

Discovering places of interest in everyday life from smartphone data

Raul Montoliu · Jan Blom · Daniel Gatica-Perez

Published online: 10 January 2012
© Springer Science+Business Media, LLC 2012

Abstract In this paper, a new framework to discover places-of-interest from multi-modal mobile phone data is presented. Mobile phones have been used as sensors to obtain location information from users' real lives. A place-of-interest is defined as a location where the user usually goes and stays for a while. Two levels of clustering are used to obtain places of interest. First, user location points are grouped using a time-based clustering technique which discovers stay points while dealing with missing location data. The second level performs clustering on the stay points to obtain stay regions. A grid-based clustering algorithm has been used for this purpose. To obtain more user location points, a client-server system has been installed on the mobile phones, which is able to obtain location information by integrating GPS, Wifi, GSM and accelerometer sensors, among others. An extensive set of experiments has been performed to show the benefits of using the proposed framework, using data from the real life of a significant number of users over almost a year of natural phone usage.

Keywords Mobile Phones · Multimodal data · Place discovering · Real-life long-term experiments

R. Montoliu (✉)
Institute of New Imaging Technologies (iNIT), Jaume I University, Castellón, Spain
e-mail: montoliu@uji.es

J. Blom
Nokia Research Center, Lausanne, Switzerland
e-mail: jan.blom@nokia.com

D. Gatica-Perez
Idiap Research Institute, Martigny, Switzerland
e-mail: gatica@idiap.ch

D. Gatica-Perez
École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland

1 Introduction

Mobile phones are emerging as an attractive option for large-scale sensing of human behaviors [3, 5, 8–11, 13, 20]. Modern smartphones are equipped with a variety of sensors, including GPS, accelerometers, Wifi and Bluetooth among others. Furthermore, mobile phones have a ubiquitous presence in the lives of most people. For instance, in many European countries, there are more mobile phone accounts than inhabitants and this situation is similar in many other countries [24]. Therefore, the use of data provided from mobile phone sensors, instead of using custommade sensor architectures, is an interesting option to study human behaviors.

The automatic learning of places of interest is one of the key tasks when studying human behavior and for designing new applications for wearable computers including mobile phones. Potential applications could be both single-user (or non-collaborative) and multi-user (or collaborative) [2]. In [18], Marmasse and Schmandt explored the idea of an agent that learns frequented locations. The user may associate a to-do list with each learned location. Then, when the user reaches a learned location, the to-do list could be displayed. Actually, a location-based application to manage a to-do list has recently been implemented as an application called *Reminders* into the *iOS5* operating system [1]. In [2], Ahsbrook and Starner used the learned locations to predict user movements. Then, it could be possible to display in advance a to-do list when the system predicts that the user is about to reach some location.

As discussed in [14], another possible application is to help mobile devices to make decisions on how to behave based on their locations. For example, a cell phone could automatically switch to a silent mode when the user enters a place where the ring is inappropriate and switch to a normal mode when the user leaves this place. Other interesting examples include inferring people's transportation modes [21, 26, 27] and mining individual life patterns [25]. When interest locations have been extracted for multiple people, these locations could be used in some applications, for instance, to recommend locations and activities [28] when the user visits an unknown city, based on data obtained from previous interactions.

In order to clarify some terms used in this paper, the concepts of *Location Point*, *Stay Point*, *Stay Region* and *Place of Interest* are defined as follows:

- **Location point:** A location point is a measurement provided by the sensor about the location of a user (e.g., the GPS coordinates) together with the time where the sensor capture the location information.

A location point is represented using the location coordinates together with the date and time where the sensor captured the location information, e.g., an hypothetical location point p_i could have been collected as follows:

$$p_i = ([46.6N, 6.5E], [2010/07/01], [16 : 34 : 57])$$

- **Stay point:** A stay point is a cluster of location points (from the same day) which represent a geographic region in which the user stayed for a while.

A stay point is represented using the coordinates of the centroid of the cluster, the date, and the time moments where the user arrived and left the stay point, e.g., an hypothetical location point sp_j could have been estimated as follows:

$$sp_j = ([46.6N, 6.5E], [2010/07/01], [16 : 30 : 00], [17 : 54 : 34])$$

- **Stay region:** A stay region is a cluster of stay points (from several days) with the same semantic meaning.
A stay region is represented using the coordinates of the centroid of the cluster and the minimum and maximum coordinates of the stay points belonging to the cluster, e.g., an hypothetical stay region sr_k could have been estimated as follows:

$$sr_k = ([46.6N, 6.5E], [46.595N, 46.610N], [6.492E, 6.505E])$$

Therefore, a stay region can be represented by using a rectangle centered at the centroid of the cluster whose size depends on the minimum and maximum coordinate.

- **Place of interest:** A place of interest is defined as a place where a user usually goes and stays for a while. From this definition, a stay region and a place of interest can be considered as synonymous. These places are “of interest” for the user himself since they are the locations where the user spend the most of the time, which, in general, are the most important places for the user. Depending on the application, discovered places could be of interest, not only for the user himself, but for a community, as, for instance, in the case of an application designed to recommend places to visit/go. In this case, a selection of the places discovered for a set of users, could be “of interest” for another set of users looking for places to go according to some features (as for instance the participants in a social network).

As it has been previously exposed in some previous works [2, 14], for most location systems, including GPS, multiple measurements in the same location do not necessarily yield the exact same coordinates due to errors and variations in the measured phenomena. This is true even if the user stops for a while at precisely the same point every day. Therefore, two estimated stay points could have the same semantic meaning (e.g., “work”, “home”, “museum”, etc.) but not necessarily the same exact coordinates. To deal with this problem, the concept of stay region is used. Once stay regions have been estimated, it is possible (depending of the application) to replace the coordinates of each stay point for the coordinates of the centroid of the stay region where the stay point belongs. Therefore, two stay points having the same semantic meaning (i.e., both are into the limits of the stay region) could now share the same location information.

Figure 1 shows an illustrative example of the three concepts. In the left map, the location points extracted for a hypothetical user visiting an European city are shown. Each color corresponds to a different day. Note that the paths followed by the user are different across the two days. The user stayed in the two days in the areas of the *Parc Olympique* and the *Lausanne train station*. One of the days (blue), the user also stayed in the area of *Parc de Milan*. The middle map shows the stay points estimated for each day using as input data the location points shown in the left map. Note that the location of the stay points do not fall in the same coordinates, but they represent the same semantic meaning, i.e., *Parc Olympique* and *Lausanne train station*. The right map shows the stay regions estimated using the previous stay points. Each stay region is represented using a rectangle. Now, using the stay regions estimated, it is possible, for instance, to describe the user activity in a day in terms of the visited stay regions. In the example, for the red day the user activity could be described as: *Parc Olympique* → *Lausanne train station*, and for the blue day as: *Parc Olympique* → *Parc de Milan* → *Lausanne train station*.

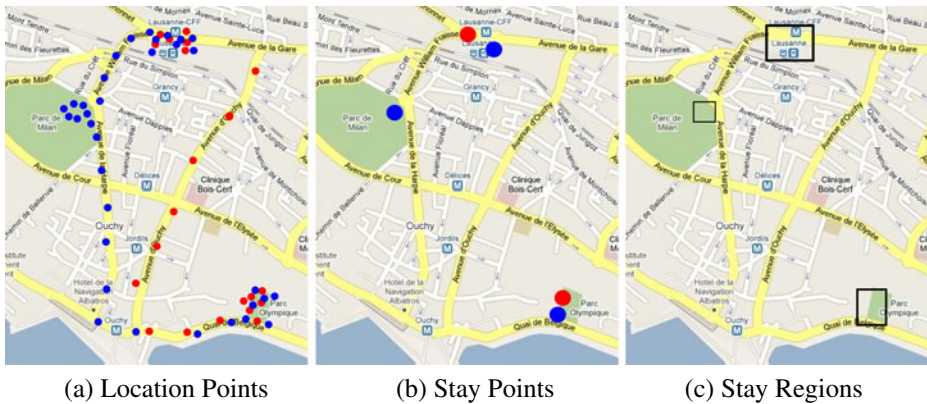


Fig. 1 *Left:* location points obtained for a hypothetical user during two days (red and blue). *Middle:* stay points discovered for the two days. *Right:* stay regions estimated using the previous stay points as input data

In this paper, an algorithm to accurately estimate stay points and stay regions from the real life of users from mobile phone sensor data is presented. The maximum size of the region and the minimum time that a user must be in this region are controlled using two parameters. In the proposed algorithm, an important constraint has been introduced, which consists in that between two consecutive location points their time difference must also be bounded. The use of this new constraint arises from the use in this work of data obtained from users' real lives. Thanks to this constraint all the consecutive location points included in a stay point are close in time. To estimate stay regions from stay points, a clustering algorithm must be used. In this paper, two clustering algorithms have been studied. The first one is a density-based method [7], previously used in [25], and the second one is a grid-based one presented in [28]. A preliminary version of this work appeared in [19].

One of the most important aspects of this work is that mobile phones are used to obtain user location data, instead of custom-made architectures or professional GPS sensors. The use of mobiles phones for large-scale sensing of human activity is an interesting alternative since people wear mobile phones almost all the time and therefore data can be obtained in a natural manner and without the necessity to wear another device, which could be uncomfortable to the user or unrealistic [6]. In addition, the data obtained by mobiles phones reflects better the real life of users, since the phones used in the experiments reported in this paper were their real (and unique) phones. However, one import drawback of using mobile phones is that location data might not be obtained everytime, since the GPS sensors included in mobile phones do not provide the same quality as professional dedicated GPS sensors. Then, when using mobile phones, there are more missing location data due to sensor failure to obtain GPS coordinates.

To collect data, a novel client-server system has been used, which records GSM, GPS, Bluetooth, Wifi and motion, in conditions that are feasible for large-scale sensing and comfortable for users. The client component of the system is installed in the mobile phone of the user to collect data. To save battery and also to provide more location information where GPS might not be available, the system scans Wifi

access points (APs) to build a private library of geocoded base stations, allowing future location observations when the user's phone observes a previously learned Wifi AP. In addition, it is also possible to recognize periods of time when the phone is static using the accelerometer and other sensors. Thanks to the use of this system and also to the way in which the information provided by the system has been used in this work, more location points can be obtained with respect to the case of only using the GPS sensor, thus allowing for large-scale sensing along normal user days.

In order to test the proposed approach, two kinds of experiments have been performed using data obtained from the mobiles phones of a set of volunteers. The first one has been designed to demonstrate the good performance of our approach on the task of discovering places of interest. Our approach has been compared with other important techniques found in the literature. In the second one, an study of the kind of places discovered for a large number of users have been performed, obtaining cues of interest to understand where people go in their real lives.

Summarizing, the main contributions of this work are as follows:

- We propose a new framework to accurately estimate places of interest from users' real lives location data. The proposed framework deals with missing location data, an inevitable problem when working with data from users' real lives.
- We manage the information obtained by the use of a robust sensing system which provides location data in conditions that are feasible for large-scale sensing and comfortable for users. Our approach is multimodal since location information is obtained from multiple sensors such as GPS, Wifi APs, GSM, accelerometer, among other.
- We conduct extensive experiments using location mobile phone data collected by a set of volunteers over a large time period to verify the effectiveness of the proposed approach discovering places of interest.
- We perform a study on where users stay in their real lives according to the places of interest discovered for a large set of volunteers using mobile phone data collected along almost a year.

This rest of the paper is organized as follows: Section 2 provides a discussion of the previous works in the area of interest location learning and recognition. In Section 3, a preview of the proposed approach is presented. Section 4 describes how location data can be obtained by using mobile phones and the proposed system for sensing data. In Section 5, the algorithms to estimate stay points and stay regions are explained in detail. The experiments performed to validate our work are discussed in Sections 6 and 7. Section 8 presents a discussion about how the performance of real-life applications can be benefited by using the algorithms presented in this paper. Finally, the most important conclusions are drawn in Section 9.

2 Related work

In this section, we review two families of methods to estimate places of interest, geometry-based [2, 14, 18] and fingerprint-based [12, 15].

2.1 Geometry-based methods

Geometry-based techniques use location data (such as GPS) to produce coordinates, circles, or polygons to describe the significant places where the user goes. These algorithms take a history of location points and find places where the person stays for significant periods of time using a clustering-based technique.

The main drawbacks of GPS include its inability to function well indoors, its occasional lack of accuracy due to the geometry of visible satellites, and the loss of signal in urban canyons and other “shadowed” areas. Early work on place learning from GPS used the loss of signal to infer the location of important indoor places as buildings [2, 18]. Marmase and Schmandt *ComMotion* algorithm [18], initially identified a place as a region, bounded by a certain fixed radius around a point, within which GPS disappears and then reappears. However they found that the GPS receiver often took several minutes to acquire its location when exiting a building and, therefore, when the signal was regained, the user was no longer at the place he had left. They modified their algorithm to identify a place as a position where the GPS signal is lost three or more times within a given radius. This approach is sufficient to identify indoor places that are smaller than a certain size (e.g., a home), but do not account for outdoor places and larger indoor places (an office complex or a convention center), and is prone to generating false positives (caused by the many possible outdoor GPS shadows).

A similar but improved approach to extracting significant places is proposed by Ashbrook and Starner [2]. Their algorithm defines a place as any logged GPS coordinate with an interval of time t between it and the previous point. Similar to Marmase and Schmandt, they realized that better results can be obtained changing the methodology to register a place when the signal is lost, and so is not dependent upon signal acquisition time. These candidate places are then merged using a variant of k-means clustering. Ashbrook and Starner’s technique is able to overcome the place-size limitations and most of the false positives that handicap Marmasse and Schmandt’s approach. However, the use of GPS signal loss to infer place still leaves us unable to infer important outdoor places and multiple places within a single building.

The previous algorithms depend on properties of the GPS satellite signals to work properly. To avoid this dependence, Kang et al. designed a new algorithm using temporal point clustering [14]. It takes as input a stream of timestamped coordinates derived from a location estimation system, and performs the segmentation and merging steps simultaneously using time-based clustering. One of the main differences from previous works is that location information is obtained using the PlaceLab system [17], which allows the devices to locate themselves by finding radio beacons such as Wifi APs, GSM cell phone towers, and fixed bluetooth devices that already exist in the environment. The main advantage of using this system instead of real GPS data is that it can obtain location information in many places where GPS cannot, and so, more location points can be obtained. Therefore, algorithms can estimate more accurately the places of interest. The main drawback is that PlaceLab has a critical dependence on the availability of beacon locations, otherwise it is impossible to estimate the location of the user. Although there are some public and user-created databases, there are still a lot of beacons without location information in practice, for instance, most of the ones corresponding to the geographical region where our experiments were conducted. Ye et al. [25] recently presented a similar

algorithm that also uses temporal-based clustering, but in this case, location data is obtained using GPS sensors. The main difference (in the strategy to obtain places of interest) is that this algorithm works offline where the one presented in [14] works online.

2.2 Fingerprint-based methods

Fingerprint algorithms, in contrast, obtain a list of places where the user goes, but provide no direct information about where the place is geographically located. Fingerprint-based techniques detect stable radio environments that indicate a stay. They define the fingerprint of a place as a vector of currently visible cell towers or Wifi access points, and use it to recognize when the device returns to a place.

Two of the most important fingerprint-based algorithms are *BeaconPrint* [12] and *PlaceSense* [15]. Both have been designed to discover places by continuously monitoring the radio beacons in the environment around a mobile device. They discover places detecting a stable radio environment that indicate an entrance to a place and recognizing when it changes signaling a departure. One of the biggest challenges is dealing with intermittent beacons. According to [15], *PlaceSense* works better in this case than *BeaconPrint* by using multiple successive scans and also thanks to a more robust technique since it uses separate mechanisms for detecting entrances and departures.

Both papers performed a comparative study of the accuracy of their algorithms against geometry-based techniques. In both cases, fingerprint-based algorithms obtained better accuracy results on learning and recognizing places, mainly because geometry-based methods use data from GPS sensors while fingerprint-based use data from GSM and Wifi sensors, which have more coverage in cities. However, the main drawback of fingerprinting is that the exact location information cannot be obtained using these kinds of techniques. Then, for many applications where the location of the place must be known, geometry-based ones are the only choice.

3 An overview of our approach

The algorithm to learn interest locations strongly depends on the data that can be obtained from the sensor. Most systems in the literature for human activity sensing have used custom-made sensor architectures rather than the integrated frameworks existing on phones [2, 12, 14, 18]. However, the need to carry additional devices and their social acceptability limits the use of some of these architectures at large-scale.

In this work, Nokia N95 8GB smartphones have been used to collect location data. Figure 2 shows the framework of the proposed approach. Data has been collected using a client-server system which records GSM, GPS, Bluetooth, Wifi, motion, as well as most of the smartphone applications, in conditions that are practically feasible, with respect to phone usage and battery consumption, for large-scale data collection. One of the main advantages of this system is the design of a set of operation modes (in the client side) that imposes a tradeoff between sensing sampling rate and battery consumption, resulting in efficient sampling at

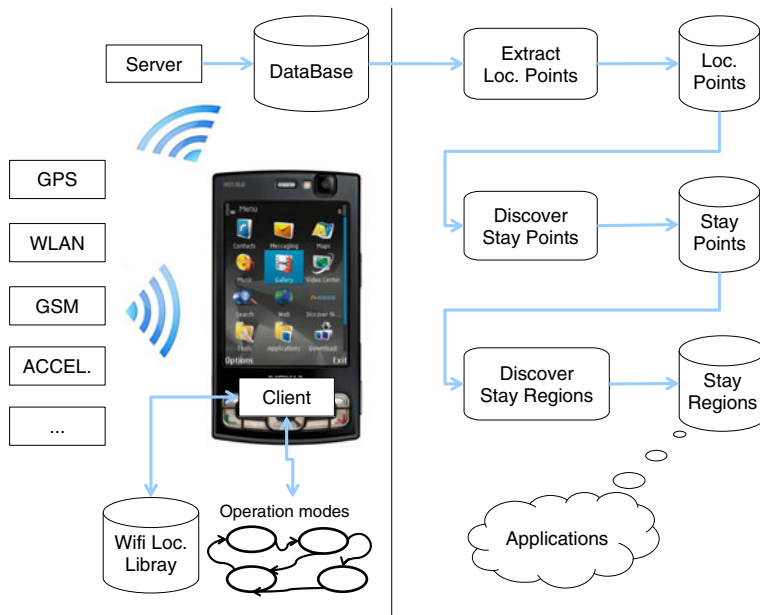


Fig. 2 Framework of the approach presented in this paper. It is divided in two parts, The first (*left*) is the system to capture multimodal data by using mobile phones. The second part extracts user location data to first discover stay points and after to discover stay regions. The resulting stay regions can be used by many applications

reasonable computational cost. The system consists of a client program that runs on the phone to collect and transmit data over Internet, and a secure back-end database server to store data. The client stores data until a Wifi connection is available to upload the logged data to the server, which usually happens at home or at work.

From the database data, location data is extracted (see Section 4). Then the algorithm to estimate stay points is performed using the data obtained for each day of a user. Using the stay points discovered for the user for all the days, the algorithm to discover stay regions is executed. Both algorithms are explained in Section 5. Once places of interest have been extracted, they can be used in many applications as the ones discussed in Section 1.

4 Obtaining location points from multimodal mobile phone data

While the extraction of location data directly from GPS is conceptually attractive, it has two practical limitations when using mobile phones. First, GPS is expensive in terms of battery consumption (if GPS is active continuously on an N95, the battery would drain completely in three hours). Second, the GPS sensor on phones fail in practice more often than not, even when using assisted GPS. At the same time, dealing with real-life human location has two inherent advantages. On one hand,

most people are habitual and tend to spend a significant proportion of time in exactly the same places [6, 8, 9]. On the other, smartphones are endowed with multiple sensors from which location can be inferred.

Our work relies on a location estimation system, recently described in [16], that addresses the above four points. This system tackles the critical problem of battery consumption by defining an adaptive sensing strategy for location estimation, which integrates the basic knowledge about human habits with the use of multiple phone sensors. More concretely, the phone sensing strategy is defined by a set of *operating modes*, each of which decides what sensors ought to be activated, at what sampling rate, and how to make transitions between each other.

The knowledge about habitual behavior is represented as follows [16]. For each user, the phone periodically scans for Wifi APs to build a map of georeferenced Wifi APs that are observed for a reasonable time period. Urban areas in the developed world often have a high density of Wifi APs in most neighborhoods, so the phone client can build a map of the user's habitual locations, and use this knowledge to avoid activating the GPS sensor unnecessarily. Each phone learns its own Wifi AP location map using the GPS data extracted in the time periods when the phone observes a Wifi, if available. After that, the Wifi AP location is estimated as the centroid of all GPS observations obtained for each AP. From then on, each time the phone observes a Wifi in its map, it switches off the GPS sensor and assigns the corresponding learned location, thus saving battery. This operating mode is called *Wifi map*. This method has similarities to PlaceLab [17], as both try to obtain additional location points to those that the sole GPS sensor can provide by using Wifi APs. The system we use, however, has the important advantage of not needing an external database of beacon locations, but rather learns it by itself. This is important since there are still a lot of beacons without available location information at the geographical region where our experiments were conducted. The system we use also bears similarities with the sensing system reported in [23], which is aimed at recognizing user activities.

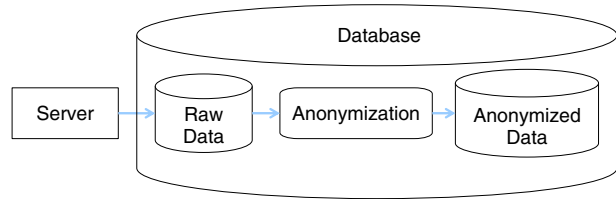
The phone also detects when it is *static* by combining the accelerometer, GSM, and Wifi sensors [16]. In this mode, GPS is also switched off to save battery. The phone enters this mode when no significant motion is detected using the accelerometer and the phone remains connected to the same GSM tower or Wifi AP for some time. Similarly to the Wifi location map creation, static location can be estimated using the previous GPS data obtained (if available) just before entering into this mode.

Summarizing, with the multimodal system, there are four possible situations regarding location estimation:

1. In *Wifi Map* mode, location is obtained from the learned Wifi location map.
2. In *Static* mode, location can be obtained from the location obtained just previous to enter such mode (if available).
3. If the phone is in neither of the previous modes, location is directly obtained from GPS when the latter is active and working.
4. If the phone is in neither of the previous modes, and GPS fails (due to no coverage, phone switched off, etc), it is not possible to obtain location.

In real life, the last situation is still common. Therefore, algorithms to learn places of interest should handle this issue.

Fig. 3 Raw data collected by the system is anonymized. A user can access to read, modify or delete his own data. The user can also give read permissions to other users to access his data



Regarding energy consumption, the client running on a N95 results in significant savings of battery life, and allows for the use of the phone on a daily basis in regular conditions. Based on an empirical evaluation, three conditions can be observed [16]:

- In the case where the set of *operating modes* is not used, and all sensors are on, the phone is operational for less than three hours before the battery drains.
- In contrast, if the phone stays constantly in the *Wifi Map* operating mode, which is one of the most common ones in practice (occurring in average on 40% of the time), the phone battery lasts for over 20 hours.
- Finally, if the phone stays without interruptions in outdoors (i.e., with Wifi and GPS sensors active), the battery drains in about 8 hours.

In practice, the phone operates between the battery duration bounds described in the last two cases, and can in principle last for a “normal day” for most people, only requiring to be plugged in for recharging once a day, typically when people return home in the evening.

4.1 About protecting users’ privacy

Protecting users’ privacy is important when working with real mobile phone data. In our case, user data collected by the system has been anonymized using an appropriate protocol [16]. Figure 3 shows this process. In particular, location data has been anonymized by removing the last decimals of the coordinates. Therefore, the anonymized version of location data has lower accuracy than the raw one. Users can gain access to the raw version of their data to visualize, remove or modify any datum that they want. In addition, a user can give read permission to any other user to access its data.

5 Algorithms to discover places of interest

For each day and for each user, the user’s mobile phone provides a list of consecutive location points $l_p = (p_1, p_2, \dots, p_N)$ where N is the number of location points. Each p_i is defined using a 3-tuple: $p_i = (lat, long, T)$ containing the location (latitude and longitude) and the time when the sensor obtained this location. Note that $\forall i \in [2, N]$: $T_i > T_{i-1}$. Also note that all $p_i \in l_p$ have been collected in the same day, then T refers only to the hour, minutes and seconds when the location information was captured.

As explained previously (see Section 4), location points could be obtained from GPS sensor, from *Wifi map* modes or from *static* modes.

5.1 Estimating stay points from location points

Using the list of consecutive location points l_p for a user and for a day, the objective of the algorithm is to obtain a list of stay points $l_{sp} = (sp_1, sp_2, \dots, sp_M)$, where M is the number of resulting stay points. Each sp_j is defined using a 4-tuple: $sp_j = (lat, long, T^{\text{start}}, T^{\text{end}})$ containing the location, and time when the stay point started and ended. Note that $\forall j \in [2, M - 1]: sp_j.T^{\text{start}} > sp_{j-1}.T^{\text{end}}$ and $sp_j.T^{\text{end}} < sp_{j+1}.T^{\text{start}}$.

The proposed algorithm to estimate stay points from location points is an improved extension to the one proposed at [25]. In this algorithm, a stay point represents a geographic region in which a user stays for a while. More formally, there exists a stay point from location point p_s to location point p_e if the two following constraints are fulfilled:

$$\text{SpaceDistance}(p_s, p_e) < D_{\max} \quad (1)$$

$$\text{TimeDifference}(p_s, p_e) > T_{\min} \quad (2)$$

where D_{\max} and T_{\min} are two tuning parameters. D_{\max} is the maximum distance that user can cover in a place to be considered as stay point. T_{\min} is the minimum time that the user must be in the same place to be considered as stay point.

In the proposed algorithm, a new constraint is added:

$$\text{TimeDifference}(p_k, p_{k+1}) < T_{\max}, \forall k \in [s, e - 1] \quad (3)$$

This new constraint establishes that between two consecutive location points the time difference must be bounded by T_{\max} . Then, all the consecutive location points belonging to a stay point must be close in time. The proposed steps to estimate stay points from location points are resumed in Algorithm 1.

Figure 4 illustrates the behavior of the algorithm showing a sequence of location points in a latitude-longitude coordinate space. Two consecutive location points are connected with lines. The line is green when the time difference between the two

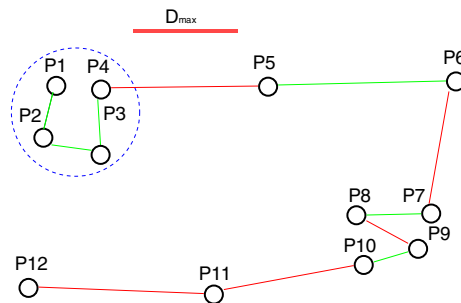


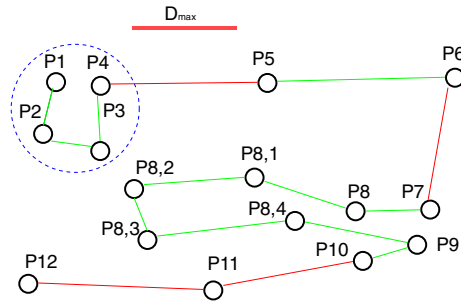
Fig. 4 An example of the behavior of the proposed technique to estimate stay points from location points. Two consecutive location points are connected with a *green line* when the time difference between the two connected location points is less than T_{\max} and it is *red* otherwise. A stay point is detected in the set (p_1, p_2, p_3, p_4) , but there is not a stay point in the set (p_7, p_8, p_9, p_{10}) since time distance between p_8 and p_9 is bigger than T_{\max} (see text for details)

Algorithm 1 Algorithm to estimate stay points from location points**Require:** $l_p = (p_1, \dots, p_N)$: List of location points. T_{\min}, T_{\max} : Time thresholds. D_{\max} : Distance threshold.**Ensure:** l_{sp} : List of resulting stay points. $i \leftarrow 1$; $l_{sp} \leftarrow \emptyset$;**while** $i < N$ **do** $j \leftarrow i + 1$;**while** $j < N$ **do** $t \leftarrow \text{TimeDifference}(p_j, p_{j-1})$;**if** $(t > T_{\max})$ **then** $i \leftarrow j$; break ;**end if** $d \leftarrow \text{SpaceDistance}(p_i, p_j)$;**if** $d > D_{\max}$ **then** $t \leftarrow \text{TimeDifference}(p_i, p_{j-1})$;**if** $t > T_{\min}$ **then** $[lat, long] \leftarrow \text{EstimateCentroid}(p_k | k \in [i, j - 1])$; $T^{\text{start}} \leftarrow p_i.T$; $T^{\text{end}} \leftarrow p_{j-1}.T$; $sp \leftarrow [lat, long, T^{\text{start}}, T^{\text{end}}]$; $l_{sp} \leftarrow l_{sp} \cup sp$;**end if** $i \leftarrow j$; break ;**end if** $j \leftarrow j + 1$;**end while****end while**

connected location points is less than T_{\max} and it is red otherwise. A stay point is detected (grouping p_1, p_2, p_3 and p_4) since we are assuming that the time difference between p_1 and p_4 is bigger than T_{\min} and the physical distance between p_1 and p_4 is less than D_{\max} . But, even assuming that the time difference between p_7 and p_{10} is bigger than T_{\min} and the physical distance is less than D_{\max} , there is not a stay point in the set p_7, p_8, p_9 and p_{10} , since the time difference between p_8 and p_9 is bigger than T_{\max} , exceeding the maximum time allowed between two consecutive location points.

When working with real-life data, it is possible to often have situations with substantial time difference between two consecutive location points, which is mainly due to the limitation of accurately sensing location in many real places. One possibility is the one shown in Fig. 5 that illustrates the real behavior of the user. Between location points p_8 and p_9 , the user visited many other locations (from $p_{8,1}$ to $p_{8,4}$), but they could not be captured by the sensor. Note that without using the third constraint (see

Fig. 5 An example of which can be the real situation between location points p_8 and p_9 (see text for details)



Eq. 3) an incorrect stay point could have been obtained between location points p_7 and p_{10} .

5.2 Estimating stay regions from stay point

For each user and for each day, the algorithm for estimating stay points is executed. The process of estimating stay regions consists of performing a clustering technique using all the stay points extracted for a user. The general steps to estimate stay regions from stay points for a user are summarized in Algorithm 2. Note that we are assuming that we are working with data from Q days, and therefore there are Q different list of stay points l_{sp} , one for each day.

Some of the clustering techniques that have been used for similar problems in the past include k-means or variants as in [2], density-based as in [25] and grid-based as in [28]. Density-based clustering produces better results than k-means but has the problem of not constraining the clusters size. To deal with this problem, Zheng et al. designed in [28] a new grid-based clustering algorithm which constrains the cluster size. From the results obtained in the experiments shown in Section 6, the grid-based is a better option to perform this task.

6 Evaluation

This Section presents the first group of experiments to evaluate the performance of our approach on discovering places of interest.

Algorithm 2 Algorithm for estimating stay regions from stay points

Require: $l_{sp}^1, \dots, l_{sp}^Q$: set of stay point lists.

Require: D_{\max} : Distance threshold.

Ensure: l_{sr} : List of resulting stay regions.

$SP \leftarrow \emptyset$;

for $d \leftarrow 1 : Q$ **do**

$SP \leftarrow SP \cup l_{sp}^d$;

end for

$l_{sr} \leftarrow \text{Clustering}(SP, D_{\max})$;

6.1 Data acquisition

Our data has been collected from 8 volunteers using the N95 smartphone during 5 continuous months of real life. Users live in three different European communities ranging from small villages to a mid-size city. Participants used the mobile phones in a normal manner, since it was their real (and unique) phone. No instructions were given to the participants about which places to visit or about how long they should have stayed in such places. The only recommendation was to carry the phone as frequently as they could, something they would naturally do as it was their real phone.

Instead of having to recharge the phone several times a day (as in [15]) users only needed to recharge once a day (typically during the night). In total, more than 550.000 location points were obtained served from 24/7 phone data.

Data were obtained from raw database, since volunteers shared their data with the authors. The reason for using raw data instead of the anonymized version is because raw data provides richer information and more accurate location data, and it is very important to evaluate the performance of the algorithms by using as accurate location data as possible.

6.2 Evaluation methodology and collecting the ground truth

To evaluate the effectiveness of the proposed framework, we must first decide on a criterion for evaluation. Intuitively, a place extraction scheme should be judged on how well it identifies the places that a user regards as important. Therefore, it is necessary to have a ground truth of the places reflecting the real-life patterns of users.

Three are the possibilities for collecting ground truth from users:

- Users can follow a scripted tour of places to stay [15].
- Users can keep a diary of the name and time they entered and left every place during the collecting time [12, 15].
- Users can fill, at the end of the collecting time, a survey of the places where they stayed.

The first two options are uncomfortable for users when collecting data during a long period of time. Furthermore, the scripting option might not be looking at truly interesting places from the users viewpoint, or be unrealistic (e.g., requesting people to visit as many places as they can). In addition, the results highly depend on the involvement of each user with the scripted task. The third option above has the main drawback that some places might not be remembered after some time. In our work, to collect ground truth, users filled a survey with the places where they remembered to have stayed for more than 30 min.

To evaluate our real-life location-of-interest discovery method, we used an evaluation system partially based on the one recently proposed in [15]. Stay regions (i.e., places of interest) extracted using the proposed framework are called *Discovered*, and places remembered by users are called *Remembered* places. Remembered places that have not been discovered are called *Missed*, while places that are both remembered and discovered are called *Correct*, *Merged* or *Divided*. On one hand, if two different places reported by the user have been discovered as a single place, then

the places are called *Merged*. On the other hand, if a single place reported by the user has been discovered as multiple places, it is called *Divided*. Others that have been discovered but not remembered are called *Ghost candidates*.

If we rely on participants' memory, all the places discovered but not remembered (i.e., *Ghost candidates* places) are places where the user did not stay. But, unfortunately, in practice there are a lot of places where participants stayed but that they can not actually remember them. This is a natural limitation of human recall. To deal with this issue, *Ghost candidates* places are divided in *False* and *Forgotten*. A *Ghost candidates* place is labelled as *Forgotten* if the user claims that the place has been correctly discovered *after* reviewing the results, and it is labelled as *False* in the opposite case.

Finally, it is possible that users remember places where they stayed, but they can not remember if they carried the mobile phone, if it was switched on, etc. Therefore, in principle, some places labelled as *Missed* could (strictly speaking) not be algorithmic errors, since participants can remember places that are impossible to estimate since there is not location information available for this moment of the day. Unfortunately, it is impossible to deal with this issue.

With the above definitions, the objective of an automatic method is to obtain more *Correct* and *Forgotten* places, while reducing *False* and *Missed* places. To evaluate the performance, a set of measures (P_α , P_β , R , F_α and F_β), where # stands for “number of”, and P , R and F denote *Precision*, *Recall* and *F-measure*, have been defined as follows:

$$P_\alpha = \frac{\#Correct}{\#Discovered} \quad (4)$$

$$P_\beta = \frac{\#Correct + \#Forgotten}{\#Discovered} \quad (5)$$

$$R = \frac{\#Correct}{\#Remembered} \quad (6)$$

$$F_\alpha = 2 * \frac{P_\alpha * R}{P_\alpha + R} \quad (7)$$

$$F_\beta = 2 * \frac{P_\beta * R}{P_\beta + R} \quad (8)$$

Note that, while P_α and F_α are strict measures, P_β and F_β introduce a correction factor for users' forgetfulness.

6.3 Experiments

Several experiments have been performed to show the performance of the proposed approach:

- Experiment 1 shows that with the combined use of the system explained in Section 3 and our data management strategy (see Section 4), more location points can be obtained with respect to using only the GPS sensor.
- Experiment 2 provides a comparison between two clustering techniques to obtain stay regions from stay points. We demonstrate that the grid-based technique obtains better results than the density-based one.

Table 1 Percentage of time when user location data have been obtained directly from GPS, *Wifi map* mode, *static* mode, and not obtained

GPS	Wifi map	Static	No location
4%	35%	24%	37%

- Experiment 3 provides a comparison of our method with the ones presented by Kang et al. [14] and by Ye et al. [25]. We demonstrate that on our real-life dataset, our proposed approach outperforms [14] and [25] on discovering the users' most important places.

6.3.1 Experiment 1: extracting location points

To assess the effectiveness of the system for sensing location, we first divided each day of the 5-months period in slots of 5 min. Then, for each slot, we counted the number of location points of each source, i.e., if the location point has been obtained directly by the GPS sensor, by the *Wifi map* mode or by the *static* mode. The source with the maximum number of entries is the one that is assigned to this time slot. If there are no location points in a time slot, it is labelled as “No location”. This process is repeated for all days and for all users.

Table 1 shows the average values for all users and for all days. Note how by using the state-based sensing system, approximately for 63% of the day it is possible to estimate the location of a user. This is an interesting result, as some studies [15, 17] and our own experiments, show that GPS coverage is available only 5–30% of the time on average for a device carried by users during a typical day.

6.3.2 Experiment 2: density-based vs grid-based for stay region clustering

In this experiment a density-based clustering technique is compared against a grid-based one for clustering stay points into stay regions. First, stay points for all participants and for all days have been estimated using the proposed approach (see Algorithm 1), setting $T_{\min} = 30$ min., $D_{\max} = 250$ m. and $T_{\max} = 10$ min. Then, both clustering techniques have been applied to estimate stay regions.

DBSCAN [7] has been selected as representative density-based clustering technique. This algorithm has two main parameters, the minimum number of points in a cluster and the maximum distance between two points belonging to the same cluster. The minimum number of points has been set to 2 and the maximum distance to D_{\max} . *DBSCAN* also detects outliers, that in our case are isolated stay points. Each outlier detected, a cluster (stay region) with only the isolate stay point has been created.

The grid-based technique is the one presented in [28]. The only parameter of this algorithm is the maximum size allowed for a cluster. We used D_{\max} for this purpose. Note that in the grid-based method the maximum size of a cluster is bounded by D_{\max} (the same value used for discovering places), while in the density-based one, the clusters size is not bounded for any quantity (even if smaller values than D_{\max} are used as maximum distance between two points).

Figure 6 shows the typical results of both clustering techniques for a particular user. It is clear that the grid-based technique obtains better results since *DBSCAN* tends to merge stay points with different semantic meaning in the same clusters.



Fig. 6 **a** Stay points. **b** Stay regions obtained using a density-based clustering algorithm. **c** Stay regions obtained using a grid-based clustering algorithm

In particular, using our real-life dataset, *DBSCAN* has obtained 67 *Correct* and 26 *Merged* places, while the grid-based has obtained 89 *Correct* and only 4 *Merged*.

6.3.3 Experiment 3: comparative results on stay point learning

We compare the proposed approach against Kang et al. [14] and Ye et al. [25] ones, for estimating stay points from location points. An important point of this experiment is that the three algorithm have been tested in the same conditions regarding the input location data, since our objective (in this experiment) is to test which strategy is the best for learning stay points from location points. Therefore, the input of the three algorithms are the same location points obtained with the proposed framework instead of just GPS data. In addition, the same parameters have been used in the three algorithms, i.e., $T_{\min} = 30$ min., $D_{\max} = 250$ m and $T_{\max} = 10$ min. The three algorithms use T_{\min} and D_{\max} , but only the proposed technique uses T_{\max} .

After the stay points have been extracted using the three algorithms, the grid-based clustering technique has been applied, using each set of extracted stay points to obtain the places of interest. Tables 2 and 3 show the results obtained for the three algorithms.

Table 2 *Discovered, Remembered, Correct, Missed, False, Forgotten, Merged and Divided* obtained for all users and all days

	Disc.	Remem.	Correct	Missed	False	Forg.	Merg.	Div.
[14]	290	125	88	33	42	161	4	0
[25]	314	125	96	25	46	172	4	0
Proposed	212	125	89	31	30	92	4	0

Table 3 Results obtained using as input the location points obtained for all users and all days

	R	P_α	F_α	P_β	F_β
[14]	0.70	0.30	0.42	0.86	0.77
[25]	0.76	0.31	0.44	0.85	0.81
Proposed	0.71	0.42	0.53	0.85	0.78

In bold best values

The proposed technique discovered fewer stay regions than [25] due to the use of the new constraint (see Equation 3). However, the number of places considered as Correct and Missed is very similar. Therefore, if we rely only in user remembered places (i.e., considering all *Ghost candidate* places as False) the proposed technique obtains better results according to the F_α measure. Note that, while human recall is imperfect, users tend to remember the most important places in their real lives and therefore our method seems to be more effective at extracting the most relevant places for a person.

On the other hand, as discussed previously, users forget a lot of places that they visit but cannot remember when they fill a survey. To know how many *Ghost candidate* places could be considered as *Forgotten*, we showed the results to the participants and asked them about the number of *Forgotten* places. In this cases the performance of the three algorithms is quite similar, being [25] slightly better than the proposed one.

6.4 Differences across participants

Our data has been collected from 8 volunteers. Figure 7 shows the users' data distribution (box plot) for measures R , P_α , P_β , F_α and F_β obtained using the proposed approach. Regarding the R value, similar values have been obtained (see Fig. 7e) for the majority of the users, with the exception of two users, one with a very low value, (i.e., the method did not discover the most of the remembered places), and the other showing an opposite pattern, (i.e., the discovery was highly accurate). The data distribution for P_α and P_β measures, and also for F_α and F_β , are more spread than in the case of R . In general, the inclusion of the *Forgotten* place category makes the β measures less spread than the α measures.

6.5 Parameters setting

The algorithms proposed in this work, have only three parameters: D_{\max} , T_{\min} and T_{\max} .

D_{\max} is the maximum distance that a user can cover in a place to be considered as a stay point (see Section 5.1) and also is the maximum size of a stay region (see Section 5.2). On the one hand, big values of D_{\max} could merge several places in only one. On the other side, small values could divide a place. The correct value depends on the application. From our experience, values from 200 to 300 meters produce the best results.

T_{\min} is the minimum time that the user must be in the same place to be considered as a stay point. High values allow the discovery of places where the user stayed for long time (Home, Work, etc.). On the other hand, small values allow the discovery of places where the user stayed for a few minutes (Bus stop, Train station, etc.). Similar

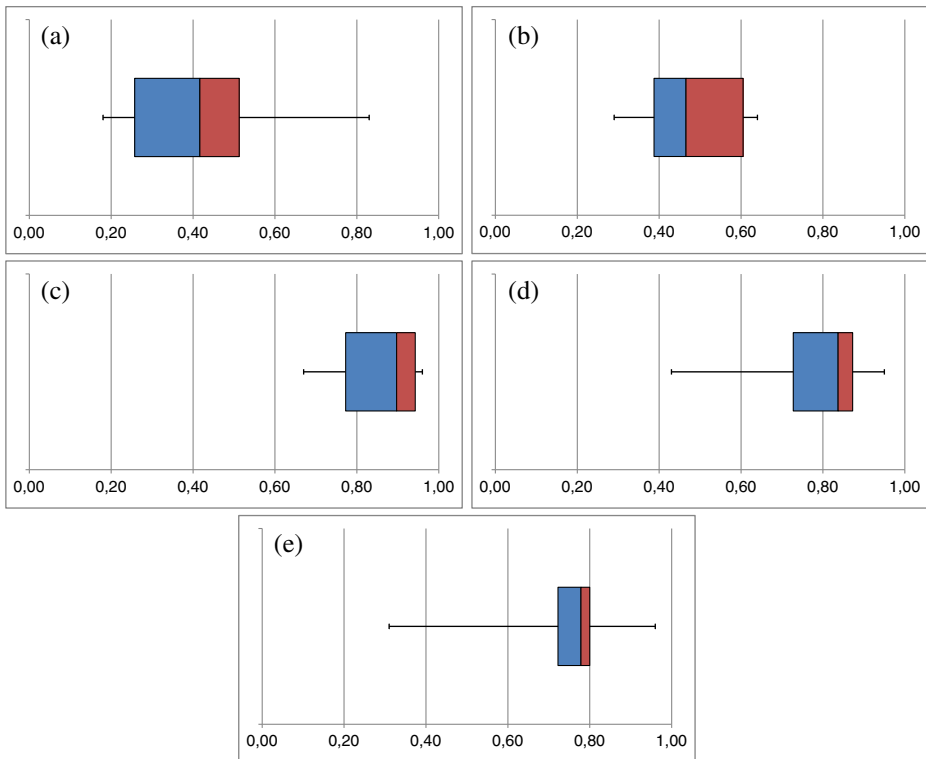


Fig. 7 Users' data box plots: **a** P_α , **b** F_α , **c** P_β , **d** F_β and **e** R

to D_{\max} , the correct value depends on the application. From our experience, values from 20 to 40 min produce the best results.

Finally, between two consecutive location points the time difference must be bounded by T_{\max} (see Section 5.1). High values of T_{\max} allow to discover more places, but more *False* places could be discovered too. This could be useful in some applications where we are interested in discovering a lot of places of interest. On the other hand, by setting T_{\max} to a low value, only the most significant places for users are going to be discovered, which can be also useful to another application where we are only interested in the most meaningful places for users. The optimum value for T_{\max} thus, also depends on the application.

7 Analyzing discovered places of interest

This section presents the second group of experiments, where we have used data provided by a large group of participants from the Lausanne data collection campaign [16]. In this case, the objective is twofold: one one hand, the experiments contribute to the evaluation of the performance of our approach on discovering places of interest. On the other hand, the experiments investigate which kind of

places can be obtained using the proposed approach and also study where people stay in their daily lives.

7.1 Data acquisition

In this case, data has been collected from 124 volunteers using the N95 smartphone during almost a year of real life (in particular, 335 days) [16]. As in the previous case, users live in different communities ranging from small villages to a mid-size city. The participants were instructed to use the phones as their primary phone and no instructions were given to users about which places to visit or about how long they should stay in specific places. In contrast to the experiments shown in Section 6, the used data come from the anonymized version to respect users' data privacy. Therefore, location data have less accuracy than in the previous set of experiments.

7.2 Evaluation methodology and collecting the ground truth

Our approach to discover places of interest was used to discover the most important places of interest of each user. The algorithm parameters were set to: $T_{\min} = 30$ min., $D_{\max} = 250$ m and $T_{\max} = 10$ min (i.e., the same configuration than in the experiments showed in Section 6).

Eight discovered places were selected for each participant. Five of the eight places were the most frequently visited places of the user. Places as *Home* and *Work* almost always fall in the top five list. The other three were randomly chosen from the remaining list of less frequently visited places. In total, 979 places of interest were selected. Note that the number of selected places of interest is not 992 (i.e., 124×8), since for some users less than 8 places were discovered.

The small list of selected places was shown to each user using a web application that places each location on a map. Participants labeled each place to acquire labels for the locations, using a set of 22 mutually exclusive, predefined labels shown in Table 4. In addition, the number of times (visits) that the user stayed at each place and the duration of the stays (in minutes) were also calculated automatically.

Using the labels provided by the participants, a new set of measures of the accuracy of the method on discovering real places of interest was defined as follows (where # stands for “number of”):

$$Acc_{\text{know}}^u = 1 - \frac{\text{\#unknown places}}{\text{\#selected places}} \quad (9)$$

$$V_{\text{know}}^u = 1 - \frac{\text{\#visits to unknown places}}{\text{\#total visits}} \quad (10)$$

$$D_{\text{know}}^u = 1 - \frac{\text{duration of visits to unknown places}}{\text{total duration}} \quad (11)$$

A place is considered as unknown when a user labeled this place using the label “*I don't know*”. V_{know}^u can be interpreted as the amount of visits to known places with respect to the total number of visits to the selected places for the user u , and D_{know}^u as the duration (in minutes) of the stays at known places with respect to the total duration of the stays in all selected places for user u . Values of Acc_{know}^u close to 1

Table 4 List of labels

Label	Meaning
0	My home
1	My freetime home
2	Holiday resort or vacation spot
3	Home of a friend, relative or colleague
4	Restaurant or bar
5	My main workplace
6	My other work place
7	The work place of my friend, colleague or relative
8	My main school or college place
9	My other school or college place
10	School/College place of my friend, colleague or relative
11	The school or daycare of my child
12	Other location my child visits
13	Location related to transportation
14	Shop or shopping center
15	Cultural or entertainment location
16	Night club
17	Place for indoor sports
18	Place for outdoor sports
19	Place for my non sports related hobby
20	Place to hang out or relax (e.g., park)
21	Other
22	I don't know

indicate that the user u knows most of the discovered places and values of V_{know}^u and D_{know}^u close to 1.0 indicates that the user u stays most of the time in known places, i.e., that the proposed method is able to detect places with semantic sense for the individuals.

Another measure to analyze the behavior of our method is the ability to detect what are perhaps the two most important places for people: *Work* and *Home*. Since some of the participants are students, school/college have also been included in the *Work* category. Then, according to the labels shown in Table 4, *Home* places are the ones with labels 0 and 1, and *Work* places are the ones with labels 5, 6, 8, and 9. With these definitions, two new measures Acc_{home} (number of users where *Home* has been detected with respect to the total number of users) and Acc_{work} (number of users where *Work* has been detected with respect to the total number of users) can be estimated as follows (where # stands for “number of”):

$$Acc_{\text{home}} = 1 - \frac{\text{\#users with discovered Home places}}{\text{\#total users}} \quad (12)$$

$$Acc_{\text{work}} = 1 - \frac{\text{\#users with discovered Work places}}{\text{\#total users}} \quad (13)$$

7.3 Experiments

Several experiments have been performed to assess our approach:

- Experiment 4 studies the activity level of each user, i.e., on how much users used their mobile phones and contributed data.

- In experiment 5, we study the accuracy of the method on discovering the most important places for users.
- Experiment 6 studies popular place categories, i.e., places where people go in their daily lives using the labeling process explained previously in this section.

7.3.1 Experiment 4: estimating users activity level

In the sample, not all the participants signed up for the campaign at the beginning. In addition, not all volunteers used the phone in the same manner. Therefore, it is important to estimate the activity level of each participant to have a measure of the reliability of each user's data. For this purpose, we estimated for each user, f_{day}^u as the percentage of days where the mobile phone of the individual u was active and f_{duration}^u as the percentage of minutes of a day (with respect to the complete day) where the user u stayed in any of the small set of the eight or less selected places. A day is considered as active when at least a location point could be collected, and inactive in the opposite case. The measures f_{day}^u and f_{duration}^u can be estimated as follows:

$$f_{\text{day}}^u = \frac{\Omega^u}{\Omega} \quad (14)$$

$$f_{\text{duration}}^u = \frac{\sum_{i=1}^{\Pi^u} \delta_i^u}{\Omega^u \times 24 \times 60} \quad (15)$$

where Ω is total number of days of the campaign (335), Ω^u is the number of active days of user u , Π^u is the number of selected places for user u (usually 8) and δ_i^u is the duration of all the stays (in minutes) at the i_{th} place for user u .

Table 5 shows the mean and standard deviation of f_{day}^u and f_{duration}^u for all users, for the 92 users (74%) with $f_{\text{day}}^u > 50\%$ (users' mobile phone were active more than half of the days) and for the 57 users (46%) with $f_{\text{day}}^u > 75\%$ (they were active more than the 75% of the days). There are strong differences across users on the number of active days, since some users participated in the campaign since the beginning and others much less time. In addition, there are users who used the mobile phone almost everyday while others used it less frequently. Figure 8a shows the box plot of the active days for all users. Taking into account only the most active users (i.e., users with $f_{\text{day}}^u > 75\%$), the value is less spread than when using all users, but there are important differences.

Regarding the mean of f_{duration}^u for all users, the stay times on the selected places represents the 37.86% of the day with strong differences across users (see second row of Table 5 and Fig. 8b). When this value is estimated using only the most active users

Table 5 Mean of the percentage of active days and the percentage of minutes of a day where the user stayed in the selected places

	All	>50%	>75%
$\overline{f_{\text{day}}^u}$	65.23 (26.47)	78.15 (15.10)	88.55 (7.23)
$\overline{f_{\text{duration}}^u}$	37.41 (15.86)	38.94 (14.88)	44.14 (13.14)

Standard deviation is shown in parenthesis. The first column shows data for all users, the second one for the users with $f_{\text{day}}^u > 50\%$ and the third one for the users with $f_{\text{day}}^u > 75\%$

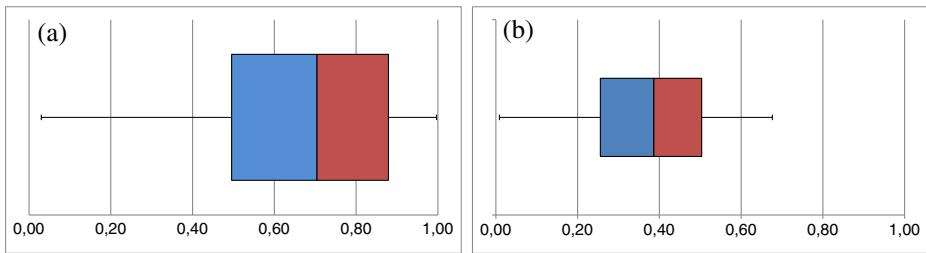


Fig. 8 Box plot of f_{day}^u (a) and f_{duration}^u (b) for all users

(i.e., users with $f_{\text{day}}^u > 75\%$), the value represents 44.14% of the day, i.e., almost half day.

Note that f_{duration}^u should not be confused with the value estimated in the Experiment 1. The 63% obtained in that experiment can be interpreted as the percentage of the day where location points can be estimated using the sensing approach presented in Section 3, while the mean of U_{duration}^u can be interpreted as the percentage of the day where a user stayed in the small set of selected places, which gives by construction less stay time.

7.3.2 Experiment 5: accuracy evaluation

Table 6 shows the mean and standard deviation for Acc_{know}^u , V_{know}^u and D_{know}^u measures. The places detected by the proposed method are almost always known by the users and there is not significant differences when the activity level of the users is taken into account. The number of visits and the duration of the stays in the known places also represent almost the total duration of the selected places, i.e., the unknown places represent a very small portion, almost insignificant, of the users' time. The unknown places may be algorithmic errors or places with very low significance for users, since users can not remember what kind of place they are. It is also important to emphasize that for 57.26% of the users, all places were known, and for 83.87% of the users the number of unknown places was 1 or less.

Table 7 shows the estimated values for Acc_{home} and Acc_{work} . *Home* places have been detected for almost all the users, without differences with respect to the users' activity level. This result can be considered as expected since *Home* places are the most important for a user and they are usually the places where users stay for the most of the time in a day. Furthermore, the method is able to detect most of the users' *Work* places, but the accuracy level is smaller than in the case of *Home* places.

Table 6 Mean of the accuracy measures on discovering places of interest

	All	>50%	>75%
$\overline{Acc}_{\text{know}}^u$	0.92 (0.11)	0.92 (0.11)	0.91 (0.11)
$\overline{V}_{\text{know}}^u$	0.99 (0.05)	0.99 (0.04)	0.99 (0.04)
$\overline{D}_{\text{know}}^u$	0.99 (0.04)	0.99 (0.04)	0.99 (0.05)

Standard deviation is shown in parenthesis. The first column shows data for all users, the second one for the users with $f_{\text{day}}^u > 50\%$ and the third one for the users with $f_{\text{day}}^u > 75\%$

Table 7 Estimated values for Acc_{home} and Acc_{work}

	All	>50%	>75%
Acc_{home}	0.97	0.97	0.97
Acc_{work}	0.79	0.82	0.83

The first column shows data for all users, the second one for the users with $f_{day}^u > 50\%$ and the third one for the users with $f_{day}^u > 75\%$

This may be due to some users became unemployed or changed the location of their employment during the campaign.

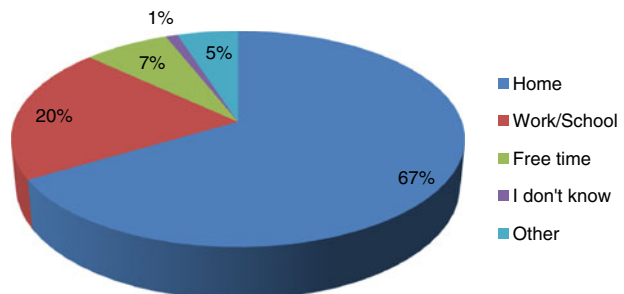
7.3.3 Experiment 6: popular place categories

Thanks to the labeling process, we can study where individuals tend to stay in their real lives. We have divided the selected places in five categories: *Home*, *Work* (School places are included in this category), *Free Time*, *Other* and *Unknown*. For each category and for each user, we have estimated the percentage of the time (according to the overall duration of the stays) to each kind of place. Figure 9 shows the average for all users of the percentage of time of the day for each category. There are not significant differences in the percentages when using data of the most active users.

As commented before, users tend to stay at home most of the time (67%). The second kind of place is related to work or/and school (20%). Free time places represent 7% of the time in a day. Other kinds of places represent 5% of the total. Finally, unknown places represent only 1% of the day.

On the one hand, official work time in the local area where participants live is 42 hours/week. It represents, approximately 25% of the time in a day, quite close to the percentage estimated (20%). Clearly, there are sources of error in these estimates, including the fact that 21% of the *Work* places were not recovered by the method for the whole population (see Table 7), which likely leads to underestimate the time spent at work. In addition, many participants might have holidays during the campaign (data was collected from November, 2010 to August, 2011). On the other hand, according to [4], people stay at home 15.7 hours in a day. It represents, approximately, 65% of the time in a day. The small difference with our estimation

Fig. 9 Percentage of time where users tend to stay in their real lives



(67%) could be explained by the fact that holidays and winter weather conditions might increase time at home.

Note that these percentages are estimated using the total duration in the small set of selected places and therefore, although they do not represent the total amount of duration of a day, it could be interpreted as a preliminary estimation.

8 On applications

In this section, we present a discussion about how some real-life application can be benefited by using the algorithms proposed in this paper to automatically discover places of interest. These algorithms could be included in the mobile device as a part of a more complex application or even as a service of the device operating system. In general, in current applications working with places, the places are generated by the own users. With the proposed methodology, these places can be automatically discovered, simplifying the places of interest generation process and therefore, allowing a more satisfying user experience for application users. Some example of applications are as follows:

- **Reminders:** The last release of the *Apple's* operating system for mobile phones and tablets *iOS5* [1] includes a location-based application to associate to-do lists with places. It is called *Reminders*. In this application, the places of interest are manually defined by the users themselves. When a user arrives to a location of interest, the user can manually include this place in the set of places with a to-do list. It could be more convenient for the user that these places were automatically discovered, as for instance, by using the approach presented in this paper.
- **Phone operation mode switch:** A mobile device user could subscribe to a service for the automatic configuration of the device operation mode, e.g., to switch to a silent mode when the user enters into a hospital. The application behavior could be as follows: when the device detects that the user is in a place included in the list of places provided by the service, the device applies the configuration recommended by the service for that place. In addition, the list of places provided by the service could be automatically generated by how the user sets the operation mode, e.g., when the user stays in a place and changes the operation mode of the device to a silent mode, the application could add this new place along with the operation mode, to the list of places provided by the service. Obviously, as in many other collaborative applications where users provide their location to a service, a strict protocol regarding data privacy should be followed, always requiring permission from the users.
- **Personal multimedia applications on a mobile device:** One application in this group could be, for instance, the ability to organize photos and videos according to the place in which the content was generated. As in thm[–]e previous cases, the places could be automatically generated by the algorithms proposed in this work.
- **Place and activity recommendation:** Places and activities recommendation is a key application for mobile users and many social networks have been developed using this idea as a key concept. For instance, two popular services, from an

ever growing list of available applications [22], are *FourSquare*¹ and *Gowalla*.² In both cases, by using the proposed methodology, the places of interest could be automatically discovered as opposed to a model where the places are user generated.

- **Favorite places in friends-based social networks:** According to the amount of time that a user stays in a place, it could be considered as a favorite place. This can be estimated using the proposed algorithms. In addition, in a privacy friendly way, the user could share this information with his friends using a social network as, for instance, *Facebook*.³

In general, the collaborative applications examples above point out that privacy is an important issue on location based applications, that has to be incorporated in order to yield user acceptance.

9 Conclusions

In this paper, a new framework to discover places-of-interest from multimodal mobile phone data has been presented. Mobile phones have been used as sensors to obtain location information from users' real lives. To obtain richer user location points a client system has been installed in the mobile phones, which is able to obtain location information by using GPS, Wifi, GSM and accelerometer sensors. Thanks to the use of this system, it is possible to obtain location data for 63% (approximately) of the day in real life (i.e., not in scripted experiments or artificially encourage mobility). Location data are first clustered in stay points by using a time-based method which allows to discover the most significant places remembered by users thanks to the use of a constraint that avoids large time periods between two consecutive location points. Then, stay points are clustered into stay regions (places of interest) using a grid-based clustering technique.

The experiments performed in this work have demonstrated that the proposed framework can obtain more location points corresponding to actual life of people than using only the GPS sensor and without the necessity of having a beacon location database. Our evaluation used 24/7 continuous data over 5 months for 8 people. In addition, we compared our proposed method and two state-of art techniques (using the same input location points) obtaining better results for discovering the most significant places of interest (according to the users themselves) and similar results discovering other places less important for user.

We also conducted a second group of experiments where we used data provided for a large set of participants (124 volunteers) along almost a year. In these experiments, we demonstrate the good performance of our approach discovering the most important places of interest and also study where people go in their daily days.

Future work will focus on implementing the place-of-interest learning technique in the client part of the sensing system of our framework. In this way, when the system discovers a new place, the system could ask to the user about labeling the place to

¹<http://www.foursquare.com>

²<http://gowalla.com/>

³<http://www.facebook.com>

add semantic meaning to the places discovered. In addition, when the user reaches a learned place, the device can prompt whether the user is at the given location, improving and simplifying the place-of-interest evaluation procedure.

Acknowledgements This work was supported by Nokia Research Center Lausanne (NRC) through the LS-CONTEXT project. R. Montoliu was also supported by the Spanish Ministerio de Ciencia e Innovación under project Consolider Ingenio 2010 CSD2007-00018. Part of this work was done while R. Montoliu visited Idiap. We thank Niko Kiukkonen (NRC) and Olivier Bornet (Idiap) for their contribution to data collection, Trinh-Minh-Tri Do (Idiap) for help with data processing, and all the volunteers in the experiments for their participation.

References

1. Apple-Inc (2011) Reminders. A better way to do to-dos. <http://www.apple.com/ios/features.html>. Accessed 24 October 2011
2. Ashbrook D, Starner T (2003) Using gps to learn significant locations and predict movement across multiple users. *Pers Ubiquit Comp* 7:275–286
3. Bamis A, Savvides A (2011) Exploiting human state information to improve gps sampling. In: *Proceedings of the IEEE international conference on pervasive computing and communications workshops (PerCom '11)*, pp 32–37
4. Brasche S, Bischof W (2005) Daily time spent indoors in german homes - baseline data for the assessment of indoor exposure of german occupants. *Int J Hyg Environ Health* 208(4):247–253
5. Choujaa D, Dulay, N (2010) Predicting human behaviour from selected mobile phone data points. In: *Proceedings of the 12th ACM international conference on ubiquitous computing (UbiComp '10)*, pp 105–108
6. Eagle N, Pentland AS (2009) Eigenbehaviors: identifying structure in routine. *Behav Ecol Sociobiol* 63(7):1057–1066
7. Ester M, Kriegel HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Proceedings of the second international conference on knowledge discovery and data mining*, pp 226–231
8. Farrahi K, Gatica-Perez D (2008) Discovering human routines from cell phone data with topic models. In: *Proceedings of the IEEE international symposium on wearable computers (iswc '08)*
9. Farrahi K, Gatica-Perez D (2008) What did you do today?: discovering daily routines from large-scale mobile data. In: *Proceeding of the 16th ACM international conference on multimedia (MM '08)*, pp 849–852
10. Gatica-Perez D (2010) Modeling human behavior with mobile phones. In: *Proceedings of the ACM multimedia 2010 international conference (MM'10)*, pp 1783–1784
11. Gonzalez M, Hidalgo C, Barabasi A-L (2008) Understanding individual human mobility patterns. *Nature* 453(7196):779–782
12. Hightower J, Consolvo S, LaMarca A, Smith I, Hughes J (2005) Learning and recognizing the places we go. In: *Proceedings of the seventh international conference on ubiquitous computing (UbiComp '05)*, pp 159–176
13. Jensen B, Larsen J, Jensen K, Larsen J, Hansen L (2010) Estimating human predictability from mobile sensor data. In: *Proceedings of the 2010 IEEE international workshop on machine learning for signal processing (MLSP '10)*, pp 196–201
14. Kang JH, Welbourne W, Stewart B, Borriello G (2005) Extracting places from traces of locations. *Mobile Comput Commun Rev* 9(3):58–68
15. Kim DH, Hightower J, Govindan R, Estrin D (2009) Discovering semantically meaningful places from pervasive rf-beacons. In: *Proceedings of the 11th international conference on ubiquitous computing (UbiComp '09)*, pp 21–30
16. Kiukkonen N, Blom J, Dousse O, Gatica-Perez D, Laurila J (2010) Towards rich mobile phone datasets: Lausanne data collection campaign. In: *Proceedings of ACM international conference on pervasive services (ICPS '10)*
17. LaMarca A, Chawathe Y, Consolvo S, Hightower J, Smith IE, Scott J, Sohn T, Howard J, Hughes J, Potter F, Tabert J, Powledge P, Borriello, G, Schilit BN (2005) Place lab: device positioning using radio beacons in the wild. In: Gellersen H-W, Want R, Schmidt A (eds) *Proceedings of the third international conference on pervasive computing*, pp 116–133

18. Marmasse N, Schmandt C (2000) Location-aware information delivery with commotion. In: Proceedings of the 2nd international symposium on handheld and ubiquitous computing (HUC '00), pp 157–171
19. Montoliu R, Gatica-Perez D (2010) Discovering human places of interest from multimodal mobile phone data. In: Proceedings of the 9th international conference on mobile and ubiquitous multimedia (MUM2010)
20. Raento M, Oulasvirta A, Eagle N (2009) Smartphones: an emerging tool for social scientists. *Sociol Methods Res* 37(3):426–454
21. Reddy S, Burke J, Estrin D, Hansen M, Srivastava M (2008) Determining transportation mode on mobile phones. In: Proceedings of the 2008 12th IEEE International Symposium on Wearable Computers (ISWC '08), pp 25–28
22. Schapsis C (2011) Location based social networks, location based social apps and games - links. <http://bdnooz.com/lbsn-location-based-social-networking-links/#axzz1dIylB0oK>. Accessed 10 November 2011
23. Wang Y, Lin J, Annaram M, Jacobson QA, Hong J, Krishnamachari B, Sadeh N (2009) A framework of energy efficient mobile sensing for automatic user state recognition. In: Proceedings of the 7th international conference on mobile systems, applications, and services (MobiSys '09), pp 179–192
24. Wikipedia (2011) List of countries by number of mobile phones in use. http://en.wikipedia.org/w/index.php?title=List_of_countries_by_number_of_mobile_phones_in_use&oldid=455154839. Accessed 13 October 2011
25. Ye Y, Zheng Y, Chen Y, Feng J, Xie X (2009) Mining individual life pattern based on location history. In: MDM 2009, tenth international conference on mobile data management, pp 1–10
26. Zheng Y, Li Q, Chen Y, Xie X, Ma W-Y (2008) Understanding mobility based on gps data. In: Proceedings of the 10th international conference on ubiquitous computing (UbiComp '08), pp 312–321
27. Zheng Y, Liu L, Wang L, Xie X (2008) Learning transportation mode from raw gps data for geographic applications on the web. In: Proceeding of the 17th international conference on world wide web (WWW '08), pp 247–256
28. Zheng VW, Zheng Y, Xie X, Yang Q (2010) Collaborative location and activity recommendations with gps history data. In: Proceedings of the 19th international world wide web conference (WWW '10)



Raul Montoliu received the B.Sc. degree in Computer Engineering from the Jaume I University, Castellon de la Plana, Spain in 1998, and the Ph.D. degree in Computer Engineering from the same institution in 2008. He is currently an Assistant Lecturer at the Department of Computer Science and Engineering and Senior Researcher at the Institute of New Imaging Technologies (INIT), both from Jaume I University. His current research interests include human and social behavior from sensor data, human detection, human action recognition from video, and surveillance applications.



Jan Blom works as Principal Researcher at Nokia Research Center Lausanne. His educational background is in Human Computer Interaction. His research focuses on the use of sensor data to generate contextual user experiences. He has worked for Nokia Research Center since 2003. Prior to joining the Lausanne lab, he was based in Bangalore as well as Helsinki.



Daniel Gatica-Perez (S'01, M'02) received the B.S. degree in Electronic Engineering from the University of Puebla, Mexico in 1993, the M.S. degree in Electrical Engineering from the National University of Mexico in 1996, and the Ph.D. degree in Electrical Engineering from the University of Washington, Seattle, in 2001, receiving the Yang Research Award for his doctoral work. He is now a senior researcher at Idiap Research Institute, Martigny, Switzerland, where he directs the Social Computing group. His recent work has developed statistical methods to analyze small groups at work in multisensor spaces, populations using cell phones in urban environments, and on-line communities in social media. He currently serves as Associate Editor of the IEEE Transactions on Multimedia, Image and Vision Computing, Machine Vision and Applications, and the Journal of Ambient Intelligence and Smart Environments.