INTERNATIONAL JOURNAL ON SMART SENSING AND INTELLIGENT SYSTEMS VOL. 9, NO. 4, DECEMBER 2016

www.s2is.org
International Journal on Smart Sensing and Intelligent Systems
ISSN 1178-5608

SMARTPHONE-BASED ENVIRONMENTAL SENSING USING DEVICE LOCATION AS METADATA

Kaori Fujinami
Graduate School of Engineering
Department of Computer and Information Sciences
Tokyo University of Agriculture and Technology
2-24-16 Naka-cho, Koganei, Tokyo, Japan
Emails: fujinami@cc.tuat.ac.jp

Submitted: July 5, 2016 ***Accepted: Nov. 3, 2016*** ***Published: Dec. 1, 2016***

Abstract- The people-centric sensing community is paying substantial attention to the smartphone as an ad hoc, low-cost, and dense sensing method because it permits people to participate easily in sensing activities, i.e., just by carrying it as usual. People carry their smartphones in various manners, rendering measurement results unreliable. For example, humidity is typically higher in a trouser pocket than around the neck as a result of sweat. In this article, we propose a platform for people-centric sensing that considers the on-body position of a smartphone as metadata. A general architecture is presented, and a universal serial bus-based external sensing module for an Android-based terminal is developed. A heatstroke alert map that visualizes the heatstroke risk is presented as an application based on both the collected raw data and metadata using the platform.

Index terms: environmental sensing, people-centric sensing, heatstroke, smartphone, on-body device localization.

I. INTRODUCTION

The recent advancement of technologies such as microelectromechanical systems (MEMS) with high performance and low-power computation have enabled the development of smartphones—mobile telephones augmented with various sensors. The integration of such sensors in a smartphone permits distributed sensing, ranging from individual sensing for personal use to people-centric sensing for public use [4][12][21][24][25][26][27][29][38]. Such smartphone-based sensing enables low-cost and geographically high-density sensing; however, the nature of a smartphone as an object of daily use, i.e., ease of carrying, poses a problem, especially in the case in which a person carries a smartphone directly. According to a study of phone carrying, 17% of people determine the position for storing a smartphone based on contextual restrictions, e.g., no pocket in the T-shirt, too large a phone size for a trouser pocket, or comfort during an ongoing activity [6]. These factors vary throughout the day; thus, users change the positions frequently. This suggests that a sensor embedded in a smartphone might generate meaningless data if a person carries the smartphone. Meaningless data cause unreliable sensing and waste the computational power of a terminal as well as the network bandwidth. For example, a smartphone (with temperature and humidity sensors) stored in a trouser pocket must not be used to sense heatstroke risk because the risk level calculated from the temperature and humidity data tends to be underestimated as a result of body heat and perspiration [38]. This suggests that a context, the *on-body position of a sensor*, has high potential for improving the quality of sensor-dependent services. In the paradigm of people-centric sensing, the storing position of a smartphone terminal is considered as a key context for reliable measurement [18].

In this article, we propose a people-centric sensing platform that uses the positional information of a sensor node (smartphone) as metadata for improving the quality of sensing. We develop an external sensing module that is connected to an Android smartphone terminal to enable the measurement of a variety of environmental information in everyday use. A software framework on an Android terminal is also presented. The remainder of the article is organized as follows. Section II examines related work in terms of people-centric sensing with a smartphone. The necessity of taking into account the position of a sensor on the body is pointed out in Section III, and a framework for a reliable people-centric sensing system is proposed. Our proposed system—Tiny, Adjunctive, and Lightweight Environmental Sensing for Extending Android (TALESEA), a universal serial bus (USB)-based sensor module for an Android-based terminal—is developed in Section IV. Section V describes a map visualization of the heatstroke risk level as an application. Finally, Section VI concludes the article.

II. RELATED WORK

Smartphone-based environmental sensing is receiving attention owing to the popularity of smartphones and their associated communication infrastructure [4][12][18]. There are two types of sensing using smartphones, moving object-based and pedestrian-based. In moving object-based sensing, a smartphone is attached to an object such as a car or bicycle [29][5][24][34][26], ensuring that the placement of the smartphone can be restricted to predefined positions, such as the dashboard or handle, although diversity in the types of car and bicycle must still be taken into account. Alternatively, a smartphone can be carried directly by a person in pedestrian-based sensing [19][22][25][27]. In that case, the high degree of carrying freedom can degrade the quality of the sensed data in the absence of proper handling of the body's positional information.
The importance of sensor context, such as the position of the sensor on the body, in ensuring data reliability was pointed out by Lane et al. [18]. NoiseTube [19] is a smartphone-based noise level-sensing system and the possibility of the normal use of a mobile phone conflicting with the use of a noise level meter has been mentioned in connection with it. For example, carrying a phone terminal in a trouser pocket might provide measurement results different from those of a terminal hanging from the neck. In [23], environmental states of a city such as carbon dioxide (CO_2) concentration, temperature, humidity, and air pressure were visualized using a data collection module attached to a smartphone; however, the visualization might have contained data from inappropriate sensing conditions resulting from the storing position of the microphone, i.e., the smartphone, that were not taken into account. In people-centric sensing, information regarding positioning on the earth, e.g., latitude, longitude, and orientation, is usually captured using a Global Positioning System (GPS) receiver, compass, and gyroscope, along with the target sensor data. It is interesting that the storing position has been found to affect even these sensors [3][36]. These facts suggest that the position of a sensor (smartphone) on the body must be taken into account in a reliable people-centric sensing system.

On-body localization is a topic of broad and current interest for researchers in the wearable and ubiquitous computing communities. Supported positions range from the head to the ankle in the case of wearable sensors [20][32][35], whereas smartphone-based approaches deal with various “container” positions, such as a jacket, chest, or trouser pocket, bags, and the user’s hand [15][11][1][8]. Intended applications include activity recognition and measurement [11][15] [20][32], notification modality control [11][8], and reliable environmental sensing [11][22][38]. Miluzzo et al. proposed a microphone-based on-body localization method for environmental sensing, in which the simple placement of “inside pocket” or “outside pocket” was the characteristic to detect [22]. We have also investigated a method for identifying nine typical positions on the body, including bags [11], as well as an application framework [10]. We consider that such detailed localization could improve the quality of data substantially. In this article, we show an example of the need for detailed position sensing, and we propose a framework for a reliable people-centric sensing system using the application framework.

III. SMARTPHONE-BASED RELIABLE ENVIRONMENTAL SENSING

a. Issues in Environmental Sensing Using Smartphones

As described above, people carry their smartphones in various manners. We can imagine that the temperature and humidity sensor readings might differ among various storing positions on the body, e.g., the neck or a chest or trouser pocket. Here we show an example to understand the dependency of temperature and humidity sensor readings on storing positions. Four body positions of a sensor that are popular for storing a smartphone were selected, front or back trouser pocket, chest pocket, and around the neck (hanging). Note that “hanging from the neck” has a special significance, because the sensor can measure the ideal air condition and thus utilize it in the comparison. The data were collected from at most six participants who stored sensors in the four positions in 20–30 minutes of walking on a road paved with asphalt in August 2011. Tiny data loggers (SHTDL-1/2 [31]) were used in the data collection.

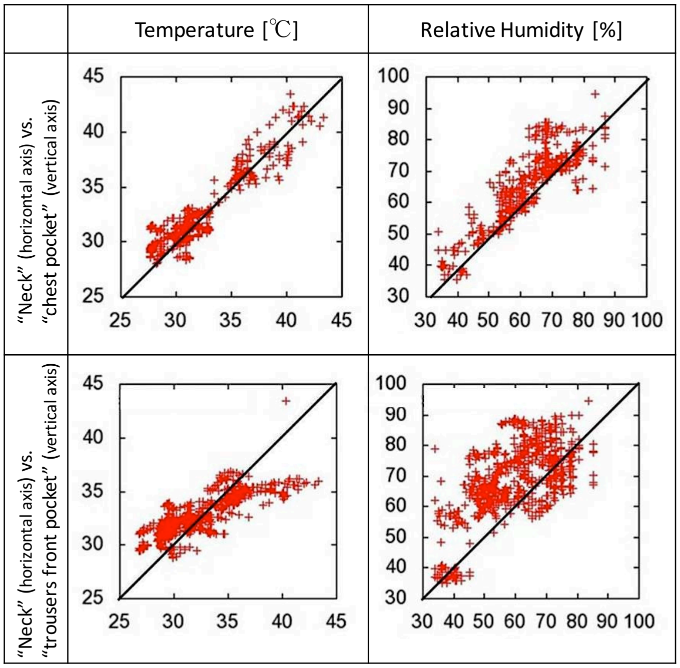


Figure 1. Difference of measurements in storing positions. (Left: neck vs. chest pocket. Left: neck vs. front pocket of trousers.) Note that "neck" is an ideal sensing condition owing to openness to the air.

The differences between the ideal measurement condition and the others are shown by scatter plots, in which the horizontal axis indicates the measurement from a sensor hanging from the neck, and the vertical axis represents the measurements from the other positions, i.e., from a sensor in a chest pocket (Figure 1 top) and a front trouser pocket (bottom). The plots close to the diagonal line indicate that the measurements from the neck and the chest pocket (or front trouser pocket) differ little. These figures show that the difference between the chest pocket and neck is small; however, the difference between trouser pockets and neck is large. We consider that this is because the outside temperature is very high, and a trouser pocket shades the sensor from the sun's heat. On the other hand, the heat might not be shaded in the chest pocket, since a chest pocket is nearly exposed to the air. These positional dependencies might degrade the quality of analysis of the collected data.

b. System Architecture

To address the issue, we propose a people-centric sensing system that allows explicit handling of the placement of a sensor node (smartphone). Here, we focus on opportunistic sensing, rather than participatory sensing, because people are usually aware of the position of a terminal in participatory sensing, e.g., holding in a hand to take a picture. The top-level functionalities of the system, sensing, sensor data sharing, and applications that use the collected information, are illustrated in Figure 2. A sensing node is a smartphone with an additional sensing module consisting of target sensors such as temperature and humidity sensors, on-body device localization functionality, and position-handling components.

The device position can be handled in three manners, automatic, manual, and semi-automatic. Automatic handling involves calibrating the measured data. Calibration requires defining the formula by collecting large numbers of data pairs from source and target positions. In the case in which such data collection is infeasible, an alternative solution is to filter out data from desired positions. In the temperature and humidity sensor example, data obtained from a trouser front pocket are not used, whereas measurements from the neck or chest pocket can be used. A similar approach was used in outside noise pollution monitoring [27], by filtering out sound data that were believed to be obtained indoors, while moving at high speed, or stored in pockets. An issue in this approach is that the number of available data might be reduced. At the other end of the data-handling continuum, the system can ask a user to put his or her device at a designated position, e.g., hanging from the neck. However, this is obtrusive for the user if the position is not one typically used. In this case, strong motivation must be provided to the user to follow instructions from the system. Semi-automatic handling is between automatic and manual handling in that the final decision is entrusted to the user by providing information regarding the storing position and possible risk of the current position. Which of the three handling methods is used depends on the policy of the application designer. Thus, the platform is designed to provide positional information as metadata of the target sensing data via application programming interfaces (APIs) to achieve flexible handling.

A prototype system was implemented for Android OS, on which a component of on-body device localization runs as an Android OS service [10]. The implemented component classifies five popular storing positions on the body, neck, front or back trouser pocket, and chest or jacket pocket, with an accuracy of 72.3%. Since the classification component is loosely coupled with the handling component, the localization algorithm is easily replaced or updated with more sophisticated versions appearing later [11].

The processed or raw data with positional information are sent to a data sharing service on the Internet such as xively[1] [37] in conjunction with global coordinates and timestamps. The shared data are analyzed further to understand the current environmental states in a particular area or predict future states. Heat map visualization is often used as a consumer application. In the future, pedestrian or cyclist navigation in which a route is suggested to a user to avoid heatstroke or sunburn might be provided based on the collected data.

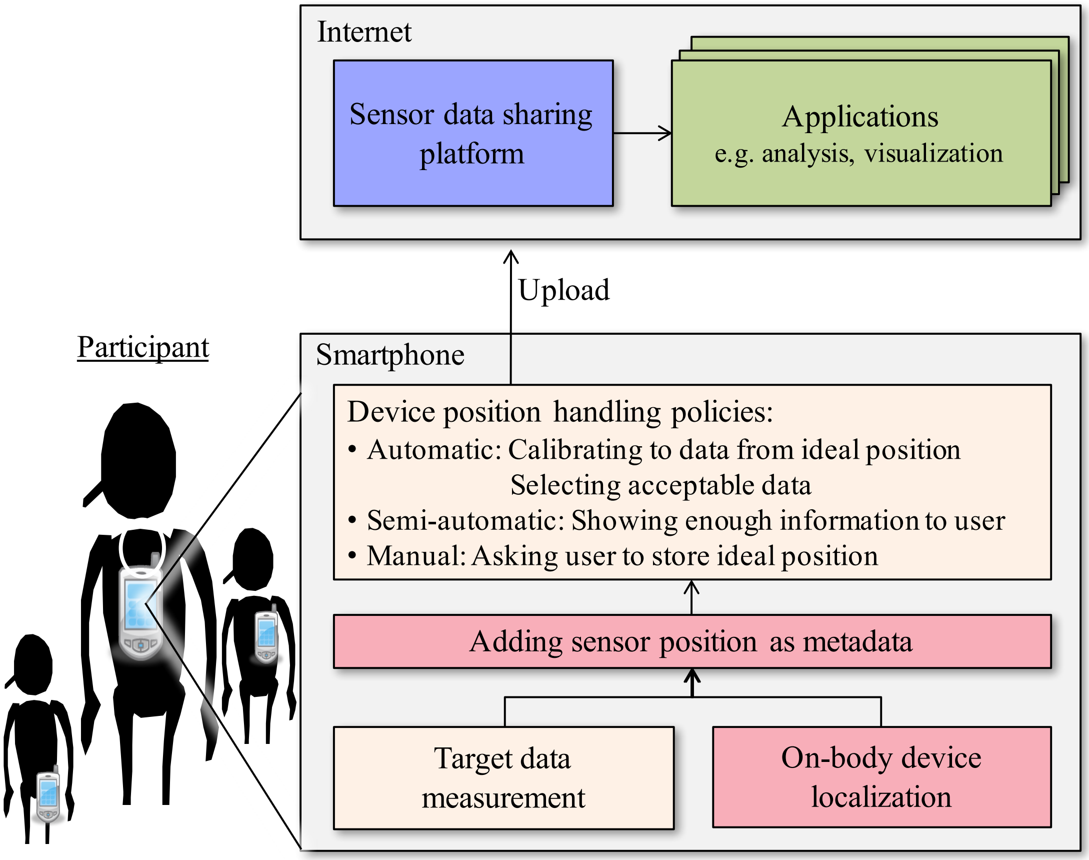


Figure 2. Basic framework of people-centric sensing that considers the on-body position of a sensor.

---

[1] Formerly known as Pachube or COSM.

IV. TALESEA: SENSING MODULE FOR ANDROID-BASED SMARTPHONE SENSING

We describe the design and implementation of a sensor module for Android-based smartphone sensing.

a. Motivation

Smartphones currently use a wide range of sensing modalities such as audio, proximity, light level, and magnetic fields; however, environmental sensors for temperature, humidity, CO_2, and ultraviolet have not yet been popular for smartphones. TALESEA was developed to accelerate smartphone-based environmental sensing.

b. Hardware Configuration

TALESEA consists of a micro-controller (Arduino Pro mini (8 MHz, 3.3 V)) connected to an Android terminal via a USB. Any sensor and output element can be connected to the micro-controller. A temperature and relative humidity sensor module (Sensirion Inc.'s SHT-71 [28]) was integrated as the default sensor. Furthermore, two surface-mounted light-emitting diodes (LEDs) were embedded to communicate the status of TALESEA to the user. Additionally, six analog inputs and five digital inputs/outputs were available for general-purpose input/output (GPIO). Figure 3 shows the appearance of TALESEA attached to Samsung's Galaxy NEXUS. The dimensions are W48 × H15 × D30 mm, and the weight is 19 g.

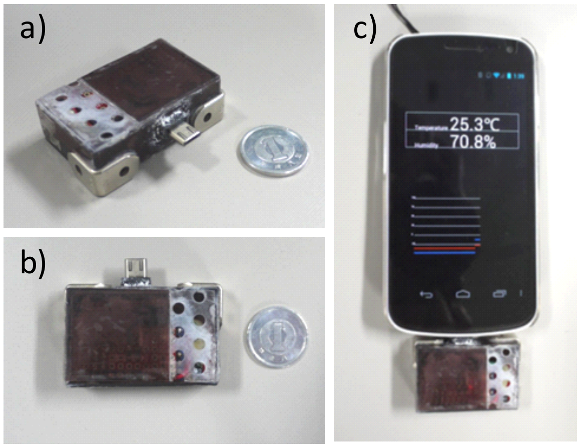


Figure 3. Appearance of TALESEA ((a) and (b))
and an Android terminal connected to TALESEA (c).

c. Software Components

Figure 4 illustrates the software components on the TALESEA side and the Android-based terminal. TALESEA communicates with the Android device using a USB-API that supports Android OS 3.1 or later. TALESEA Driver Service, running in the background as an Android Service, sends commands to Arduino Controller. Commands include a sensing request, an output request, e.g., turning on/off an on-board LED, rotating a motor by 30 degrees, and requests for referring/updating internal states of TALESEA. In the implementation, Arduino Controller reads values from temperature and humidity sensor ports at 1 Hz, and averaged values are sent to TALESEA Driver in response to a sensing command issued by TALESEA Driver every ten seconds. Then, TALESEA Driver updates a record in a database (SQLite) with the latest value from TALESEA. The measured data are shared with various applications asynchronously via the database. Applications include not only the "sensor position handling" one, but also any type of application that uses positional information, e.g., the data viewer shown in Figure 3(c). Here, the TALESEA API encapsulates the Android Contents Provider API from application developers.

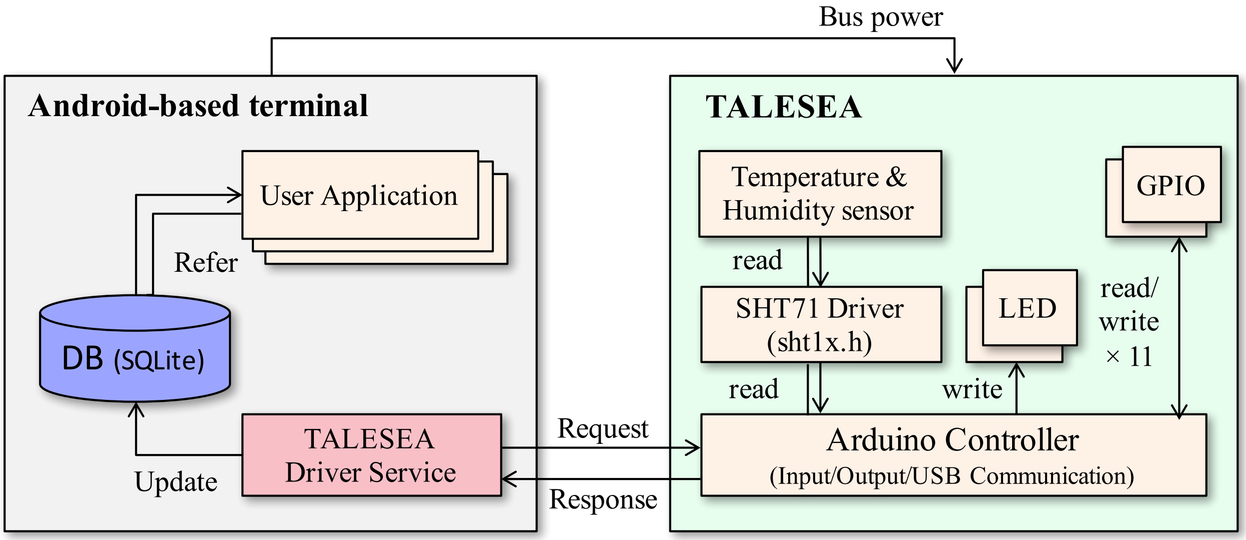


Figure 4. Software components for TALESEA-based system.

d. Power Consumption of the TALESEA Module

In opportunistic sensing in the people-centric sensing paradigm, sensing is conducted in the background for a long period of time. Thus, power consumption is an important issue for people participating in the sensing campaign. We measured the power consumption of the TALESEA sensing module and analyzed critical points. We used a Galaxy Nexus (CPU: OMAP4460, 1.2 GHz, Battery: 1750 mAh), for which the power consumption of five components was recorded using a program based on the Android API that logs the remaining power in a file as a percentage. The components included (1) a power supply for the TALESEA sensing module, (2) sensing data polling from the Android terminal to the TALESEA sensing module (0.1 Hz), (3) GPS measurement (50 Hz), (4) uploading to a Pachube-based data sharing service on the Internet over a 3G communication channel, and (5) on-body device localization (25 Hz).

The power consumption as a percentage after the system runs for one hour is 37.0%, indicating that if we assume a linearly increasing power consumption, it takes about three hours to deplete the charge. Then, to perform a more detailed analysis, we measured the power consumption of each component individually by letting them work separately. Table 1 summarizes the results. The table shows that the power supply to the TALESEA sensing module used most of the power consumption, followed by on-body device localization and data uploading. Furthermore, to see more detail in the power supply to the TALESEA sensing module, we investigated the consumption current of the elements from their datasheet. As a result, the microcontroller (Arduino Pro mini 3.3 V, 8 MHz) consumed the most current (40 mA per I/O pin), while those of the FT232RL-based USB serial adapter and SHT-71 temperature/humidity sensor were 50 mA (maximum) and 0.55 mA, respectively.

Table 1: Power consumption of TALESEA-based device position-aware data collection system

Functionality	**Power consumption [%] per hour**
Power supply to TALESEA sensing module	13.7
Communication with Android	0.1
GPS measurement	5.5
Uploading to server on the Internet	8.5
On-body device localization	9.2

The drain on the power supply can be reduced by using a “sleep mode” supported by Arduino, which is achieved by limiting some functions of Arduino and resuming functionality through an external interrupt. In contrast to acceleration signals, temperature and humidity sensing does not require a high sampling frequency. Therefore, the module can be put into “sleep mode” between measurements and conduct sensing triggered by a timer interrupt. In addition, the power consumption of the USB serial adapter is a potential bottleneck in the power supply to the TALESEA sensing module. Not only the sleep of the microcontroller itself, but also the sleep of the USB controller must be considered. Furthermore, we consider that context-aware power saving can be achieved. For example, depending on the application, there can be a case in which the current device's position and activity render the measurement useless. The module can go into "sleep mode" once it detects such a situation.

On-body localization is the second-highest power consuming functionality, as shown in Table 1. This is because the component was not tuned for power consumption, but for recognition performance. As shown by Krause et al., battery life is affected by factors such as feature domains, i.e., time domain or frequency domain, size of the feature calculation window, or sampling rate [17]. Careful selection of these parameters is required, taking into account the accuracy of localization. The power consumption related to GPS measurement and uploading to the server can be reduced by suppressing the number of measurements by collaborating with nearby terminals [33], limiting to road segments [14], and limiting periods of moving [40]. In addition, combining low-power consumption localization techniques with GPS-based positioning contributes to saving energy [16].

V. MAP VISUALIZATION OF HEATSTROKE RISK LEVEL: AN APPLICATION

Based on the framework, we implemented a map visualization system for heatstroke alert as an application.

a. Motivation of Sharing Heatstroke Risk Level

In recent years, the number of people suffering from heatstroke has increased. Heatstroke occurs as a result of thermoregulatory dysfunction under heat stress. According to a survey by the Fire and Disaster Management Agency of Japan, 55,852 people across the country in the period of May to September of 2015 were transported to the hospital, and in the worst case also died (0.2% of patients) [9]. Various consumer products for heatstroke risk alert are available on the market [7]. However, the available information concerns the level of heatstroke risk at the moment, which indicates just a "point" rather than an "area," hence it is difficult for people to find the route to the station with the lowest risk, for example. On the other hand, a web application can provide a forecast of the level of risk at any location based on the information from a weather station; however, the spatial resolution of the information is limited to the city level. Therefore, combining the local measurements with the global position has the potential to improve the spatial and time resolutions, and allow querying the risk level at any location. The information can be used not only to provide people with safe walking/biking paths, but also to assist administrative agencies in future urban planning.

b. Leveraging the Positional Information of Measurement Devices

Wet-bulb globe temperature (WBGT) is considered to be the most informative index for environmental thermal conditions that reflect the probability of heatstroke, because it consists of three important factors, air temperature, air humidity, and radiant heat [39]. However, consumer portable devices usually use an approximate formula [13] without the radiant heat term, as shown below, owing to the large form factor of a global thermometer (see Equations (1) and (2)).

$$WBGT = 0.567 \times T_a + 0.393 \times E(T_a, Rh) + 3.94 \qquad (1)$$

$$\text{where} \qquad E(T_a, Rh) = \frac{Rh}{100} \times 6.105 \times e^{\frac{7.27 \times T_a}{237.7 + T_a}} \qquad (2)$$

Here, T_a and Rh are the air temperature and relative humidity, respectively, which are easily measured in daily life. As described in Section III-a, the temperature and relative humidity measurements might not be correct if the device is not outdoors or not open to the air, e.g., in a front trouser pocket. The calculated WBGT value is also affected by on-body positional dependency. A lower-than-actual level of risk suggested by an underestimate might render the user careless. Nevertheless, people would not mind if their devices were to be carried in an undesirable manner. In Section III-b, three types of device-position handling methods were introduced. As a first step, we took the *semi-automatic* approach in this application, in which the on-body position of the device is explicitly presented to the user to facilitate the interpretation of the reliability of the calculated WBGT by him- or herself.

c. Implementation

An uploading application on the terminal side obtains the latest temperature and relative humidity values from a local database (DB) and then calculates an approximate WBGT value. Global positioning information is obtained from a GPS receiver on a terminal. In addition to these basic data, the level of heatstroke risk (1 to 5) is determined based on the range of WBGT values [2]. This is often used in consumer heatstroke alert devices as a user-friendly indicator [7].

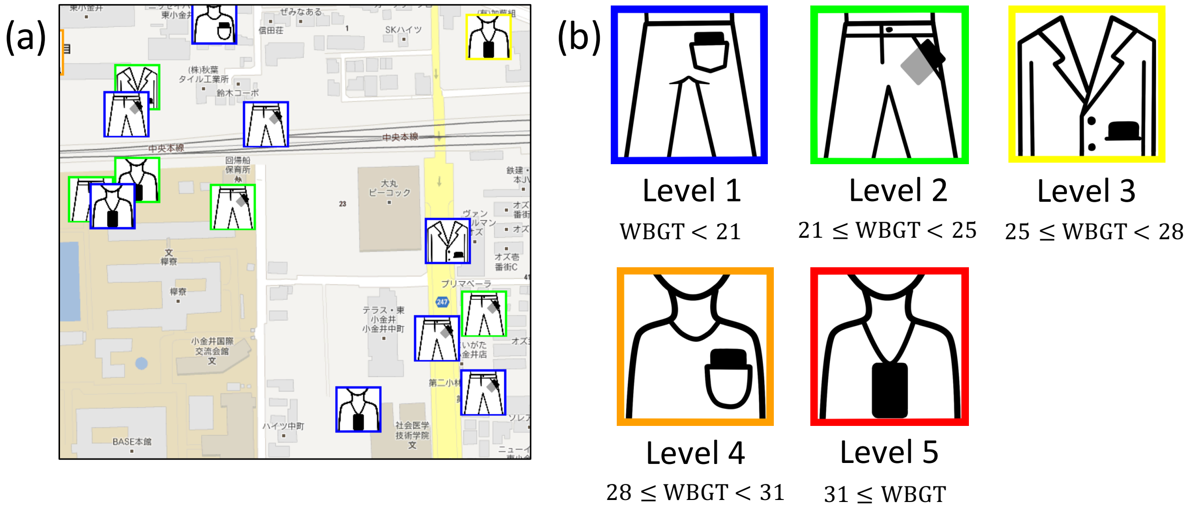


Figure 5. Visualization of collected data on a map: a) plots on the map and b) icons indicating the levels of risk and the storing positions.

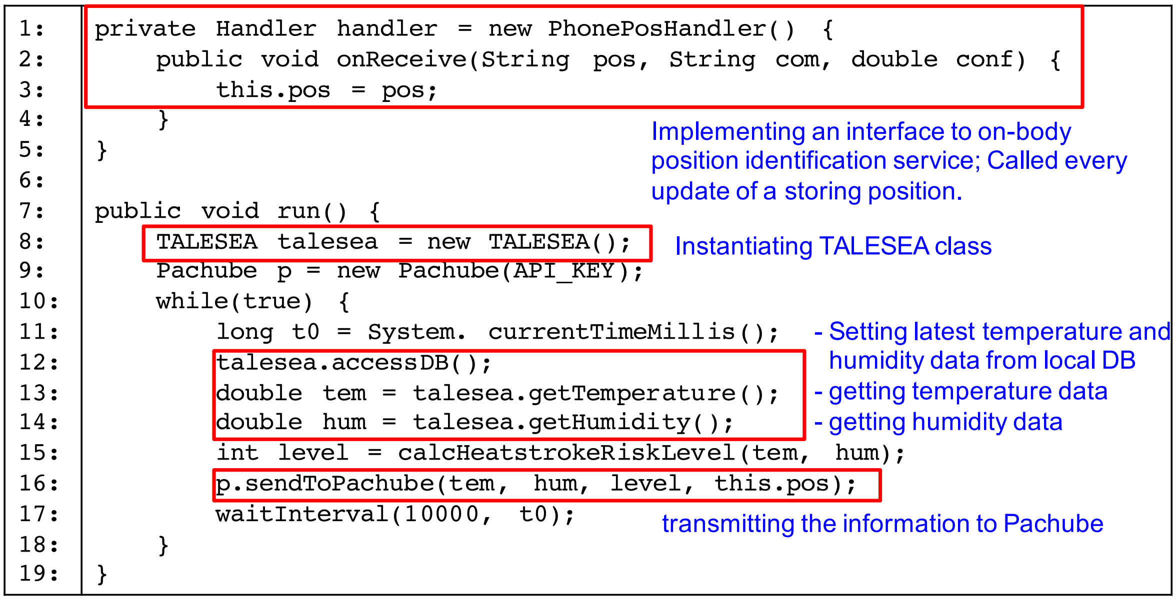
These types of information are shared and visualized on commercial web services. The data from terminals are shared on Pachube[2], an open scalable platform that utilizes the Internet of Things by connecting any devices and sharing data securely. An application component on the terminal, i.e., User Application in Figure 4, uses jPachube to post the data to Pachube. Here, jPachube is a Java wrapper for the Pachube API. The collected information is visualized using the Google Map API (Figure 5 (a)), in which an icon and its border indicate the storing position and the level of heatstroke risk (Figure 5 (b)), respectively. Further detailed information such as the exact values of temperature and relative humidity and the measurement time can be obtained by clicking the icon. Such visualization is an application realized on the server side (Application in Figure 2).

Figure 6 shows a code snippet of the uploading application. The information of the storing position is updated every time the sensor position recognition component (Figure 2) detects a new position and is set to an instance variable `pos` via the `onReceive` method. In the `run` method, the data are obtained from the DB (line 12–14 of Figure 6), and the risk level is

---

[2] Currently, Pachube has evolved into xively [37].

estimated (line 15). Then, the raw data, the risk level, and the storing position are sent to Pachube (line 16). As shown in lines 8, 12, 13, and 14, access to TALESEA is encapsulated by the `TALESEA` class, which allows an application developer to concentrate on handling the application data rather than handling USB communication or DB access through proprietary APIs.



```
1:  private Handler handler = new PhonePosHandler() {
2:      public void onReceive(String pos, String com, double conf) {
3:          this.pos = pos;
4:      }
5:  }
6:
7:  public void run() {
8:      TALESEA talesea = new TALESEA();
9:      Pachube p = new Pachube(API_KEY);
10:     while(true) {
11:         long t0 = System. currentTimeMillis();
12:         talesea.accessDB();
13:         double tem = talesea.getTemperature();
14:         double hum = talesea.getHumidity();
15:         int level = calcHeatstrokeRiskLevel(tem, hum);
16:         p.sendToPachube(tem, hum, level, this.pos);
17:         waitInterval(10000, t0);
18:     }
19: }
```

Implementing an interface to on-body position identification service; Called every update of a storing position.

Instantiating TALESEA class

- Setting latest temperature and humidity data from local DB
- getting temperature data
- getting humidity data

transmitting the information to Pachube

Figure 6. Code snippet of the application.

VI. CONCLUSION

In this article, we proposed a people-centric sensing system that takes into account the on-body position of a smartphone terminal (with sensors). A positional dependency of the environmental sensors is presented with an example of the temperature and relative humidity sensor readings. This implies that the quality of data collected by such a sensing paradigm might be low without taking account of the storing position of the sensor. A visualization system of the heatstroke risk level was implemented based on a basic framework of an on-body localization-aware human-probe system. A tiny sensor module for an Android-based smartphone, TALESEA, was investigated. Evaluation of the power consumption revealed the power profile of TALESEA-based people-centric sensing, in which the power supply to the TALESEA module was most depleted. High-level APIs were provided to developers that allow developers to use a smartphone's position as metadata for sensing data easily. The current version of the visualization system simply shows the storing position as self-interpretive information for the user. However, another type of information can be considered that provides a calibrated value as if the measurements were outdoors, with the possibility of over- or under-estimation of the measurement from a current position. We have already proposed this idea [38]. Absolute humidity is also calculated from temperature and relative humidity and can be used to estimate the risk of flu [30]. To build robust calibration formulas and extend the application area from the cold season (flu) to the hot season (heatstroke), large-scale data collection involving on-body positions, engaged activities, and clothing must be conducted.

ACKNOWLEDGMENT

The author thanks Mr. Satoshi Murata and Mr. Shigeki Hosokawa for their efforts in implementing TALESEA sensing module and heatstroke alert visualization application. This work was partially supported by the Fusion of Science and Technology (FOST).

REFERENCES

[1] K. Alanezi and S. Mishra, "Design, implementation and evaluation of a smartphone position discovery service for accurate context sensing", *Computers & Electrical Engineering* **2015**, *44*, pp. 307 – 323.

[2] M. Asayama, "Guideline for the Prevention of Heat Disorder in Japan", Global Environmental Research 13(1), (2009), pp. 19–25.

[3] J. R. Blum, D. G. Greencorn, and J. R. Cooperstock, "Smartphone sensor reliability for augmented reality applications", Proceedings of the 9th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, 2012.

[4] A. T. Champbell, N. D. Lane, E. Miluzzo, R. A. Peterson, H. Lu, X. Zheng, M. Musolesi, K. Fodor, S. B. Eisenman, and G.S. Ahn, "The Rise of People-Centric Sensing", IEEE Internet Computing., July/August Issue, (2008), pp. 12-21.

[5] D. Chen, K. T. Cho, S. Han, Z. Jin, and K. G. Shin, “Invisible Sensing of Vehicle Steering with Smartphones”, Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services; ACM: New York, NY, USA, 2015; MobiSys ’15, pp. 1–13.

[6] Y. Cui, J. Chipchase, F. Ichikawa, “A Cross Culture Study on Phone Carrying and Physical Personalization”, Proceedings of Human-Computer Interaction International 2007, pp. 483–492, 2007.

[7] Design Factory Inc. “Life Support Checker Series”, `http://www.necchu-sho.com/en/index.html`. (accessed: May 9, 2016)

[8] I. Diaconita, A. Reinhardt, D. Christin, and C. Rensing, “Inferring Smartphone Positions Based on Collecting the Environment’s Response to Vibration Motor Actuation”, Proceedings of the 11th ACM Symposium on QoS and Security for Wireless and Mobile Networks; ACM: New York, NY, USA, 2015; Q2SWinet ’15, pp. 99–106.

[9] Fire and Disaster Management Agency of Japan, “Emergency transport situation due to heatstroke in 2015”, `http://www.fdma.go.jp/neuter/topics/houdou/h27/10/271016_houdou_1.pdf.` (In Japanese. accessed: May 9, 2016)

[10] K. Fujinami, S. Kouchi, and Y. Xue, “Design and Implementation of an On-body Placement-Aware Smartphone”, Proceedings of the 32nd International Conference on Distributed Computing Systems Workshops (ICDCSW), pp. 69–74, June 2012.

[11] K. Fujinami, “On-Body Smartphone Localization with an Accelerometer”, Information 2016, Vol. 7, Article No. 21.

[12] J. Goldman, K. Shilton, J. Burke, D. Estrin, M. Hansen, N. Ramanathan, S. Reddy, V. Samanta, M. Srivastava, and R. West, “Participatory Sensing: A citizen- powered approach to illuminating the patterns that shape our world”, Foresight and Governance Project, White Paper (2009).

[13] K. Grimmer, E. King, T. Larsen, T. Farquharson, A. Potter, P. Sharpe, and H. Wit, “Prevalence of hot weather conditions related to sports participation guidelines: A South Australian investigation”, Journal of Science and Medicine in Sport 9(1-2), (2006), pp. 72–80.

[14] H. Höpfner and M. Schirmer, “Energy efficient continuous location determination for pedestrian information systems”, Proceedings of the 11th ACM International Workshop on Data Engineering for Wireless and Mobile Access, pp.58-65, 2012.

[15] O. D. Incel, "Analysis of Movement, Orientation and Rotation-Based Sensing for Phone Placement Recognition", *Sensors* **2015**, *15*, Article No. 25474.

[16] M. B. Kjærgaard, J. Langdal, T. Godsk, and T. Toftkjær. Entracked, "Energy-efficient robust position tracking for mobile devices", Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services, pp. 221–234, 2009.

[17] A. Krause, M. Ihmig, E. Rankin, D. Leong, S. Gupta, D. Siewiorek, A. Smailagic, M. Deisher, and U. Sengupta, "Trading off Prediction Accuracy and Power Consumption for Context-Aware Wearable Computing", Proceedings of the 2005 9th IEEE International Symposium on Wearable Computers, pp. 20-26, October 2005.

[18] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, A. T. Campbell, "A survey of mobile phone sensing", IEEE Communications Magazine, 48(9), (2010), pp. 140–150.

[19] N. Maisonneuve, M. Stevens, M. E. Niessen, and L. Steels, "NoiseTube: Measuring and mapping noise pollution with mobile phones", Proceedings of the 4th International ICSC Symposium on Information Technologies in Environmental Engineering, pp. 215-228, 2009.

[20] A. Mannini, A. M. Sabatini, and S. S. Intille, "Accelerometry-based recognition of the placement sites of a wearable sensor", *Pervasive and Mobile Computing* **2015**, *21*, 62–74.

[21] A. Manzoor, C. Patsakis, A. Morris, J. McCarthy, G. Mullarkey, H. Pham, S. Clarke, V. Cahill, and M. Bouroche, "CityWatch: exploring sensor data to manage cities better", Transactions on Emerging Telecommunications Technologies., vol. 64, (2014), pp. 64-80.

[22] E. Miluzzo, M. Papandrea, N. Lane, H. Lu, and A. Campbell, "Pocket, bag, hand, etc.-automatically detecting phone context through discovery", Proceedings of the 1st International Workshop on Sensing for App Phones, PhoneSense 2010 (2010).

[23] T. Miyaki and J. Rekimoto. Sensonomy: Envisioning folksonomic urban sensing. Adjunct Proceedings of Ubicomp 2008 Workshop, 2008.

[24] P. Mohan, V. N. Padmanabhan, R. Ramjee, "Nericell: Rich Monitoring of Road and Traffic Conditions Using Mobile Smartphones", Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems; ACM: New York, NY, USA, 2008; SenSys '08, pp. 323–336.

[25] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, and P. Boda. 2009. "PEIR, the personal environmental impact report, as a platform for participatory sensing systems research", Proceedings of the 7th international conference on Mobile systems, applications, and services, ACM, New York, NY, USA, 55-68.

[26] S. Reddy, K. Shilton, G. Denisov, C. Cenizal, D. Estrin, and M. Srivastava, "Biketastic: sensing and mapping for better biking", Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1817-1820, 2010.

[27] L. Ruge, B. Altakrouri, and A. Schrader, "SoundOfTheCity – Continuous Noise Monitoring for a Healthy City", Proceedings of the 2013 IEEE International Workshop on Smart Environments and Ambient Intelligence, pp. 670-675, 2013.

[28] Sensirion AG., "Digital humidity and temperature sensor SHT-71", `http://www.sensirion.com/en/products/humidity-temperature/humidity-sensor-sht71/`.

[29] F. Seraj, B. J. Zwaag, A. Dilo, T. Luarasi, and P. Havinga, "RoADS: A Road Pavement Monitoring System for Anomaly Detection Using Smart Phones", Proceedings of the 1st International Workshop on Machine Learning for Urban Sensor Data, 2014.

[30] M. Shoji, "Correlation between seasons and influenza epidemics", J. Natl. Inst. Public Health 48(4): pp. 282-290, 1999.

[31] SysCom Corp. `http://syscom-corp.jp`. (accessed: May 9, 2016)

[32] T. Sztyler and H. Stuckenschmidt, "On-body Localization of Wearable Devices: An Investigation of Position-Aware Activity Recognition", Proceedings of 2016 IEEE International Conference on Pervasive Computing and Communications, pp. 1-9, 2016.

[33] N. Thepvilojanapong, S. Konomi, and Y. Tobe, "Energy–Efficient Human Probes for High-Resolution Sensing in Urban Environments", In IEEJ Transaction on Electronic Engineering, Vol.6, (2011), pp.273–279.

[34] N. Thepvilojanapong, K. Sugo, Y. Namiki, and Y. Tobe, "Recognizing bicycling states with HMM based on accelerometer and magnetometer data", Proceedings of SICE 2011 Annual Conference, pp. 831–832, 2011.

[35] A. Vahdatpour, N. Amini, and M. Sarrafzadeh, "On-body device localization for health and medical monitoring applications", Proceedings of the 2011 IEEE International Conference on Pervasive Computing and Communications, pp. 37–44, 2011.

[36] C. Vaitl, K. Kunze, P. Lukowicz, "Does on-body location of a GPS receiver matter?", Proceedings of the 2010 International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2010), pp. 219–221, 2010.

[37] xively, `https://xively.com`. (accessed: May 9, 2016)

[38] Y. Xue, S. Hosokawa, S. Murata, S. Kouchi, and K. Fujinami, "An On-Body Placement-Aware Heatstroke Alert on a Smartphone", Proceedings of the 2012 International Conference on Digital Contents and Applications, pp. 226–234, 2012.

[39] C. P. Yaglou and D. Minard, "Control of heat casualties at military training centers", Arch. Ind. Hlth. 16(4), (1957), 302–305.

[40] Z. Zhuang, K. H. Kim, and J. P. Singh, "Improving Energy Efficiency of Location Sensing on Smartphones", Proceedings of Annual International Conference on Mobile Systems, Applications, and Services, pp. 315-330, June 2005.