

## Research Article

# A Study of Mobile Sensing Using Smartphones

**Ming Liu**

*School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, China*

Correspondence should be addressed to Ming Liu; [wing.lm@gmail.com](mailto:wing.lm@gmail.com)

Received 8 December 2012; Accepted 15 January 2013

Academic Editor: Chao Song

Copyright © 2013 Ming Liu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Traditional mobile sensing-based applications use extra equipments which are unrealistic for most users. Smartphones develop in a rapid speed in recent years, and they are becoming indispensable carry-on of daily life. The sensors embedded in them provide various possibilities for mobile applications, and these applications are helping and changing the way of our living. In this paper, we analyze and discuss existing mobile applications; after that, future directions are pointed out.

## 1. Introduction

The word sensing builds a bridge between real world and virtual world; with the help of various sensors, man-made devices are able to feel the world like God-made creatures do. Bell may be the first generation of sensors; people tie up a bell to a string so that when there is a vibration on the string, the bell will ring. Bell is a very powerful and effective sensor; it contains two parts: detection and processing. When a bell detects a vibration, it will generate a period of ringing and the volume of the ringing is proportional to the amplitude of the vibration. However, bell is the kind of sensor that connects real world to real world. With the development of electronic devices, a new man-made world has been building. This world is called virtual world; many complicated calculations are running in this world so that people in real world can enjoy their lives. Virtual world needs data to keep running, and it is far from enough to input data into the virtual world depending on human operations. Sensor is a way to sense the world and interpret the sensed information to the data form of virtual world; therefore, sensing becomes an important part of research field and industry field.

Early sensing-based applications are mostly used for research purposes or used in some specific areas. References [1, 2] propose localization methods for finding odor sources using gas sensors and anemometric sensors. Reference [3] uses a number of sensors embedded into a cyclist's bicycle

to gather quantitative data about the cyclist's rides; this information would be useful for mapping the cyclist experience. Reference [4] uses body-worn sensors to build an activity recognition system, and [5] uses body-worn sensors for healthcare monitoring. Reference [6] proposes a robotic fish carrying sensors for mobile sensing. Also, in Wireless Sensor Networks (WSN), there are a lot of sensing-based applications. References [7, 8] deploy wireless sensors to track the movement of mobile objects. References [9, 10] deploy sensors for monitoring volcano.

People-centric sensing mentioned in [11] uses smartphones for mobile sensing. Smartphones are very popular and becoming indispensable carry-on for people in recent years; they are embedded with various sensors which could be used for many interesting applications. Unlike specific sensors which are used for specific areas, sensors in smartphones could provide unlimited possibilities for applications to help and change the life of people; also, using smartphone instead of specific equipment makes an application easier to be accepted by users.

In this paper, we will discuss some existing interesting sensing-based applications using smartphones and give some possible future directions. Section 2 gives detailed descriptions of sensors embedded in modern smartphones; Section 3 introduces some sensing-based applications; Section 4 gives a conclusion and future directions.

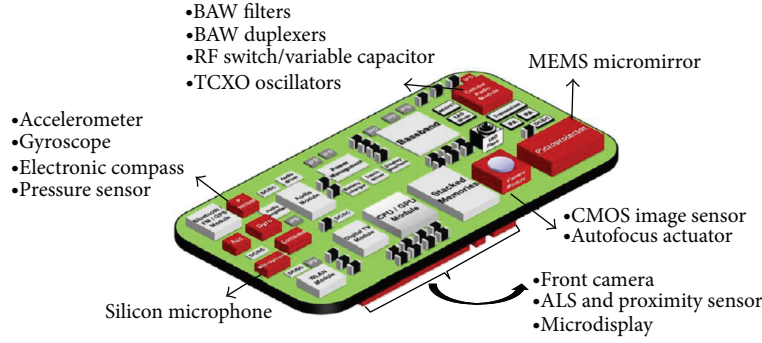


FIGURE 1: Sensors inside of smartphones.

## 2. Sensors in Smartphones

As Figure 1 shows, modern smartphones have several kinds of sensors. The most popular sensors which most smartphones have are accelerometer, gyroscope, magnetometer, microphone, and camera. In this section, we will discuss the characteristics of the sensors.

**2.1. Accelerometer.** An accelerometer measures proper acceleration, which is the acceleration it experiences relative to free fall and is the acceleration felt by people and objects. To put it another way, at any point in spacetime the equivalence principle guarantees the existence of a local inertial frame, and an accelerometer measures the acceleration relative to that frame. Such accelerations are popularly measured in terms of g-force [12].

The principle of accelerometer is using inertial force. Try to imagine a box with six walls, a ball is floating in the middle of the box because no force is added to the ball (e.g., the box may be in the outer space) [13]. When the box moves to the right direction, the ball will hit the left wall. The left wall is pressure sensitive that it can measure the force of hitting applied to the left wall; therefore, the acceleration can be measured. Because of gravity, when the box is placed at earth, the ball will keep pressing the bottom wall of the box and give constant  $\sim 9.8 \text{ m/s}^2$  acceleration. The gravity force will affect the measurement of accelerometer for measuring speed or displacement of an object in a three-dimension. The gravity force must be subtracted before any measurement. However, the gravity force can be taken as an advantage of detecting the rotation of a device. When a user rotates his smartphone, the content he/she is watching will switch between portrait and landscape. As Figure 2 shows, when the screen of smartphone is in a portrait condition,  $y$ -axis will sense the gravity; when the screen of smartphone is in a landscape condition,  $x$ -axis will sense the gravity. According to this, users can rotate their screens without affecting their reading experiences.

In theory, the displacement can be calculated as

$$d(t) = d_0 + v_0 t + \int_0^T a(t) dt, \quad (1)$$

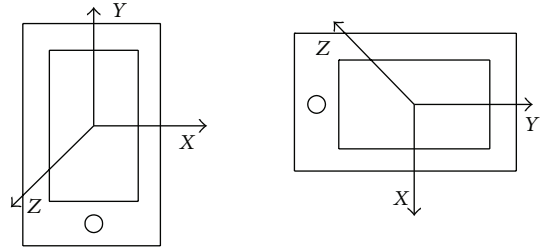


FIGURE 2: Screen rotation.

where  $d(t)$ : displacement,  $d_0$ : initial displacement,  $v_0$ : initial velocity, and  $a(t)$ : acceleration.

Equation (1) is a continuous function; the  $a(t)$  we get in real environment is discrete due to sampling. To calculate the displacement according to discrete values, (2) has to be used as

$$\int_{t(0)}^{t(n)} a(t) dt = \sum_{i=1}^n \left( \frac{a(i-1) + a(i)}{2} \right) \Delta t, \quad (2)$$

where  $a(t)$ : continuous acceleration,  $a(i)$ :  $i$ th sample, and  $\Delta t$ : time increment.

Then, the velocity and displacement can be calculated as the following [14]:

$$\begin{aligned} v(i) &= v(i-1) + \frac{a(i-1) + a(i)}{2} \Delta t, \\ d(i) &= d(i-1) + \frac{v(i-1) + v(i)}{2} \Delta t. \end{aligned} \quad (3)$$

The value the accelerometer returns is three-dimensional as Figure 2 shows; therefore,  $a(t)$  will be calculated as the following shows:

$$a(t) = \vec{a}_x + \vec{a}_y + \vec{a}_z, \quad (4)$$

where  $\vec{a}_x$ ,  $\vec{a}_y$ , and  $\vec{a}_z$  are vectors.

**2.2. Gyroscope.** Accelerometer is good at measuring the displacement of an object; however, it is inaccurate to measure

the spin movement of the device, which is an easy thing for gyroscope.

A gyroscope is a device for measuring or maintaining orientation, based on the principles of angular momentum. Mechanically, a gyroscope is a spinning wheel or disk in which the axle is free to assume any orientation. Although this orientation does not remain fixed, it changes in response to an external torque much lesser and in a different direction than it would be without the large angular momentum associated with the disk's high rate of spin and moment of inertia. The device's orientation remains nearly fixed, regardless of the mounting platform's motion because mounting the device in a gimbal minimizes external torque [15].

Gyroscope is a very sensitive device; it is good at detecting the spin movement. Same as accelerometer, gyroscope returns three-dimensional values; the coordinate system is as Figure 2 shows. The value gyroscope returns is angular velocity which indicates how fast the device rotates around the axes. The gyroscope can be calculated as

$$g(t) = \vec{g}_x + \vec{g}_y + \vec{g}_z, \quad (5)$$

where  $g(t)$ : angular velocity  $\vec{g}_x$ ,  $\vec{g}_y$ , and  $\vec{g}_z$ : vectors of angular velocity around  $x$ -,  $y$ -, and  $z$ -axes.

**2.3. Magnetometer.** A magnetometer is a measuring instrument used to measure the strength and perhaps the direction of magnetic fields [16]. Accelerometer and gyroscope are able to detect the direction of a movement; however, the direction is a relative direction; it obeys the coordinate system that a smartphone uses. Sometimes, different smartphones need to synchronize their directions; therefore, a magnetometer is needed to get an absolute direction (the direction obeys the coordinate system of earth).

The magnetometer returns three-dimensional values; if the device is placed horizontally, the orientation angle can be calculated as

$$\text{angle} = \arctan \frac{M_y}{M_x}. \quad (6)$$

Until now, we introduced three types of sensors: accelerometer, gyroscope, and magnetometer. With the help of the three types of sensors, smartphone can estimate its own all kind of movements. However, in real environment, errors of measurement happen all the time; we will talk about a way to correct the offset error of magnetometer, and the other two sensors may use the same way to correct their errors.

Firstly, place the magnetometer horizontally, rotate it in a uniform speed, measure the value of  $M_x$  and  $M_y$ , put  $z$  axis horizontally, and rotate to measure the value  $M_z$ . Calculate the offset value on the three axes as

$$\begin{aligned} X_{\text{offset}} &= \frac{(\text{MAX}(M_x) + \text{MIN}(M_x))}{2}, \\ Y_{\text{offset}} &= \frac{(\text{MAX}(M_y) + \text{MIN}(M_y))}{2}, \\ Z_{\text{offset}} &= \frac{(\text{MAX}(M_z) + \text{MIN}(M_z))}{2}, \end{aligned}$$

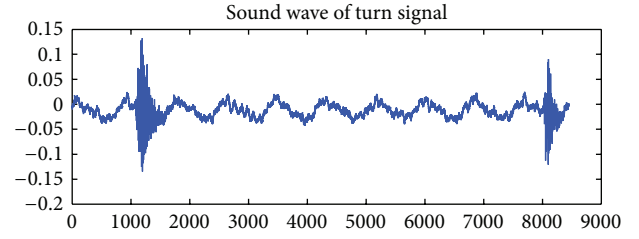


FIGURE 3: A sound record of turn signal of automobile.

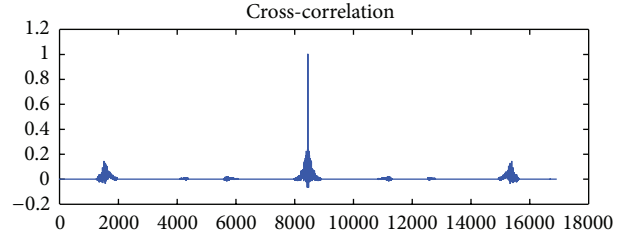


FIGURE 4: Cross-correlation.

$$M'_x = M_x - X_{\text{offset}},$$

$$M'_y = M_y - Y_{\text{offset}},$$

$$M'_z = M_z - Z_{\text{offset}}.$$

(7)

**2.4. Microphones.** Microphone is a very common sensor; it is usually used for recording sound. The problem is how to deal with the recorded sound. The most common way is to find a known period of sound in a sound record. Cross-correlation is a method to search a piece of signal in a mixed signal [17]. In signal processing, cross-correlation is a measure of similarity of two waveforms as a function of a time lag applied to one of them. This is also known as a sliding dot product or sliding inner product. It is commonly used for replacing a long signal for a shorter ones, which is known feature. It also has applications in pattern recognition, single particle analysis, electron tomographic averaging, cryptanalysis, and neurophysiology [18].

Cross-correlation can be calculated as (8) shows. Suppose that the known symbol pattern of sound wave of turn signal is  $ts$  of length  $L$ ,  $X(n)$  is the complex number representing the  $n$ th received symbol, then the cross-correlation at a shift position  $p$  is

$$C(ts, X, p) = \sum_{k=1}^L ts^* [k] X[k + p]. \quad (8)$$

If we use a sound wave to cross-correlate itself, for example, the sound wave as Figure 3 shows, the result will be shown as in Figure 4. The spike indicates the existence of a piece of signal.

**2.5. Camera.** Camera captures vision information from real world. From the human perspective, vision contains most

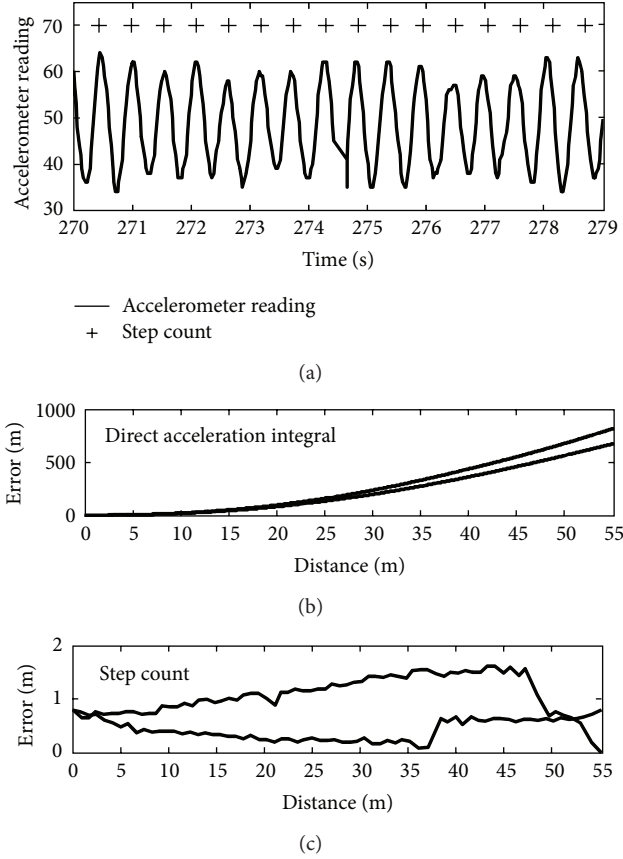


FIGURE 5: (a) Accelerometer readings (smoothened) from a walking user. (b) Displacement error with double integration for two users. (c) Error with step count method.

information we get. However, pattern recognition in computer area is not mature enough to work as human does. In this section, we will briefly introduce principle the pattern recognition.

A photo the camera records can be expressed as a matrix of light intensity of each pixel (here we take the grey-mode photo as an example). Suppose that the source matrixes (or as we call them dictionary) are  $A_{s1}, A_{s2}, \dots, A_{sn}$ , as the matrix needed to be recognized is  $A_d$ , then the pattern recognition is proceeded as (9) shows.  $j$ th matrix in the dictionary is the result:

$$j = \arg \max A_{sj} \cdot A_d. \quad (9)$$

Pattern recognition is far more complicated than (9); there are many good algorithms in pattern recognition area, like SIFT [19–23] and SVM [24–29]. But the recognition rate is still not good enough for practical applications.

### 3. Applications

In this section, we will introduce a few interesting sensing-based applications using smartphones. We divide the applications we are going to discuss into two categories: (1) accelerometer, gyroscope, and magnetometer; (2) microphone and camera.

#### 3.1. Accelerometer, Gyroscope, and Magnetometer

**3.1.1. Trace Track.** Searching people in a public place is difficult; for example, a person is in a conference hall, a library, or a shopping mall, there are many crowds around, it is very difficult to find the target person. Even if the person tells you where he/she is, it is frustrating to find the place in an unfamiliar environment. Maps may be helpful but not always handy. Smartphones provide possibilities to develop an electronic escort service using opportunistic user intersections. Through periodically learning the walking trails of different individuals, as well as how they encounter each other in space time, a route can be computed between any pair of persons [30]. The escort system could guide a user to the vicinity of a desired person in a public place. Escort does not rely on GPS, WiFi, or war driving to locate a person—the escort user only needs to follow an arrow displayed on the phone [30].

Escort system presents an interest idea that a smartphone is an escort which tells you how many steps you walk and which direction you are heading to [30], see Figure 5(a). GPS is a way to achieve the idea; however, GPS cannot work in an indoor environment; WiFi localization is a good indoor localization method; however, it cannot guarantee, there are enough WiFi fingerprints for localization. Escort system uses accelerometer and gyroscope to achieve the idea. However, the displacement calculated based on accelerometer is inaccurate; the reasons are the jerky movement of smartphone in people's pocket and inherent error of measurement [31–33]; the displacement error may reach 100 m after 30 m walk, see Figure 5(b). Escort identifies an acceleration signature in human walking patterns to avoid the problem. This signature arises from the natural up- and down-bounces of the human body while walking and can be used to count the number of steps walked [30]. The physical displacement can then be computed by multiplying step count with the user's step size which is, a function of the user's weight and height [30], see Figure 5(c). Escort varies the step size with an error factor drawn from a Gaussian distribution centered on 0 and standard deviation 0.15 m [30]. This better accommodates the varying human step size [30].

Compass (magnetometer) is also used in the escort system. Like accelerometer has measurement errors, the compass noise is caused by several factors, including user sway, movement irregularities, magnetic fields in the surroundings, and the sensor's internal bias. Because these factors are related to the user, surroundings, and the sensor, the noise is difficult to predict and compensate. To characterize the compass noise, escort ran 100 experiments using 2 Nokia 6210 Navigator phones and has observed an average bias of 8 degrees and a standard deviation of 9 degrees [30]. In addition to this large noise range, escort made two consistent observations as follows: (1) when the user is walking in a constant direction, the compass readings stabilize quickly on a biased value and exhibit only small random oscillations, and (2) after each turn, a new bias is imposed [30]. Escort identifies two states of the user: walking in a constant direction and turning based on the two observations. Turns are identified when the compass headings change more significantly than due to



random oscillations. The turn identification algorithm uses the following condition [30]:

$$\text{Avg}(t_{i+1} - \text{Avg}(t_i)) > \frac{\text{StdDev}(t_i) + \text{StdDev}(t_{i+1})}{2} + G, \quad (10)$$

where  $\text{Avg}(t_i)$  denotes the average compass readings over a  $t_i$  time period (e.g., 1 second),  $\text{StdDev}(t_i)$  is the standard deviation of compass readings during  $t_i$ , and  $G$  is a guard factor. While on a constant direction, escort compensates the stabilized compass reading with the average bias and reports the resulting value as the direction of the user. During turns, escort considers the sequence of readings reported by the compass [30].

Similar usage of accelerometer and magnetometer appears in [34]. It uses accelerometer to estimate the walking trace of people and correct the trace periodically by GPS. Accelerometer used as a supplement of GPS localization is a popular way, and many researches focused on this [35–40].

Smartphone is not only used on walking people but also on people on vehicles.

**3.1.2. Dangerous Drive.** When drivers are sitting in a vehicle, their smartphones are able to measure the acceleration, velocity, and turns through sensors embedded. Because the smartphone is very popular, using smartphone is an easy way to implement mobile sensing.

Driving style can be divided into two categories: nonaggressive and aggressive. To study the vehicle safety system, we need to understand and recognize the driving events. Potentially aggressive driving behavior is currently a leading cause of traffic fatalities in the United States, and drivers are unaware that they commit potentially aggressive actions daily [41]. To increase awareness and promote driver safety, a novel system that uses Dynamic Time Warping (DTW) and smartphone-based sensor fusion (accelerometer, gyroscope, magnetometer, GPS, and video) to detect, recognize, and record these actions without external processing being proposed [41].

The system collects the motion data from the the accelerometer and gyroscope in the smartphone continuously at a rate of 25 Hz in order to detect specific dangerous driving behaviors. Typical dangerous driving behaviors are hard left and right turns, swerves, and sudden braking and acceleration patterns. These dangerous driving behaviors indicate potentially aggressive driving that would cause danger to both pedestrians and other drivers. The system first determines when a dangerous behavior starts and ends using endpoint detection. Once the system has a signal representing a maneuver, it is possible to compare it to stored maneuvers (templates) to determine whether or not it matches an aggressive event [41].

References [42–44] use accelerometer and gyroscope for driving event recognition. Reference [42] includes two single-axis accelerometers, two single-axis gyroscopes, and a GPS unit (for velocity) attached to a PC for processing. While the system included gyroscopes for inertial measurements, they were not used in the project. Hidden Markov Models

(HMM) were trained and used only on the acceleration data for the recognition of simple driving patterns. Reference [43] used accelerometer data from a mobile phone to detect drunk driving patterns through windowing and variation threshold. A Driver Monitor System was created in [44] to monitor the driving patterns of the elderly. This system involved three cameras, a two-axis accelerometer, and a GPS receiver attached to a PC. The authors collected large volumes of data for 647 drivers. The system had many components, one of them being detection of erratic driving using accelerometers. Braking and acceleration patterns were detected, as well as high-speed turns via thresholding. Additionally, the data could be used to determine the driving environment (freeway versus city) based on acceleration patterns.

References [45–47] use accelerometer and gyroscope to detect gesture recognition. The uWave paper [47] and the gesture control work of Kela et al. [48] explore gesture recognition using DTW and HMM algorithms, respectively. They both used the same set of eight simple gestures which included up, down, left, right, two opposing direction circles, square, and slanted 90-degree angle movements for their final accuracy reporting. Four of these eight gestures are one-dimensional in nature. The results proved that using DTW with one training sample was just as effective as HMMs.

**3.1.3. Phone Hack.** Accelerometer is so sensitive that it can even detect finger press at the screen of the smartphone. Reference [49] shows that the location of screen taps on modern smartphones and tablets can be identified from accelerometer and gyroscope readings. The findings have serious implications, as we demonstrate that an attacker can launch a background process on commodity smartphones and tablets, and silently monitor the user's inputs, such as keyboard presses and icon taps [49]. While precise tap detection is nontrivial, requiring machine-learning algorithms to identify fingerprints of closely spaced keys, sensitive sensors on modern devices aid the process [49]. Reference [49] presents TapPrints, a framework for inferring the location of taps on mobile device touchscreens using motion sensor data combined with machine-learning analysis. By running tests on two different off-the-shelf smartphones and a tablet computer, the results show that identifying tap locations on the screen and inferring English letters could be done with up to 90% and 80% accuracy, respectively [49]. By optimizing the core tap detection capability with additional information, such as contextual priors, the core threat may further be magnified [49].

On both Android and iOS systems, location sensor like GPS is the only one which requires explicit user access permission, the reason probably is that people are not willing to be tracked and consider their locations information as privacy. But accelerometer and gyroscope sensors do not require explicit user permission on of the two mentioned operating systems. Because the sensors are mostly used for gaming, Android system does not restrict the access of accelerometer and gyroscope; background services with the two sensors are allowed. Moreover, there is work aimed at the standardization of JavaScript access to a device's accelerometer and gyroscope

sensors in order for any web application to perform, for example, website layout adaptation [49]. Reference [49] shows that accelerometer and gyroscope sensor data can create a serious privacy breach. In particular, it is demonstrated that it is possible to infer where people tap on the screen and what people type by applying machine-learning analysis to the stream of data from these two motion sensors. The reason the work focuses on the accelerometer and gyroscope sensors is that they are able to capture tiny device vibrations and angle rotations, respectively, with good precision [49].

Figure 6 shows a sample of the raw accelerometer data after a user has tapped on the screen—the timing of each tap marked by the digital pulses on the top line. As we can see, each tap generates perturbations in the accelerometer sensor readings on the three axes, particularly visible along the  $z$ -axis, which is perpendicular to the phone's display. Gyroscope data exhibits similar behavior and is not shown [49]. Some related works [50–52] are also about using accelerometer and gyroscope for phone hacks.

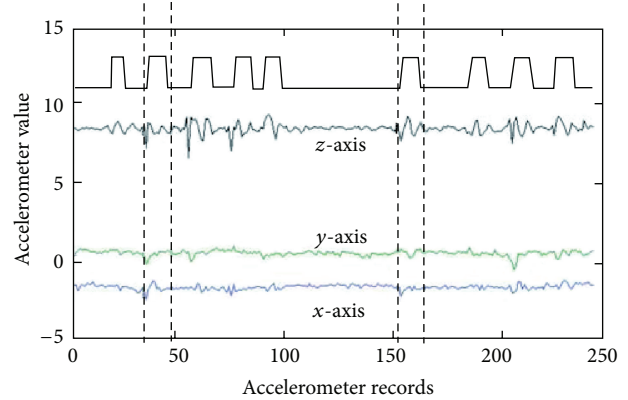


FIGURE 6: The square wave in the top line identifies the occurrence of a tap. Two particular taps have also been highlighted by marking their boundaries with dashed vertical lines. Notice that the accelerometer sensor readings (on the  $z$ -axis in particular) show very distinct patterns during taps. Similar patterns can also be observed in the gyroscope.

**3.1.4. Phone Gesture.** The first two sections are about using accelerometer, gyroscope, and magnetometer to detect a long-distance movement; actually, if the sensors are used to detect gestures, some interesting applications may appear.

The ability to note down small pieces of information, quickly and ubiquitously, can be useful. Reference [53] proposes a system called PhonePoint Pen that uses the in-built accelerometer in mobile phones to recognize human writing. By holding the phone like a pen, a user should be able to write short messages or even draw simple diagrams in the air. The acceleration due to hand gestures can be converted into an image and sent to the user's Internet email address for future reference [53].

The work is done without gyroscope, because the smartphone it used lacks gyroscope. There are two main issues to be solved: (1) coping with background vibration (noise); (2) computing displacement of phone.

Accelerometers are sensitive to small vibrations. Figure 7(a) reports acceleration readings as the user draws a rectangle using 4 strokes (around  $-350$  units on the  $z$ -axis is due to earth's gravity). A significant amount of jitter is caused by natural hand vibrations. Furthermore, the accelerometer itself has measurement errors. It is necessary to smooth this background vibration (noise) in order to extract jitter-free pen gestures. To cope with vibrational noise, the system smoothes the accelerometer readings by applying a moving average over the last  $n$  readings ( $n = 7$ ). The results are presented in Figure 7(b) (the relevant movements happen in the  $xy$ -plane, so we removed the  $z$ -axis from the subsequent figures for better visual representation) [53].

The phone's displacement determines the size of the character. The displacement  $\delta$  is computed as a double integral of acceleration, that is,  $\delta = \int (\int a dt) dt$ , where  $a$  is the instantaneous acceleration. In other words, the algorithm first computes the velocity (the integration of acceleration) followed by the displacement (the integration of velocity). However, due to errors in the accelerometer, the cumulative acceleration and deceleration values may not sum to zero

even after the phone has come to rest. This offset translates into some residual constant velocity. When this velocity is integrated, the displacement and movement direction become erroneous. In order to reduce velocity-drift errors, the velocity to zero at some identifiable points needs to be reset. The stroke mechanism described earlier is therefore used. Characters are drawn using a set of strokes separated by short pauses. Each pause is an opportunity to reset velocity to zero and thus correct displacement. Pauses are detected by using a moving window over  $n$  consecutive accelerometer readings ( $n = 7$ ) and checking if the standard deviation in the window is smaller than some threshold. This threshold is chosen empirically based on the average vibration caused when the phone was held stationary. All acceleration values during a pause are suppressed to zero. Figure 8(a) shows the combined effect of noise smoothing and suppression. Further, velocity is set to zero at the beginning of each pause interval. Figure 8(b) shows the effect of resetting the velocity. Even if small velocity drifts are still present, they have a small impact on the displacement of the phone [53].

## 3.2. Microphone and Camera

**3.2.1. Surround Sense.** There are some research works [54–57] about using smartphones to sense the context of surroundings. A growing number of mobile computing applications are centered around the user's location. The notion of location is broad, ranging from physical coordinates (latitude/longitude) to logical labels (like Starbucks, McDonalds). While extensive research has been performed in physical localization, there have been few attempts to recognize logical locations. Reference [57] argues that the increasing number of sensors on mobile phones presents new opportunities for logical localization. Reference [57] postulates that ambient sound, light, and color in a place convey a photoacoustic signature that can be sensed by the phone's camera and microphone. In-built accelerometers in some phones may

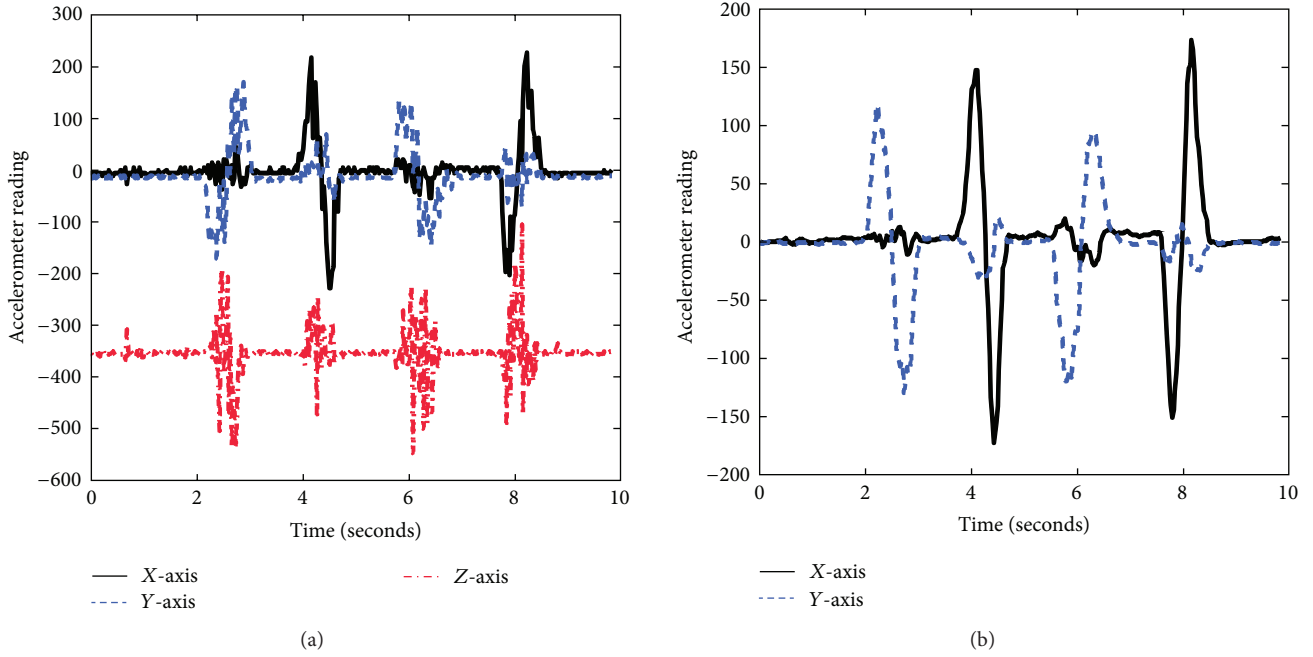


FIGURE 7: (a) Raw accelerometer data while drawing a rectangle (note gravity on the z-axis). (b) Accelerometer noise smoothing.

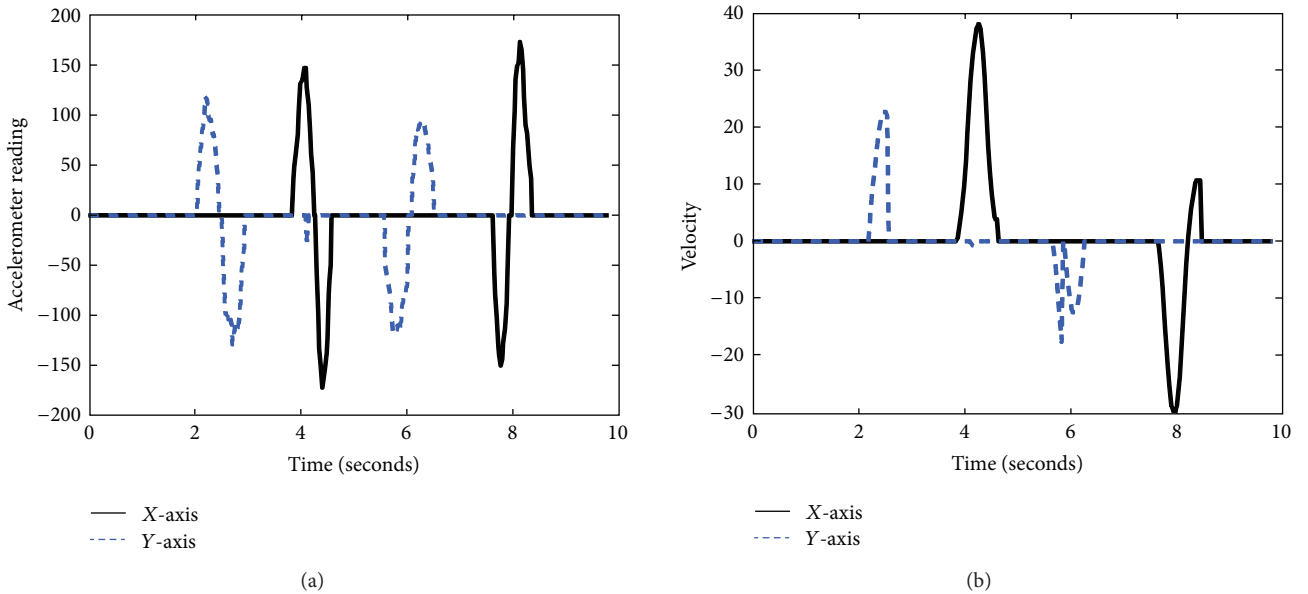


FIGURE 8: (a) Accelerometer readings after noise smoothing and suppression. (b) Resetting velocity to zero in order to avoid velocity drifts.

also be useful in inferring broad classes of user motion, often dictated by the nature of the place. By combining these optical, acoustic, and motion attributes, it may be feasible to construct an identifiable fingerprint for logical localization. Hence, users in adjacent stores can be separated logically, even when their physical positions are extremely close [57] see Figures 9 and 10.

Reference [57] takes advantage of microphone and camera to collect the surrounding fingerprint so that it can provide logical localization.

**3.2.2. Localization.** References [58–60] use audio for localization and the accuracy is improved to a high level. Reference [58] operates in a spontaneous, ad hoc, and device-to-device context without leveraging any preplanned infrastructure. It is a pure software-based solution and uses only the most basic set of commodity hardware—a speaker, a microphone, and some form of device-to-device communication—so that it is readily applicable to many low-cost sensor platforms and to most commercial-off-the-shelf mobile devices like cell phones and PDAs [58]. High accuracy is achieved

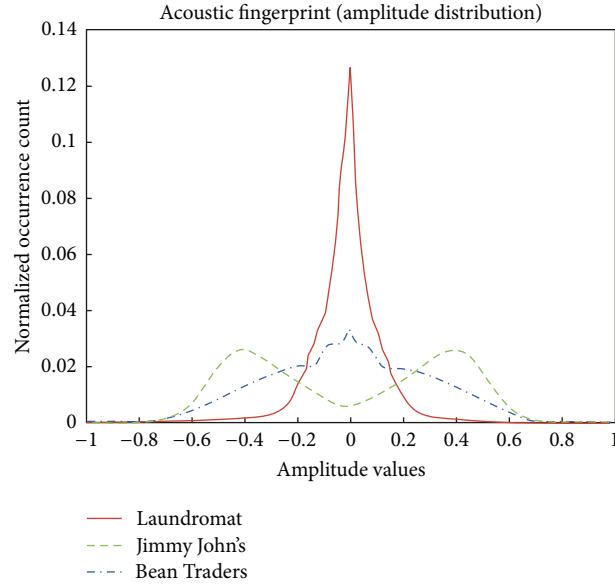


FIGURE 9: Sound fingerprints from 3 adjacent stores.

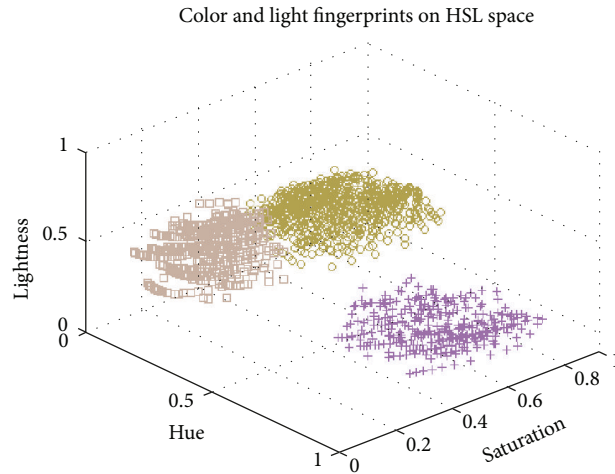


FIGURE 10: Color/light fingerprint in the HSL format from the Bean Traders' coffee shop. Each cluster is represented by a different symbol.

through a combination of three techniques: two-way sensing, self-recording, and sample counting. To estimate the range between two devices, each will emit a specially designed sound signal (“Beep”) and collect a simultaneous recording from its microphone [58]. Each recording should contain two such beeps, one from its own speaker and the other from its peer [58]. By counting the number of samples between these two beeps and exchanging the time duration information with its peer, each device can derive the two-way time of flight of the beeps at the granularity of the sound sampling rate [58]. This technique cleverly avoids many sources of inaccuracy found in other typical time-of-arrival schemes, such as clock synchronization, non-real-time handling, and software delays [58].

Reference [58] sends beep sound to calculate the distance between two objects; the microphone will distinguish original sound and echo to get the time interval and, therefore, get a distance, see Figure 11.

#### 4. Discussions and Conclusion

Sensors are the key factors of developing more and more interesting applications on the smartphones, and the sensors make the smartphone different from traditional computing devices like computer. Most applications used accelerometer and gyroscope because they are somehow the most accurate sensors. However, the vision contains huge information. We



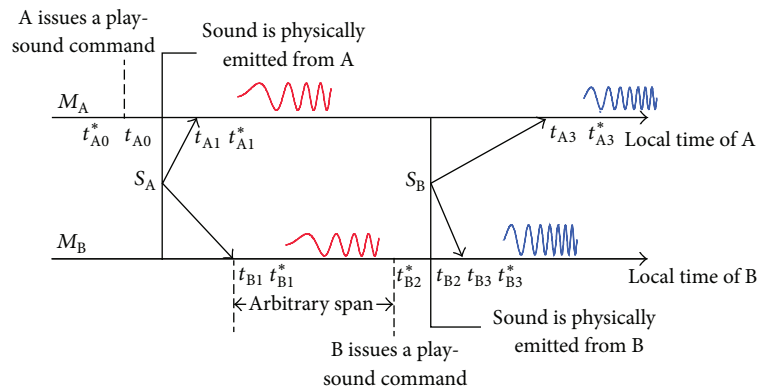


FIGURE 11: Illustration of event sequences in BeepBeep ranging procedure. The time points are marked for easy explanation and no timestamping is required in the proposed ranging mechanism.

believe that camera and pattern recognition will be used more and more in the future.

## Acknowledgments

This work is supported by National Science Foundation under Grant nos. 61103226, 60903158, 61170256, 61173172, and 61103227 and the Fundamental Research Funds for the Central Universities under Grant no. ZYGX2010J074 and ZYGX2011J102.

## References

- [1] H. Ishida, K. Suetsugu, T. Nakamoto, and T. Moriizumi, "Study of autonomous mobile sensing system for localization of odor source using gas sensors and anemometric sensors," *Sensors and Actuators A*, vol. 45, no. 2, pp. 153–157, 1994.
- [2] H. Ishida, T. Nakamoto, and T. Moriizumi, "Remote sensing of gas/odor source location and concentration distribution using mobile system," *Sensors and Actuators B*, vol. 49, no. 1-2, pp. 52–57, 1998.
- [3] S. B. Eisenman, E. Miluzzo, N. D. Lane, R. A. Peterson, G. S. Ahn, and A. T. Campbell, "BikeNet: a mobile sensing system for cyclist experience mapping," *ACM Transactions on Sensor Networks*, vol. 6, no. 1, pp. 1–39, 2009.
- [4] T. Choudhury, G. Borriello, S. Consolvo et al., "The mobile sensing platform: an embedded activity recognition system," *IEEE Pervasive Computing*, vol. 7, no. 2, pp. 32–41, 2008.
- [5] B. Lo, S. Thiemjarus, R. King et al., "Body sensor network—a wireless sensor platform for pervasive healthcare monitoring," in *Proceedings of the 3rd International Conference on Pervasive Computing*, 2005.
- [6] X. Tant, D. Kim, N. Usher et al., "An autonomous robotic fish for mobile sensing," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '06)*, pp. 5424–5429, October 2006.
- [7] A. Arora, P. Dutta, S. Bapat et al., "A line in the sand: a wireless sensor network for target detection, classification, and tracking," *Computer Networks*, vol. 46, no. 5, pp. 605–634, 2004.
- [8] Y. C. Tseng, S. P. Kuo, and H. W. Lee, "Location tracking in a wireless sensor network by mobile agents and its data fusion strategies," *Information Processing in Sensor Networks*, pp. 554–554, 2003.
- [9] G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh, "Monitoring volcanic eruptions with a wireless sensor network," in *Proceedings of the 2nd European Workshop on Wireless Sensor Networks (EWSN '05)*, pp. 108–120, February 2005.
- [10] G. Werner-Allen, K. Lorincz, M. Welsh et al., "Deploying a wireless sensor network on an active volcano," *IEEE Internet Computing*, vol. 10, no. 2, pp. 18–25, 2006.
- [11] A. T. Campbell, S. B. Eisenman, N. D. Lane et al., "The rise of people-centric sensing," *IEEE Internet Computing*, vol. 12, no. 4, pp. 12–21, 2008.
- [12] Accelerometer, <http://en.wikipedia.org/wiki/Accelerometer>.
- [13] A Guide To using IMU (Accelerometer and Gyroscope Devices) in Embedded Applications, [http://www.starlino.com/imu\\_guide.html](http://www.starlino.com/imu_guide.html).
- [14] M. Arraigada and M. Partl, "Calculation of displacements of measured accelerations, analysis of two accelerometers and application in road engineering," in *Proceedings of the Swiss Transport Research Conference*, 2006.
- [15] Gyroscope, <http://en.wikipedia.org/wiki/Gyroscope>.
- [16] Magnetometer, <http://en.wikipedia.org/wiki/Magnetometer>.
- [17] S. Sen, R. Roy Choudhury, and S. Nelakuditi, "CSMA/CN: carrier sense multiple access with collision notification," *IEEE/ACM Transactions on Networking*, vol. 20, no. 2, pp. 544–556, 2012.
- [18] Cross-correlation, <http://en.wikipedia.org/wiki/Cross-correlation>.
- [19] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [20] H. Bay, T. Tuytelaars, and L. Gool, "SURF: speeded up robust features," in *Proceedings of the Computer Vision (ECCV '06)*, pp. 404–417, Springer, Berlin, Germany, 2006.
- [21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 886–893, June 2005.
- [22] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.
- [23] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-Up Robust Features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

- [24] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [25] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and other Kernel-Based Learning Methods*, Cambridge University Press, 2000.
- [26] C. C. Chang and C. J. Lin, "LIBSVM: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, article 27, 2011.
- [27] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.
- [28] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, ACM press, New York, NY, USA, 1999.
- [29] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, 2001.
- [30] I. Constandache, X. Bao, M. Azizyan, and R. R. Choudhury, "Did you see Bob?: human localization using mobile phones," in *Proceedings of the 16th Annual Conference on Mobile Computing and Networking (MobiCom '10)*, pp. 149–160, September 2010.
- [31] D. M. Boore, "Effect of baseline corrections on displacements and response spectra for several recordings of the 1999 Chi-Chi, Taiwan, earthquake," *Bulletin of the Seismological Society of America*, vol. 91, no. 5, pp. 1199–1211, 2001.
- [32] D. M. Boore, C. D. Stephens, and W. B. Joyner, "Comments on baseline correction of digital strong-motion data: examples from the 1999 Hector Mine, California, earthquake," *Bulletin of the Seismological Society of America*, vol. 92, no. 4, pp. 1543–1560, 2002.
- [33] D. M. Boore, "Analog-to-digital conversion as a source of drifts in displacements derived from digital recordings of ground acceleration," *Bulletin of the Seismological Society of America*, vol. 93, no. 5, pp. 2017–2024, 2003.
- [34] I. Constandache, R. R. Choudhury, and I. Rhee, "Towards mobile phone localization without war-driving," in *Proceedings of the IEEE INFOCOM*, pp. 1–9, March 2010.
- [35] J. Paek, J. Kim, and R. Govindan, "Energy-efficient rate-adaptive GPS-based positioning for smartphones," in *Proceedings of the 8th Annual International Conference on Mobile Systems, Applications and Services (MobiSys '10)*, pp. 299–314, June 2010.
- [36] D. H. Kim, Y. Kim, D. Estrin, and M. B. Srivastava, "SensLoc: sensing everyday places and paths using less energy," in *MobiSys 8th ACM International Conference on Embedded Networked Sensor Systems (SenSys '10)*, pp. 43–56, November 2010.
- [37] K. Lee, I. Rhee, J. Lee, S. Chong, and Y. Yi, "Mobile data offloading: how much can WiFi deliver?" in *Proceedings of the 6th International Conference on Emerging Networking Experiments and Technologies (Co-NEXT '10)*, p. 26, December 2010.
- [38] A. Thiagarajan, J. Biagioni, T. Gerlich, and J. Eriksson, "Cooperative transit tracking using smart-phones," in *Proceedings of the 8th ACM International Conference on Embedded Networked Sensor Systems (SenSys '10)*, pp. 85–98, November 2010.
- [39] A. Schulman, V. Navda, R. Ramjee et al., "Bartendr: a practical approach to energy-aware cellular data scheduling," in *Proceedings of the 16th Annual Conference on Mobile Computing and Networking (MobiCom '10)*, pp. 85–96, September 2010.
- [40] S. P. Tarzia, P. A. Dinda, R. P. Dick, and G. Memik, "Indoor localization without infrastructure using the acoustic background spectrum," in *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys '11)*, pp. 155–168, July 2011.
- [41] D. A. Johnson and M. M. Trivedi, "Driving style recognition using a smartphone as a sensor platform," in *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems (ITSC '11)*, pp. 1609–1615, 2011.
- [42] D. Mitrović, "Reliable method for driving events recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 2, pp. 198–205, 2005.
- [43] J. Dai, J. Teng, X. Bai, Z. Shen, and D. Xuan, "Mobile phone based drunk driving detection," in *Proceedings of the 4th International Conference on Pervasive Computing Technologies for Healthcare*, pp. 1–8, March 2010.
- [44] K. C. Baldwin, D. D. Duncan, and S. K. West, "The driver monitor system: a means of assessing driver performance," *Johns Hopkins APL Technical Digest*, vol. 25, no. 3, pp. 269–277, 2004.
- [45] G. Ten Holt, M. Reinders, and E. Hendriks, "Multi-dimensional dynamic time warping for gesture recognition," in *Proceedings of the Conference of the Advanced School for Computing and Imaging (ASCI '07)*, 2007.
- [46] R. Muscillo, S. Conforto, M. Schmid, P. Caselli, and T. D'Alessio, "Classification of motor activities through derivative dynamic time warping applied on accelerometer data," in *Proceedings of the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '07)*, pp. 4930–4933, 2007.
- [47] J. Liu, L. Zhong, J. Wickramasuriya, and V. Vasudevan, "uWave: accelerometer-based personalized gesture recognition and its applications," *Pervasive and Mobile Computing*, vol. 5, no. 6, pp. 657–675, 2009.
- [48] J. Kela, P. Korpipää, and J. Mäntyjärvi, "Accelerometer-based gesture control for a design environment," *Personal and Ubiquitous Computing*, vol. 10, no. 5, pp. 285–299, 2006.
- [49] E. Miluzzo, A. Varshavsky, S. Balakrishnan et al., "Tapprints: your finger taps have fingerprints," in *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, pp. 323–336, Low Wood Bay, Lake District, UK, 2012.
- [50] L. Cai and H. H. Chen, "TouchLogger: inferring keystrokes on touch screen from smartphone motion," in *Proceedings of the 6th USENIX Conference on Hot Topics in Security*, p. 9, San Francisco, Calif, USA, 2011.
- [51] E. Owusu, J. Han, S. Das et al., "ACCessory: password inference using accelerometers on smartphones," in *Proceedings of the 12th Workshop on Mobile Computing Systems & Applications*, pp. 1–6, San Diego, Calif, USA, 2012.
- [52] L. Cai, S. Machiraju, and H. Chen, "Defending against sensor-sniffing attacks on mobile phones," in *Proceedings of the 1st ACM Workshop on Networking, Systems, and Applications for Mobile Handhelds*, pp. 31–36, Barcelona, Spain, 2009.
- [53] S. Agrawal, I. Constandache, and S. Gaonkar, "PhonePoint pen: using mobile phones to write in air," in *Proceedings of the 1st ACM Workshop on Networking, Systems, and Applications for Mobile Handhelds*, pp. 1–6, Barcelona, Spain, 2009.
- [54] B. Clarkson, K. Mase, and A. Pentland, "Recognizing user context via wearable sensors," in *Proceedings of the 4th International Symposium on Wearable Computers*, pp. 69–75, October 2000.
- [55] S. Gaonkar, J. Li, and R. R. Choudhury, "Micro-Blog: sharing and querying content through mobile phones and social participation," in *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services*, pp. 174–186, Breckenridge, Colo, USA, 2008.
- [56] E. Miluzzo, N. D. Lane, and K. Fodor, "Sensing meets mobile social networks: the design, implementation and evaluation

- of the CenceMe application,” in *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, pp. 337–350, Raleigh, NC, USA, 2008.
- [57] M. Azizyan and R. R. Choudhury, “SurroundSense: mobile phone localization using ambient sound and light,” *SIGMOBILE Mobile Computing and Communications Review*, vol. 13, no. 1, pp. 69–72, 2009.
- [58] C. Peng, G. Shen, Y. Zhang, Y. Li, and K. Tan, “BeepBeep: a high accuracy acoustic ranging system using COTS mobile devices,” in *Proceedings of the 5th ACM International Conference on Embedded Networked Sensor Systems (SenSys '07)*, pp. 1–14, Sydney, Australia, November 2007.
- [59] A. Mandai, C. V. Lopes, T. Givargis, A. Haghighat, R. Jurdak, and P. Baldi, “Beep: 3D indoor positioning using audible sound,” in *Proceedings of the 2nd IEEE Consumer Communications and Networking Conference (CCNC '05)*, pp. 348–353, January 2005.
- [60] C. Peng, G. Shen, Y. Zhang, Y. Li, and K. Tan, “BeepBeep: a high accuracy acoustic ranging system using COTS mobile devices,” in *Proceedings of the 5th ACM International Conference on Embedded Networked Sensor Systems (SenSys '07)*, pp. 397–398, Sydney, Australia, November 2007.



