ZIL: An Energy-Efficient Indoor Localization System Using ZigBee Radio to Detect WiFi Fingerprints

Published in:
IEEE Journal on Selected Areas in Communications

Document Version:
Peer reviewed version

Queen's University Belfast - Research Portal:
Link to publication record in Queen's University Belfast Research Portal

ZIL: An Energy-Efficient Indoor Localization System Using ZigBee Radio to Detect WiFi Fingerprints

Jianwei Niu, *Senior Member, IEEE,* Bowei Wang, Lei Shu, *Member, IEEE,* Trung Q. Duong, *Member, IEEE,* Yuanfang Chen, *Member, IEEE*

Abstract—WiFi fingerprint-based indoor localization has received considerable attention to enjoy higher deployment practicability, because of the ubiquitous APs (Access Points) and WiFi-enabled smart devices. In existing WiFi-based localization methods, smart mobile devices consume quite a lot of power as WiFi interfaces need to be used for frequent AP scanning during the localization process. In this work, we design an energy-efficient indoor localization system called ZIL (ZigBee assisted Indoor Localization) based on WiFi fingerprints via ZigBee interference signatures. ZIL uses ZigBee interfaces to collect mixed WiFi signals which include non-periodic WiFi data and periodic beacon signals. However, WiFi APs cannot be identified from these WiFi signals by ZigBee interface directly. To address this issue, we propose a method, including RSS quantification and normalization, to detect WiFi APs and their signal strengths to form WiFi fingerprints from the signals collected by ZigBee interface. We propose a novel fingerprint matching algorithm to align a pair of fingerprints effectively. To improve the localization accuracy, we design the KNN (K-Nearest Neighbor) classification method with three different weighted distances and find that the KNN algorithm with the Manhattan distance performs best. Extensive experimental results show that ZIL implemented on TelosB motes can achieve the localization accuracy of 87% which is competitive compared to state-of-the-art WiFi fingerprint-based approaches, and save energy by about 68% on average compared to the approach based on WiFi interface.

Index Terms—ZigBee, indoor localization, energy saving, WiFi fingerprint.

I. INTRODUCTION

Indoor localization techniques have undergone a rapid development recently, with many innovative methods springing up, most of which still depend on extra facilities. The increasingly deployed WiFi APs (Access Points) enable users to access Internet via wireless networks pervasively. Hence, many existing indoor localization systems [1] [2] take advantage of the off-the-shelf WiFi APs to estimate the locations of WiFi-enabled devices, such as laptops, mobile phones, pads, etc. WiFi fingerprint-based indoor localization method has become a better choice as it requires no extra infrastructures.
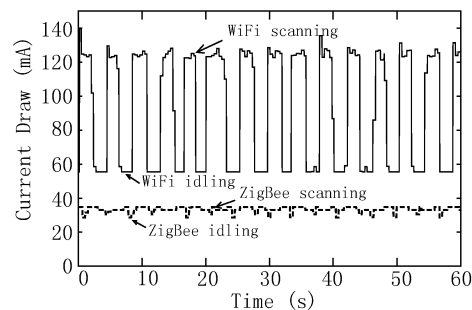
Fig. 1. The measurements of current draw of ZigBee compatible TelosB motes and WiFi interface in scanning mode.

It is known that the energy consumption of wireless interfaces has long been an essential problem for WiFi-based indoor localization systems. To achieve real-time localization, WiFi-enabled devices have to constantly scan the WiFi channels, resulting in high power consumption and reduced battery runtime (e.g., mobile phones consume more than 40% of the whole power on WiFi according to [3]). Fig. 1 illustrates the current draw comparison of the scanning WiFi and ZigBee interfaces on a laptop when it runs a fingerprint-based localization algorithm [1]. As such, the methods in [4] [5] are proposed to employ ZigBee for WiFi fingerprinting to realize high energy efficiency and low estimation error. Although mobile devices could save energy by avoiding excessive listening and scanning operations of WiFi interfaces, it is still a challenge to make proper use of ZigBee assisted WiFi-based fingerprinting for energy saving.

In this paper, we propose a new ZigBee assisted WiFi fingerprint-based localization system called ZIL (ZigBee assisted Indoor Localization). Compared with existing systems, our main contributions are as follows:

- We utilize the low-power 802.15.4 ZigBee wireless interface (instead of 802.11 WiFi interface) to detect and recognize WiFi APs, which significantly saves energy by avoiding the excessive listening and scanning operations using WiFi interfaces. Moreover, we present a method to quantize and normalize the fingerprints.
- We design a localization module, which is composed of two parts: a novel fingerprint matching algorithm and three variants of the KNN (K-Nearest Neighbor) algorithm to achieve the best performance of our approach.

- We implement a fingerprint-based indoor localization system called ZIL. Our experimental results show that ZIL could achieve the localization accuracy of 87% and save energy of 68% as compared to the WiFi-based method.

The rest of the paper is organized as follows. In Section II, literature review is provided. Section III describes the system architecture of ZIL, while Section IV presents the detailed design of ZIL. The experimental results are shown in Section V. Finally, we draw conclusions in Section VI.

II. RELATED WORK

Many approaches have been proposed for indoor localization in recent years. One class of the approaches is the RF signal range-based indoor localization, including the TOA (Time-of-Arrival) [6], TDOA (Time-Difference-of-Arrival) [7], AOA (Angle-of-Arrival) [8], DOA (Direction-of-Arrival) [9] and RSS (Received Signal Strength). For example, the Calamari [10] designed by Kamin Whitehouse is based on the TDOA with ultrasonic. Another class is the range-free indoor localization, such as DV-Hop (Distance Vector-Hop) [11], MDS (MultiDimensional Scaling) [12], APIT (Approximate Point-In Triangulation Test) [13], etc. HiRLoc [14] is a high-resolution range-free localization scheme, which determines the locations of sensors based on the intersection of the areas covered by the beacons transmitted by multiple reference points.

Recently, fingerprint-based localization has become more popular. The fingerprint-based localization process includes two main phases. The first phase, called off-line or training phase, involves capturing and storing the signatures and features of each reference location into a fingerprint database. The second phase, called online or testing phase, comprises the estimation of an unknown location by mapping the measured fingerprint with the database. There are three key aspects for fingerprint-based localization: fingerprint collection, fingerprint selection and fingerprint matching. The first aspect normally requires an exhaustive site survey to build an RSS fingerprint map. Horus system [15] scans the WiFi channel to collect fingerprints and identifies different causes for the WiFi channel variations. Surroundsense [16] utilizes ambient sound, light, color and WiFi signals to form identifiable fingerprints for logical localization. Some researchers proposed acoustics [17] [18] and social interaction [19] [20] as the environment features. As for the second aspect, the existing work like [21] propose several methods to select the fingerprints for localization. For example, the local strongest signal points are selected to form fingerprints; fisher criterion is used to quantify the discrimination ability; random combination is used to dynamically create a fingerprint matrix based on a certain criterion. For the third aspect, Bayesian inference [22] [23], Euclidian distance [22] [24] [25], Manhattan distance [26] and compressive sensing [21] [27] are employed to match the corresponding fingerprints.

Due to low-power consumption and low cost, ZigBee [28] has been adopted for indoor localization. The products with ZigBee interface [29] can be easily implemented and consume little power to operate for years. Many ZigBee modules have USB interfaces and hence can be easily connected with mobile devices. Moreover, some cell phone manufacturers (e.g., Nokia and Pantech&Curitel [30]) also provide smart phones with built-in ZigBee interfaces.

Recent work has been done on the dual ZigBee-WiFi [4] [5] to establish an efficient connection between WiFi APs and mobile phones integrated with ZigBee interfaces. This method has addressed the challenges of time synchronization, frequency switching and frame collisions, which could deliver high energy efficiency, low estimation error, and real-time connectivity. Xing et al. investigated the co-existence of ZigBee and WiFi [31], and developed ZiFi [32] by utilizing Zigbee-WiFi interferences.

To determine the identity (i.e., MAC address) of each AP, our previous work ZiFind [33] added mappers to collect beacon signals via WiFi interface, recording the time information and BSSIDs (Basic Service Set Identifiers), to match and identify the signals measured by clients and improve the accuracy. However, introducing mappers inevitably complicates the system implementation and increases the cost. In this work, we employ ZigBee radios for WiFi fingerprint-based indoor localization, using ZigBee interfaces on the clients to capture beacon frames broadcasted by WiFi APs, without the mappers designed in ZiFind. We adopt the RSS quantification and normalization to recognize and preprocess the WiFi fingerprints. Furthermore, we propose a fingerprint-based localization module, which consists of a novel fingerprint matching method and three variants of weighted KNN algorithm to improve localization accuracy. Our extensive experimental results show that our approach outperforms the state-of-the-art WiFi fingerprint-based indoor localization methods in terms of energy consumption while keeping a competitive localization accuracy.

III. SYSTEM ARCHITECTURE

The system architecture of ZIL is shown in Fig. 2. ZIL is based on the client-server architecture: the client could be a WiFi-enabled mobile device equipped with a ZigBee interface, while the server could be a desktop or a laptop, calculating the location of the client. There are two phases for the localization process: training and testing phases. During the training phase, the client utilizes its ZigBee interface to capture WiFi signals at each reference location. Fig. 3 shows the WiFi signals collected by ZigBee interfaces in a whole day. The RSS quantification and normalization modules process these data and generate the WiFi fingerprints of each location. The client sends the processed fingerprints to the server through its WiFi interface. The server stores them to form a fingerprint database. The database will be updated to adapt the changes of the radio environment. During the testing phase, the client collects WiFi signals with its ZigBee interface to form localization requests and sends them to the server through its WiFi interface. The server receives the requests and uses the localization module to calculate the location of the client. The whole process will be discussed in detail in the next section.

Some mobile devices with ZigBee interfaces are shown in Fig. 4. TazTag company produced the world's first tablet inte-
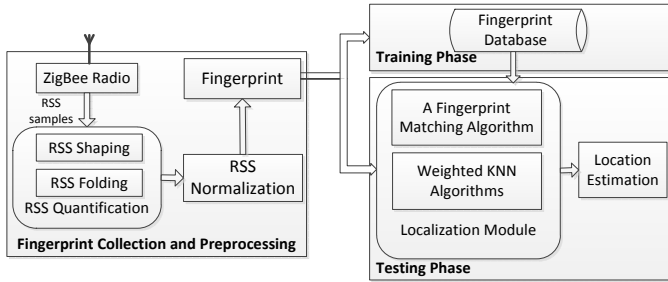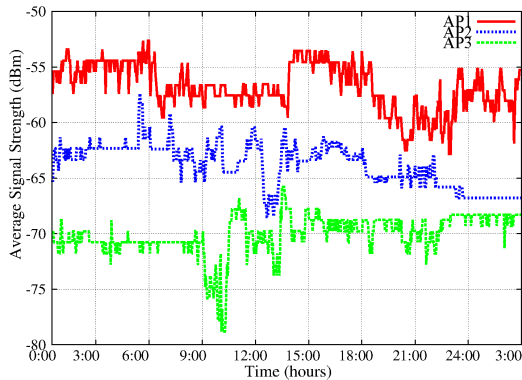
Fig. 2. The system architecture of ZIL.

Fig. 3. WiFi signals of three APs measured by ZigBee interface in one day.

grating NFC (Near Field Communication) and ZigBee called TazPad V2 (Fig. 4 (a)) [34]. Fig. 4 (b) shows the world's first mobile phone equipped with a ZigBee module [30]. We implemented our system based on TelosB motes shown in Fig. 4 (c).

IV. DESIGN OF ZIL

As GPS (Global Positioning System) cannot properly work indoors, pundits have proposed many indoor localization schemes based on WiFi fingerprinting. Due to limited battery energy supply, frequent channel scanning for WiFi APs is not an ideal option. In our system, we equip the client with a ZigBee interface to collect periodic WiFi beacon signals, which substantially reduces the energy consumption. A novel localization module is implemented on the server to improve the localization accuracy. In this section, we will discuss each
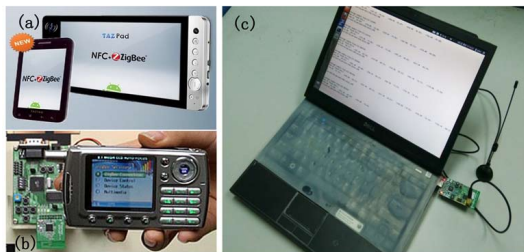

Fig. 4. (a) The world's first Android tablet PC integrated with ZigBee radio [34]. (b) The world's first mobile phone integrated with a ZigBee interface produced by Pantech&Curitel [30]. (c) A laptop client with a ZigBee interface via a USB interface in our testbed.

part of our system in detail.

A. Design of Client

In our testbed, the client device (Fig. 4 (c)) is a WiFi-enabled laptop equipped with a ZigBee interface, which continuously collects beacon frames of nearby WiFi APs. As the ZigBee interface keeps scanning the 802.11 channels for a certain period of time, most available APs in the vicinity can be detected. We set the scanning time to 3s which is long enough to thoroughly scan all the available WiFi APs. The collected RSS samples will then be quantified and normalized to form fingerprints for each location.

As the ZigBee interface cannot decode the beacon frames to obtain the MAC (Media Access Control) address or BSSID of each AP, it is impossible to directly recognize APs from the RSS samples. Therefore we design a folding method to extract some features from the collected RSS samples to distinguish each AP. As shown in Fig. 2, our method consists of two steps: the first step is the RSS quantification to concert the chaotic RSS samples, and the second step is RSS normalization. The RSS quantification includes the RSS shaping and the RSS folding. The RSS shaping is to quantify the signal strength to "0" or "1", and the RSS folding removes the aperiodic radio signals (e.g., the data signals transmitted by WiFi devices) and keeps the periodic WiFi beacon signals. The results of the RSS folding are then normalized to obtain the real beacon frames to form the location fingerprints. The fingerprints will then be sent to the server through the WiFi interface.

1) RSS Quantification: The period of WiFi beacon broadcasting is 102.4 ms (IEEE 802.11 specification), i.e., an AP broadcasts one beacon frame every 102.4 ms. In our experiments, we set the RSS sampling period to 122 μs, and therefore about 839 RSS samples will be captured in each beacon period. According to the IEEE 802.11 specification, the time for transmitting a beacon frame is greater than 400 μs, so our sampling scheme will not miss any beacon frame [35].

As shown in Fig. 5, the RSS samples collected by ZigBee interface include WiFi data signals and WiFi beacon signals, and the WiFi APs cannot be distinguished directly. We designed RSS Quantification and RSS Normalization to identify the WiFi APs from these RSS samples.
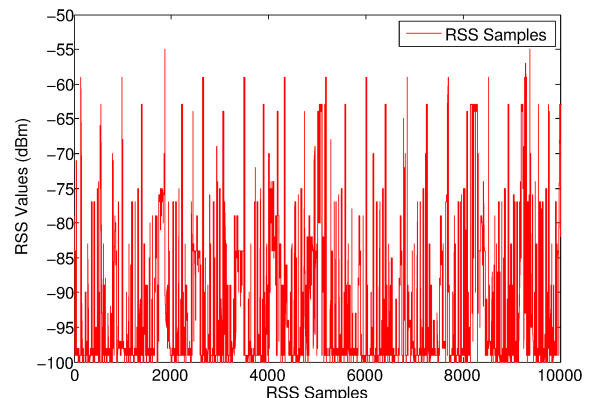

Fig. 5. A series of RSS samples collected by ZigBee interface.

We define L as the number of beacon periods for capturing WiFi fingerprints and N as the number of RSS samples collected in one beacon period. Then the process would cost $L * 102.4$ms and collect $L * N$ RSS samples. We group the RSS samples together and define them as a matrix $S[i][j]$ $(i \in [1, N], j \in [1, L])$. To filter out the beacon frames with poor signal quality, we define a threshold as -90 dBm. We use Eq. (1) to quantify the signal strength into "0" or "1".

$$RSSI = \begin{cases} 1, & \text{for } RSSI \geq \text{-90 dBm} \\ 0, & \text{for } RSSI < \text{-90 dBm} \end{cases} \quad (1)$$

Then we define a matrix $S^*[i][j] (i \in [1, N], j \in [1, L])$ to store the shaped RSS samples.

Then we fold the shaped RSS samples. Fig. 6 shows an example after RSS shaping and folding, which amplifies the beacon signals and reduces the noise. In Fig. 6, we assume that there are five APs and the scenarios for three beacon intervals are displayed. Then we define the $PHASE$ of each AP as:
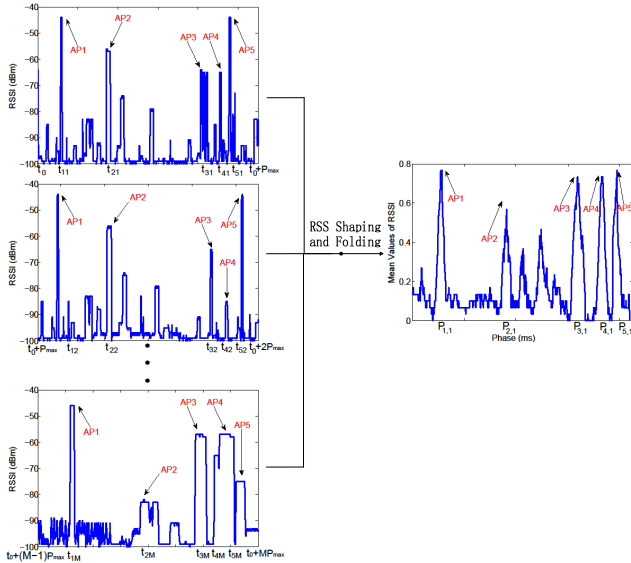

Fig. 6. The process of RSS shaping and folding.

$$P_{i,k} = \frac{1}{M} \sum_{j=1+(k-1)M}^{kM} (\Delta t_{ij} - \lfloor \Delta t_{ij}/P_{max} \rfloor \times P_{max}) \quad (2)$$

where $P_{i,k}$ $(i \in [1, N], k \in [1, K])$ is the $PHASE$ of the i-th AP in the k-th fingerprint, P_{max} is the beacon interval (102.4ms) and $\Delta t_{ij} = t_{ij} - t_0 (i \in [1, N])$. t_{ij} is the time stamp of the i-th AP in the j-th beacon interval and t_0 is the time origin. M is the number of beacon intervals, which is an empirical constant (30 in our experiments).

Then we adopt RSS folding to remove aperiodic noise (WiFi data signals) and keep periodic AP beacon signals denoted by $R'_{i,k} \subseteq [0, 1]$ using Eq. (3). We fold each M (30 in our experiments) beacon periods and obtain $K = \lfloor L/M \rfloor$ WiFi fingerprints.

$$R'_{i,k} = \frac{1}{M} \sum_{j=1+(k-1)M}^{kM} S'[i][j] \quad (i \in [1, N], \ k \in [1, K]) \quad (3)$$

$R'_{i,k}$ implies the probability of AP i appearing in the k-th fingerprint. Then we can obtain the average real RSSI of the i-th AP in the k-th fingerprint ($R_{i,k}$) by folding the matrix of raw RSS samples $S[i][j]$ using Eq. (4).

$$R_{i,k} = \frac{1}{M} \sum_{j=1+(k-1)M}^{kM} S[i][j] \quad (i \in [1, N], \ k \in [1, K]) \quad (4)$$

2) RSS Normalization: Not all nonzero peaks in $R'_{i,k}$ represent real WiFi APs after the above-mentioned folding. However, the probability that the peaks with large values are real WiFi APs is high. Therefore we define a threshold α (an empirical constant, 0.6 in our experiments) to filter out the fake beacon frames, as shown in Fig. 7. Then we retain the peaks above α and remove the rest. We term this process "RSS normalization".
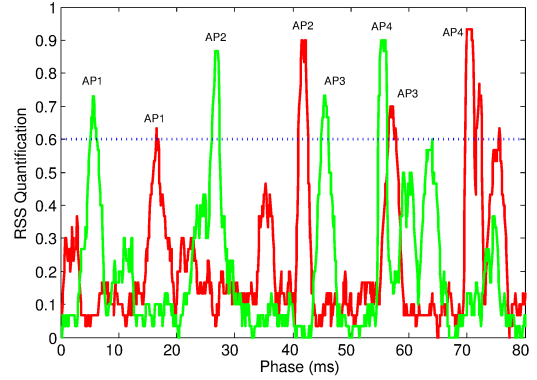
Fig. 7. Normalizing Transition of two folded RSS series.

Formally, we define $F_k = \{(P_{i,k}, R_{i,k})|i \in [1, N'], \ k \in [1, K]\}$ as the k-th WiFi fingerprint, where $P_{i,k}$ and $R_{i,k}$ are the $PHASE$ and $RSSI$ of the i-th AP in the k-th fingerprint, respectively, and N' denotes the number of APs after RSS normalization. The time stamp of the k-th fingerprint is denoted by T_k which can be calculated by Eq. (5), where $t_{i,k}$ is the time stamp of the i-th AP.

$$T_k = \frac{1}{N'} \sum_{i=1}^{N'} t_{i,k} \quad (5)$$

Then we define $F'_k = \{F_k, T_k\}$, which will be sent to the fingerprint database on the server of ZIL.

B. Design of Server

The server is the core component of our system, whose primary function is to build the fingerprint database and run the localization module. During the training phase, the server receives and stores the labelled fingerprint data from the clients. In the testing phase, the localization request which includes a

fingerprint (consisting of 30 beacon periods, which cost about 3s to collect) from a client is sent to the server in real time, and processed using our proposed fingerprint matching algorithm. Then we calculate the distance (similarity) between each pair of fingerprints using three weighted distances. The following sections will discuss the localization process in detail.

In the localization module, a novel fingerprint matching algorithm is proposed to match the fingerprints from the database with the localization request. The main idea of the algorithm is to execute cyclic shifts to find the optimum matching. After cyclical shifting, some locations still miss some APs for the following two reasons: 1) The $RSSI$ may fold on the mistaken $PHASE$ and form a false positive signal; 2) Due to the signal delay and clock drift, the signals that should be kept may have been removed in the RSS normalization process. To address the problem, we insert some virtual APs (their RSSIs are $-129dBm$ in our experiments as interference signals [36]) to make up for the missing APs.

Given two fingerprints denoted as F_a and F_b, and the numbers of APs in F_a and F_b as N_a and N_b, respectively. We give an example of matching two fingerprints in Fig. 8. The rationale of the cyclical shifting is to align each pair of APs in F_a and F_b, and record the number of matched APs in each comparison. After all comparisons, the one with the largest number of matched APs is the best match.

The pseudocode of our fingerprint matching algorithm is illustrated in Algorithm IV.1. The time complexity of the fingerprint matching algorithm is $O(N^2)$, where N is the number of APs. The matching process is explained as follows:

1. Initially AP_{a1} is aligned with AP_{b1} and c points to AP_{a1} in F_a, as shown in Fig. 8 (a). F_b is cyclically shifted to let d point from the first AP to the last AP in order. Fig. 8 (b) shows the second cyclical shifting as d points to the second AP in F_b. In each cyclical shifting of F_b, F_a and F_b are compared to calculate the number of matched APs. Formally, we define a matrix H, whose elements are the number of matched APs between the two fingerprints, as shown in Eqs. (6), (7) and (8):

$$H = (h_{cd})_{N_a \times N_b} \qquad (6)$$

where $c \in [0, N_a), d \in [0, N_b)$. The number of matched APs for each comparison is denoted as h_{cd} which can be calculated by,

$$h_{cd} = \sum_{m=0}^{N_a-1} \sum_{n=0}^{N_b-1} S(|\Delta m - \Delta n|) \qquad (7)$$

$$S(x) = \begin{cases} 1, & if \ x < 10 \\ 0, & else \end{cases} \qquad (8)$$

where $\Delta m = P_m - P_c$ and $\Delta n = P_n - P_d$. P_m and P_c are the $PHASE$ of the m-th and c-th APs in F_a, and P_n and P_d are the $PHASE$ of the n-th and d-th APs in F_b. If Δm or Δn is smaller than 0, it should be set to $\Delta m + P_{max}$ or $\Delta n + P_{max}$, where P_{max} is the beacon interval ($102.4ms$). We define a threshold of the $PHASE$ difference of Δm and Δn as $10ms$ (an empirical constant) to determine whether the m-th and n-th APs match, as shown in Eq. (8).

Algorithm IV.1 *Fingerprint Matching* Algorithm

Input: F_a - a fingerprint from the fingerprint database; F_b - a fingerprint from the localization request; N_a - the number of APs in F_a, and N_b - the number of APs in F_b; $\{c, d\}$ - the pointers in F_a and F_b for cyclical shifting; $\{m, n\}$ - the pointers in F_a and F_b for comparing APs in each cyclical shifting; H - the matrix whose elements are the number of matched APs between F_a and F_b.
Output: The matched sequence of F_a and F_b.

1: **for all** $c \in [0, N_a)$ **do**
2:      **for all** $d \in [0, N_b)$ **do**
3:          **for all** $m \in [0, N_a)$ **do**
4:              **for all** $n \in [0, N_b)$ **do**
5:                  **if** $(\Delta m = P_m - P_c) < 0$ **then**
6:                      $\Delta m = \Delta m + P_{max}$
7:                  **end if**
8:                  **if** $(\Delta n = P_n - P_d) < 0$ **then**
9:                      $\Delta n = \Delta n + P_{max}$
10:                 **end if**
11:                 **if** $|\Delta m - \Delta n| < 10$ **then**
12:                     $h_{cd} += 1$
13:                 **end if**
14:             **end for**
15:         **end for**
16:         $H \leftarrow h_{cd}$
17:     **end for**
18: **end for**
19: $(c, d) \leftarrow argmax\{H\}$
20: **for all** $m \in [0, N_a)$ **do**
21:     **for all** $n \in [0, N_b)$ **do**
22:         **if** $(\Delta m = P_m - P_i) < 0$ **then**
23:             $\Delta m = \Delta m + P_{max}$
24:         **end if**
25:         **if** $(\Delta n = P_n - P_j) < 0$ **then**
26:             $\Delta n = \Delta n + P_{max}$
27:         **end if**
28:         **if** $\Delta m - \Delta n > 10$ **then**
29:             insert the $PHASE$ of n to F_a with $R = -129$
30:         **else if** $\Delta n - \Delta m > 10$ **then**
31:             insert the $PHASE$ of m to F_b with $R = -129$
32:         **end if**
33:     **end for**
34: **end for**
35: **return**

---

2. When the APs in F_b have finished all the N_b cyclical shiftings, c points to the second AP in F_a. In each cyclical shifting of F_a, F_b is cyclically shifted for N_b times and the number of matched APs is calculated as done in step 1. After a total of $N_a * N_b$ cyclical shiftings, we find the largest value from H and the corresponding c and d, which indicate the best match of the two fingerprints, as shown in Fig. 8 (c).

3. As some APs in F_a (or F_b) are missing the corresponding APs in F_b (or F_a), we insert some virtual APs (their RSSIs are $-129dBm$ in our experiments as interference signals [36]) to make up for the missing APs for the purpose of comparison, as shown in Fig. 8 (d).

After obtaining the matched fingerprints, we need to calculate the distance (similarity) between a pair of fingerprints. There exist a number of methods for computing the distance between a pair of fingerprints. In this work, we select the following three methods to evaluate our approach: weighted Euclidian distance, weighted Manhattan distance and relative entropy, and their calculation formulas are shown in Eqs. (11),
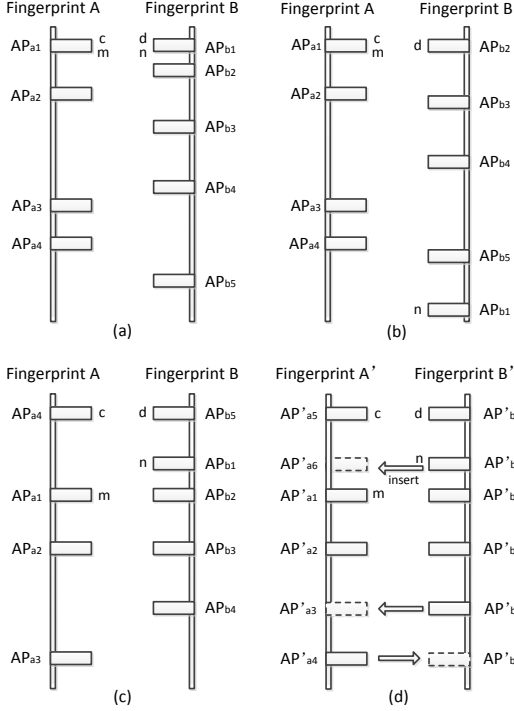
Fig. 8. An example of matching two fingerprints. (a) Two raw fingerprints before matching. (b) Alignment after one cyclic shift. (c) The best match between the two fingerprints. (d) The insertions of the missing APs.

(12) and (14), respectively.

KNN has been widely used in indoor localization [33] [26]. However, all features are treated equally in KNN, which leads to that the influence of key features may be submerged in other non-contributing features (curse of dimensionality) and hence the accuracy of KNN is low. For example, the APs with stronger $RSSI$ should be treated as being more important than those with weaker $RSSI$, as they are not equally important [37] [38]. In this work, we propose a weighted method to assign different weights based on the importance of features. To assign a weight to each AP, we define the correlation coefficient between AP u and AP v as $\alpha_{u,v}$ in Eq. (9) to construct the correlation coefficient matrix A. Let R_u denote the vector which consists of all the RSS instances of AP u in the fingerprint database; $Cov(R_u, R_v)$ is the covariance and $D(a)$ is the variance of a. Then, $\alpha_{u,v}$ can be calculated by,

$$\alpha_{u,v} = \frac{Cov(R_u, R_v)}{\sqrt{D(R_u) * D(R_v)}} \tag{9}$$

After calculating the correlation coefficient matrix A, we define the weight of AP u as w_u in Eq. (10), where $A_u = \{\alpha_{u,1}, \alpha_{u,2}, \ldots, \alpha_{u,n}\}$ is the u-th row vector of A. If the correlation between AP u and other APs is higher, which indicates that AP u brings more redundant information, the weight of AP u is lower.

After calculating the weight of each AP, we can calculate three different distances between two fingerprints F_a and F_b using Eqs. (11), (12) and (14) as the input of KNN. The time

complexity of the KNN algorithm is $O(K)$, where K is the size of training set.

$$w_u = \frac{1}{A_u \cdot A_u^T} \tag{10}$$

$$d_{Euclidian}(F_a, F_b) = \sqrt{\sum_{u=1}^{N'} w_u \times (R_{u,a} - R_{u,b})^2} \tag{11}$$

$$d_{Manhattan}(F_a, F_b) = \sum_{u=1}^{N'} w_u \times |R_{u,a} - R_{u,b}| \tag{12}$$

The relative entropy $D_{KL}(P||Q)$ is also called KL (Kullback-Leibler) divergence. It is actually a non-symmetric measurement of the information lost in approximating between two probability distributions P and Q, whose theoretical basis is the Fisher information [39]. If P equals Q, their relative entropy is 0. Thus we use it as a distance measurement. As the vectors in the relative entropy should meet the conditions that the sum of elements of a vector should be equal to 1 and each element should be greater than 0, the folded $RSSI$ series can be compressed in a simple step. For example, a folded $RSSI$ series $(0.3, 0.8, 0.2, 0.7, 0.4, 0.6)$ can be compressed to $(0.1, 0.267, 0.067, 0.233, 0.133, 0.2)$ in proportion, making the sum of elements equal to 1. Although it is often regarded as a metric or a distance, the relative entropy is not a true metric as it is not symmetric: the relative entropy from P to Q is generally not the same as the one from Q to P. We construct the fuzzy relative entropy [40] as Eq. (13) and use Eq. (14) to calculate the symmetric relative entropy.

$$E_{KL}(F_a||F_b) = \sum_{u=1}^{N'} w_u \times [R'_{u,a} \ln \frac{R'_{u,a}}{\frac{R'_{u,a}}{2} + \frac{R'_{u,b}}{2}} \\ + (1 - R'_{u,a}) \ln \frac{R'_{u,a}}{1 - \frac{R'_{u,a}}{2} - \frac{R'_{u,b}}{2}}] \tag{13}$$

$$D_{KL}(F_a||F_b) = E_{KL}(F_a||F_b) + E_{KL}(F_b||F_a) \tag{14}$$

V. EXPERIMENTATION

A. Experimental Setup

We develop our system on TelosB motes, and conduct extensive experiments at the tenth floor of New Main Building in Beihang University. The area of the floor is about 1,600 square meter, with 50 rooms in total and we use 28 of those rooms for our experimentation. The size of each room is about 3.75 by 8 m^2. There are more than 100 APs deployed in the building. Both the client and the server of ZIL are built on Lenovo ThinkPad laptops, where the client is also equipped with a TelosB mote. The laptops are running on Ubuntu with Linux kernel version 3.2.0-35.

The client scans at least 11,000 times (3s per time) and collects about 275,000,000 RSS samples in each room during the training phase. The beacon interval is $102.4ms$, and we

fold every 30 periods to form a fingerprint, which costs $3s$ to collect about 25,000 RSS samples. So a total of 308,000 fingerprint samples in the 28 rooms form the training set. We divide fingerprint samples into two parts in the following experiments: the training and testing data sets. WiFi channels 1, 6 and 11 are commonly used, and in our site survey we find that most of WiFi APs operate on channel 6. So the channel of ZigBee is set to 17, which can fully overlap with WiFi channel 6 [41] [42]. In fact, our approach does not require to detect all the WiFi APs, though the more APs are obtained, the better results can our approach have.

In our site survey, there are, on average, 20 APs that can be detected in each room. The client and the server also turn on their WiFi interfaces, in case of data transmitting between them. After RSS quantification and normalization, the fingerprints are sent to the server from clients through WiFi interfaces.

B. Accuracy and Energy Consumption with ZigBee Interface

Our testbed environment, containing 28 rooms, is complex enough as it contains a patio, corners and walking people. All these factors, adding reflection and diffraction, make the localization accuracies varied. The room-level localization accuracy is defined as the ratio of the number of correctly localized rooms to that of all localized rooms. Fig. 9 shows the layout of our experiment site, where each room is labeled with the corresponding localization accuracy based on 500 localization requests sent by the client in each room. The range of localization accuracy is from 60% to 100%. The accuracies of four out of the 28 rooms are above 90%, and those of another four rooms are between 70% and 80%. The accuracies of 78% of the 28 rooms are above 80%. Only two rooms have accuracies between 60% and 70%. On average, the accuracy of ZIL can reach 87% and the incorrectly estimated locations are usually close to the rooms where the client is actually located.
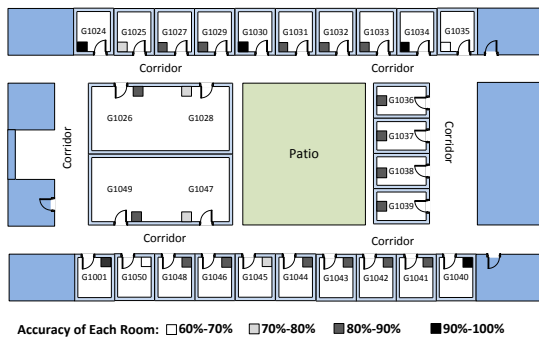

Fig. 9. The layout of our testing environment.

In this work, to accurately measure energy consumption of ZigBee and WiFi interfaces, we use a multimeter to directly measure the current draw of the TelosB mote and WiFi interface while they are working on capturing beacon frames, as shown in Fig. 10 (a). The galvanometer and voltmeter of the multimeter measure the current draw and voltage from the USB interface to get the power of ZigBee or WiFi interface

at a time. Then the energy consumption of ZigBee or WiFi interface in a period of time can be calculated by discrete integration of successive sampling power values. Fig. 10 (b) shows the localization accuracy and energy consumption of a WiFi-based fingerprint method and ZIL with different localization scanning times. In comparison, ZIL is slightly inferior to the WiFi-based method in terms of accuracy as the fingerprints of the ZigBee interface are noisier. However, the ZigBee interface reduces energy consumption by about 68% as compared with the WiFi interface, which motivates us to use ZigBee in indoor localization.
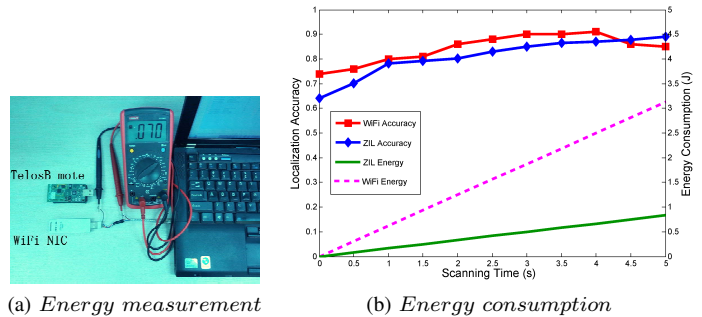
(a) *Energy measurement*   (b) *Energy consumption*

Fig. 10. Localization accuracies and energy consumption of the WiFi and Zigbee-based localization systems with different localization scanning times.

C. Performance of Different Distance Metrics

We adopt the following five metrics to evaluate the localization accuracy of our approach: Euclidian-W-KNN, Manhattan-W-KNN, KL-W-KNN, R-KNN [33] and naive Bayesian [3]. The distances of Euclidian-W-KNN, Manhattan-W-KNN and KL-W-KNN are calculated by Eqs. (11), (12) and (14), respectively.

We set k in KNN to 500 and select ten rooms for comparison. We randomly select 500 fingerprints out of the total 11,000 fingerprints as the testing set of each room, while the remaining 10,500 fingerprints are used for the training set. As shown in Fig. 11, the accuracies for rooms 1024, 1030 and 1049 are higher, above 80% in most cases. The performance of KL-W-KNN fluctuates more dramatically, reaching 97% in room 1030, but dropping below 40% in room 1044. By analyzing accuracy variances, Manhattan-W-KNN is more stable than the other two metrics, and can achieve the accuracy of about 87% on average. Manhattan-W-KNN performs close to R-KNN and 9% higher than Bayesian, on average. Moreover, the accuracy variance of R-KNN is the smallest (0.0012), compared with Manhattan-W-KNN (0.0036) and Bayesian (0.0039).

D. Impact of k in KNN Algorithms

In this subsection, we will investigate the impact of the parameter k on KNN. The range of k is set from 100 to 1000 during the experiment. We select ten rooms to conduct this experiment, and the results are shown in Fig. 12. The accuracies of rooms 1024 and 1030, as shown in Fig. 12 (a) and (c), are relatively stable and high. This is because these two
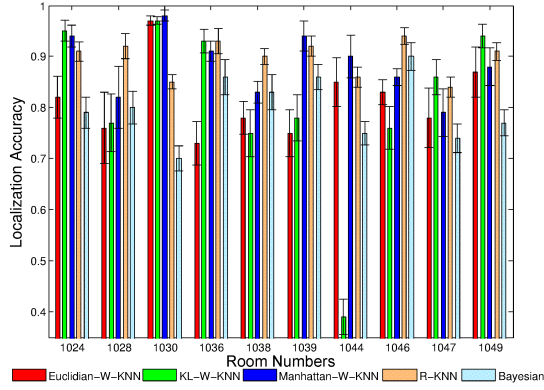
Fig. 11. Localization accuracies of 10 rooms using Euclidian-W-KNN, KL-W-KNN, Manhattan-W-KNN, R-KNN and Bayesian (95% confidence interval).

rooms are relatively independent and the distances between the rooms and other rooms are large enough to diminish the interference.

It is apparent from Fig. 12 that for all the rooms, the localization accuracy of Manhattan-W-KNN fluctuates the least and does not fall drastically, which makes it the best choice among the three metrics. The localization accuracy of Euclidian-W-KNN is lower than Manhattan-W-KNN and varies significantly with the increase of k. KL-W-KNN performs well in rooms 1024, 1047 and 1049, but performs badly in rooms 1038, 1044 and 1046. Overall, Manhattan-W-KNN is the best choice for our system.

Based on the data of the selected ten rooms, we compute the average accuracies of the five algorithms. We randomly select 500 fingerprint samples to form the testing set and the remaining samples constitute the training set. The experimental results are displayed in Fig. 13. When k is smaller than 250, Manhattan-W-KNN performs best among all the five algorithms. When k is greater than 250, R-KNN is the best method.

To further investigate the impact of k in Manhattan-W-KNN and R-KNN, we adopt FP (*False Positive*) and FN (*False Negative*) rates as the evaluation metrics. We randomly select 1,000 fingerprints out of the total 11,000 fingerprints as the testing set of each room, while the remaining fingerprints are used for the training set. We repeat for ten times of cross-validation to estimate the FP and FN rates of Manhattan-W-KNN and R-KNN for each k value. The results are shown in Figs. 14 and 15, where k ranges from 450 to 700 at a step of 50.

When k is large, both Manhattan-W-KNN and R-KNN need to consider more distant instances in the training set when calculating the location. The distant instances usually contain more noisy data than near instances. However, when k is too small, the noise in near instances may be a decisive factor to estimate the location. Therefore, it is possible for us to determine the k value with which the algorithms could perform best.

As shown in Figs. 14 and 15, the $best/worst/AVG$ curves represent the minimum/maximum/averaged FP and FN rates in all the testing rooms, respectively. In Fig. 14, the best, worst

(a) 1024

(b) 1028

(c) 1030

(d) 1036

(e) 1038

(f) 1039

(g) 1044

(h) 1046
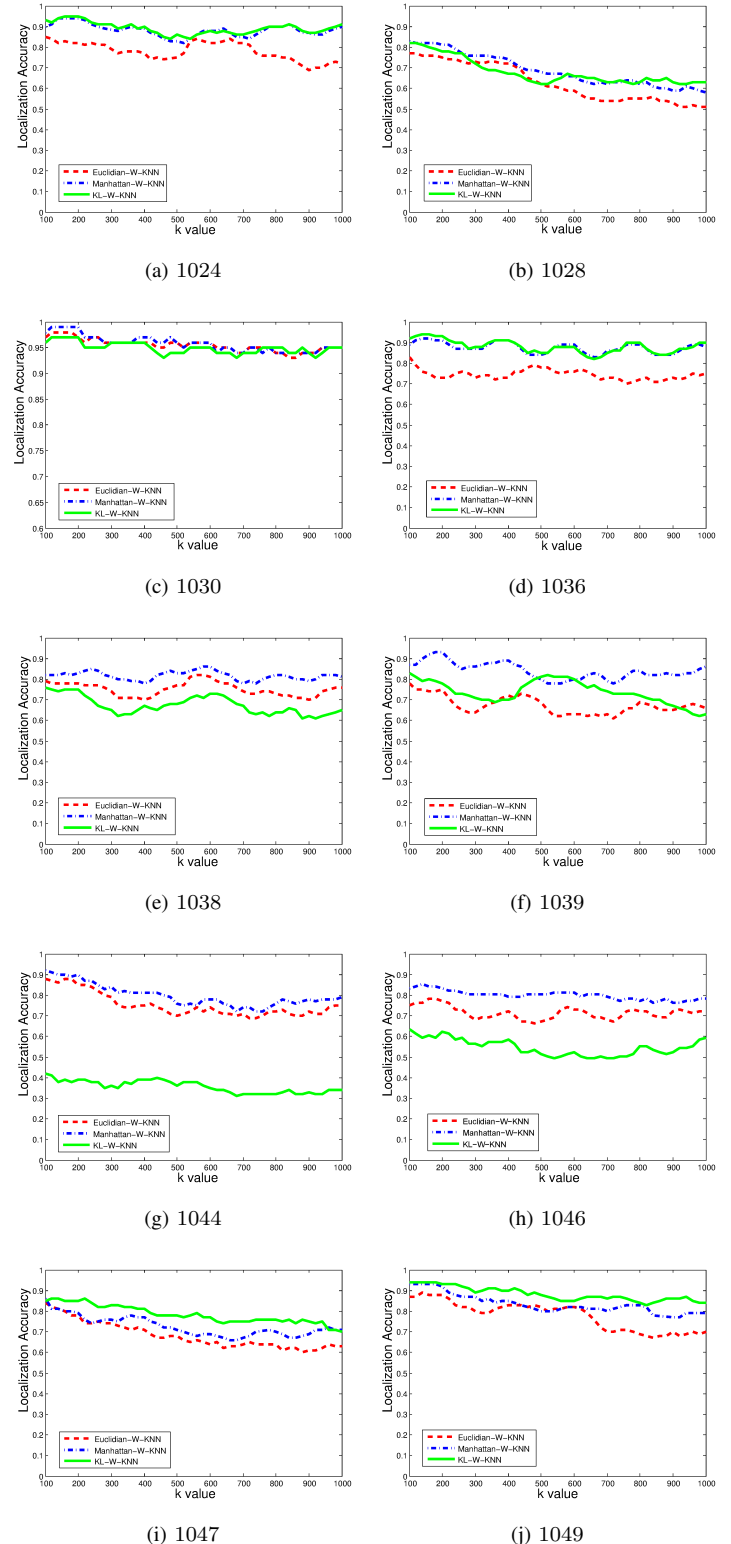

(i) 1047

(j) 1049

Fig. 12. Impact of k on three KNN variants for the ten selected rooms.

and average FP rates of Manhattan-W-KNN are below 0.67%, 4.1% and 2.1%, respectively. The best, worst and average FP rates of R-KNN are below 1.2%, 3.5% and 1.6%, respectively. We notice that the performance of the two algorithms do not fluctuate much with the increase of k from 450 to 700, and
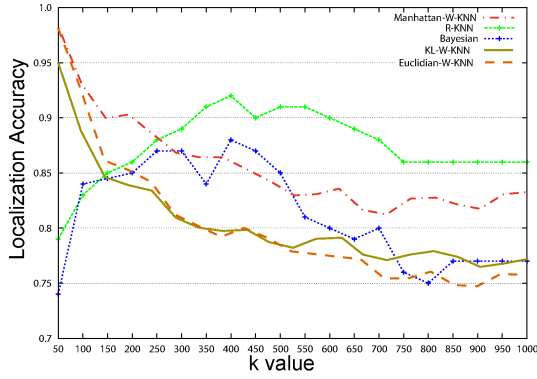
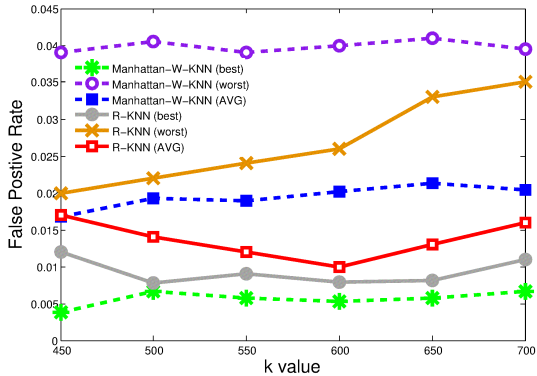Fig. 13. Average localization accuracies of the five algorithms.

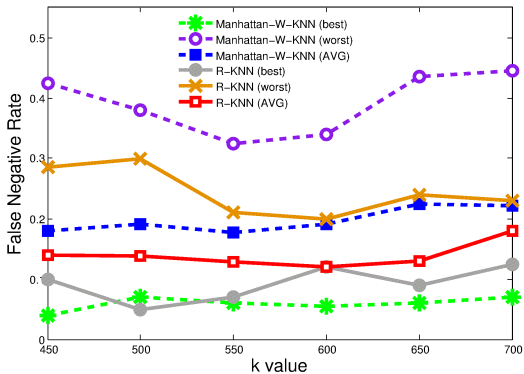Fig. 14. FP rates of Manhattan-W-KNN and R-KNN in different rooms.

Fig. 15. FN rates of Manhattan-W-KNN and R-KNN in different rooms.

both of them perform best at about $k = 550$.

A similar pattern is shown in Fig. 15. We can observe that the worst FN rate of Manhattan-W-KNN deteriorates significantly when k is larger than 600. Overall, both of them perform best when k ranges from 550 to 650.

E. Accuracy Over Time

In this subsection, we investigate the localization accuracy of ZIL in one day. As shown in Fig. 3, RSS signals do not fluctuate much between 00:00 am and 9:00 am, whereas during working hours, the signals change significantly due to the large amount of WiFi data transmissions. We evaluate the localization accuracy in rooms 1049 and 1047 for 24 hours

consecutively, and the results are shown in Fig. 16. The result shows that the accuracy increases from 00:00 am to 08:00 am, from 86% to 98%, and reaches its peak value (98%) at 08:00 am for room 1049. The accuracy starts to drop from 08:00 am to 10:00 pm continuously with the increase of WiFi traffic. The solid line is the average accuracy of these two rooms.
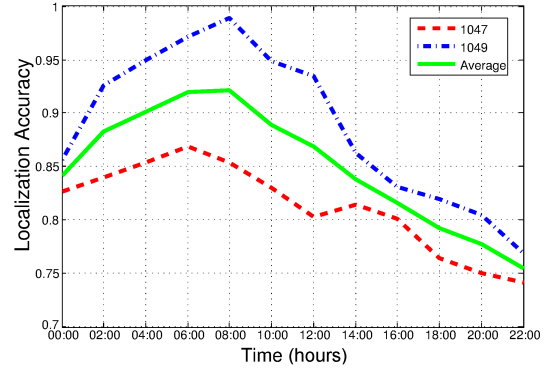

Fig. 16. Accuracies for rooms 1047 and 1049 in one day.

F. Impact of Training Set Size

Generally, KNN can have a satisfying accuracy when the training size is sufficient. In most cases, a larger training set will lead to a higher accuracy since the impact of random errors in the training set decreases. However, it is a labor intensive job to collect a large training data set. Besides, the computational cost will increase with the increase of the training data size for the KNN algorithm. Therefore, the training data size should be properly determined. We adopt Manhattan-W-KNN and R-KNN to investigate the relation between the impact of training set size and the localization accuracy in this experiment.

We test Manhattan-W-KNN and R-KNN with ten different sizes of training sets which are selected from the original training set containing about 308,000 fingerprint samples, and the experimental results are shown in Fig. 17. As shown in Fig. 17, the localization accuracies increase quickly when the size of the training set increases from 600 to 4,810, and after that both of them can achieve stable and high localization accuracies.
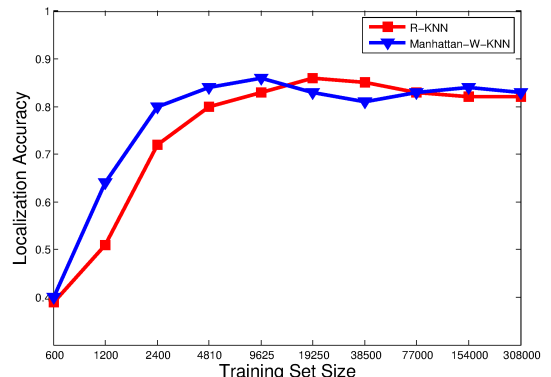

Fig. 17. Impact of training set size on Manhattan-W-KNN and R-KNN.

VI. Conclusions

In this paper, we designed ZIL, an indoor localization system using low-power ZigBee radio to detect and identify WiFi fingerprints, which delivers significant energy-saving and competitive localization accuracy according to our experiments in an office building. We designed RSS quantification and normalization schemes to recognize WiFi beacon frames and form a fingerprint database, and proposed a novel fingerprint matching algorithm to align two fingerprints. We designed the KNN algorithm with three weighted distances to evaluate the accuracy of ZIL and found that the weighted Manhattan distance has the best performance. Saving about 68% energy on average compared with the method using WiFi interfaces, our approach can provide users the localization accuracy of 87%, which outperforms existing work, such as ZiFind and the Bayesian classification method.

Acknowledgment

References

[1] Haeberlen, F. Andreas, L. Eliot, R. Andrew M., W. Algis, K. Dan S., and E. Lydia, "Practical robust localization over large-scale 802.11 wireless networks," in *Proceedings of the 10th annual international conference on Mobile computing and networking*, ser. MobiCom '04. New York, NY, USA: ACM, 2004, pp. 70–84. [Online]. Available: http://doi.acm.org/10.1145/1023720.1023728

[2] J.-g. Park, B. Charrow, D. Curtis, J. Battat, E. Minkov, J. Hicks, S. Teller, and J. Ledlie, "Growing an organic indoor location system," in *In Proceedings of the 8th international conference on Mobile systems, applications, and services*, ser. MobiSys, New York, NY, USA, Jun. 2010, pp. 271–284.

[3] T. Pering, Y. Agarwal, R. Gupta, and R. Want, "Coolspots: reducing the power consumption of wireless mobile devices with multiple radio interfaces," in *Proceedings of the 4th international conference on Mobile systems, applications and services*, ser. MobiSys, Uppsala, Sweden, Jun. 2006, pp. 220–232.

[4] J. Tao, G. Noubir, and S. Bo, "Wizi-cloud: Application-transparent dual zigbee-wifi radios for low power internet access," in *In Proceedings of 30th Annual IEEE Conference on Computer Communications*, ser. InfoCom, Shanghai, China, April 2011, pp. 1593–1601.

[5] L. Wenxian, Z. Yanmin, and H. Tian, "Wibee: Building wifi radio map with zigbee sensor networks," in *In Proceedings of 31th Annual IEEE Conference on Computer Communications*, ser. InfoCom, Orlando, FL, Mar. 2012, pp. 2926–2930.

[6] M. Youssef, A. Youssef, C. Rieger, U. Shankar, and A. Agrawala, "Pinpoint: An asynchronous time-based location determination system," in *Proceedings of the 4th international conference on Mobile systems, applications and services*, ser. MobiSys, Uppsala, Sweden, Jun. 2006, pp. 165–176.

[7] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proceedings of the sixth annual international conference on Mobile computing and networking*, ser. MobiCom, Boston, Massachusetts, Aug. 2000, pp. 32–43.

[8] D. Niculescu and B. Nath, "Ad hoc positioning system (aps) using aoa," in *In Proceedings of IEEE INFOCOM*, ser. In Proceedings of 26th Annual IEEE Conference on Computer Communications, 2003, pp. 1734–1743.

[9] P. Kemppi, T. Rautiainen, V. Ranki, F. Belloni, and J. Pajunen, "Hybrid positioning system combining angle-based localization, pedestrian dead reckoning and map filtering," in *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, ser. IPIN, Zurich, 2010, pp. 1–7.

[10] K. Whitehouse, "The design of calamari: an ad-hoc localization system for sensor networks," in *http://www.cs.virginia.edu/ whitehouse/research/localization/*, 2002.

[11] N. Dragos and N. Badri, "Dv based positioning in ad hoc networks," in *Telecommunication Systems*, ser. Volume 22, Issue 1-4, January 2003, pp. 267–280.

[12] Y. Zheng, W. Chenshu, and L. Yunhao, "Locating in fingerprint space: wireless indoor localization with little human intervention," in *Mobicom'12 Proceedings of the 18th annual international conference on Mobile computing and networking*, ser. Mobicom, New York, NY, USA, August 2012, pp. 269–280.

[13] H. Tian, H. Chengdu, B. Brian M., S. John A., and A. Tarek, "Range-free localization schemes for large scale sensor networks," in *MobiCom '03 Proceedings of the 9th annual international conference on Mobile computing and networking*, ser. Mobicom, New York, NY, USA, September 2003, pp. 81–95.

[14] L. Lazos and R. Poovendran, "Hirloc: high-resolution robust localization for wireless sensor networks," in *Selected Areas in Communications, IEEE Journal on (Volume:24, Issue:2)*, ser. JSAC, February 2006, pp. 233–246.

[15] Y. Moustafa and A. Ashok, "The horus wlan location determination system," in *MobiSys '05 Proceedings of the 3rd international conference on Mobile systems, applications, and services*, ser. MobiSys, Seatle, WA, Jun. 2005, pp. 205–218.

[16] M. Azizyan, I. Constandache, and R. Roy Choudhury, "Surroundsense: mobile phone localization via ambience fingerprinting," in *Proceedings of the 15th annual international conference on Mobile computing and networking*, ser. MobiCom '09. New York, NY, USA: ACM, 2009, pp. 261–272. [Online]. Available: http://doi.acm.org/10.1145/1614320.1614350

[17] L. Hongbo, G. Yu, Y. Jie, S. Sidhom, C. Yingying, and Y. Fan, "Push the limit of wifi based localization for smartphones," in *Proceedings of the 18th annual international conference on Mobile computing and networking*, ser. MobiCom, Istanbul, Turkey, Aug. 2012, pp. 305–316.

[18] T. Stephen P., D. Peter A., D. Robert P., and M. Gokhan, "Indoor localization without infrastructure using the acoustic background spectrum," in *MobiSys '11 Proceedings of the 9th international conference on Mobile systems, applications, and services*, ser. Mobisys, Washington, DC, USA, June 2011, pp. 155–168.

[19] J. Junghyun, G. Yu, C. Long, L. Banghui, Z. Jun, Z. Ting, and N. Jianwei, "Social-loc: Improving indoor localization with social sensing," in *SenSys '13 Proceedings of the 11th ACM Conference on Embedded Networked Sensor System*, ser. ACM SenSys, Rome, Italy, November 2013.

[20] J. Manweiler, N. Santhapuri, R. Roy Choudhury, and S. Nelakuditi, "Predicting length of stay at wifi hotspots," in *In Proceedings of 32nd Annual IEEE Conference on Computer Communications*, ser. InfoCom, Turin, Italy, April 2013.

[21] F. Chen, A. Wain Sy Anthea, V. Shahrokh, and T. Zhenhui, "Received-signal-strength-based indoor positioning using compressive sensing," in *IEEE Transactions on Mobile Computing archive Volume 11 Issue 12*, ser. TMC, December 2012, pp. 1983–1993.

[22] N. Rajalakshmi, C. Krishna Kant, and P. Venkata N., "Centaur: locating devices in an office environment," in *Mobicom '12 Proceedings of the 18th annual international conference on Mobile computing and networking*, ser. Mobicom, New York, NY, USA, August 2012, pp. 281–292.

[23] K. Das and H. Wymeersch, "Censoring for bayesian cooperative positioning in dense wireless networks," in *Selected Areas in Communications, IEEE Journal on (Volume:30, Issue:9)*, ser. JSAC, October 2012, pp. 1835–1842.

[24] A. Conti, M. Guerra, D. Dardari, and N. Decarli, "Network experimentation for cooperative localization," in *Selected Areas in Communications, IEEE Journal on (Volume:30, Issue:2)*, ser. JSAC, February 2012, pp. 467–475.

[25] T. Guang, J. Hongbo, Z. Shengkai, and K. Anne-Marie, "Connectivity-based and anchor-free localization in large-scale 2d/3d sensor networks," in *MobiHoc '10 Proceedings of the eleventh ACM international symposium on Mobile ad hoc networking and computing*, ser. MobiHoc, Chicago, Illinois, USA, September 2010, pp. 191–200.

[26] N. Jianwei, L. Banghui, C. Long, and G. Yu, "Ziloc: Energy efficient wifi fingerprint-based localization with low-power radio," in *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, ser. WCNC, Shanghai, Shanghai, China, April 2013, pp. 4558–4563.

[27] W. Jin, T. Shaojie, Y. Baocai, and L. Xiang-Yang, "Data gathering in wireless sensor networks through intelligent compressive sensing,"

in *In Proceedings of 31th Annual IEEE Conference on Computer Communications*, ser. InfoCom, Orlando, FL, March 2012, pp. 603–611.

[28] L. Bras, M. Oliveira, N. Borges de Carvalho, and P. Pinho, "Low power location protocol based on zigbee wireless sensor networks," in *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, ser. IPIN, Zurich, Switzerland, Sep. 2010, pp. 1–7.

[29] A. ZigBee, "Zigbee certified products," in *http://www.zigbee.org/*, 2013.

[30] Pantech&Curitel, "The world first mobile phone integrated with a zigbee radio produced by pantech&curitel," in *http://pantech.com/*, 2004.

[31] H. Jun, X. Guoliang, Z. Gang, and Z. Ruogu, "Beyond co-existence: Exploiting wifi white space for zigbee performance assurance," in *ICNP '10 Proceedings of the The 18th IEEE International Conference on Network Protocols*, ser. ICNP, Kyoto, Japan, Oct. 2010, pp. 305–314.

[32] Z. Ruogu, X. Yongping, X. Guoliang, S. Limin, and M. Jian, "Zifi: wireless lan discovery via zigbee interference signatures," in *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, ser. MobiCom '10. New York, NY, USA: ACM, 2010, pp. 49–60. [Online]. Available: http://doi.acm.org/10.1145/1859995.1860002

[33] G. Yuhang, N. Jianwei, Z. Ruogu, and X. Guoliang, "Zifind: Exploiting cross-technology interference signatures for energy-efficient indoor localization," in *In Proceedings of 32th Annual IEEE Conference on Computer Communications*, ser. InfoCom, Turin, Italy, April 2013, pp. 2940–2948.

[34] T. inc., "The world first android tablet supported by nfc and zigbee," in *http://taztag.com/*, 2012.

[35] G. Hiertz, S. RWTH Aachen Univ., Aachen Max, Z. Rui, D. Denteneer, and L. Berlemann, "Principles of ieee 802.11s," in *Computer Communications and Networks, 2007. ICCCN 2007. Proceedings of 16th International Conference on*, ser. ICCCN, Honolulu, HI, August 2007, pp. 1002–1007.

[36] I. Cisco Systems, "Voice over wireless lan 4.1 design guide," in *Cisco Systems, Inc.*, ser. Cisco Validated Design I, January 2010.

[37] L. B., S. J., D. A., and R. C., "Indoor positioning techniques based on wireless lan," in *First IEEE International Conference on Wireless Broadband and Ultra Wideband Communications*, ser. WBUWC, March 2007.

[38] M. W., X. W., N. W., and X. L., "Secure and robust wi-fi fingerprinting indoor localization," in *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on*, ser. IPIN, Guimaraes, Portugal, September 2011, pp. 1–7.

[39] J. C. Principe, "Information theoretic learning," in *Information Science and Statistics*, ser. ISS, 2010, pp. 16–18.

[40] Y. Shi and F. Jin, "Fuzzy object recognition based on fuzzy relative entropy," in *First International Workshop on Education Technology and Computer Science*, ser. ETCS, Wuhan, Hubei, March 2009, pp. 899–903.

[41] S. Pollin, I. Tan, B. Hodge, and C. Chun, "Harmful coexistence between 802.15.4 and 802.11: A measurement-based study," in *Cognitive Radio Oriented Wireless Networks and Communications, 2008. CrownCom 2008. 3rd International Conference on*, ser. Crowncom, Singapore, May 2008, pp. 1–6.

[42] K. Shuaib, M. Boulmalf, F. Sallabi, and A. Lakas, "Co-existence of zigbee and wlan, a performance study," in *Wireless Telecommunications Symposium, 2006. WTS '06*, ser. WTS, Singapore, April 2006, pp. 1–6.

Bowei Wang received the BE degree from School of Information Engineering, Nanchang University, China. He is now pursuing the MS degree from Beihang University, China. His research interests include mobile computing, pervasive computing and indoor localization techniques.

Lei Shu received the B.Sc. degree in computer science from South Central University for Nationalities, Wuhan, China, in 2002, the M.Sc. degree in computer engineering from Kyung Hee University, Seoul, Korea, in 2005, and the Ph.D. degree from the Digital Enterprise Research Institute, National University of Ireland, Galway, Ireland, in 2010. He is a Professor with the Guangdong University of Petrochemical Technology, Maoming, China. He was a Specially Assigned Research Fellow with the Department of Multimedia Engineering, Graduate School of Information Science and Technology, Osaka University, Osaka, Japan, and was a Research Scientist with the Digital Enterprise Research Institute (DERI), National University of Ireland, Galway, Ireland. He has authored and coauthored over 100 papers in related conferences, journals, and books.

Prof. Shu is a member of the Association for Computing Machinery.

Trung Q. Duong was born in HoiAn, Quang Nam Province, Vietnam, in 1979. He received the B.S. degree in electrical engineering from Ho Chi Minh City University of Technology, Vietnam, in 2002, and the M.Sc. degree in computer engineering from Kyung Hee University, South Korea, in 2005. In April 2004, he joined the electrical engineering faculty of the Ho Chi Minh City University of Transport, Vietnam. He was a recipient of the Korean Government IT Scholarship Program for International Graduate Students from 2003 to 2007. In December 2007, he joined the Radio Communication Group, Blekinge Institute of Technology, Sweden, as a research staff member working toward his Ph.D. degree. He was a Visiting Scholar at Polytechnic Institute of New York University, NY, from December 2009 to January 2010 and then at Singapore University of Technology and Design from July 2012 to August 2012. He received his Ph.D. degree in telecommunication systems from Blekinge Institute of Technology, Sweden, in 2012. His current research interests include cross-layer design, cooperative communications, and cognitive radio networks.

Dr. Duong is a frequent reviewer for numerous journals/conferences and a TPC member for many conferences including ICC, WCNC, VTC, PIMRC. He is a TPC co-chair of the International Conference on Computing, Managements, and Telecommunications 2013 (ComManTel). He was awarded the Best Paper Award of IEEE Student Paper Contest-IEEE Seoul Section in December 2006 and was a finalist for the best paper award at the IEEE Radio andWireless Symposium, San Diego, CA, in 2009.

Jianwei Niu received his Ph.D. degrees in 2002 in computer science from Beijing University of Aeronautics and Astronautics (BUAA, now Beihang University). He was a visiting scholar at School of Computer Science, Carnegie Mellon University, USA from Jan. 2010 to Feb. 2011. He is a professor in the School of Computer Science and Engineering, BUAA. He is now a IEEE senior member. He has published more than 100 referred papers and filed more than 30 patents in mobile and pervasive computing. He served as the DySON workshop co-chair of Infocom 2014, the Program Chair of IEEE SEC 2008, Executive Co-chair of TPC of CPSCom 2013, TPC members of InfoCom, Percom, ICC, WCNC, Globecom, LCN, and etc. He has served as associate editor of Int. J. of Ad Hoc and Ubiquitous Computing, associate editor of Journal of Internet Technology, editor of Journal of Network and Computer Applications. He won the best paper award in ICC 2013, WCNC 2013, ICACT 2013, CWSN 2012 and the 2010 IEEE International Conference on Green Computing and Communications (GreenCom 2010). His current research interests include mobile and pervasive computing.

Yuanfang Chen received her M.E. degree at Department of Software Engineering in 2007 and her PhD degree at Department of Computer Science in 2013, both from Dalian University of Technology, China. Currently, she is a PhD student of Université Pierre et Marie CURIE (University of Paris VI), France for her second PhD degree. She was an assistant researcher of Illinois Institute of Technology (advisor: Xiang-Yang Li), U.S.A., from Sept. 2009 to Sept. 2010. She is the assistant editor of Industrial Networks and Intelligent Systems. She has served as a session chair of MobiQuitous 2013, a volunteer of Mobicom & Mobihoc 2010 and a publicity co-chair of International Symposium on Mobile and Wireless Network Security 2011. She has served as a TPC member of several conferences such as Globecom 2014, ChinaCom 2013, 2014 and MobiApps 2014. She has served as a reviewer of several journals and conferences such as Ad Hoc & Sensor Wireless Networks and TPDS. Her research interests are focused on Internet of Things, Collaborative Analytics, Computational Intelligence, Sensing Intelligence, Knowledge Discovery and Wireless Sensor Networks. She is also involved in a number of European projects (e.g., Collaborative Analytics Platform: https://itea3.org/project/cap.html) and contributes to the ITU-T.