Noname manuscript No.
(will be inserted by the editor)

Locating in Crowdsourcing-based DataSpace: Wireless Indoor Localization without Special Devices

Yuanfang Chen · Lei Shu · Antonio M. Ortiz · Noel Crespi · Lin Lv

Received: date / Accepted: date

Abstract Locating a target in an indoor social environment with a Mobile Network is important and difficult for location-based applications and services such as targeted advertisements, geosocial networking and emergency services. A number of radio-based solutions have been proposed. However, these solutions, more or less, require a special infrastructure or extensive pre-training of a site survey. Since people habitually carry their mobile devices with them, the opportunity using a large amount of crowd-sourced data on human behavior to design an indoor localization system is rapidly advancing. In this study, we first confirm the existence of crowd behavior and the fact that it can be recognized using location-based wireless mobility information. On this basis, we design "Locating in Crowdsourcing-based DataSpace" (LiCS) algorithm, which is based on sensing and analyzing wireless information. The process of LiCS is crowdsourcing-based. We implement the prototype system of LiCS. Experimental results show that LiCS achieves comparable location accuracy to previous approaches even without any special hardware.

Yuanfang Chen, Antonio M. Ortiz, Noel Crespi
Institut Mines-Télécom, Télécom SudParis, France
E-mail: yuanfang.chen@etu.upmc.fr,
antonio.ortiz_torres@telecom-sudparis.eu, noel.crespi@mines-telecom.fr

Lei Shu
Guangdong University of Petrochemical Technology, China
E-mail: lei.shu@lab.gdupt.edu.cn

Lv Lin
Dalian University of Technology, China
E-mail: lvlin_george@mail.dlut.edu.cn

1 Introduction

Mobile indoor localization and navigation have become very popular in recent years [8,18]. Some mobile indoor localization systems have been successfully used in a great number of applications such as the location detection of products stored in a warehouse, the location detection of medical personnel or equipment in a hospital, the location detection of firemen in a building on fire, the location detection of police dogs trained to find explosives in a building, and finding tagged maintenance tools and equipment scattered all over a plant. However, a challenging issue remains for mobile indoor localization. A number of existing approaches require infrastructures (e.g., indoor beacons) to achieve reliable accuracy or extensive pre-training before system deployment (e.g., WiFi signal fingerprinting).

With the dramatic increase of the size and frequency of mass events and the potential functionality of mobile devices, the study of crowd dynamics based on Mobile Networks has become an important research area [9, 12]. Firstly, as a base of crowdsourcing-based technology, the crowd behavior of human is worthy to be deeply studied, whether in the virtual world or in the real world. By analyzing and studying the crowd behavior, we can extract some useful conclusions about how humans behave when they are in large groups. For the crowdsourcing-based technologies such as Quora, Yahoo Answers and Google Answers, using the extracted conclusions at group and community levels [22] (a group or a community can be considered as a crowd) will be helpful to develop and improve this type of technology. Furthermore, predicting the formation of a crowd is helpful in some emergency situations, e.g., evacuation route control; and even the prediction is also beneficial for studying and improving the performance of public infrastructures, e.g., network usage during a mass event. Secondly, with the increasing ubiquity of sensing the real world context with smart mobile devices (e.g., the physical location of a mobile device) [10,17], we investigate the rising opportunities for mobile crowdsourcing to obtain real-time physical data. Based on the data, a bridge between the virtual world and the real world can be built. It also means that the crowdsourcing is extended from the virtual world to the real world to help solve many real-world problems. Hence, the crowdsourcing technology can be used for the location estimation problem in the physical world, and can solve the challenging problem of mobile indoor localization, to achieve "special-infrastructure-free"[1] and even "pre-training-free".

In this article, we first focus on recognizing human crowd behavior by analyzing the data measured by internet-accessible mobile phones from a location-aware online social network. By crowd behavior recognition, we understand that the movement of a large number of individuals has a pattern and can be attributed, depending on relevant parameters such as the friendship between individuals[2] and check-in locations (with time) of these individuals.

Then, we propose LiCS, an indoor localization algorithm that considers trace data from individuals' mobile devices and a location estimation model. The trace data includes the following information: MAC addresses of devices, MAC addresses of signal transmitters and corresponding RSSIs (generally, the RSSI (Received Signal Strength Indication) can be used to indicate the value of RSS). With LiCS, mobile devices periodically report their trace data to a Data Analysis Center (DAC). The DAC runs a machine learning algorithm that accepts the wireless-based trace data as features of user mobility patterns, and periodically estimates the locations of mobile devices in real time. Since we use wireless information obtained from the social environment around us, LiCS can achieve fine-grained localization.

To validate this design, we implement a prototype system to conduct long-term experiments in two research laboratories and a corridor of a middle-size academic building covering over $39,725m^2$. Experimental results show that LiCS achieves comparable location accuracy to previous approaches even without a site survey. Moreover, LiCS provides a room-level localization service.

The main contributions of this work are summarized as follows:

1. We design a crowd recognition model. One of the main challenges in crowd behavior recognition is to infer the most likely crowd behavior using the data collected from a set of persons. We use check-in times and locations ($Time$ and *Location_id*; we convert each latitude/longitude coordinate of the earth into a unique *Location_id*) to quantify the trace of each individual. Then, a clustering algorithm[3] is used to find the likely crowds.
2. We design LiCS, an indoor localization algorithm for social environments based on the target's trace data. LiCS utilizes automatic selftraining for target trace data without any specific configuration for mobile devices.
3. We implement the prototype system of LiCS on Android devices, and perform an extensive set of experiments.

The rest of this article is organized as follows. Section 2 introduces related work. Section 3 describes our system model. Section 4 shows the existence of human crowd behavior in our daily lives and how to recognize it from the trace data of individuals. The "Locating in Crowdsourcing-based DataSpace" (LiCS) algorithm is presented in Section 5, while the evaluation of LiCS algorithm and the analysis of results are shown in Section 6. The article concludes with a brief summary in Section 7.

2 Related Work

In this study, the related work concerns two main aspects: (i) mobile-device-based indoor localization; (ii) crowdsourcing-based technology and crowdsourcing-based indoor localization (crowd-sourced data can be used to improve the performance of indoor localization).

2.1 Wireless Indoor Localization

In the mobile computing community, a user can carry a sensing device (such as a smart phone) to move randomly or within a field of fixed sensors [3]. In either case, the

[1] A "special infrastructure" means that the infrastructure consists of customized equipment. LiCS is based on Received Signal Strength (RSS) that exists in any wireless equipment, so LiCS can be directly supported by existing wireless infrastructures around us.

[2] For a location-aware online social network, if B is in the friend list of A, we consider that there is friendship between A and B, and the relationship is directed.

[3] An Expectation-Maximization (EM) clustering algorithm [7] is used in this article. The EM assigns a probability distribution for each trace record (instance), which indicates the probability of each instance belonging to each of the clusters. The EM can automatically decide how many clusters to create.

knowledge of user's location can be useful for some applications (called Location-Based Services (LBS) [6]). Outdoor localization is well solved by GPS, but indoor localization remains a challenge in many cases. For the indoor localization issue, a number of algorithms have been proposed in the past two decades. Generally, these algorithms fall into two categories: (i) some works are based on sensors or beacons which are installed throughout the environment [15][16][2]; so special hardware is required to run these algorithms; (ii) in recent approaches, the location is determined by observing RSS. Two techniques are widely used in this kind of algorithms: fingerprinting-based and model-based technique. A large body of indoor localization algorithms adopt fingerprint matching as the basic scheme of location determination. The main idea of this technique is to fingerprint the surrounding signatures at some locations in the area of interest and then build a fingerprint database. The location of a user is then estimated by matching the new measured fingerprint of the user against the database. Many kinds of signatures have been exploited by researchers such as WiFi signals [19] (e.g., LiFS algorithm [20]), Radio Frequency (RF) signals [1][21] and Frequency Modulation (FM) radio signals [4]. The considerable manual costs and efforts, in addition to the inflexibility to environmental dynamism, are the main drawbacks of fingerprinting-based algorithms. The model-based technique calculates locations based on geometrical or statistical models rather than searches for best-fit signatures from pre-built reference databases [14]. For instance, the prevalent Log-Distance Path Loss (LDPL) model [11] builds up a semi-statistical function between RSS values and RF propagation distances. However, these model-based algorithms trade the measurement efforts at the cost of decreasing localization accuracy.

2.2 Crowdsourcing-based Technology

Crowdsourcing is one specific form of harvesting wisdom and contributions from individuals of crowds; based on this harvesting scheme, some applications or services are developed and the performance of some algorithms can be improved: it can be called crowdsourcing-based technology. Some researchers explore crowdsourcing based on the use of mobile devices. Moreover, as an important basic service, wireless mobile-device-based indoor localization technologies such as GSM-based, Bluetooth-based or WiFi-based localization, have been extended by using crowdsourcing data to support and improve the performance of location estimation [13] [20]. Current crowdsourcing-based localization approaches depend on calibration of the space of interest. Such calibration tends to be onerous, because it has to be repeated for every new space and each time there is a significant change in a given space (e.g., a change in the placement of signal transmitter). LiCS is aimed at eliminating the need for such explicit calibration effort and the need for any kind of map concerning the space of interest. LiCS exploits the advantage of model-based technique (versatility and conciseness) and avoids its drawback (accuracy loss for localization) by training a localization model using real-time trace data of individuals in crowds from the physical world. Figure 1 shows an example for the motivation of applying LiCS and the system architecture of LiCS.

3 System Model

The prototype system is developed on Android smartphones and follows mobile-based network-assisted architecture (Fig. 1(b)). Our system model can be considered as a wireless network where there are N fixed signal transmitters $T = \{t_1, t_2, ..., t_i, ..., t_N\}$ and M mobilizable signal receivers $R = \{r_1, r_2, ..., r_i, ..., r_M\}$. The parameter $p(t)$ denotes the estimate of a mobile terminal location at time t. And the parameter $p'(t)$ is used to denote the real location of a mobile terminal at time t. Moreover, the parameter $d(t)$ is the measured distance between the estimated location $p(t)$ and the real location $p'(t)$, and we use the "step" to measure the distance between $p(t)$ and $p'(t)$ as the localization error at time t in our evaluation experiments (Section 6).

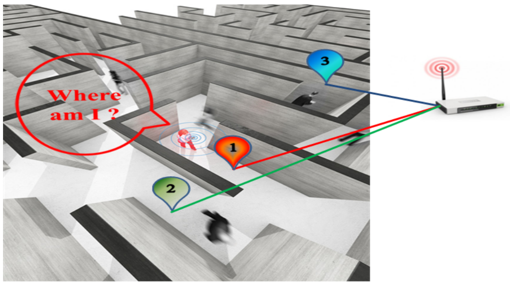
The problem of crowdsourcing-based location estimation can be defined as an identification procedure. The matched fingerprints can be identified from the model-assisted fingerprint database. And the model learns the real environment with wireless signals (RSS) and can be trained in real time with the wireless information that is submitted by mobile terminals.

4 Crowd Behavior Recognition

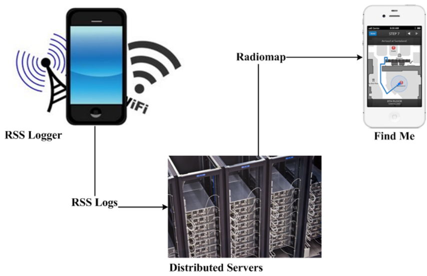
As the basis of crowdsourcing-based location estimation, we need to confirm the existence of crowd behavior; and it can be recognized from wireless mobile data of people's daily lives. In this section we formalize a series of processing steps which can be used to infer crowd behavior from location-based wireless information. We also present a mathematical model for recognizing the crowd behavior of a population.

First, based on the collected data from a location-based mobile social network using its public API [5][4],

[4] The collected data with anonymous mobile devices from Brightkite is used to correlate, model, evaluate and analyze

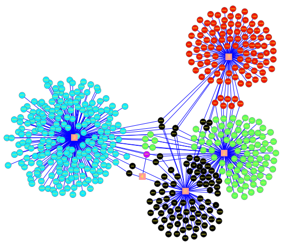


(a) Motivation example. The location-finding ability of indoor localization systems heavily depends on the broadcasts from wireless equipment. For LiCS, any individual can use the trained model to estimate his/her location, and then his/her RSS can be used to train the model again. With the crowd-sourced RSS, the model is trained to make its parameters accordant with the practical situations.

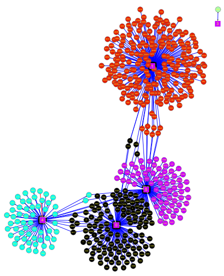


(b) System architecture. As an improvement, LiCS exploits the advantage of model-based technique (versatility and conciseness) and avoids its drawback with the training for a location estimation model: crowd-sourced data is used to train the model and the model is installed in distributed servers.

Fig. 1 Motivation example and system architecture.

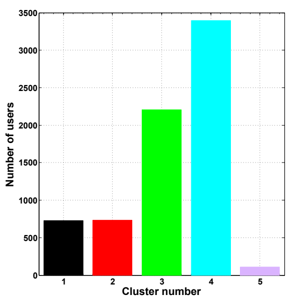


(a)

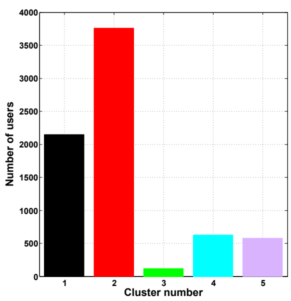


(b)

Fig. 2 (a) Clustering result with *Friendship* attribute. (b) Clustering result with *Location_id* attribute. For clearness, we only show one-month clustering result using *Friendship* and *Location_id* attribute as the classes of clustering evaluation, respectively. 5 clusters can be found for the two clustering processes, and we use different colours to distinguish different clusters.



(a)



(b)

Fig. 3 Number of users for each cluster. (a) Number of users for each cluster using *Friendship* attribute as the class of clustering evaluation. (b) Number of users for each cluster using *Location_id* attribute as the class of clustering evaluation.

we confirm whether the crowd behavior of individuals can be identified. Figure 2 shows the results of data clustering with *Friendship* and *Location_id* attribute, respectively (these two attributes are evaluation classes of clustering), using the Expectation-Maximization (EM) algorithm [7]. From Fig. 2, the crowd behavior (we define "crowd" in Definition 1) has been recognized using the processed data (in our study we process the original collected data). Moreover, we can find that different evaluation classes have different clustering accuracy levels, so different attributes have different impacts on crowd behavior. Figure 3 shows the number of users for each cluster corresponding to Fig. 2 (Fig. 2 only shows the users who are "1-hop" friends, but in Fig. 3, the users who are multihop friends are also counted for each cluster). For Fig. 3, more detailed explanations are necessary: why some clusters are composed by thousands of users when we use "Friendship" attribute as the class of clustering evaluation? Because (i) we use "multihop" friends in the clustering evaluation; that is to say, if A is a friend of B within the range r and B is a friend of C within the range r, A, B and C all will belong to a same crowd; (ii) the error of clustering evaluation is existent; (iii) we use the evaluation dataset which is from one special month, e.g., some data of the dataset is from a major celebration (a great number of users are gathered together), to show the impact of attributes on crowd behavior[5].

the relationships between the check-in time, locations, friendship and crowd behavior of users in 772,966 distinct places. The data consists of 58,228 nodes (users) and 214,078 friend edges (friendship is directed between any two nodes).

[5] Even if the dataset is incomplete, it still can be used to show that "the impact of attributes (friendship and check-in locations) is existent on crowd behavior".

Definition 1 A crowd can be defined as follows: a group of individuals at a "same" physical location (the range radius r for each user is $10m$) and at the "same" time (the time range for a crowd is set to 15 minutes; namely, if the check-in time difference between two users is within 15 minutes, we consider that they are "at the same time").

The characteristics of crowd behavior for each single person can therefore be inferred from his/her check-in records. We refer to this as their "individual behavior". This shows which individuals participate in a specific crowd. From the EM algorithm, a given record belongs to each cluster with certain probabilities. Moreover, likelihood is a measurement of "how good" a clustering process is and it increases in the successive iterations of EM algorithm. It is worth mentioning that the higher the likelihood, the better the model fits the data.

Second, the clustering process (crowd behavior recognition model) is described as follows. We define two parameters: (i) the user u's check-in data S^u which is a sequence of activity observations for user u; (ii) a set of unknown values θ (i.e., the serial numbers of clusters). These two parameters are used along with a Maximum Likelihood Estimation (MLE): $L(\theta; S^u) = p(S^u|\theta)$. Our purpose is to seek the MLE of marginal likelihood. In other words, we need to find the most probable θ which the user u belongs. The EM algorithm iteratively applies the following two steps to achieve our purpose:

1. Expectation step (E step): calculate the expected value of the log-likelihood function under the current established clusters ($\theta^{(t)}$):

   $$Q(\theta|\theta^{(t)}) = E_{S^u,\theta^{(t)}}[\log L(\theta; S^u)];$$

2. Maximization step (M step): find the appropriate value of parameter θ, which maximizes this quantity:

   $$\theta^{mle} = \arg\max_{\theta} Q(\theta|\theta^{(t)}).$$

MLE estimates θ by finding a value of θ that maximizes $Q(\theta|\theta^{(t)})$, and the estimation result can be flagged as: θ^{mle}.

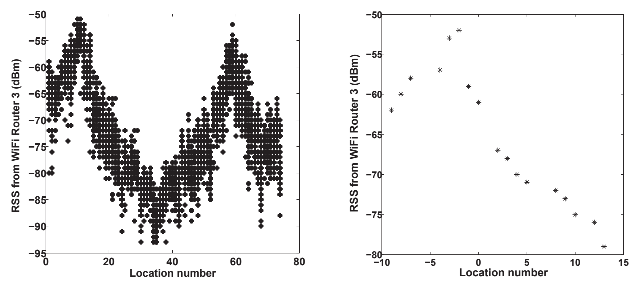
Further, based on our recognition model, we add the spatio-temporal pattern of crowd into crowd behavior (clustering). We use a triple $q = (\theta, p_i, t_i)$ to replace θ. Then the expectation-maximization process becomes:

1. Expectation step: $Q(q|q^{(t_i)}) = E_{S^u,q^{(t_i)}}[\log L(q; S^u)]$, where p_i is the position characteristic of location measurement m_i, and t_i is the timestamp of m_i. Moreover, $q^{(t_i)}$ is a set of current established clusters with their locations and timestamps;
2. Maximization step: choose q to maximize $Q(.)$, $q^{mle} = \arg\max_{q} Q(q|q^{(t_i)})$.

Finally, in order to preserve the integrity of our model, the recognition accuracy must be measured. In our crowd behavior recognition model, the value of the log-likelihood can be used to measure the accuracy. For instance, using $Friendship$ attribute as the evaluation class, based on one-month data, the log likelihood of crowd identification is: -16.42186, and using *Location_id* attribute as the evaluation class, the log likelihood is: -16.87742. Their accuracy is different, and $Friendship$ attribute is more effective for improving the recognition ability of the model.

5 Locating in Crowdsourcing-based DataSpace

As an effective measurement, RSS is easily available from various wireless signals which are from most off-the-shelf wireless equipment such as WiFi- or Bluetooth-compatible devices. And a large number of RSS-based indoor localization algorithms are proposed. However, considering RSS as a database to support indoor localization (e.g., RSS fingerprint space), it is time-consuming and labor-intensive. Especially, from extensive experiments, we observe that the RSS is vulnerable due to environmental dynamism (an example is shown in Fig. 4(a)). How to avoid these weaknesses to improve the performance of RSS-based indoor localization? It is worth noting that the trend of RSS change is obvious between different locations (Figure 4(b)).



(a) An example: the fluctuation range of RSS for some locations. For example, at the location 60, the RSS is not a value, it is a range of values. So the fingerprint of a location cannot be denoted by the absolute value of RSS.

(b) Changing trend of RSS. $location\ number = 0$ is the location of a signal transmitter; as a signal receiver moves away from the transmitter, the RSS is decreasing.

Fig. 4 Instability of RSS and changing trend between different locations.

To try to avoid these weaknesses, we present LiCS and the details are shown as follows.

Input: Signal triples[6] from individuals.

First, at time $t+1$, the value $p(t+1)$ can be estimated based on "observed locations" $p(t), p(t-1), p(t-2), ..., p(t-k+1)$. Moreover, the relationship between the output $p(t+1)$ and the input $p(t), p(t-1), p(t-2), ..., p(t-k+1)$ has the following mathematical representation (Eq. 1):

$$p(t+1) = \alpha_0 + \sum_{j=1}^{q} \alpha_j g(\beta_{0j} + \sum_{i=1}^{p} \beta_{ij} p(t-i+1)) + \epsilon, \quad (1)$$

where $\alpha_j (j = 0, 1, 2, ..., q)$ and $\beta_{ij} (i = 0, 1, 2, ..., p; j = 1, 2, 3, ..., q)$ are the connection weights between time series, p is the number of "observed locations", q is the number of nodes of hidden layer[7] and ϵ is noise of the estimation. The logistic function $g(x) = \frac{1}{1+e^{-x}}$ is used as a hidden-layer transfer function. In this study, an optimal location estimation model is built by training the model with the wireless data collected from the real physical space around us (the collected data can be denoted as signal triples). The training steps are shown as follows:

Step 1: Cluster the signal triples. Partition all triples into several clusters, using an Expectation-Maximization (EM) clustering algorithm. Moreover a cluster center can be obtained for each cluster. Each cluster is given a unique number as its location.

Step 2: Input some selected time-serial signal triples with corresponding locations of clusters into Eq. 1 for learning the optimal configuration of parameters, α_j, β_{ij} and ϵ. A location estimation model with optimal parameter configuration can be obtained. Moreover, the growth of logistic function $g(.)$ satisfies: the initial stage of growth is approximately exponential; then, as saturation begins, the growth slows, and at maturity, the growth stops. So if the training time is long enough, the parameter configuration of Eq. 1 will gradually converge to the optimal solution.

Then, a target can be located with the optimal location estimation model. (i) Give the target a start location $p(0)$. Calculate the Euclidean distances between a received new signal triple and all cluster centers (the new signal triple is from the target). If the shortest Euclidean distance is relative to the cluster k, $p(0) = k$. (ii) Calculate the location at time $t+1$. Using the trained Eq. 1 as "optimal location estimation model", from the start location $p(0)$, we can obtain time-serial locations. And based on "observed locations", $[p(t), p(t-1), ..., p(0)]$, $p(t+1)$ can be calculated.

The real-time location of a target is obtained by our algorithm. Moreover, (i) the optimal location estimation model is periodically trained by new signal triples; and (ii) if more signal transmitters can be detected by receivers, it will help to distinguish different locations more effectively for achieving higher localization accuracy.

Output: The real-time location (cluster number) of a target. For some special applications, if the absolute coordinates of clusters are available, the absolute physical location of target will be known.

Note that Eq. 1 is the core of LiCS. From above descriptions of the algorithm LiCS, we can find that these parameters will affect the accuracy of LiCS: the connection weights between time series, α_j and β_{ij}. The parameter α_j reflects the importance of hidden-layer transfer for the location estimation of time $t+1$. In other words, it denotes the degree of correlation between different time series. The parameter β_{ij} reflects the importance of the j^{th} node in the hidden layer, when the hidden layer transfers the influence of observed location $p(t-i+1)$ to $p(t+1)$. The appropriate values of these parameters for the model of location estimation at time $t+1$ will be helpful to improve the accuracy of localization.

The variables that are used in LiCS are summarized in Tab. 1.

6 Evaluation

We develop the prototype system of LiCS on the increasingly popular Android OS which supports WiFi and Bluetooth. We conduct long-term experiments in two laboratories ($84m^2$ and $53m^2$, respectively) and a corridor (Fig. 5) of a middle-size academic building where a number of WiFi routers without location information have been installed. Moreover, in each experimental site, we install three Bluetooth transmitters (laptop-embedded Bluetooth transmitters are used in our experiment, so they are not special devices; the signal of Bluetooth is full-coverage for each experimental site). The experiment lasts one month using 9 volunteers. We measure WiFi signals and Bluetooth signals.

[6] A triple can be denoted as $[RSS, MAC_T, MAC_R]$. For the specific RSS of a location, MAC_T is the MAC address of corresponding signal transmitter and MAC_R is the MAC address of corresponding signal receiver.

[7] In machine learning, using the hidden layer enables greater processing power and system flexibility. The nodes of hidden layer are named as hidden nodes. Hidden nodes are the nodes that are neither in the input layer nor the output layer. These nodes are essentially hidden from view, and their number and organization can typically be treated as a black box to people who are interfacing with the system.

Table 1 Variables and explanations

Variables	Explanations
$[RSS, MAC_T, MAC_R]$	A triple consists of three elements, where RSS is Received Signal Strength received by a mobile terminal, MAC_T is the MAC address of corresponding signal transmitter and MAC_R is the MAC address of corresponding signal receiver. The triple is used to train the location estimation model and structure the fingerprint database.
$p(t)$	We use the variable to denote the location of a mobile terminal at time t.
α_j	The variable is the connection weight between time series, and it reflects the importance of hidden-layer transfer for the location estimation of time $t+1$.
β_{ij}	The variable is the connection weight between time series, and it reflects the importance of j^{th} node in the hidden layer, when the hidden layer transfers the influence of observed location $p(t-i+1)$ to $p(t+1)$.

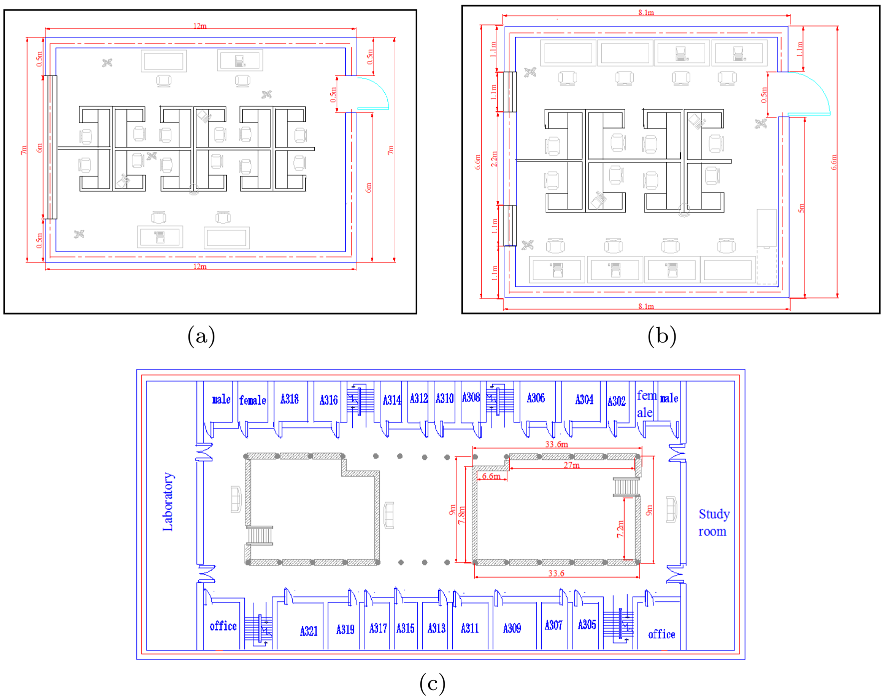


Fig. 5 Floor plans of experimental sites. (a) Laboratory covering over $84m^2$. (b) Laboratory covering over $53m^2$. (c) Corridor covering over $302m^2$.

6.1 Experimental Setup

Each volunteer carries a mobile phone and can take any activity in any area of experiment. The trace data of each volunteer is recorded every 30 seconds during working hours (from 9:00 a.m. to 10:00 p.m.). Moreover, the trace data from volunteers covers most of the areas of experiment.

In our experiments, for LiCS, we use WiFi and Bluetooth signals. Bluetooth is a wireless technology standard for exchanging data over short distances, so Bluetooth signals attenuate more rapidly with distance compared with WiFi signals.

We compare our algorithm with LiFS [20] under the same experimental conditions. LiFS is an RSS-based indoor localization algorithm (using WiFi signals). The key idea behind LiFS is that human motion can be applied to connect previously independent radio fingerprints under certain semantics. In LiFS, absolute values of RSS are used to establish a fingerprint database. When a user sends a location query with his/her current RSS fingerprint, LiFS retrieves the fingerprint database and returns the matched fingerprints as well as the corresponding locations.

6.2 Performance Evaluation

In this section, we show the comparative results of LiFS and LiCS. We estimate 248 location queries, and cumulate all localization errors of these queries (Cumulative

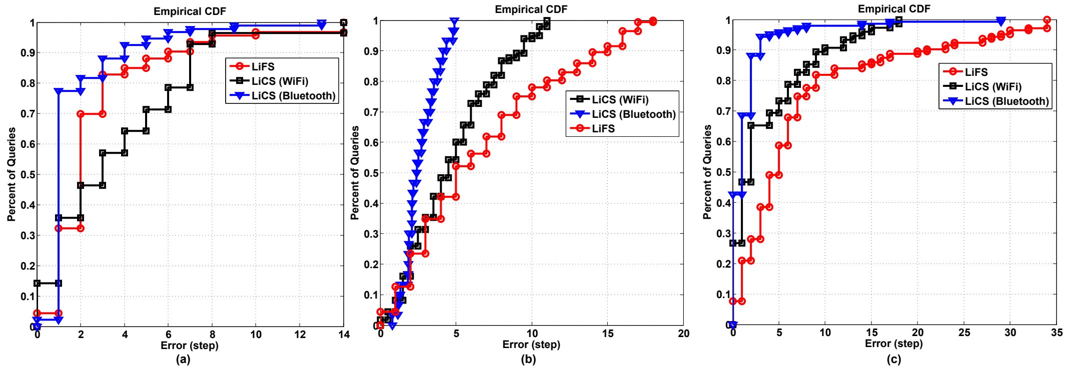


Fig. 6 CDFs of localization errors for both algorithms in three different experimental sites. (a) CDF of localization errors in the laboratory covering over $84m^2$. (b) CDF of localization errors in the laboratory covering over $53m^2$. (c) CDF of localization errors in the corridor covering over $302m^2$.

Distribution Function (CDF)[8]) for both algorithms, respectively. The results are shown in Fig. 6 and Fig. 7.

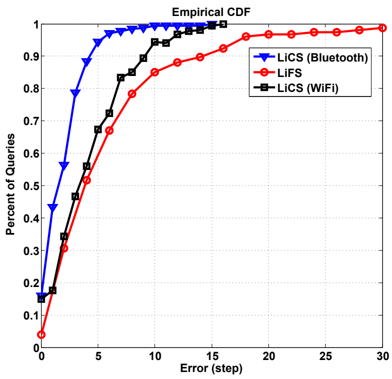


Fig. 7 CDF of localization errors for the average of three experimental sites.

The unit of the estimated error for localization is *step* $\approx 1.2m$. The average localization error of LiCS with Bluetooth signals is $2.6575m$, LiCS with WiFi signals is $4.35m$, and LiFS is $5.95m$. The maximum localization error of LiCS with Bluetooth signals is $33.6m$, LiCS with WiFi signals is $34.8m$ and LiFS is $40.8m$.

For comparing the overall performance of three experimental sites for both algorithms, we calculate the average results of three experimental sites for LiCS (Bluetooth), LiCS (WiFi) and LiFS. From Fig. 7, we can find that our LiCS is better than LiFS for the average of three experimental sites. For example, the localization errors of 95% queries are less than $6m$ for LiCS (Bluetooth), 69% queries are less than $6m$ for LiCS (WiFi) and 60% for LiFS. LiCS uses the model training with real-time data, so the localization accuracy for location queries is improved compared with LiFS. Moreover, why the localization accuracy of LiCS (Bluetooth) is higher than LiCS (WiFi)? Based on the attenuation characteristics of Bluetooth signals and WiFi signals, the differences of RSS between different locations for Bluetooth are greater than the differences for WiFi. So using the RSS of Bluetooth to distinguish different locations is more accurate than using the RSS of WiFi. If we can distinguish different locations more clearly, we can obtain higher localization accuracy based on an RSS-based fingerprint database.

Furthermore, from the experimental results of Fig. 6, we can find that: (i) on average, for localization accuracy, LiCS is better than LiFS, in the three different experimental sites. For example, the localization errors of 80% queries are less than $2.4m$ for LiCS (Bluetooth), and 70% for LiFS, in the laboratory covering over $84m^2$; (ii) the localization accuracy of LiCS (Bluetooth) is better than LiCS (WiFi) and LiFS. For example, the localization errors of 50% queries are under $2.4m$ for LiCS (Bluetooth), while about 30% for LiCS (WiFi) and about 25% for LiFS, in the laboratory covering over $53m^2$. Bluetooth improves the average localization error up to 39% compared with LiCS (WiFi), and up to 55% compared with LiFS. Because the signal strength of Bluetooth is changed sharply between locations, which makes the distinction of signal strength between different locations more remarkable (the "remarkable" is conducive to improving the accuracy of localization).

Moreover, LiFS is based on a priori database (some human intervention is necessary in the build phase of a database). LiCS is crowd-sourced, so only wireless information is required, which is received by humans in their daily lives. And the locating process of LiCS is

[8] Cumulative Distribution Function describes the probability that a real-valued random variable X with a given probability distribution will be found at a value less than or equal to x. It can be formulated as $F_X(x) = P(X \leq x)$.

automatic and priori-database-free. LiCS is based on WiFi and Bluetooth information which is readily available. The above-mentioned features of LiCS make the rapid deployment of system possible.

7 Conclusion

Crowdsourcing is a distributed problem-solving scheme that has emerged in recent years. It exploits the potential and wisdom of crowds to support various applications and to improve the performance of various algorithms in a cost-effective fashion. In our study, we investigate the existence of crowd behavior in the real world and it can be recognized using location-aware wireless mobility information of people's daily lives. Furthermore, we present a model for crowd behavior recognition. On this basis, by sensing and collecting WiFi and Bluetooth information from the social surroundings around us, we propose a time-serial location estimation model. Using this model, we design and implement LiC-S, an indoor target localization algorithm based on two aspects, (i) mobile devices carried by individuals, and (ii) a location estimation model which is trained by the collected data from individuals about WiFi and Bluetooth information. The experimental results show that LiCS achieves competitive location accuracy without any special infrastructure. This work sets up a novel perspective to crowd-sourced indoor localization algorithms.

Acknowledgment

Lei Shu's work is supported by the Guangdong University of Petrochemical Technology Internal Project (2012RC0106).

References

1. Bahl, P., Padmanabhan, V.N.: Radar: An in-building rf-based user location and tracking system. In: Proceedings of the 19th Annual International Conference on Computer Communications (INFOCOM), pp. 775–784. IEEE (2000)
2. Borriello, G., Liu, A., Offer, T., Palistrant, C., Sharp, R.: Walrus: wireless acoustic location with room-level resolution using ultrasound. In: Proceedings of the 3rd International Conference on Mobile systems, Applications, and Services (MobiSys), pp. 191–203. ACM (2005)
3. Chen, M., Gonzalez, S., Vasilakos, A., Cao, H., Leung, V.C.: Body area networks: A survey. Mobile Networks and Applications **16**(2), 171–193 (2011)
4. Chen, Y., Lymberopoulos, D., Liu, J., Priyantha, B.: Fm-based indoor localization. In: Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services (MobiSys), pp. 169–182. ACM (2012)
5. Cho, E., Myers, S.A., Leskovec, J.: Friendship and mobility: user movement in location-based social networks. In: Proceedings of the 17th International Conference on Knowledge Discovery and Data Mining (SIGKDD), pp. 1082–1090. ACM (2011)
6. Dey, A., Hightower, J., de Lara, E., Davies, N.: Location-based services. Pervasive Computing **9**(1), 11–12 (2010)
7. Do, C.B., Batzoglou, S.: What is the expectation maximization algorithm? Nature Biotechnology **26**(8), 897–899 (2008)
8. Gu, Y., Lo, A., Niemegeers, I.: A survey of indoor positioning systems for wireless personal networks. IEEE Communications Surveys & Tutorials **11**(1), 13–32 (2009)
9. Gupta, A., Kalra, A., Boston, D., Borcea, C.: Mobisoc: a middleware for mobile social computing applications. Mobile Networks and Applications **14**(1), 35–52 (2009)
10. Győrbíró, N., Fábián, Á., Hományi, G.: An activity recognition system for mobile phones. Mobile Networks and Applications **14**(1), 82–91 (2009)
11. Jung, S., Lee, C.o., Han, D.: Wi-fi fingerprint-based approaches following log-distance path loss model for indoor positioning. In: Proceedings of the 1st MTT-S International Microwave Workshop Series on Intelligent Radio for Future Personal Terminals, pp. 1–2. IEEE (2011)
12. Kjærgaard, M.B., Wirz, M., Roggen, D., Troster, G.: Mobile sensing of pedestrian flocks in indoor environments using wifi signals. In: IEEE International Conference on Pervasive Computing and Communications (PerCom), p-p. 95–102. IEEE (2012)
13. Rai, A., Chintalapudi, K.K., Padmanabhan, V.N., Sen, R.: Zee: Zero-effort crowdsourcing for indoor localization. In: Proceedings of the 18th Annual International Conference on Mobile Computing and Networking (MobiCom), pp. 293–304. ACM (2012)
14. Roos, T., Myllymaki, P., Tirri, H.: A statistical modeling approach to location estimation. IEEE Transactions on Mobile Computing **1**(1), 59–69 (2002)
15. Savvides, A., Han, C.C., Strivastava, M.B.: Dynamic fine-grained localization in ad-hoc networks of sensors. In: Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MOBICOM), pp. 166–179. ACM (2001)
16. Scott, J., Dragovic, B.: Audio location: Accurate low-cost location sensing. Pervasive Computing **3468**, 1–18 (2005)
17. Tang, F., Guo, S., Guo, M., Li, M., Wang, C.l.: Towards context-aware ubiquitous transaction processing: a model and algorithm. In: IEEE International Conference on Communications (ICC), pp. 1–5. IEEE (2011)
18. Varshney, U., Vetter, R.: Mobile commerce: framework, applications and networking support. Mobile Networks and Applications **7**(3), 185–198 (2002)
19. Wu, K., Xiao, J., Yi, Y., Gao, M., Ni, L.M.: Fila: fine-grained indoor localization. In: Proceedings of the 31st Annual International Conference on Computer Communications (INFOCOM), pp. 2210–2218. IEEE (2012)
20. Yang, Z., Wu, C., Liu, Y.: Locating in fingerprint space: wireless indoor localization with little human intervention. In: Proceedings of the 18th Annual International Conference on Mobile Computing and Networking (MobiCom), pp. 269–280. ACM (2012)
21. Youssef, M., Agrawala, A.: The horus wlan location determination system. In: Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services (MobiSys), pp. 205–218. ACM (2005)
22. Zhang, D., Guo, B., Yu, Z.: The emergence of social and community intelligence. Computer **44**(7), 21–28 (2011)