Building a Personal Symbolic Space Model from GSM CellID Positioning Data

Filipe Meneses and Adriano Moreira

Department of Information Systems
University of Minho, Portugal
{meneses, adriano}@dsi.uminho.pt

Abstract. The context in which a person uses a mobile context-aware application can be described by many dimensions, including the, most popular, location and position. Some of the data used to describe these dimensions can be acquired directly from sensors or computed by reasoning algorithms. In this paper we propose to contextualize the mobile user of context-aware applications by describing his/her location in a symbolic space model as an alternative to the use of a position represented by a pair of coordinates in a geometric absolute referential. By exploiting the ubiquity of GSM networks, we describe a method to progressively create this symbolic and personal space model, and propose an approach to compute the level of familiarity a person has with each of the identified places. The validity of the developed model is evaluated by comparing the identified places and the computed values for the familiarity index with a ground truth represented by GPS data and the detailed agenda of a few persons.

Keywords: location; GSM; positioning; inference; space model.

1 From Position to Location, to Space Models, to Context

The position of a mobile user, described by a pair of coordinates in a geometric absolute referential, is one of the most used dimensions of context in location- and context-aware applications. Real time acquisition of the user's position can be done using a number of different technologies. The Global Positioning System (GPS) [1] is, probably, the most popular and used of these technologies. Although GPS positioning is free and has worldwide coverage it has the disadvantage of working only in open areas, demanding a line-of-sight between a few GPS satellites and the receiver. There are, however, other well known solutions for positioning that complement GPS and that try to overcome some of its limitations. Systems like the Active Bat [2] or the Ubisense [3], that provide geometric position indoors, can be used to complement some of these limitations or to support applications in different scenarios. WiFi networks are also becoming available in an increasing number of places and can also be used for positioning a mobile user [4].

Geometric positioning, by it-self, described by a point in a 2D or 3D geometric referential (such as a pair of coordinates in the WGS-84 datum) are of little use for the

C. Giannelli (Ed.): Mobilware, LNICST 0007, pp. 309–324, 2009.

majority of applications. More important is the location of the mobile user, often represented as a symbolic descriptor in a symbolic referential. Common examples include the name of streets, as used in car navigation systems, or the ZIP code used in many location-based services that provide nearby restaurants, monuments, etc. We therefore distinguish position from location, where position is mostly a geometric description in a relative (e.g. Active Bat) or absolute (e.g. WGS-84) referential, and where location is mostly the human-readable name of a place.

Going from position to location requires a transformation operation. In GPS based systems, this transformation is mostly based on maps. In WiFi based positioning, this transformation can rely on local databases or on network services, but are dependent on the availability of universal geo-referenced databases (e.g. the Herecast service that transforms WiFi radio signatures into locations [5]). The Active Badge system [6] provides a similar transformation by converting infrared beacons into the identification of a room inside a building.

The context of a person, even considering position and location, is however more than a simple pair of coordinates or the name of a place – with whom a person is, the type of place, the current state of that place (e.g. crowded or not), or how often a person visits that place are equally important characteristics of his/her context. Thus, the user's context is much more than the context of the device from where the position or location is acquired.

Cellular mobile networks, such as GSM, also provide the functionalities from where the user's position can be retrieved, with the advantage that the mobile phones are well integrated in everyday life and are turned on most of the time. Given the additional fact that GSM networks are ubiquitous, a mobile phone can be exploited as a good proxy to capture the context of a mobile user. The positioning data can be even more valuable for location- and context-aware applications if high level contextual information about the users could be inferred from it.

In this article we describe a solution that enables the semantic enrichment of the user's context based on:

a) the ubiquity of the GSM infrastructure;

b) the scope of a device (the mobile phone) that, given its typical use, has a great potential to better model the user's context;

c) a personalized space model built from the self spatial-temporal behaviour of the user, and;

d) the use of inference techniques to estimate high-level parameters used in the user's context modelling (the familiarity level with the current place).

In sections 2 and 3 we present a mathematical model that exploits the GSM cellID information to identify the places visited by the users and to compute a familiarity index for each of those places. The collection of identified places, together with the corresponding familiarity indexes, constitutes a personal symbolic space model that can be used by context-aware applications. Among others, this model may be used by applications in the security area (e.g. by taking special actions when a person is visiting a place for the first time) or in recommendation systems (e.g. by avoiding to disturb a user with local data when he/she is well aware of the neighbourhood).

The process used to build the proposed model relies only on the GSM cellID of the cells visited by the phone, dispenses any previous knowledge of the network topology and does not require any human intervention.

Section 4 discusses the obtained results and the validity of the proposed model by comparing the inferred information with a ground truth represented by GPS traces and detailed diaries collected by a few persons while in their daily activities. Section 5 presents the related work while the last section presents the conclusions.

The work here described complements previous preliminary results that are described in [7], mostly by providing an extended validation of the created models.

2 Positioning Model

A GSM network is a telecommunications cellular network, whose radio infrastructure is built around a set of base stations usually installed on high places like the top of buildings or in towers spread over the terrain in order to cover a certain geographic area.

A GSM network is made of cells which have different shapes and sizes. The coverage area of a cell is defined by the cell configuration and by the morphology of that area, which may create cells with several square kilometres of size or with just a few thousand of square meters. Each cell can be configured to cover a wide area, a tail-shaped area or just a small area for example in a city centre.

In GSM networks the cells ensure the radio communications between the mobile terminal and the operator network core. A handset is linked to a cell, and by exchanging data with the cell base station it is able to receive and place phone calls and to transmit data. Because each cell supports only a limited number of channels, in some places, it is necessary to install more than one cell. By overlapping several cells it is possible to increase the network capacity inside a certain area.

CellID positioning provides the handset location in a symbolic referential. A map with the position of the base stations easily allows the conversion of the cellID into the geographic position of the handset. However, network operators do not make their network configuration and base stations' positions available to the public. Mapping each cell manually could be a solution but it would require a big effort and the new cells or changes in the network topology would have to be tracked in order to keep the service data updated.

On the other hand, in fact, people on their living, deal with location in a symbolic referential. "Home", "office", "friends' house" is the way people express their location and not by "I'm at 41º24'N, 8º31'W". Computers, otherwise, do not have knowledge about the link existent between the locations and deal better with absolute locations expressed by coordinates. If John is in someone's house then we know he is near the supermarket because we are aware of both places locations. However, computers do not have the notion of being "near" or "far". They need to know the exact location to compute the distance between two places and have a definition of "near" as something that is at a distance less than a certain value. "Near" is also something that varies from people to people and is dependent on the context: it can be near if someone is travelling by car but far on foot.

The positioning model adopted in this work is based on the detection of movement within the symbolic space model of the GSM cells. By detecting motion, a personal symbolic space model can be progressively created with the places where a user is seen to stay for a certain amount of time. This section describes how the movement of terminals is detected and how the proposed approach was validated using GPS positioning data.

2.1 Movement Tracking

When turned on a GSM handset is linked to a cell – the active cell – which is selected among the set of cells available in that place. In order to support the growing number of GSM users, the operators keep installing more cells and in almost all populated areas it is possible to find more than one cell covering the same area. The handset changes from one cell to another - changes the active cell - when fluctuation occurs in the radio signal level, due to fading or due to movement of the handset. Thus, the movement of a handset cannot be assumed from the change in the cellID.

When a terminal is stopped in a certain place, the temporal sequence of active cells is limited to the set of cells that cover the terminal's position. For different places the frequency with which a cell appears in the temporal sequence of active cells is different (cell fingerprint). When the user moves, we observe a bigger variation (faster and within a bigger set of cells) of the active cell in each moment.

We created the Mobility Distance and Mobility Index metrics that allow us to infer the user motion, analysing the changes in the serving cell and the amount of time spent on each cell. For these, a Cellular Positioning Record (CPR) is defined as the identification of the GSM cell being used and the time spent in the cell, represented by {cellID, LAC, MNC, MCC, stayTime}.

Each operator, in each country is identified by a Mobile Network Code (MNC) and a Mobile Country Code (MCC) assigned centrally by the ITU - International Telecommunication Union. For managing proposes, each operator can divide its network into small geographic areas, identifying each one by a Location Area Code (LAC). A LAC is made of several cells and each cell is identifiable by its cellID.

Mobility Distance represents the distance between two CPRs. If two records represent the same cell then the user has not moved or moved just inside the cell area and the Mobility Distance is zero. If the records represent two different cells then the Mobility Distance is the sum of the inverse of the time spent in each cell (bigger time intervals represent smaller distances). Equation 1 represents the Mobility Distance, where *time(r)* is the time spent in cell *r*.

$$MobilityDistance(r_1, r_2) = \begin{cases} 0 & if \quad r_1 = r_2 \\ \dfrac{1}{time(r_1)} + \dfrac{1}{time(r_2)} & if \quad r_1 \neq r_2 \end{cases} \tag{1}$$

When the user is moving fast the time spent in a place is small. Mobility Index (equation 2) considers that speed is proportionally inverse to the time spent in a place and therefore is an estimate of the level of mobility of a user. Given a list of records, Mobility Index is the sum of the Mobility Distance between each record and all the previous ones.

$$MobilityIndex(r_{1..n}) = \sum_{i=1}^{n}\sum_{j=1}^{i} MobilityDistance(\mathrm{r_i}, \mathrm{r_j}) \tag{2}$$

The user mobility level can be estimated by calculating the Mobility Index over a pre-defined period of time (*timeMin*). The Mobility Index is calculated over the set of records collected from the current time instant back to *timeMin* seconds ago. For a set of consecutive records it is possible to create a sliding window and calculate the Mobility Index as the time goes by.

The Mobility Index varies according to the size of the sliding window (*timeMin* value), being higher when calculated for larger values of the *timeMin* parameter. In [7] we show that smaller sliding windows allow to detect fast the beginning and end of user movements. In order to detect the user movements, we define a threshold based on the sliding window size. The user is considered in motion if the Mobility Index is above the pre-defined threshold.

2.2 Movement Tracking Validation

The validation of how well the Mobility Index models the mobility of a user was performed by comparing this metric with a similar metric obtained from GPS real data.

During a week, one user collected GPS positions simultaneously with the GSM cellID data (CPRs). From the GPS data the user was considered in motion when the calculated velocity was higher than 3km/h. From the GSM data the user motion was obtained from the periods of time when the Mobility Index was higher than a predefined threshold. We then calculated the correlation between the motion periods of time from the GPS data and from the GSM data as the percentage of time they were coincident.

Using the correlation value we estimated the optimum values for the window size (*timeMin*) and the threshold value used in the Mobility Index. Figure 1 shows the cumulative correlation for a sliding window of 10 minutes and a threshold of 6.

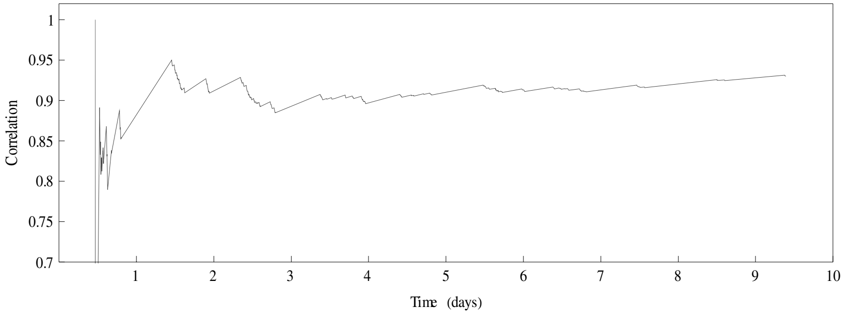

Fig. 1. Correlation between the mobility periods calculated from GPS data and from the Mobility Index

The correlation increases or decreases when GPS and GSM movement detection do match or do not match. The results present in figure 1 show a good correlation between the moving periods calculated from the GPS data and from the Mobility Index. Good correlation was achieved for a threshold value between 0.3 and 9. However, lower values reduce slightly the correlation but increase the number of situations where the movements of the user were correctly detected.

3 Personal Symbolic Space Model

3.1 Place Definition

The Mobility Index allows us to detect when the user starts and stops moving. At the moment the user starts moving, we characterize the previously visited place by creating a fingerprint with the list of cells observed during the time the user was not moving.

A fingerprint is the list of cells observed during the stay in a place and the time percentage stayed in each one, the total time spent on that place and a timestamp. A fingerprint is represented by:

$$FP=\{\{\{cellID_1, timePercentage_1\},\{cellID_2, timePercentage_2\},\ldots,\{cellID_k, timePercentage_k\}\}, totaltime, timeStamp\}\ .$$

Every time a user visits a place a new fingerprint is created, using the data collected during the time the user spent on that place. However, a user can visit the same place many times, leading to different fingerprints for the same place. It is therefore necessary to identify which fingerprints represent the same place.

The user's mobile phone uses one of the available cells and changes, in an unpredicted way, among the available cells. Thus, several visits to the same place result in fingerprints where the time percentage associated to each cell is different.

Although fingerprints for the same place may be different, they have some similarity. They are created with the cells used during the visit to a place and, thereby, are composed by the same cells or by a subset of the available cells. Clustering similar fingerprints allows us to create a cluster for each place, composed of the similar fingerprints. A cluster has the same structure as a fingerprint but it is created by the union of similar fingerprints instead of being created from the raw data acquired by the phone.

To cluster similar fingerprints is necessary to compute the similarity between fingerprints. To measure the distance between two different fingerprints two functions are used: an adaptation of the Hamming Distance and a Similarity Distance function.

Based on an adaptation of the Hamming Distance we created a distance function (HDistance in equation 3) that measures the similarity between two fingerprints. For a cell present in both fingerprints we calculate the absolute difference between the percentages of time spent in that cell in each fingerprint. If the user spends the same amount of time in both fingerprints then there is no difference and the calculated value is zero. For a cell present in just one of the fingerprints, it is considered zero percent in the other one. In the adapted Hamming Distance function the similarity between two fingerprints is half of the sum of distance between each cell present in both fingerprints. For two completely different fingerprints the calculated value is one and for two perfectly equal fingerprints the calculated distance is zero.

$$
\begin{aligned}
FP^{A} &= \left\{\left\{\left\{c_1^{A}, p_1^{A}\right\}, \left\{c_2^{A}, p_2^{A}\right\}, \ldots, \left\{c_n^{A}, p_n^{A}\right\}\right\}, tt^{A}, ts^{A}\right\} \\
FP^{B} &= \left\{\left\{\left\{c_1^{B}, p_1^{B}\right\}, \left\{c_2^{B}, p_2^{B}\right\}, \ldots, \left\{c_m^{B}, p_{\;m}^{B}\right\}\right\}, tt^{B}, ts^{B}\right\} \\
CL &= \left\{c_1, c_c, \ldots, c_k\right\} = \left\{c_1^{A}, c_2^{A}, \ldots, c_n^{A}\right\} \cup \left\{c_1^{B}, c_2^{B}, \ldots, c_m^{B}\right\}
\end{aligned}
\tag{3}
$$

$$
HDistance\left(FP^{\mathrm{A}}, FP^{\mathrm{B}}\right) = \frac{\sum_{i=1}^{k}\left|p_i^{A} - p_i^{B}\right|}{2}
$$

Equation 3 shows the adapted Hamming Distance, where c_i^x is the i$^{\text{th}}$ cell of a fingerprint (FP$^{\mathrm{X}}$), tt^{x} is the total time spent on the fingerprint and ts^{x} is a timestamp. For each cell (c_i^x) there is a percentage (p_i^x) of the total time of the fingerprint that was spent on that cell. Joining all the distinct cells present in both fingerprints creates list of cells (*CL*) with a length of *k* elements.

Experiences with real data shows that two different visits of the same user to the same place may result in two different fingerprints for which the Hamming Distance is high. This is the result of a user's mobile phone in one visit being most of the time in a certain cell while in another visit it uses mainly other cell. Thus, the use of the same set of cells, regarding the percentage of time spent in each one, can also be used as an indicator of visiting the same place. The Similarity Distance function measures the distance between a fingerprint and a cluster by calculating the percentage of cells of the fingerprint, with a percentage of time equal or superior of 1%, that are present in a cluster (or other fingerprint). The Similarity Distance returns a value between 0 (if all cells are in the cluster) and a maximum of 1 (if all cells are not present).

The distance between two fingerprints (*FPDistance*) is computed considering the Hamming Distance and the Similarity Distance (equation 4).

$$
FPDistance\left(FP^{\mathrm{A}}, FP^{\mathrm{B}}\right) = (0.5 \times HDistance(FP^{A}, FP^{B})) + (0.5 \times SimilarityDistance(FP^{A}, FP^{B}))
\tag{4}
$$

Joining two fingerprints creates a cluster in which the total time spent on the cluster is the sum of the time spent on both fingerprints, and the timestamp is the oldest of the timestamps. The cells of the cluster is a list of all the cells present in both fingerprints having a percentage of time calculated proportionally between the time spent in each fingerprint and the total time spent on both fingerprints. A cluster has the same structure as a fingerprint and, therefore, subsequent fingerprints can easily be joined to a cluster.

A new cluster algorithm has been developed to cluster fingerprints, because data to be clustered is symbolic (the fingerprints) and because the clusters are to be discovered/created in real time. Many clustering algorithms demand that they have all the data available to start doing the clustering process, which invalidates the use in this system because data must be clustered while it is being collected by the mobile phone, and without previous knowledge about the total number of records that will be collected. Moreover, the total number of clusters to be created is not known in advance which invalidates also the use of many existent clustering algorithms like the k-means. The number of clusters is the number of places visited by the user which cannot be pre-determined, varying from one person to another and growing as time goes by and as the user moves and visits new areas.

The first fingerprint represents the first place visited by the user and leads to the first cluster. After that, for each new fingerprint the similarity between the fingerprint and every existent cluster is calculated. Because a cluster has the same structure of a fingerprint, the Fingerprint Distance (*FPDistance* in equation 4) is used to calculate the similarity/distance between a fingerprint and a cluster.

If the similarity (*FPDistance*) between the fingerprint and the most similar cluster is smaller than a pre-defined threshold then the fingerprint is joined to the cluster. An algorithm parameter defines the minimum similarity between a cluster and a fingerprint in order to join them.

Another algorithm parameter defines the maximum number of clusters (*kMaxNumberClusters*) that can be created[1]. If a fingerprint cannot be joined to an existent cluster (it is not similar enough to any existent cluster) and the maximum number of clusters has not yet been reached then a new cluster is created. After the maximum number of clusters has been reached the fingerprint can be used to create a new cluster that will replace an existent one, or discarded. To determine if a cluster should be removed and replaced by a new one it was created a Fingerprint to Cluster Importance Ratio (FCIR in equation 5) that measures the relative importance of a fingerprint in relation to a cluster. If the FCIR is smaller than one for all the clusters then the fingerprint is discarded. If the importance ratio is higher than one for one or more clusters then the cluster with the highest ratio is replaced by the fingerprint.

$$\mathrm{FCIR}(cl, fp) = \frac{KldgIdx(tt(fp))}{FgIdx(Age(cl, fp)) \times KldgIdx(tt(cl))} \tag{5}$$

This process allows old and spurious clusters to be replaced by new and more relevant clusters.

Clusters recently changed correspond to places recently visited by the user, and clusters where the user spent more time are also more valuable than places not visited for a long time or visited for a short period of time. The FCIR uses the Forget Index (*FgIdx)* that is calculated using the time elapse since the last fingerprint was added to the cluster (relative age) and the Knowledge Index (*KldgIdx*) that is calculated using the total time spent on a cluster (these functions are detailed in [7]).

To measure how much a person forgets about a place after a certain amount of time without going there is something that cannot be done easily. There is no mathematical equation that can be applied, universally, to everybody. People's memory is something that is very personal, varying as a result of many factors. Besides, as time goes by, places also change, with the construction of new buildings, new roads, etc. and some places change faster than others.

The knowledge of a place cannot also be measured by a simple equation that is universally applied to everybody. Some persons tend to know a place better and faster than others. Besides, the knowledge about a place can be influenced by a number of factors like the mean of transportation used, the purpose of the visit or the time of the day.

[1] This parameter exists only to limit the amount of memory to be used in the mobile device.

3.2 Familiarity Index

An important place is usually visited more often by a user and the user spends more time there (like home or his office). However, places not visited for a long period of time can still be important in some circumstances. For example, it can be important to know that the user is visiting a place which has not been visited for a long time. Because a cluster is a small data structure then it is possible to keep a high number of clusters in a small device with limited storage capacity.

The familiarity level with a certain place varies according to the total time spent in that place and with the time elapsed since the user visited a place for the last time. Equation 6 shows the function created to model the familiarity level that a user has with a cluster.

$$FmIdx(cl) = KldgIdx(cl) \times FgIdx(cl) \tag{6}$$

4 Results

In this section we show the results achieved by the presented algorithm. We show how data was collected and we overlap the achieved results with the ground truth to check the quality of the proposed solution.

4.1 Data Collection

An application was developed to collect the GSM cellID data. It runs on a Symbian mobile phone and creates a log file with the timestamp and the {cellID, LAC, MNC, MCC} data. It checks the cellID every eight seconds and records it on the log file whenever it changes. Considering that a phone can stay for several hours in the same cell, the application creates a record on the file every fifteen minutes even if the cell does not change. This way it is possible to distinguish between when the mobile phone is linked for a long period of time to the same cell, from the fact that application is not being executed (no data is being collected).

Three different users collected data during several consecutive weeks and, simultaneously, manually recorded their movements. User A lives in the centre of a big city and works in a smaller village, located 35 km away, travelling by car between both cities. During data collection time most of his movements were made inside those two places and travelling between them. User B lives in an average size city and works in the University campus located in that same city. His movements are mainly inside this city, including visits to supermarkets, to the children's schools, to relatives' houses, etc. User C works in the same University campus but lives in the countryside, in a rural area, 17 kilometres away from the University. Data was collected for a period of several consecutive weeks and contains data for highly populated areas where there are numerous GSM cells and also data collected in rural areas, where the average size of a cell is larger and the number of available cells in each place is reduced.

To manually record their movements, each user used a diary to register at what time he/she arrived at a place and what time he/she left that place. This manually recorded movement data (user diary) acts as ground truth and allows assessement of the quality of the results achieved, allowing the comparison of the results achieved by the proposed algorithms with the reality.

Although data was collected into a log file and later processed according to the algorithms described in the previous section, we must emphasize that all records were processed in the order they were created. Thus, the achieved results are exactly the same as would be achieved if the records were processed in real time.

4.2 Clustering Process Parameters

The clustering process, described in section 4.1, is dependent on two variables: the maximum number clusters that can be created and the similarity threshold that must be achieved by a fingerprint to be joined to an existent cluster.

Defining the maximum number of clusters as a high number does not cause any constraints to the system, neither does it influences the performance or the quality of the results. A cluster is a very simple data structure that occupies, on average, less than 350 bytes (the exact size occupied by each cluster depends on the number of cells and the number of visits to the place). We defined the limit to 100 clusters but this limit was not reached by any of our tests users.

If the similarity threshold is defined to a very high number then only very similar fingerprints are joined to existing clusters resulting in different clusters for the same place. Good results were achieved joined fingerprints that have up to 65% of similarity. This threshold level makes the system create different clusters for different places (avoiding two places ending up in the same cluster) but, unfortunately, in some restricted circumstances, it creates more than one cluster for the same place (see results described in next section).

4.3 Trial Users' Results

Table 1 summarizes the results obtained after processing the data with the algorithms presented in the previous sections.

Although results achieved by the trial users show that the proposed algorithms detect most of the places, a more detailed analysis of the achieved results can explain many of the errors.

User A has been in only four different places: his home, the workplace, a village 3km away from his workplace and into a friend's house. The two visits made to the friend's house took between 5 and 10 minutes and were not detected by the system because they were too short. Some of the 6 visits made to the village located near his

Table 1. Results achieved after processing all records collected during several weeks

	User A	User B	User C
Places visited by the user	4	27	29
Places detected	3 (16 clusters)	19 (21 clusters)	22 (30 clusters)
Places not detected	1	8	7
False positives	1 cluster	4 clusters	6 clusters

workplace were not detected (also short duration) or were detected but the fingerprints were joined to clusters that represent the workplace.

User A lives in the very centre of a big city and while at home the system detected 19 different cells. Such a high number of cells and the long periods of time spent at home made the system create a total of 11 different clusters just for one place. However, 7 of those clusters were made with only one or two fingerprints and each one has less than an hour of total time. These clusters represent less than 1.2% of the total time of 442 hours spent by the user at home.

User A travels 35 km between his house and his workplace every day by car, using the same road that crosses a rural area. For the user A the most common error was the creation of fingerprints that do not correspond to any place visited by the user (false positives). However, analysing the clustering process results shows that false positive errors occur always when the user is travelling between his house and the workplace. The trip is made through a rural area, lasts 40 minutes and crosses two valleys. The relatively slow speed through a low populated area causes the user to be inside the same reduced set of GSM cells for several minutes. Thus, the mobility index decreases below the threshold level creating a fingerprint. The clustering process grouped all these fingerprints in the same cluster, which would represent the "driving through place". Indeed, travelling everyday along the same route makes the user to know the road very well and thus be familiar with the surrounding area. User B and C false positives also result from travelling in relatively low speed (traffic jam, narrow and twisted roads inside a natural park, etc).

Trial users B and C spend most of the days at the university campus, which is located in a city with thousands of students. To support the communications for a big number of students and population around the campus, the mobile network operators have installed a considerable number of cells. Thus, the set of cells available in nearby places are not the same, but not different enough to distinguish those places.

Trial user B has family members that live 500 meters away from the University and went to a restaurant located 350 meters away from the campus. Those very near geographic places not always were distinguished from the campus.

Similarly, user C errors result mostly from a one week trip to a foreign country to participate on a conference and from visits made to a friend's house located 700 meters away from the University. While away, the conference took place in a hotel located 500 meters away from the one where the user was hosted. So short distance, made the system misclassify some of the visits to those places.

Some places were visited only once while others were visited a number of times. Examples of places often visited are the users' home and workplace. Table 2 summarizes the results achieved showing the number of places that had all the visits correctly detected, the number of places that had some of the visits detected and number of visits made to places that did not have any visit detected by the system.

The system is capable of detecting more than 75% of the places and 50% or more of all places visited by the user were always correctly detected. Results show that 85% of the visits made to all the places were detected by the system. Short duration visits and places geographically very near are the main flaw of the proposed solution.

Table 2. Results achieved considering the visits made to each place

User A	2 places	53 visits	53 detected visits	100%
	1 place	6 visits	3 detected visits	50%
	1 place	3 visits	0 detected visits	0%
User B	18 places	58 visits	58 detected visits	100%
	3 places	71 visits	62 detected visits	87%
	6 places	9 visits	0 detected visits	0%
User C	12 places	18 visits	18 detected visits	100%
	10 places	133 visits	116 detected visits	87%
	7 places	11 visits	0 detected visits	0%

Figure 2 shows the user agenda overlaid with the results of the movement tracking process and recognizer results. The gray areas represent the time spent on a place, according to the user B agenda. The solid line (over the gray area) show the user movement tracking process: it is at the high level when the user is classified as being visiting a place and is at the low level when it classifies the user as moving.

The lines under each row represent the results of the recognizing process: wider lines represent the moments the system correctly identifies the user location and the thinner line represents the moments when the system identified a cluster that does not represent the user's current location.

The figure shows that tracking and recognizer processes works well, compared with the ground truth provided by the user agenda.

Visits to a place cannot be detected if their duration is not long enough to allow the Mobility Index to decrease below the threshold. When the user arrives at a place the sliding window contain cells used by the user's phone before he arrives at that place.

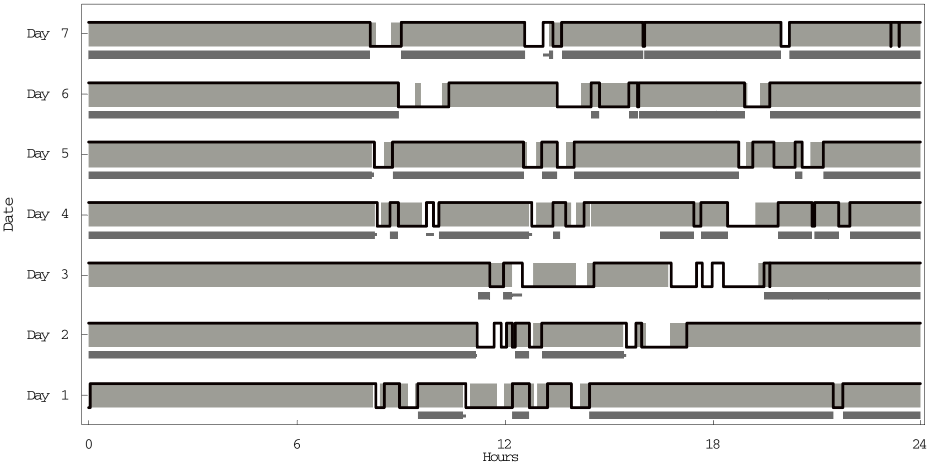

Fig. 2. User agenda overlapped with the tracking and recognizer processed results

Only after being in a place for several minutes the system will start to use the same limited set of cells and the Mobility Index will eventually decrease below the threshold level. Thus, the movement tracking process usually detects the user as being visiting a place only a few minutes after the timestamp recorded in the user agenda.

A place is only recognized by the system after finishing the first visit to a place (the cluster is created after the visit). The recognizer is never capable of identifying a place when the user visits a place for the first time (any of the existent clusters represent that place) or it incorrectly identifies a place that is usually not far from the real place (sometimes places geographically near have similar clusters because of the use of the same subset of cells).

4.4 Using the Personal Symbolic Space Model (PSSM)

Knowing if a user is visiting a place or moving between places can be valuable information to some applications. However, we also compute a familiarity index to each place which is based on the time spent in a place and on the amount of time elapsed since the last visit.

The recognizer is the process that identifies the user location, identifying the cluster in which the user is located. It searches the current fingerprint within the clusters and identifies the user location by selecting the cluster nearest to the current fingerprint. It uses the same measuring process that is used in the clustering process, including the maximum distance that can be measured between the fingerprint and a cluster. When the fingerprint is far from all the clusters, the user is in an unidentified place.

As more time is spent in a place, bigger are the chances to correctly identify the cluster that represents that place because as time goes by the fingerprint changes and represents the place better. When a user arrives at a place the fingerprint is made with the first set of cells. As the user spends more time in a place, the fingerprint will contain more data collected in that place.

Although we clustered the fingerprints based on their proximity, in some cases the system creates more than one cluster for one place. It happens in places where a very large number of cells are available. Trial user A lives in the centre of a big city and for his home we detected at least 8 cells. For user B and C, clients of two different network operators, we detected 6 and 8 different cells respectively just for the university campus area, where they spend most of their days.

If the system creates two clusters for a place it will influence the familiarity index which is calculated for each cluster individually. In the cases where the user spends 50% of the time in each cluster the familiarity index is always half of the real number. In cases where the user spends most of the time in one of the clusters the familiarity index is near the expected value for the cluster where the user spends more time (80% of time in a cluster means 80% of the real value) and far from the real value for the cluster that is «visited» rarely.

Figure 3a and 3b show the Familiarity Index computed for trial user B, starting from the beginning of the learning process. Figure 3a presents the Familiarity Index with the current place, computed as the position data was being collected and processed. The graph value varied between zero (when the user is in motion or visiting a place for the first time) and the familiarity index calculated for the visited

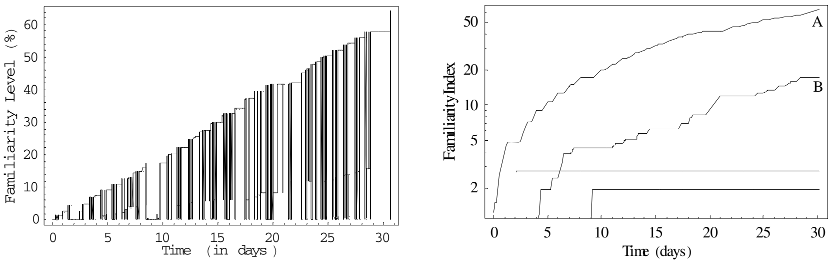

Fig. 3. Familiarity index with the places visited and familiarity index calculated for the different clusters

place/cluster. The general trend of the Familiarity Index is to grow because as time goes by the total amount of time spent by the user in some places is also growing and thus the familiarity index for those places also grows.

Figure 3b shows the Familiarity Index calculate for each cluster. The most familiar place (line A) is a cluster that represents the user's home. The second most familiar place (line b) is the workplace which is the second place where the user spent more time.

5 Related Work

Hopefully, one day, location systems will be ubiquitous, accurate and have availability to be used everywhere by everybody. Meanwhile, many authors seek for the best possible solution with the existent technology.

Mobile telephone positioning technologies have evolved a lot in the last years. The wireless E911 program [8] was one of the biggest motivations for the deployment of new and accurate positioning technologies, demanding that network operators must provide the location of a handset in case of emergency with a precision of 300 meters in most cases. This E911 initiative promoted the development of accurate location services but unfortunately the operators do not make it available to allow creation of location-based services and applications.

A number of research projects used GSM networks to acquire the user's position. BeaconPrint [9] uses WiFi and GSM radio fingerprints, collected at someone's mobile device, to automatically learn the places they go and detect when they return to those places. BeaconPrint is up to 90% accurate in learning and recognizing places. Although it achieves good results it is a multi-sensor data approach, which makes it difficult to apply to real users.

Place Lab [10] is a software approach providing low-cost, easy-to-use positioning for location-enhanced computing applications. It uses radio beacons sensed from WiFi access points, GSM networks and fixed Bluetooth devices to estimate the device's position. It uses also a GPS receiver whenever it is available. When the GPS receiver is not available, the radio sensed beacons are converted into geographic points using a database that maps each beacon ID to a geographic point. The Place Lab approach is dependent on the existence of a database or a GPS receiver to build the user own database and provides the user location as a geographic point. Place Lab goal is to bootstrap the dissemination of location-based services and applications by creating a platform that would use whatever technology the user may have. By placing the algorithms proposed in this paper over Place Lab platform it is possible to enhance the user's context description, by adding the familiarity level for the current location for the user.

In [11] the goal is to detect the places that people visit in their everyday lives using only client-based GSM phones. The achieved results show that it is possible to locate users inside a building, with considerable precision, just using client-based GSM position. However, results were achieved using the radio signal level and also the list of nearby cells and this kind of data is not available in many handsets, reducing the potential number of users. Similarly, results achieved by [12], [13] and [14] were only possible by computing data that include radio signal level from the current cell and from other nearby cells (used by the GSM standard). The solution described in this paper may not reach the same accuracy but has the advantage of using only cellID that can be read from a wide number of handsets, creating a more universal solution, and with enough accuracy to trace most of the places visited by a user.

In [15], Laasonen presents a framework for recognizing personal locations in cellular networks, using cell-based location data. Although this approach was able to identify visited places it is not a pervasive system, demanding the user to give a name to each place and it is not able to distinguish between more important places and those that were visited occasionally. The ContextPhone [16], that relies on Lassonen work [15], show that mobile phones are well suited for context-aware computing.

6 Conclusions

Be able to acquire the user location everywhere, without using a specific equipment or relying on specific sensors network is fundamental to bootstrap the dissemination of location-based services and applications. The proposed solution, based on the cellID of the GSM network, has the advantage of being built over a widely disseminated device and not being dependent on any network service. It is a generic solution that allows the user to build a personal symbolic referential and use it to provide high-level context information to context-aware applications.

The proposed solution can run on limited devices, like mobile phones, demanding a small quantity of memory to store data and very low CPU capabilities to execute the algorithms. The clustering process and the familiarity index produce valuable data that can be used by a number of different kinds of applications in the selection of services and data or to adapt the application behaviour according to the user familiarity level with the surrounding place.

References

1. National Space-Based Positioning, Navigation, and Timing Coordination Office. Global Positioning System (2007), `http://www.gps.gov/`
2. Want, R., et al.: The Active Badge Location System. In: ACM Transactions on Information Systems, pp. 91–102 (1992)

3. Ubisense Ltd. Ubisense - Precise Real-time Location (2007), http://www.ubisense.net/
4. LaMarca, A., et al.: Place Lab: Device Positioning Using Radio Beacons in the Wild. In: Gellersen, H.-W., Want, R., Schmidt, A. (eds.) PERVASIVE 2005. LNCS, vol. 3468, pp. 116–133. Springer, Heidelberg (2005)
5. Herecast. Herecast: WiFi Location-Based Services/802.11 Positioning System (2007), http://www.herecast.com/
6. Harter, A., Hopper, A.: A distributed location system for the active office. In: IEEE Network, pp. 62–70 (1994)
7. Meneses, F., Moreira, A.: Using GSM CellID positioning for place discovering. In: Locare 2006: First Workshop on Location Based Services for Health Care in the International Conference on Pervasive Computing Technologies for Healthcare, Innsbruck, Austria (2006)
8. FCC - Federal Communications Commission. Enhanced 911 - Wireless Services (2006), http://www.fcc.gov/911/enhanced/
9. Hightower, J., et al.: Learning and Recognizing the Places We Go. In: Beigl, M., Intille, S.S., Rekimoto, J., Tokuda, H. (eds.) UbiComp 2005. LNCS, vol. 3660, pp. 159–176. Springer, Heidelberg (2005)
10. Place Lab. Place Lab web site (2006), http://www.placelab.org
11. Varshavsky, A., et al.: Are GSM phones THE solution for localization? In: 7th IEEE Workshop on Mobile Computing Systems and Applications (HotMobile 2006). IEEE Computer Society, Washington (2006)
12. Otsason, V., et al.: Accurate GSM Indoor Localization. In: Beigl, M., Intille, S.S., Rekimoto, J., Tokuda, H. (eds.) UbiComp 2005. LNCS, vol. 3660, pp. 141–158. Springer, Heidelberg (2005)
13. Sohn, T., et al.: Mobility Detection Using Everyday GSM Traces. In: Dourish, P., Friday, A. (eds.) UbiComp 2006. LNCS, vol. 4206, pp. 212–224. Springer, Heidelberg (2006)
14. Chen, M., et al.: Practical Metropolitan-Scale Positioning for GSM Phones. In: Dourish, P., Friday, A. (eds.) UbiComp 2006. LNCS, vol. 4206, pp. 225–242. Springer, Heidelberg (2006)
15. Laasonen, K., Raento, M., Toivonen, H.: Adaptive On-Device Location Recognition. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 287–304. Springer, Heidelberg (2004)
16. Raento, M., et al.: ContextPhone: A Prototyping Platform for Context-Aware Mobile Applications. In: IEEE Pervasive Computing, pp. 51–59 (2005)