HTrack: An Efficient Heading-Aided Map Matching for Indoor Localization and Tracking

Yongfeng Wu, Pan Chen, Fuqiang Gu, *Student Member, IEEE*, Xiaoping Zheng, and Jianga Shang, *Member, IEEE*

***Abstract*— Indoor localization has become a hot topic in recent years because of its wide applications. Map matching is a popular method used to improve the localization accuracy without adding hardware. However, the existing map matching methods are usually computationally expensive, leading to the unsuitability of running on resource-limited devices such as smartphones. In this paper, we present an efficient map matching system for indoor localization, called *HTrack*, which uses a hidden Markov model, considering the user's heading and spatial information. By considering user's heading information, we significantly reduce the number of candidate states for each step, and hence improve the computational efficiency. The experimental results show that the *HTrack* outperforms the state-of-the-art methods (more than 25% localization accuracy improvement), and consumes about five times less energy than the state-of-the-art methods.**

***Index Terms*— Indoor localization, map matching, Wi-Fi fingerprinting, hidden Markov model.**

I. Introduction

INDOOR localization and tracking are important for a wealth of applications such as personnel tracking, healthcare, location-based marketing and advertising, and location-based social networks. A number of indoor localization and tracking methods have been proposed, which vary from each other in terms of the used localization technique, accuracy, coverage, computational complexity, and cost of deployment and maintenance [1]–[4]. Each method has its own advantages and limitations. For example, the method based on ultra-wide band (UWB) technique [5] can achieve a centimeter-level accuracy, but it requires specific hardware that is expensive; Wi-Fi-based localization [6] is cheap since it can make use of existing infrastructure, but the achieved accuracy is relatively low.

Manuscript received November 22, 2018; accepted December 21, 2018. Date of publication January 7, 2019; date of current version March 18, 2019. This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFB0502200, in part by the National Natural Science Foundation of China under Grant 41271440, and in part by the Fundamental Research Funds for National University, China University of Geosciences, Wuhan, under Grant G1323511822. The associate editor coordinating the review of this paper and approving it for publication was Prof. Huang Chen Lee. *(Corresponding author: Jianga Shang.)*

Y. Wu, P. Chen, X. Zheng, and J. Shang are with the Faculty of Information Engineering, China University of Geosciences, Wuhan 430074, China, and also with the National Engineering Research Center for Geographic Information System, Wuhan 430074, China (e-mail: wuyongfeng@cug.edu.cn; chenpan@cug.edu.cn; zheng.xiaoping@cug.edu.cn; jgshang@cug.edu.cn).

F. Gu is with the Department of Infrastructure Engineering, University of Melbourne, Parkville, VIC 3010, Australia (e-mail: fuqiangg@student.unimelb.edu.au).

Digital Object Identifier 10.1109/JSEN.2019.2891313

In order to overcome the limitations of different methods, one may need to fuse several localization methods (also known as hybrid methods). For instance, the combination of Wi-Fi fingerprinting and pedestrian dead reckoning (PDR) can overcome the failure problem of Wi-Fi fingerprinting in the Wi-Fi signal-blind areas and the accumulative error problem of PDR. Apart from the combination of Wi-Fi fingerprinting and PDR, there are other hybrid methods such as *multimodal fingerprinting* [7], and *triangulation-based fusion* [8]. The main issue of hybrid methods is that the required infrastructure may not be available in many environments or it may be available at a high cost.

A promising solution to achieving accurate localization and tracking with no additional hardware is to fuse the localization methods with spatial information such as a floor plan, and landmarks. Map matching is a commonly-used method that utilizes spatial information to improve localization and tracking accuracy [9]–[11]. It takes as input a sequence of sensor data to identify the correct path segment on which a user is walking and to determine the user's location on that segment. While map matching can achieve a satisfactory accuracy, it is usually computationally expensive and hence are not suitable for being implemented on resource-limited devices such as mobile phones.

In this paper, we propose an efficient method map matching method to improve the localization accuracy as well as the computational efficiency, which is based on a hidden Markov model (HMM) and a fingerprint graph. The reason why we select the HMM is that HMM can consider the temporal relationship, which is suitable for localization as the current location depends on the previous location. Different from traditional trajectory matching methods (e.g. *VTrack* [9]), where the input is a sequence of noisy locations derived from Wi-Fi receive signal strength (RSS) measurements, our model takes as input raw Wi-Fi RSS measurements to avoid the information loss when transforming RSS measurements into locations. Compared to the state-of-the-art systems, our method achieves much higher accuracy and consumes less computational time.

In summary, our main contributions in this paper are as follows:

- We propose an efficient map matching method for indoor localization and tracking, which is based on a hidden Markov model, and develop a prototype system called *HTrack*. By utilizing user's heading information,

we significantly reduce the number of candidate states for each step and hence improve the computational efficiency.
- We propose the concept of fingerprint graph that fuses Wi-Fi fingerprints with indoor spatial information. The nodes in the fingerprint graph contain not only Wi-Fi fingerprints, but also indoor semantic information (e.g., elevator, and staircase). The edges connecting two nodes are walkable paths with direction information.
- We evaluate our system in a museum building with an area of about 5000 m^2 and an office building (about 3300 m^2). Experimental results show that the proposed approach outperforms the state-of-the-art methods by more than 25% in terms of localization accuracy.

The rest of this paper is organized as follows: in Section II, we review the related work. In Section III, we introduce the overview of the proposed system. In Section IV, we elaborate the proposed heading-aided map matching method. Experimental results and analysis are given in Section V. Finally, we conclude this paper in Section VI.

II. Related Work

In recent years, plenty of indoor localization methods have been proposed and implemented due to the demand of location-based services and applications [12], [13]. Among these methods, Wi-Fi fingerprinting [14]–[16] is one of the most commonly-used methods due to the ubiquity of Wi-Fi infrastructure and the popularity of mobile devices. While Wi-Fi fingerprinting can achieve a relatively satisfactory accuracy, it requires to construct a radio map via site survey. The site survey process is time-consuming and labor-intensive. To reduce the effort and time required to build the radio map, many research works have been done, mainly including crowdsourcing [17] and Wi-Fi SLAM (short for simultaneous localization and mapping) [18]. However, crowdsourcing requires user's active participation or achieves low accuracy, and Wi-Fi SLAM suffers from heavy computational load. In addition to Wi-Fi-based methods, methods based on radio frequency identification (RFID) [19], UWB [20], inertial sensors [21], capacitive sensor [22], pyroelectric IR [23], vision [24], etc. have also been proposed for indoor localization and tracking. Spatial information is often used to enhance the accuracy of these methods without requiring additional hardware.

In the following, we review the related works using spatial information to improve indoor localization, including landmark-based methods, map filtering, and map matching.

A. Landmark-Based Methods

Landmarks are defined as location points in indoor environments where at least one type of sensor data presents a distinctive, stable, and identifiable pattern [25]–[27]. There are several research works that utilize landmarks to enhance indoor localization. An early system using landmarks for indoor localization is *UnLoc* [25], which achieves a median localization error of 1.69m by combining PDR with landmarks. *SemanticSLAM* [26] further extends the *UnLoc* system using the SLAM technique, which decreases the median localization error to 0.53m. An activity landmark-based indoor mapping system is presented in [28], which is called *ALIMC*. By detecting the activity landmarks, *ALIMC* achieves a mapping accuracy of about 0.8-1.5m within the 80th percentile. *APFiLoc* [29] uses a particle filter to fuse PDR, landmarks, and a floor plan, which achieves a localization accuracy of less than 2m with 80% confidence. A system of using the Kalman filter to integrate Wi-Fi, PDR, and landmarks is introduced in [30] where Wi-Fi fingerprinting is used to calculate the initial location and landmarks are used to correct the accumulative error of PDR. It reports an average localization error of less than 1m.

While landmark-based localization seems a promising solution for indoor localization, it faces also several challenges. One major challenge is landmark detection, which involves the design of suitable features and the determination of appropriate threshold value. Existing works usually design detection features manually and set the thresholds of detecting different landmarks by empirical analysis, which may vary from scene to scene. There is a lack of a universal method that can learn useful features of detecting landmarks automatically. Also, it is challenging to deal with the data association problem, which means how to determine the correct landmark when there are multiple landmarks nearby. An additional challenge is the omission issue that certain landmarks may be missed in some cases. For example, a door landmark will be missed if the door is left open since most door detection methods assume that the user conducts a sequence of activity (e.g., walking-standing for opening the door-walking) when passing the door.

B. Map Filtering

Map filtering is a commonly-used method that uses a particle filter to integrate map constraints in order to improve the localization accuracy. The rationale behind map filtering is to use a set of weighted particles to represent the probability distribution of location state [31]. These particles propagate forward according to a state model. If a particle violates spatial constraints (e.g., crossing a wall), its weight will be set to a certain value (e.g., zero) [32]. In this way, these particles violating spatial constraints are removed and thereby the resulted accuracy is higher. A number of map filtering systems have been developed in the literature. An early system using a particle filter to fuse foot-mounted inertial sensor data and spatial information is proposed by Woodman and Harle [32], which achieves a localization error of 0.73m within 95% of the time. *Zee* [33] leverages an augmented particle filter to simultaneously estimate location and user-specific step length. By using a placement-independent step counting and orientation estimation methods, *Zee* achieves an accuracy of under 2m. Yu *et al.* [34] utilizes a two-layer extended Kalman filter and auxiliary value particle filter to integrate Wi-Fi measurements, inertial sensor data, and map information.

The main limitation of map filtering is its heavy computation. To achieve a satisfactory accuracy, map filtering usually requires a large number of particles (e.g., 1000 particles). Particle propagation and continuous detection of whether

particles violate spatial constraints are computationally expensive. To reduce the computational complexity of map filtering, Hilsenbeck *et al.* [35] proposes a graph-based, low-complexity sensor fusion method, which applies a particle filter on a graph model extracted from a floor plan. By constraining the propagation of particles in narrow spaces on one-dimensional edges, the number of required particles are significantly reduced. Liao *et al.* [36] proposes an efficient tracking method that uses a particle filter on a Voronoi graph and achieves an average localization error of about 1.7m. However, the construction of these spatial models is challenging. While manual methods are slow and labor-intensive, automated methods are still in their infancy and not yet applicable in general practical scenarios [37], [38].

C. Map Matching

Map matching can be categorized into two types: *point-to-point matching*, and *trajectory matching*. Point-to-point methods match location points with the places of indoor spaces in the light of floor plans. It is simple and computationally efficient, but it relies heavily on how shape points are utilized in the network and hence is sensitive to the way in which the path network is digitized [39].

Trajectory matching aims to obtain a global optimal estimation by matching the captured trajectory with the geometry and topology information of corners, corridors, and rooms [40]. Trajectory matching is more robust, accurate, but more complex than point-to-point matching. *VTrack* [9] is a vehicle localization system, which improves localization accuracy by using a HMM for map matching of low-power GPS and Wi-Fi localization results. *VTrack* [9] uses the speed limit to remove unreasonable localization results and generates a smooth trajectory by time label based interpolating. Then the generated trajectory is taken as the map matching observation and the road network information is taken as hidden states. Seitz *et al.* [41] propose a HMM-based navigation algorithm that integrates Wi-Fi fingerprinting and PDR, in which the Wi-Fi fingerprints are considered as hidden states and dead reckoning is taken for state transition. WTrack [42] employs a HMM to model the walk pattern of indoor pedestrians and computes the location of a user by continuously sensing the pre-defined walk pattern. *MapCraft* [11] is an indoor pedestrian localization system, which is based on conditional random fields (CRFs). *MapCraft* enhances the raw trajectory by designing various edge feature functions and node feature functions. The edge feature functions represent the transition of hidden states. The node feature functions represent the relationship between the observations and states. Mapel [43] uses CRFs to fuse geomagnetism measurements with pedometer readings to locate a user. Trajectory matching is more robust than point-to-point matching, but it is more complex and has poorer real-time capability.

While existing map matching methods especially trajectory matching methods can significantly improve localization accuracy, they usually have heavy computational load, which limits their applicability on resource-limited devices. Therefore, the main difference between our method and existing map matching methods is that we consider using a fingerprint graph and heading information to reduce the computational cost apart from improving the localization accuracy.

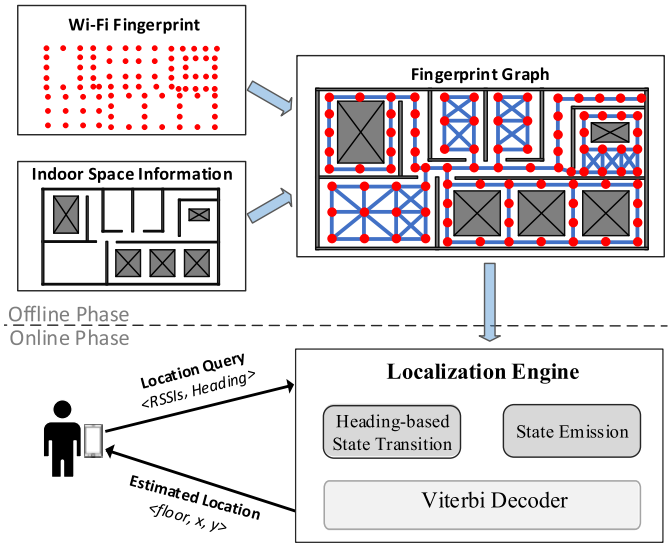

Fig. 1. System architecture.

III. System Overview

The architecture of the proposed system is shown in Fig. 1, which consists of two phases: an offline phase and an online phase. In the offline phase, we collect discrete Wi-Fi fingerprints by a site survey process. Each fingerprint consists of a location and corresponding RSS measurements collected from visible access points (APs) at this location. The collected fingerprints are then combined with indoor spatial information to construct a fingerprint graph, which can be used to localize the user and match the trajectory of the user.

In the online phase, the user walks in the indoor environment carrying a smartphone. To query a location, the smartphone requires to measure and send the collected Wi-Fi RSS measurements and user's heading to the localization engine. The localization engine takes the sequence of these measurements as observations, and the nodes in fingerprint graph as hidden states. Specifically, the localization engine is implemented using a HMM, and it includes three modules, namely heading-based state transition, state emission, and Viterbi decoder. Once an observation is received, the heading-based state transition and state emission modules will calculate the state transition and emission probabilities according to the current state, user's heading, and the fingerprint graph. Then, the Viterbi decoder module finds the most possible sequence of locations and estimates the user's location.

IV. Heading-Aided Map Matching

In this section, we describe the proposed heading-aided map matching based on a HMM. Let $S = \{s_1, s_2, \cdots, s_N\}$ be a set of N hidden states (which are nodes in the fingerprint graph), and $O = \{o_1, o_2, \cdots, o_T\}$ represent a sequence of T observations (which are Wi-Fi RSS measurements and heading measurements). Thus, the localization and tracking using map

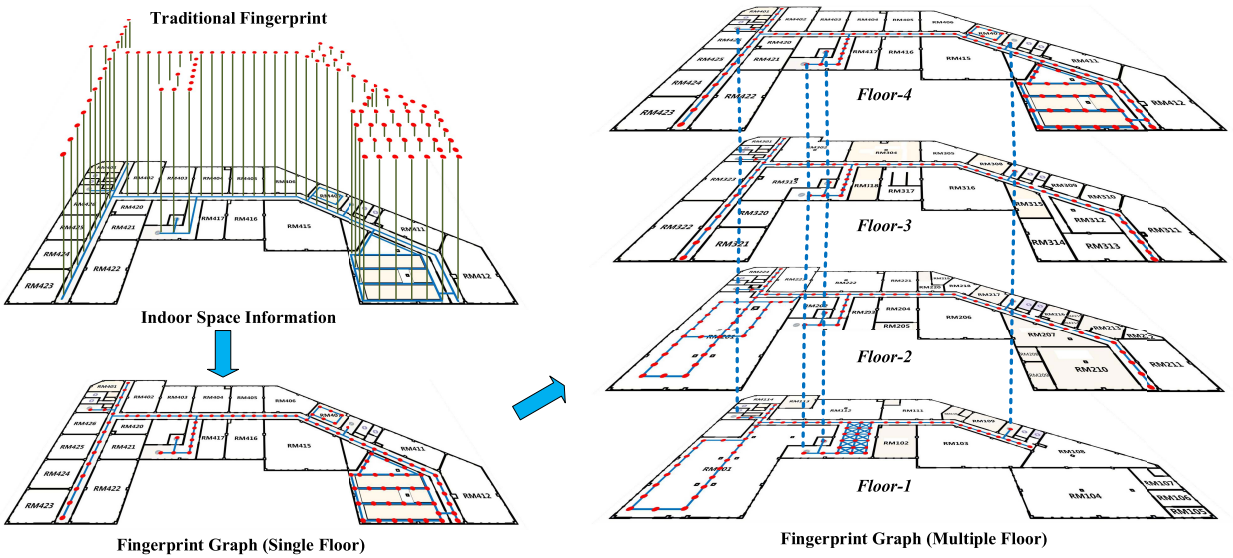

Fig. 2. The process of constructing a fingerprint graph.

matching are modeled as a decoding problem. Namely, given the observation sequence O and a model $\lambda = (A, B, \pi)$, choose a corresponding node sequence Q which best explains the observation sequence O. In the model λ, A is the state transition matrix which describes the transition probability from one node to another, B is the emission matrix which describes the probability of emitting an observation from a node, and π is the initial state distribution. In our method, π is the emission probability of o_1. Therefore, the model λ can be simplified as $\lambda = (A, B)$. In the following, we introduce step by step the construction of a fingerprint graph, calculation of transition probabilities and emission probabilities, and the efficient matching (decoding process). The symbols used in this paper are given in Table I.

TABLE I

LIST OF SYMBOLS AND CORRESPONDING DEFINITIONS

Symbol	Definition
S	Set of hidden states (nodes)
s_i	The i-th node
$rss_{i,j}$	RSS of the node s_i from AP j
$e_{i,j}$	An edge connecting node s_i and node s_j
N	Number of hidden states (nodes)
O	Set of observations
$o_{t,j}$	Measured RSS from AP j at time instant t
T	Number of observations
M	Number of APs
k	Number of selected candidate states
l	Size of candidate list
A	State transition matrix
B	Emission matrix
π	Initial state distribution
Q	Result sequences calculated by the HMM
G	Fingerprint graph
E	Set of edges connecting two nodes
L	Location of a node
I	Indoor semantic information
f_i	Floor level of node s_i
h	Direction of an edge
θ_t	Direction of user at time instant t
δ_θ	Standard deviation of measured orientation
γ_f	Constant transition probability

A. Construction of Fingerprint Graph

The construction of fingerprint graph is shown in Fig. 2. First, a Wi-Fi radio map is constructed through a site survey, and each fingerprint in the radio map contains the RSS measurements from all visible APs and the corresponding location where the RSS measurements are collected. It is observed that indoor space consists of narrow areas (e.g. hallways) and open areas (e.g. lobbies). To enable faster collection of fingerprints, we treat the corridor as one dimensional line and collect the fingerprint along the middle line of the corridor. In the open areas, users can move freely in any direction. The open space is first divided into a grid of cells, and then corresponding fingerprints are collected at the center of each cell.

Based on the constructed radio map, we can construct a fingerprint graph by combining the radio map with a floor plan. Let $G = \langle S, E \rangle$ denote the fingerprint graph, S is the set of nodes (each node corresponds to a fingerprint), and E is the set of edges connecting two nodes. Each node in the fingerprint graph is defined as:

$$s = \langle L, I, RSS, f \rangle \tag{1}$$

where L is the location of the node, I is the indoor semantic information of the node, RSS is the RSS measurements collected at the location L, and f is the floor level. The indoor semantic information includes elevator and staircase, etc. The adjacent nodes are connected by edges under spatial constraints. Each edge is defined as:

$$e = \langle s_i, s_j, h \rangle \tag{2}$$

where s_i and s_j are the two nodes connected by the edge, and h is the direction information of the edge.

B. Transition Probabilities

The transition probability represents the likelihood of a state transiting to another state, which depends on the user's heading

and spatial constraints represented in the fingerprint graph. The user's heading is estimated by combining the compass readings and gyroscope readings [44]. The transition probability of state s_j transiting to state s_i is defined as:

$$p(s_i|s_j,\theta_t,G) = w(s_i,s_j)p(s_i|s_j,\theta_t), \quad i \neq j \tag{3}$$

where $w(s_i,s_j)$ is constrained by the topology of the fingerprint graph. Its value is determined by

$$w(s_i,s_j) = \begin{cases} 0, & e_{i,j} \notin G \\ 1, & e_{i,j} \in G \end{cases} \tag{4}$$

where $w(s_i,s_j)$ equals 1 if the fingerprint graph G contains an edge connecting s_i and s_j; otherwise, $w(s_i,s_j)$ equals 0. $p(s_i|s_j,\theta_t)$ represents the geometry constraint of the fingerprint graph. Its value is calculated by a normal probability distribution as follows:

$$p(s_i|s_j,\theta_t) = \frac{1}{\sqrt{2\pi}\sigma_\theta} exp\Big(-\frac{(\theta_t - h_{i,j})^2}{2{\sigma_\theta}^2}\Big) \tag{5}$$

where $h_{i,j}$ is the direction from v_i to v_j, θ_t is the heading of the user and their range is $0 \sim 2\pi$, and σ_θ is standard deviation of measured orientation.

The state transition between different floors is different from horizontal transition. When a user moves in vertical passages such as staircases, elevators and escalators, the user's heading is not reliable. Therefore, we set the corresponding transition probability to a constant γ_f, namely

$$p(s_i|s_j,\theta_t,G) = \gamma_f, \quad f_i \neq f_j \tag{6}$$

where γ_f is set to 0.1 after empirical analysis. Note that the transition between different floors can be easily detected by using the barometer readings, but the barometer may not be available on many smartphones. Therefore, we set the vertical transition probability to a constant and detect the change of different floors using a sequence of Wi-Fi RSS measurements. This is because single Wi-Fi RSS measurement may not be accurate to detect the floor level, but multiple RSS measurements will be able to determine the floor level via majority voting.

It is also noted that a state can transit to itself because a user may stand at a location for a period of time, such as drinking water, using a vending machine, and talking with others. During that period of time, the heading information of the pedestrian is invalid. We set the self-transition probability to a constant:

$$p(s_i|s_j,\theta_t,G) = \gamma_s, \quad i = j \tag{7}$$

where γ_s is set to 0.1 after empirical analysis.

C. Emission Probabilities

The emission probability is the probability of a state emitting the observation. In reality, this emission probability represents the proximity of a pedestrian and one state at time t. In this study, the emission probability is only related to the signal distance between the RSS measurements in the fingerprint graph and the newly-collected RSS by the smartphone. We use normal distribution probability to describe it [45]. The emission probability is defined as

$$p(o_t|s_i) = \frac{1}{\sqrt{2\pi}\sigma_r} exp\left(-\frac{\sum_{j=1}^{M}(o_{t,j} - rss_{i,j})^2)}{2{\sigma_r}^2}\right) \tag{8}$$

where o_t is the RSS measurement collected by the smartphone at time t and $o_{t,j}$ is the RSS of the j-th AP at this time instant t. Similarly, s_i is the i-th node of the fingerprint graph and $rss_{i,j}$ is the RSS of the j-th AP in the node s_i, M is number of APs in the node s_i, and σ_r is the standard RSS deviation of s_i.

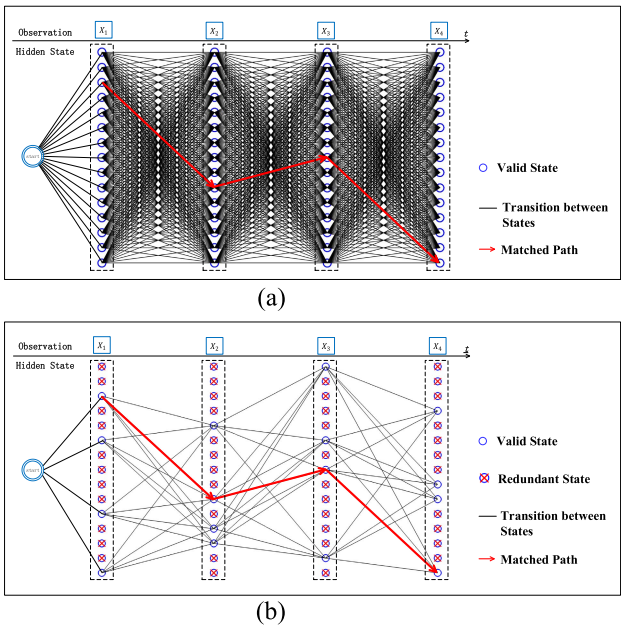

Fig. 3. The computational complexity comparison of HMM-based map matching with heading-aided map matching (our method). (a) Conventional HMM methods need to calculate the probability of all hidden states, including transition probability and emission probability before using the Viterbi algorithm to determine the optimal path (red line). (b) In the heading-aided map matching, the computation is reduced by considering pedestrian orientation and spatial information. The black lines connecting hidden states indicate the calculation of hidden states.

D. Efficient Matching

After calculating the transition probabilities and emission probabilities, we can use the Viterbi decoder to solve the HMM-based map matching problem. However, the computational complexity of conventional Viterbi decoding methods is heavy, which can reach $O(N^2T)$ as shown in Fig. 3(a), where N is the number of hidden states, and T is the number of observations. Since the number of hidden states in the fingerprint graph for a large building is usually large, the computational complexity of conventional HMM-based map matching may prohibit its applications on resource-limited smartphones.

To reduce the computational complexity, we propose an efficient matching method by removing invalid states. Once an observation is received, the candidate states are extracted from the fingerprint graph according to the user's heading and previous valid hidden states. We select the candidate states that have a connection to the previous states from the fingerprint

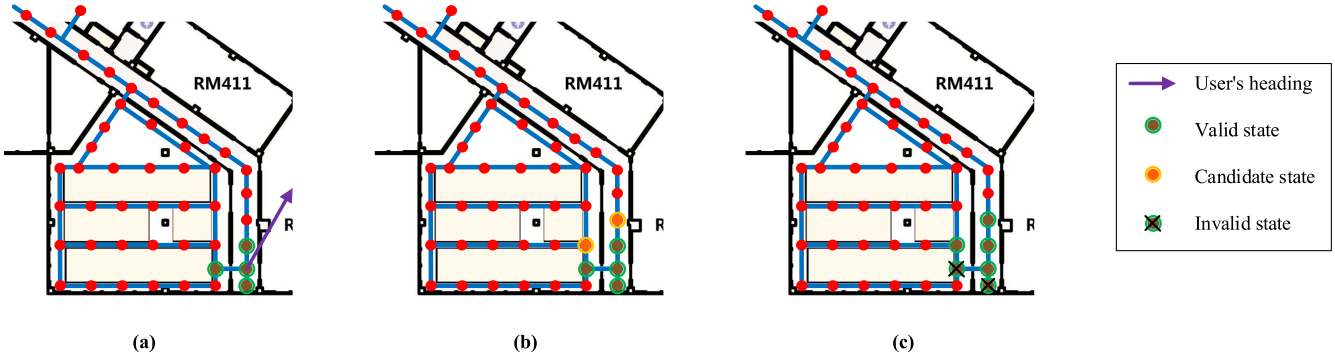

Fig. 4. The selection process of candidate nodes where the number of selected candidate states k equals 4 and the size of candidate list l equals 6.

graph, as shown in Fig. 3(b). In this way, the nodes that have no connection to the previous valid states are removed, and hence the computational complexity is significantly reduced.

Specifically, the selection process of candidate states is illustrated in Fig. 4. In the beginning, initial states are associated with different probabilities according to their RSS similarity with the observation. The higher the RSS similarity, the larger the probability. We select k states with the highest probabilities as shown in Fig. 4(a). When a new observation consisting of Wi-Fi RSS measurements and heading is received, some new candidate states are added into the candidate list according to spatial constraints and the observation (Fig. 4(b)). The probability of all the states in the candidate list is then calculated. Only k states with the highest probability are kept and the other states are considered as invalid and removed as demonstrated in Fig. 4(c). The proposed method shrinks the transition probability matrix from $A_{n\times n}$ into $A_{k\times l}$. The corresponding Viterbi decoding is then written as:

$$C_{1\times l}(t) = (C_{1\times k}(t-1) \times A_{k\times l}) \cdot B_{1\times l}(t) \tag{9}$$

where $C_{1\times l}$ is the scores of the trajectory ending at the l hidden states and $B_{1\times l}(t)$ is the emission probability matrix after the selections of the largest k values.

The computational complexity of the proposed method is $O(N + klT)$, where k and l are constant. Therefore, the computational complexity of our method is actually is $O(N + T)$, which is much lower than that of conventional methods ($O(N^2T)$) [1].

V. Evaluation

To evaluate the proposed method, a series of experiments were conducted in two multi-floor buildings. One building is a four-floor office building with an area of 3300 square meters for each floor and contains elevators, staircases, corridors, and office rooms, etc. There are 338 APs visible during experiments. The number of collected fingerprints for the first floor to the fourth floor is 51, 49, 61, and 99, respectively. The distance interval between two fingerprints has an influence on the number of hidden states that affects the computational consumption and on the density of the radio map. A large interval can reduce the computational time but degrade the localization accuracy. On the contrary, a small interval may achieve better localization accuracy but increase the computational cost. Therefore, we set the distance interval to 3 meters to balance the computational cost and the localization accuracy. The other building is a five-floor museum building with an area of 5000 square meters. The experiments in the museum building were conducted on the second floor and 346 APs were detected, and the number of collected fingerprints is 259.

The proposed method was implemented on the two smartphones (one Samsung Galaxy Note 4 [46] and one Google Nexus 6 [47]. During the experiments, sensor data were collected and recorded with their respective timestamps. The frequency of Wi-Fi scan was set to 1 Hz and the frequency for the digital compass and gyroscope was 15 Hz. There were 18 planned paths in the museum building, and users walked along each path three times. For the office building, there were 2 planned paths, and users walked along each path five times. Therefore, we collected 64 trajectories in total. These paths have various lengths ranging from 48 meters to 378 meters.

To evaluate the accuracy, markers were set along the planned path with a distance interval of 3 meters. Users were required to record the marker's ID and corresponding timestamps to obtain the ground truth location. During the experiments, users are asked to walk along the planned path at a constant speed, and keep the smartphone in hand with the Y axis of the smartphone being parallel with the user's heading. The ground truth location between two markers was derived by using time-based interpolation method [48].

To justify the superiority of the proposed method, we compare our method with the following methods:

- *Weighted K-nearest neighbor* (WKNN) [49], a conventional deterministic method based on the raw trajectories. The number of nearest neighbors is set to 4 after experimental analysis via experiments conducted in both buildings.
- *Horus* [50], is one of the most famous probabilistic framework methods. The Horus system can address the wireless channel variations and achieve high accuracy.
- *Grid-based Bayesian* [45], is a grid-based Bayesian location sensing system. It also falls into the category of probabilistic framework method.
- *VTrack* [9], which uses a HMM to associate GPS trajectories with road segments, we replace the GPS trajectories with WKNN trajectories.

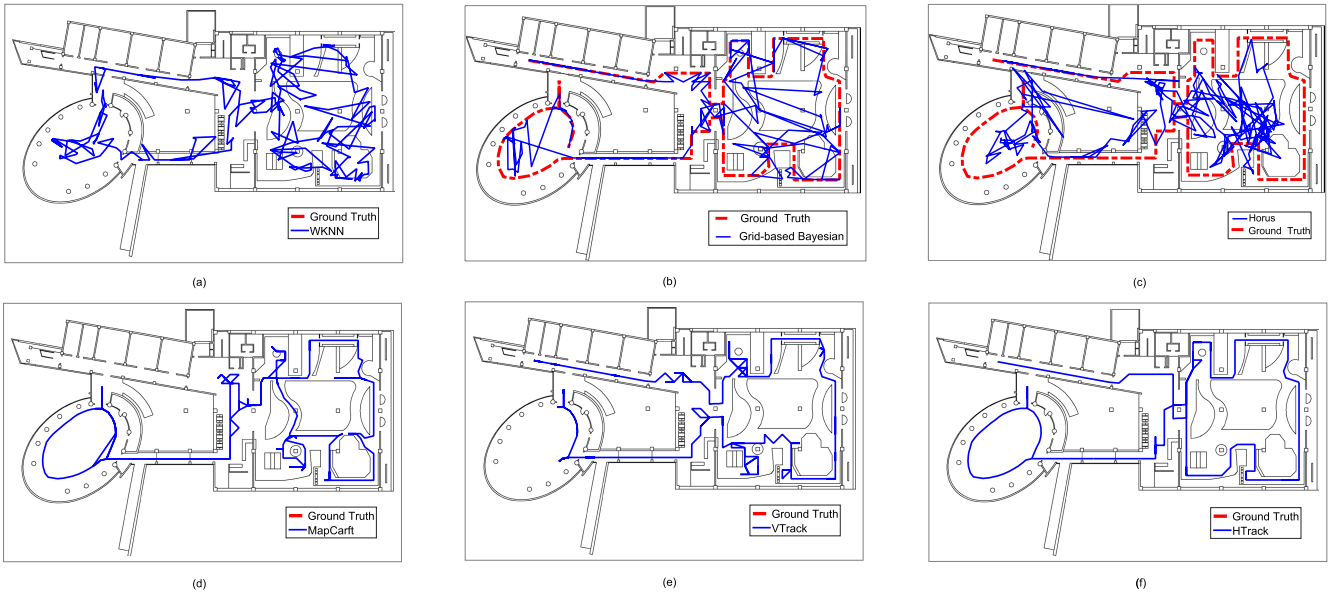

Fig. 5. The planned path and the trajectories computed by different methods in the office building. (a) The planned path (in red) and the trajectory computed by the *WKNN*. (b) The trajectory computed by the *Grid-based Bayesian method*. (c) The trajectory computed by the *Horus*. (d) The trajectory computed by the *MapCraft*. (e) The trajectory computed by the *VTrack*. (f) The trajectory computed by the proposed *HTrack*.

- *MapCraft* [11], which uses CRFs model and combines different feature functions to handle the map matching problem.
- *SkyLoc* [51], a fingerprinting-based indoor floor identifiacation approach. We combine *SkyLoc* with *WKNN* to localize in multi-floor environments.

A. Accuracy Comparison in Single-Floor Environments

We first compare the accuracy of the proposed system with the state-of-the-art systems that were proposed for single-floor environments, including *WKNN*, *VTrack*, and *MapCraft*. Fig. 5 shows the planned path in the museum building and the trajectories derived from different methods. It can be seen that the trajectory obtained from our system, *HTrack*, best matches with the planned path, followed by the *MapCraft* method. The *VTrack* system cannot work properly in the oval-shaped indoor environments. The results from the *WKNN* method have frequent jumps, and cannot provide a smooth trajectory. This is because the *WKNN* does not use any spatial information, while both the *VTrack* and *MapCraft* systems consider spatial information. The proposed *HTrack* uses both spatial information and user's heading, and hence performs the best. It can be also concluded from the experimental results in the office building that the trajectory computed by the proposed system matches best with the planned path, as shown in Fig. 6. Since the structure of the office building is simpler and more regular than the museum building, the estimated trajectories by all methods in the office building have better match with planned path than in the museum building.

We also use the cumulative distribution function (CDF) of localization error to quantitatively evaluate the accuracy of different methods. The X-axis of the CDF figure represents the localization error, and the Y-axis represents the percentage of location estimations whose error is below the value in the X-axis. For example, a data point (1, 0.5) in a CDF figure means that the percentage of location estimations with errors below 1 m is 50%. Fig. 7 and Table II show the localization error of different methods in the museum experiments, from which we can see that the *WKNN* performs the worst and achieves a mean localization error of 4.9m because it does not use any spatial information. Both the *MapCraft* system and *VTrack* system improve the localization error over the *WKNN* by using spatial constraints, achieving a mean localization error of 4.3m and 4.2m, respectively. The performance of the *VTrack* system is slightly better than the *MapCraft* system. The proposed *HTrack* system performs the best and has the lowest localization error (a mean error of 3m). Fig. 8 and Table III show the localization error of different methods computed from the experiments in the office building, in which the *HTrack* system also outperforms other methods, achieving a mean localization error of 3m compared to 4.2m (*MapCraft*), 5m (*VTrack*), and 6.2m (*WKNN*).

B. Accuracy Comparison in Multi-Floor Environments

We also evaluate the proposed method in a multi-floor environment, and compare it with the *SkyLoc* system that was proposed for multi-floor localization. As the *MapCraft* and *VTrack* systems have not considered the change between floors, they are not suitable for comparison in the multi-floor experiments, and hence are not considered in the multi-floor experiment. As shown in Fig. 9, the planned route covers three floors of the office building with corridors, halls, office room, staircases, and elevators, etc. The planned route and the trajectory given by the *SkyLoc* method is shown in Fig. 9(a), from which we can see that the estimated trajectory is very noisy with cross-wall and cross-floor problems. On the contrary, the proposed *HTrack* system, shown in Fig. 9(b), matches with the planned route very well even in the multi-floor environment.

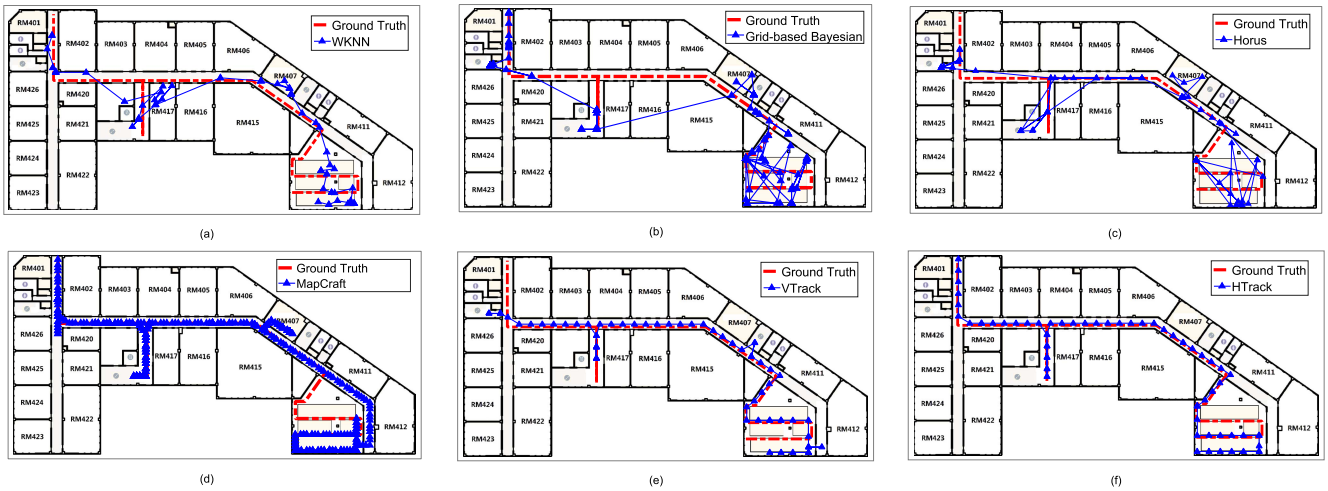

Fig. 6. The planned path and the trajectories computed by different methods in the office building. (a) The planned path (in red) and the trajectory computed by the *WKNN*. (b) The trajectory computed by the *Grid-based Bayesian method*. (c) The trajectory computed by the *Horus*. (d) The trajectory computed by the *MapCraft*. (e) The trajectory computed by the *VTrack*. (f) The trajectory computed by the proposed *HTrack*.

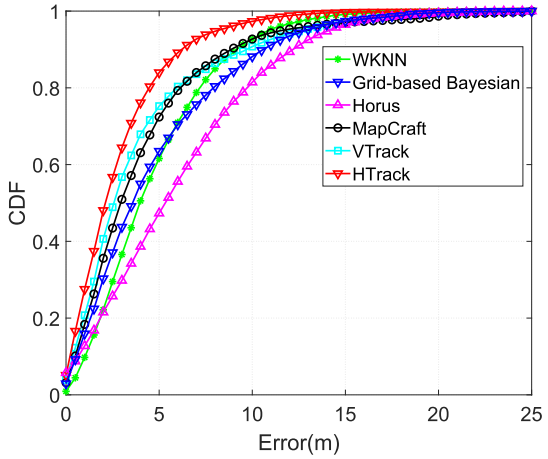

Fig. 7. Localization error of different methods in the museum experiments.

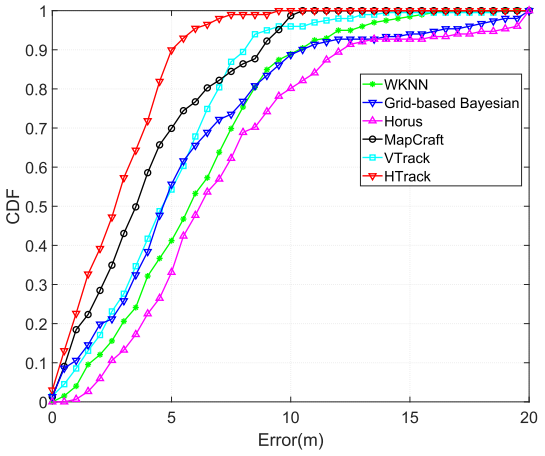

Fig. 8. Localization error of different methods in the office building experiments.

TABLE II
COMPARISON WITH OTHER LOCALIZATION SYSTEMS IN THE MUSEUM EXPERIMENTS

Method	Mean	Median	67%	RMS
WKNN	4.91m	4.20m	6.00m	5.91m
Grid-based Bayesian	4.98m	3.83m	6.47m	5.75m
Horus	6.13m	5.57m	7.64m	7.75m
MapCraft	4.34m	3.19m	4.50m	6.00m
VTrack	4.15m	2.83m	4.00m	5.83m
HTrack	3.06m	2.34m	3.50m	4.11m

Fig. 10 and Table IV show the localization error computed by the *SkyLoc* and the *HTrack*, from which we can see that the *HTrack* performs better than the *SkyLoc*. Specifically, the floor recognition achieved by the *SkyLoc* and the *HTrack* is 69.49% and 93.72%, respectively. The mean localization error is about 6.1m for the *SkyLoc* and 3.4m for the *HTrack*. The main reason why the *HTrack* performs better than the *SkyLoc* is that the *HTrack* uses the topological constraint and semantic information contained in the fingerprint graph.

TABLE III
COMPARISON WITH OTHER LOCALIZATION SYSTEMS IN THE OFFICE EXPERIMENTS

Method	Mean	Median	67%	RMS
WKNN	6.22m	6.03m	7.40m	6m
Grid-based Bayesian	5.98m	4.84m	7.71m	6.35m
Horus	7.19m	6.06m	8.76m	7.95m
MapCraft	4.18m	3.77m	4.80m	5.05m
VTrack	5.03m	4.90m	6.20m	5.79m
HTrack	3.01m	2.99m	4.15m	3.56m

C. Parameter Analysis

We then analyze the effect of the number of candidate states on the localization error via an experiment conducted in the museum building. The candidate states are the k hidden states with the highest probability after each transition and emission calculation, and essentially reflect the possible range of pedestrian movement. The larger the value of k, the larger the area that HTrack uses to analyze. The median, mean, 67%, and root mean square (rms) error metrics are used to evaluate the effect. As shown in Fig. 11, the localization error

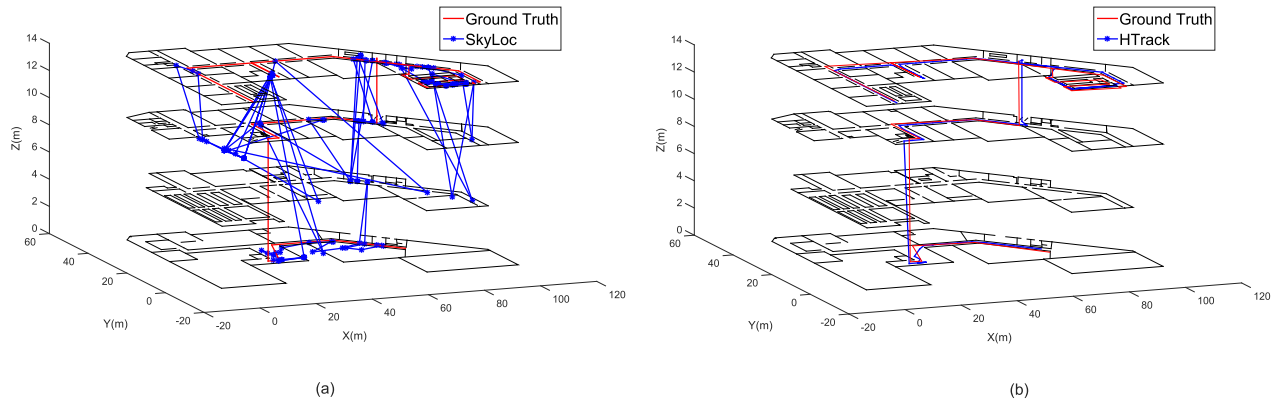

Fig. 9. The planned path and the trajectories computed by our method and the SkyLoc method in the multi-floor environment.

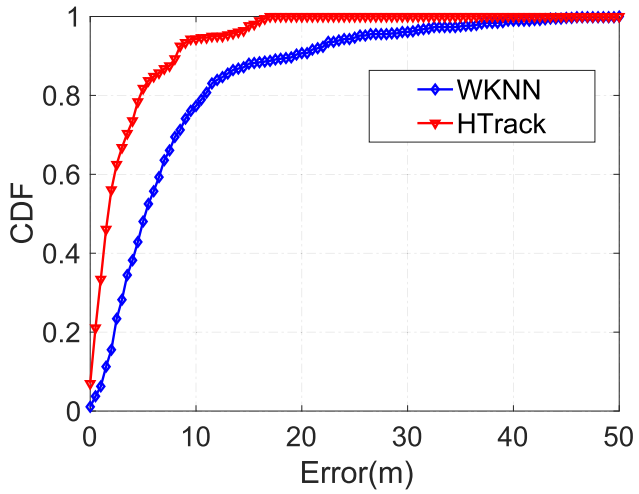

Fig. 10. Localization error of different methods in multi-floor experiments.

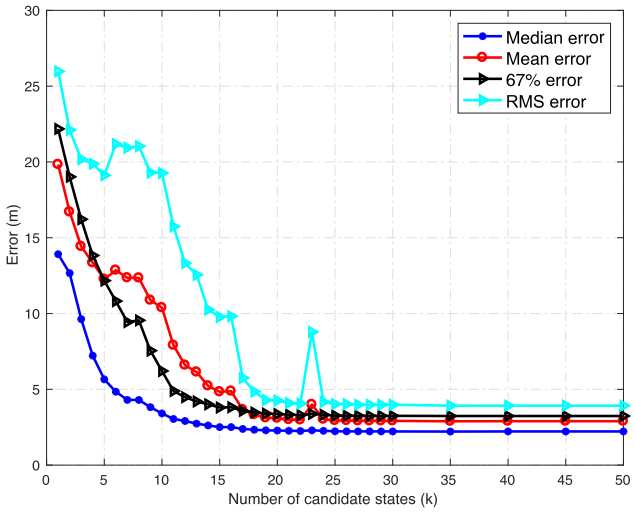

Fig. 11. The effect of the number of candidate states on the localization error.

TABLE IV
LOCALIZATION ACCURACY COMPARISON IN MULTI-FLOOR ENVIRONMENT

Name	SkyLoc	HTrack
Floor Recognition	69.49%	93.72%
Mean	6.10m	3.37m
Median	4.92m	1.98m
67%	8.00m	3.50m
RMS	11.81m	4.93m

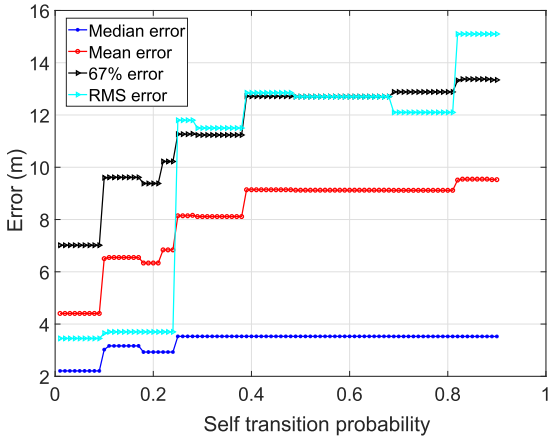

Fig. 12. The effect of self transition probability on the localization error.

decreases as the number of candidate states increases. In the beginning, the localization error dramatically drops down with the increase of the number of candidate states until the number of candidate states rises to 25, where further increasing the candidate states does not reduce the localization error. This may because increasing the number of candidate states will expand the searching area of finding the optimal path. When the number of candidate states reaches a certain value, the localization error will no longer increase as the corresponding searching area has covered all the possible transition states.

In addition, We analyze the effect of the self transition probability and the cross-floor transition probability of hidden states on the proposed method. We increase the two transition probabilities from 0.01 to 0.9 with 0.01 as step size. The experimental results are shown in Fig. 12 and Fig. 13, respectively. When the self transition probability is less than or equal to 0.1, the proposed method performs the best. While the change of the cross-floor transition probability has no effect on the localization error. Therefore we set the value of two probabilities to 0.1.

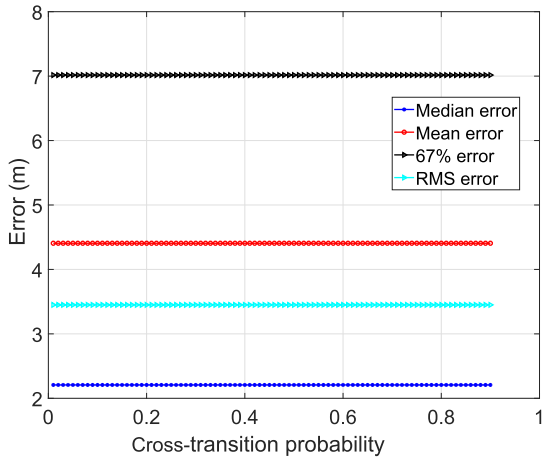

Fig. 13. The effect of cross-floor transition probability on the localization error.

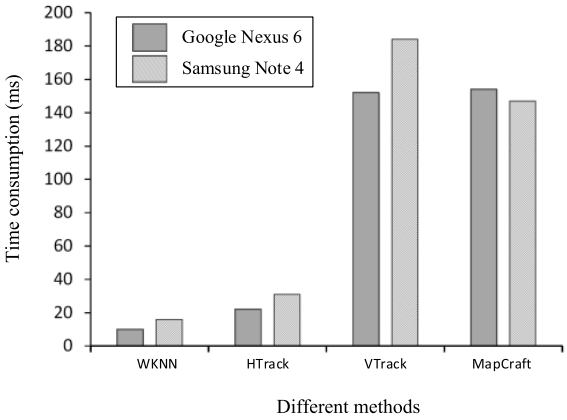

Fig. 14. Computational cost of different systems in the museum experiments.

D. Computational Cost

We compare the computational cost of the proposed system *HTrack* with that of WKNN, *VTrack* and *MapCraft*. The energy consumption calculation method in [52] is used.

Fig. 14 shows the time consumed by different methods, and Fig. 15 shows the corresponding energy consumption. The proposed *HTrack* system consumes a bit more time than the WKNN method, but about five times less time than the *VTrack* and *MapCraft* systems. The main reason why the WKNN performs the fastest is that the WKNN does not use any spatial information, and the other three methods including our *HTrack* consider using spatial constraints to improve the localization accuracy. Since the proposed *HTrack* considers the heading information and uses a fingerprint graph, it consumes much less time than the *VTrack* and *MapCraft* systems. For the energy consumption, a similar pattern is witnessed. Specifically, the energy consumed by the *HTrack* almost doubles that of the WKNN, but is five times less than the *VTrack* and four times less than the *MapCraft*. It is interesting to see that there is a big difference in both the consumed time and energy between two smartphones (Google Nexus 6 and Samsung Note 4). This may because the two phones have different hardware specifications (e.g, CPU, memory), and the background processes running on each phone might be different.

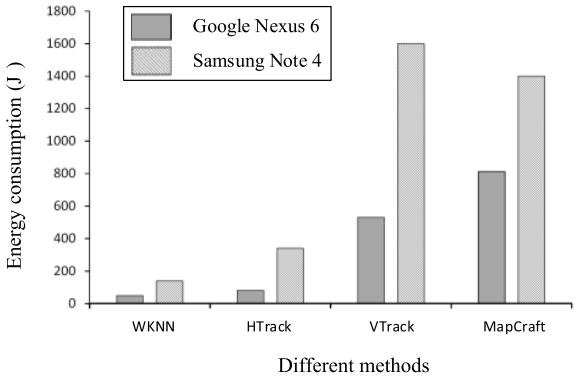

Fig. 15. Computational cost of different systems in the museum experiments.

VI. Conclusion and Future Work

In this paper, we present an efficient heading-aided map matching method and its prototype system. By considering the user's heading information and using a fingerprint graph, we reduce the computational cost of matching significantly. Experimental results show that the proposed method obtains a better localization accuracy and consumes less computational cost than the state-of-the-art methods. In the future, we will consider the effect of different device poses on the localization performance of the proposed method. It is also an interesting topic to design a more efficient method to construct the fingerprint graph.

References

[1] J. Xiao, Z. Zhou, Y. Yi, and L. M. Ni, "A survey on wireless indoor localization from the device perspective," *ACM Comput. Surv.*, vol. 49, no. 2, 2016, Art. no. 25.

[2] J. Shang, X. Hu, F. Gu, D. Wang, and S. Yu, "Improvement schemes for indoor mobile location estimation: A survey," *Math. Problems Eng.*, vol. 2015, Mar. 2015, Art. no. 397298. [Online]. Available: https://www.hindawi.com/journals/mpe/2015/397298/cta/

[3] P. Davidson and R. Piché, "A survey of selected indoor positioning methods for smartphones," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 1347–1370, 2nd Quart., 2017.

[4] Z. Yang, C. Wu, Z. Zhou, X. Zhang, X. Wang, and Y. Liu, "Mobility increases localizability: A survey on wireless indoor localization using inertial sensors," *ACM Comput. Surv.*, vol. 47, no. 3, p. 54, Apr. 2015.

[5] Q. Fan, B. Sun, Y. Sun, and X. Zhuang, "Performance enhancement of MEMS-based INS/UWB integration for indoor navigation applications," *IEEE Sensors J.*, vol. 17, no. 10, pp. 3116–3130, May 2017.

[6] K. Chen, C. Wang, Z. Yin, H. Jiang, and G. Tan, "Slide: Towards fast and accurate mobile fingerprinting for Wi-Fi indoor positioning systems," *IEEE Sensors J.*, vol. 18, no. 3, pp. 1213–1223, Feb. 2018.

[7] Y. Shu, C. Bo, G. Shen, C. Zhao, L. Li, and F. Zhao, "Magicol: Indoor localization using pervasive magnetic field and opportunistic WiFi sensing," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 7, pp. 1443–1457, Jul. 2015.

[8] A. Catovic and Z. Sahinoglu, "The Cramer–Rao bounds of hybrid TOA/RSS and TDOA/RSS location estimation schemes," *IEEE Commun. Lett.*, vol. 8, no. 10, pp. 626–628, Oct. 2004.

[9] A. Thiagarajan *et al.*, "VTrack: Accurate, energy-aware road traffic delay estimation using mobile phones," in *Proc. 7th ACM Conf. Embedded Netw. Sensor Syst.*, 2009, pp. 85–98.

[10] A. Thiagarajan, L. Ravindranath, H. Balakrishnan, S. Madden, and L. Girod, "Accurate, low-energy trajectory mapping for mobile devices," in *Proc. 8th USENIX Conf. Netw. Syst. Design Implement. (NSDI)*, 2011, pp. 267–280.

[11] Z. Xiao, H. Wen, A. Markham, and N. Trigoni, "Indoor tracking using undirected graphical models," *IEEE Trans. Mobile Comput.*, vol. 14, no. 11, pp. 2286–2301, Nov. 2015.
[12] Z. Farid, R. Nordin, and M. Ismail, "Recent advances in wireless indoor localization techniques and system," *J. Comput. Netw. Commun.*, vol. 2013, Aug. 2013, Art. no. 185138. [Online]. Available: https://www.hindawi.com/journals/jcnc/2013/185138/cta/
[13] S. He and S.-H. G. Chan, "Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 466–490, 1st Quart., 2015.
[14] C. Wu, Z. Yang, Y. Liu, and W. Xi, "WILL: Wireless indoor localization without site survey," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 4, pp. 839–848, Apr. 2013.
[15] K. Chintalapudi, A. P. Iyer, and V. N. Padmanabhan, "Indoor localization without the pain," in *Proc. 16th Annu. Int. Conf. Mobile Comput. Netw.*, 2010, pp. 173–184.
[16] A. Saeed, A. E. Kosba, and M. Youssef, "ICHNAEA: A low-overhead robust WLAN device-free passive localization system," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 1, pp. 5–15, Feb. 2014.
[17] X. Liu, Y. Zhan, and J. Cen, "An energy-efficient crowd-sourcing-based indoor automatic localization system," *IEEE Sensors J.*, vol. 18, no. 14, pp. 6009–6022, Jul. 2018.
[18] B. Ferris, D. Fox, and N. Lawrence, "WiFi-SLAM using Gaussian process latent variable models," in *Proc. 20th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2007, pp. 2480–2485.
[19] W. Ruan, L. Yao, Q. Z. Sheng, N. J. G. Falkner, and X. Li, "Tagtrack: Device-free localization and tracking using passive RFID tags," in *Proc. 11th Int. Conf. Mobile Ubiquitous Syst., Comput., Netw. Services (MOBIQUITOUS)*. Brussels, Belgium: ICST, 2014, pp. 80–89, doi: 10.4108/icst.mobiquitous.2014.258004.
[20] B. Großwindhager *et al.*, "UWB-based single-anchor low-cost indoor localization system," in *Proc. 15th ACM Conf. Embedded Netw. Sensor Syst.*, 2017, p. 34.
[21] R. Harle, "A survey of indoor inertial positioning systems for pedestrians," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1281–1293, 3rd Quart., 2013.
[22] A. R. Akhmareh, M. T. Lazarescu, O. B. Tariq, and L. Lavagno, "A tagless indoor localization system based on capacitive sensing technology," *Sensors*, vol. 16, no. 9, p. 1448, 2016.
[23] R. C. Luo and O. Chen, "Wireless and pyroelectric sensory fusion system for indoor human/robot localization and monitoring," *IEEE/ASME Trans. Mechatronics*, vol. 18, no. 3, pp. 845–853, Jun. 2013.
[24] M. Ramezani, D. Acharya, F. Gu, and K. Khoshelham, "Indoor positioning by visual-inertial odometry," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. IV-2/W4, pp. 371–376, Sep. 2017.
[25] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: Unsupervised indoor localization," in *Proc. 10th Int. Conf. Mobile Syst., Appl., Services*, 2012, pp. 197–210.
[26] H. Abdelnasser *et al.*, "SemanticSLAM: Using environment landmarks for unsupervised indoor localization," *IEEE Trans. Mobile Comput.*, vol. 15, no. 7, pp. 1770–1782, Jul. 2016.
[27] F. Gu, K. Khoshelham, J. Shang, and F. Yu, "Sensory landmarks for indoor localization," in *Proc. IEEE 4th Int. Conf. Ubiquitous Positioning, Indoor Navigat. Location Services (UPINLBS)*, Nov. 2016, pp. 201–206.
[28] B. Zhou, Q. Li, Q. Mao, W. Tu, and X. Zhang, "Activity sequence-based indoor pedestrian localization using smartphones," *IEEE Trans. Human-Mach. Syst.*, vol. 45, no. 5, pp. 562–574, Oct. 2015.
[29] J. Shang, F. Gu, X. Hu, and A. Kealy, "APFiLoc: An infrastructure-free indoor localization method fusing smartphone inertial sensors, landmarks and map information," *Sensors*, vol. 15, no. 10, pp. 27251–27272, 2015.
[30] Z. Chen, H. Zou, H. Jiang, Q. Zhu, Y. C. Soh, and L. Xie, "Fusion of WiFi, smartphone sensors and landmarks using the Kalman filter for indoor localization," *Sensors*, vol. 15, no. 1, pp. 715–732, Jan. 2015.
[31] J. Hightower and G. Borriello, "Particle filters for location estimation in ubiquitous computing: A case study," in *Proc. UbiComp, Ubiquitous Comput.*, N. Davies, E. D. Mynatt, and I. Siio, Eds. Berlin, Germany: Springer, 2004, pp. 88–106.
[32] O. Woodman and R. Harle, "Pedestrian localisation for indoor environments," in *Proc. 10th Int. Conf. Ubiquitous Comput.*, 2008, pp. 114–123.
[33] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 293–304.
[34] C. Yu, H. Lan, F. Gu, F. Yu, and N. El-Sheimy, "A map/INS/Wi-Fi integrated system for indoor location-based service applications," *Sensors*, vol. 17, no. 6, p. 1272, 2017.
[35] S. Hilsenbeck, D. Bobkov, G. Schroth, R. Huitl, and E. Steinbach, "Graph-based data fusion of pedometer and WiFi measurements for mobile indoor positioning," in *Proc. ACM Int. Joint Conf. Pervas. Ubiquitous Comput.*, 2014, pp. 147–158.
[36] L. Liao, D. Fox, J. Hightower, H. Kautz, and D. Schulz, "Voronoi tracking: Location estimation using sparse and noisy sensor data," in *Proc. IROS*, 2003, pp. 723–728.
[37] L. Díaz-Vilariño, K. Khoshelham, J. Martínez-Sánchez, and P. Arias, "3D modeling of building indoor spaces and closed doors from imagery and point clouds," *Sensors*, vol. 15, no. 2, pp. 3491–3512, 2015.
[38] X. Hu, H. Fan, A. Zipf, J. Shang, and F. Gu, "A conceptual framework for indoor mapping by using grammars," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. IV-2/W4, p. 335, Sep. 2017.
[39] C. E. White, D. Bernstein, and A. L. Kornhauser, "Some map matching algorithms for personal navigation assistants," *Transp. Res. C, Emerg. Technol.*, vol. 8, nos. 1–6, pp. 91–108, 2000.
[40] K.-C. Lan and W.-Y. Shih, "On calibrating the sensor errors of a PDR-based indoor localization system," *Sensors*, vol. 13, no. 4, pp. 4781–4810, 2013.
[41] J. Seitz, J. Jahn, J. G. Boronat, T. Vaupel, S. Meyer, and J. Thielecke, "A hidden Markov model for urban navigation based on fingerprinting and pedestrian dead reckoning," in *Proc. IEEE 13th Conf. Inf. Fusion (FUSION)*, Jul. 2010, pp. 1–8.
[42] X. Niu, M. Li, X. Cui, J. Liu, S. Liu, and K. R. Chowdhury, "WTrack: HMM-based walk pattern recognition and indoor pedestrian tracking using phone inertial sensors," *Pers. Ubiquitous Comput.*, vol. 18, no. 8, pp. 1901–1915, 2014.
[43] H. Wu, S. He, and S. H. G. Chan, "A graphical model approach for efficient geomagnetism-pedometer indoor localization," in *Proc. IEEE Int. Conf. Mobile Ad Hoc Sensor Syst.*, Oct. 2017, pp. 371–379.
[44] W. Kang, S. Nam, Y. Han, and S. Lee, "Improved heading estimation for smartphone-based indoor positioning systems," in *Proc. IEEE Int. Symp. Pers. Indoor Mobile Radio Commun.*, Sep. 2012, pp. 2449–2453.
[45] T. Roos, P. Myllymäki, H. Tirri, P. Misikangas, and J. Sievänen, "A probabilistic approach to wlan user location estimation," *Int. J. Wireless Inf. Netw.*, vol. 9, no. 3, pp. 155–164, 2002.
[46] Google. *Google Nexus 6*. Accessed: Aug. 10, 2018. [Online]. Available: https://www.gsmarena.com/motorolanexus6-6604.php
[47] Samsung. *Samsung Note 4*. Accessed: Aug. 10, 2018. [Online]. Available: https://www.gsmarena.com/samsunggalaxynote4-6434.php
[48] T. Toftkjær and M. B. Kjærgaard, "The impact of sensor errors and building structures on particle filter-based inertial positioning," *Pervas. Mobile Comput.*, vol. 8, no. 5, pp. 764–776, 2012.
[49] C. Rizos, A. G. Dempster, B. Li, and J. Salter, "Indoor positioning techniques based on wireless LAN," in *Proc. 1st IEEE Int. Conf. Wireless Broadband Ultra Wideband Commun.*, 2007, pp. 1–7.
[50] M. Youssef and A. Agrawala, "The *Horus* location determination system," *Wireless Netw.*, vol. 14, no. 3, pp. 357–374, 2008.
[51] A. Varshavsky, A. LaMarca, J. Hightower, and E. de Lara, "The SkyLoc floor localization system," in *Proc. 5th Annu. IEEE Int. Conf. Pervas. Comput. Commun. (PerCom)*, Mar. 2007, pp. 125–134.
[52] L. Zhang *et al.*, "Accurate online power estimation and automatic battery behavior based power model generation for smartphones," in *Proc. 8th IEEE/ACM/IFIP Int. Conf. Hardw./Softw. Codesign Syst. Synth.*, Oct. 2010, pp. 105–114.

Yongfeng Wu, photograph and biography not available at the time of publication.

Pan Chen, photograph and biography not available at the time of publication.

Fuqiang Gu, photograph and biography not available at the time of publication.

Xiaoping Zheng, photograph and biography not available at the time of publication.

Jianga Shang, photograph and biography not available at the time of publication.