SmartDC: Mobility Prediction-Based Adaptive Duty Cycling for Everyday Location Monitoring

Yohan Chon, Elmurod Talipov, Hyojeong Shin, and Hojung Cha, *Member*, *IEEE*

Abstract—Monitoring a user's mobility during daily life is an essential requirement in providing advanced mobile services. While extensive attempts have been made to monitor user mobility, previous work has rarely addressed issues with predictions of temporal behavior in real deployment. In this paper, we introduce SmartDC, a mobility prediction-based adaptive duty cycling scheme to provide contextual information about a user's mobility: time-resolved places and paths. Unlike previous approaches that focused on minimizing energy consumption for tracking raw coordinates, we propose efficient techniques to maximize the accuracy of monitoring meaningful places with a given energy constraint. SmartDC comprises unsupervised mobility learner, mobility predictor, and Markov decision process-based adaptive duty cycling. SmartDC estimates the regularity of individual mobility and predicts residence time at places to determine efficient sensing schedules. Our experiment results show that SmartDC consumes 81 percent less energy than the periodic sensing schemes, and 87 percent less energy than a scheme employing context-aware sensing, yet it still correctly monitors 90 percent of a user's location changes within a 160-second delay.

Index Terms—Location, mobility learning, mobility prediction, adaptive sensing, energy efficient

1 Introduction

Understanding and predicting human mobility is a fundamental resource for broad-domain applications. For instance, service providers predict a user's behavior (e.g., having lunch or going home) and provide appropriate services in a timely manner (e.g., sending lunch coupons or home preheating). Understanding human mobility, in general, provides useful information for traffic engineering, urban planning, predicting the spread of human and electronic viruses, and resource management in mobile communications [2], [3], [4], [5]. All these applications benefit from understanding and predicting human mobility: time-resolved places and paths.

Mobile phones are widely used for tracing human mobility since mobile phones 1) have almost 100 percent penetration, 2) are closely tied to daily life, and 3) are capable of locating themselves using various approaches. The global positioning system (GPS) and wireless positioning system (WPS) using cell tower and Wi-Fi access points (AP) are common technologies that provide a user's raw coordinates (i.e., latitude and longitude) [6], [7]. Ambient fingerprints are often constructed to recognize semantic places with room-level accuracy using radio beacons (e.g., cell towers, Wi-Fi APs, and Bluetooth) and surrounding factors (e.g., light, color, texture, and sound patterns) [8], [9]. A simple choice for monitoring mobility is to periodically sense a user's location context. Such a scheme, however, significantly reduces the battery's lifetime in mobile devices. To optimize energy consumption for continuous sensing, various approaches have been proposed. These include sensor selection by movement detector using accelerometers [9], [10], [11], [12], minimizing energy consumption within accuracy requirements [13], [11], minimizing location error for a given energy budget [14], [10], [12], and utilizing a prediction-based approach [14], [15], [16]. While extensive attempts have been made to continuously examine a user's mobility with less energy consumption, we argue that previous work did not fully consider regular patterns in human mobility to reduce energy consumption in real deployments.

Our research goal is to develop a framework that continuously provides location context with minimum energy consumption. The key motivations of our work are as follows: 1) finding meaningful places is a key target of human-centric mobile services, since people spend approximately 87 percent of their time indoors [17]; and 2) human mobility is amenable to prediction because of habitual time-resolved movements with reasonably small variation [2]. Thus, we focus on monitoring meaningful places (i.e., points of interest, or POIs) using the regularity of individual mobility pattern. The main idea is that the system senses location context based on a predicted schedule; that is, when the movement to the next location will take place. In other words, when a user visits a place our system makes predictions on departure times (i.e., residence time in the place). The key technical challenges are 1) simultaneous learning and predicting a user's mobility, 2) adaptive duty cycling that covers both the regularity and the randomness in human mobility, and 3) minimizing energy consumption.

In this context, this paper proposes SmartDC: mobility prediction-based adaptive duty cycling for everyday location monitoring. SmartDC comprises three components: *mobility learner*, *mobility predictor*, and *adaptive duty cycling*. *Mobility learner* uses unsupervised learning to incrementally

- *The authors are with the Department of Computer Science, Yonsei University, Shinchon-dong 134, Yonsei Engineering Research Center (YERC) 225A, Seodaemun-gu, Seoul 120-749, Korea.*
E-mail: {yohan, elmurod, hjshin, hjcha}@cs.yonsei.ac.kr.

Manuscript received 25 May 2012; revised 16 Nov. 2012; accepted 12 Dec. 2012; published online 14 Jan. 2013.
For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number TMC-2012-05-0269.
Digital Object Identifier no. 10.1109/TMC.2013.14.

TABLE 1
Summary of Features in Related Systems for Everyday Location Monitoring

System	Major Context	Sensor Selection						Adaptive Scheduling Policy (Constraints)	Mobility Prediction	Evaluation Platform
		Movement Detector		*Path Tracking*		*Place Recognition*				
EnLoc [14]	Raw Coordinates	-	-	Cell GPS Wi-Fi	Adaptive	-	-	Minimize Location Error (Energy Budget)	Heuristic	Unknown
Jigsaw [10]	Raw Coordinates	Accel[a]	Always	GPS	Adaptive	-	-	Minimize Location Error (Energy Budget)	-	Apple iPhone Nokia N95
Zhuang [12]	Raw Coordinates	Accel	Always	GPS WPS[b]	Adaptive	-	-	Adjust GPS Sample Rate (Battery Level)	-	HTC G1
RAPS [11]	Raw Coordinates	Accel	12.5%	GPS	Adaptive	-	-	Minimize Energy Consumption (Location Error)	-	Nokia N95
CAPS [13]	Raw Coordinates	-	-	Cell GPS	Always Adaptive	-	-	Minimize Energy Consumption (Location Error)	-	Google Nexus One
iLoc [16]	Meaningful Place	GSM Wi-Fi	30sec 2min	GPS	Always	GSM Wi-Fi	30sec 2min	Fixed Interval	Viterbi Algorithm	P4-2GHz PC
SensLoc [9]	Meaningful Place	Accel	50%	GPS	10sec	Wi-Fi	10sec	Fixed Interval	-	HTC G1
SmartDC	Meaningful Place	GSM Wi-Fi	2min Adaptive	GPS WPS	Adaptive	Wi-Fi	Adaptive	Maximize Place Detection (Energy Budget)	MDP	HTC Desire

a. Accelerometer, b. Wireless positioning systems provided by Android platform

collect mobility patterns in colloquial terms. Based on our previous work [18], [19], we developed a personalized scheme that collects POI's raw coordinates and also recognizes POIs with room-level accuracy. *Mobility predictor* uses a location predictor to predict departure time to the next location. We implemented both location-dependent and location-independent predictors, and compared their cost and performance. *Adaptive duty cycling* uses a Markov decision process (MDP) to determine the efficient sensing moment for a given energy budget. The proposed scheme maximizes the accuracy of mobility monitoring based on the regularity in individual mobility. The primary contributions of our work are:

- Designing simultaneous and incremental mobility learning and prediction components (Section 4).
- An extensive performance analysis of several location predictors for the estimation of predictable regularity in human mobility (Section 5).
- Evaluating a real system for everyday location monitoring using traces of 57 users (Section 5).

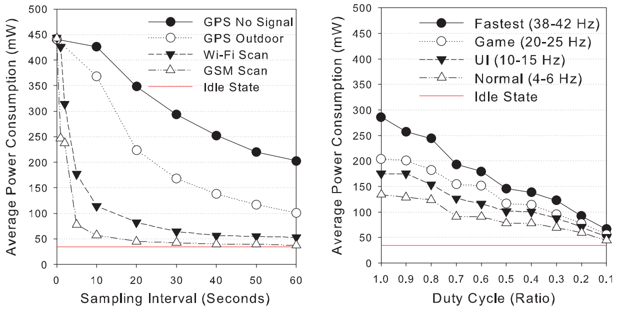
We implemented SmartDC on the Android framework as a service, and evaluated the system using traces of users for over four weeks in real environments. While the performance depended on a given energy budget, our extensive evaluation showed that SmartDC simultaneously performed mobility learning and prediction, and outperformed previous systems in terms of accuracy versus energy consumption.

2 Related Work

We first discuss the previous work on energy-efficient sensing, then describe the related studies on human mobility prediction. Table 1 summarizes the key features of related systems. According to the target context, we categorized context-monitoring systems into two groups: raw coordinates and meaningful places. To reduce energy consumption, most systems use a movement detector to activate different sensors in both moving and stationary states [9], [10], [11], [12], [16]. The user's speed estimated by GPS is also commonly used to switch around localization techniques [9], [11].

To track a user's raw coordinates, many approaches have been proposed to minimize energy consumption with adaptive sensor scheduling [10], [11], [12], [13], [14]. EnLoc [14] uses a dynamic programming technique to minimize location error for a given energy budget. A heuristic approach with a mobility tree is used to predict the next sensing time. The system, however, does not implement incremental learning, but instead uses a manually constructed mobility tree. Jigsaw [10] recognizes a user's activity using an accelerometer signal and uses MDP process to adjust the GPS sampling rate. Zhuang et al. [12] use location profiles and battery levels to adjust the GPS sampling rate. With these schemes, however, none uses an unsupervised prediction module or recognizes a place with room-level accuracy. Works in [11], [13] minimize energy consumption by adaptively using a GPS, only if the estimated location error exceeds an accuracy threshold. To estimate uncertainty in a location, each system uses different approaches. RAPS [11] uses moving distance, space-time history, and cell tower-based blacklisting. CAPS [13] uses cell-sequence matching to minimize the GPS sampling at revisited paths. In contrast to these systems, SmartDC uses an individual mobility pattern to estimate the uncertainty of a user's behavior without powering up the sensors. Additionally, we focus on detecting meaningful places, which is an essential component in advanced mobile services.

To recognize meaningful places, ambient fingerprints have been proposed to provide POIs with room-level accuracy. SensLoc [9] uses the Tanimoto coefficient of the Wi-Fi vector to detect the entrance and departure time. The system reduces energy consumption based on sensor selection, but the scheme does not use mobility prediction to optimize the sensing schedule. iLoc [16] uses the cosine similarity of GSM and Wi-Fi vectors to detect movement and meaningful places. The system uses the Viterbi algorithm to predict future location, but sensor scheduling is not considered in reducing the energy consumption.



(a) (b)

Fig. 1. HTC desire energy profile. We measured power consumption of several sensors, such as (a) GSM, Wi-Fi, and GPS with periodic sampling; and (b) accelerometer with duty cycle. The profile data and test software are downloadable from our website (http://lifemap.yonsei.ac.kr).

Isaacman et al. [20] use cell tower connections to monitor human mobility. The cell tower-based approach generates places with coarse granularity (i.e., a few hundred meters), and certainly has limits in capturing mobility at a fine-grained level. We analyzed these approaches to design a mobility learner, and to develop an adaptive sensor scheduling based on mobility prediction.

Extensive researches have been made on mobility prediction, especially in networking communities [4], [21]. The key concept is to compare the current pattern with historical data, and to extract similar patterns to predict next location. Works in [4] using a two-year trace on campus-wide wireless network found that the complex predictors were at best only negligibly better than the simple Markov predictor in practice. In addition, the simple predictor is appropriate for use on resource-constrained devices, such as smartphones, because of low computation overheads and modest storage requirements [21]. Considering that our goal is to minimize energy consumption on mobile devices, we implemented and compared simple predictors: we chose a Markov-based model [22], [23] as a location-dependent predictor, and nonlinear time series analysis [24] as a location-independent predictor. We also apply time-aided scheme and fallback mechanism to filter out redundant information in history data.

To the best of our knowledge, SmartDC is the first system that applies a simultaneous mobility learning and prediction scheme to mobile phones. The proposed system gradually learns a user's mobility pattern, and optimizes sensing schedule using the predictable regularity of individual behavior.

3 Preliminary Study

Continuous mobility learning without duty cycling is not a viable scheme because of the significant energy usage. We measured the average power consumption of each sensor to estimate the energy cost of mobility learning. To maintain the measurement consistency, we used the Monsoon Power Monitor to deliver a constant voltage of 3.7 V and halted all applications except for a system service. Fig. 1 shows the energy profile of HTC Desire. The idle state consumes 34.5 mW. Sensors are listed in the order of energy consumption: GPS, accelerometer, Wi-Fi, and GSM. The mobile phone performs GSM scans without additional energy cost because it always maintains a cell tower connection for voice communications. The duty cycle of the accelerometer should be carefully designed, since operating the accelerometer can incur significant energy usage. The application should lock the CPU to continuously use the accelerometer and prevent sleep state for continuous sensing. Although the accelerometer itself uses very little power, the continuous use of accelerometer needs to keep the CPU as well as associated high-power components active to access the sensor data [25]. Thus, computing the accelerometer signal with a 50 percent duty cycle at the lowest sampling rate (4-6 Hz) for 10 minutes consumes more energy than turning on a GPS with 1-minute intervals for 5 minutes. For this reason, most of the mobile platforms (e.g., Android, iPhone, and Nokia Maemo) restrict continuous sampling of acceleration while the screen is turned off [26]. Although a dedicated microcontroller may reduce sensing energy [25], even the latest smartphones do not employ such additional processor for sensing purpose. The proposed scheme, therefore, does not use an accelerometer for the pragmatic reason.

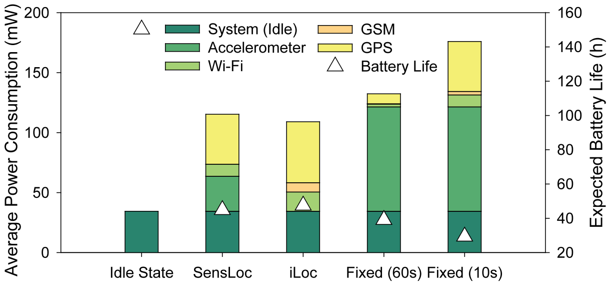


Fig. 2. Power consumption analysis on work related to mobility learning. We used proposed parameters of related works as shown in Table 1. The fixed scheme uses a fixed time interval. We assumed that the GPS is turned on only if a user is moving.

Based on the energy profile, we estimated the energy consumption of various mobility learning schemes. We made two assumptions: 1) A user spends 3 hours to move around outdoors each day [17]; and 2) the movement detector always recognizes user movements correctly. For example, SensLoc uses GPS and Wi-Fi every 10 seconds while a user moves, and the system activates an accelerometer with a 50 percent duty cycle when a user is stationary. Thus, the stationary state consumes 67.8 mW, and the moving state consumes 447.8 mW, which is derived from the sum of the idle state (34.5 mW), the GPS reading every 10 seconds outdoors (333.2 mW), and the Wi-Fi scanning every 10 seconds (80.1 mW). The average power consumption is $447.8 \text{ mW} \times \frac{3}{24} + 67.8 \text{ mW} \times \frac{21}{24} = 115.3 \text{ mW}$.

Fig. 2 presents the power consumption of several schemes. The expected battery lifetime is 29 to 48 hours if a smartphone runs only mobility learning schemes. A previous study showed that, without mobility learning, 60 percent of people used their smartphone from 14 to 41 hours with a single battery charge [27]. This means that mobility learning may reduce battery lifetime by at least 16 percent, and by 53 percent at most. Such energy consumption is a burden to users, since mobility learning is not a primary function of smartphones. Considering that the expected lifetime of idle state is 150 hours, previous

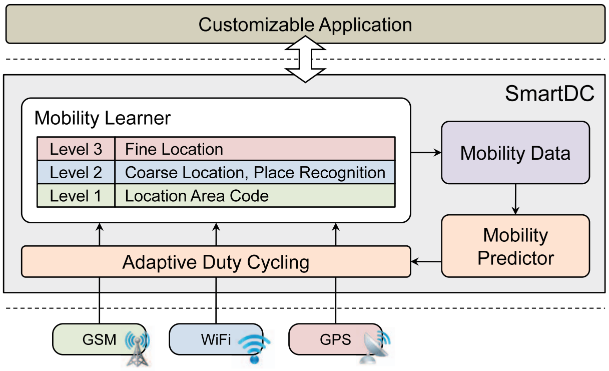


Fig. 3. System architecture of SmartDC.

learning schemes have room for lifetime improvement. The optimal scenario is that the system turns on sensors only when a user changes her states (i.e., entrance and departure moments). Our main idea is to adaptively sense the moment that includes a considerable possibility of state change. We predict the state change using the regularity of individual mobility patterns. The proposed system uses an energy budget as a constraint to customize sensing schedules to diverse usage.

4 SmartDC System

We first describe SmartDC usage scenario, then present the technical details. SmartDC runs as a background service in mobile phones. The energy budget for SmartDC is initially defined as the percentage of remaining battery charge, but a user may manually change it. When a user stays at a place for a certain period of time, SmartDC considers it a meaningful place and generates a node with place signatures that include location, Internet connectivity, visiting time, residence time, and the Wi-Fi fingerprints. The system incrementally constructs users' meaningful places with room-level accuracy in their daily lives, and recognizes the revisited places using the stored place signatures. With SmartDC, the mobile phone can always notify/share the change of places to internal and third-party applications.

Fig. 3 shows the overall architecture of SmartDC. The system comprises mobility learner, mobility predictor, and adaptive duty cycling. In the context of this work, location is a room-level place. In the mobility learner, we use GSM, Wi-Fi, and GPS, which are the common components in current mobile phones. We define three levels of sensing according to the sensor's energy consumption. Initially, SmartDC uses a fixed time interval (e.g., 2 minutes) to collect a user's mobility data. Based on the collected mobility data, the mobility predictor predicts residence time at locations and rewards for each location sensing. Finally, adaptive duty cycling makes use of a dynamic programming technique to determine a sensing schedule. With the predicted sensing schedule, the system activates Wi-Fi sensing to monitor the change of locations. When an exception occurs the system switches to a sensing mode with a fixed interval for mobility learning. Consequently, SmartDC simultaneously performs learning and prediction to minimize energy consumption with robust location monitoring.

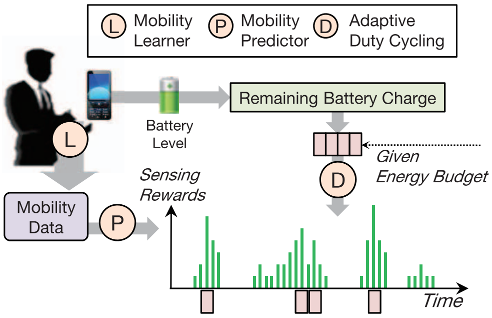


Fig. 4. The conceptual problem. The sensing rewards are derived using residence time and sensing cost from the mobility history. The goal is to allocate energy budget to maximize the rewards within given energy budget.

4.1 Problem Definition

The key problem in location monitoring is choosing an optimal location-sensing interval. We formulated the location-sensing policy as a MDP. An MDP is a stochastic process that contains a 4-tuple $\langle \mathbb{S}, \mathbb{A}, P, \mathbb{R} \rangle$. The finite set of states in which $\mathbb{S} = \{l_1, \ldots, l_m\}$ is a set of meaningful places that were previously visited by the user. Action set $\mathbb{A}$ is a set of actions taken on states, i.e., the length of time before the next location sensing. $P_{ij}(a)$ is the transition probability from state l_i to l_j, when action a is taken. $\mathbb{R} = \{R_x(a_x), R_{x+1}(a_{x+1}), \ldots\}$ is an expected reward at the transition between the states for the action taken. The reward is location-monitoring accuracy, and our goal is to maximize the cumulative function of rewards, defined as follows:

$$\text{maximize} \sum_{x=0}^{\infty} R_x(a_x) \text{ subject to } C \leq E, \tag{1}$$

where C is the total consumed energy for the location monitoring process and E is a given energy constraint. The solution for this problem involves designing an optimal policy π that will be explained in Section 4.4. Fig. 4 illustrates the defined problem.

4.2 Mobility Learner

The role of mobility learner is to collect individual mobility history without impending users. Although the concept of mobility learner is similar to the previous work [9], [16], we designed sensing levels to minimize the usage of power-intensive sensors. The basic idea is that fine-grained sensing is activated only if coarse-grained sensing fails to obtain accurate information.

The first level uses GSM to obtain the location area code (LAC)[1] to detect exceptions within the predicted sensing schedule. The first level continuously monitors the LAC with minor energy consumption, since a mobile phone basically updates the LAC for voice communication. The system does not activate the second level until the next sensing time, if the observed LAC follows a predicted pattern. Otherwise, if the first level detects an exception, the system immediately uses the second level to collect a new pattern of individual mobility.

[1] LAC is a unique number within a cellular radio network. Location area comprises several radio cells.

The second level uses Wi-Fi scanning to recognize a change of places and revisited places. The basic operation is that if a user is stationary, the signal fingerprints of surrounding Wi-Fi APs are relatively similar to each other. We use a scan window to perform multiple scans to tolerate noisy signals [9], [16], [28]. Given a window size w and sampling interval t_w, the Wi-Fi similarity function $\mathcal{S}$ is defined using the Tanimoto Coefficient, which is a widely used technique for measuring the similarity between two vectors [9], [29]:

$$\mathcal{S} = \begin{cases} \text{different (move)}, & \text{if } \frac{\vec{f}_{t_1}\cdot\vec{f}_{t_2}}{\|\vec{f}_{t_1}\|^2+\|\vec{f}_{t_2}\|^2-\vec{f}_{t_1}\cdot\vec{f}_{t_2}} \le \varphi, \\ \text{same (stationary)}, & \text{else} \end{cases}$$

where $\vec{f}_{t_i}$ is the Wi-Fi vector scanned at t_i for duration w, φ is the similarity threshold, and the output is a similarity estimated between 0.0 and 1.0. The movement detector uses $\mathcal{S}(\vec{f}_{t-1}, \vec{f}_t)$ to detect change of places, while place recognition uses $\mathcal{S}(\vec{f}_i, \vec{f}_j)$ to identify a visited place where $\vec{f}_i$ and $\vec{f}_j$ is the Wi-Fi vector scanned at place l_i and l_j, respectively. The system generates meaningful places when it detects the stationary state. When a user revisits a place, the system reuses physical location information without the activating GPS sensor, and aggregates mobility data. The second level also uses wireless communication to obtain location from the WPS provided by Android.

The third level activates GPS to acquire fine location, if the system fails to get accurate location in the level 2. Algorithm 1 presents pseudocode of mobility learner.

Algorithm 1. Three level sensing in mobility learner.
Input: Previous fingerprint $\vec{f}_{t-1}$, predicted pattern $\mathbb{P}$
Output: Current location l_t
Notation: LAC ℓ_t, fingerprint $\vec{f}_t$, movement state s_t, similarity threshold φ, location-accuracy threshold ϕ

```
$\ell_t \leftarrow \text{getLAC}()$                          #Level 1 using GSM
if $\mathbb{P}.\text{contains}(\ell_t)$ then
    return
end if
$\vec{f}_t \leftarrow \text{scanWiFi}()$                        #Level 2 using Wi-Fi
if $\mathcal{S}(\vec{f}_{t-1}, \vec{f}_t) \le \varphi$ then
    $s_t \leftarrow \text{Move}$
else
    $s_t \leftarrow \text{Stationary}$
end if
$l_t \leftarrow \text{getCoarseLoction}()$                   #Level 2 using WPS
$l_t{}' \leftarrow \text{makeLocation}(s_t, l_t, \vec{f}_t)$          #Place Recognition
if $l_t{}'.\text{getAccuracy}() \le \phi$ then
    return $l_t{}'$
end if
$l_t{}' \leftarrow \text{getFineLocation}()$                 #Level 3 using GPS
return $l_t{}'$
```

4.3 Mobility Predictor

The role of mobility predictor is to estimate the reward function R based on action set $\mathbf{A}$ and transition probabilities P. In other words, when a user visits a place, the system makes prediction on stay duration in a place.

We use two types of location predictor: Markov predictor [22], [23] as a location-dependent predictor, and nonlinear time series [24] analysis as a location-independent predictor. Previous works found that such predictors were efficient in accuracy and memory usage aspects in predictions for human mobility [4], [24]. Our scheme contributes to the prediction of human movements with room-level accuracy, while previous works [2], [3], [4] used associated cell-tower or AP for a user's location.

We briefly describe the Markov predictor and nonlinear time series analysis. The Markov predictor is widely used as a location-dependent model [4], [22], [23]. The order-k Markov predictor extracts historical patterns by matching recent k location sequences. The basic assumption is that people tend to spend similar residence time at the same places. Consider a user's movement history $H = (l_1, t_1^a, s_1), \ldots, (l_n, t_n^a, s_n)$, in which t_i^a is the arrival time and s_i is the stay duration at location l_i. From H, we extract the location history $L = l_1, l_2, \ldots, l_n$, and the finite set of m visited places. From L, the recent k location context is $L(n-k+1, n) = l_{n-k+1}, \ldots, l_n$. Then, the order-$k$ (or $O(k)$) Markov predictor generates the reward function R at the current location l_i, defined as follows:

$$R_i = \sum_{j=1}^{m} (w_{ij} P_{ij}),$$

where $w_{ij} = \Pr(l_{x+1} = l_j \mid L(x-k+1, x) = l_{i-k+1}, \ldots, l_{i-1}, l_i)$ and P_{ij} is the form of a discrete histogram distribution from the stay-duration set $\{s_x \mid L(x-k+1, x+1) = l_{i-k+1}, \ldots, l_{i-1}, l_i, l_j\}$. Indeed, w_{ij} is the weight of visit frequency from location l_i to l_j, and P_{ij} is the residence-time distribution from l_i to l_j. Here, $\sum_{j=1}^{m} w_{ij} = 1$ and $\int_0^\infty R_i(t)dt = \int_0^\infty P_{ij}(t)dt = 0$ or 1.

A location-independent model uses temporal features without location information. The basic assumption is that people tend to spend similar staying time at similar times of a day. We used a variant of NextPlace [24] as a location-independent model. NextPlace uses nonlinear time series analysis to extract similar patterns from historical data. From H, NextPlace extracts arrival time history $A = t_1^a, t_2^a, \ldots, t_n^a$ and current context $c = A(n-k+1, n) = t_{n-k+1}^a, \ldots, t_{n-1}^a, t_n^a$. Then, P is a form of discrete histogram distribution from the stay-duration set $\{s_x \mid f(A(x-k+1, x), c) < \sigma\}$, where f is the similarity function of two vectors and σ is a given threshold. We use the maximum norm for f and $\sigma = 10\%$ of deviation, as suggested in [24]. The model considers the similarity between sequences of arrival time without location information.

We apply the time-aided scheme and the fallback mechanism 1) to filter out redundant information from extensive data, and 2) to compensate the none-prediction (i.e., $\int_0^\infty R_i(t)dt = 0$). The time-aided scheme uses a pair state of location and arrival time (l_x, t_x^a) with quantized time buckets (e.g., hour interval) [4]. The assumption of this scheme is that the stay duration is dependent on the arrival time of place. To generate residence-time distribution, the time-aided Markov predictor uses the stay-duration set $\{s_x \mid L(x-k+1, x+1) = l_{i-k+1}, \ldots, l_{i-1}, l_i, l_j$ and $A(x-k+1, x) = c\}$, where A is the history of arrival time and c is recent k arrival time sequences. The fallback mechanism uses the low-order predictor if the high-order predictor has no prediction result. For example, the Markov model uses $O(k-1)$ predictor if $\int_0^\infty R_i(t)dt = 0$ in $O(k)$ predictor.

Instead of choosing the location with the highest probability, we extract all locations that have the transit probability to determine an efficient sensing schedule. To cover the error from the mobility learning, we add a Gaussian noise to the generated R. The noise distribution is empirically found in real trace as described in Section 5.3. Then, discrete action set in one-minute interval is defined as $\mathbf{A}' = \{a_i \mid 0 \leq i \leq n, R(a_i) > 0\}$, where n is the number of quantized actions that have positive probability in R.

4.4 Adaptive Duty Cycling

The goal of adaptive duty cycling is to maximize the accuracy of monitoring mobility, and to optimize a sensing policy on diverse smartphone usage with a given energy budget.

Randomness is inherent in human movements although humans tend to move with reasonably small variation. In other words, a user does not always follow the previously observed movement patterns. Thus, the accuracy of the prediction-based adaptive duty cycling may decrease due to the randomness of human behavior. To overcome this problem, we split a given energy budget E into two parts: energy for prediction E_{PRD} (i.e., the case of following patterns in mobility history), and energy for exception E_{EXP} (i.e., the case of moving with a new pattern). To split E, we use potential predictability in individual mobility. The system automatically measures predictability by comparing the current pattern with the historical data. For example, potential predictability is 0.8 if a user follows the previously observed pattern eight times out of a total of 10 visit counts. Then, the exception ratio is 0.2 and the system allocates 20 percent of given energy for monitoring exception.

We first present the sensing policy using E_{PRD}. Let $V_x(\mathbf{A}', E)$ be an optimal sensing schedule at state $(\mathbf{A}', E)$, which is also the maximal sensing rewards at the current location l_x within a given energy budget E and action set $\mathbf{A}'$. When the system detects entrance at a certain place l_x, the mobility predictor estimates the residence time distribution P that is used for reward function R_x. The sensing cost c_i is an estimated energy consumption at a_i, depending on the stored accuracy of location information. If the accuracy is worse than location-accuracy threshold ϕ, the cost is the sum of sensing levels 2 and 3. Otherwise, the cost is the energy consumption of level 2. Our goal is to maximize the cumulative function of rewards, defined as follows:

$$\text{maximize} \sum_{i=0}^{|\mathbf{A}'|} R_x(a_i) \text{ subject to } C_{PRD} \leq E_{PRD},$$

where C_{PRD} is the total consumed energy for the location monitoring process and E_{PRD} is a given energy constraint. The solution for this problem involves designing an optimal policy π and computing the value function $V(\cdot)$, which is expressed as

$$\pi = \arg\max_a \left\{ \sum_{i=0}^{|\mathbf{A}'|} P_x(a_i)(R_x(a_i) + V_x(a_{i+1}, e - c_i)) \right\},$$

$$V_x(\mathbf{A}', E_{PRD}) = \sum_{i=0}^{|\mathbf{A}'|} P_x(\pi)(R_x(a_i) + V_x(a_{i+1}, e - c_i)),$$

where e is the remaining energy budget. In words, optimal sensing moments are allocated to maximize the rewards of sensing within a given energy budget. Algorithm 2 presents the pseudocode of adaptive duty cycling using the dynamic programming technique. The output of dynamic programming is the set of sensing moment, and inputs are the current time, energy budget, and generated reward function.

Algorithm 2. Allocation policy of energy for prediction.
Input: Sensing moment i, remaining energy budget e
Output: Set of sensing moments derived from $V(\cdot)$

```
if $e < 0$ then                         # Boundary condition 1
    return invalid
end if
if $i > |\mathbf{A}'|$ then                     # Boundary condition 2
    return 0
end if
if $V_t(a_i, e)$ is valid then               # Memoization
    return $V_t(a_i, e)$
end if
    $V_t(a_i, e) \leftarrow$ invalid
for $k$ is $i+1$ to $|\mathbf{A}'|$ do
    $V_t(a_i, e) = \max(V_t(a_i, e), R_t(a_k) + V_t(a_{k+1}, e - c_k))$
end for
return $V_t(a_i, e)$
```

The system allocates E_{EXP} to minimize the longest blank-term (i.e., continuous duration without location sensing). Let $B = \{b_0, b_1, \ldots, b_k\}$ be a set of blank terms, and b_i the length of the ith blank term in minutes. Given j location sensing opportunities, we allocate it to B to minimize the longest blank term. For example, we assume that b_i is an 8-length blank term (oooooooo). Then it can be divided into three shorter blank terms (b_i, b_{i+1}, b_{i+2}) if two location sensing is allocated to b_i (ooxooxoo). The goal is to minimize the longest length of blank term, defined as

$$\text{minimize} \max_{0 \leq i \leq k} b_i \text{ subject to } C_{EXP} \leq E_{EXP}, \tag{7}$$

where C_{EXP} is the total consumed energy for dividing all blank terms. Let r be a maximum length in B, we find the optimal r by using a binary search between 0 to $\max_{0 \leq i \leq k} b_i$. When r is fixed, the required energy C_{EXP} for reducing all elements in B to less than r can be calculated in $O(k)$. Then, we decrease r if C_{EXP} is smaller than E_{EXP}, otherwise we increase r. Algorithm 3 presents the pseudocode of the allocation policy.

Algorithm 3. Allocation policy of energy for exception.
Input: Set of blank term B, energy budget for exception E_{EXP}
Output: Optimal length of blank term r

```
$low = 0, high = \max(B), r = \max(B)$
while $low < high$ do
    $mid = low + \frac{high - low}{2}$              #Fix blank term
    $C_{EXP} = \sum_{i=0}^{|B|} \frac{b_i}{mid+1}$          #Required energy
    if $C_{EXP} > E_{EXP}$ then              #Energy exceed $E_{EXP}$
```

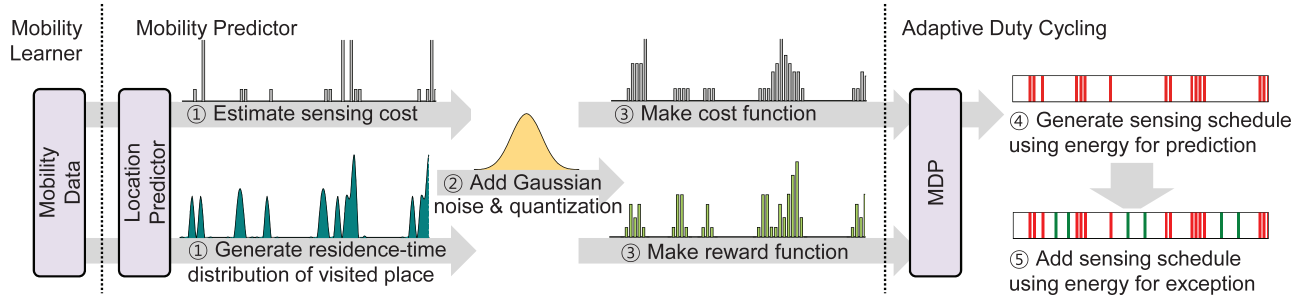

Mobility Learner
Mobility Predictor
Adaptive Duty Cycling
Mobility Data
Location Predictor
① Estimate sensing cost
① Generate residence-time distribution of visited place
② Add Gaussian noise & quantization
③ Make cost function
③ Make reward function
MDP
④ Generate sensing schedule using energy for prediction
⑤ Add sensing schedule using energy for exception

Fig. 5. The process of prediction-based adaptive duty cycling. 1) Location predictor generates residence-time distribution and sensing cost using collected data. 2) Gaussian noise is derived from the empirical error of mobility learner. 3) The system makes quantized rewards and cost functions in minutes. 4) MDP outputs predicted sensing schedule. 5) Energy for exception is used to minimize longest blank term.

```
            $low = mid + 1$
        else
            $high = mid$
            $r = \min(r, mid)$
        end if
    end while
    return $r$
```

Fig. 5 illustrates the overall process of prediction-based adaptive duty cycling.

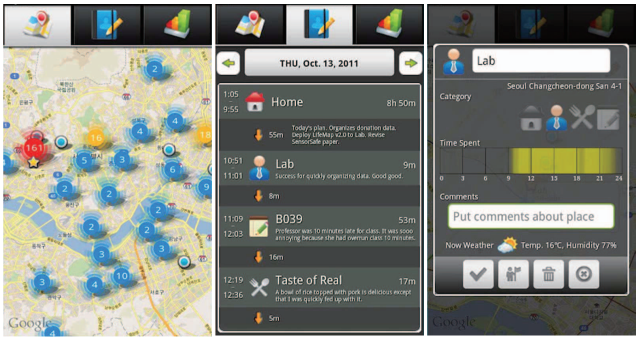
5 Evaluation

We evaluate SmartDC in three aspects: mobility learning, mobility prediction, and adaptive duty cycling. We mainly focus on the performance of prediction-based adaptive duty cycling, which is our major objective.

5.1 Implementation

To evaluate the proposed scheme, we implemented SmartDC on the Android SDK 2.1 running on commercial mobile phones equipped with GPS, GSM/CDMA, and Wi-Fi. Our implementation specifically focused on tracing a user's mobility based on intuitive UI design. Fig. 6 shows a screen capture of the UI running on an HTC Desire. The system indicates a user's meaningful places on the Google map and the list to confirm/modify the place signatures, such as residence time, place name, and mobility history.

In our experiment, we omitted the analysis of some of the parameters for the learning scheme, since our mobility learner uses a scheme similar to that of previous studies. The time interval t_w is set to two minutes, as suggested by Ma et al. [16]. The Wi-Fi scanning intervals and window size are 10 and 30 seconds, respectively, and the similarity threshold φ of the Wi-Fi vector is set to 0.7 [9]. The accuracy threshold ϕ for the WPS is set to 500 meters [15]. The GPS is activated for 30 seconds for single positioning, which is common in GPS usage.



THU, Oct. 13, 2011

1:05 – 9:55 Home 8h 50m

55m Today's plan. Organizes donation data. Deploy LifeMap v2.0 to Lab. Revise SensorSafe paper.

10:51 – 11:01 Lab 9m
Success for quickly organizing data. Good good.

8m

11:09 – 12:03 B039 53m
Professor was 10 minutes late for class. It was sooo annoying because she had overrun class 10 minutes.

16m

12:19 – 12:36 Taste of Real 17m
A bowl of rice topped with pork is delicious except that I was quickly fed up with it.

5m

Lab

Seoul Changcheon-dong San 4-1

Category

Time Spent

Comments

Put comments about place

Now Weather Temp. 16℃, Humidity 77%

(a) (b) (c)

Fig. 6. User interface of SmartDC. The application displays a user's meaningful places on (a) the Google map, and (b) the list. It also visualizes (c) place information, including place name and mobility history. The application called LifeMap is available in android market.

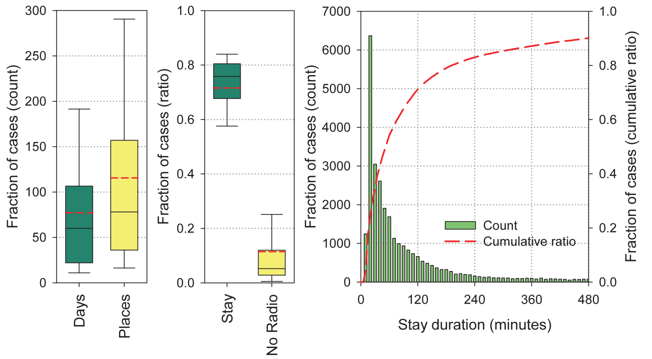
To reduce computation overheads in adaptive duty cycling, we 1) converted the float value to an integer value with 10^{-3} precision, 2) used discrete time intervals in minutes, and 3) scaled down the energy budget, dividing it by the energy consumption of Wi-Fi scanning, which is the minimum cost in our scheme.

We set the energy budget E and the sensing cost c as follows: The maximum energy budget is $(1{,}400\ \mathrm{mA} \times 3.7\ \mathrm{V} - 34.5\ \mathrm{mW}) \times 3,600\ \mathrm{s} = 18.5\ \mathrm{kJ}$, which is the available battery capacity excluding the energy cost of idle state. If the battery level and energy constraints are 30 and 10 percent, the allowed energy budget is $18.5\ \mathrm{kJ} \times 0.3 \times 0.1 = 555\ \mathrm{J}$. Based on the energy profile in Section 2, the level 2 sensing cost is the energy consumption of Wi-Fi scanning: $114.5\ \mathrm{mW} \times 30\ \mathrm{s} = 3.5\ \mathrm{J}$. The level 3 sensing consumes $3.5\ \mathrm{J} + 440.8\ \mathrm{mW} \times 30\ \mathrm{s} = 16.7\ \mathrm{J}$: the energy of level-2 sensing and reading GPS for 30 seconds.

5.2 Data Collection

We collected real traces from 57 users (26 students and 31 office workers) over four weeks in Seoul, Korea. SmartDC was running as a background service to automatically collect the user's mobility and to trace sensor usage time. Participants installed SmartDC on their primary phones. To collect the ground truth, the participants explicitly labeled the place names and kept a diary of places they had visited with the entrance and departure times. Fig. 7a shows the description of data set. In the data traces, participants visited altogether 6,600 meaningful places with 30,700 stays, and spent 74,000 hours in places. On average, a participant collected traces for 77.1 days and visited 155.5 places. A subset of data set is available in CrawDad research communities [30].

We measured the entropy of collected data traces. The proposed system is ineffective when a user is located in regions that do not have Wi-Fi coverage because the key scheme of our movement detector uses Wi-Fi scanning. Thus, we defined *ineffectivity rate* as the ratio of time spent by a user in regions without Wi-Fi coverage to the total



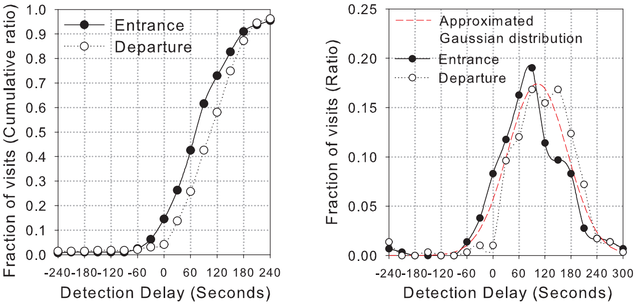
(a) (b) (c)

Fig. 7. The description of data set: (a) collection days and number of places, (b) the ratio of residence time and no Wi-Fi coverage time, and (c) the distribution of stay duration at each visit.

spent time. The collected traces show that the ineffectivity rate is trivial. The median of ineffectivity rate is about 5.2 percent, as illustrated in Fig. 7b. In other words, participants spent about 5.2 percent of their time in non-Wi-Fi regions. They spent 76 percent of their time in places for staying behavior and 71 percent of staying behavior is less than 2 hours, as shown in Figs. 7b and 7c. This means that the system could reduce about 76 percent of energy consumption for location sensing if it could accurately predict the residence time at each place.

5.3 Mobility Learning

We evaluate the performance of mobility learning in two ways: place detection and energy consumption. We briefly described the performance of mobility learner because the component uses simple fixed time interval (The detail is available in [1]). We defined the accuracy of place learning as a measure of meaningful places correctly discovered by the system. The accuracy is 0.76 for the number of places and 0.93 for the number of visits. The result indicates that collected data are sufficient to use as a mobility history for prediction. In addition, 80 percent of the visits are detected within a 139-second delay in entering the places, and a 161-second delay in departing the places, as illustrated in Fig. 8a. Fig. 8b indicates that the error in mobility learning is represented as an approximated Gaussian distribution. We applied this distribution as a noise to mobility predictor to cover the error in learning.



(a) (b)

Fig. 8. Detection delay of departure time and entrance time in (a) cumulative probability, and (b) probability distribution.

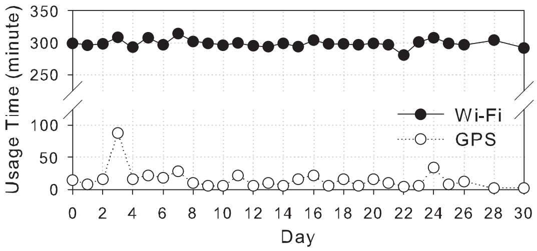


Fig. 9. The daily usage time of sensors in mobility learner.

We present each sensor's energy consumption for mobility learning. Fig. 9 shows the daily usage time of sensors. The usage time of GSM is negligible (i.e., less than 20 seconds), because the computation time to update an LAC takes only a few milliseconds. GPS is typically activated in transit, since we turn on the GPS only if the system cannot obtain accurate location information from both the WPS and historical mobility. The active time of Wi-Fi is uniformly distributed due to the fixed time interval. The average usage times of Wi-Fi and GPS are 300 ± 12 minutes and 16 ± 7 minutes. The GPS usage time is trivial because WPS provides accurate location information in Seoul, Korea. Such usage time would be increased if a user lives in regions that do not provide sufficient WPS data. Although WPS has poor coverage, the GPS usage time would be gradually decreased because the proposed system reuses location information in historical data by matching Wi-Fi fingerprints of places. The average energy consumption is $1,831 \pm 228$ J, which would reduce battery lifetime by $12 \pm 4\%$ in real deployments. In the following section, we present the potential predictability of human mobility, and the effectiveness of prediction-based adaptive duty cycling to minimize such energy consumption.

5.4 Potential Predictability

We investigate the upper bound of predictability in individual mobility patterns to estimate the optimal performance of mobility prediction. To measure maximal probability, we consider that the current pattern is predictable if it is a previously observed one, since we use a history-based predictor. We defined two metrics: *revisit ratio* U, and *predictable movement ratio* U_m. U indicates the maximal predictability of location prediction, which is defined as

$$U = \frac{\text{No. of revisits}}{\text{No. of visits}}.$$

Since our system determines the efficient sensing schedule, predictable residence time at a revisited place is a more important factor than the next location. Thus, we further measured each user's predictable movement ratio defined as

$$U_m = \frac{\text{Sum. of previously observed residence time}}{\text{Sum. of residence time}}.$$

U_m indicates the maximal predictability of mobility prediction. We consider current residence time to be predictable if the difference to historical data is within a ± 3 deviation (68-second) of error in mobility learner.

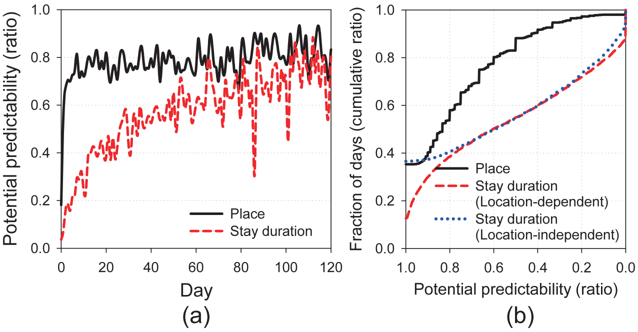


Fig. 10. Potential predictability in participants' mobility pattern. Higher probability means that a user follows previously observed mobility patterns. Prediction about location U contains more predictable regularity than prediction about movement behavior U_m.

Fig. 10 represents the daily predictability of all participants. Higher probability means that higher predictability is inherent in the individual mobility pattern. The result indicates that residence time prediction U_m is less predictable than location prediction U. U rapidly increased to $78 \pm 8\%$ after one week, but U_m slowly increased and required about three month to reach around $72 \pm 9\%$ predictability, as illustrated in Fig. 10a. In other words, humans tend to revisit frequently visited places, but their staying behavior contains relatively more randomness.

Predictability about movement behavior U_m shows that 50 percent of days contain more than 56 percent predictability, as shown in Fig. 10b. Both the location-dependent and location-independent predictors show similar predictability. This indicates that the system could, at most, reduce 56 percent of energy for location sensing if the predictor algorithm is 100 percent accurate. The predictability, however, did not change significantly, even though we considered only weekday mobility.

To understand potential regularity with temporal features, we segmented each week into $7 \text{ days} \times 24 \text{ hours} \times 60 \text{ minutes} = 10,080 \text{ minute intervals}$. We measured the number of visits at the most visited locations in every minute. Fig. 11 shows the average predictability of all participants. During night time (0-7 am), high predictability is shown as $64 \pm 4\%$, while the predictability of mealtime (12-1 pm and 6-7 pm) shows low predictability as $39 \pm 4\%$. The result reveals the behavior tendency of human; that is, human tends to return home at evening and spend night time at home. Thus, the system may reduce relatively more energy usage in night time than work hour.

In the remainder of this paper, we omit the analysis on U. Instead, we focus on U_m because our goal is to determine a sensing schedule that is strictly connected to prediction about residence time.

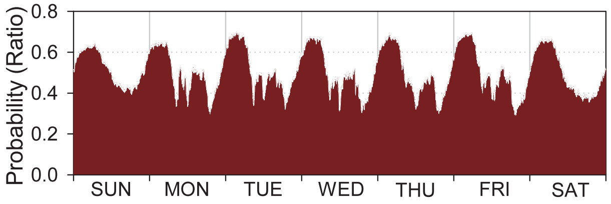


Fig. 11. The probability of most visited location during the corresponding minute-long period. 0.66 at Tuesday 1 am means that participants are located 66 percent of days in his/her most visited location at Tuesday 1 am.

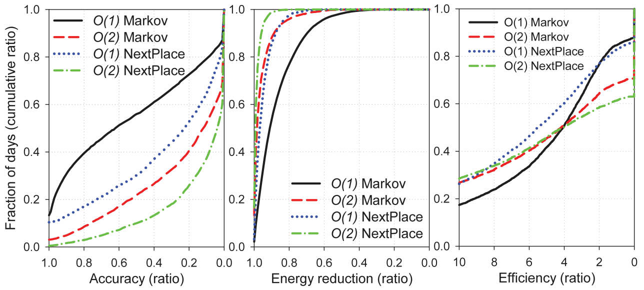


Fig. 12. The performance of Markov and NextPlace under the infinite energy budget. The leftmost curves indicate better performance in accuracy, energy reduction, and efficiency.

5.5 Mobility Prediction

We evaluated the performance of location-dependent and location-independent predictors with time-aided scheme and fallback mechanism. Here, we set the energy budget infinite to eliminate the effect of energy constraints. We defined three metrics: *accuracy*, *energy reduction*, and *efficiency*. The predictor generates three outcomes: 1) The predicted schedule correctly senses the change of location within the error of mobility learning (*correct prediction*); 2) The predicted schedule misses the moment of location change beyond the error range of mobility learning (*incorrect prediction*); and 3) The predicted schedule is none, since the current pattern has not been seen before (*none-prediction*). The accuracy is defined as: $\frac{(1)}{(1)+(2)+(3)}$, and the energy reduction is defined as

$$\text{energy reduction} = 1 - \frac{\text{No. of sensing with prediction}}{\text{No. of sensing in fixed interval}}.$$

Since we assume that energy budget is infinite, accuracy indicates the maximal accuracy of predictors, and the energy reduction indicates the number of observed patterns using each predictor. Energy reduction converges to zero as the number of observed pattern increases. Finally, efficiency is defined as

$$\text{efficiency} = \frac{\text{accuracy}}{1 - \text{energy reduction}}.$$

The optimal way derives high accuracy and efficiency, which means the predictor precisely senses a user's mobility with a minimum number of sensings.

Under the condition of an infinite energy budget, order-1 predictor uses all observed patterns at current location, while high-order predictors use a sequential pattern of visited locations. We found that, counterintuitively, using more context (i.e., sequential patterns) is not efficient in predictions about residence time at current place, as illustrated in Fig. 12. Although high-order predictors used less energy than order-1 predictors, they failed to predict many cases, and their efficiency is also not significantly outperformed. The result indicates that context about previously visited locations (e.g., workplace or restaurant) does not help to predict residence time at current places (e.g., home).

The none-prediction case also causes performance deterioration of high-order predictors that contain about

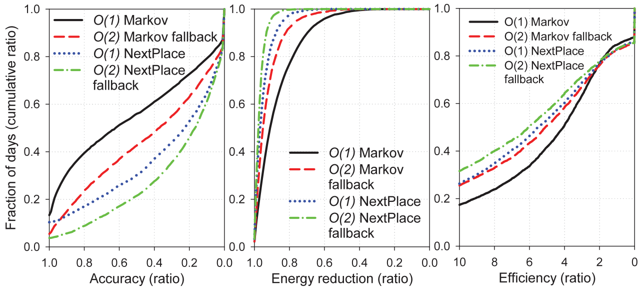


Fig. 13. The performance of Markov and NextPlace predictors with fallback under the infinite energy budget.

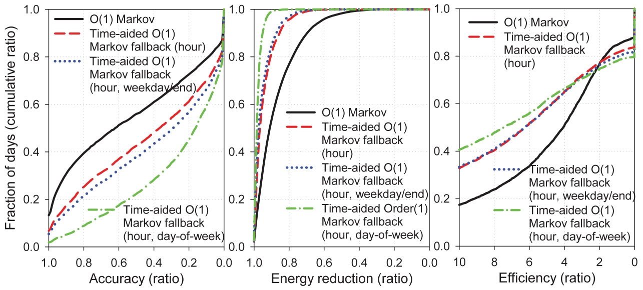


Fig. 15. The performance of time-aided Markov under the infinite energy budget.

35 percent none-prediction cases. To cover the effect of none-prediction, we applied a fallback mechanism, which uses the result of the low-order predictor if the high-order predictor has no prediction [4]. Indeed, fallback improved the accuracy and efficiency of high-order predictors as illustrated in Fig. 13. The overall efficiency of high-order predictors with fallback is higher than order-1 predictors, but the accuracy of order-1 predictors significantly outperforms high-order predictors.

We investigated the accuracy of predictors according to the type-of-places to explore the reason of incorrect prediction. We defined three type-of-place according to the number of visits: *Top-2 places* are the most visited two locations (usually home and workplace); *major places* are locations where a user visits more than three times; the rest places are *minor places*. We found that location-dependent predictor is robust in top 2 and major places, and location-independent predictor outperforms location-dependent one in minor places, as shown in Fig. 14. In case of top 2 and major places, the location-dependent model correctly predicts 82 percent of cases, while the location-independent model generates 1.6 times higher incorrect predictions. However, the location-independent model correctly predicts about 5,200 cases of stay behavior in minor places, which were missed in the Markov model due to the small number of visits. In other words, the stay duration in minor places is strongly correlated with arrival time, but the stay duration in the top-2 places is dependent on the place. Thus, we use location-independent predictors as a fallback of location-dependent one to compensate the none-prediction cases in minor places.

We applied temporal state to predictors to investigate the correlation between human movement behavior and temporal features. We quantized time in 24 (24 hour), 48 (24 hour × weekday/weekend), or 168 (24 hour × 7) buckets. Then, the state of predictor forms location and time pairs. The fallback mechanism reduces a two-dimensional state (location, time) to one dimensional (location) if time-aided Markov fails to predict. The assumption is that humans tend to move with daily or weekly regularity. Fig. 15 shows the performance of time-aided Markov predictors. The accuracy of $O(1)$ Markov with 168-buckets was far from $O(1)$ due to the insufficient number of observed patterns (i.e., 49 percent none-prediction cases). Despite the smaller number of sensings, time-aided predictor's efficiency was clearly higher than $O(1)$. It meant that a time-aided predictor accurately filtered the redundant patterns for prediction about residence time. The use of 24 buckets outperforms 48 buckets case in accuracy. The fallback slightly increases the accuracy of a time-aided Markov predictor.

Fig. 16 shows the performance of the best variants of predictors. The accuracy of $O(1)$ Markov significantly outperforms other predictors because of the numerous samples. The time-aided Markov shows high efficiency because it accurately filtered out redundant information using temporal features. Considering the accuracy and efficiency of predictors, we chose $O(1)$ Markov as the accurate predictor and time-aided $O(1)$ Markov with 24 buckets as the high-efficiency predictor. In the following, we evaluate the prediction-based adaptive duty cycling with energy constraints.

5.6 Adaptive Duty Cycling

In this section, we investigate the cost and performance of prediction-based adaptive duty cycling. We analyze the sensitivity of parameters and the effectiveness of the prediction method. We then show the energy consumption and accuracy of the overall system.

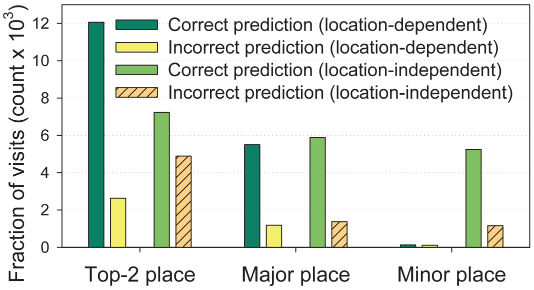


Fig. 14. Amount of correct prediction and incorrect prediction according to the type-of-places.

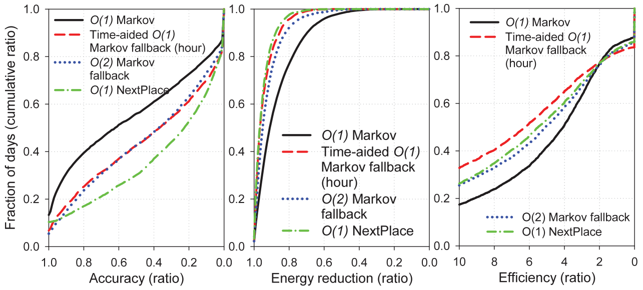


Fig. 16. The performance of best variants of predictors under the infinite energy budget.

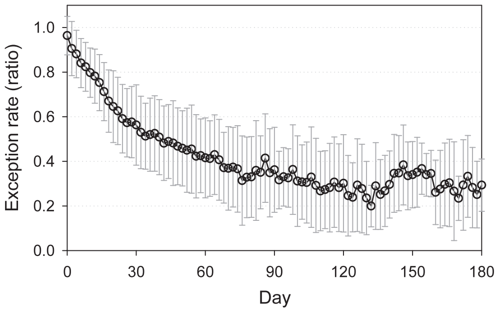


Fig. 17. The average exception ratio of all participants. The exception rate is gradually decreased as the mobility learning is continuous.

The proposed system splits the energy budget into two parts: prediction and exception. The system automatically derives a personalized exception ratio using the average of daily predictability U_m during two recent weeks. Fig. 17 illustrates the measured exception ratio of all participants. The exception rate gradually decreases as the mobility data increases. Since the proposed system automatically learns a user's mobility, the exception ratio will be adaptively changed if a user changes her mobility pattern (e.g., change jobs, or vacation).

We ran the emulation to evaluate the effectiveness of the adaptive duty cycling with the following assumptions: 1) People use their smartphone for 28 hours with a single battery charge, which is the average lifetime of typical users' smartphones [27]; and 2) the system correctly recognizes revisited places. Fig. 18 describes two steps in the emulation and three possible cases:

Case *PRD*. The predicted sensing schedule contains the ground truth of a location change moment. Detection delay is calculated using the first sensing moment after the ground truth.

Case *PRD-LRN*. The system uses a fixed interval after the predicted sensing schedule is finished.

Case *LRN*. The system uses a fixed interval because a place does not have mobility history.

We use the following notations to evaluate the performance of the prediction-based adaptive duty cycling:

$\mathcal{N}$: Number of sensing events using Wi-Fi (cost).

$\mathcal{D}$: Place detection delay in seconds (accuracy).

We do not explicitly state the number of GPS readings because it is dependent on the location accuracy obtained in sensing level 2.

We first present the computation time of core functions in our algorithm. The computational complexity of adaptive duty cycling is $O(TE_{PRD} + B\log_2(E_{EXP}))$, where T is the maximal time in minutes among observed residence time, E_{PRD} is the energy budget for the prediction, E_{EXP} is the energy budget for the exception, and B is the number of blank terms. The complexity is equal to the cost of filling a $T \times E_{PRD}$ array for dynamic programming and the cost of finding a minimum blank term using a binary search. Here, the complexity of finding entries of a Markov scheme is $O(1)$ because we use a hash-table for it. Fig. 19 presents the computation time on a Samsung Galaxy. The computation time is less than 100 milliseconds in about 95 percent visits. Such computation time is sufficient to process in real time, without impeding users.

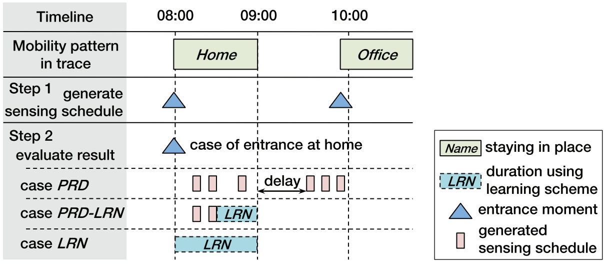


Fig. 18. Two steps and three outputs in the emulation.

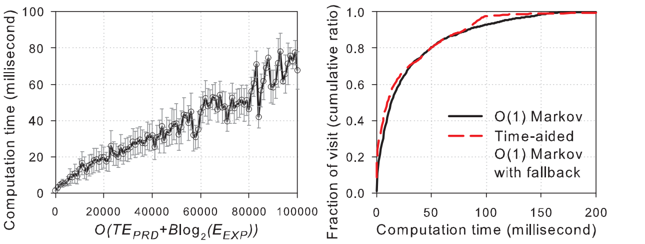


Fig. 19. The computation time for processing.

The energy budget constraint E is a major factor in determining an optimal sensing schedule. SmartDC limits the energy usage to less than the given energy budget. Thus, a smaller E may derive more missed places due to the limited sensing opportunity. The efficient usage of energy involves maximizing the number of detected location changes within smaller sensing delays. Fig. 20 shows the results according to various energy budgets. We considered the cases of *PRD* and *PRD-LRN*, but not the *LRN* case, since the cost and performance of the *LRN* case is equal to the learning scheme. Intuitively, a larger energy budget derives smaller detection delays and more sensing events. In the case of a 10 percent energy constraint, adaptive duty cycling consumes only $41 \pm 7\%$ energy of the case using fixed interval, and still detects 80 percent of location changes within an additional 80 ± 12 seconds. In addition, time-aided $O(1)$ Markov consumes 72 percent energy of the $O(1)$ Markov, and the delay of 80 percent of the location change is negligible (i.e., about 15 seconds). The result reveals that using temporal features is a more efficient way than using all historical data for predictions.

The major reason for energy saving is that 1) people tend to follow the observed pattern in mobility history, and

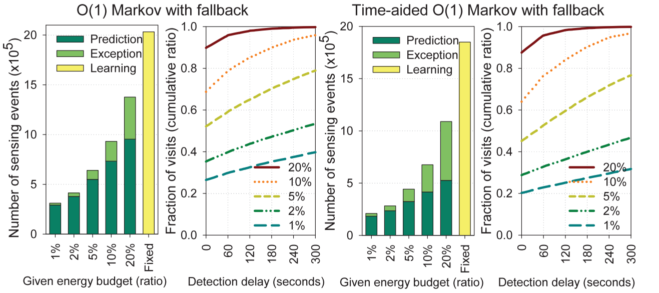


Fig. 20. The cost and performance according to various energy budgets. The detection delay is additional delay compared to learning scheme. Zero-second delay includes the cases that show less delay than for a fixed interval of learning scheme (i.e., 2 minute).

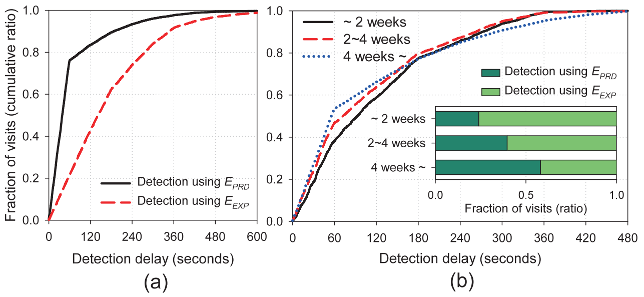


Fig. 21. The delay and detected type of all participants in (a) CDF and (b) according to collection period. The detection delay derived from E_{PRD} is significantly shorter than the delay derived from E_{EXP}.

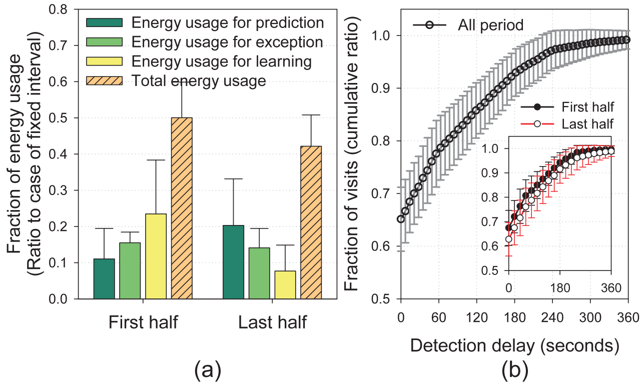


Fig. 22. (a) The energy usage ratio to fixed time interval and (b) detection delay. Zero-second delay includes the *LRN* cases.

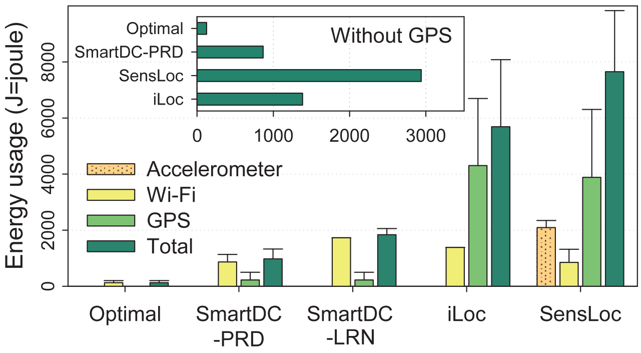


Fig. 24. Daily energy usage. To estimate the energy usage of related systems, we utilized our energy profile and sensing policy in [9], [16]. The detailed sensing policy is presented in Table 1. The optimal case uses a manual sensing by a user. Inset indicates the daily energy usage without GPS.

2) our system successfully allocates a given energy budget based on individual mobility pattern. Fig. 21 shows the correlation between detection delay and detected type (i.e., detection using E_{PRD} or E_{EXP}). The detection using E_{PRD} derives a smaller delay (i.e., the leftmost curve in Fig. 21a). In other words, the system detects a user's movements within a small delay when a user follows his/her previous pattern. The result implicitly reveals that the system also covers the randomness of a user's movements by using E_{EXP}. The proposed system gradually decreases detection delay as the collection period increases, as illustrated in Fig. 21b. Indeed, the ratio of detection using E_{PRD} increases as the collection period increases.

Finally, we evaluated the performance of the overall system including cases *PRD*, *PRD-LRN*, and *LRN*. Fig. 22 presents the energy usage ratio and detection delay. Compared to a fixed time interval, the proposed system consumes $46 \pm 6\%$ energy: $30 \pm 5\%$ of energy for adaptive duty cycling and $15 \pm 11\%$ of energy for learning. The system detects 90 percent of location change shows less than about 156-seconds delay. Fig. 22a indicates that the system reduces energy consumption as the collection period increases because the system used prediction-based adaptive duty cycling instead of fixed time interval. Despite less energy consumption, the detection delay in the last-half period is similar to delay in the first-half period, as illustrated in inset of Fig. 22b. This reveals that our system successfully detected a user's mobility using regular patterns without additional delay. Consequently, the proposed system is robust to monitor user's mobility accurately, regardless of the amount of learning data.

Fig. 23 shows the sensors' activate time of a day among collected data. Compared to the results from learning schemes, the sensor active time is reduced based on the predicted time of a location change. The sensing schedule during nighttime specifically reveals that the proposed system successfully predicts residence time at the revisited places: it allocates E_{PRD} around the office-going hour while E_{EXP} covers the nighttime (10 pm to 6 am). The average usage time of Wi-Fi is 180 ± 55 minutes. The average energy consumption for one day is 974 ± 352 J, which is 47 percent less energy consumption than learning schemes without prediction.

Fig. 24 shows the comparison of energy usage among the related systems. In sensor usage aspects, the proposed system consumes 81 percent less energy than the periodic sensing schemes without accelerometer [16], and 87 percent less energy than a scheme that employs context-aware sensing [9], while additional detection delay is about 150 seconds. When a user uses a smartphone for 28 hours with single battery charge [27], the expected battery lifetime of the proposed system is 26.4 ± 0.5 hours, which is 6.6 hours longer than the lifetime of related systems (i.e., 19.8 ± 2.1 hours). The major factors for energy saving are

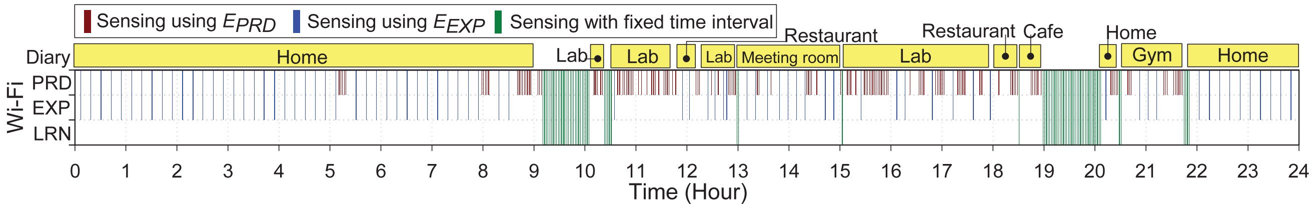


Fig. 23. Sensor active time of prediction-based adaptive duty cycling. The diary indicates the ground truth of mobility.

1) adaptive duty cycling based on mobility prediction, and 2) multiple sensing levels to reuse location information without activating power-intensive sensors. Without energy consumption on GPS for path tracking, our approach still consumes 38 percent less energy than the periodic sensing schemes, and 70 percent less energy than context-aware schemes, as shown in the inset of Fig. 24.

In summary, our approach successfully saves energy for everyday location monitoring, and detects 90 percent of location changes within a 150-second delay, depending on given energy constraints and individual mobility patterns. The effectiveness of such delays depends on the application. This delay is viable in applications that use human mobility patterns, such as social applications, healthcare applications, environment-related applications, epidemics, and urban planning. Applications requiring sensitive delays, such as reminder applications and advertisement services, are minimally affected by such delays. While the proposed system uses shorter time intervals for reducing detection delays, the system still contributes to minimizing energy consumption through prediction-based adaptive duty cycling.

6 Conclusion

In this paper, we proposed SmartDC to solve the energy issue of continuous sensing in real deployments. To the best of our knowledge, we are the first to implement a practical system that simultaneously performs mobility learning and prediction for everyday location monitoring, based on off-the-shelf smartphones. We designed sensing levels and an adaptive duty cycle using MDP with automatically collected mobility data. We evaluated various mobility predictors for prediction of residence time in practice. The experiments show that our approach saves 81 to 87 percent energy over previous work, while its place detection delay is increased by approximately 160 seconds. Such energy saving resulted in about 6-hours-longer lifetime in a day. We believe that our approach can be used as a building block toward expanding the domain of mobile services and to gathering individual human mobility patterns for research purposes. In addition, our prediction-based approach is suitable for a wide range of emerging mobile applications related to the temporal predictability.

Acknowledgments

This paper is an extended version of the authors' previous work published in [1]. This work was supported by the National Research Foundation of Korea grant funded by the Korean Government, Ministry of Education, Science, and Technology (No. 2011-0006464 and No. 2011-0015332).

References

[1] Y. Chon, E. Talipov, H. Shin, and H. Cha, "Mobility Prediction-Based Smartphone Energy Optimization for Everyday Location Monitoring," *Proc. Ninth ACM Conf. Embedded Networked Sensor Systems,* pp. 82-95, 2011.

[2] M.C. Gonzalez, C. Hidalgo, and A.-L. Barabasi, "Understanding Individual Human Mobility Patterns," *Nature,* vol. 453, no. 7196, pp. 779-782, June 2008.

[3] C. Song, Z. Qu, N. Blumm, and A.-L. Barabasi, "Limits of Predictability in Human Mobility," *Science,* vol. 327, no. 5968, pp. 1018-1021, 2010.

[4] L. Song, D. Kotz, R. Jain, and X. He, "Evaluating Next-Cell Predictors with Extensive Wi-Fi Mobility Data," *IEEE Trans. Mobile Computing,* vol. 5, no. 12, pp. 1633-1649, Dec. 2006.

[5] P. Wang, M.C.G. amd C.A. Hidalgo, and A.-L. Barabasi, "Understanding the Spreading Patterns of Mobile Phone Viruses," *Science,* vol. 324, no. 5930, pp. 1071-1076, 2009.

[6] F. Alizadeh-Shabdiz and E.J. Morgan, "System and Method for Estimating Positioning Error within a WLAN-Based Positioning System," US Patent 2008/0 108 371 A1, 2008.

[7] A. LaMarca et al., "Place Lab: Device Positioning Using Radio Beacons in the Wild," *Proc. Third Int'l Conf. Pervasive Computing,* pp. 116-133, 2005.

[8] M. Azizyan, I. Constandache, and R.R. Choudhury, "Surround-sense: Mobile Phone Localization via Ambience Fingerprinting," *Proc. 15th Ann. Int'l Conf. Mobile Computing and Networking,* pp. 261-272, 2009.

[9] D.H. Kim, Y. Kim, D. Estrin, and M.B. Srivastava, "SensLoc: Sensing Everyday Places and Paths Using Less Energy," *Proc. Eighth ACM Conf. Embedded Networked Sensor Systems,* pp. 43-56, 2010.

[10] H. Lu et al., "The Jigsaw Continuous Sensing Engine for Mobile Phone Applications," *Proc. Eighth ACM Conf. Embedded Networked Sensor Systems,* pp. 71-84, 2010.

[11] J. Paek, J. Kim, and R. Govindan, "Energy-Efficient Rate-Adaptive GPS-Based Positioning for Smartphones," *Proc. Eight Int'l Conf. Mobile Systems, Applications, and Services,* pp. 299-314, 2010.

[12] Z. Zhuang, K.-H. Kim, and J.P. Singh, "Improving Energy Efficiency of Location Sensing on Smartphones," *Proc. Eighth Int'l Conf. Mobile Systems, Applications, and Services,* pp. 315-330, 2010.

[13] J. Paek, K.-H. Kim, J.P. Singh, and R. Govindan, "Energy-Efficient Positioning for Smartphones Using Cell-ID Sequence Matching," *Proc. Ninth Int'l Conf. Mobile Systems, Applications, and Services,* pp. 293-306, 2011.

[14] I. Constandache, S. Gaonkar, M. Sayler, R. Choudhury, and L. Cox, "EnLoc: Energy-Efficient Localization for Mobile Phones," *Proc. IEEE INFOCOM,* pp. 2716-2720, 2009.

[15] K. Lin, A. Kansal, D. Lymberopoulos, and F. Zhao, "Energy-Accuracy Trade-Off for Continuous Mobile Device Location," *Proc. Eighth Int'l Conf. Mobile Systems, Applications, and Services,* pp. 285-298, 2010.

[16] Y. Ma, R. Hankins, and D. Racz, "iLoc: A Framework for Incremental Location-State Acquisition and Prediction Based on Mobile Sensors," *Proc. 18th ACM Conf. Information and Knowledge Management,* pp. 1367-1376, 2009.

[17] N. Klepeis et al., "The National Human Activity Pattern Survey (NHAPS): A Resource for Assessing Exposure to Environmental Pollutants," *J. Exposure Analysis and Environmental Epidemiology,* vol. 11, no. 3, pp. 231-252, 2001.

[18] Y. Chon and H. Cha, "LifeMap: A Smartphone-Based Context Provider for Location-Based Services," *IEEE Pervasive Computing,* vol. 10, no. 2, pp. 58-67, Apr.-June 2011.

[19] Y. Chon, E. Talipov, and H. Cha, "Autonomous Management of Everyday Places for Personalized Location Provider," *IEEE Trans. Systems, Man, and Cybernetics Part C: Applications Rev.,* vol. 42, no. 4, pp. 518-531, July 2012.

[20] S. Isaacman et al., "Human Mobility Modeling at Metropolitan Scales," *Proc. ACM 10th Int'l Conf. Mobile Systems, Applications, and Services,* pp. 239-252, 2012.

[21] A. Nicholson and B. Noble, "BreadCrumbs: Forecasting Mobile Connectivity," *Proc. 14th Ann. Int'l Conf. Mobile Computing and Networking,* pp. 46-57, 2008.

[22] J.-K. Lee and J.C. Hou, "Modeling Steady-State and Transient Behaviors of User Mobility: Formulation, Analysis, and Application," *Proc. Seventh ACM Int'l Symp. Mobile Ad Hoc Networking and Computing,* pp. 85-96, 2006.

[23] L. Song, U. Deshpande, U.C. Kozat, D. Kotz, and R. Jain, "Predictability of WLAN Mobility and Its Effects on Bandwidth Provisioning," *Proc. IEEE INFOCOM,* pp. 1-13, 2006.

[24] S. Scellato, M. Musolesi, C. Mascolo, V. Latora, and A.T. Campbell, "NextPlace: A Spatio-Temporal Prediction Framework for Pervasive Systems," *Proc. Ninth Int'l Conf. Pervasive Computing,* 2011.

[25] B. Priyantha, D. Lymberopoulos, and J. Liu, "LittleRock: Enabling Energy-Efficient Continuous Sensing on Mobile Phones," *IEEE Pervasive Computing,* vol. 10, no. 2, pp. 12-15, Apr.-June 2011.

[26] N. Lane et al., "A Survey of Mobile Phone Sensing," *IEEE Comm.,* vol. 48, no. 9, pp. 140-150, Sept. 2010.

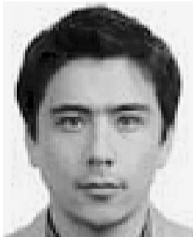
[27] H. Falaki et al., "Diversity in Smartphone Usage," *Proc. Seventh Int'l Conf. Mobile Systems, Applications, and Services,* pp. 179-194, 2010.

[28] D.H. Kim, J. Hightower, R. Govindan, and D. Estrin, "Discovering Semantically Meaningful Places from Pervasive RF-Beacons," *Proc. 11th Int'l Conf. Ubiquitous Computing,* pp. 21-30, 2009.

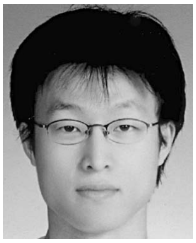
[29] P. Jaccard, "The Distribution of the Flora in the Alpine Zone," *New Phytologist,* vol. 11, no. 2, pp. 37-50, 1912.

[30] D. Kotz and T. Henderson, "CRAWDAD: A Community Resource for Archiving Wireless Data at Dartmouth," *IEEE Pervasive Computing,* vol. 4, no. 4, pp. 12-14, Oct.-Dec. 2005.

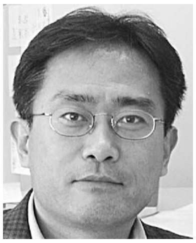
Yohan Chon received the BS and MS degrees in 2009 and 2011, respectively, from Yonsei University, Seoul, Korea, where he is currently working toward the PhD degree in the Department of Computer Science. His research interests include mobile sensing systems, human mobility prediction, and context-aware pervasive computing.



Elmurod Talipov received the MS and PhD degrees in computer science from Yeungnam University in 2007 and Yonsei University in 2012, respectively. He is currently a researcher at Samsung Electronics, Korea. His research interests include ad hoc and wireless sensor networks, delay-tolerant networks, mobile computing, and wireless communication systems.



Hyojeong Shin received the BS, MS, and PhD degrees in computer science in 2005, 2007, and 2012, respectively, from Yonsei University, Seoul, Korea. His research interests include mobile and pervasive computing, wireless sensor networks, embedded systems, and distributed networking systems.



Hojung Cha received the BS and MS degrees in computer engineering from Seoul National University, Korea, in 1985 and 1987, respectively. He received the PhD degree in computer science from the University of Manchester, United Kingdom, in 1991. He is currently a professor in the Department of Computer Science, Yonsei University, Seoul, Korea. His research interests include wireless and mobile systems, embedded operating systems, and sensor network systems. He is a member of the IEEE.