LifeMap: A Smartphone-Based Context Provider for Location-Based Services

LifeMap, a smartphone-based context provider operating in real time, fuses accelerometer, digital compass, Wi-Fi, and GPS to track and automatically identify points of interest with room-level accuracy.

Yohan Chon and Hojung Cha
Yonsei University

The widespread dissemination of smartphones has enabled pervasive sensing environments that allow collecting and capturing the user's context, which refers to information that can help characterize a user's activity or status in a given situation. Understanding user context is a prerequisite for providing human-centered services that improve quality of life. A smartphone is an appropriate device to infer user context because data on the frequent interactions between users and their devices can be easily collected using various kinds of embedded sensors. Furthermore, a smartphone makes it possible to generate and use social context, in addition to individual context, through Internet connectivity.

The availability of context-aware smartphone services makes several scenarios possible. For example, using social-context information, a company scouting for locations to display its advertisements can obtain useful information on various places frequented by urban women in their twenties in the evening. Parents could acquire information on where their children usually go at certain times of the day. The Board of Health Ministry could trace the routes taken by people infected with specific diseases.

Location information forms a core context in a pervasive computing environment. Several approaches are used to determine user location. GPS is a common solution in open, outdoor environments. However, previous research shows that a GPS signal is available only 4.5 percent of the time during a typical user's day.[1] This suggests that average users spend much of their time indoors, where GPS service is normally restricted. The Wi-Fi positioning system is an effective alternative to GPS for indoor environments.[2,3] Although this method provides a reasonable degree of accuracy, the radio map must be constructed offline at an additional cost to obtain accurate location information. Other methods for indoor localization include specialized real-time locating systems (RTLS)[4] or inertial measurement unit (IMU)-based navigation systems.[5] These methods also require a costly infrastructure or additional hardware. Therefore, the prevailing technology used for indoor locating systems hardly satisfies the need for a cost-effective solution, nor does it provide the required room-level locating accuracy. Collecting user context indoors is essential for location-based services because the quality of service depends on the accuracy of mobile users' location information.

To address these limitations, we developed LifeMap, a smartphone-based context provider. Implemented on commercial smartphones,

Published by the IEEE CS ■ 1536-1268/11/$26.00 © 2011 IEEE

it can provide advanced location-based services for mobile users. In particular, LifeMap uses inertial sensors in the smartphone to provide indoor location information. The information is combined with GPS and Wi-Fi positioning systems to generate user context in daily life. The system doesn't require additional infrastructure or costly hardware. LifeMap uses an event-driven technique that reduces energy consumption by using a minimum set of sensors to define context in a given situation. The system is practical and efficient, and it provides accurate location information about mobile users in indoor and outdoor environments.

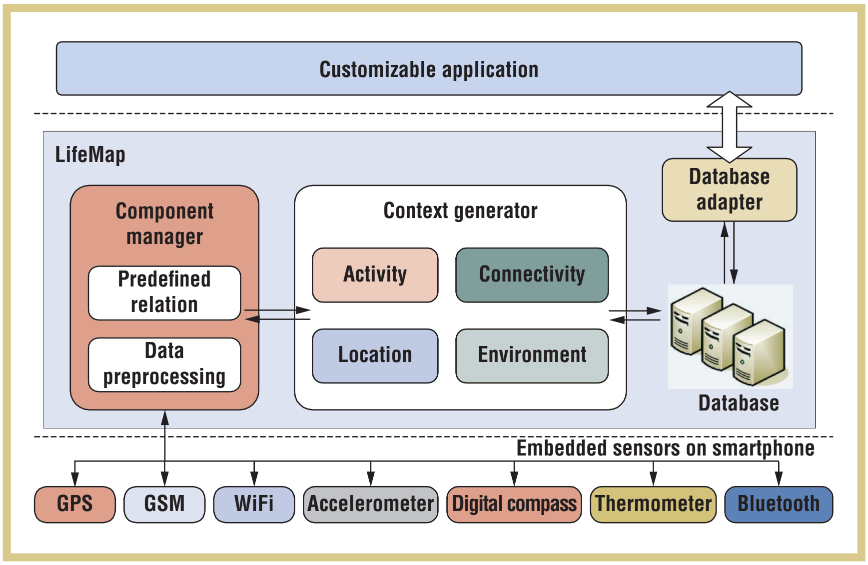

Figure 1. LifeMap platform. To extract user context, LifeMap uses various sensors commonly found in the latest smartphones.

LifeMap Platform

LifeMap's targeted localization accuracy is at the room level because it is necessary to accurately observe users to provide successful location-based services. LifeMap consists of three components:

- The *component manager* interfaces with hardware directly to abstract sensor data and provide high-level information.
- Using this high-level information, the *context generator* produces context nodes (points of interest [POIs] containing the user context) and edges (paths encompassing minimal context on the user movement) to construct a context map in the form of a graph. The context map is stored in a database to match and aggregate user contexts.
- The *database adapter* is a wrapper module to provide user context to the internal user interface and other applications.

Figure 1 shows the LifeMap platform's overall architecture.

Although LifeMap generates accurate context, it would be less useful if its battery consumption were high. Therefore, we choose to leave the accelerometer hardware component always turned on because of its relatively low power consumption.[6]

We define two types of user motion, *moving* and *stationary*, depending on the standard deviation of the accelerometer value. If a user is walking, running, or moving in a vehicle, the motion is defined as moving; the motion is stationary if the user is staying at one location. When the component manager detects a change in the user's motion, a minimum set of sensors is activated via a predefined relationship. By obtaining the context from the activated sensors, the component manager immediately deactivates irrelevant sensors for energy efficiency. Figure 2 illustrates the motion-based decision rules.

Context Generator

The goal of LifeMap is to construct a context map on the geographical map. LifeMap generates a categorized user context into four parts:

- *Location* represents the user's position on Earth. Location information is specified with properties of the scanned Wi-Fi access point to determine room-level identification.
- *Activity* is defined by two characteristics: user motion and smartphone usage. User motion includes moving in a vehicle, walking, running, or even staying at rest. Smartphone usage includes messaging, calling, taking a picture, and browsing the Web.
- *Connectivity* shows the current network connection status of the GSM (Global System for Mobile Communications, originally from Groupe Spécial Mobile) or Wi-Fi.
- *Environment* is a set of circumstances around the user.

Table 1 gives detailed information for each categorized context.

We mainly focus on presenting the locating scheme because location is a core context in modern mobile services. The key concepts are tracking indoor location using inertial sensors in the smartphone and aggregating the identical location using the properties of pervasive Wi-Fi access points. In our approach, the aggregation process refines location information based on historical data because the system determines an identical place through room-level identification. (A detailed technical

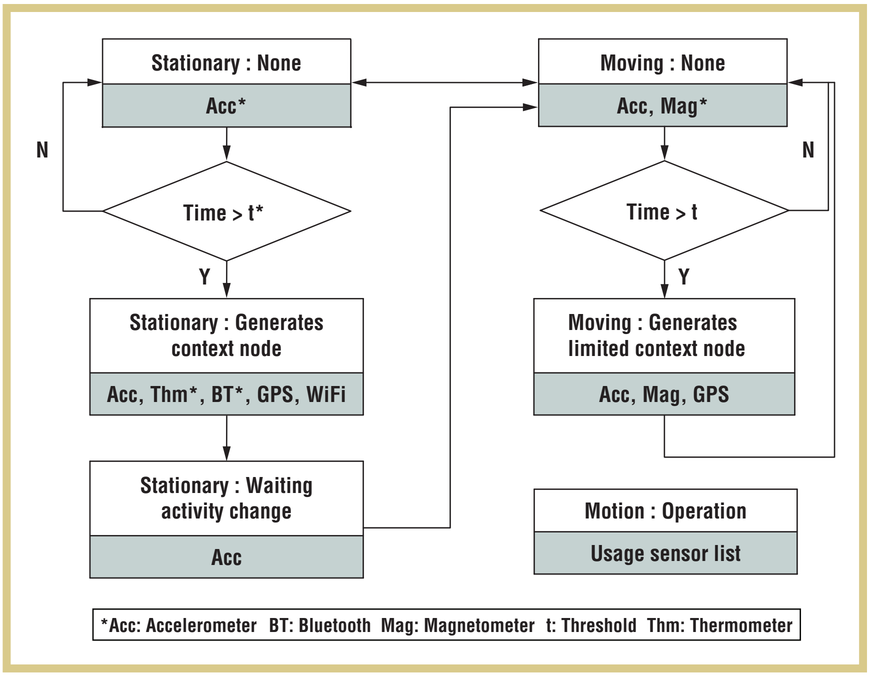

Figure 2. Decision rule of the motion-based event-driven approach. A minimum set of sensors is activated when the component manager detects a change in the user's motion.

description of the locating scheme and parameter analysis are available in our technical report.[6])

Locating Scheme

A representative indoor locating solution is an IMU-based method that uses double integration of acceleration measurements. This method can estimate the user's current location without communicating with other components. Previous work on IMU-based systems assumes strapdown usage, in which users must fasten sensors around their waist or on their shoes. The conventional strapdown system is not applicable to smartphones because fastening the device to the human body is impractical and filtering noise signals demands nontrivial computation.

The existing methods to determine device orientation for vector transformation at runtime also pose problems, as Figure 3 illustrates. The scheme is inappropriate for smartphones because sampling at high frequency and complex computation are needed to obtain accurate measurements. A gyroscope could be used to enhance tracking accuracy, but even the latest smartphones do not employ a gyroscope due to its relatively high cost. Hence, using a smartphone for location-based services in indoor environments requires specifically considering a nonrestricted position, low processing power, and usage of a limited set of inertial sensors in the device.

A practical positioning and tracking solution for users in indoor environments relies on both an accelerometer and a digital compass. When a user starts to move, classification data acquired from both the accelerometer and the digital compass are used to approximate the user's location. Tracking

TABLE 1
Sources of categorized context.

Category	Attributes	GPS	Accelerometer	Digital compass	Wi-Fi	Other	Remarks
Location	Latitude	X	X	X		GSM	
	Longitude	X	X	X		GSM	
	Altitude	X					
	Error bound	X	X	X			
	Identification				X		
Activity	Motion	X	X				
	Behavior					Phone	
Connectivity	No. of satellites	X					
	Connection				X	GSM	
	No. of access points				X		
Environment	Temperature					Thermometer	
	Speed	X					Traffic condition
	No. of devices					Bluetooth	No. of surrounding devices

the user location requires solutions for identifying the user's movement and determining where and how far the user has moved.

To detect user movement, we use a synthetic acceleration vector of the three-axis accelerometer. Given an accelerometer input vector for a certain duration (for example, 3 seconds), movement is detected if the standard deviation of the acceleration is greater than a certain threshold (for example, 0.49 m/s^2 [meters per second squared]), which is defined as moving motion; otherwise, motion is stationary. A peak detection algorithm is then employed to mark the candidate time points, which could be user steps. We use the mean and standard deviation of acceleration to eliminate insignificant peaks and generate worthy candidates that are significantly away from the average.

However, the peak detection algorithm is prone to miscounting steps during a user's irregular behavior, such as a swing, vibration, or impact. Given that the user's steps are periodically repeated, we determine the candidate peak as a step when its previous acceleration signal is similar to the current one. In other words, we use the acceleration dataset of previous movements as the classification data. This approach is reasonable because the classification data is customized for each person and for specific situations.

To determine the similarity of the signals, we compare the max axis and the forward axis of the current peak with those of the previous peak. The max axis is the axis perpendicular to the ground among the three axes of the accelerometer. The acceleration of the max axis is the cardinal value because the user's movement does not generate a larger acceleration than the acceleration of gravity. We define the forward axis as the most similar axis to the direction of user movement. The forward axis is determined by its variance because a walking motion influences the direction of movement more than the orthogonal direction. Then, abstract orientation is defined as a combination of max axis and forward axis.

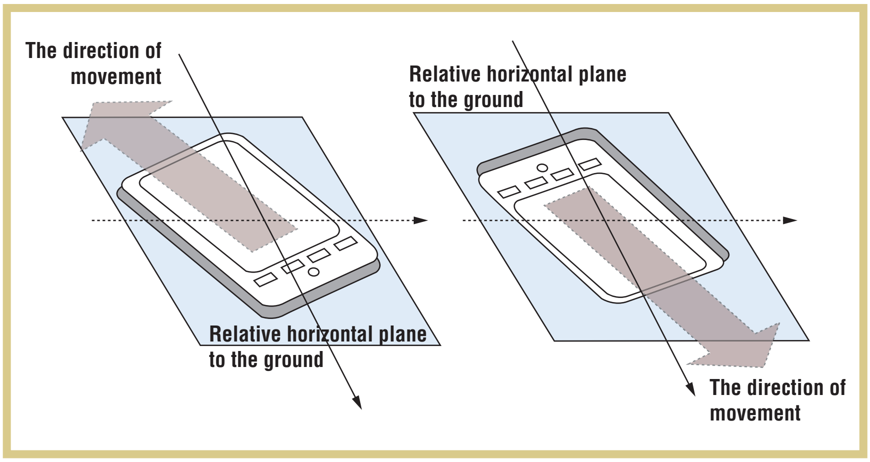

Figure 3. Existing methods for determining device orientation. Both cases have identical acceleration at the device aspect (that is, upward of the device screen), but they have different acceleration at the human aspect because the device orientation differs. In addition, the system cannot eliminate the acceleration of gravity completely if measured orientation is inaccurate.

Hence, a peak is considered a step if the current peak's abstract orientation is equal to that of the previous peak. Our scheme would still miss step counts that occur when users change their motion (for example, switches from calling to messaging), but we obtain the overall trend of the user movement, which is our intent. Figure 4 shows the result of the peak detection scheme.

For our application, we must also obtain information about where the user is heading. We restrict the direction of user movement within the line of the sensor's axis because transforming a local acceleration into a global one is a nontrivial operation for smartphones. Based on this idea, we first find abstract orientation. Next, we must determine whether the movement is forward or backward on the line of the forward axis. A key factor to decide the direction is the velocity, which is obtained from integrating the acceleration. Additionally, the current usage of the smartphone should be considered. We can easily determine the direction of movement while the user is using the phone, such as watching a video or talking over the phone because the smartphone's orientation is obvious in this case. This approach simplifies the determination of the movement direction, as Figure 5 illustrates. Because the acquired direction of movement is unstable, the direction should be further enhanced. The direction of movement is compensated when the GPS signal gives the heading information. Among four directions within the sensor's axis on the horizontal plane, the revision method selects the most similar direction to the GPS signal's heading information. The revised direction is maintained until the abstract orientation is changed. The inherent limitation of our tracking model is the error caused by discordance between the user's direction of movement and the axis of the inertial sensors.

The user's moving distance is calculated by multiplying the step count by the step length that is manually input by the user. The measured distance is used to infer the estimated location's error bound, which is initialized as the accuracy of the GPS signal. When a GPS signal is unavailable, a summation of the previous accuracy and the measured moving distance are used as the error bound.

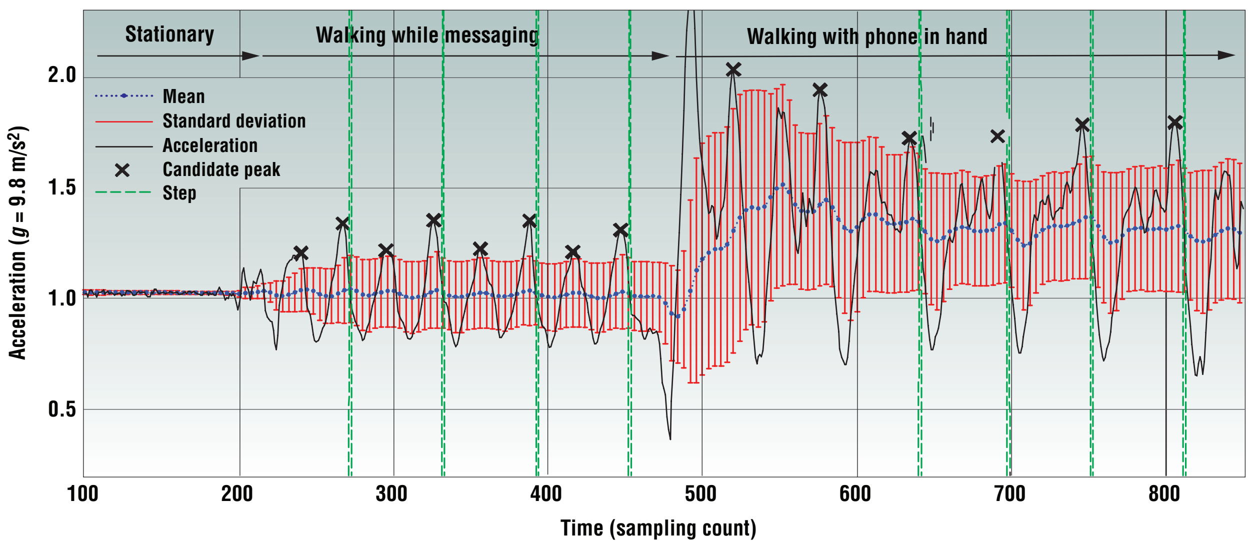

Figure 4. The result of peak detection. LifeMap identifies candidate peaks as a step if the previous abstract orientation is equal to the current one.

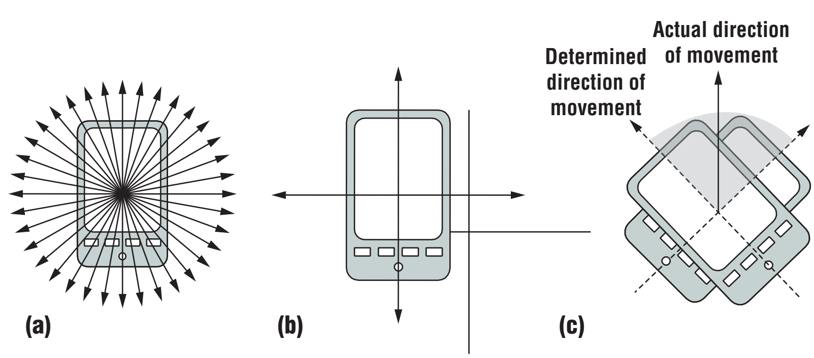

Figure 5. Simplified determination of a moving direction. (a) Given all possible directions of movement, (b) we limit the directions of movement within the line of the sensor's axis. (c) The inherent error from discordance occurs between the user's direction of movement and the axis of the inertial sensors. Although heading is determined appropriately, a maximum error of 45 degrees can occur, depending on the device's orientation.

When a user stays at a given location for a certain period of time, the user state is considered stationary and the place is called a POI. We set 10 minutes as the default time, but the user can change the threshold as needed. The stationary state is used to generate a context node in LifeMap, which is considered the major place for location-based services. In addition to location information obtained with inertial sensors, we use a set of Wi-Fi access points at the POI to specify the room-level identification.

Matching and Aggregation Process

Location is a key criterion in the matching and aggregation process. The matching process is performed to find an identical place on the LifeMap. We use an error bound of each node to select candidate nodes. If the location of one node is included in the error bound of another, the room-level identification is compared. We determine room-level identification from the properties of the scanned access point set, which includes the unique ID of the access point and the root mean square of signal difference. When the matching rate exceeds a certain threshold, both nodes are considered identical.

A user generally follows a similar pattern in the daily routine, but the physical location is not necessarily generated exactly. The aggregation process should reduce the number of redundant nodes without deteriorating the quality of the context. When two POIs are found to be identical in the matching process, LifeMap aggregates the POIs to refine the context. After aggregating the POIs, the paths are determined for aggregation. For instance, if one POI has six out edges, we obtain at most six destinations. If two or more paths have the same destination, the Euclidean distance between the

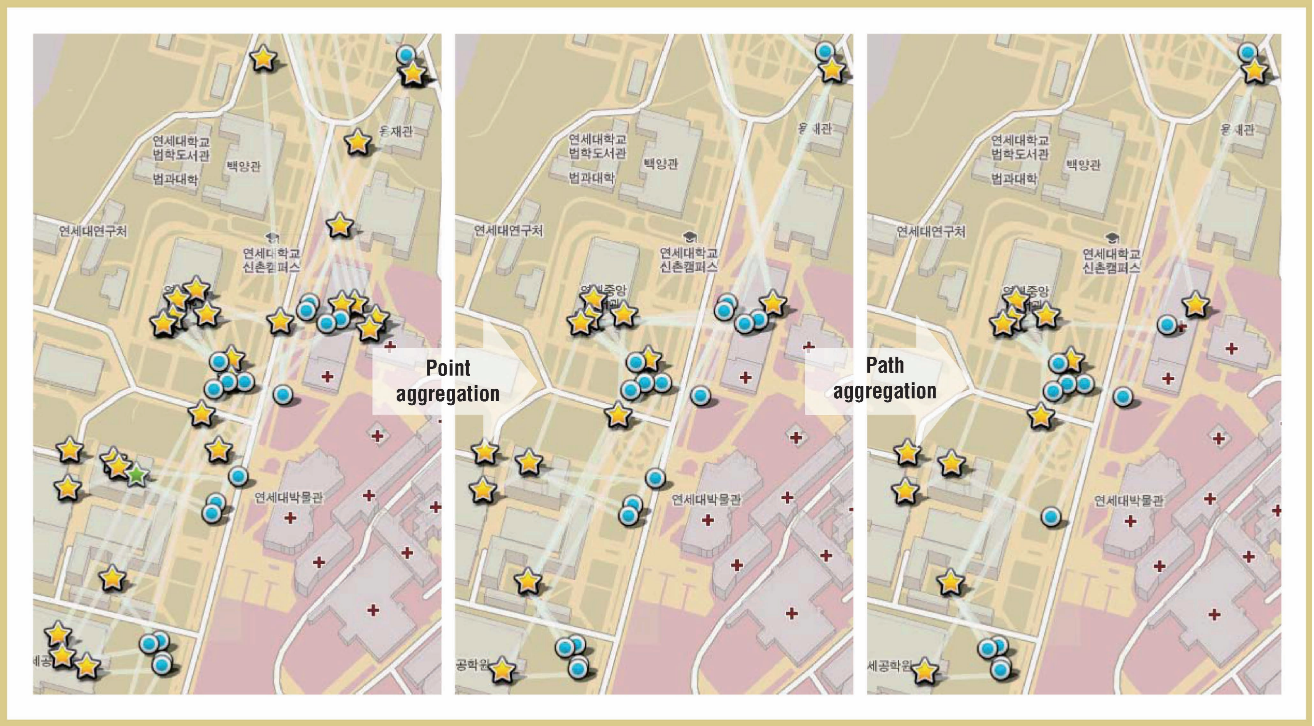

Figure 6. Aggregating identical points of interest (POIs) and similar paths. The stars indicate a stationary state, and the circles indicate a moving state. Here, LifeMap aggregates 34 stationary states and 21 moving states to produce 14 and 12, respectively.

paths is computed to find the identical routes. The path with fewer nodes is considered to be the superseding path to store routes with a minimal number of nodes. Figure 6 illustrates the aggregation process.

Experiments

We implemented and tested LifeMap on the HTC Hero and G1 Android phones. Android is an open source software stack for mobile devices and provides a software developer's kit (SDK) to support various smartphone features (see http://developer.android.com). The phones have several built-in sensors and communication components. The LifeMap implementation performs sensing, converts raw data to high-level user context, activates or deactivates required sensors, aggregates user context, and provides user context to end-user applications. Various kinds of applications can be developed using the LifeMap platform. We have, in fact, implemented a life-logging application to evaluate and visualize LifeMap on smartphones.

We conducted experiments for more than a week using undergraduate and graduate students at Yonsei University. Our participants engaged in normal daily activities with their LifeMap-enabled Android phones. LifeMap ran as a background service without impeding other tasks on the smartphones.

Figure 7 shows snapshots of our application that displays the stored LifeMap data. Users can confirm detailed information about POIs on the Web-based Google Maps (http://maps.google.com) and upload their contexts or download the contexts of other users. The application provides a list of POIs that a user can name.

Because the GPS-based outdoor tracking is a straightforward process, we focus on the locating accuracy of LifeMap in indoor environments. LifeMap basically uses an accelerometer and digital compass. The major source of location error is distortion of where the user is heading. For example, Figure 8 shows that a user's motion affects his or her heading accuracy. Small user movements such as messaging or calling generate relatively stable results, but the heading becomes unstable if the device is, for example, placed in a pocket while walking.

We ordered the cases of user motion by the amount of uncertainty involved. That is, in decreasing order of uncertainty, they are running; walking with the phone in a bag, in a pocket, or in hand; walking while on a call; and walking while messaging. Even with the estimated heading, the error bound of the generated location should include the actual location. Depending on the user's motion, we increase the error bound to the step length (minimum) and double the step length (maximum), which are loosely estimated to include the actual location within the error bound.

Figure 7. Our Android application. Users can confirm daily information, such as sensor usage time, moving distance, and number of steps in the first tab. The second tab shows the generated user context. The detailed information, including error bound, is displayed when a user selects a specific node. The last tab shows the list of POIs that a user can name.

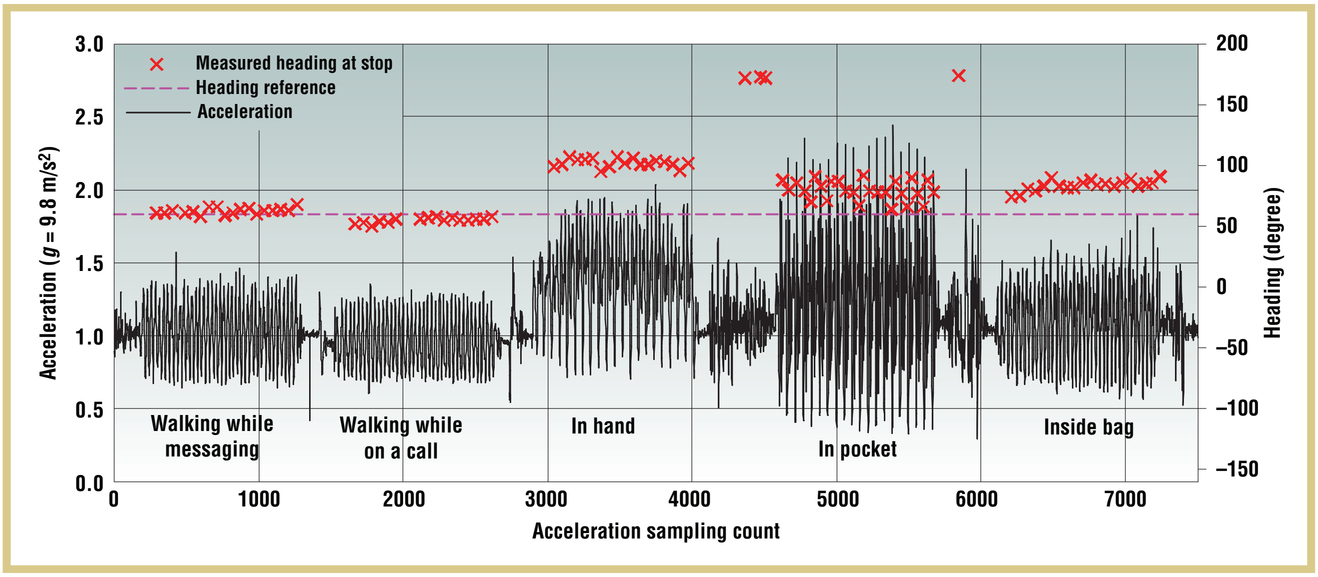

Figure 8. Heading accuracy in different motions. The actual heading is approximately 60 degrees. The error is caused by discordance between the user's direction of movement and the axis of the inertial sensors.

Figure 9a shows the user paths in an indoor environment. The total walking distance was 186 meters, and the estimated walking distance was 171 meters, with 7.9 meters average error. The measured walking distance is underestimated because of the missing steps during irregular motions such as turning corners, starting to walk, and stopping. The tracking error is mainly caused by a twisted heading direction. When the phone is placed in a position that shakes (such as in a pocket), the direction of movement was twisted due to the device swing. In addition, environmental factors, such as an electronic apparatus, cause the distortion of the magnetic value. Figure 9b shows that the error increases with moving distance. Although the drift error existed, we obtained an overall trend of user movement with inertial sensors, which is ultimately our main interest. The cumulative error in finding POIs is compensated in the aggregation process.

Figure 10 shows the aggregated user context of three students collected for a week on a university campus. Among 54 POIs, 46 points are generated indoors where the GPS signal is unavailable. Thus, the experiment participants manually obtained the ground truth through Google Maps. Of the indoor-generated POIs, 91 percent of the points indicate the actual position within the estimated error bound, and the average error is 25.6 meters. The aggregation process helped to reduce the POIs from 54 to 18. Still, 83 percent of the points contain their reference locations, and the average error is reduced to 17.7 meters because the nodes with smaller error bounds have been chosen as the superseding one.

One drawback of our method is that the system might generate inaccurate locations if the GPS signal accuracy is poor because the system establishes an initial location through the GPS signal. Locating a user on a 2D map is difficult in many multistory buildings. Therefore, in future work, we need an enhanced method to classify the floor level.

Energy efficiency is critical for mobile devices. The collected user traces show that the average amount of time in the moving state is less than three hours a day. This indicates that the average movement ratio is less than 20 percent in one day. This finding is in agreement with other research results.[7] Our study estimated that the battery lifetime of a Hero smartphone running LifeMap is less than 28 hours when the moving state is about 20 percent in one day.[6] Accordingly, our motion-based, event-driven approach does not lead to a severe impact on a smartphone's battery lifetime, but there is still room for further improvement.

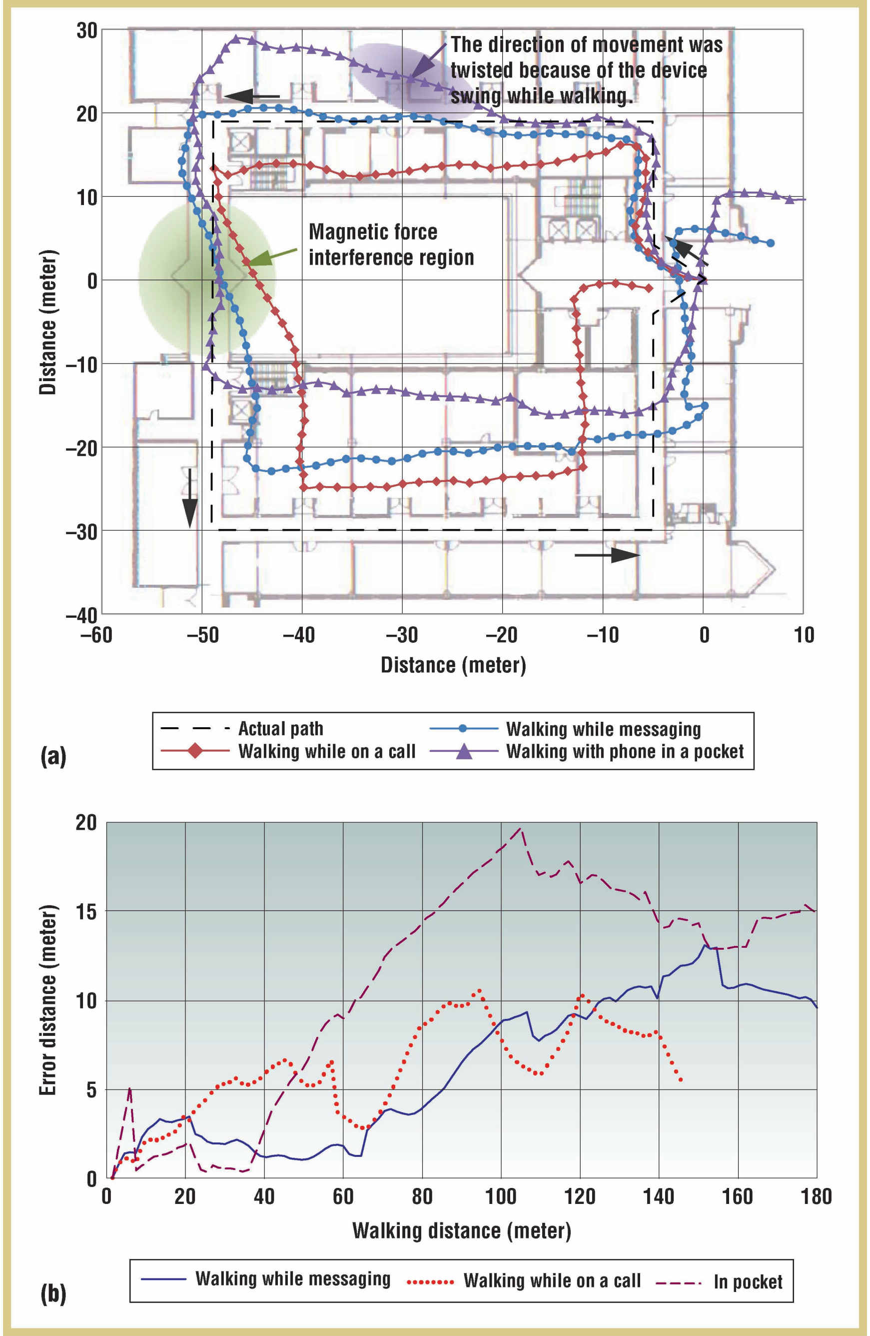

Figure 9. Indoor tracking results with diverse user motions. (a) Magnetic interference in indoor environments causes errors, and (b) error distance increases with walking distance. The results show a greater error if the device is located in an unstable position, such as in a pocket.

Many interesting applications, such as life-logging applications and human-centered delay-tolerant networking, are possible

Figure 10. The daily trace of three students during a week. Blue stars represent the moving state, yellow stars show the POI without GPS connection, and green stars show the POI with the GPS connection.

with LifeMap. Indoor navigation with LifeMap in public places, such as department stores or exhibition halls, is also a useful application if an administrator provides an indoor floorplan.

We are currently studying issues of energy minimization of LifeMap using human-centric location-prediction techniques. We also plan to publish LifeMap on the Android market to provide and share the social context among research communities.

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea grant funded by the Korean government, Ministry of Education, Science and Technology (number 2010-0000405).

REFERENCES

1. A. LaMarca et al., "Place Lab: Device Positioning Using Radio Beacons in the Wild," *Proc. 3rd Int'l Conf. Pervasive Computing*, LNCS 3468, Springer, 2005, pp. 116–133.

2. P. Bahl and V.N. Padmanabhan, "RADAR: An In-Building RF-based User Location and Tracking System," *Proc. 19th Ann. Joint Conf. IEEE Computer and Comm. Societies* (INFOCOM), 2000, IEEE Press, 2000, pp. 775–784.

3. 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, Patent and Trademark Office, 2008.

4. A. Cavanaugh et al., "WPI Precision Personnel Locator System: Rapid Deployment Antenna System and Sensor Fusion

the AUTHORS

Yohan Chon is a PhD candidate in computer science at Yonsei University, Korea. His research interests include mobile sensing systems, human mobility prediction, and context-aware pervasive computing. Chon has an MS in computer science from Yonsei University. Contact him at yohan@cs.yonsei.ac.kr.

Hojung Cha is a professor in computer science at Yonsei University, Korea. His research interests include pervasive and mobile systems, embedded operating systems, and sensor network systems. Cha has a PhD in computer science from the University of Manchester, England. Contact him at hjcha@cs.yonsei.ac.kr.

Related Work in Context-Aware Mobile Systems

Active research has recently been conducted on context-aware mobile systems. CenceMe generates inference on individual contexts and enables the use of social context by sharing personal information through social networking applications.[1] The limited hardware resources of mobile devices have been studied. The Energy Efficient Mobile Sensing System (EEMSS) generates high-level context, using an accelerometer, microphone, and GPS.[2] The system reduces energy consumption by enabling only a minimum set of sensors with appropriate duty cycles. Work is also focused on recognizing 11 predefined user states, such as working, meeting, and resting.

SurroundSense suggested an ambience fingerprinting scheme that considers Wi-Fi fingerprinting as well as optical, acoustic, and motion attributes as major sources of user context.[3] PlaceSense exploited the response rates of radio beacons to identify a logical place through robust beacon inference.[4] Oliver Woodman and Robert Harle automatically constructed an indoor radio map by using a foot-mounted inertial measurement unit (IMU) with a detailed building model.[5] Although the system achieves scalability in large environments, a foot-mounted IMU is impractical for general users. Martin Mladenov and Michael Mock implemented a step counter with a commercial smartphone, which is independent of the phone's position on the human body.[6] The system, however, does not provide the direction of the user's movement.

CompAcc introduced a scheme that tracks users by recording a person's walking pattern using the digital compass and accelerometer in a smartphone, then matching it against possible path signatures from a map.[7] The scheme does not consider free movement of the smartphone user. Users should therefore manually adjust their directions of movement to the direction faced by the back of the phone when the phone is held vertically.

These works investigated the inference of user context, system scalability, and energy efficiency of mobile devices. However, research on context-aware systems, especially in indoor environments, is a relatively new area with many challenges. LifeMap is unique in its ability to provide location information in indoor environments and to visualize the user's context. The system also combines logical identification of places with geographical information. To the best of our knowledge, our system is the first attempt to track indoor location with unconstrained phone placement by employing a smartphone's inertial sensors.

REFERENCES

1. E. Miluzzo et al., "Sensing Meets Mobile Social Networks: The Design, Implementation, and Evaluation of the CenceMe Application," *Proc. 6th ACM Conf. Embedded Network Sensor Systems* (SenSys), ACM Press, 2008, pp. 337–350.
2. Y. Wang et al., "A Framework of Energy Efficient Mobile Sensing for Automatic User State Recognition," *Proc. 7th Int'l Conf. Mobile Systems, Applications, and Services* (MobiSys), ACM Press, 2009, pp. 179–192.
3. M. Azizyan, I. Constandache, and R.R. Choudhury, "SurroundSense: Mobile Phone Localization via Ambience Fingerprinting," *Proc. Ann. Int'l Conf. Mobile Computing and Networking* (MobiCom), ACM Press, 2009, pp. 261–272.
4. D.H. Kim et al., "Discovering Semantically Meaningful Places from Pervasive RF-Beacons," *Proc. 11th Int'l Conf. Ubiquitous Computing* (Ubicomp), ACM Press, 2009, pp. 21–30.
5. O. Woodman and R. Harle, "Pedestrian Localisation for Indoor Environments," *Proc. 10th Int'l Conf. Ubiquitous Computing* (UbiComp), ACM Press, 2008, pp. 114–123.
6. M. Mladenov and M. Mock, "A Step Counter Service for Java-Enabled Devices Using a Built-in Accelerometer," *Proc. 1st Int'l Workshop Context-Aware Middleware and Services* (CAMS), ACM Press, 2009, pp. 1–5.
7. I. Constandache, R.R. Choudhury, and I. Rhee, "Towards Mobile Phone Localization Without War-Driving," *Proc. Ann. Joint Conf. IEEE Computer and Comm. Societies* (INFOCOM), IEEE Press, 2010, pp. 1–9.

for 3D Precision Location," *Proc. Inst. Navigation, Int'l Technical Meeting*, vol. 1, Inst. of Navigation (ION), 2010, pp. 384–389.

5. O. Woodman and R. Harle, "Pedestrian Localisation for Indoor Environments," *Proc. 10th Int'l Conf. Ubiquitous Computing*, ACM Press, 2008, pp. 114–123.
6. Y. Chon, E. Talipov, and H. Cha, *Autonomous Management of Personalized Location Provider for Mobile Services*, tech. report, Dept. of Computer Science, Yonsei Univ., 2010.
7. 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* (CIKM), ACM Press, 2009, pp. 1367–1376.