdsourcing data. Then we set a threshold γ. If Δv is larger than γ, we mark the jth AP as a may-altered AP. By the same way, we make an initial judgment on all APs and mark all may-altered APs. The remaining APs in this piece of data are considered as the unaltered APs whose locations do not change with high probability.

2) FINE IDENTIFICATION

After finding the unaltered APs based on the coarse identification, we further utilize these unaltered APs to identify the altered APs from the may-altered APs, namely fine identification. Assuming that AP_j and AP_k are detected as may-altered APs in the previous step, similar to the coarse identification, we first learn the mapping function $\mathcal{H}_j{}'$, $1 \leq j \leq p$ based on *GBDT* regression, which reflects the mapping of the RSS values from the unaltered APs to the may-altered APs in the original radio map as

$$V_j = \mathcal{H}_j'(V_1, \cdots, V_{j-1}, V_{j+1}, \cdots, V_{k-1}, V_{k+1}, \cdots, V_p) \quad (11)$$

where the RSS values as the input to $\mathcal{H}_j'$ are only taken from the unaltered APs. Because AP_j and AP_k are detected as the may-altered APs in the coarse identification, they are not taken as the input to $\mathcal{H}_j'$ in Equation (11). If AP_j are the altered APs, the predicted value obtained by the learned function $\mathcal{H}_j'$ will greatly deviate from the true value. According to the prediction function $\mathcal{H}_j'$, we calculate the difference between the predicted value and the measured one for the may-altered APs denoted as $\Delta v_j'$ for the jth AP. If $\Delta v_j'$ is smaller than γ, the location of the jth AP is considered to be static. Otherwise, the jth AP is considered to be moved and an alarm is given for this AP. The above procedure repeats for all the may-altered APs found in coarse identification, i.e., AP_j and AP_k. Note that the number of the may-altered APs in the algorithm can be any value (larger than 2 in the above example).

In order to prevent mislabeling the unaltered APs, we identify the altered APs by long-term measurement with a large set of crowdsourcing data. We know that given a certain amount of collected data, AAIFU can record the alarm frequency of each AP. According to our empirical experience, the alarm frequencies of the altered APs are significantly larger than the unaltered ones.

The critical issue of identifying the altered APs with high alarm frequency from other APs is to find the breakpoint of the sorted alarm frequencies, which can divide the APs into two classes, the altered APs and the unaltered APs. We consider this issue as a two-class clustering problem in one dimension, which can be efficiently solved by the Jenks natural breaks optimization method [50]. This data clustering method can determine the best arrangement of values into different classes. It is to minimize the average deviation of each class from class mean, while maximize the deviation of each class from the means of other groups. In other words, the method seeks to reduce the variance within classes and maximize the variance between classes. Specifically, AAIFU starts the altered AP identification by sorting the frequency values of all APs in decreasing order. Then, with each frequency value denoted as f as a breakpoint, it divides the ordered data into two classes denoted as C_1 and C_1. Finally, it calculates the sum of Squared Deviations from Class Means (SDCM) by

$$SDCM \triangleq \sum_{i=1}^{2} \sum_{f \in C_i} (f - \bar{f}_i)^2 \quad (12)$$

where $\bar{f}_i$ is the mean of frequencies within class C_i, $i = 1, 2$. AAIFU checks all possible combinations and finds the break point with the lowest SDCM to represent the smallest frequency variation within the class. Finally, AAIFU marks the APs in the class of higher frequency as the altered ones and reports them.

D. RADIO MAP UPDATING

Given the identified altered APs, we can update their values in the radio map. To update the RSS values of the altered AP in the radio map, the easiest way is to directly replace the outdated RSS values with new ones. Since the locations in the crowdsourcing data are unavailable, we propose a radio map updating approach based on the relationship among the RSS values of different APs. We implement a *GBDT* regression to efficiently and accurately achieve radio map updating. Using *GBDT* regression, we can accurately find the mapping relationship between the unaltered APs and the altered APs in the crowdsourcing data to represent new signal space environment after changing the locations of APs.

Let R_t be the set of N crowdsourcing data. Given m identified altered APs, the number of the unaltered APs is $p - m$. The *GBDT* regression function $\mathcal{F}$ is defined as

$$T_{al} = \mathcal{F}(T_{un}) \quad (13)$$
$$T_{al} = [V_a, V_b, \cdots, V_e]_{N \times m} \quad (14)$$
$$T_{un} = [V_1, V_2, \cdots, V_{a-1}, V_{a+1}, \cdots, V_p]_{N \times (p-m)} \quad (15)$$

where $V_a, V_b, \cdots, V_e$ refer to the RSS values collected from the altered APs and $V_1, V_2, \cdots, V_{a-1}, V_{a+1}, \cdots, V_p$ refer to the RSS values collected from the remaining APs.

Once $\mathcal{F}$ has been trained according to the crowdsourcing data, the remaining task is to update the radio map. For a location l_i, $1 \leq i \leq n$ in the radio map, the RSS values of the altered APs in the new fingerprint $f_i = \{v_1^i, v_2^i, \cdots, v_p^i\}$ are required to be updated. Therefore, f_i of location l_i is updated by

$$[\hat{v}_a^i, \hat{v}_b^i, \cdots, \hat{v}_e^i]_{1 \times m} = \mathcal{F}(t_{un}^i) \quad (16)$$
$$t_{un}^i = [v_1^i, v_2^i, \cdots, v_{a-1}^i, v_{a+1}^i, \cdots, v_p^i]_{1 \times (p-m)} \quad (17)$$

where $[v_1^i, v_2^i, \cdots, v_{a-1}^i, v_{a+1}^i, \cdots, v_p^i]_{1 \times (p-m)}$ denotes the RSS observations of the unaltered APs in the original radio map at the location l_i, and $[\hat{v}_a^i, \hat{v}_b^i, \cdots, \hat{v}_e^i]_{1 \times m}$ is the predicted new RSS of the altered APs at the location l_i. As a result, we replace the RSS of the altered APs with the predicted values in the radio map at the location l_i. Therefore, we update the entire radio map with $\mathcal{F}$.

V. THE AAIFU SYSTEM

The overall AAIFU system shown in Figure 1 is divided into two main parts: a data collection terminal and a server for

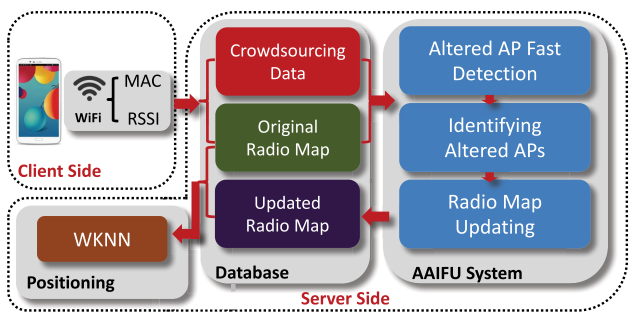


FIGURE 1. AAIFU system overview.

data processing. The data collection terminal is built up by a self-designed Android application running on smartphones. In this terminal, the collected data is sent to a cloud server for offline data processing, i.e., for detecting altered APs, updating radio map and indoor positioning. Note that, during radio map updating, users do not need to explicitly participate in measuring and uploading data. All data can be automatically collected through the application running on smartphones. Since radio map updating can be periodically executed once a day for example in out-of-service time, precisely at night, AAIFU updates the radio map timely and does not affect normal positioning.

A. AAIFU IMPLEMENTATION

Because identifying the altered APs and updating the radio map are time and power consuming, AAIFU adopts the algorithm for fast detection of the altered APs introduced in Section IV-B as a preliminary detection to estimate whether there are any altered APs according to the crowdsourcing data. As shown in Figure 2, we give the flow chart of AAIFU to indicate the algorithm procedure. If no altered APs are detected, the original radio map is used for positioning. Because the altered APs do not occur frequently, this quick preliminary detection can greatly accelerate the entire process. If the altered APs are detected, AAIFU moves to the next stage for identifying the altered APs comprised of the coarse identification and the fine identification introduced in Section IV-C. The coarse identification labels the may-altered APs and the fine identification further detects the altered APs in the may-altered ones to avoid mislabeling. Once the altered APs are identified, AAIFU updates the radio map as introduced in Section IV-D based on these altered APs and the crowdsourcing data. However, if the altered APs are more than one third of the total APs, the radio map updating via AAIFU is not necessary and it is better to conduct site surveying for a new radio map. If only a small portion of APs changes their positions, AAIFU can accurately update the original radio map without human participation.

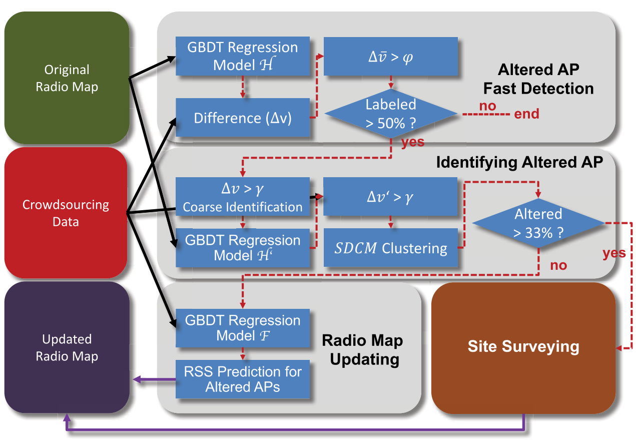


FIGURE 2. AAIFU flow chart.

B. POSITIONING WITH WKNN

After updating the radio map, we adopt a commonly used WKNN fingerprint matching method for positioning in AAIFU. In general, WKNN is to calculate the similarity between the measured RSS values and each reference point in the radio map, find the top k nearest reference points and compute the weighted average of the locations of the selected k reference points to get the target position.

The similarity between the fingerprint $f_i = \{v_1^i, v_2^i, \cdots, v_p^i\}$ and the target vector $V = \{v_1, v_2, \cdots, v_p\}$ is calculated based on the Euclidean distance, where the fingerprint f_i in the radio map is corresponding to the location l_i. The Euclidean distance is as follows

$$D_i = \sqrt{\sum_{j=1}^{p} (v_j^i - v_j)^2}, \tag{18}$$

where D_i represents the Euclidean distance between the target vector and the reference fingerprint corresponding to the position l_i, $1 \leq i \leq n$. We calculate this distance, select the top k fingerprints with the smallest Euclidean distance, and finally obtain the position of the target vector

$$\hat{l} = \sum_{i=1}^{k} w_i l_i \tag{19}$$

where

$$w_i = \frac{1}{D_i \sum_{i=1}^{k} \frac{1}{D_i}} \tag{20}$$

and $\hat{l}$ is the predicted position of the target vector. In our work, we set k to 3 in WKNN.

Besides the pattern matching algorithms (WKNN) for fingerprint matching, the features selected for fingerprint matching also affect the positioning accuracy. In fingerprint matching, RSS values from different APs are considered as the features used for pattern matching. We need to select the RSS values from certain APs with high importance on the location estimation to improve the positioning accuracy. The RSS values collected from these APs are called significant features. For feature selection in the area of machine learning, the easiest way is to select the features with high variance. In this work, to evaluate the importance of APs, we freely walk around and record the maximum and minimum values of each AP. We further sort the difference between the maximum and minimum values of each AP, then select the APs with larger differences as the significant features used in pattern matching (WKNN) for our positioning.

VI. EXPERIMENTS AND EVALUATION RESULTS

A. EXPERIMENT SETUP

To evaluate the proposed AAIFU system, we conduct a set of experiments in a teaching building of $1000m^2$ at Jilin University as shown in Figure 3. Specifically, the testing area consists of a corridor and 12 rooms, including laboratories, offices and classrooms. The testing area is crowded with

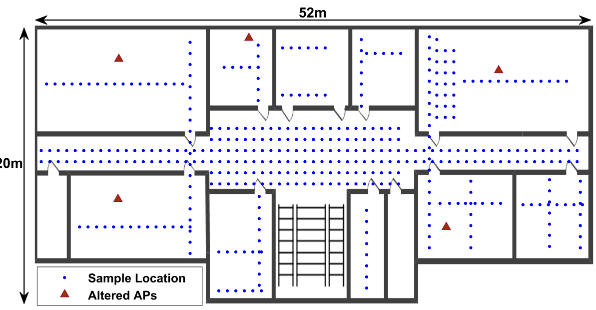


FIGURE 3. Experimental environment layout.

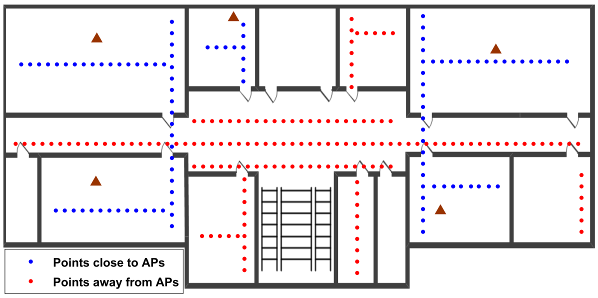


FIGURE 4. Test data layout in teaching building.

various APs. During the experiment, we can collect approximately 60 APs signals, in which some are from the other floors and their RSS is constantly weak. In order to eliminate the impact of these weak APs, we select 37 active APs as the significant features introduced in Section V-B. We specially arrange 5 APs with location changes. The test phones in the experiments are Huawei P9, Huawei P10, OPPO rs, Meizu Note6, Huawei Mate8 and Honor V10.

Our experiment consists of two phases: initial phase and testing phase. In the initial phase, the corridor is surveyed with a density of $0.8m \times 0.8m$ and the rooms crowded with facilities and desks are only surveyed in the aisles. The survey of the experimental areas produces 427 sample locations. For each sample location, we collect the RSS values from 37 APs as fingerprints. To mitigate the influence of noise, the RSS in each fingerprint is calculated in the average value over 30 seconds. In the testing phase, we change the locations of 5 selected APs to evaluate the effectiveness of AAIFU. We assume that the RSS of the unaltered APs in the radio map is consistent with the current environment, so that after updating the radio map, the RSS of the unaltered APs in the radio map remain unchanged.

In order to better illustrate the effect of radio map updating, we divide the test data into two parts, the points close to the locations of the altered APs (blue points in Figure 4) and the points away from the locations of the altered APs (red points in Figure 4).

B. EXPERIMENT RESULTS

1) FAST DETECTION OF ALTERED APS

A preliminary detection algorithm is designed in AAIFU to avoid time and power consuming for the altered APs identifying and radio map updating if there are no altered APs.

As introduced in Section IV-B, we calculate the difference Δv in Equation (10) between the estimated RSS according to the *GBDT* regression model and the measured RSS of each AP. Figure 5(a) indicates Δv in the test with five altered APs and without device diversity handling in crowdsourcing data. Figure 5(b) indicates Δv in the test with five altered APs and with device diversity handling in crowdsourcing data. We find that Δv from parts of APs are significantly larger than the others, especially for the altered ones, because the altered APs destroy the *GBDT* regression models generated from the original radio map. We can also get that, with the device diversity handling, the RSS differences of the altered APs get higher and the RSS differences of the unaltered APs keep small. This is because the compensation for the device differences in crowdsourcing data can help the data from different devices to better fit the regression model and make the regression prediction results reflect the actual situation more accurately, which help us to accurately detect the altered APs. Figure 5(c) shows Δv in the test without any altered APs and the device diversity handling in crowdsourcing data. In this figure, the majority of Δv keep small and few Δv get large due to the fluctuation of RSS. Therefore, in fast detection of the altered APs, we sort Δv of all APs in descending order and calculate the average value $\Delta \bar{v}$ of the first half of these differences.

Figure 6 shows $\Delta \bar{v}$ with the effect of device diversity handling and altered APs. We find that $\Delta \bar{v}$ in the test with the altered APs is clearly larger than the threshold φ set to 10 and $\Delta \bar{v}$ in the test without the altered APs is smaller than φ. With device diversity handling in crowdsourcing data, $\Delta \bar{v}$ in the test gets larger, which helps to detect the existence of altered APs. It is mainly because the device diversity handling can reduce the difference of RSS from different devices to well fit the regression model learned from the original radio map constructed by specific devices. Therefore, we deal with the device diversity in all the crowdsourcing data for the experiments in the following work. Moreover, the more altered APs in the crowdsourcing data, the more obvious the presence of the altered APs is. We can easily detect their existence, because with the increase of number of the altered APs, the model we have learned from the original radio map will more mismatch with the crowdsourcing data.

TABLE 1. Time used in each phase of AAIFU.

	Fast detection	Identification	Map updating
Time used(sec)	13.4	2738	7.9

We also compare the time spent in the phase of fast detection of the altered APs, the identification of the altered APs, and the radio map updating. We use 10,086 pieces of crowdsourcing data as the input in each phase. As shown in Table 1, we find that the identification of the altered APs takes most of the time, but the fast detection spends less time.

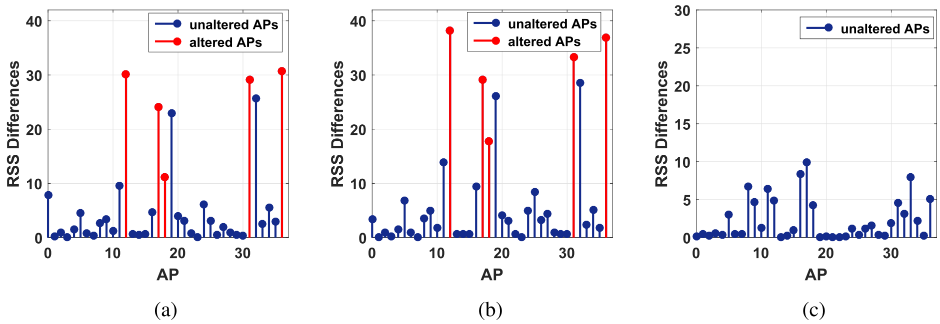


FIGURE 5. Δv **in two tests with and without altered APs. (a) Test with altered APs and no handling device diversity. (b) Test with altered APs and handling device diversity. (c) Test without altered APs and handling device diversity.**

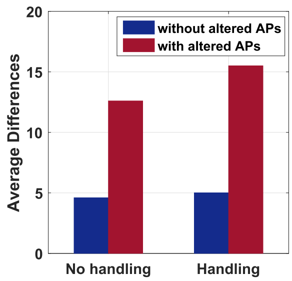


FIGURE 6. $\Delta \bar{v}$ **with the effect of device diversity handling and altered APs.**

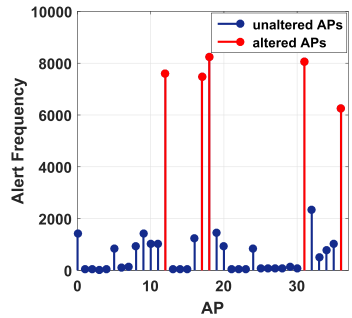


FIGURE 7. Alarmed frequencies with altered APs.

This is because fine identification takes very long time to train new regression models between the may-altered APs and the unaltered APs to prevent mislabeling the unaltered APs in the identification of the altered APs. While in the fast detection of the altered APs, the regression models only need to be built once after creating the original radio map, which is fixed and can be used directly. Therefore, the fast detection of the altered APs in AAIFU is an important phase for avoiding time and power consuming in the identification of the altered APs and the radio map updating.

2) IDENTIFICATION OF ALTERED APs

After the detection of the existence of the altered APs, AAIFU moves to identify the altered APs. In this step, for each piece of data in the crowdsourcing data set, we label the current AP as the altered AP and give an alarm if Δv is larger than the threshold γ. Then, we count alarms for each AP in the whole crowdsourcing data set. Figure 7 shows the results of the alarm counting in the tests with five altered APs. We find that the alarm times of the altered APs are significantly larger than those of the unaltered ones. It is because the altered APs do not match the regression model obtained from the original radio map, while the unaltered APs still fit it. Hence, Δv of the altered APs are high and the altered APs can be accurately labeled. Based on the alarm frequencies and the algorithm SDCM as introduced in Section IV-C, we can put all the altered APs into one cluster and the rest into the other cluster. We make seven independent experiments with different crowdsourcing data, i.e., six sets of crowdsourcing data separately from six different devices and all these crowdsourcing data together. We find that in all these seven experiments the identified altered APs are consistent with the actual moved APs.

3) RADIO MAP UPDATING

Before evaluating the radio map updating algorithm, we investigate the influence of the altered APs for different areas in the testing environment, i.e., the areas close to and away from the altered APs in which the testing locations are marked in Figure 4. Figure 8 indicates the CDF (Cumulative Distribution Function) of the positioning errors in these two

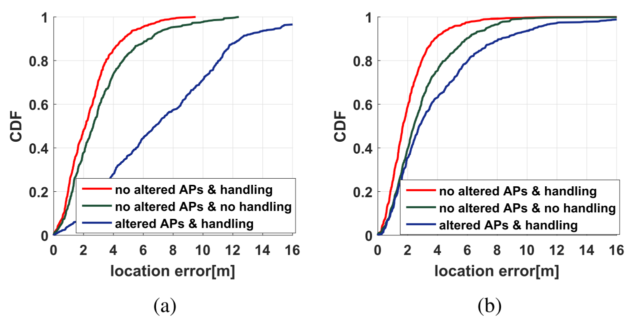


FIGURE 8. Influence of altered APs for different areas and the impact of whether handling device diversity on positioning data. (a) Test area close to altered APs. (b) Test area away from altered APs.

areas for the two tests with and without the altered APs. These positioning results are obtained based on the original radio map. It is easy to find that the positioning accuracy in both two areas degrades due to the altered APs. Moreover, in the areas close to the alerted APs, the degradation of positioning accuracy is larger, because the RSS values from the altered APs are the most significant features in the RSS vector, which introduces great impact on the positioning accuracy. Therefore, in the subsequent comparison of the radio map updating, we focus on the tests conducted in the areas close to the alerted APs, where the positioning accuracy is sensitive to the movement of APs and the effect of radio map updating can be clearly observed. In Figure 8, we also demonstrate the significance of the device diversity handling in the test data for positioning. The compensation for the device difference in the positioning data can make the data better fit the radio map constructed by the specific devices. Therefore, in the following experiments, the positioning data are first compensated by device diversity handling and then the positioning algorithms are conducted.

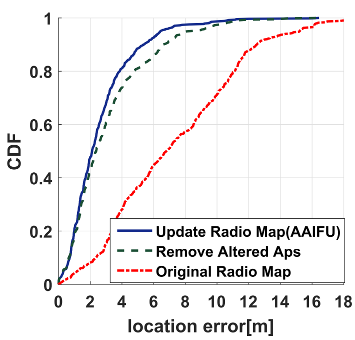


FIGURE 9. Positioning accuracy of different radio map.

TABLE 2. Positioning accuracy of different radio map.

	Mean(m)	Media(m)	90%(m)
Original Radio Map	7.2	6.5	12.5
Remove Altered APs	3.3	2.3	6.7
Update Radio Map (AAIFU)	2.6	2.1	5.3

Figure 9 shows the CDF of the positioning errors based on the different radio maps, including the original radio map, updated radio map constructed by AAIFU, and the radio map from removing the altered APs after identifying them. Table 2 summarizes the mean, median and 90% of the positioning accuracy. As shown in Figure 9, due to the existence of the altered APs, the original radio map no longer fits the signal space environment, resulting in a low positioning accuracy. Removing the altered APs from the original radio map can effectively improve the positioning accuracy because the interference of the altered APs is eliminated. However, this removing reduces the number of features (RSS) used in the fingerprint matching for the positioning. In order to effectively use the altered APs as the features for the positioning, our AAIFU learns the relationship between the altered APs and the unaltered APs via *GBDT* regression from the crowdsourcing data and update the radio map by taking the altered APs as the features for the positioning. The results in Figure 9 and Table 2 illustrate that the mean positioning accuracy of this AAIFU achieves 2.6m which outperforms the positioning approaches with the original radio map and the radio map after removing the altered APs by 63.9% and 21.2% respectively. We can conclude that our proposed AAIFU can effectively find the relationship between the unaltered and the altered APs based on *GBDT* regression model and update the radio map with high accuracy correspondingly.

4) INFLUENCE FACTORS OF RADIO MAP UPDATING

In the following, we investigate the following influence factors of the radio map updating.

a: NUMBER OF TREES IN GBDT REGRESSION

We first observe the effect of the number of estimators in the *GBDT* regression on the radio map updating. Figure 10 indicates the positioning accuracy of *GBDT* regression for the radio map updating with different number of estimators (trees). As the number of estimators increases, the mean positioning error of AAIFU gradually decreases and becomes stable when the number of trees is larger than 25. This is because *GBDT* regression is an ensemble learning approach boosting on a set of trees. In the regression, each individual tree is a weak estimator, and a model with limited number of trees is too simple to represent the relationship among the RSS values of different APs. Therefore, large number of trees can provide a more complex model to better represent the relationship. We note that a model with 25 trees is already complicated enough for our environment. Since the computation time for the radio map updating increases with the number of trees, we set this number in *GBDT* regression as 25 in our system.

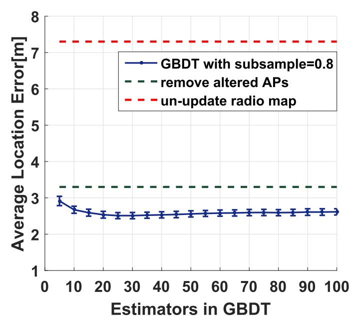


FIGURE 10. Average positioning error of updating map with different number of estimators.

b: NUMBER OF ALTERED APS

We further analyze the impact of the number of the altered APs on the radio map updating. It should be noted that in our experiment we have 37 APs, 5 of which are the altered APs and 32 are the unaltered ones. We investigate how the

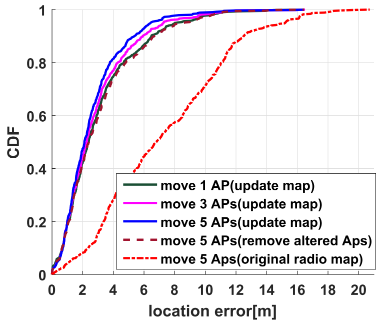
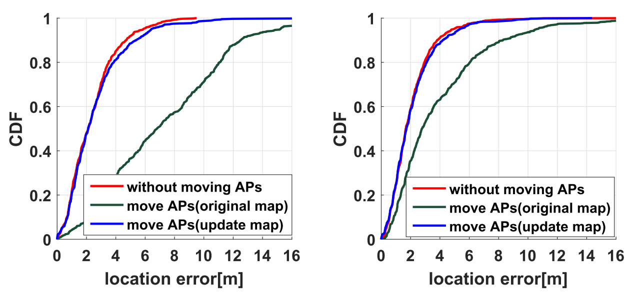


FIGURE 11. Impact of different number of APs on radio map updating.

number of altered APs influences the positioning accuracy in AAIFU. Note that in all tests the 32 unaltered APs keep the same and only the different number of altered APs are added. Figure 11 shows the CDF of mean positioning accuracy based on AAIFU by moving different number of APs. We find that as the number of the altered APs increases, the positioning accuracy based on the updated radio map gets better. It is mainly because that in a certain amount of the altered APs, *GBDT* regression in AAIFU can robustly learn the relationship between the altered and the unaltered APs and correspondingly update the radio map accurately. With more altered APs, the number of APs increases. Additionally, the RSS values of the altered APs after updating are the significant features in our tests and therefore the accuracy improves with the number of altered APs.

c: DIFFERENT POSITIONING AREAS

As observed in Figure 8, the positioning accuracy from the original radio map in the area close to and away from the altered APs is different. We find that the positioning accuracy based on the updated radio map in these two areas is also different. As shown in Figure 12, the improvement of the positioning accuracy using the updated radio map in the area close to the altered APs is much more significant than that in the area away from the altered APs. The reason is that the RSS values from the altered APs are the most significant features for the test data collected in the areas close to the altered APs. The *GBDT* regression can learn the relationship between the



(a) (b)

FIGURE 12. Impact of radio map updating on positioning accuracy of different areas. (a) Test area close to altered APs. (b) Test area away from altered APs.

altered and the unaltered APs from crowdsourcing data and accurately update the radio map by taking the altered APs as the significant features for the positioning.

d: DIFFERENT REGRESSION MODELS

We investigate the performance of different machine learning algorithms for the regression on the radio map updating in AAIFU. As shown in Figure 13 and Table 3, the mean and 90% of the positioning accuracy based on the updated radio map using *GBDT* regression are better than *SVR* and *KNN* regression algorithms. It is because *GBDT* regression combines multiple weak learners into a strong learner with high generalization ability and better representativeness to solve practical problems as mentioned in Section III, while *SVR* and *KNN* are the single regression learners. By comparing different regression models, we present that our AAIFU system is effective in radio map updating. Since our focus is on positioning accuracy and the *GBDT* regression has better performance, we adopt *GBDT* regression in the AAIFU system.

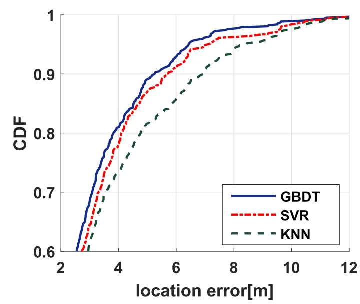


FIGURE 13. Positioning accuracy using different machine learning models.

TABLE 3. Positioning accuracy of different regression models.

	GBDT	SVR	KNN
Mean(m)	2.6	2.8	3.1
90%(m)	5.2	5.8	7.0

e: DEVICE DIVERSITY ON CROWDSOURCING DATA

In Section VI-B3 and Figure 8, we provide the effect with the device diversity handling in the positioning data. In addition to its influence on positioning accuracy, the device diversity in the crowdsourcing data will affect the phase of radio map updating and correspondingly influence the positioning accuracy. As shown in Figure 14, after handling the device diversity in the crowdsourcing data, we can train the radio map more accurately and correspondingly achieve a higher positioning accuracy. On the other hand, we find that our proposed radio map updating based on *GBDT* is robust to the device diversity problem. The reason is that the regularization is conducted in *GBDT* for radio map updating and hence it can attenuate the influence of noise in crowdsourcing data and also the impact of device diversity.

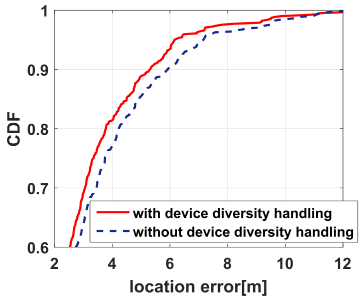


FIGURE 14. Positioning accuracy with and without device diversity handling on crowdsourcing data.

VII. CONCLUSIONS

In this work, we propose an indoor positioning system AAIFU which can automatically detect altered APs and update radio map with crowdsourcing data handled device diversity. In our study, we find that the deviation between the relationships among RSS values of different APs in crowdsourcing data and an original radio map are significantly larger for altered APs than unaltered ones. Therefore, we propose a novel crowdsourcing altered APs detection algorithm based on differences between measured RSS values in crowdsourcing data and predicted values based on a set of *GBDT* regression model for different APs in the original radio map. The algorithm can detect altered APs in our experiments with an accuracy of 100%. We deeply use new crowdsourced data to learn an updated mapping function between the RSS of the altered APs and the unaltered APs based on another set of *GBDT* regression model to further update entire radio map. We conduct a set of experiments in large a teaching building. The experimental results show that our proposed AAIFU can effectively detect altered APs and accurately updates entire radio map. Based on the updated radio map, the positioning accuracy of AAIFU gets significantly improved compared to the fingerprinting approach with the original radio map (by 63.9%) and the radio map by directly removing the altered APs (by 21.2%).

REFERENCES

[1] S. He, W. Lin, and S.-H. G. Chan, "Indoor localization and automatic fingerprint update with altered AP signals," *IEEE Trans. Mobile Comput.*, vol. 16, no. 7, pp. 1897–1910, Jul. 2017.

[2] A. M. Ladd, K. E. Bekris, A. P. Rudys, D. S. Wallach, and L. E. Kavraki, "On the feasibility of using wireless Ethernet for indoor localization," *IEEE Trans. Robot. Autom.*, vol. 20, no. 3, pp. 555–559, Jun. 2004.

[3] Z. Li and T. Braun, "Passively track WiFi users with an enhanced particle filter using power-based ranging," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7305–7318, Nov. 2017.

[4] M. Zhou, Y. Wei, Z. Tian, X. Yang, and L. Li, "Achieving cost-efficient indoor fingerprint localization on WLAN platform: A hypothetical test approach," *IEEE Access*, vol. 5, pp. 15865–15874, 2017.

[5] J. Zhang, G. Han, N. Sun, and L. Shu, "Path-loss-based fingerprint localization approach for location-based services in indoor environments," *IEEE Access*, vol. 5, pp. 13756–13769, 2017.

[6] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. Conf. Comput. Commun., 19th Annu. Joint Conf. IEEE Comput. Commun. Societies (INFOCOM)*, vol. 2, Mar. 2000, pp. 775–784.

[7] S. Han, C. Zhao, W. Meng, and C. Li, "Cosine similarity based fingerprinting algorithm in WLAN indoor positioning against device diversity," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 2710–2714.

[8] D. Han, S. Jung, M. Lee, and G. Yoon, "Building a practical Wi-Fi-based indoor navigation system," *IEEE Pervasive Comput.*, vol. 13, no. 2, pp. 72–79, Apr./Jun. 2014.

[9] M. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *Proc. Int. Conf. Mobile Syst.* New York, NY, USA: ACM, 2005, pp. 205–218.

[10] P. Mirowski, D. Milioris, P. Whiting, and T. K. Ho, "Probabilistic radio-frequency fingerprinting and localization on the run," *Bell Labs Tech. J.*, vol. 18, no. 4, pp. 111–133, 2014.

[11] D. Milioris, L. Kriara, A. Papakonstantinou, G. Tzagkarakis, P. Tsakalides, and M. Papadopouli, "Empirical evaluation of signal-strength fingerprint positioning in wireless LANs," in *Proc. 13th ACM Int. Conf. Modeling, Anal., Simulation Wireless Mobile Syst.* New York, NY, USA: ACM, 2010, pp. 5–13.

[12] Z. Li, T. Braun, X. Zhao, Z. Zhao, F. Hu, and H. Liang, "A narrow-band indoor positioning system by fusing time and received signal strength via ensemble learning," *IEEE Access*, vol. 6, pp. 9936–9950, 2018.

[13] L. Pei, M. Zhang, D. Zou, R. Chen, and Y. Chen, "A survey of crowd sensing opportunistic signals for indoor localization," *Mobile Inf. Syst.*, vol. 2016, Apr. 2016, Art. no. 4041291.

[14] B. Wang, Q. Chen, L. T. Yang, and H.-C. Chao, "Indoor smartphone localization via fingerprint crowdsourcing: Challenges and approaches," *IEEE Wireless Commun.*, vol. 23, no. 3, pp. 82–89, Jun. 2016.

[15] Q. Chen and B. Wang, "Finccm: Fingerprint crowdsourcing, clustering and matching for indoor subarea localization," *IEEE Wireless Commun. Lett.*, vol. 4, no. 6, pp. 677–680, Dec. 2015.

[16] J. Niu, B. Wang, L. Cheng, and J. J. P. C. Rodrigues, "WicLoc: An indoor localization system based on WiFi fingerprints and crowdsourcing," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 3008–3013.

[17] S.-H. Jung and D. Han, "Automated construction and maintenance of Wi-Fi radio maps for crowdsourcing-based indoor positioning systems," *IEEE Access*, vol. 6, pp. 1764–1777, 2018.

[18] Y. Sun, W. Meng, C. Li, N. Zhao, K. Zhao, and N. Zhang, "Human localization using multi-source heterogeneous data in indoor environments," *IEEE Access*, vol. 5, pp. 812–822, 2017.

[19] C. Wu, Z. Yang, C. Xiao, C. Yang, Y. Liu, and M. Liu, "Static power of mobile devices: Self-updating radio maps for wireless indoor localization," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr./May 2015, pp. 2497–2505.

[20] Z. Li, X. Zhao, and H. Liang, "Automatic construction of radio maps by crowdsourcing PDR traces for indoor positioning," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.

[21] J. Dong, M. Noreikis, Y. Xiao, and A. Yla-Jaaski, "Vinav: A vision-based indoor navigation system for smartphones," *IEEE Trans. Mobile Comput.*, to be published. doi: 10.1109/TMC.2018.2857772.

[22] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, "LANDMARC: Indoor location sensing using active RFID," in *Proc. 1st IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Mar. 2003, pp. 407–415.

[23] P. Krishnan, A. S. Krishnakumar, W.-H. Ju, C. Mallows, and S. N. Gamt, "A system for LEASE: Location estimation assisted by stationary emitters for indoor RF wireless networks," in *Proc. 23rd Annu. Joint Conf. IEEE Comput. Commun. Societies (INFOCOM)*, vol. 2, Mar. 2004, pp. 1001–1011.

[24] M. M. Atia, A. Noureldin, and M. J. Korenberg, "Dynamic online-calibrated radio maps for indoor positioning in wireless local area networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 9, pp. 1774–1787, Sep. 2013.

[25] Y. Zhuang, Z. Syed, Y. Li, and N. El-Sheimy, "Evaluation of two WiFi positioning systems based on autonomous crowdsourcing of handheld devices for indoor navigation," *IEEE Trans. Mobile Comput.*, vol. 15, no. 8, pp. 1982–1995, Aug. 2016.

[26] P. Wilk, J. Karciarz, and J. Swiatek, "Indoor radio map maintenance by automatic annotation of crowdsourced Wi-Fi fingerprints," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Oct. 2015, pp. 1–8.

[27] J. Lin, B. Wang, G. Yang, and M. Zhou, "Indoor localization based on weighted surfacing from crowdsourced samples," *Sensors*, vol. 18, no. 9, p. 2990, 2018.

[28] C. Wu, Z. Yang, Y. Liu, and W. Xi, "WILL: Wireless indoor localization without site survey," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 4, pp. 839–848, Apr. 2013.

[29] G. Shen, Z. Chen, P. Zhang, T. Moscibroda, and Y. Zhang, "Walkie-Markie: Indoor pathway mapping made easy," in *Proc. NSDI*, 2013, pp. 85–98.

[30] B. Zhou, Q. Li, Q. Mao, W. Tu, X. Zhang, and L. Chen, "ALIMC: Activity landmark-based indoor mapping via crowdsourcing," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2774–2785, Oct. 2015.

[31] C. Luo, H. Hong, and M. C. Chan, "PiLoc: A self-calibrating participatory indoor localization system," in *Proc. IEEE IPSN*, Apr. 2014, pp. 143–153.

[32] L. Li, G. Shen, C. Zhao, T. Moscibroda, J.-H. Lin, and F. Zhao, "Experiencing and handling the diversity in data density and environmental locality in an indoor positioning service," in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.* New York, NY, USA: ACM, 2014, pp. 459–470.

[33] S.-H. Fang, C.-H. Wang, S.-M. Chiou, and P. Lin, "Calibration-free approaches for robust Wi-Fi positioning against device diversity: A performance comparison," in *Proc. IEEE 75th Veh. Technol. Conf. (VTC Spring)*, May 2012, pp. 1–5.

[34] A. Goswami, L. E. Ortiz, and S. R. Das, "WiGEM: A learning-based approach for indoor localization," in *Proc. 7th Conf. Emerg. Netw. EXperiments Technol.* New York, NY, USA: ACM, 2011, p. 3.

[35] J.-G. Park, D. Curtis, S. Teller, and J. Ledlie, "Implications of device diversity for organic localization," in *Proc. IEEE INFOCOM*, Apr. 2011, pp. 3182–3190.

[36] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, 2001.

[37] J. H. Friedman, "Stochastic gradient boosting," *Comput. Statist. Data Anal.*, vol. 38, no. 4, pp. 367–378, 2002.

[38] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY, USA: Springer, 2009.

[39] G. Ridgeway, "Generalized boosted models: A guide to the gbm package," *Update*, vol. 1, no. 1, pp. 1–15, 2007.

[40] Y.-C. Huang, C.-W. Yi, W.-C. Peng, H.-C. Lin, and C.-Y. Huang, "A study on multiple wearable sensors for activity recognition," in *Proc. IEEE Conf. Dependable Secure Comput.*, Aug. 2017, pp. 449–452.

[41] T. Zebin, P. J. Scully, and K. B. Ozanyan, "Evaluation of supervised classification algorithms for human activity recognition with inertial sensors," in *Proc. IEEE SENSORS*, Oct./Nov. 2017, pp. 1–3.

[42] Z. Feng, L. Mo, and M. Li, "A random forest-based ensemble method for activity recognition," in *Proc. 37th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Aug. 2015, pp. 5074–5077.

[43] Z. S. Patrous, "Evaluating XGBoost for user classification by using behavioral features extracted from smartphone sensors," M.S. thesis, School Elect. Eng. Comput. Sci., KTH Roy. Inst. Technol., Stockholm, Sweden, 2018.

[44] Y. Yuan, S. Li, X. Zhang, and J. Sun, "A comparative analysis of SVM, Bayes and GBDT for data faults detection in WSNs," in *Proc. IEEE Int. Conf. Softw. Qual., Rel. Secur. Companion (QRS-C)*, Jul. 2018, pp. 394–399.

[45] Z. Xiao-yu, Y. Yun-yang, L. Yi-an, and G. Shang-bing, "Flame detection based on GBDT feature for building," in *Proc. Int. Smart Cities Conf. (ISC)*, Sep. 2017, pp. 1–5.

[46] L. Breiman, "Pasting small votes for classification in large databases and on-line," *Mach. Learn.*, vol. 36, nos. 1–2, pp. 85–103, 1999.

[47] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, 1996.

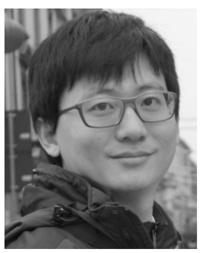
[48] I. Barandiaran, "The random subspace method for constructing decision forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 1–22, Aug. 1998.

[49] G. Louppe and P. Geurts, "Ensembles on random patches," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Bristol, U.K.: Springer, 2012, pp. 346–361.

[50] G. F. Jenks, "The data model concept in statistical mapping," *Int. Yearbook Cartography*, vol. 7, no. 1, pp. 186–190, 1967.

JIANGHONG YANG received the bachelor's degree in communication engineering from Jilin University, China, in 2017, where he is currently pursuing the master's degree with the College of Communication Engineering. His research interest includes wireless communication networks and indoor localization techniques.

XIAOHUI ZHAO received the B.E. and M.S. degrees in electrical engineering from the Jilin University of Technology, China, in 1982 and 1989, respectively, and the Ph.D. degree in applied mathematics and control theory from the Universite de Technologie de Compiegne, France, in 1993. He was a Postdoctoral Researcher with the Institute of Automatic Control, Southeast University, China, from 1994 to 1996, and he has been a Senior Visiting Scholar for half a year with the Laboratoire d' Informatique, Universite de Pierre et Marie Curie, France, in 2006. He is currently a Professor of communication engineering with Jilin University. He has authored or co-authored over 100 papers of journals and international conferences, two books in the fields of cognitive radio and signal processing. His research interests include wireless communication, cognitive radio, and adaptive signal processing. He currently serves as an Editor of the *Chinese Journal of Communications*, the *Chinese Journal of Signal Processing*, and the *Journal of China Universities of Posts and Telecommunications*.



ZAN LI received the Ph.D. degree from the University of Bern, Switzerland, in 2016. Afterwards, he was with the Alibaba Group, China, as a Senior Algorithm Engineer, from 2016 to 2017. Since 2017, he has been with the College of Communication Engineering, Jilin University, as an Assistant Professor. He was a recipient of the Best Paper Award from the WMNC 2014. His Ph.D. dissertation won the Fritz-Kutter Award 2016 (the best Ph.D. dissertation in Swiss Universities).