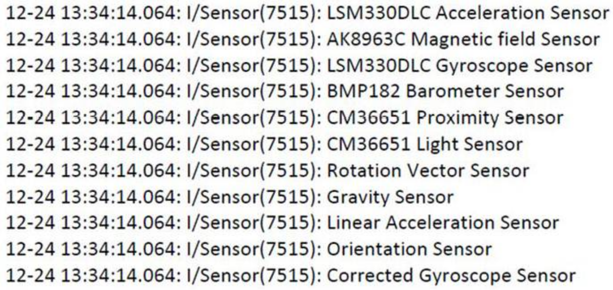
Towards a Silent Mobile Sensing Framework for Smart Cities

Farah Hariri, Guy Daher, Hussein Sibai, Karim Frenn, Sevag Doniguian, and Zaher Dawy
Electrical and Computer Engineering Department
American University of Beirut
Beirut, Lebanon
Email: {fsa20, gxd00, hms38, kff01, szd03, zd03}@aub.edu.lb

Abstract—**A key opportunity to be exploited in the coming decade is the utilization of the universal penetration of mobile connectivity to make life better for inhabitants of urban cities. There exist a wide range of possibilities to deploy smart mobile solutions for vertical sectors and markets that span health, transportation, environment, safety, entertainment, buildings, education, business, agriculture, and industry. In this paper, we consider the development of a mobile sensing framework that utilizes the embedded sensors in smartphones to seamlessly capture context information in urban environments. We present an overview of a design architecture that facilitates silent mobile sensing in addition to centralized context data visualization and analytics. Moreover, we discuss potential application scenarios that can use the presented framework to develop mobile solutions for smart cities. Finally, we present preliminary results for a case study that aims at creating public WiFi coverage maps for the campus of the American University of Beirut.**

I. INTRODUCTION

High-end smartphones are equipped with multiple embedded sensors and, thus, can capture in real time context information that can be utilized to develop innovative mobile solutions for smart cities [1]. A up-to-date smartphone includes more than 10 embedded sensors in addition to audio/video recording and GPS localization capabilities (e.g., see Figure 1).



```
12-24 13:34:14.064: I/Sensor(7515): LSM330DLC Acceleration Sensor
12-24 13:34:14.064: I/Sensor(7515): AK8963C Magnetic field Sensor
12-24 13:34:14.064: I/Sensor(7515): LSM330DLC Gyroscope Sensor
12-24 13:34:14.064: I/Sensor(7515): BMP182 Barometer Sensor
12-24 13:34:14.064: I/Sensor(7515): CM36651 Proximity Sensor
12-24 13:34:14.064: I/Sensor(7515): CM36651 Light Sensor
12-24 13:34:14.064: I/Sensor(7515): Rotation Vector Sensor
12-24 13:34:14.064: I/Sensor(7515): Gravity Sensor
12-24 13:34:14.064: I/Sensor(7515): Linear Acceleration Sensor
12-24 13:34:14.064: I/Sensor(7515): Orientation Sensor
12-24 13:34:14.064: I/Sensor(7515): Corrected Gyroscope Sensor
```

Fig. 1. Sensors available in a Galaxy Note II smartphone accessed using an Android API.

However, smartphones have not been fully exploited yet taking into account their powerful computing, storage, and sensing capabilities [2]. Crowd-sensing is a growing field that benefits from the pervasive dominance of smartphones in everyday life in order to collect large-scale sensor data [3], [4]. Crowd-sensing applications can in general be categorized into two types. The first type is personal sensing, in which the aim is to capture information mainly related to the smartphone's user, e.g., user's activity mode. The second type is community sensing, in which sensor data is collected from many smartphones in order to monitor phenomena occurring in the environment around the users, e.g., traffic congestion level. Community sensing can be further divided into participatory sensing in which the user gets involved in the sensing process, and opportunistic sensing which takes places seamlessly without any user intervention [3].

In this paper, we present the design architecture, prototype implementation, and application scenarios for a silent mobile sensing framework that is being developed at the American University of Beirut (AUB). The long-term objective is to have a scalable and flexible framework that can be reused for different crowd-sensing applications and that facilitates dynamic spatiotemporal visualization over geographical maps and the development of mobile solutions using data as a service. Moreover, we present an overview with preliminary sample results for utilizing the developed framework to create public WiFi coverage maps for AUB's campus. The coverage maps will include both WiFi signal strength level in addition to WiFi connection quality level.

Many mobile sensing projects focused on individual sensing applications [5]. For example, the SenSay application [6] aims at recognizing user context to improve the phone's usability. SoundSence [7] is an application that makes use of the microphone to detect user-based sound events. Other examples include an e-coaching message-based system for exercise and physical condition monitoring [8], an emergency situations recognizer [9], and a context recognizer for social networking purposes [10]. Recently, there is a shift towards community sensing applications where the emphasis is on phenomena affecting the environment or the society rather than the individual users [5]. VTrack [11] is an example of a transportation and traffic related application, CenceMe [12] is an example of a social networking oriented mobile application, and UbiFit Garden [13] is an example of a health oriented mobile application.

In Section II, we present the design and prototype implementation of the silent mobile sensing framework that ie being

---

This work was made possible by a research grant from Ericsson Region Middle East. The statements made herein are solely the responsibility of the authors.

developed at AUB. In Section III, we discuss sample preliminary results with focus on creating public WiFi coverage maps. Finally, challenges are highlighted and conclusions are drawn in Section IV.

II. DESIGN ARCHITECTURE AND PROTOTYPE IMPLEMENTATION

In this section, we present the design architecture and prototype implementation for a silent mobile sensing framework that is being developed at AUB. The aim is to be able to capture seamlessly a wide range of spatiotemporal contextual information in urban cities using the smartphones of mobile users. The captured sensor data provides wealth of opportunities to better understand the city dynamics and to implement customized mobile solutions towards the vision of smart cities. The design is divided into four main modules: silent mobile sensing module, cloud server and storage module, map visualization module, and mobile solutions development module. Figure 2 presents a general description of the presented framework.

A. Silent Mobile Sensing Module

The first module in the framework is the silent sensing mobile application that is capable of acquiring seamlessly context data from sensors embedded in smartphones with a pre-defined configuration including sensor and sensing rate selection. The mobile application should be silent without any interference with the user's normal operation of the device, and should be lightweight to consume limited energy and storage resources. The captured raw data is then stored in a local database with processing over several intelligent steps that include: data filtering to exclude in-valid or in-applicable measurements, data compression to reduce the local storage requirements, and efficient data uploading using the available wireless interfaces. For the efficient uploading, an experimental study is conducted to determine the adequate upload rate and size of batches, i.e., to determine how frequently to upload the stored sensor data. This depends on the number and type of sensor measurements in addition to the wireless connectivity availability and battery capacity status. The captured sensor data is purged from the smartphone after it has been uploaded to a given remote server.

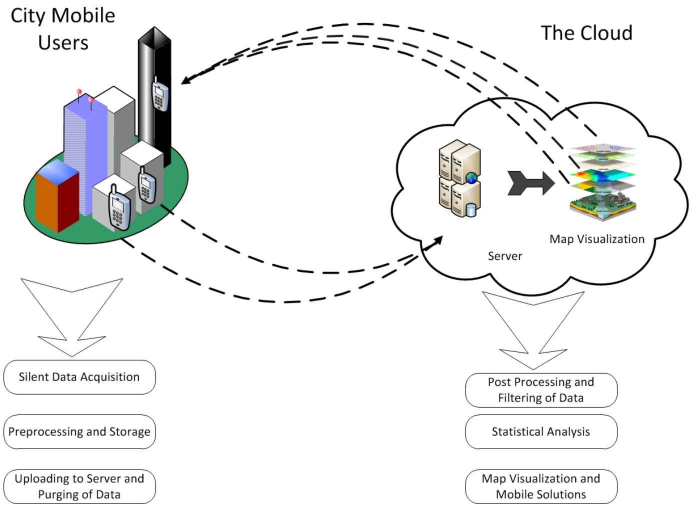
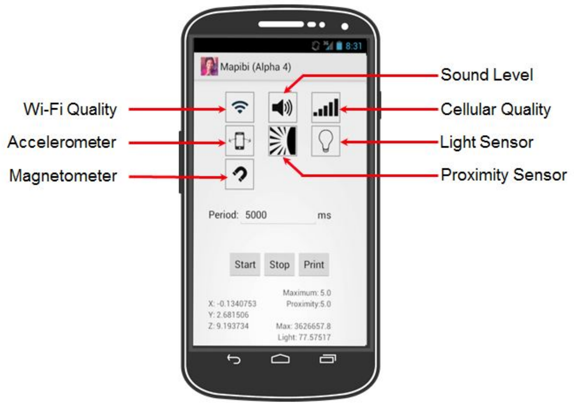


Fig. 2. Silent mobile sensing framework general description.

In terms of implementation, an Android mobile application has been developed named "Mapibi". The application runs silently in the background and records sensor data without the need of any user intervention. The sensed raw data is stored in a local SQLite database with the time and GPS location attributes. The graphical user interface (GUI) shows only the available sensors in the device and offers the user the possibility of activating and deactivating sensors for the measurements. The current version of the prototype GUI design is shown in Figure 3. After pressing the start button, the Mapibi application initiates silent sensing and storing in the local database. The sensing rate is pre-configured by the user depending on the required timing accuracy of the sensed data. Then, unreliable data is filtered out and only valid measurements are uploaded to a remote server for further processing.



Period: 5000 ms

X: -0.1340753
Y: 2.681506
Z: 9.193734

Maximum: 5.0
Proximity:5.0

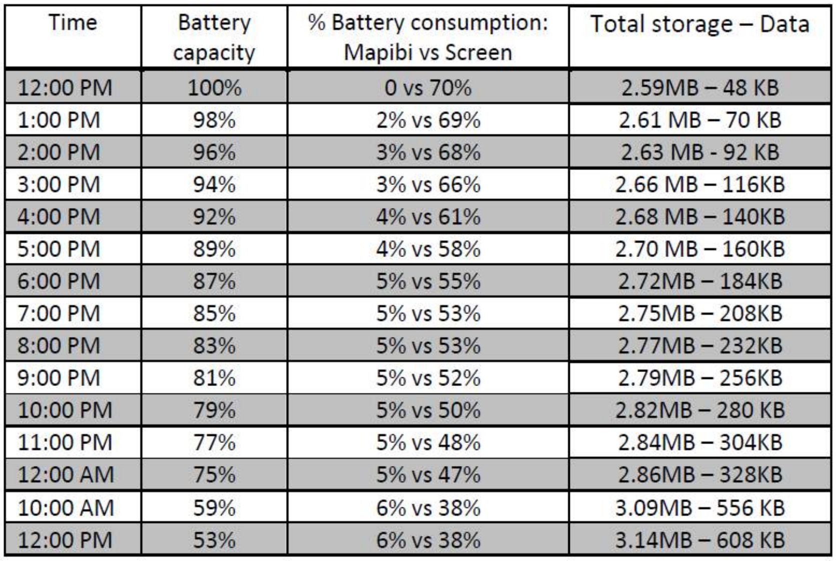
Max: 3626657.8
Light: 77.57517

Fig. 3. Silent sensing mobile application prototype GUI.

B. Cloud Server and Storage Module

The silent sensing application will be installed on a relatively large number of smartphones. The captured sensor data from each smartphones running the application will be uploaded on a continuous basis to a remote server for storage and post-processing. The collected data at the server will be utilized for two main objectives. The first objective aims at the spatiotemporal visualization of city context data on geographical maps (see Section II.C). The second objective includes data analytics to offer customized mobile solutions to users focusing on challenges in cities (see Section II.D).

In terms of implementation, an Apache HTTP server and a MySQL database have been used with an automated PHP based interconnection between them. In order to provide basic privacy, no data about the identity of the smartphones or the users is collected. An arbitrary anonymous ID is assigned by the server for each upload session. Upon request, data can be



Time	Battery capacity	% Battery consumption: Mapibi vs Screen	Total storage – Data
12:00 PM	100%	0 vs 70%	2.59MB – 48 KB
1:00 PM	98%	2% vs 69%	2.61 MB – 70 KB
2:00 PM	96%	3% vs 68%	2.63 MB - 92 KB
3:00 PM	94%	3% vs 66%	2.66 MB – 116KB
4:00 PM	92%	4% vs 61%	2.68 MB – 140KB
5:00 PM	89%	4% vs 58%	2.70 MB – 160KB
6:00 PM	87%	5% vs 55%	2.72MB – 184KB
7:00 PM	85%	5% vs 53%	2.75MB – 208KB
8:00 PM	83%	5% vs 53%	2.77MB – 232KB
9:00 PM	81%	5% vs 52%	2.79MB – 256KB
10:00 PM	79%	5% vs 50%	2.82MB – 280 KB
11:00 PM	77%	5% vs 48%	2.84MB – 304KB
12:00 AM	75%	5% vs 47%	2.86MB – 328KB
10:00 AM	59%	6% vs 38%	3.09MB – 556 KB
12:00 PM	53%	6% vs 38%	3.14MB – 608 KB

Fig. 4. Sample results for energy and storage requirements of the Mapibi silent mobile sensing application assuming WiFi signal level sensing, time stamp recording, 5 seconds sampling interval, and duration of one day.

retrieved from the MySQL database and automatically converted to JSON format for processing in the map visualization module.

C. Spatiotemporal Visualization Module

This module aims at the visual representation of the data that has been stored and post-processed at the remote server. The output will include multiple layers of spatiotemporal data overlaid over geographical maps that capture user-centric contexts such as, e.g., people mobility patterns, traffic congestion, user density distribution, wireless network coverage maps, noise pollution levels, etc. This module is important to capture dynamically and in real-time context variation over both the spatial and temporal dimensions and, thus, it helps better understand aspects related to city dynamics. In terms of prototype implementation, the Google Maps JavaScript API was used with a customized Heatmap Layer and a time line for interactive scrolling over the temporal dimension (see Figure 6).

D. Mobile Solutions Development Module

The loop between users and the server is closed with the development of customized mobile solutions that provide the captured sensor data as a service. This leads to a wide range of potential application scenarios that can be built based on the presented silent mobile sensing framework. The following are some selected examples: i. Utilizing the accelerometer sensor with GPS localization to design intelligent mobile solutions in order to address traffic congestion in cities; ii. Capturing cellular/WiFi signal strength levels with GPS localization to create public spatiotemporal coverage maps in order to guide city inhabitants to good connectivity areas or to help operators identify coverage holes (see Section III.B for more details); iii. Monitoring people's density and mobility patterns in selected parts of the cities over time, e.g., to optimize bus routing and schedules within a large enterprise or campus; iv. Applying signal processing techniques to audio signals recorded silently via the smartphone's microphone to create noise pollution maps in order to help locating quiet places for relaxation or crowded places for socialization inside the city.

III. SAMPLE PROTOTYPE TESTING RESULTS

In this section, we present sample prototype testing results focusing on the storage/energy requirements of the Mapibi silent sensing mobile application and on the utilization of the developed framework to create public WiFi coverage maps.

A. Silent Sensing Mobile Application: Storage and Energy Requirements

Continuously collecting data from smartphone sensors results in some challenges. For example, recording and processing sensor data at a high sampling rate lead to an increase in the energy consumption and, thus, lead to significant battery drainage. Moreover, the low accuracy of sensors, at times, leads to difficulties in getting reliable data. In order to better understand the requirements of continuous sensing mobile applications, we conducted an experimental study using a Galaxy Note II smartphone running the Mapibi mobile application with different configurations. Figure 4 presents a summary of the obtained results for the following scenario: WiFi signal strength sensing and recording, time stamp recording, sensing interval every 5 seconds, and no other applications are running. The Android OS reported that the Mapibi application consumed only 2.8% (6% of the 47% consumed) of the battery's total energy capacity over a duration of 24 hours with a data storage of 608 KB (around 24 KB stored per hour). However, when the sampling rate was reduced to 1 second with additional storage of GPS location data, the energy consumption of the Mapibi application increased to 33.11% of the battery's total energy capacity over a duration of 12 hours with a data storage of 2,32 MB (around 200 KB stored per hour). This demonstrates the impact of the sensing rate and the number/type of sensors on the energy and storage requirements of continuous silent mobile sensing applications; thus, this motivates the development of lightweight silent mobile sensing applications, efficient storage techniques with advanced compression algorithms, and intelligent approaches to avoid in-accurate or in-valid sensing activity or sensor measurements.

B. Towards Public WiFi Coverage Maps

The presented silent mobile sensing framework is being tested for a mobile solution that aims at creating public WiFi coverage maps. The aim is to perform seamless collection of WiFi coverage data from smartphones while connected to AUB's wireless network, apply post-processing and visualization at the remote server side, and share the outcome dynamically over a spatiotemporal map that can be publicly accessed via an interactive web interface.

For the WiFi coverage maps mobile solution, the information that will be captured in the smartphone includes: time of the recording, GPS location, WiFi signal strength level, and WiFi connection quality level. Before uploading the data, certain conditions will be checked to make sure that the data is valid, e.g., data with invalid GPS location or invalid WiFi signal level will be discarded. It is important to highlight that capturing the WiFi signal strength level is straightforward using existing Android APIs; however, capturing the WiFi connection quality level is more challenging as it depends on the signal strength in addition to the access network load and the end-to-end Internet connection congestion.

Figure 5 presents sample experimental measurement results that we have conducted in [14] to quantify the impact of WiFi signal strength level and access network load on effective download bit rate (which maps to connection quality level) and energy consumed from the smartphone's battery. Results show that as the signal strength increases on the right vertical axes from -70 dBm to -40 dBm, the effective download bit rate for WiFi increases from 400 Kbps to 1.8 Mbps on the x-axis, which leads to a reduction on the energy consumption. Similarly, as the WiFi access network load increases from low to medium to high, the effective download bit rate decreases notably even though the signal strength level was high during the measurements.

In order to capture the WiFi connection quality level, a "speed test" based approach is implemented that downloads data of a given size from a server and records the download time in order to empirically evaluate the effective download bit rate. Some challenges that need to be addressed include determining the size of the data to be downloaded for accurate bit rate estimation, setting appropriate triggers to initiate the bit rate estimation function only when needed, finding an approach to perform WiFi-related sensing while the device is in sleep mode, and differentiating between private and public WiFi connectivity.

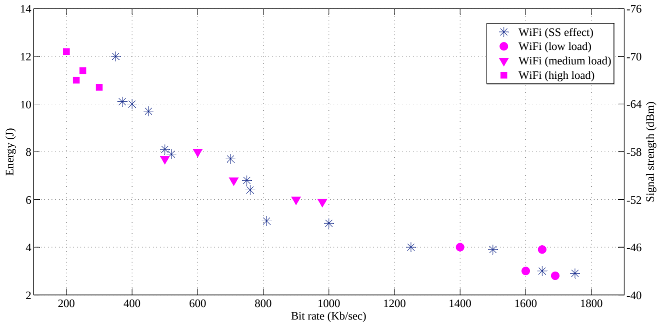


Fig. 5. Effective download bit rate and consumed energy as a function of both WiFi access network load and WiFi signal strength level. The WiFi measurement points represented by * correspond to the right vertical axes of signal strength levels.

Preliminary testing of the WiFi coverage maps mobile solution has been initiated at AUB using the smartphones of five mobile users. The Mapibi application was downloaded to their devices and data was recorded over part of AUB's campus. A sample spatiotemporal visualization with a heat map layer for the WiFi signal strength data collected in the Faculty of Engineering and Architecture area is shown in Figure 6 for demonstration purposes. The green color represents high WiFi signal strength, whereas the yellow color represents low WiFi signal strength.

IV. CONCLUSION

We presented the design architecture and prototype implementation details results for a silent mobile sensing framework that is being developed at the American University of Beirut. Moreover, we discussed example scenarios for using the framework to develop mobile solutions for smart cities with preliminary sample results based on a case study to create public WiFi coverage maps. The long-term goal is to utilize wireless connectivity and mobile network penetration to create context maps using data extracted seamlessly from smartphones via silent mobile applications. The extracted data and the created context maps can then be utilized to identify problems and opportunities, capture infrastructure capabilities, and trigger customized solution development in order to improve the quality of life of inhabitants in cities.

Towards achieving this long-term goal, there are research challenges that still need to be addressed including lightweight silent sensing mobile applications with relatively low energy and storage requirements, user incentives, user privacy, device platform inter-operability, advanced sensing functionalities, etc. For example, concerning energy consumption, the following factors play an important role: the upload data batch size, the frequency of uploading, the amount of pre-processing at the device level, the number and type of sensors that need to be captured, and the frequency of sensing.

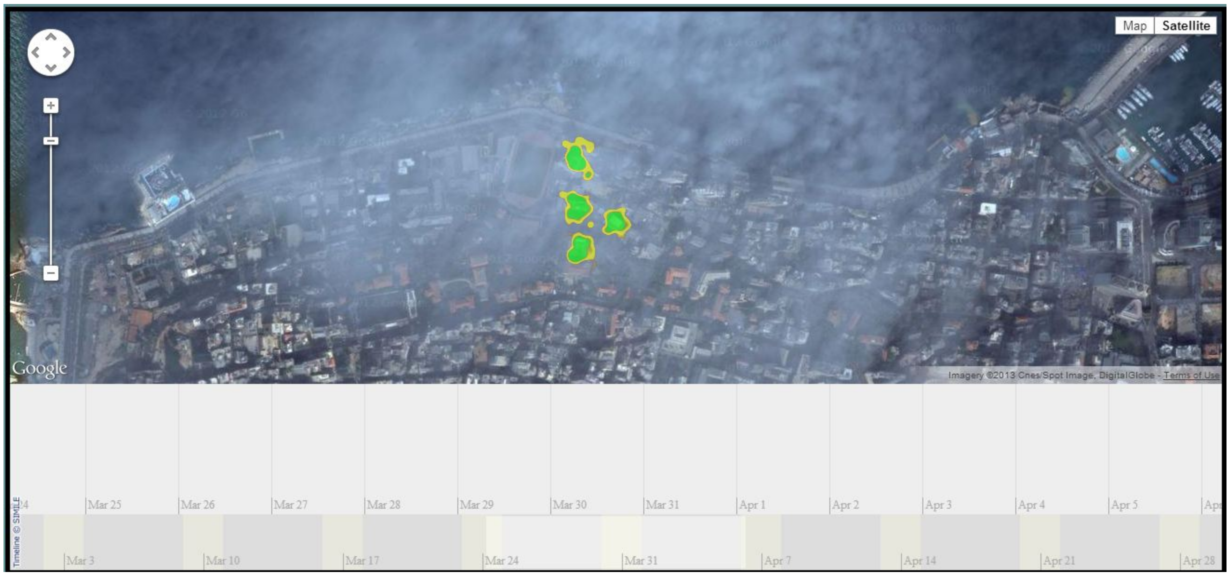


Fig. 6. Preliminary spatiotemporal map visualization for WiFi signal strength levels collected using the Mapibi silent mobile sensing application on part of AUB's campus.

REFERENCES

[1] Ericsson ConsumerLab Insight Summary Report, "City Life," May 2012.

[2] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, "A survey of mobile phone sensing," *IEEE Communications Magazine*, vol. 48, no. 9, pp. 140–150, September 2010.

[3] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: Current state and future challenges," *IEEE Communications Magazine*, vol. 49, no. 11, pp. 32–39, November 2011.

[4] M.-R. Ra, B. Liu, T. L. Porta, and R. Govindan, "Medusa: A programming framework for crowd-sensing applications," in *ACM MobiSys 2012*, June 2012.

[5] Y. Xiao, P. Simoens, P. Pillai, K. Ha, and M. Satyanarayanan, "Lowering the barriers to large-scale mobile crowdsensing," in *Fourteenth Workshop on Mobile Computing Systems and Applications (ACM HotMobile 2013)*, February 2013.

[6] D. Siewiorek, A. Smailagic, J. Furukawa, A. Krause, N. Moraveji, K. Reiger, J. Shaffer, and F. L. Wong, "Sensay: A context-aware mobile phone," in *7th IEEE International Symposium on Wearable Computers (ISWC 2003)*, October 2003.

[7] H. Lu, W. Pan, N. D. Lane, T. Choudhury, and A. T. Campbell, "Soundsense: Scalable sound sensing for people-centric applications on mobile phones," in *MobiSys 2009*, June 2009.

[8] Y. Kawahara, H. Kurasawa, and H. Morikawa, "Recognizing user context using mobile handsets with acceleration sensors," in *IEEE International Conference on Portable Information Devices (PORTABLE'07)*, May 2007.

[9] J. Himberg, K. Korpiaho, H. Mannila, J. Tikanmaki, and H. T. Toivonen, "Time series segmentation for context recognition in mobile devices," in *IEEE International Conference on Data Mining (ICDM 2001)*, December 2001.

[10] A. C. Santos, J. M. Cardoso, D. R. Ferreira, P. C. Diniz, and P. Chaínho, "Providing user context for mobile and social networking applications," *Pervasive and Mobile Computing*, vol. 6, no. 3, pp. 324–341, January 2010.

[11] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson, "Vtrack: Accurate, energy-aware road traffic delay estimation using mobile phones," in *7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009)*, November 2009.

[12] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng, and A. T. Campbell, "Sensing meets mobile social networks: The design, implementation and evaluation of the CenceMe application," in *6th ACM conference on Embedded Network Sensor Systems (SenSys 2008)*, November 2008.

[13] S. Consolvo, D. W. McDonald, T. Toscos, M. Y. Chen, J. Froehlich, B. Harrison, P. Klasnja, A. LaMarca, L. LeGrand, R. Libby, *et al.*, "Activity sensing in the wild: A field trial of UbiFit garden," in *Annual SIGCHI Conference on Human Factors in Computing Systems*, April 2008.

[14] S. Taleb, M. Dia, J. Farhat, Z. Dawy, and H. Hajj, "On the design of energy-aware 3g/wifi heterogeneous networks under realistic conditions," in *27th IEEE International Conference on Advanced Information Networking and Applications (AINA-2013) - Ninth International Workshop on Heterogeneous Wireless Networks (HWISE-2013)*, March 2013.