WicLoc: An Indoor Localization System based on WiFi Fingerprints and Crowdsourcing

Jianwei Niu[1], Bowei Wang[1], Long Cheng[1] and Joel J. P. C. Rodrigues[2,3]
[1]State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, China
[2]Instituto de Telecomunicações, University of Beira Interior, Portugal
[3]University ITMO, St. Petersburg, Russia
Email: niujianwei@buaa.edu.cn, wangbowei1219@cse.buaa.edu.cn, chenglong@buaa.edu.cn and joeljr@ieee.org

***Abstract*—WiFi fingerprint-based indoor localization techniques have been proposed and widely used in recent years. Most solutions need a site survey to collect fingerprints from interested locations to construct the fingerprint database. However, the site survey is labor-intensive and time-consuming. To overcome this shortcoming, we record user motions as well as WiFi signals without the active participation of the users to construct the fingerprint database, in place of the previous site survey. In this paper, we develop an indoor localization system called WicLoc, which is based on WiFi fingerprinting and crowdsourcing. We design a fingerprint model to form fingerprints of each location of interest after fingerprint collection. We propose a weighted KNN (K-Nearest Neighbor) algorithm to assign different weights to APs and achieve room-level localization. To obtain the absolute coordinate of users, we design a novel MDS (Multi-Dimensional Scaling) algorithm called MDS-C (Multi-Dimensional Scaling with Calibrations) to calculate coordinates of interested locations in the corridor and rooms, where anchor points are used to calibrate absolute coordinates of users. Experimental results show that our system can achieve a competitive localization accuracy compared with state-of-the-art WiFi fingerprint-based methods while avoiding the labor-intensive site survey.**

***Index Terms*—Indoor localization; WiFi Fingerprints; Crowdsourcing; K-Nearest Neighbor; Multi-Dimensional Scaling**

I. Introduction

Recent years have witnessed wireless indoor localization techniques undergoing a rapid development, most of which depend on extra facilities. With the increase of WiFi APs (Access Points) deployed in the public, many existing methods [1] [2] have taken advantage of the off-the-shelf WiFi APs for indoor localization. RSSIs (Received Signal Strength Indications) collected from APs, which follow the PLM (Path Loss Model) [3], can be regarded as the features of an interested location. WiFi fingerprint-based indoor localization has been widely used as it requires no extra infrastructures, which contains two phases: training phase and testing phase.

The site survey during the training phase is the main drawback of WiFi fingerprint-based indoor localization, as it is labor-intensive and time-consuming. In addition, as the reflection and diffraction constantly happen in indoor environment, RSSIs are susceptible to the environment, thus the one time collected fingerprint database is inapplicable and should be reconstructed over time. Consequently, some methods [4] [5] [6] have proposed to dynamically collect crowdsourced WiFi signals as well as user motions with mobile devices to avoid this kind of site survey.

In this paper, we propose a novel WiFi fingerprint-based and crowdsourcing indoor localization system called WicLoc (*Wi*Fi fingerprints and *c*rowdsourcing indoor *Loc*alization). Compared with existing indoor localization systems, our main contributions are as follows:

- We collect WiFi signals with user motions using crowdsourcing and design a fingerprint model to get WiFi fingerprints of each interested location. Considering the turns of users and the inevitably overlaps between paths, it is a key issue to identify the fingerprints of the same locations exactly.
- We propose a weighted KNN (K-Nearest Neighbor) algorithm to assign different weights to APs and achieve room-level localization. To obtain the absolute coordinate of users, we propose a novel MDS (Multi-Dimensional Scaling) algorithm called MDS-C (Multi-Dimensional Scaling with Calibrations), which calculates the coordinates of the corridor and rooms, and use anchor points to match with the corresponding points in the map for calibration.
- We implement a novel WiFi fingerprint-based and crowdsourcing indoor localization system called WicLoc. Experiments show that our system can achieve a competitive localization accuracy compared with state-of-the-art WiFi fingerprint-based methods, such as LiFS [7] and EZ [8].

The rest of the paper is organized as follows. Section II provides the related work. Sections III describes the design of our system in detail. Section IV shows the experimental results followed by the conclusion in section V.

II. Related Work

Many approaches have been proposed for indoor localization in recent years. One class is the range-based indoor localization, including the TOA (Time-of-Arrival) [9], T-DOA (Time-Difference-of-Arrival) [10], AOA (Angle-of-Arrival) [11], DOA (Direction-of-Arrival) [12] and RSS (Received Signal Strength). Another class is the range-free indoor localization, such as DV-Hop (Distance Vector-Hop) [13], MDS [14], APIT (Approximate Point-In Triangulation) [15], etc.

978-1-4673-6432-4/15/$31.00 ©2015 IEEE

Fingerprint-based localization method has become widely used in recent years. The main idea is to collect fingerprints of each interested location in the surrounding environment and build a fingerprint database. Then the fingerprints in the localization request are matched with the database to obtain the location estimation. Horus system [16] identifies different causes for the WiFi channel variations and uses the probabilistic location determination technique to overcome them. Surroundsense [17] utilizes ambience sound, light, color and WiFi signals to form identifiable fingerprints for logical localization. However, all these fingerprint collections need a site survey, which is labor-intensive and time-consuming. Methods without site survey have been proposed to automatically collect fingerprints using crowdsourcing. A research system called WILL [18] fully combines user movements with WiFi fingerprints to achieve a room-level accuracy of 80%. LiFS [7] is a research system based on WILL, which mixes the methods of the inertial navigation and WiFi fingerprinting together. Furthermore, Unloc [4] and Zee [5] all take advantage of user motions to avoid any explicit effort on users. Unloc uses dead-reckoning techniques to track mobile devices and recalibrates the locations using landmarks, which have unique physical characteristics in the environment, with the averaged localization error of $1.69m$. Zee leverages the inertial sensors to track users in an indoor environment, estimates the stride lengths of different users and employs WiFi-based particle initialization to enable faster convergence to improve the accuracy. In our method, we use crowdsourcing to avoid any explicit user participation and the site survey, use a modified MDS algorithm to calibrate the absolute coordinates of users and achieve a competitive localization accuracy compared with previous methods.

III. System Design

In this section, we will introduce our designed system in detail. Firstly, we briefly describe the system architecture shown in Fig. 1. Two phases including training and testing are adopted. In the training phase, we collect WiFi signals with user motions using mobile devices taken by participants. After collection, a fingerprint model is adopted to extract WiFi signals from user paths and form WiFi fingerprints of each interested location. We propose a weighted KNN algorithm to assign weights to different APs and obtain the room-level localization. Then a modified MDS algorithm is used to transform the distances of fingerprints into the high dimension space. Considering the deviation brought by the classical MDS algorithm, in our proposed algorithm, we use anchor-point calibration to further diminish the deviation. WiFi fingerprints are then stored in the fingerprint database according to their corresponding locations. In the testing phase, absolute coordinates of requesting mobile devices are calculated by our MDS-C algorithm.

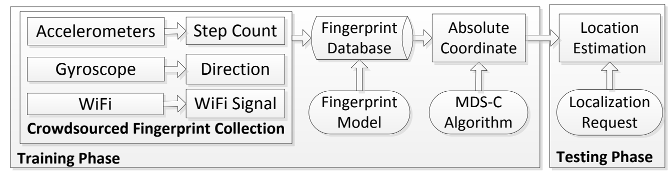


Fig. 1. The system architecture of WicLoc.

A. Crowdsourced Fingerprint Collection

To reduce the burden of participants (e.g., students from our university), applications installed in their mobile phones automatically record the accelerometer and gyroscope readings with WiFi signals during their daily time. For most of the students, their daily trajectories within the campus are almost fixed. When accelerometer detects two continuous steps or gyroscope detects a turn, WiFi signals are captured and stored in the fingerprint database. After a certain period of crowdsourced collection, we can obtain the path of each student, with measuring points on the path expressed as $< f_{ij}, ASC, GY >$, where $f_{ij} = \{M_{ij}, R_{ij}\}$ $(i \in [1, M], j \in [1, N])$, ASC is the accumulated step count and GY is a boolean variable to indicate a turn when $GY = 1$. We assume that there are a total of M APs that can be detected in the site and N measuring points on this path. M_{ij} and R_{ij} are the MAC (Media Access Control) address and RSSI of the j-th measuring point of the i-th AP on this path.

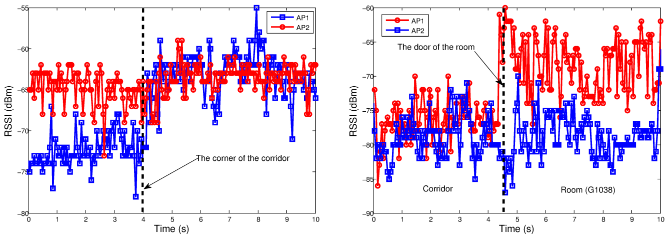
Some methods have been proposed to calculate the walking distances and orientations using accelerometer and gyroscope. However, the average error grows to over 150 $meters$ after 60 $seconds$ of operation [19]. We use ERSP (Energy-efficient Real-time Smartphone Pedometer) [20], which accurately and energy-efficiently infers the real-time human step count within 2 seconds using the accelerometer, without the need for noise filtering or specific placement and orientation of smartphone on the body. Moreover, we use the gyroscope to detect turns from user motions which can be clearly identified by the sharp change. Compared with the orientation sensor, we can determine the phone's orientation at any time, even in magnetically-interfered areas [21].

B. Fingerprint Model

After the crowdsourced fingerprint collection, we get many user paths with the corresponding features including WiFi signals, step counts and turns. These paths inevitably overlap, which motivates us to build a fingerprint model to identify fingerprints of different locations and merge the similar fingerprints.

We divide our experimental area into a 2*2 meter grid. We also build a coordinate system over the area to obtain the absolute coordinate of each interested location. User motions include walking and turning, and we will consider both to distinguish the data of rooms from corridor. There are three possible scenarios where a turn may happen: arriving at a corner, going into a room and turning around in a room or corridor. Fig. 2 shows RSSI variations of two APs when a user arrives at a corner of the corridor or near a door.

As shown in Fig. 2, RSSIs change continuously as the user keeps walking. At the moment of arriving at a corner



(a) RSSI variations at a corner (b) RSSI variations near a door

Fig. 2. RSSI variations of two APs at a corner and a door.

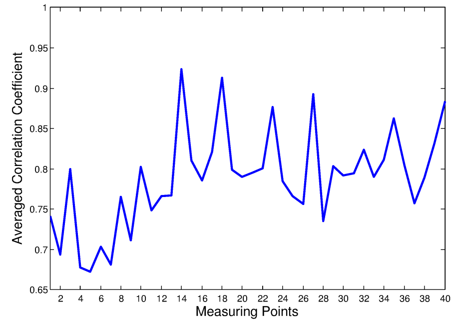


Fig. 3. Averaged correlation coefficients of 40 measuring points.

of the corridor, as shown in Fig. 2 (a), RSSI of $AP1$ rapidly increases. The similar situation happens when a user is going into a room through the door. In Fig. 2 (b), RSSIs of $AP1$ and $AP2$ all appear a sudden increase near the door. To find the differences of these actions, we define two thresholds through our extensive experiments on all the corners and the doors of rooms. If we detect a turn happens at the j-th measuring point and R_j expresses the vector of RSSIs, an action can be determined as follows: $a)$ $|R_{j-1} - R_{j+1}| \geq 15dBm$ indicates that the user is near the door and going into or out of the room; $b)$ $10dBm \leq |R_{j-1} - R_{j+1}| < 15dBm$ shows the user passing a corner of the corridor; $c)$ $|R_{j-1} - R_{j+1}| < 10dBm$ shows the user turning around in a room or corridor. We choose the turning points near the doors and corridor as the anchor points for the following MDS-C algorithm.

To determine whether each two sets of RSSIs belong to the same interested location, we calculate the similarity between them in this densely distributed AP situation. Generally, we define two series of RSSIs as R_i and R_j, and define the correlation coefficient $r_{ij} = Cov(R_i, R_j)/\sqrt{D(R_i) * D(R_j)}$ as the metric. If r_{ij} is larger than a pre-defined threshold, we assume that R_i and R_j are captured at the same location. To determine the pre-defined threshold, we conduct a experiment on 40 measuring points and calculate the averaged correlation coefficient of each point. The result shown in Fig. 3 indicates that all the averaged correlation coefficients are above 0.65, which let us determine the threshold as 0.65. Thus, if $r_{ij} \geq 0.65$, R_i and R_j are merged as the same interested location. Otherwise, if $r_{ij} < 0.65$, R_i and R_j belong to different locations. Then the fingerprints of each interested location (the fingerprint model) is expressed as $F_l = \{f_{1l}, f_{2l}, \cdots, f_{Ml}\}$ $(l \in [1, L])$, where L is the number of interested locations.

C. Manhattan-W-KNN Algorithm

To map the points of each cluster to the corresponding room and achieve room-level indoor localization, we use KNN (K-Nearest Neighbor) algorithm to classify the location based on the fingerprints. However, all features are treated equally in KNN, which leads to the submersion of the key features in other non-contributing features. In this work, we use a weighted method to assign different weights to different features based on the importance of features (named Manhattan-W-KNN). To assign a weight to each AP, we define the correlation coefficient between AP a and AP b as $c_{ab} = Cov(I_a, I_b)/\sqrt{D(I_a) * D(I_b)}$ to construct the correlation coefficient matrix C. Let I_a denote the vector which consists of all the RSS instances of AP a in the fingerprint database. After calculating the correlation coefficient matrix C, we define the weight of AP a as w_a in Eq. (1), where $C_a = \{c_{a1}, c_{a2}, \ldots, c_{aM}\}$ is the a-th row vector of C. If the correlation between AP a and other APs is higher, which indicates that AP a brings more redundant information, the weight of AP a is lower.

$$w_a = \frac{1}{C_a \cdot C_a^T} \tag{1}$$

After calculating the weight of each AP, we calculate the Manhattan distance between two fingerprints F_g and F_h using Eq. (2) as the input of Manhattan-W-KNN.

$$d_{Manhattan}(F_g, F_h) = \sum_{a=1}^{M} w_a \times |R_{ag} - R_{ah}| \tag{2}$$

D. MDS-C Algorithm

In this section we will present our modified MDS algorithm in detail. To obtain the absolute coordinate of users, we create a distance matrix of fingerprints and use the MDS algorithm to transform the distances into the high dimension space. As the distance estimations in the classical MDS algorithm produce errors, the estimated locations usually differ from correct locations. Thus we propose the MDS-C algorithm to calibrate the estimated locations with anchor points. The steps of MDS-C are as follows:

Step 1. Dijkstra's shortest path algorithm [22] is adopted to calculate pairwise shortest distances of fingerprints with the time complexity of $O(L^3)$. The results form the squared distance matrix D and each element d_{gh}^2 in D indicates the squared shortest distances of fingerprints g and h.

Step 2. To obtain the relative coordinate matrix X of fingerprints, we define the inner product matrix $B = XX^T$ to express as the doubly center form of D. B can be again expressed as,

$$B = -\frac{1}{2}JDJ^T, \tag{3}$$

where $J = E - \frac{1}{L}II^T$. E is a L-Dimensional unit matrix and I is a L-Dimensional vector with each element equals 1. Obviously B is a symmetric positive semidefinite matrix and performs SVD (Singular Value Decomposition) to be transformed as,

$$B = VAV^T, \tag{4}$$

where V is the eigenvector matrix and A is the eigenvalue matrix. We retain the first 2 of L eigenvectors to get the 2-D relative coordinate matrix X as,

$$X = VA^{1/2} \tag{5}$$

Given sufficient anchor nodes (three or more in 2-D space), the coordinates of the anchors are mapped to their absolute coordinates through a linear transformation [23], and thus the relative coordinate matrix X is converted to the absolute coordinate matrix X'.

Step 3. Let $F_R = \{F_1, F_2, \cdots, F_K\}$ denote as the anchor points set, where K is the number of locations that turns at the corners and the doors are detected. Furthermore, let $G = \{G_1, G_2, \cdots, G_K\}$ denote as the set that contains the real coordinates of the corners and the doors in the geographical space. The problem to be solved is to match F_R with G. Only two situations exist: F_R matches with the positive sequence of G and F_R matches with the negative sequence of G. As F_R and G cannot be directly compared, we construct three sets $H^{(e)} = \{H_1^{(e)}, H_2^{(e)}, \cdots, H_{K-1}^{(e)}\}$ $(e \in [1, 3])$, where $H_k^{(1)} = ||F_{k+1} - F_k||_2$, $H_k^{(2)} = ||G_{k+1} - G_k||_2$ and $H_k^{(3)} = ||G_k - G_{k+1}||_2$. $||\cdot||_2$ indicates the 2-Norm to calculate the Euclidian distance and $k \in [1, K]$. Then the correlation coefficients can be calculated by,

$$r_1 = \frac{Cov(H^{(1)}, H^{(2)})}{\sqrt{D(H^{(1)}) * D(H^{(2)})}}, \tag{6}$$

$$r_2 = \frac{Cov(H^{(1)}, H^{(3)})}{\sqrt{D(H^{(1)}) * D(H^{(3)})}} \tag{7}$$

If $r_1 \geq r_2$, F_R matches with the positive sequence of G; otherwise, F_R matches with the negative sequence of G. Up to now, we obtain the matching sequence between F_R and G.

Step 4. To calibrate the absolute coordinates, we use the LS (Least Square) approach to match the coordinates of F_R with G. Assume that, the coordinate of each $F_k \subseteq F_R$ is $x_k = [x_k^{(1)}\ x_k^{(2)}]^T$ $(k \in [1, K])$ and the coordinate of each $G_k \subseteq G$ is $y_k = [y_k^{(1)}\ y_k^{(2)}]^T$ $(k \in [1, K])$. To achieve the goal of LS which is to minimize $\sum_{k=1}^{K} ||x_k - y_k||_2$, we define the calibrated set as $\hat{X} = A^{\dagger}B$ where $\hat{X} = \{\hat{X}_1, \hat{X}_2, \cdots, \hat{X}_K\}$ and $A^{\dagger} = (A^TA)^{-1}A^T$. The Moore-Penrose pseudoinverse matrixes of A and B are calculated by,

$$A = \begin{bmatrix} 2(x_1^{(1)} - y_K^{(1)}) & 2(x_1^{(2)} - y_K^{(2)}) \\ \vdots & \vdots \\ 2(x_{K-1}^{(1)} - y_K^{(1)}) & 2(x_{K-1}^{(2)} - y_K^{(2)}) \end{bmatrix}, \tag{8}$$

$$B = \begin{bmatrix} (x_1^{(1)} - y_K^{(1)})^2 + (x_1^{(2)} - y_K^{(2)})^2 + d_{K,K}^2 - d_{1,1}^2 \\ \vdots \\ (x_{K-1}^{(1)} - y_K^{(1)})^2 + (x_{K-1}^{(2)} - y_K^{(2)})^2 + d_{K,K}^2 - d_{K-1,K-1}^2 \end{bmatrix} \tag{9}$$

where $d_{k,k} = ||y_k - x_k||_2$ $(k \in [1, K])$ is the 2-Norm to calculate the Euclidian distance between coordinates x_k and y_k. $\hat{X}$ is the calibrated absolute coordinate set of anchor points.

IV. Experimental Results

A. Experimental Setup

In this section, we present the experimental results. We 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 the southern part with 28 rooms and the circular corridor as our experimental area, shown in Fig. 4. The size of each room is about 3.75 by $8\ m^2$. There are more than 100 APs detected at the tenth floor of the building.

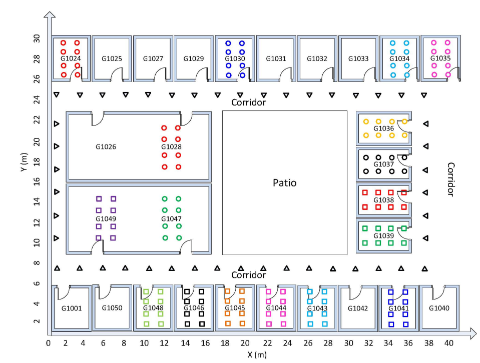


Fig. 4. The layout of our testing environment with the absolute coordinates of interested locations.

To collect fingerprints in a crowdsourced manner, we randomly select 17 students from 17 rooms. The background applications installed in their mobile phones automatically which are carried in the daily time record the accelerometer and gyroscope readings with WiFi signals along the path. We employ the WiFi interface to collect WiFi signals when detecting two continuous steps or a turn along the path. During one week (15 hours a day from $9:00$ am to $24:00$ pm), we totally collect $2,400$ paths which contain about $480,000$ fingerprints stored in the fingerprint database with the interested location information. The turns are labeled to identify the fingerprints of each measuring point.

B. Localization Accuracy

To determine the localization performance, we estimate the localization error which is the Euclidian distance between the estimated location and the corresponding real location. We use half of the $480,000$ fingerprints as the testing set to request localization and the remaining fingerprints as the training set. We adopt Manhattan-W-KNN and MDS-C algorithms in this experiment, and select 12 anchor nodes for coordinate calibration. We also implement LiFS [7] and EZ [8] to compare their performance with our system over the same experimental data.

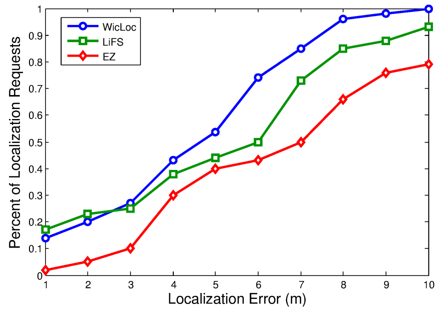


Fig. 5. CDF of localization errors in WicLoc, LiFS and EZ.

Fig. 5 shows that, the averaged localization error of WicLoc is $4.65m$, which is smaller than that of LiFS ($5.88m$) and EZ ($7m$). EZ estimates the relative location between mobile clients and APs without priori knowledge, and the localization errors of 50% fingerprints are under $7m$. LiFS matches the fingerprint space with the stress-free floor space together and the localization errors of 50% fingerprints are under $6m$. In WicLoc, we pay attention to the WiFi fingerprint processing and the coordinate calibration, and the localization errors of 70% fingerprints are under $6m$ while those of 50% fingerprints are under $5m$. Overall, WicLoc can achieve a competitive localization accuracy compared with the state-of-the-art WiFi fingerprint-based methods.

C. Performance Comparison of Classification Algorithms

To determine whether a fingerprint is generated from a certain room, we evaluate three classification algorithms: Manhattan-W-KNN, KNN and Bayesian. We set k in KNN to 500 and randomly select ten rooms for comparison. We randomly select $1,000$ fingerprints out of the total $85,000$ fingerprints as the testing set of each room, while the remaining $84,000$ fingerprints are used for the training set. The localization accuracy in Fig. 6 is defined as the ratio of the number of successful localizations to the number of all localization requests. As shown in Fig. 6, the accuracies of Manhattan-W-KNN and KNN for rooms 1024, 1030, 1044, 1046 and 1049 are higher, above 80% in most cases. The variances of localization accuracies of Manhattan-W-KNN, KNN and Bayesian are 0.0036, 0.005 and 0.0039, respectively. Thus Manhattan-W-KNN is more stable than the other two metrics and can achieve the accuracy of about 87% on average, which is 7% higher than that of KNN and 9% higher than that of Bayesian.

D. Performance Comparison of MDS Algorithms

To estimate the performance of our proposed MDS-C algorithm, we conduct an experiment to compare it with the classical MDS algorithm. We use $132,000$ fingerprints from half of the fingerprints as the testing set to request localization

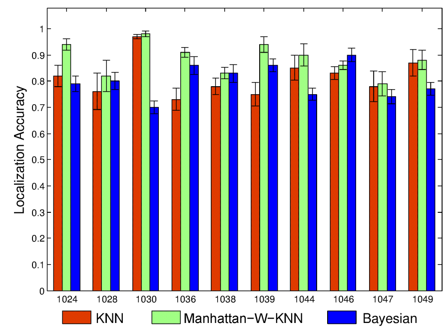


Fig. 6. Localization accuracies of 10 rooms using Manhattan-W-KNN, KNN and Bayesian (95% confidence interval).

and the remaining $240,000$ fingerprints as the training set. We run experiments with different numbers of anchor nodes ranging from 3 to 19, as shown in Fig. 7.

As the result shown in Fig. 7, the localization errors exhibit a continuous downtrend in both MDS and MDS-C as the increase of anchor nodes. We can observe that the trend of localization errors become relatively flat when using 15 and 11 anchor nodes in MDS and MDS-C, respectively. Overall, MDS-C can achieve a better localization performance and smaller averaged localization error ($4.65m$) than that of MDS (about $5.88m$).

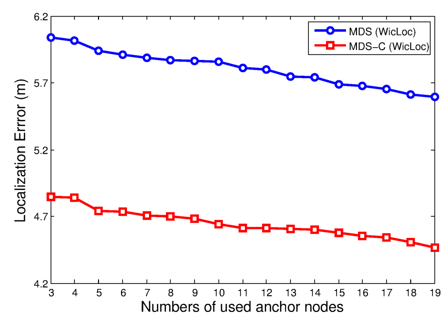


Fig. 7. Comparison of MDS and MDS-C algorithms.

E. Impact of Training Set Size

Generally a larger training set will lead to a higher accuracy since the impact of random errors in the training set decreases. The result of this experiment will shed light on how to properly determine how many fingerprints should we collect to achieve a stable and high localization accuracy.

We randomly select 20%, 40%, 60%, 80% and 100% of the $480,000$ fingerprints as the training dataset to run this experiment. Fig. 8 shows the localization errors of different sizes of training sets on WicLoc, LiFS and EZ. In WicLoc, we use Manhattan-W-KNN and MDS-C algorithms, and select 12 anchors for coordinate calibration. We can observe that, before the training set size reaches 80% percent of all the fingerprints, the localization errors decrease quickly for the three methods. After that, the localization errors decrease slower than before, indicating relatively stable accuracies of about $4.6m$ (WicLoc), $5.8m$ (LiFS) and $7m$ (EZ).

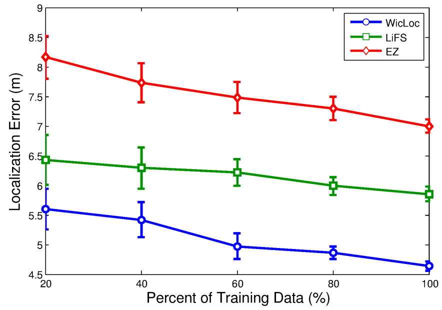


Fig. 8. Impact of training set size on WicLoc, LiFS and EZ (95% confidence interval).

V. Conclusion

In this paper, we implement a WiFi fingerprinting and crowdsourcing-based indoor localization system called WicLoc. We collect accelerometer and gyroscope readings and WiFi fingerprints in a crowdsourced manner, and design a fingerprint model to form fingerprints of each interested location. We design a weighted KNN algorithm to achieve the room-level localization. To obtain the absolute coordinates of users, we design a novel MDS algorithm called MDS-C, which uses anchor points to calibrate absolute coordinates. Experimental results show that our system can achieve a competitive localization accuracy compared with state-of-the-art WiFi fingerprint-based methods, such as LiFS and EZ, while avoiding the labor-intensive site survey. Our ongoing work focuses on making WicLoc appropriate for various environment.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (61170296 and 61190125), 973 Program (2013CB035503) and the R&D Program (2013BAH35F01).

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, 2004, pp. 70–84.

[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] R. T. S., "Wireless communications: Principles and practice, 2nd." in *ed. Prentice Hall*, ser. WC2002, 2002.

[4] W. He, S. Souvik, E. Ahmed, F. Moustafa, Y. Moustafa, and C. Romit Roy, "No need to war-drive: unsupervised indoor localization," in *MobiSys '12 Proceedings of the 10th international conference on Mobile systems, applications, and services*, ser. Mobisys, Jun. 2012, pp. 197–210.

[5] R. Anshul, C. Krishna Kant, P. Venkata N., and S. Rijurekha, "Zee: zero-effort crowdsourcing for indoor localization," in *Mobicom '12 Proceedings of the 18th annual international conference on Mobile computing and networking*, ser. Mobicom, Aug. 2012, pp. 293–304.

[6] K. Yungeun, C. Yohan, and C. Hojung, "Smartphone-based collaborative and autonomous radio fingerprinting," in *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, ser. IEEE TRANS. SYST., MAN, CYBERN., SYST., Dec. 2010, pp. 112–122.

[7] 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, Istanbul, Turkey, Aug. 2012, pp. 269–280.

[8] C. Krishna, P. I. Anand, and P. Venkata N., "Indoor localization without the pain," in *MobiCom '10 Proceedings of the sixteenth annual international conference on Mobile computing and networking*, ser. Mobicom, Chicago, Illinois, USA, Sep. 2010, pp. 173–184.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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, Istanbul, Turkey, August 2012, pp. 269–280.

[15] 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.

[16] 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.

[17] 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, 2009, pp. 261–272.

[18] W. Chenshu, Y. Zheng, L. Yunhao, and X. Wei, "Will: Wireless indoor localization without site survey," in *Parallel and Distributed Systems, IEEE Transactions on (Volume:24, Issue: 4)*, ser. TPDS, Apr. 2013, pp. 1045–9219.

[19] O. Woodman, "An introduction to inertial navigation," in *Technical Report 696, University of Cambridge*, ser. TR, 2007, p. 3.

[20] T. Oshin and S. Poslad, "Ersp: An energy-efficient real-time smartphone pedometer," in *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, ser. SMC, Manchester, UK, Oct. 2013, pp. 2067–2072.

[21] C. Barthold, K. Subbu, and R. Dantu, "Evaluation of gyroscope-embedded mobile phones," in *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on*, ser. SMC, Anchorage, AK, Oct. 2011, pp. 1632–1638.

[22] N. Jasika, N. Alispahic, A. Elma, and K. Ilvana, "Dijkstra's shortest path algorithm serial and parallel execution performance analysis," in *MIPRO, 2012 Proceedings of the 35th International Convention*, ser. MIPRO, Opatija, May 2012, pp. 1811–1815.

[23] S. Yi and W. Ruml, "Improved mds-based localization," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies (Volume:4)*, ser. Infocom, Hong Kong, China, Mar. 2004, pp. 2640–2651.