IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received September 6, 2018, accepted October 23, 2018, date of publication November 2, 2018, date of current version December 3, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2879311

BTrack: Using Barometer for Energy Efficient Location Tracking on Mountain Roads

HAIBO YE[1,2], WENHUA YANG[1], YUNYU YAO[1], TAO GU[3], (Senior Member, IEEE), AND ZHIQIU HUANG[1]

[1]College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China
[2]The State Key Laboratory of Novel Software Technology, Nanjing 210023, China
[3]School of Computer Science and IT, RMIT University, Melbourne, VIC 3001, Australia

Corresponding author: Haibo Ye (yhb@nuaa.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61702261, in part by the China Postdoctoral Science Foundation under Grant 2017M621742, and in part by the Foundation of the State Key Laboratory of Novel Software Technology under Grant KFKT2017B15.

ABSTRACT This paper presents the design and implementation of BTrack, a new approach for location tracking on mountain roads. It is as accurate as the GPS on common smartphones, but costs much less energy. To design such a location system, we carried out a nation-wide online survey to confirm the desirability for low power location tracking system, and the result is positive. Then, we proposed BTrack, and it makes two key technical contributions. The first is to propose a dynamic transition matrix hidden Markov model to combine the barometer and accelerometer reading hints for estimating the location of the user. The second is to design some novel techniques for parameter estimation, and proposed an adaptive algorithm to reduce the computational complexity. The field studies show that the accuracy of BTrack is no worse than GPS in 56% cases, meanwhile, the energy consumption is only about 30%. Compared with the existing works, BTrack is more suitable for location tracking on mountain roads.

INDEX TERMS Location tracking, barometer, smartphone sensors.

I. INTRODUCTION

This paper describes the design and implementation of BTrack, a novel energy efficient system to locate and track a moving user with a smartphone on mountain roads. Imagine that when you are visiting a mountain area tourist attraction and want to find a best path which can cover more famous spots, a navigation service is what you need. However, the most common situation is, you open the map application to find your location and then close it for energy saving. The GPS module which provides the location service here is one of the most power-hungry components on smartphones, especially for continuous tracking, and it may deplete batteries within hours. Therefore, a better outdoor tracking system is needed.

We believe a tracking system that tracks mobile users on the mountain roads should ideally satisfy the following three important requirements. The first is generality, which means that it should be easily deployed on existing common mobile devices such as the smartphone, and should not require any hardware or firmware changes of the devices. Meanwhile, it should not require extra infrastructure deployment. The infrastructure deployment is often difficult in the mountain areas. The second is accuracy, which means that it should be accurate enough to satisfy users' need. The GPS accuracy can be used as a reference because of its wide usage for outdoor localization, and thus we believe that a tracking system can serve most location based services if the accuracy is no worse than the GPS. The last is energy efficiency, which means that the energy consumption should be low enough since the battery of the mobile device is limited, especially when the device is used for a continuous tracking for a long time. It's often difficult to get a chance to recharge the battery in the outdoors.

To design a location system that best fulfill these requirements, we first take GPS into account. The GPS has good generality, but the high energy consumption for continuous locating is also well known [13], [14]. Recent technologies use technologies to minimize the GPS sampling rate for energy saving [29]. It seems to have good energy efficiency and accuracy. However, the real energy saving is limited, because there is extra energy consumption when the GPS model is initializing and powering off [9].

2169-3536 © 2018 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Furthermore, the accuracy cannot be ensured especially on mountain roads. For example, work [16] propose to estimate user moving distance by the number of user steps, but they assume that the step size is a contestant value, which is unreasonable since the step size is obviously affected by the situation of mountain roads. Other techniques based on wireless fingerprint [5], [26] or video [17] can be accurate and energy efficient but not general. They require to deploy extra infrastructure in the environment for localization such as Wi-Fi access points, Bluetooth beacons and cameras.

In the mountain area, there often exists a clear altitude difference at different parts of the road. Based on this observation, we argue that the altitude change can be used as a hint of location change. With the help of a road map, we can track users' locations on the road based on their altitude change. The altitude change [11], [24] can be measured by the barometer sensor on the smartphone, and it is energy efficient. However, it is not trivial to realize a tracking system that satisfy all the three requirements mentioned above for the following challenges: **a) the barometer reading is affected by the weather, so it not only changes when the user altitude changes but also changes when the weather changes.** If we directly map the barometer reading change to altitude change, the noise can be tens of meters, and in this way, the accuracy requirement cannot be satisfied. **b) the road map can be complicated, and the altitude change information is often not enough for localization.** For example, in some cases, even with the same altitude change, the possible location of the user can be not unique. Some extra information is needed to assist localization, and should also keep the generality and energy efficiency of the system. Which information to choose and how it is used for localization are challenging.

In this paper, we propose BTrack faced these challenges, to satisfy all the three above-mentioned requirements. BTrack only makes use of the low power barometer and accelerometer senors on smartphone, which is very energy efficient. It dose not need any extra infrastructure or device change, which is very practical for real usage. The accuracy of BTrack is no worse than GPS, sometimes even better than GPS. To our best knowledge, this paper is the first work addressing these requirements for tracking a user on mountain roads. In summary, we make following contributions:

1. A dynamic transition matrix Hidden Markov Model is proposed to effectively combine the barometer and accelerometer reading hints for more accurately estimating the location of the user.
2. We designed three novel techniques for parameter estimation of the model, and proposed an adaptive transition matrix calculation algorithm to reduce the computational complexity.
3. We carried out a nation-wide online survey to confirm the desirability for BTrack, and we conducted extensive field studies to analyze the performance of BTrack. The field study shows that BTrack performs no worse than GPS in 56% cases, meanwhile, the energy consumption is only about 30%.

In the rest of this paper, we first confirm the motivation in Section II by an online survey. Later we give the problem definition and propose the model in Section III. We solve the model in Section IV. Section V shows the evaluation results. Section VI discusses the related work, and finally, Section VII concludes the paper.

II. MOTIVATION

The idea of BTrack originated from the authors' bad experience in a famous tourist resort in a mountain area. We need a mobile application which can play as a tourist guide, and the guide can locate our locations and help us to find the way. We did find some applications but all with bad user experience. The power consumption is extremely high due to the continuous GPS sampling, and the location accuracy is also poor in the non-flat mountain areas. Since the users are the direct beneficiaries, to motivate a new locating and tracking technology, the authors also needed to know the views of the users. There are three main questions to be answered: Do you need a tool for locating and tracking on non-flat areas? Do you feel current technologies and applications perfectly satisfy your requirement? What is your requirement about the power consumption of this tool running on your mobile device? These questions were meant to establish the desirability of BTrack from the users, before taking steps to implement such a tool.

To answer these questions, we carried out a nation-wide online survey in China using BaiduMTC, this web site is specialized for survey and testing of new mobile technologies and applications. The less reliable responses were filtered out if there exist conflicting choices from the same user. We stopped the survey until 800 valid responses were received. Participants were paid for completing the survey, and the reward is consistent with prevailing compensation rates on mtc.baidu.com. The survey engine had mechanisms to prevent repeated entries by the same user or robot entries. Survey respondents covered 51 cities in 17 different provinces in China, and among them 56% were male and 44% female, ranging from 14 to 55 years old (mean 30.4). In the survey, we first ask the users to choose a characteristic which matches himself most. Based on which, the participants are firstly labeled as exerciser (40%), traveler (38%) or others (22%).

Do you need a tool for locating and tracking on non-flat areas? Users agreed that locating and tracking is a basic requirement and have been widely used in many applications today. The exercisers and travelers need the tool more compared to others, 89% of the exercisers and 81% of the travelers said they often use tools for locating and navigating, this is an important tool and is very useful when doing sports or traveling outside. It helps to record useful data such as running speed and trajectory when doing sport exercise. It can also help to find the way and guide the tour when traveling outside. They believed that tracking their location information is useful and can be used in many applications. For other users, 76% said the most commonly used tools are

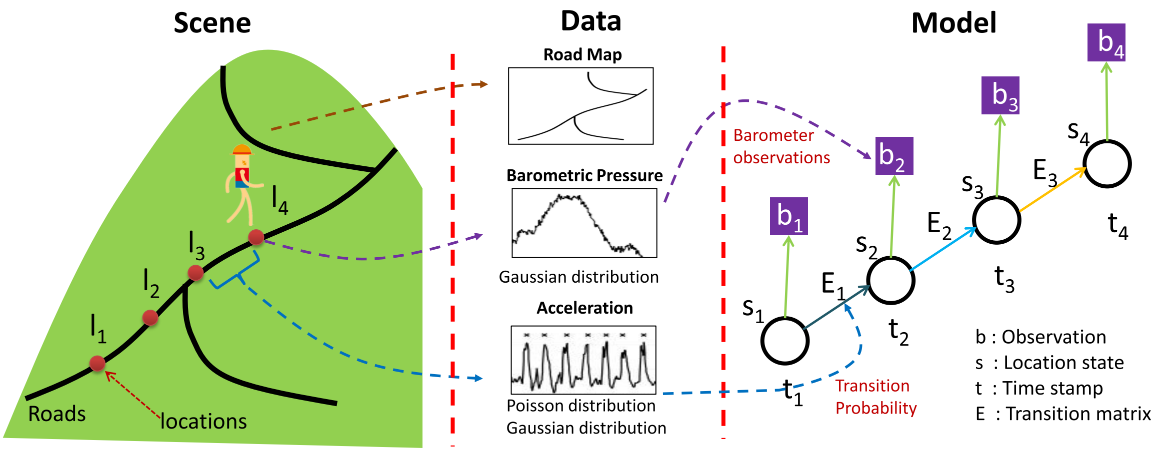


FIGURE 1. Overview of BTrack.

the applications such as google maps, they often use it for car and pedestrian navigation.

Do you feel current technologies and applications well satisfy your requirement? When asked whether their requirements are perfectly satisfied, 77% exercisers said current tools are not well satisfied, and the most unsatisfactory aspect is the energy consumption, the power consumption is high when do continuous locating. The second unsatisfactory aspect is the accuracy, they pointed out that the accuracy is often low in the center city and mountain areas. The results were similar in travelers and other users. In all respondents, 92% of them agreed they need some more energy efficient and accurate technologies for locating and tracking.

What is your requirement about the power consumption of this tool running on your mobile device? We found that users care about the power consumption of every application running on their mobile devices, if the power consumption is too high, it will cause a great influence on its usability. According to our survey, under normal conditions, 81% participants choose other methods to find the way instead of using a navigation tool if the expected hourly power consumption is more than 10% of the device battery, while the common energy performance of today's navigation applications based on GPS is about 10% to 20% per hour. Especially for those travelers visiting unfamiliar areas, they often need continuous navigation service to assistant the tour, and meanwhile very sensitive to energy consumption, its a very tricky thing when the phone battery is empty.

Taking survey responses as ordinal values, we computed the correlations between these responses from different kinds of users. Statistically significant positive correlations were found between interest in BTrack and each of the (i) being an outdoor exerciser, (ii) be reliant on the smartphone, and (iii) being a frequent traveler. This means that individuals with more exercising and traveling are precisely those who need BTrack more.

In summary, we obtained three key observations from the online survey. First, locating and tracking of the smartphone is a basic requirement for many mobile applications. Second, user requirement is not well satisfied by current technologies and applications, especially the power consumption and accuracy. Finally, if the power consumption is less than 10% per hour, the users are very likely to have this application run in their smartphones. The above results confirmed users's desirability for better locating and tracking technology, and further motivated us to propose a new solution to fulfill users' requirements. Next, we describe design, implementation, and actual deployment-based evaluation of BTrack.

III. PROBLEM DEFINITION AND MODELING

In this section, we first give the formal problem definition, including some basic definitions and the framework of our model. Next we give the detail of our DTM_HMM model.

A. PROBLEM DEFINITION

Definition 1 (An Observation of a Barometer Reading b_t): We represent a barometer reading observation of the moving user at time t as such a tuple $b_t = (t, bar)$, where t denotes the time and bar represents the barometer readings value.

Definition 2 (An observation of the Acceleration Readings $a_{t-1,t}$): We represent a acceleration readings observation of the moving user from time $t-1$ to t as such a tuple $a_{t-1,t} = (t-1, t, acc)$, where $t-1$ and t denotes the time and acc represents the acceleration readings sequence within that time period.

Definition 3 (A Road Map) M: A road map is a set of connected roads which is represented by $M = \{r_1, r_2, ...\}$, where r denotes a road of the road map.

Definition 4 (A Location) l_t: The location of a moving user at time t is represented by $l_t = (t, loc)$, where t denotes the time and loc is the location on the road map.

User Tracking: User tracking refers to the process of continually knowing the location of the user on the road map. The tracking trace of the user is $Trace = \{l_1, l_2, ...l_t, ...\}$, were l_t is the location of the user at time t.

Fig. 1 is the framework of our approach. The left side shows the scene, the purpose is to track a moving user on the mountain road. Assume the road map is already known.

We collect two types of data from smartphone, the barometer readings and the acceleration readings. We use the barometer readings to estimate the altitude change of the user, and assume the error follows the Gaussian distribution. Similarly, we extract the displacement of the user using acceleration readings, and assume different probability density function according to the road type.

The right side of Fig. 1 shows the model. We model the spatiotemporal conditional dependencies of user moving trace using a probabilistic model called Hidden Markov Model. A HMM models a system which are assumed to be a Markov process, and the model contains a sequence of unobserved states. The process evolving over time in this model are the changing locations of the moving user on the road map. The locations of the user cannot be directly observed, and we consider them as *hidden*. The observed barometer reading of the user is conditioned on the hidden state (location). In addition, the displacement of the user in a time interval specifies the local dependencies between the next and previous hidden states (locations). Based on the overview in Fig. 1, we formally define the problem as follows.

Problem Definition: For a moving user, given an road map M in T time stamps, the barometer reading observations $\{b_t\}_{t=1:T}$, and the acceleration readings $\{a_{t_i,t_{i+1}}\}_{i=1:T}$ on M, the goal is to accurately and efficiently infer the hidden location states $\{l_t\}_{t=1:T}$ of the user in each time stamp.

B. DTM_HMM: DYNAMIC TRANSITION MATRIX HIDDEN MARKOV MODEL

We first show some notations and meanings in Table 1. For example, L denotes all the location states, and L_t means the location state at time stamp t. We need to estimate three group of variables in our model before using it to find out the locations. The initial location state probability matrix Π, the location transition probability matrix E, and the probability distribution P of the barometer reading observation. In detail, each element π^s in Π denotes the initial probability that the user begins in location state s. Each element $E_{t-1,t}$ represents the transition probability matrix for location states on the road map from time stamp $t-1$ to t, it is a matrix of size $S * S$. For the observation distribution parameters P, it contains the Gaussian distribution parameters $g^s(\cdot)$ for all location states.

TABLE 1. Notations and meanings.

Notations	Meanings
B	All the barometer reading observations.
L	All the locations states.
Π	The initial location state probability matrix.
π^s	The initial probability that the user begins in location state s.
S	Number of location states.
T	Number of time stamps.
$a_{t-1,t}$	The acceleration readings from time $t-1$ to t.
$g^s(\cdot)$	The distribution function of barometer readings for location state s.
l_t^s	The probability of being in location state s at t.
P^s	The parameters of the probability density function $g^s(\cdot)$.
$E_{t-1,t}$	The state transition probability matrix for locations on the road map from time $t-1$ to t.

With the above specification, we give the log likelihood of the observation data and hidden variables. It can be written as follows,

$$lnP(B,L) = lnP(L_1) + \sum_{t=2}^{T} lnP(L_t|L_{t-1}) + \sum_{t=1}^{T} lnP(B_t|L_t) \tag{1}$$

In this formula 1, $lnP(L_1)$ is the initial probability of the user location states, $\sum_{t=2}^{T} lnP(L_t|l_{L-1})$ represents the probability that location states L_{t-1} in time stamp $t-1$ transit to the states L_t in the next time stamp t, and $\sum_{t=1}^{T} lnP(B_t|L_t)$ is the probability of observation B_t conditioned on the location state L_t.

The initial probability of the location state at fist time stamp is

$$lnP(L_1) = \sum_{s=1}^{S} \pi^s \tag{2}$$

Since the the location state transit probability is dynamically determined by the user displacement during time stamp $t-1$ and t. The transition matrix here is dynamic, which is different from the normal HMM model where the transition matrix is constant. So, the log probability of location state transiting from time stamp $t-1$ to t is

$$lnP(L_t|L_{t-1}) = \sum_{i=1}^{S}\sum_{j=1}^{S} l_{t-1}^{s_i} l_t^{s_j} ln(E_{t-1,t}(i,j)) \tag{3}$$

Since the barometer reading observation is related to the location state, the probability of barometer reading observation B_t given the location states L_t can be represented as

$$lnP(B_t|L_t) = \sum_{i=1}^{S} l_t^s ln(g^s(B_t)) \tag{4}$$

In order to solve the model, we first need to get the initial probability of the location states π^s, the distribution function parameter P^s of the barometer reading observation and the state transition matrix E. Then, we maximize the expected complete log-likelihood as follows.

$$\begin{aligned}(P^s, E, \pi^s) &\leftarrow \underset{P,E,\Pi}{argmax}\, lnP(B,L,P,E,\Pi)\\ &\quad subject\ to \sum_{i=1}^{S} E(i,s) = 1, \quad E(i,s) \in [0,1],\ \forall i, s.\\ &\qquad\qquad \sum_{i=1}^{S} \pi^s = 1, \quad \pi^s \in [0,1],\ \forall s.\end{aligned} \tag{5}$$

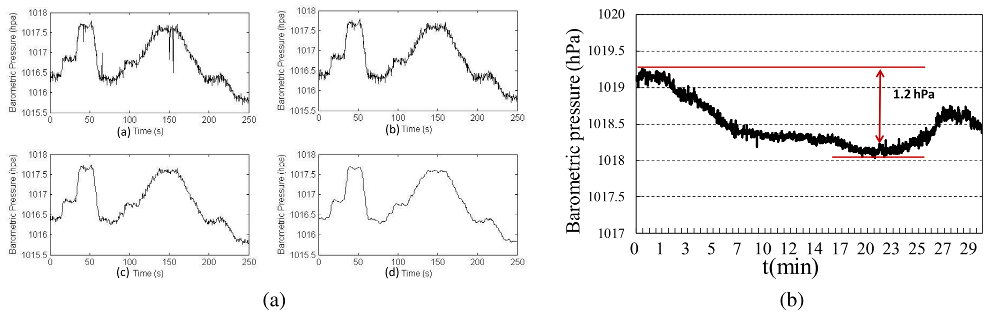


FIGURE 2. Data characteristic and data preprocessing. (a) Data preprocessing example. (b) Variation in atmospheric pressure by weather.

IV. SOLVING THE MODEL

Based on sensor based mobile context detection and statistical parameter estimation methods, we are able to get the three groups of parameters in the model. For maximizing the expected complete log-likelihood, we proposed an improved Viterbi algorithm to address the computational challenge.

First, for the three groups of parameters, we need to solve the following three technique problems respectively.

Initial Location Estimation: In order to get the values of π^s, we need to estimate the probabilities of the user's initial location. In another word, we need to estimate the user's initial location, in our implementation, it is calculated by a GPS reading sample of the user's initial location. However, BTrack does not depend on the GPS, any other way can be used to get a initial location instead of the GPS.

Barometer Reading Noise Remove: The barometer reading is conditioned on the altitude of the location. The distribution function parameter P^s can be got by the altitude of the location and the barometer sensor parameters. Unfortunately, the observed barometer reading contains noise caused by the weather change, we need to remove the noise before using it.

Displacement Estimation: The state transition matrix $E_{t-1,t}$ contains the probability that the user can reach location s_j from s_i within time $(t-1, t)$. We estimate the probability by detecting the displacement of the user using the acceleration readings $a_{t-1,t}$.

A. DATA COLLECTING AND PREPROCESSING

The barometer sampling rate is 2 readings per second, the accelerometer sampling rate is 10 readings per second. The raw sensing data contains noise, when visualized, the curve is rough and sometimes contains isolated points. Fig. 2(a)(a) displays the raw barometer readings which apparently contain noise. The isolated points are obvious, we remove them using a simple variation based solution. We filter a point out if the value change from the previous point is 5 times larger then the average in the last ten seconds, and the result is shown in Fig. 2(a)(b). After that, to filter out the high frequency parts of the data, we make use of a low-pass filter as shown in equation 6,

$$Y(n) = \beta X(n) + (1-\beta)Y(n-1) \tag{6}$$

$$f = \frac{\beta}{2\pi t} \tag{7}$$

where $X(n)$ is the n th sensor reading and $Y(n)$ is the output. β is the filter coefficient. If the value of β is too high, the filter result will not be obvious. If the value is set too low, the original data pattern may be lost. We set the filter coefficient for barometer reading and accelerometer readings separately,[1] the values are set based on the comparative experiments for different values using a 180 hours barometer data set and 18 hours accelerometer data set. f is the cut off frequency, and the filtering result is shown in Fig. 2(a)(c), and the high frequency parts of the data are diminished. After that, the values are smoothed with a reasonable window size[2] and the final result is shown in Fig. 2(a)(d).

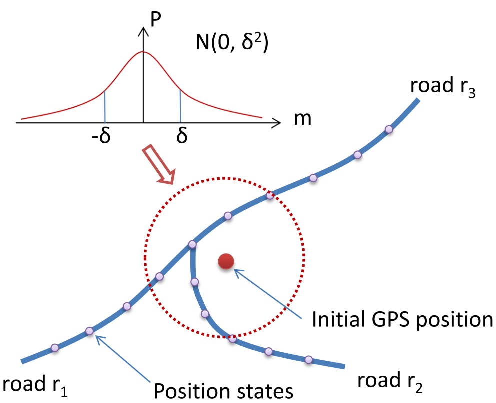
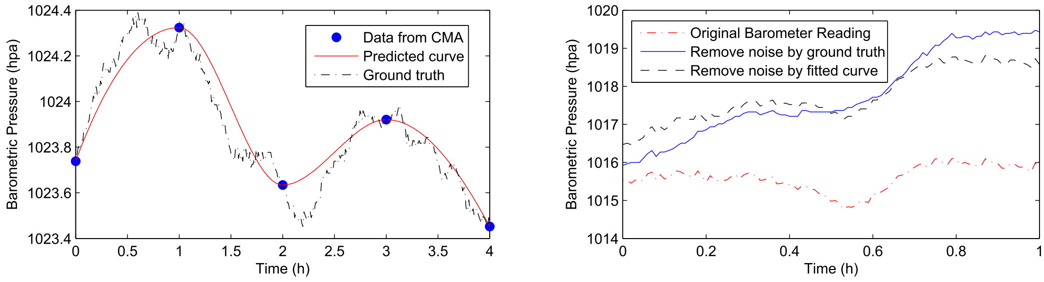


FIGURE 3. Probability density function of the initial location state.

B. INITIAL LOCATION ESTIMATION

In our system, the initial location of the user is got by a GPS sample, and we assume the error follows the Gaussian distribution $N(0, \sigma_{gps}^2)$, where σ_{gps} is the standard deviation with a value of 6 meters. In this way, we can get the probability of the user in every location. Fig. 3 shows an example, we map the latitude and longitude of the GPS sample to the two-dimensional road map, then we can get the probability of each location state pi^s based on its plane distance to the GPS location using the Gaussian density function. Notice that we

[1]The filter coefficient is set to 0.5 for barometer and 0.6 for accelerometer.
[2]The window size are 5 seconds for barometer and 1 second for accelerometer.



(a) (b)

FIGURE 4. Solutions to remove noise caused by weather. (a) Reconstruct the curve of the barometric pressure readings. (b) Filtering barometer reading noise caused by weather.

didn't use the altitude information of GPS because the error is often too large to estimate the location.

C. BAROMETER READING NOISE REMOVE

The barometer sensor is very common in smartphones today. The most commonly used are BMP series barometers from Bosch company, such as BMP180 and BMP280. When used to measure the altitude change, the accuracy can be within 1 meter [7]. However, this accuracy can only be achieved when the barometric pressure keeps unchanged during the measurement process. This is unrealistic since the barometric pressure changes by time as the weather conditions change, Fig. 2(b) shows an example, the barometric pressure changed by a max variation of 1.2 *hPa* in half an hour, which may result in an error of about 10 meters in altitude. In order to make sure that the observed barometer reading matches the distribution function, we need to first filter the noise caused by the weather.

An intuitive and feasible way to catch the change is to measure the barometric pressure continually in certain locations. Therefore, we tried to find the real time meteorological data from open access services provided by meteorological departments. The good news is their meteorological stations have real time weather data such as temperature, humidity and pressure. However, the provided service only updates the data once every hour, which is unacceptable because the accumulative error can be tens of meters in an hour. The other way is to predict the weather caused pressure change, it can help us to minimize the error. After a lot of effort, we got the weather forecast data from the national meteorological center of CMA [6]. It provides predicted pressure change hourly. The updating rate is still too low. To get the fine granted pressure change data. Our solution is to construct the complete curve of the barometric pressure using curve fitting.

As an example, a 4 hour pressure forecast is shown by the points in Fig. 4(a). Here we only get one reading for every hour. In order to reconstruct detailed pressure change curve, after many attempts, we observed that the cubic based fitted curve can reconstruct the curve well, and the fitted curve of the 24 hour pressure data is shown by the red line in Fig. 4(a).In this way, we get continuous pressure readings. Compared to the ground-truth which is shown by the black dotted line, we can see that the two curves matches well, except for a few parts where the curve fluctuates a lot.

Knowing the weather caused pressure change, we are able to extract the pressure change only caused by user altitude change from the barometer readings. As show in Fig. 4(b), the red dotted line is the raw barometer readings of a user when walking on a hillside road for 45 minutes. We first filter the weather noise by the predicted pressure change, the result is shown by the black line. For the reference, the blue line is the result after filtering the weather noise by manually collected pressure change ground-truth.

Now, the barometer reading change directly reflects the altitude change of the user. For example, the barometer reading change from h_0 to h_s is then calculated by equation 8.

$$\Delta b_s = b_0((\frac{h_s}{44330})^{5.255} - (\frac{h_0}{44330})^{5.255}) \tag{8}$$

where Δb_s is the barometer reading change from the initial location to the current location, h_s and h_0 is the altitudes of location s and the initial location, and p_0 is the sea level standard atmospheric pressure (101325 Pa).

The measured barometer reading after weather noise remove is still not perfect, it contains the inherent error of the sensor. Here, we assume the error of barometer reading follows the Gaussian distribution, it follows $N(0, \sigma_b^2)$, and the probability density function is:

$$f(b_{err}) = \frac{1}{\sqrt{2\pi}\sigma_b} e^{-\frac{b_{err}^2}{2\sigma_b^2}}, \quad \sigma_b = 15 \tag{9}$$

where σ_{al} is the standard deviation, and the value of σ_b is a statistic result from our experiment result. In this case, the distribution function parameter of the barometer reading observation P^s is $N(\Delta b_s, \sigma_b^2)$.

D. DISPLACEMENT ESTIMATION

The state transition matrix $E_{t-1,t}$ contains the probability that the user can reach location s_j from s_i within time $(t-1, t)$, which is equal to the probability the user displacement is equal to the distance between s_j and s_i. Consequently, we only need to find the possibility density function of user displacement using the acceleration readings $a_{t-1,t}$ between the locations s_i and s_j.

Displacement estimation based on acceleration is not new for localization and navigation solutions. There are two common ways, the first is to calculate the distance based on double integrating of acceleration. This approach is feasible for estimating car speeds but not for pedestrians because of large cumulative error. The other is a common used method, the periodic nature of pedestrian walking is reflected in accelerometer readings, we can count the number of steps a person has walked, and therefore derive the displacement of the user. On flat roads, the displacement is calculated by the product of the step number and step size of the user, but the error will accumulate over time [20]. A feasible solution is to reset the error by reference points or landmarks. However, in mountain roads, the situation will be more complicated, the roads may have different slopes and different step heights, which lead to different displacement for a footstep in different roads. The accumulated error will soon become very large in short distances. In our solution, we proposed a reasonable solution for displacement estimation which cares about these factors.

We get the displacement by the product of the step number and step size. The step number can be extracted from the acceleration readings, the technology is now very mature such as [16] and google activity API. For the step size, we propose different step size estimation solutions for stair roads and non-stair roads respectively. We first claim an assumption that some information of the road can be obtained from the road map, such as stair height, road slope and road conditions.

For stair roads, the step size can be estimated by the height of the steps. We assume a user climbed k stairs after climbing for n steps. The value of n may not be the same as k in real situations. For example, sometimes we may climb two stairs for a step. To handle this problem, we assume variable k follows the Poisson Distribution. This assumption is reasonable since the human step behavior is highly inhomogeneous. The probability density function of the climbed stairs value k is:

$$f_{stair}(k) = \frac{\lambda^k}{k!}e^{-\lambda}, \quad k = 0, 1, \ldots \tag{10}$$

where k is the number of stairs, and λ is the expected number of stairs. The value of λ is set to $1.1n$ in our setting, it is a statistical result from our experiment data. In this way, we can get the transition probability between location states based on the number of steps of a user.

$$E_{t-1,t}(i,j) = f_{stair}(n_{i,j}) = \frac{(1.2n)^{n_{i,j}}}{n_{i,j}!}e^{-(1.2n)} \tag{11}$$

where $n_{i,j}$ is the number of stairs between location state i and state j, and n is the number of steps of the user from $t-1$ to t.

For non-stair roads, the situation is a bit more complicated. We believe there are two factors that will affect the step size, the road slope and the road condition. Each factor independently leads to change accumulation of the displacement of each step. We quantify their affection as variables α, β, and the real step size d is calculated by the following equation:

$$d = d_0 * (1-\theta) \tag{12}$$

$$\theta = \alpha + \beta \tag{13}$$

where d_0 is the normal step size of the user which is already known. The variable α is the slope affection, and β is the road condition affection.

We assume variables α and β follow the Gaussian distributions $N(\mu_{rs}, \sigma_\alpha^2)$, $N(\mu_\beta, \sigma_\beta^2)$ respectively. They are independent of each other. In this way, θ also follow the Gaussian distribution $N(\mu_\alpha + \mu_\beta, \sigma_\alpha^2 + \sigma_\beta^2)$, and the probability density function is:

$$f_{step}(\theta) = \frac{1}{\sqrt{2\pi(\sigma_\alpha^2+\sigma_\beta^2)}}exp[-\frac{(\theta-(\mu_\alpha+\mu_\beta))^2}{2(\sigma_\alpha^2+\sigma_\beta^2)}] \tag{14}$$

where μ is the expected value and σ is the standard deviation. The values of μ_α, σ_α, μ_β and σ_β are set based on the statistical parameter estimation methods. We define 5 kinds of slopes, and 4 types of road conditions, and statistically collected and calculated these values.

In this way, we can get the transition probability between location states based on the number of steps of a user.

$$E_{t-1,t}(i,j) = P(d = \frac{d_{i,j}}{n}) = f_{step}(1 - \frac{d_{i,j}}{nd_0}) \tag{15}$$

where $d_{i,j}$ is the distance between location state i and state j, and n is the number of steps of the user from $t-1$ to t.

E. FINDING MOST PROBABLE SEQUENCE OF HIDDEN LOCATION STATES

In the previous sections, we get all the three groups of parameters: 1) the GPS error distribution $N(0, \sigma_{gps}^2)$ for the initial probability of the location states π^s, 2) The observation distribution function parameters P^s is $N(\Delta b_s, \sigma_b^2)$, and 3) the state transition probability matrix $E_{t-1,t}(i,j)$ can be calculated by $f_{stair}(n_{i,j})$ for stair roads and $f_{step}(1-\frac{d_{i,j}}{nd_0})$ for non-stair roads. Next, the way is to find the list of location states which can maximize the expected complete log-likelihood, based on formulas 1-4, the expected complete log likelihood is as follows.

$$lnP(B|A,L,P,E,\Pi) = \sum_{t=2}^{T}\sum_{i=1}^{S}\sum_{j=1}^{S} l_{t-1}^{s_i} l_t^{s_j} ln(E_{t-1,t}(i,j)) + \sum_{t=1}^{T}\sum_{i=1}^{S} l_t^s ln(g^s(b_t)) + \sum_{s=1}^{S}\pi^s \tag{16}$$

A common solution to find the sequence of hidden states is the Viterbi Algorithm, it is a dynamic programming algorithm for finding the most likely sequence of hidden states, especially in the context of Hidden Markov Models. The complexity of a Viterbi Algorithm is $O(T \times |S|^2)$. However, in our model, it is intractable to do energy efficient tracking for any reasonable road map with the solution due to the following reasons. 1) The number of hidden states S is very large, for example, a road map of $1km$ contains 1000

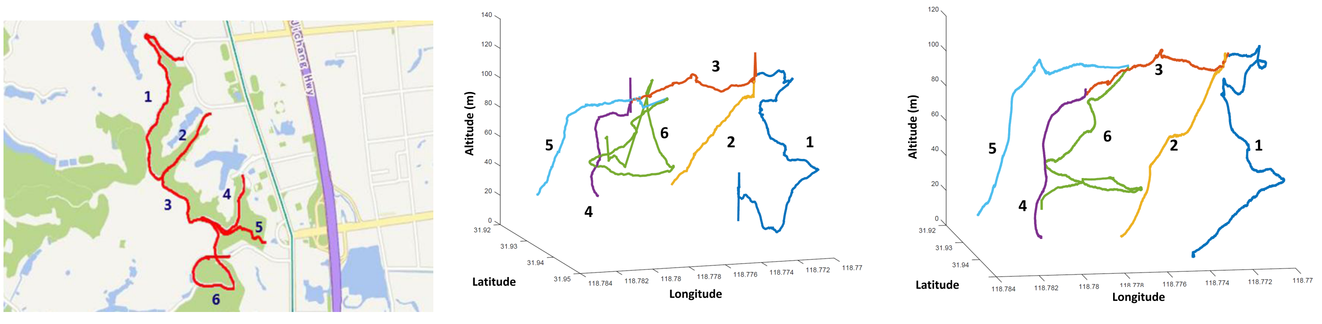


FIGURE 5. Maps of the mountain roads in the field study. (a) The 2D map of the mountain roads. (b) The 3D map generated by GPS data. (c) The 3D map generated by a electronic total station.

location states if the location interval is set to 1 meter. 2) the variable of the transition matrix should be updated in every time slot, 3) the transition matrix itself is $S \times S$, the storage requirement is high. To address this computational challenge, we proposed a approach which can adaptively change the size of the location states S in different time periods. It is based on the observation that the user trace is a sequence of continuous locations, it may not jump to a location which is far away from the previous location in a short time slot. The basic idea is to choose the location states not far away from the previous state based on a certain strategy. The strategy is to choose the locations within the maximum range that the user can reach. The detail is given in Algorithm 1.

Algorithm 1 Adaptively change the location state set

Input: The road map M and the latest known location l_0 at time t_1, the acceleration readings a_{t_1,t_2} collected from t_1 to t_2, the maximum step size d_0^{max}.
Output: The location state set S_{t_1,t_2};

```
Initialization: set S_{t1,t2} to empty;
Extract the number of steps: n = extractSteps(a_{t1,t2});
Compute the maximum displacement: d = n * d_0^max;
for each location l_i ∈ M do
    if the distance between l_0 and l_i is less than d then
        add l_i to S_{t1,t2}
    end if
end for
return S_{t1,t2}
```

After reducing the number of states, we use the Viterbi Algorithm to find the list of location states with the highest probability, and output the result to the tracking system. It's important to notice that, in our implementation of BTrack, when the highest probability is smaller than a threshold, it will request to get an initial position and start another locating process.

V. EVALUATION

To evaluate the performance of BTrack, we conducted some field studies on the CuiPing mountain near our university. As shown in Fig. 5(a), the field study area contains six road sections. The evaluation contains two parts. We first spend a week on collecting the experiment data and evaluate the performance of BTrack. After that, we compared BTrack with three existed representative solutions, a common GPS-based approach, a typical pedestrian dead reckoning based approach, and a wireless fingerprinting based approach [28].

A. EVALUATE THE PERFORMANCE OF BTRACK

The map data are extracted from the existed map service. Since the altitude information of the map are not accurate enough, we made a survey and draw of the road map with the help of an electronic total station. The 3D road map is shown in Fig. 5(c). We didn't use GPS altitude because of its low accuracy, Fig. 5(b) shows the 3D road map constructed by GPS, where the altitude error is unacceptable. We realized BTrack as an android application, and four Nexus 6 smartphones are used in the field study. At first, we did some experiments and evaluated the performance of the technologies used in BTrack. For example, the performance of initial location estimation, weather noise filtering and displacement estimation. These experiments also helped us to find some better settings of the parameters in the model. After that, we evaluated the accuracy and energy performance of BTrack by totally of about 10 kms of tracking data.

1) INITIAL LOCATION ESTIMATION

To evaluate the accuracy of initial location estimation of BTrack, we randomly chose 20 locations in the road map and collected about 240 GPS samples. In the collection process, when the participant arrives a location, he will first record the location ID, then wait for five seconds before he records the GPS sample. After that, he moves to the next location. The process continues until we get enough GPS samples. The accuracy is shown in Fig. 6(a), the average error is about 6 meters and the standard deviation is about 4.3 meters, and the result helped our parameter setting in the initial location estimation model.

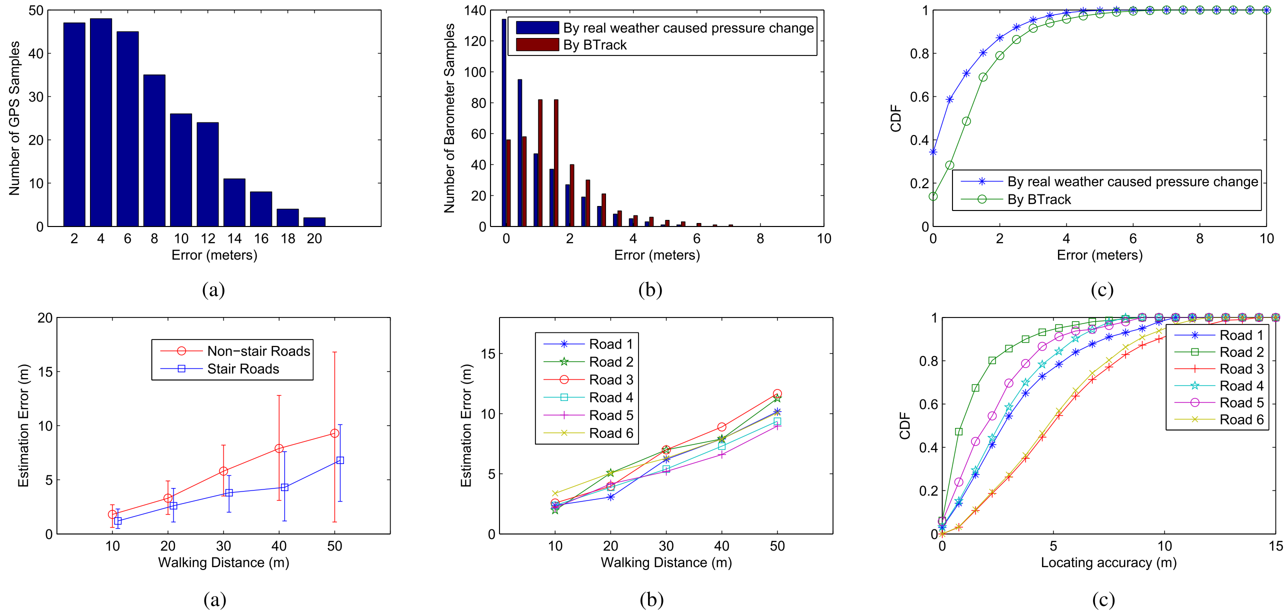


FIGURE 6. Evaluation results of the field study on the mountain roads. (a) Initial location estimation. (b) Accuracy of removing weather noise. (c) Cumulative density function (CDF) of removing weather noise. (d) Displacement estimation accuracy. (e) Displacement accuracy in different non-stair roads. (f) Locating accuracy on the six roads.

2) FILTERING WEATHER NOISE

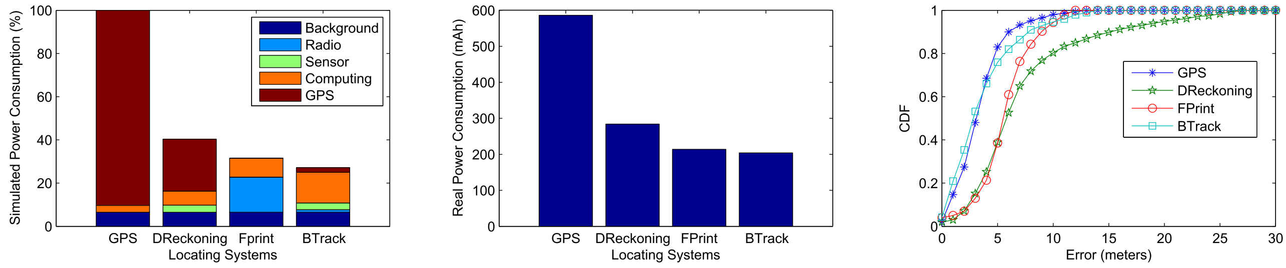
Here we evaluate the performance of removing the weather noise based on curve fitting. We randomly selected 400 barometer data points from the experiment trace data with the altitude ground-truth. The error can be got by the deviation between the barometer calculated altitude and the real altitude. We first show the deviation of the altitude change calculated by the barometer reading after noise filtering using BTrack. For comparison, we also show the deviation calculated by noise filtering using real weather caused pressure change. The real weather caused pressure change is recorded by an electronic barometer sensor deployed at the foot of the mountain during the whole process of the field study. The detailed performance is shown in Fig. 6(b) and Fig. 6(c). The red columns in Fig. 6(b) and green line in Fig. 6(c) show the weather noise filtering result of BTrack. A typical result is that the error is less than 2 meters in 79% of the cases, and the performance is very close to that of continually barometric pressure change monitoring.

3) DISPLACEMENT ESTIMATION

We first use the google activity recognition API to calculate the step number based on accelerometer readings, and then evaluate the displacement estimation accuracy for stair roads and non-stair roads separately. For stair roads, the accuracy is related to the moving habit of the user, in our field study, there are three participants and we choose three stair roads for the test. For non-stair roads, we choose six road sections with different road slopes and road conditions. During the test, every participant will walk through every road for three times. We recorded the points when he walks for every 10 meters, until 50 meters. All the accelerometer data and road data are logged and recorded, and we analyse the data off-line after the test. Fig. 6(d) shows the error of displacement estimation for stair and non-stair roads, and shows the relation between the accuracy and the walking distance. The error increases when the walking distance is longer, and the accuracy on stair roads is better than that on non-stair roads. Fig. 6(e) shows that the error of displacement estimation didn't show much difference on the six non-stair roads.

4) LOCATING ACCURACY

In order to evaluate the locating accuracy of BTrack, we need to get the real locations of the user as the ground-truth, however, this cannot be done easily. Before the test, we manually marked many signs along the road, each sign has an ID and the distance between every sign is five meters. When the participant is walking along the road, another participant will record the time when he arrives every sign. In this way, we can record the location ground-truth. The locating accuracy is measured by the comparison between BTrack output and the recorded user real location. Notice that we only evaluate the accuracy along the road and not considering the road width. Fig. 6(f) shows the average error when walking on the six different road sections. The error is less than 5 meters in about 80% situations. In detail, the accuracy on roads 2 and 5 are best because the road slope is higher, and the barometer reading change is clear when walking along the roads. Instead, the roads 3 and 6 have lower accuracy where the road slope is not obvious, and the accuracy rely more on displacement estimation rather than the barometric pressure observation.



(a) (b) (c)

FIGURE 7. Evaluation results when comparing with the related works. (a) Simulated power consumption. (b) Real power consumption. (c) Locate accuracy CDF compared to the related works.

5) POWER CONSUMPTION

The power consumption sources of BTrack contain two parts. The first part is sensing and communicating, the second part is computing. Once started, BTrack software continuously collects barometer readings and accelerometer readings, and hourly access the weather service on internet for barometric pressure data. The main purpose of computing is to run BTrack algorithms to calculate the location of the user. The location of the user is updated every 5 seconds. Measurements were performed on Nexus 6 using the Monsoon Power Monitor. Based on the result, for a typical 3220 mAh and 3.8 V smartphone (Nexus 6), the energy consumption for tracking a user for half an hour is averagely 6.3% (204 of 3220 mAh), this is acceptable for the users based on our survey.

B. EVALUATE BTRACK WITH EXISTED WORKS

In the second phase, we compare BTrack with three existed approaches.

1) THE COMMON GPS BASED APPROACH

This is an intuitive solution and it is very common used today. The way is to continually sample the GPS coordinates and then map them to the road using the road-mapping algorithm. To evaluate the accuracy, we test it under the same scenario as BTrack, we log its output location when the user arrives every location sign we pre marked along the roads, and the error is calculated by the distance between the output location and the marked location.

2) THE PHONE BASED PEDESTRIAN DEAD RECKONING [13], [29]

This covers the methods for tracking the user by dead reckoning. The main idea is to use the low power sensors on smartphone as an assistance for GPS to estimate the location of the user. In this case, the GPS sampling frequency can be reduced and the power consumption can be minimized. Here we make a typical implementation, we use the accelerometer and compass to estimate the moving distance and moving direction, and use the GPS to recalibrate after a certain number of steps. The location signs we pre marked along the roads are used as the ground-truth, and the error is calculated by the distance between the output location and the marked location.

3) THE WIRELESS FINGERPRINT BASED APPROACH [28]

The solution is to make use of the wireless fingerprint to locate the moving user. In our evaluation, we implemented a prototype locating system based on the approach from patent US20140012529 [28], which locates the user by the Bluetooth fingerprinting. We totally deployed 50 Bluetooth beacons, and one in every 10 meters along the road. Before locating the user, we did a site survey and built the fingerprint map which map the fingerprints to the locations. After that, we can locate the user by looking up the fingerprinting map.

In order to evaluate the performance of these solutions, we developed a prototype application based on these three approaches, and evaluated them under the same scenario as BTrack. The pre marked location signs are used as the ground-truth. During the test, one participant takes four full charged nexus 6 smartphones, and running the four locating systems respectively. When he is moving along the road, all the data are logged on the disk, including the location output, power file and other sensing data. Another participant follows him and record the time stamps when he passes every location sign. After the test, we collected about 10 kms of tracking data for evaluation.

Fig. 7(c) shows the CDF of the locating error. BTrack is better than the GPS based approach in 56% cases, and the error is less than 5 meters in 76% cases. Fig. 7(a) shows the simulation results when we break-down the power consumption sources of these systems, and Fig. 7(b) shows the real power consumption for half an hour. The energy consumption of BTrack is much less than the GPS based approach. BTrack mainly uses the barometer and accelerometer sensors and the GPS energy consumption is negligible. The dead reckoning based approach is more energy efficient than the GPS based approach, but the accuracy is pretty low. The fingerprint based approach is more accurate than the dead reckoning approach, and it is also energy efficient. However, the deployment of the Bluetooth infrastructure is expensive and complicated, it's very hard to be widely used in the mountain areas. Comparing with the three approaches, BTrack has a good accuracy and the low energy consumption.

VI. RELATED WORK

GPS is the most common used locating system for outdoor locating and tracking. However, the drawbacks are obvious.

The power consumption is high [9], and the accuracy can be affected by obstructions such as trees on mountain areas and tall buildings in the city center. The GPS model of today's most smartphones make use of the single-frequency positioning method, which is affected by the ionospheric delay. Although some new smartphones support double-frequency positioning, the extra energy consumption further reduced the energy efficiency. In BTrack, we make use of the low power sensors on smartphones, with the help of the 3D map of mountain roads, BTrack can track the user with very low energy consumption, and the accuracy is no worse than GPS.

For better energy efficiency, researchers try to utilize some techniques for reducing the GPS sampling rate. Ramos *et al.* [14] can infer the data necessary to perform GPS calculation, but they need a reference point nearby. SmartDC [4] proposed a mobility prediction-based adaptive duty cycling scheme, and the work [30] conserve energy in the framework level. Others utilize the low power sensors on smartphones to assist GPS [29]. The most commonly used technology is dead reckoning, it is the process of calculating the next position by a previously determined position. They often make use of the accelerometer to detect the step number of the user [16]. For example, the work [13] realized a collection of techniques to cleverly determine when to turn on GPS and efficiently estimates user movement using a duty-cycled accelerometer. However, the accuracy cannot be always ensured and it needs cooperation with other devices. Bhattacharya *et al.* [1] proposed sensor management strategies for trajectory tracking, but it can't effectively save power when accuracy requirement is high. To sum up, although the GPS sampling rate is reduced, they still heavily dependent on GPS, the real energy saving is limited, because periodically initializing and powering off the GPS model will cause extra energy consumption [9]. Furthermore, the accuracy cannot be ensured especially on mountain roads. Not only because of the weak GPS signal but also because of the complicated road conditions, and the common distance estimation techniques will cause large error. Comparing to these solutions, BTrack is more suitable for user tracking on mountain roads.

There also exist other solutions which can track the user without GPS. PlaceLab [10] is one of the earliest solutions which can track devices using wireless signal in the wild. Some solutions locate by distance measurement between the target and the reference points. The location of the user can be calculated by the TOA [2], [3], RSS [12], and TOF [22] distance measurements based on wireless references. However, these solutions needs high density infrastructure deployment and the high precision equipments are very expensive. Other solutions locate by wireless fingerprints [19]. The solutions [8] and [26] locate the user by RSSI-based fingerprint feature, they need an off-line process to build the fingerprint map before locating the user. This process needs a large amount of human work, and the accuracy is not perfect. Teng *et al.* [17] make use of visual signals to help improve the accuracy of wireless localization, but extra video equipments are needed. The wireless infrastructure is poor in mountain areas, unlike city areas where it is ample. Although these solutions ensured low energy consumption at the target device, but paid more elsewhere. BTrack do not depend on these infrastructures and the total cost is very low. Other techniques such as WheelLoc [21] only make use of low power sensors and cell tower information, but the accuracy is not good enough compared to BTrack.

There is an increasing availability of barometer embedded smartphones recently. Although it is first introduced for aiding GPS [27], people find more usage scenarios. Due to the barometer's high sensitivity, it is well suited for floor-change detection [18], [25]. It can also be used to detect door opens and closes [23] in buildings. Sankaran use it for transportation detection [15], and proved that the barometer is more energy efficient than traditional sensors such as accelerometers. As far as we known, BTrack is the first solution which makes use of the barometer as the major sensor for location tracking on mountain roads.

VII. CONCLUSION AND DISCUSSION

In our work, we demonstrate the approach for location tracking on mountain roads. It satisfies all the three requirements of an ideal tracking system on mountain roads. First, BTrack only makes use of the low power barometer and accelerometer senors on smartphone, which is very energy efficient. Then, the accuracy BTrack is no worse than GPS, sometimes even better than GPS. At last, BTrack dose not need any extra infrastructure or device change, which is very practical for real usage. Compared to the existing works, we find that BTrack is more suitable for location tracking on mountain roads. However, BTrack still has some limitations. For example, HiMeter performs not very well on roads with very small slope. At that time, the accuracy may reduce obviously. We will consider this problem in our future work.

REFERENCES

[1] S. Bhattacharya, H. Blunck, M. B. Kjærgaard, and P. Nurmi, "Robust and energy-efficient trajectory tracking for mobile devices," *IEEE Trans. Mobile Comput.*, vol. 14, no. 2, pp. 430–443, Feb. 2014.

[2] Y.-T. Chan, W.-Y. Tsui, H.-C. So, and P.-C. Ching, "Time-of-arrival based localization under NLOS conditions," *IEEE Trans. Veh. Technol.*, vol. 55, no. 1, pp. 17–24, Jan. 2006.

[3] K. W. Cheung, H. C. So, W. K. Ma, and Y. T. Chan, "Least squares algorithms for time-of-arrival-based mobile location," *IEEE Trans. Signal Process.*, vol. 52, no. 4, pp. 1121–1130, Apr. 2004.

[4] Y. Chon, E. Talipov, H. Shin, and H. Cha, "SmartDC: Mobility prediction-based adaptive duty cycling for everyday location monitoring," *IEEE Trans. Mobile Comput.*, vol. 13, no. 3, pp. 512–525, Mar. 2014.

[5] K. Dong, W. Wu, H. Ye, M. Yang, Z. Ling, and W. Yu, "Canoe: An autonomous infrastructure-free indoor navigation system," *Sensors*, vol. 17, no. 5, p. 996, 2017.

[6] N. M. C. Forecast System Laboratory. *National Meteorological Center*. Accessed: Apr. 2018. [Online]. Available: http://eng.nmc.cn/

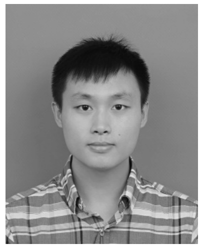
[7] R. B. GmbH. *BMP280 Barometric Pressure Sensors*. Accessed: Apr. 1, 2018. [Online]. Available: https://www.bosch-sensortec.com/bst/products/all_products/bmp280

[8] S. He and S.-H. G. Chan, "INTRI: Contour-based trilateration for indoor fingerprint-based localization," *IEEE Trans. Mobile Comput.*, vol. 16, no. 6, pp. 1676–1690, Jun. 2017.

[9] M. B. Kjærgaard, J. Langdal, T. Godsk, and T. Toftkjær, "EnTracked: Energy-efficient robust position tracking for mobile devices," in *Proc. Int. Conf. Mobile Syst., Appl., Services*, 2009, pp. 221–234.

[10] A. LaMarca *et al.*, "Place lab: Device positioning using radio beacons in the wild," in *Proc. Int. Conf. Pervasive Comput.* Berlin, Germany: Springer, 2005, pp. 116–133.
[11] G. Liu *et al.*, "Beyond horizontal location context: Measuring elevation using smartphone's barometer," in *Proc. ACM Int. Joint Conf. Pervas. Ubiquitous Comput., Adjunct Publication*, 2014, pp. 459–468.
[12] A. Narzullaev, Y. Park, K. Yoo, and J. Yu, "A fast and accurate calibration algorithm for real-time locating systems based on the received signal strength indication," *AEU-Int. J. Electron. Commun.*, vol. 65, no. 4, pp. 305–311, 2011.
[13] J. Paek, J. Kim, and R. Govindan, "Energy-efficient rate-adaptive GPS-based positioning for smartphones," in *Proc. Int. Conf. Mobile Syst., Appl., Services*, 2010, pp. 299–314.
[14] H. S. Ramos, T. Zhang, J. Liu, N. B. Priyantha, and A. Kansal, "LEAP: A low energy assisted GPS for trajectory-based services," in *Proc. Int. Conf. Ubiquitous Comput.*, 2011, pp. 335–344.
[15] K. Sankaran, M. Zhu, X. F. Guo, A. L. Ananda, M. C. Chan, and L.-S. Peh, "Using mobile phone barometer for low-power transportation context detection," in *Proc. ACM Conf. Embedded Netw. Sensor Syst.*, 2014, pp. 191–205.
[16] M. Susi, V. Renaudin, and G. Lachapelle, "Motion mode recognition and step detection algorithms for mobile phone users," *Sensors*, vol. 13, no. 2, pp. 1539–1562, Jan. 2013.
[17] J. Teng, B. Zhang, J. Zhu, X. Li, D. Xuan, and Y. F. Zheng, "EV-Loc: Integrating electronic and visual signals for accurate localization," *IEEE/ACM Trans. Netw.*, vol. 22, no. 4, pp. 1285–1296, Aug. 2014.
[18] S. Vanini and S. Giordano, "Adaptive context-agnostic floor transition detection on smart mobile devices," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops*, Mar. 2013, pp. 2–7.
[19] Q. D. Vo and P. De, "A survey of fingerprint-based outdoor localization," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 491–506, 1st Quart., 2015.
[20] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: Unsupervised indoor localization," in *Proc. Int. Conf. Mobile Syst., Appl., Services*, 2012, pp. 197–210.
[21] H. Wang, Z. Wang, G. Shen, F. Li, S. Han, and F. Zhao, "WheelLoc: Enabling continuous location service on mobile phone for outdoor scenarios," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 2733–2741.
[22] J. Wang, Q. Gao, Y. Yu, X. Zhang, and X. Feng, "Time and energy efficient TOF-based device-free wireless localization," *IEEE Trans. Ind. Informat.*, vol. 12, no. 1, pp. 158–168, Feb. 2017.
[23] M. Wu, P. H. Pathak, and P. Mohapatra, "Monitoring building door events using barometer sensor in smartphones," in *Proc. UbiComp*, 2015, pp. 319–323.
[24] H. Ye, K. Dong, and T. Gu, "Himeter: Telling you the height rather than the altitude," *Sensors*, vol. 18, no. 6, p. 1712, 2018.
[25] H. Ye, T. Gu, X. Tao, and J. Lu, "Scalable floor localization using barometer on smartphone," *Wireless Commun. Mobile Comput.*, vol. 16, no. 16, pp. 2557–2571, 2016.
[26] A. Zhang, Y. Yuan, Q. Wu, S. Zhu, and J. Deng, "Wireless localization based on RSSI fingerprint feature vector," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 11, p. 528747, 2015.
[27] J. Zhang, E. Edwan, J. Zhou, W. Chai, and O. Loffeld, "Performance investigation of barometer aided GPS/MEMS-IMU integration," in *Proc. Position Location Navigat. Symp.*, Apr. 2012, pp. 598–604.
[28] J. Zhu *et al.*, "Bluetooth beacon based location determination," U.S. Patent 8 965 398, Feb. 24, 2015.
[29] X. Zhu, Q. Li, and G. Chen, "APT: Accurate outdoor pedestrian tracking with smartphones," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 2508–2516.
[30] Z. Zhuang, K.-H. Kim, and J. P. Singh, "Improving energy efficiency of location sensing on smartphones," in *Proc. Int. Conf. Mobile Syst., Appl., Services*, 2010, pp. 315–330.

HAIBO YE received the Ph.D. degree in computer science from Nanjing University in 2016. He is currently a Lecturer with the Department of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China. His research interests include mobile localization and navigation, mobile and pervasive computing.



WENHUA YANG received the Ph.D. degree in the computer science and technology from Nanjing University. He is currently an Assistant Professor with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics. His research interests include software engineering, self-adaptive software systems, and IoT systems.

YUNYU YAO was born in 1996. She received the B.S. degree from the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, in 2018. She is currently pursuing the Ph.D. degree with University College London. Her research interests include mobile localization and navigation.

TAO GU (SM'13) received the B.Sc. degree from the Huazhong University of Science and Technology, the M.Sc. degree from Nanyang Technological University, Singapore, and the Ph.D. degree in computer science from the National University of Singapore. He is currently an Associate Professor with the School of Computer Science and Information Technology, RMIT University. His research interests include mobile and pervasive computing, wireless sensor networks, distributed network systems, sensor data analytics, cyber physical system, Internet of Things, and online social networks. He is a member of the ACM.

ZHIQIU HUANG received the B.S. and M.S. degrees from the National University of Defense Technology, China, and the Ph.D. degree from the Nanjing University of Aeronautics and Astronautics, China. He is currently a Professor and a Tutor of Ph.D. candidate with the Nanjing University of Aeronautics and Astronautics, China. His major research interests include formal method, software engineering, and cloud computing.