

## Research Article

# InContexto: Multisensor Architecture to Obtain People Context from Smartphones

**Gonzalo Blázquez Gil, Antonio Berlanga, and José M. Molina**

*Group of Applied Artificial Intelligence, University Carlos III of Madrid, 28270 Colmenarejo, Spain*

Correspondence should be addressed to Gonzalo Blázquez Gil, [gbgil@inf.uc3m.es](mailto:gbgil@inf.uc3m.es)

Received 15 July 2011; Revised 29 December 2011; Accepted 3 January 2012

Academic Editor: Mo Li

Copyright © 2012 Gonzalo Blázquez Gil et al. 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.

The way users interact with smartphones is changing after the improvements made in their embedded sensors. Increasingly, these devices are being employed as tools to observe individuals habits. Smartphones provide a great set of embedded sensors, such as accelerometer, digital compass, gyroscope, GPS, microphone, and camera. This paper aims to describe a distributed architecture, called *inContexto*, to recognize user context information using mobile phones. Moreover, it aims to infer physical actions performed by users such as walking, running, and still. Sensory data is collected by HTC magic application made in Android OS, and it was tested achieving about 97% of accuracy classifying five different actions (still, walking and running).

## 1. Introduction

Traditionally, Internet has been accessed from a desktop computer. However, nowadays Internet access is also extended to the mobile phone or commonly called smartphone. The penetration rate of these devices is growing rapidly. For example, in the USA, 27% of mobile phone users had a smartphone at the end of 2010 in some countries of Europe (France, Germany, Italy, Spain, and the UK), smartphone penetration was even larger, reaching 31.1% (comScore 2011 whitepaper, <http://www.comscore.com/>). By 2011, smartphones sales are projected to overcome desktop computer. Hence, smartphone is becoming increasingly popular as a personal computer, and becoming the main computer and communication device in people's lives.

Indeed, nowadays, smartphones do not only provide internet access; besides, they are provided by a countless number of sensors. Microphones and digital cameras are the most common ones; however, they are being equipped with new sensors: accelerometer, gyroscope, compass, magnetometer, proximity sensor, light sensor, GPS, and so forth [1]. Taking advantage of these features, developers have created new amazing app in order to improve user smartphone experience [2]. The embedded sensors allow the device to adapt to environment conditions, use of battery, lighting

conditions, and sound. For example, light sensor controls screen brightness in order to preserve battery life. When the user is using the smartphone in a dark place, the screen brightness is reduced.

Moreover, thanks to smartphone mobile connection over different radio channels it is possible to consider them as a new sensor inside Ambient Intelligence (AmI) Environments. Smartphone ability to act based on sensory information extends user concept. Now the user is provided by a new set of sensory abilities. Smartphones are characterized by multiple sensors retrieving scenario context information in order to recognize inconspicuous activity of individuals and react to their needs.

First of all, in order to determine user needs, it is necessary to know their status and the context where it is located. User status is considered a combination of physical activity and emotional state. Their needs are different if a person runs doing a sport (probably he/she needs to complement his/her activity with music) or he/she runs to avoid a dangerous situation where the essential need is to track your position and advise the emergency services.

Activity recognition aims to perceive which activity is taking place. In these applications, high classification accuracy is always desired. Daily, human beings make ordinary actions such as a cooking, reading or watching TV, chatting

with other people or on the phone, and driving [3]. The ability of activity recognition seems so natural and simple for us; however, actually it requires complicated functions of sensing, learning, and inference for computers [4].

Traditionally, activity recognition is carried out through video systems like those described in [5, 6]. However, recent researches in activity recognition show that microelectro-mechanical systems (MEMSs) are becoming another way to face this problem [7]. They can return a real-time measurement of acceleration along the  $x$ -,  $y$ -, or  $z$ -axis to be used as a human motion detector.

In general, placing more accelerometers on different body positions improves pattern recognition performance [8]. At the same time, a wearable system must be inconspicuous and operate during long periods of time [9]. However, people are reluctant to wear strange devices over the body. In this case, smartphones are especially well-suited to accomplish this task since they have integrated MEMS and people consider them as friendly devices. Smartphones may obtain and process physical phenomena from embedded sensors (MEMS) and send this information to remote locations without any human intervention [10].

For that reason, it may be possible to consider a smartphone like a nonintrusive device to obtain activity Context from people [10]. Indeed, smartphones experience almost the same physical forces, temperature, and noise of the person who carries them out. If you track their actions, you are tracking people actions.

Although smartphones are considered as a single device, they provide several sources of information, mainly MEMS, internet connection, and human interaction to gather all this information in order to reach better results in activity recognition problem. Information fusion techniques [11] aim to *combine observations from a number of different sensors to provide a robust and complete description of an environment or process of interest*.

However, to handle all the information from the different sensors is pretty costly. In an extreme case, each sensor may have its own processor to manage the local data and cooperate with other sensor nodes. Traditionally, activity recognition system usually employs hard sensor (MEMS) nevertheless, there are other user information sources available in the smartphones. Users daily share their personal information on social networks sites, Facebook, Linkin, Twitter, and so on. These type of sensors are called soft sensors in information fusion researches which are referred as human observer that provides his/her point of view of something.

Information fusion techniques have been proved in several and complex scenarios [12], but they have not been used in smartphone devices. The principal achievements of information fusion systems are robustness, increased confidence, reduced ambiguity and uncertainty, and improved resolution. For that reason, taking advantages of information fusion techniques, an smartphone architecture has been deployed in order to collect user data and infer user context from smartphones.

In the literature, there are mainly two different ways to obtain user activity using MEMS. Classical techniques

just take into account ad hoc accelerometers sensors, for example, in [8] Bao and Intille present a multisensor system wearing six accelerometers around the body, which reaches about 80% of accuracy with different actions. In other research, Barralon et al. [13] describe an activity recognition system for eHealth applications where every patient wears a single sensor on the chest. The final results show that the systems are able to differentiate among walk or no walk over 80% of times. Although these systems reach good accuracy, in practice, they are quite uncomfortable, and also they are considered intrusive by the users. On the other hand, recent researches use smartphones to accomplish activity recognition problem. One of the most famous works is Miluzzo et al.'s [10]. The stronghold of CenceMe system is that it sees SNS as a site where you can share user activity information instead of a sensor where you can obtain user information.

Summarizing, this paper is focused on the description of inContexto, an information fusion architecture which retrieves smartphone context information as well as the user who carries it. Besides, inContexto architecture lays the guidelines to collect user information from every provided sensor in the smartphone, whether it is a hard sensor or soft sensor. Finally, inContexto activity recognition module was tested obtaining an overall performance over 97% of accuracy classifying still, walking, running, riding a bike and lying user actions. Besides, a public dataset has been published with the activity recognition data.

The paper is ordered as follows: Section 2 depicts the actual state-of-art of activity recognition using smartphones and information fusion techniques. Section 3 aims to describe the different user information sources using smartphones. Section 4 presents the proposal architecture according to the context sources, and preliminary results from an initial deployment indicate the potential for accurate, context-aware, and personalized sensing. Results of the chosen activity recognition techniques are shown in Section 5, and finally, Section 6 shows the conclusions and future work.

## 2. Related Works

Regarding the fields of sensor fusion and activity recognition separately, both are well treated in the scientific literature. In this section, firstly, we will focus on research works that use smartphones to retrieve user activity context and subsequently information fusion architecture is described in order to implement one in our work.

**2.1. Information Fusion Architectures.** Multisensor fusion architectures are not common in smartphone applications. Nevertheless, there are just a few researches [2] using this information fusion techniques. Ganti et al. presented an architecture for lifestyle monitoring, but it just collects data from sensors in the smartphone, and subsequently, information is sent to a computer desktop for data analysis.

In our case, information fusion is necessary to integrate the data from the different sensors (hard and soft sensors) in

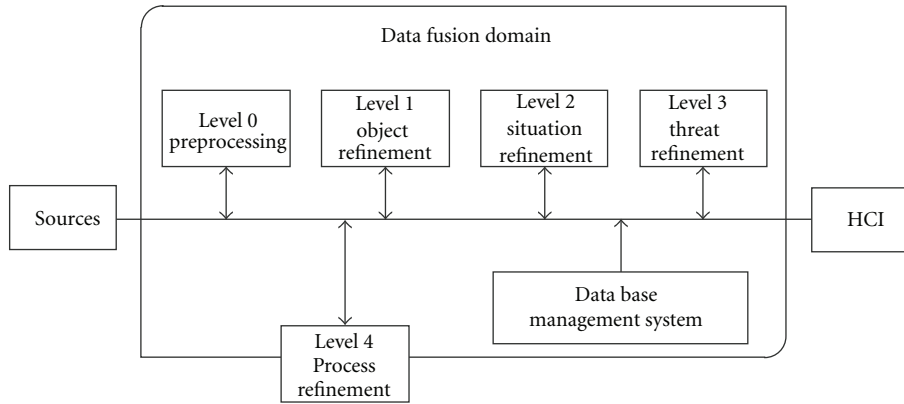


FIGURE 1: JDL information fusion model.

order to extract the relevant information on the users. Normally, data fusion architectures are based on a centralized system; however, this algorithm presents high computational cost increasing energy consumption. Thus, in order to prevent this problem, a distributed architecture is designed sharing computational process between the smartphone and cloud servers.

Below, most common general information architectures are described in order to consider pros and cons to use them in a mobile device.

**2.1.1. JDL Architecture.** Historically, data fusion methods were developed basically for military applications. The military community has developed a layout of functional architectures based on the joint directors of laboratories model for multisensory systems. In recent years, these methods have been applied to civilian applications [14] but never in mobile device.

The JDL model was never intended to decide a concrete order on the data fusion levels. Levels are not alluded to be processed consecutively, and it can also be executed concurrently. Figure 1 depicts JDL data fusion process high level model.

(i) *Level 0.* subobject data assessment is associated with predetection activities such as pixel or signal processing, spatial or temporal registration.

(ii) *Level 1.* At this level, to identify and locate objects is attempted. Hence, the object situation by fusing the attributes from diverse sources is reported. The steps included at this stage are:

- (a) alignment: processing of sensor measurement to achieve common time base and a common spatial reference,
- (b) association: a process by which the closeness of sensor measurement is completed,
- (c) correlation: a decision-making process which employs an association technique as a basis for allocation sensor measurement to the fixed or tracked location of an entity,

(d) correlator-tracker: a process which generally employs both correlation and fusion component processes to transform sensor measurements into states and covariance for entity track,

(e) classification: a process by which some level of identity an entity is established either as a member of a class, a type within a class, or a specific unit within a type.

(iii) *Level 2.* Attempts to construct a picture from incomplete information provided by level 1, that is, to relate the reconstructed entity with an observed event. Entities are associated with environmental, doctrinal, and performance data.

(iv) *Level 3.* It interprets the results from level 2 in terms of the possible opportunities for operation. It analysed pros and cons of taking one action over another one.

(v) *Level 4.* Process refinement is an element of resource management and used to close the loop by retasking resources (e.g., sensors, communications, and processing) in order to support the objectives.

Taking into account that JDL model is considered an abstract model, it is not a guideline to implement information fusion architecture. However, it makes easier to distribute which components should run on the cloud or in the mobile phone.

**2.1.2. Waterfall Fusion Architecture.** The waterfall IF model was proposed by Markin et al. [15] (see Figure 2). This architecture emphasizes the processing functions on the lower levels. However, waterfall model omits any feedback data flow instead of JDL model in which every level is interconnected. The relationship between waterfall architecture and JDL model is as follows.

- (i) Sensing and signal processing correspond to level 0.
- (ii) Feature extraction and pattern processing match with level 1.

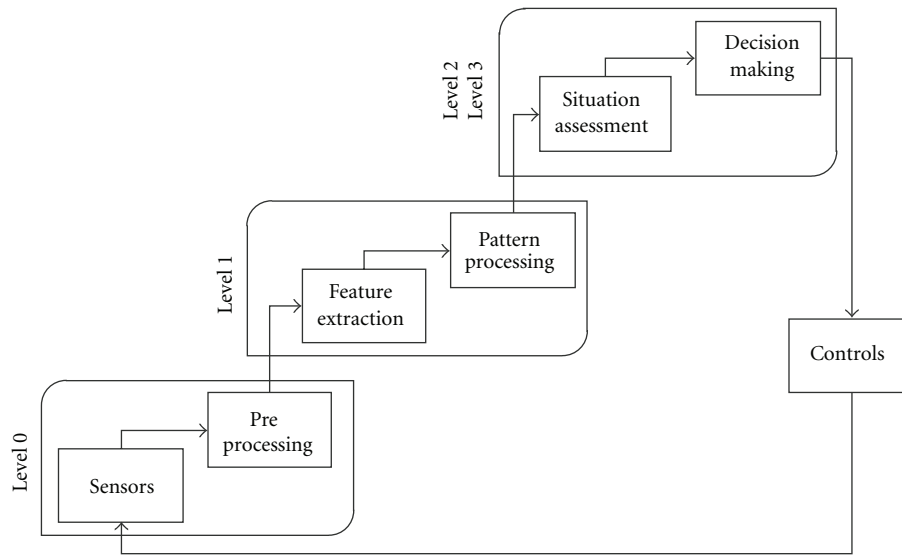


FIGURE 2: Waterfall information fusion model.

- (iii) Situation assessment is similar to situation refinement in JDL model, level 2.
- (iv) And finally, decision making corresponds to the third JDL level 3.

Although, waterfall model is more accurate in analysing the fusion process than other information fusion models, it presents some drawbacks, for example, the omission of any feedback data flow.

Taking into account pros and cons of both architectures, inContexto has been designed relying on JDL model. Its modularity gives us the advantages to divide some component on the smartphones and others on the cloud. Hence, it is able to operate in distributed systems.

**2.2. Mobile Phone Activity Recognition Architectures.** Normally, in the literature, there are two kinds of researches to obtain activity using mobile devices. The first one has been focused on ad hoc solution, and the second one and more recent is using smartphones solutions. Each activity recognition architectures are briefly described below with information on how inContexto builds on or differs from the ones described.

**2.2.1. Ad Hoc Activity Recognition Architectures.** Barralon et al. [13] work describes an MEMS architecture and shows the results of the time spent in three postural states (lying, sitting, standing) and the periods of walking in an eHealth scenario using a unique accelerometer, placed on the patients chest.

The study determines the global position of the patients of the sensor wearer, and they calculate the position of the patient considering the inclination of the sensor in every axis and then quantify this value. Finally, the study was made to evaluate the health of the patient, and they obtain about 76% of accuracy rate.

On the other hand, Bao and Intille describe an architecture [8] to acquired human motion using five biaxial

accelerometers worn on different parts of the body from 20 subjects. Extracted features from each accelerometer were: signal mean, energy, frequency-domain entropy, and correlation of acceleration and subsequently classify using a decision tree, obtaining an overall accuracy rate of 84%. Although they reach a good accuracy, this architecture presents, a big problem, to wear five devices over the body.

**2.2.2. Smartphone Activity Recognition Architectures.** One of the most notable contributions presented up to now in mobile phone activity recognition is called CenceMe [16]. A mobile sensing architecture to obtain and share user physical activities on a social network.

Although CenceMe does not use social networks sites to collect information (they only use accelerometer), they introduce SNS into the activity recognition field sharing user activity on Facebook. The proposed architecture is split in three layers (sense, learn, and share):

- (i) sense layer aims to collect raw sensor data from sensors embedded in the phone in the Apple iPhone in order to track body movements,
- (ii) in learn layer, they propose to use a variety of data mining techniques to infer user rules. These techniques are used to interpret three-axis accelerometer raw data extracted in the sensor layer,
- (iii) their approach aims to share activity information in a web portal where sensor data and inferences are easily displayed.

Chon and Cha [17] present LifeMap architecture, a Smartphone-based context provider for location-based services. Authors split their architecture in four components:

- (i) all the sensors are placed on the low level; this level sends the obtained information to the component manager where information is processed

and provide, high-level information, using high-level information from the component manager.

- (ii) the context generator generates a point of interest (POI) which contains the user context. The context map is stored in a database to match and aggregate user contexts,
- (iii) and finally, the database adapter is an interface to provide user context to other applications.

Our work differs from existing solutions in that it does not rely on external mobile devices nor the accelerometer position when the user wears it. In contrast, using a smartphone as a nonintrusive device permits to obtain user movements with embedded sensors. On the other hand, GPS only solutions work well for classification of activities with different speed; however, it is necessary for another sensor to distinguish between similar speed activities such as riding a bike or running. Accelerometer-based technique presents best results in that way. Finally, the most significant difference between our work and existing works is that we describe an architecture to handle information from different information sources (Accelerometer, Gyroscope, GPS, SNS, etc.) using information fusion techniques. Although one sensor was offline, it is possible to generate user information handling the other sensors.

Table 1 shows a summary of works that have taken place in this space along with the types of activity modes inferred, the test user base, and the classification accuracy.

### 3. Describing Smartphone Context

First of all, in order to use context correctly, it is crucial to define what researchers think context is. In general, context aware is represented by applications which change their behaviour according to the conditions around them, in this case the smartphone conditions. Applications and services react specifically to their surroundings, location, and time. Summarizing, their behavior is able to change according to circumstances.

In 1994 was introduced the term context-aware computing by Schilit and Theimer [18]. *They defined context as a software which adapts according to its location of use, the collection of nearby people and objects, and changes to those objects over time.* Subsequently, some other researchers try to formally define context, for example, Schmidt et al. [19] define context as knowledge about the user's and IT device's state, including surroundings, situation, and location. it one of the most accurate definition given by Dey and Abowd [20]. they defined context as: *any information that can be used to characterize the situation of entities (i.e., whether a person, place, or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves.*

Hence, everything in the world may be considered as an entity, for example, a bedroom has its own context, the people who is lying in, number of furniture, and so forth. Dey and Abowd defined the three kinds of entities:

TABLE 1

Research	Classes	Sensors	Mobile
CenceMe [10]	Still, walk, run	Accelerometer	Yes
lifeMap [17]	Still, walk, motor	Accelerometer, magnetometer, wifi, and GPS	Yes
Borriello [3]	Still, walk, stairs up and down, riding elevator, and brushing	GSM, Wifi	No

- (i) places: it represents a point or an area, for example, buildings, rooms, village, and so forth,
- (ii) person: an individual or groups of people,
- (iii) objects: electronic devices, physical objects, and so forth.

Each entity is characterized by four categories (Identity, location, status or activity, and time). According to Dey and Abowd's definition about context and entity, this work presents smartphone entity representation (see Figure 3).

(1) *Identity.* In order to identify one person is possibly to use different sources, hardware or software, hardware identification, as MAC address, presents several problems because you are identifying the smartphone instead of the person who carries it. Hence, if another user manipulates the same device, there will be identification problems. However, using software identification as Facebook platform (FP), this problems would be solved.

(2) *Location.* Location aware could be the main factor in the development of context applications. Nevertheless, location aware is only one aspect of context aware as a whole [21]. Location context may be described as an application dependent on the geographical location. Location answers the questions where is the action taking place? For example, it is possible to define a running action; however, it could be interesting to define where is he/she running? and where is he/she running to? To obtain location using mobile phones is really simple however, in outdoor environments; GPS provides a good solution to determine the location of mobile devices; however, in GPS-denied areas such as urban, indoor, and subterranean environments, unfortunately, an effective solution does not exist. Besides, every location system provides in its own way location data. Recently, W3C has reloaded a Geolocation API [22] to standardize an interface to get back the geographical location information for a client device.

(3) *Status or Activity.* Talking about status it is necessary to differentiate user status, and mobile phone status. smartphone status mainly refers to communication behaviour: calls and calls attempts, sent and received SMS, SMS content, battery level, wireless connections, and so forth. On the contrary, user status does not refer just to her/his calendar (working, sleeping, free-time, etc.), otherwise the relevant





FIGURE 3: Entity representation using smartphones.

information about the user, normally, is included in the user profile as an instance (name, date of birth, where is she/he was born, etc.). As it was described previously, people movements are reflected in mobile devices sensors. The generated information can be used to identify different activities (e.g., running, walking, standing, cycling etc.) that the user is performing. These kinds of actions are obtained by low-level sensors provided by the mobile phone (accelerometer, Gyroscope, light sensor, microphone, etc.). For example, accelerometer is able to describe the physical movements of the user carrying the phone.

(4) *Time*. activities taken by the user or the user's status do not have any meaning if it is impossible to set the action in a place and in time. For that reason time is an essential in context-aware applications.

**3.1. Describing Sources of Mobile Context.** According to entity representation (see Figure 3), this section aims to describe different sources of context in smartphones, and also it matches user actions with smartphones sensors.

**3.1.1. Hard Sensor.** The camera and microphone are probably the most used sensors in AML systems. However, these sensors present several issues. In order to retrieve user information, it is necessary to process all the information and transform it from raw data to features.

Basically, using this kind of sensors, it is possible to obtain basic actions taken by the user such as running, walking,

standing, talking, and listening music. These actions are obtained by low-level sensors provided by the mobile phone (accelerometer, gyroscope, light sensor, microphone, etc.).

- (1) **Accelerometer:** A triaxial accelerometer is a sensor that returns a real-valued estimate of acceleration along the  $x$ -,  $y$ - and  $z$ -axes from its velocity. Accelerometers can be used as motion detectors as well as for body position and posture sensing [23]. Collected data from the accelerometer has the following attributes: time, acceleration along three axes ( $x$ ,  $y$ , and  $z$ ), not including gravity.

Accelerometer provides data from the origin of coordinates of the device which is placed in the lower-left corner with respect to the screen, with the  $X$ -axis horizontal and pointing right, the  $Y$ -axis vertical and pointing up, and the  $Z$ -axis pointing outside the front face of the screen. In this system, coordinates behind the screen have negative  $Z$ -values (Figure 4). Hence, if the mobile device is worn on a pocket, it is not clear which axis or axes represent the real world coordinates. In the next section, it is presented how to transform, using digital compass, coordinates from smartphone representation to real-world one which will be described.

Accelerometer sensor is well fit to be used to infer pedestrian movements due to acceleration data of walking or running displays distinct phases and periodicity of the signal; however, it is very difficult to differentiate transportation modes.



FIGURE 4: Smartphone Android OS coordinates origin.

- (2) Digital compass provides two kinds of measures: the first one is the orientation whose values are in radians/second and measure the rate of rotation around the  $x$  (roll),  $y$  (pitch), and  $z$  (yaw or azimuth). Also, the coordinate system is the same as is used for the acceleration sensor.

Digital compass reports the angle between the magnetic north and the mobile phone's  $y$ -axis (orientation measurement). All values are in microTesla (uT), and it measures the ambient magnetic field in the  $x$ ,  $y$  and  $z$  axes.

This sensors do not have a concrete value describing user actions, but it is usually used to determine user movements direction.

- (3) Gyroscopes are the most commonly used sensors for measuring angular velocity and angular rotation in many navigation and homing applications. They measure how quickly an object rotates and, specifically, measure the rate of rotation around the  $X$ -,  $Y$ - and  $Z$ -axes. The coordinate system is exactly the same as is used for the acceleration sensor. Gyroscopes are the only inertial sensors that provide measurement of rotations without being affected by external forces, including magnetic or gravitational or fabrication imperfections.
- (4) Location sensor: there are three ways to locate the smartphone, first of all using a GPS, in this case every smartphone provides an assisted GPS [24]. A-GPS improves the performance by adding information, through another data connection (Internet or other),

more than unassisted GPS in order to receive and process signals as computationally costly, minimizing the amount of time and information is required from the satellites. The A-GPS receiver uses satellite to locate itself, but it can do more quickly and using weaker signals than an unassisted GPS. Normally, an A-GPS provides 2–4 meters error.

The second way to locate the smartphone is using GSM cell tower triangulation. This technique is reduced and more accurate than GPS; however, the energy consumption is reduced as well. According to the application goals, it is necessary to balance the accuracy and the energy consumption, and it could be enough a coarse location (GSM) instead of a precision location (GPS).

Finally, using Internet connection (Wifi) is possible to locate the smartphone thanks to W3C that has reloaded a Geolocation API to standardize an interface to get back the geographical location information for a client device (Geolocation API <http://dev.w3.org/geo/api/spec-source.html>).

**3.1.2. Soft Sensor.** Social networks sites (SNSs) are increasingly popular these days. In [25] is described social network site as: *Web-based services that allow individuals to (1) construct a public or semipublic profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system. The nature and nomenclature of these connections may vary from site to site.*

Each SNS is implemented with specific features; however, all of them have a common point which consists of visible profiles. Daily, SNS users share their personal information, and SNS manage as uncountable gigabytes of useless user information. Why do not we use these data to obtain user context information?

Typically, user profiles include descriptors such as age, location, and interest schools attended. User profiles are becoming more precise: music preferences, movies, clothes, friendship relationships, personal agenda, and so forth.

**3.2. Context Action Concept.** Context action concept (see Figure 5) is the result of combining all the different contexts (identity, location, activity, and time context) according to Dey and Abowd's definition as well as each category which describes a concrete action. This paper distinguish two kind of activities: basic activities (e.g., walking, talking, running, cooking etc.) which cannot be decomposed into more simple actions and composite activities (context activities) which are composed by various simple actions (e.g., giving lectures, talking, standing, and relationships with other people).

For example, consider the following scenario, someone is sitting in her/his living-room watching TV. The accelerometer and the microphone may detect whether the user is sitting (Motion-Activity context) or the user is near a sound source (Sound-Activity context). If you use the both context and it is able to locate the action, Location context, (location is happening in the living room) it could figure out that the

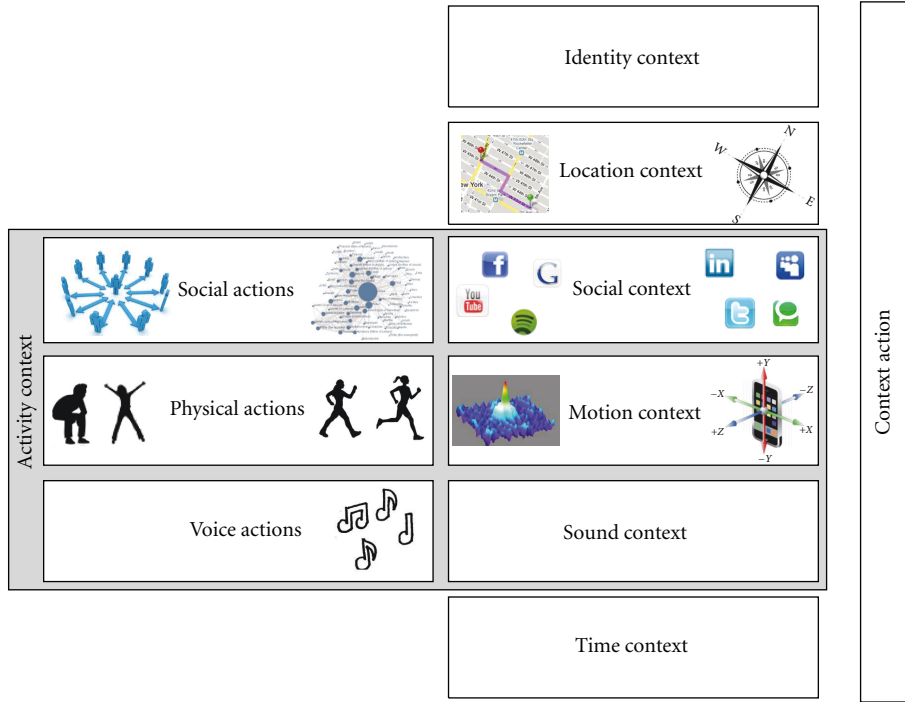


FIGURE 5: Context Action concept.

person is sitting in the living room watching TV (Context Action).

**4. InContexto: Architecture Definition**

In this section, inContexto is described (Figure 6) a multi-modal architecture to obtain context from a user who carries out an smartphone.

It is based on the JDL model which proposes five different levels in order to transform input data into decision. These levels are called signal feature assessment (L0), entity assessment (L1), situation assessment (L2), impact assessment (L3), and process assessment (L4). Observational data may be combined from the raw data (or observation) level to a state vector level, or at the decision level.

Combining information fusion and activity recognition techniques in a smartphone is not a vana task due to energy restrictions and the computational cost of these techniques. Hence, it is important to highlight that, nowadays, it is not clear what architectural components should run on the device and what should run on the cloud. In this case, it is proposed that L0 and L1 are implemented in the smartphone; on the contrary, the other ones are executed in backend infrastructure.

InContexto is implemented following a distributed architecture where a communication component is designed to associate the smartphones with the backend server.

*4.1. Data Collection Level 0.* Data collection level aims to transform raw data (accelerometer, gyroscope, location, light sensor, and soft sensors) into processed data easy to manage by the features selection level.

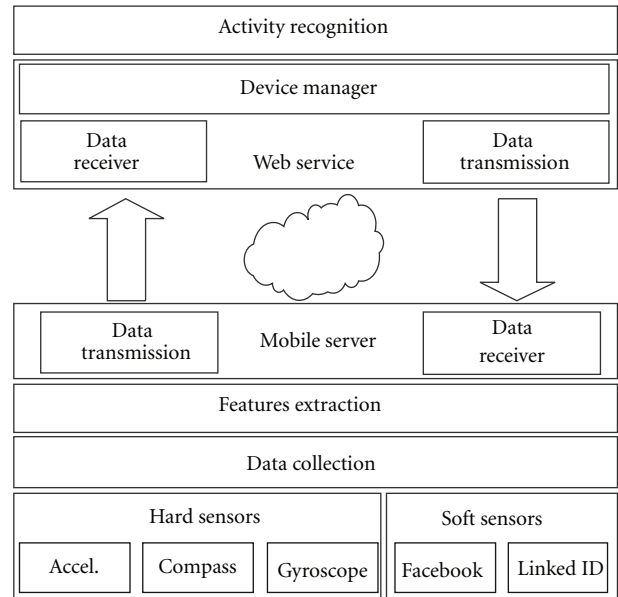


FIGURE 6: Overall overview of inContexto architecture.

It is largely to recall that the presented architecture is developed to obtain user context in a nonintrusive way. For that reason an smartphone instead of ad-hoc sensors has been chosen since smartphones can supersede these sensors, by reducing user’s rejection since they are considered daily communication tools.

Hard sensor data is accessed through Android OS API, in concrete sensor manager class which provides methods to



obtain all the mobile sensors. A low-level sensing module continuously gathers relevant information about the user activities using sensors. Thanks to Android OS that provides background processing, it is possible to run services without human control.

In order to provide an effective and efficient description of patterns, preprocessing is often required to improve performance, removing noise and redundancy in measurements. In this study, the accelerations and azimuths of the pedestrian were mainly collected with a smartphone with Android operating system. Android OS provides four different sampling frequencies. These frequencies are not fixed and depend on the operating system, and there is no control over it.

The sampling frequency can be adjusted according to the action studied. In this case, relying on the next study [26], the sampling frequency range requiring to obtain human actions is from 0.6 Hz to 2.5 Hz. Consequently, to prevent aliasing problem, the Nyquist-Shannon sampling theorem is as follows:

$$F_E \geq 2 * F_{Max}. \quad (1)$$

Sampling frequency is not clear in Android OS since it provides only four different sampling frequencies (fastest, game, normal and UI), and the value is not constant. The value depending on the computational workload of the smartphone but normally fastest sampling frequency is 50 Hz.

Besides, accelerometer and GPS raw data have been stored into a sliding window of 512 samples (approximately 5 seconds), 256 of which overlap with consecutive ones. Sliding windows with 50% overlap have been defined in previous works [8].

Besides, extracting features from a window is a fairly effective way to preserve class separability and can represent the characteristics of different activity signals in each window.

Social networks have plenty of information, and most of this information is unused. Thus, the selected features collected from different social networks are social network ID, social network name, born on, lives in, and relationships with others.

Acquiring context from soft sensors is not a banal work. Social network information is accessed thanks to provided APIs by the SNS. Hence, it is necessary that the user log into the site. In this first contact, inContexto was connected with facebook friends and smartphone agenda in order to create ties with people.

Facebook platform (FP) is a connect service which lets third-party application to retrieve SNS features [27]. Besides, FP is an open standard that describes how users can be authenticated in a decentralized manner (Figure 7), obviating the need for services to provide their own ad hoc systems and allowing users to consolidate their digital identities [28].

Facebook platform leverages OAuth 2.0 for authentication and authorization process. First of all, inContexto user authenticates using Facebook as an identity provider. Later,

Facebook sends a message that permits inContexto access to the user basic profile (name, profile picture, gender, and friend list).

*4.2. Smartphone-Placed Problem.* Although there are multiple researches that show the best position to wear sensors [3], sensor placement is a real problem in activity recognition based on MEMS. Minimal changes in sensor placement or orientation create different data and a wrong activity classification. Although previous work suggests that the best place to wear this sensor is the hip [8]; however, in this work, the data collection process was made while the smartphone was worn in a trousers pocket. Hence, it is mandatory to create a system which enables our system to assume a random and possibly changing orientation for the mobile phone.

*4.3. Features Extraction Level 1.* JDL model depicts that in this level is made object detection process. Although normally object detection is not trivial task, in this case if it is due to tracking mobile phone user actions.

Features extraction level involves the extraction of symbolic features from sensor data obtained in L0. Features can be defined as the abstractions of raw data. The raw sensor data acquired by phones, independent of the amount or source (e.g., accelerometer, camera), are worthless without interpretation. The objective of feature extraction is to represent an activity with the main characteristics of a data segment.

This level aims to process and select which features are better to identify an action. The module processes several sensor observations (a sliding window) into a vector features that help discriminate between activities. The features extraction level is also implemented in the mobile phone.

In the literature, mainly there are two types of extract features from accelerometer raw data. The first ones are those techniques which use frequency properties analysis (DWT, CWT, and STFT), and secondly those that create a vector with statistical methods (SMA, signal mean, correlation, etc.). Barralon et al. [29] present a comparison study using wavelet and frequency features. They present that walking mode is characterized either by the foot impacts on the floor or by chest oscillations. Although the CWT and DWT methods present same performances, the CWT suffers time-consuming problem. Summarizing, frequency methods provide several advantages, one in particular is its resilience to signal level variations.

On the other hand, statistical features presented in [30] are other possible features to infer activities using accelerometer raw data. In this case, mean, standard deviation, and average energy are used which is well-fitted to distinguish sedentary activities from athletic activities; and also correlation between axes. Signal magnitude area (SMA) used in [31] provides good results. SMA is equal to the sum of acceleration magnitude summations over three axes of each window normalized by the window length, giving a total of thirteen attributes (Figure 8).

Furthermore, soft sensor L1 module aims to generate a meta-agenda collecting information for each available

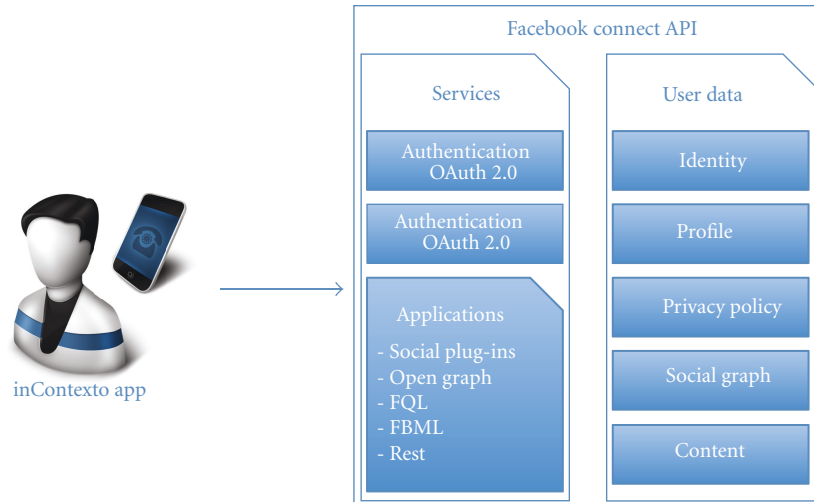


FIGURE 7: Facebook platform connect architecture.

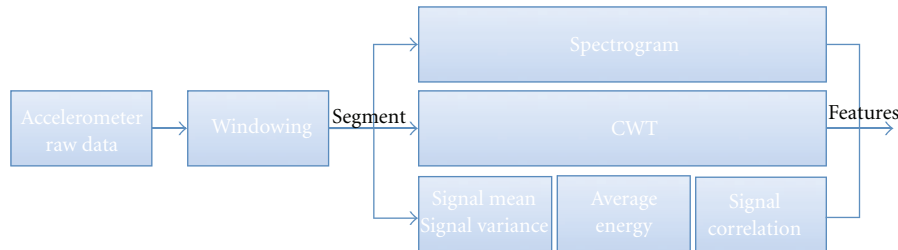


FIGURE 8: Hard sensor smartphone inContexto architecture.

SNS and the smartphone agenda (SA). The meta-agenda is composed by every person the user knows either on Facebook or SA. Probably, most of these contacts have an instance in both sides (SA and Facebook), therefore, they are joint in the same meta-agenda contact according to the email, name, or mobile phone number coincidence. Meta-agenda permits to create relationships between people and inContexto user (Figure 9).

Moreover, user meta-agenda contact profile is updated with all SNS information available. Summarizing, this new profile contains basically name, date of birth, Mobile phone number, email, and relationships. Besides, some other optional features are collected, for example, likes and dislikes, school degrees, employment, and so on.

**4.4. Mobile Server, Web Service, and Device Manager.** Both components aim to communicate the smartphone with the server. One of them (Mobile server) is implemented in the smartphones, and the other one (web service) is on the server. The Web service module is developed as web service which is designed to support interoperable machine-to-machine communication over a network. Web-services provide an interface which describe message format, specifically, Web services description language (WSDL) [32]. Device manager allows web services to view and control the devices attached to the service. When a device is not online, the web

server keep the last device's IP address for a while, waiting for a new connection.

**4.5. Activity Recognition Level 2.** JDL level 2 uses the vector features provided by level 1 in order to infer what single activity is the individual engaged in. In this component, it would be implemented the pattern recognition techniques (supervised learning, probabilistic classification, and model-based or instance-based learning) to figure out the action.

Activity recognition level fetches the features selected by the last level and classifies them in order to return the current activities walking, running, sitting, standing, listening to music, talking, and so forth.

J48 decision tree has been chosen since they present several advantages over traditional supervised classification methods used in smartphone sensing. In particular, decision trees are fast in reasoning, so it is a crucial feature in a real-time system like this. In addition, they allow for missing values since it is defined as a classification procedure that recursively partitions a data set into smaller subdivisions. Finally, decision trees are easily interpretable to developers because of the structure.

Level 2 processing develops a description of current user contact actions in the context of their environment. Distributions of individual objects (defined by level 1 processing) are examined to aggregate them into operationally meaningful combat units and weapon systems.

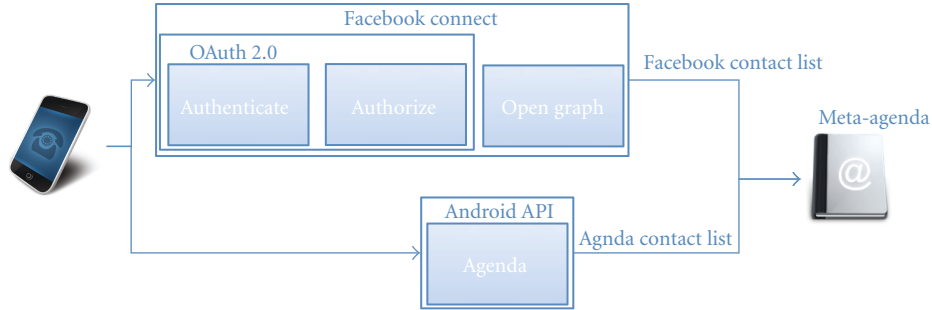


FIGURE 9: Soft sensor smartphone inContexto architecture.

If the motion context detects an activity, a corresponding message is emitted to the next level (L3), so that other sensors that may be interested in this activity will be triggered (e.g., social context).

**4.6. High Level Action Reasoning Level 3.** Finally, high level action reasoning level aims to compose all the received actions from the activity recognition level into a context action for each user. Beyond the standard reasoning model based on the subsumption ontology mechanism, it is possible to perform rule-based inferences using a description logic inference engine. At the beginning, these rules would be described by their own user in order to teach the system.

All the simple actions taken by the user would infer a global action with any relation between the other ones. For example, low level promotes running, listening to music, and free time context for a particular user. Maybe all these actions do not make sense in an individual way, but altogether, it could be possible to infer that the user is doing exercise.

For example, the accelerometer and the microphone may detect whether the user is sitting or the user is near to a sound source. If you use both the actions and it is able to locate the action (living room), it could figure out that the person is sitting in the living room watching TV (location action).

## 5. Experimentation

In order to generate enough trajectories examples to make the training process, the training data was made in a different way. This process has four steps: data collection, trajectories generation, features extraction, and Training process.

Eight male participants between the age groups of 20–37 years have been participated as subjects for the empirical data collection experiments. Users were encouraged to wear the device as much as possible in either of their pockets and perform three different activities (running, walking, and standing up).

Besides, the study relies on the power of the GPS to tag every action that the mobile phone takes. On one hand, every action which takes place outdoor (running, standing, and walking), the data acquisition layer records the speed and precision from the GPS (autotagging).

Finally, a dataset was created for the research community, and it is available online on this website (GIAA Web page <http://www.giaa.inf.uc3m.es/>).

TABLE 2: Dataset duration (min) and samples for each activity.

	Running	Standing	Walking
Instances	150,718	345,318	240,825
Minutes	32.36	77.42	40.5

**5.1. Data Collection.** In this study, the accelerations and azimuths of the pedestrian were collected with Android OS devices. The created dataset has the following attributes: 3-axis accelerometer values in the smartphone Cartesian reference system, 3-axis compass values, 3-axis accelerometer values in the real-world reference system, GPS precision, and GPS speed. Table 2 shows the number of instances for each activity. In this approximation the architecture just acquire data from digital compass, accelerometer, and Gyroscope.

Computing the inclination matrix  $I$  as well as the rotation matrix  $R$  transforms a vector from the device coordinate system to the world's coordinate system which is defined as a direct orthonormal basis.  $I$  matrix is a simple rotation around the  $X$ -axis and the rotation matrix  $R$  which is the identity matrix when the device is aligned with the world's coordinate system

$$a_{\text{realworld}} = a_{\text{smartphone}} * R * I, \quad (2)$$

where  $I$  matrix is a simple rotation around the  $X$ -axis and the rotation matrix  $R$  is the identity matrix when the device is aligned with the world's coordinate system.

Figure 10 represents the device accelerations and shows the changes of the three forces depending on the movement taken by the user (running, walking, standing). On the other hand, Figure 11 represents the transformation from the smartphone reference to the real-world reference.

This work uses GPS in order to obtain the speed of the person who is doing the action; thus, the classifier output value is the mean of the speed in the sliding window.

Three different vector features are compared in order to decide what is the best one. The first one is based on spectrogram function ((STFT), short-time fourier transform). A spectrogram is a time-varying spectral representation that shows how the spectral density of a signal varies with time. The second one is continuous wavelet transformation used to split a continuous-time signal into wavelets. Unlike Fourier transform, the CWT is able to construct a time-frequency representation of a signal which offers very good time and

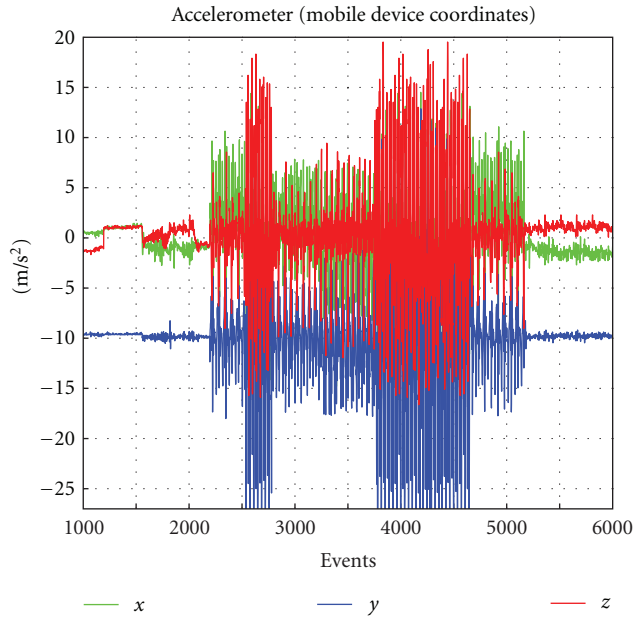


FIGURE 10: Sensing level: device 3 axes accelerations.

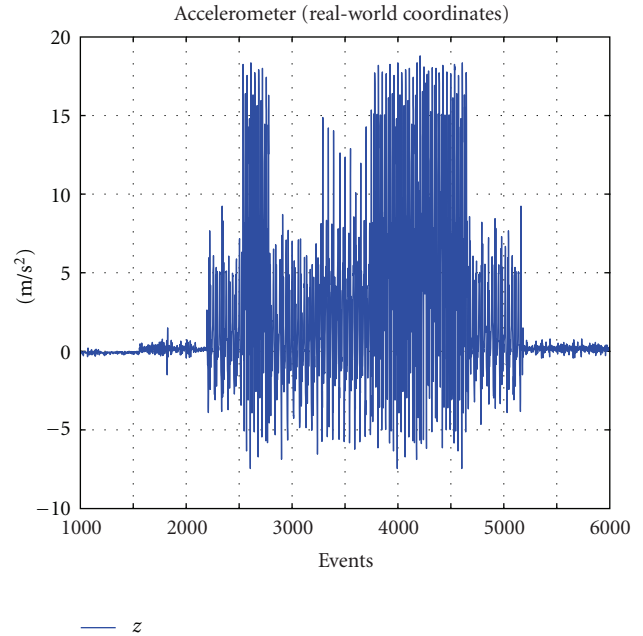


FIGURE 11: Real-world vertical acceleration.

frequency localization. Both of these techniques (STFT and CWT) present several vales (higher than 150), however, all of them are not necessary. For that reason, only the first 25 frequencies were selected such possible features. Besides, the signals need to be transformed from smartphone coordinate to real-world coordinates. Statistical method consists of eight features, consisting of signal mean, correlation between axes, energy, and variance, which are usually extracted from the triaxial acceleration data.

**5.2. Samples Generation.** It is necessary that a big amount of samples or trajectory (vector features) make correctly the training process. However, it is quite costly to generate enough samples to make this process.

In this case, the selected sample is made semiautomatically. First of all, we have 3 files corresponding each activity (running, walking, and standing up). Subsequently, a Java program has been created to mix all the activities generated a unique trajectory. Finally, all the generated trajectories have been stored to continue the pattern recognition process. However, there are some requirement to make this trajectories as real as possible:

- (1) all the trajectories start with a standing up action;
- (2) the next action could be the action besides (Figure 12) or the same action again;
- (3) the minimum duration of each action is 2 seconds and the maximum is 7 seconds;
- (4) finally, each trajectory consists in 10 actions.

When the trajectories generation process is over, it is necessary to discretize the speed value due to J48 tree users nominal values. Thus, all the samples are discretized in 5 classes.

- (i) Stop class: it is when the GPS speed measurements are less than 1 km/h.
- (ii) Walking class: speed value from the GPS is more than 1 km/h and less than 4.
- (iii) Walking fast class: in this case, GPS speed values are among 4–6 km/h.
- (iv) Running class: it is when the GPS speed measurements are more than 6 km/h and less than 10 km/h.
- (v) Running fast class: finally, the last class takes the GPS speed values more than 10 km/h.

Finally, 1000 trajectories were created to infer activities. Every trajectory is different, in duration and actions, from the other. Weka (Weka web page <http://www.cs.waikato.ac.nz/ml/weka/>) was used as the machine learning tool in this paper, and it is necessary to transform data into arff format.

The selected machine learning algorithm is a J48 classifier which is the Weka version from the C4.5 decision tree algorithm. J48 was chosen to give results in tree model which can be easily transformed into real-time applications.

The selected parameters for the J48 decision tree are:

- (i) confidence factor = 0.25,
- (ii) minimum number of objects = 2,
- (iii) unpruned = false,
- (iv) test options = 10-fold cross-validation.

**5.3. Results.** After processing the training and testing sets with the J48 classifier in Weka, the results are highly accurate in vector and spectrogram features; however, results are poorly accurate if CWT features extraction is used.

Table 3 shows results from each selected technique to extract features. The best implemented technique is vector,

TABLE 3: Features of J4 tree generated by Weka.

	Features	Leaves	Tree size	Accuracy	Mean absolute error
CWT	25	8741	17481	62.85%	0.1631
Spectrogram	25	1007	2013	95.63%	0.0198
Vector	12	648	1295	97.20%	0.0131

TABLE 4

Research	Classes	Sensors	Time (h)	Accuracy
Cenceme [10]	Still, walk, run	Accelerometer	4	78%
LifeMap [17]	Still, walk, motor	Accelerometer, magnetometer, Wifi, and GPS	28	91%
Borriello [3]	Still, walk, stairs up and down, riding elevator and brushing	GSM, Wifi	7	84%
InContexto	Still, walk and run	Accelerometer	2	97%

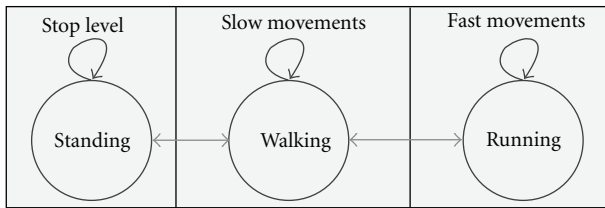


FIGURE 12: Generation trajectories model.

which is not only more accurate than the other ones; otherwise, it provides the smallest tree generated. The size of the tree is very important since this technique will be implemented in a real application in a mobile phone. A bigger size of the tree causes more energy consumption according to the increase of CPU cycles. Another way to study the quality of the feature extraction techniques is using the confusion matrix (Figure 13).

CWT technique is the worst of all the studied techniques, besides, it does not present any advantage over the other ones. Secondly, spectrogram achieves great results; besides, this technique uses only one signal (vertical movement in the real world) in order to obtain the spectrogram although confusion matrix shows that it is possible to classify an instance in a class not next to the real class. Thus, the best performance (high accurate and less tree size) is presented by vector technique. Besides, confusion matrix figure shows that vector features extraction just fail with the class near the one which is classified (e.g., running instead of running fast) (Table 4).

**5.4. Energy Consumption Problem.** Resource constraints power consumption is the main factor affecting smartphone activity recognition system. It is highly desirable that inContexto architecture is running as long as possible.

Normally, embedded sensors are placed in the same chipset. In this case, it is used as an HTC magic smartphone which is AK8976A marketed by Asahikasei Microsystems

Co., Ltd (AKM). This chipset includes a 6-axis electronic compass that combines a 3-axis geomagnetic sensor with a 3-axis acceleration sensor in an ultrasmall package. Consequently, whether your applications query the accelerometer, compass or both, it consumes the same energy power.

Besides, communication process between smartphone and the cloud consumes energy. This is very expensive and takes a toll on battery life. Reducing the number of upload, the system preserves energy. Considering computational power and energy consumption restrictions, it is necessary to select a good technique in order to balance the energy consumption and the global precision of the system. One way to do that is doing features extraction process on the mobile phone and creating a sliding window to reduce the amount of data.

Figure 14 shows inContexto energy consumption during 30 minutes. This value is about 21% of the 20% of the smartphone total energy. InContexto energy consumption is not high comparing with the screen value (66%) or Wi-Fi (11%).

## 6. Conclusions

In this paper, it inContexto was presented a distributed architecture to obtain mobile context from smartphones. The proposed architecture distributes the processing load between smartphone and a server placed on the cloud. With this approach, the energy consumption is reduced, increasing autonomy to offer a better service to the user. Also, a study comparing three different techniques in order to infer activity recognition using a J48 decision tree was presented. Besides, the study relies on inContexto architecture to collect accelerometer data. Overall, the presented work further demonstrates that using a mobile phone providing with accelerometers is enough to infer actions that user is doing.

Besides, a smartphone entity is defined according to Dey and Abowd's definition of context aware. An entity is defined as a smartphone which provides hard or/and soft sensors, provides internet connection everywhere, and is portable.



CWT						Spectrogram					
a	b	c	d	e	<-- classified as	a	b	c	d	e	<-- classified as
9462	3357	269	67	132	a = Stopping	12815	472	0	0	0	a = Stopping
3115	16957	2804	997	1852	b = Walking	473	24592	640	20	0	b = Walking
302	3601	2025	658	1652	c = WalkingFast	0	667	7217	352	2	c = WalkingFast
114	1277	775	1100	2856	d = Running	0	23	395	5501	203	d = Running
169	2144	1439	1860	20266	e = RunningFast	0	0	4	212	25662	e = RunningFast

Vector					
a	b	c	d	e	<-- classified as
12979	308	0	0	0	a = Stopping
286	25047	392	0	0	b = Walking
0	360	7639	239	0	c = WalkingFast
0	1	247	5676	198	d = Running
0	0	0	187	25691	e = RunningFast

FIGURE 13: Confusion matrix of each technique.



FIGURE 14: InContexto energy consumption.

Activity recognition systems identify and record in real-time selected features related on user activity using a smartphone. The paper describes how to face this problem using information fusion architecture in smartphones. Besides, it describes sensing module process, that is, one of the most important components in activity recognition systems.

The best given solution obtained an overall accuracy of 97.20% well to classify instances of 79250 different actions. This solution is a vector composed by energy, mean, standard deviation, and correlation of each axes.

The flexibility of the Android OS along with the phone's hardware capability allows this system to be extended, for example, creating an application which is able to send an sms or call to your relatives if you are doing strange movements.

Considering future works extends the development of the server module, and also it will extend activity classifier to more complex activities (group activities, interaction activities). Context information will be used to infer the user's emotional state, for example, according to the social network state, the music which is listened at the moment, the place where the user exists and using other hard and low sensors.

## Acknowledgment

This work was supported in part by Projects CICYT TIN2011-28620-C02-01, CICYT TEC2011-28626-C02-02, CAM CONTEXTS (S2009/TIC-1485), and DPS2008-07029-C02-02.

## References

- [1] N. Eagle and A. Pentland, "Reality mining: sensing complex social systems," *Personal and Ubiquitous Computing*, vol. 10, no. 4, pp. 255–268, 2006.
- [2] R. K. Ganti, S. Srinivasan, and A. Gacic, "Multisensor fusion in smartphones for lifestyle monitoring," in *Proceedings of the International Conference on Body Sensor Networks (BSN '10)*, pp. 36–43, June 2010.
- [3] J. Lester, T. Choudhury, and G. Borriello, "A practical approach to recognizing physical activities," in *Proceedings of the 4th International Conference on Pervasive Computing (PERVASIVE '06)*, pp. 1–16, May 2006.
- [4] L. Liao, *Location-based activity recognition*, Ph.D. thesis, 2006.
- [5] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 28, no. 6, pp. 976–990, 2010.
- [6] J. Gómez-Romero, M. A. Serrano, M. A. Patricio, J. García, and J. M. Molina, "Context-based scene recognition from visual data in smart homes: an Information Fusion approach," *Personal and Ubiquitous Computing*. In press.
- [7] A. Avci, S. Bosch, M. Marin-Perianu, R. Marin-Perianu, and P. Havinga, "Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: a survey," in *Proceedings of the 23rd International Conference on Architecture of Computing Systems (ARCS '10)*, pp. 1–10, 2010.
- [8] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," in *Proceedings of the 2nd International Conference on Pervasive Computing (PERVASIVE '04)*, pp. 1–17, Vienna, Austria, April 2004.
- [9] M. Marin-Perianu, N. Meratnia, P. Havinga et al., "Decentralized enterprise systems: a multiplatform wireless sensor network approach," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 57–65, 2007.
- [10] E. Miluzzo, N. Lane, K. Fodor et al., "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, 2008.

- [11] D. L. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 6–23, 1997.
- [12] J. Llinas, "Revisiting the jdl data fusion model ii," Tech. Rep., DTIC Document, 2004.
- [13] P. Barralon, N. Noury, and N. Vuillerme, "Classification of daily physical activities from a single kinematic sensor," in *Proceedings of the 27th Annual International Conference of the Engineering in Medicine and Biology Society (IEEE-EMBS '05)*, pp. 2447–2450, September 2005.
- [14] J. Garcia, J. Molina, A. Berlanga, and G. de Miguel, "Data fusion alternatives for the integration of millimetre radar in airport surveillance systems," in *Proceedings of IEEE International Radar Conference*, pp. 796–801, 2005.
- [15] M. Markin, C. Harris, M. Bernhardt et al., *Technology foresight on data fusion and data processing*, Publication of The Royal Aeronautical Society, 1997.
- [16] 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, 2010.
- [17] J. Chon and H. Cha, "LifeMap: a smartphone-based context provider for location-based services," *IEEE Pervasive Computing*, vol. 10, no. 2, pp. 58–67, 2011.
- [18] B. N. Schilit and M. M. Theimer, "Disseminating active map information to mobile hosts," *IEEE Network*, vol. 8, no. 5, pp. 22–32, 1994.
- [19] A. Schmidt, K. Aidoo, A. Takaluoma, U. Tuomela, K. Van Laerhoven, and W. Van de Velde, "Advanced interaction in context," in *HandHeld and Ubiquitous Computing*, pp. 89–101, Springer, 1999.
- [20] A. Dey and G. Abowd, "Towards a better understanding of context and context-awareness," in *Proceedings of the Workshop on the What, Who, Where, When, and How of Context-Awareness (CHI '00)*, vol. 4, pp. 1–6, 2000.
- [21] A. Schmidt, M. Beigl, and H. W. Gellersen, "There is more to context than location," *Computers and Graphics*, vol. 23, no. 6, pp. 893–901, 1999.
- [22] W. W. W. C. Editors, "Geolocation API specification @ONLINE," February 2010.
- [23] R. DeVaul and S. Dunn, "Real-time motion classification for wearable computing applications," 2001.
- [24] F. van Diggelen, *A-GPS: Assisted GPS, GNSS, and SBAS*, Artech House, 2009.
- [25] D. M. Boyd and N. B. Ellison, "Social network sites: definition, history, and scholarship," *Journal of Computer-Mediated Communication*, vol. 13, no. 1, pp. 210–230, 2007.
- [26] M. Henriksen, H. Lund, R. Moe-Nilssen, H. Bliddal, and B. Danneskiold-Samsøe, "Test-retest reliability of trunk accelerometric gait analysis," *Gait and Posture*, vol. 19, no. 3, pp. 288–297, 2004.
- [27] M. N. Ko, G. P. Cheek, M. Shehab, and R. Sandhu, "Social-networks connect services," *Computer*, vol. 43, no. 8, pp. 37–43, 2010.
- [28] D. Recordon and D. Reed, "OpenID 2.0: a platform for user-centric identity management," in *Proceedings of the 2nd ACM Workshop on Digital Identity Management (DIM '06)*, pp. 11–16, November 2006.
- [29] P. Barralon, N. Vuillerme, and N. Noury, "Walk detection with a kinematic sensor: Frequency and wavelet comparison," in *Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '06)*, pp. 1711–1714, August 2006.
- [30] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman, "Activity recognition from accelerometer data," in *Proceedings of the 20th National Conference on Artificial Intelligence*, vol. 3, pp. 1541–1546, July 2005.
- [31] J. Y. Yang, J. S. Wang, and Y. P. Chen, "Using acceleration measurements for activity recognition: an effective learning algorithm for constructing neural classifiers," *Pattern Recognition Letters*, vol. 29, no. 16, pp. 2213–2220, 2008.
- [32] W. W. W. C. Editors, "Web services description language 1.1 @ONLINE," Mar 2001.



# The Scientific World Journal

Hindawi Publishing Corporation  
<http://www.hindawi.com>

Volume 2013



Hindawi

- ▶ Impact Factor **1.730**
- ▶ **28 Days** Fast Track Peer Review
- ▶ All Subject Areas of Science
- ▶ Submit at <http://www.tswj.com>