Combination and Abstraction of Sensors for Mobile Context-Awareness

Borja Gamecho
Egokituz: Laboratory of HCI for Special Needs
University of the Basque Country (UPV/EHU)
Paseo Manuel Lardizabal 1, Donostia, PA 20018 Spain
borja.gamecho@ehu.es

Luis Gardeazabal
Egokituz: Laboratory of HCI for Special Needs
University of the Basque Country (UPV/EHU)
Paseo Manuel Lardizabal 1, Donostia, PA 20018 Spain
luis.gardeazabal@ehu.es

Julio Abascal
Egokituz: Laboratory of HCI for Special Needs
University of the Basque Country (UPV/EHU)
Paseo Manuel Lardizabal 1, Donostia, PA 20018 Spain
julio.abascal@ehu.es

UbiComp'13 Adjunct, September 8–12, 2013, Zurich, Switzerland.

ACM 978-1-4503-2215-7/13/09...$15.00.
http://dx.doi.org/10.1145/2494091.2496037

Abstract
In this paper, we describe a context server application for mobile computing. Its main objective is to assist developers to exploit context-aware features in their applications. This approach uses the extraction of new context information using a combination of sensors and proposes a sensing abstraction layer to avoid having to deal with specific hardware.

Author Keywords
Ubiquitous Computing, Sensor Combination, Sensor Abstraction

ACM Classification Keywords
I.2.9 [Robotics]: Sensors; H.5.2 [User Interfaces]: Input devices and strategies

Introduction
The proliferation of intelligent mobile devices in recent years has been accompanied by the incorporation of a larger number of increasingly sophisticated and complex sensors in these devices. Environment perception based on the data collected by advanced sensors has enabled the creation of mobile applications that are much more advanced than the previous ones. These applications are able to perceive and react to the environment, allowing context-awareness. Currently, there is great interest in

applications that can collect different aspects of everyday activities, opening the way for personal metrics[1] and self quantification[2]. The success of these applications suggests that people are becoming more receptive to the idea of using and wearing sensing devices on a daily basis. See for instance, the list of commercial and do-it-yourself gadgets that incorporate sensors presented by Swan [11]. Most of these devices are designed to be used with smartphones through personal area networks, such as Bluetooth. This situation portends a new generation of mobile applications that are able to react to the context in a more accurate way.

Combination of Sensors

One of the advantages of having multiple sensors with different functionalities is the possibility of combining several sources of data to obtain higher-level context information. This possibility was mentioned a long time ago by Schmidt et al. [10]. This work describes the fusion of eight sensors (temperature, pressure, and CO gas meters, a photodiode, two accelerometers, a passive IR, and a microphone) to recognize different means of transport. Additionally, it detects whether the mobile phone is in the hand, in the suitcase or in the table, among other contexts. Subsequently, multiple works presented progress in this direction. For instance, LifeMap [4] is an application that is able to recognize the context of use of a smartphone by means of regular sensors (GPS, accelerometers, compass, etc.). It can detect whether the user is walking, running, etc. in Android devices having these sensors. Similarly, "Phoneprioception" [12] studies the ability of smartphones to identify where they are (bag/pocket/hand/etc.). To do this, it combines four sensors: accelerometer, light/proximity, capacitive and multispectral. The last two sensors are prototypes incorporated specifically for this purpose.

In addition to embedded sensors, external sensors can be used to obtain supplementary context information. For instance, background information about the user can be obtained by combining information from wearable sensors [9]. Through these it is possible to perceive physiological signals (or biosignals), which allow the extraction of information on actions performed by the person wearing them, as shown in the survey published by Avci et al. [2]. Additionally, user's emotions can also detected enabling the creation of new "categories of context". For instance, the work by Haag et al. [6] succeeds in identifying the emotional state of a user in terms of valence and arousal, by combining a set of biosignals: Electromyography (EMG), Electrodermal activity (EDA), skin temperature (ST), blood volume pulse (BVP), electrocardiography (ECG) and respiration. Likewise, the work of Jang et al. [7] classifies three negative emotions (fear, surprise and stress) from four biosignals: ST, ECG, EDA and photoplethysmography (PPG). Works on biosignal acquisition combined with mobile platforms, such as Bitalino [1], appear very promising in terms of revealing emotional contexts in users of mobile devices.

Finally, external sensors deployed in the environment are able to provide information characterizing the physical space where an interaction between the user and the application occurs. Therefore, the combination of data collected by different sensors is extremely useful when developing context-aware applications.

[1] http://www.personalmetrics.us/
[2] http://quantifiedself.com/

Challenges for Developers

In behalf of such an abundance of sensor devices, developers of mobile applications face several challenges to create context-aware applications, including:

- **Sensor heterogeneity**. Due to the variety of devices and manufacturers, each sensor requires different low-level management in order to obtain the information provided.
- **Platform heterogeneity**. Different mobile development platforms use incompatible Software Development Kits (SDK) making it impossible to share the same code for the same application running in different target smartphones, even if they have the same set of sensors.
- **Pattern recognition**. Developers devote considerable effort to recognizing activities happening in the mobile phone context. For each activity the available data has to be analysed and models to match these activities have to be designed and trained.
- **Performance**. Mobile devices usually make a trade-off between the battery consumption rate and the processor activity. The proliferation of independent sensor readings and the heavy processing algorithms run by different applications critically affect the life of the battery.

To overcome some of these problems, we propose a context server application framework. Its main objective is to facilitate the development of context-aware applications independently of the specific sensors currently available in the user environment. In the next sections, we introduce the basics for this framework.

Sensor Abstraction Layer

To cope with the problems created by sensor heterogeneity it is necessary to abstract the programming interface. There are several families of mobile platforms (IOs, Android, Windows Phone, etc.) each having its own SDKs and APIs, mostly incompatible with each other. Frameworks such as PhoneGap[3], deal with this issue by using a single common programming interface for "shared components" (accelerometers, camera, network, notifications, contact list, etc.) among different smartphones. This approach helps developers to design single applications that can be compiled for different platforms.

Furthermore, external devices connected to smartphones include their own libraries to be programmed. Consequently, developers have to deal with multiple libraries if they want to use more than one device simultaneously. To overcome this problem we propose using the URC Standard[4]. URC, and its middleware UCH [13], allow the abstraction of heterogeneous devices in ubiquitous computing environments. We use this approach to abstract sensors and to provide a middleware to interact with external devices. This supports the many sensor devices that can be paired with the mobile device via Bluetooth.

Given this background, we extended the idea of abstract programmable interfaces focusing on sensors; for instance, PhoneGap only provides interfaces for a few sensors, but we propose a common Application Program Interface (API) that allows for a wider range of sensors.

[3] http://phonegap.com/
[4] http://www.openurc.org/

As it can be seen in the survey by Baldauf et al. [3], several context-aware systems have been presented that assist end-users and developers in dealing with context issues. Among them, Korpipää et al. [8] created an ontology for managing context information in mobile devices and a framework for context-aware application development. This framework for Nokia series 60 smartphones used a blackboard model. Considering the new mobile platforms and sensors that have been released in the last decade, it would be interesting to update this valuable approach. In this line, we created a taxonomy of physical sensors that feeds an ontology with the characteristics of available sensors (including device-embedded sensors, wearable sensors and sensors deployed in the environment). We clustered these based on different criteria and defined the available context information by means of rules.

As previously mentioned, our objective is to create an context server application to assist developers by including context information directly in their applications [5]. Therewith, developers do not have to deal directly with sensor and platform heterogeneity. In addition, our tool will free them from the task of discovering context patterns.

So far, our solution leaves unsolved the previously mentioned performance challenge. In fact, other frameworks, such as PhoneGap or UCH, actually increase the communication time due to the use of hidden middleware layers. Other functionalities, such as maintaining a stable Bluetooth connection and managing the ontology, can affect battery performance.

An overall view of our system can be found in Figure 1.

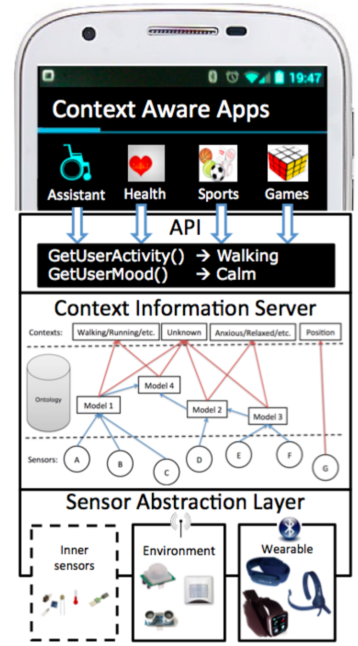

Figure 1: Description of the proposed system

Context Modeling

It is necessary to match human activities with the related data provided by the combination of sensors. In this way, the system will be able to recognize specific activities.

To this end, the context server learns activity patterns by means of the following procedure: Firstly, a set of relevant available sensors is defined to be logged for a specific activity. The logs include the time-stamp and all the data collected. In addition, the evolution of the activity can be annotated by the user. Classification techniques are used to train a model to recognize the annotated patterns. To finish, the resulting model is coded into the context server application and is added to the ontology as suitable for a specific context.

In order to identify a large number of patterns several models are required.

Validation

We are currently working on identifying activity patterns using a series of low-cost devices with sensors. For the validation of our approach, we use a smartphone and three wearable devices. The smartphone includes a set of sensors such as accelerometer, barometer, magnetometer, compass, light, etc. Regarding the wearable devices, we use a smart watch with accelerometer, a small head-worn device that provides some simple brain electroencephalography (EEG) signals, and a chest-strap to measure heart rate using electrocardiography (ECG) signals. All of them are connected to the mobile phone by Bluetooth. They are programmed using the public APIs and SDKs made available for them to develop personalized applications.

We plan a validation consisting of two phases:

- Firstly, we want to prove that our approach identifies different contexts arising from the combination of different sensor signals. In this phase, we will evaluate the effectiveness of the framework provided to developers.
- Secondly, we will test the abstraction of sensors, using other devices from different manufacturers with the same types of sensors.

Throughout the validation process we will introduce more sensors using the Android Open Accessory[5] (AOA) and more environmental sensors to reveal new contexts.

Conclusion

In this paper we presented a framework to assist the creation of context-aware applications. The principal contributions are: *sensor abstraction*, to avoid the need of dealing with specific sensors devices; and *complex context information provision* (to context-aware applications), avoiding the need of performing context recognition by each application.

Acknowledgements

EGOKITUZ is funded by the Department of Education, Universities and Research of the Basque Government (grant IT395-10) and belongs to the Basque Advanced Informatics Laboratory (BAILab) supported by the University of the Basque Country UPV/EHU (grant UFI11/45). Borja Gamecho holds a PhD scholarship from the Research Staff Training Programme of the Basque Government.

References

[1] Alves, A., Silva, H., Lourenço, A., and Fred, A. Bittalino: A biosignal acquisition system based on the arduino. In *Proc. BIODEVICES* (2013), 261–264.

[2] Avci, A., Bosch, S., Marin-Perianu, M., Marin-Perianu, R., and Havinga, P. J. M. Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: A survey. In *ARCS Workshops*, VDE Verlag (2010), 167–176.

[3] Baldauf, M., Dustdar, S., and Rosenberg, F. A survey on context-aware systems. *Int. J. Ad Hoc and*

[5] http://developer.android.com/tools/adk/index.html

Ubiquitous Computing 2, 4 (2007), 263–277.

[4] Chon, J., and Cha, H. Lifemap: A smartphone-based context provider for location-based services. *IEEE Pervasive Computing 10*, 2 (2011), 58–67.

[5] Gamecho, B., Gardeazabal, L., and Abascal, J. Augmented interaction with mobile devices to enhance the accessibility of ubiquitous services. In *Proc. MOBACC Workshop* (2013).

[6] Haag, A., Goronzy, S., Schaich, P., and Williams, J. Emotion recognition using bio-sensors: First steps towards an automatic system. In *Proc. ADS*, Springer (2004), 36–48.

[7] Jang, E.-H., Park, B.-J., Kim, S.-H., Chung, M., Park, M.-S., and Sohn, J.-H. Classification of three negative emotions based on physiological signals. In *Proc. INTELLI 2013* (2013), 75–78.

[8] Korpipää, P., Mäntyjärvi, J., Kela, J., Keränen, H., and Malm, E.-J. Managing context information in mobile devices. *IEEE Pervasive Computing 2*, 3 (2003), 42–51.

[9] Morris, D., Saponas, T. S., and Tan, D. S. Emerging input technologies for always-available mobile interaction. *Foundations and Trends in HCI 4*, 4 (2011), 245–316.

[10] Schmidt, A., Aidoo, K. A., Takaluoma, A., Tuomela, U., Laerhoven, K. V., and de Velde, W. V. Advanced interaction in context. In *Proc. HUC 1999*, Springer (1999), 89–101.

[11] Swan, M. Sensor mania! the internet of things, wearable computing, objective metrics, and the quantified self 2.0. *J. Sens. Actuator Netw. 1*, 3 (2012), 217–253.

[12] Wiese, J., Saponas, T. S., and Brush, A. J. B. Phoneprioception: enabling mobile phones to infer where they are kept. In *Proc. CHI 2013*, ACM (2013), 2157–2166.

[13] Zimmermann, G., and Vanderheiden, G. C. The universal control hub: An open platform for remote user interfaces in the digital home. In *Proc. HCI*, Springer (2007), 1040–1049.