Using GPS Logs to Identify Agronomical Activities

Armanda Rodrigues, Carlos Damásio and José Emanuel Cunha

Abstract The chapter presents an approach for collecting and identifying the daily rounds of agronomists working in the field for a farming products company. Besides recognizing their daily movements, the approach enables the collection of data about the shape and size of land parcels belonging to the company's clients. The work developed involved the design of spatial movement patterns for data collection through GPS logs, with minimal disruption of the agronomists' activities. The extracting of these patterns involved place and activity extraction, with specific algorithms proposed for marking and unmarking exploration parcels. These algorithms were evaluated by field testing with very positive results.

1 Introduction

The use of mobile sensor data for identifying spatially mapped activities is, currently, a research field under steady development (Ashbrook and Starner 2003; Hecker et al. 2011; Ye et al. 2009). Mostly, researchers aim to identify frequent locations that users go to, trends in their movement, so that commercial companies can propose/recommend products or services that they provide. Some generic algorithms have been developed, using probabilistic methods (Lee and Cho 2011;

A. Rodrigues (✉)
CITI/DI, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa,
2829-516 Caparica, Portugal
e-mail: a.rodrigues@fct.unl.pt

C. Damásio · J. E. Cunha
CENTRIA/DI, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa,
2829-516 Caparica, Portugal
e-mail: cd@fct.unl.pt

J. E. Cunha
e-mail: j.cunha@campus.fct.unl.pt

J. Huerta et al. (eds.), *Connecting a Digital Europe Through Location and Place*,
Lecture Notes in Geoinformation and Cartography, DOI: 10.1007/978-3-319-03611-3_7,

Liao et al. 2005, 2007), to extract the form of spatially mapped activities from logs generated by GPS devices, when the patterns underlying these activities have not been identified. These algorithms, although very powerful and having reported very high precision, involve a level of complexity, parameterization and training which hampers their immediate use in different domains. Moreover, a comprehensive approach towards identifying unknown patterns may prove unnecessary, when the aim of logging/tracing your steps is, in fact, collecting data for well specified operational activities.

In this chapter, we present an applied research work, which was developed in cooperation with a Portuguese farming products company (Borrego Leonor e Irmão S.A.). The aim of this work was to develop and support an inexpensive/self-providing method for collecting data about the shape and size of land parcels belonging to this company's more than 500 clients, distributed over a large geographical area. It was also of importance to be able to collect and identify the daily rounds of the company's agronomists. The work developed thus involved the design of spatial movement patterns for collecting data about the agronomists' activities, which included walking and/or driving around land parcels, as well as their daily movements to and from the clients' farms. The design of these patterns enabled the informed collection of data, through GPS logs, about the location, shape and size of clients' land parcels with minimal disruption of the agronomists' activities. This information is fundamental for the company, since it simplifies the evaluation of each client's needs. The amount of fertilizer or pesticides, needed in a particular situation, depends on the specificity of the crop, soil, area and problem to address. This capability can thus improve productivity for the company, as well as for the farmers, namely by planning ahead their stocks of products, like pesticides and fertilizers, in order to guarantee the fulfillment of each client's possible needs at the right time. The use of this method to collect this type of data is motivated by the changing rate of the exploration parcels shape and size, as crops and farmers frequently rotate in the considered region, occasionally several times per year.

1.1 User Story

The daily life of most people follows some general regular patterns, in fact recurrent patterns in time and space. This insight is the basis of our approach to extract activities from GPS logs, captured by the user's mobile device. In the case of our domain of application, a typical user story can be found in Table 1. It is particularly striking that driving periods provide immediately a segmentation of the GPS log, allowing to reduce the search space and to focus in the relevant periods. Detection of the sequences of GPS trackpoints, where the user starts and returns to the same point, provide the remaining clues to determine what she is doing. In particular, inside the non-driving periods, and by looking at these sequences, we can find, in the morning, the sequences 9–13 (visit to the field), 10–11 (marking of parcel), 11–12 (observation in the parcel), and 10–12 (visit to the parcel), and in the afternoon 16–17 (at company),

Table 1 User story

No.	Driving?	Time	Place	Description
1		7:00	Home	Turns on the smartphone, checks email and verifies the agenda for the day
2		7:15	Home	Goes to the car and drives to the favourite coffee house to have an expresso
3	Driving	7:20	Coffee house	Stops the car, walks to the coffee house, and drinks the coffee. Meanwhile, receives a call
4		7:35	Coffee house	Enters the car, and drives to company offices to get some products to deliver to clients
5	Driving	8:05	Company	Parks the car, loads the car with the goods to deliver, and takes care of the necessary paper work. The boss sets a meeting with a client for lunch time
6		8:24	Company	Leaves by car to a new parcel of a client to check the status of the crops
7	Driving	9:30	Client's office	Parks the car, meets the client, talks a while, makes some business
8		10:00	Client's office	Head to the new parcel in the client's car
9	Driving	10:20	Near parcel	The client stops her car, and they walk to the parcel
10		10:35	Parcel	The agronomist marks the parcel, checking the border plants, and returns to the beginning of the parcel
11		11:45	Parcel	Marking ends, and then goes inside the parcel to check some problems
12		12:15	Parcel	Returns to the starting point, and then goes to the car
13		12:17	Near parcel	They enter the car and return to the client's office
14	Driving	12:34	Client's office	They arrive at client's office, say goodbye
15		12:40	Client's office	Starts the car and drives in a hurry to the office
16	Driving	13:05	Company	Parks the car, and boss is already waiting
17		13:15	Company	They start walking to a nearby restaurant
18		13:25	Restaurant	Reach the restaurant, and have a business lunch
19		15:35	Restaurant	Leave the restaurant and walk back to the company's office
20		15:45	Company's office	Enter the building, and stay inside for a meeting
21		17:55	Company	Leaves the office, and enters the car
22	Driving	18:03	Gas station	No gas. Stops to fill-in
23		18:17	Gas station	Drives home
24	Driving	18:30	Home	Stops the car and turns-off the phone

18–19 (at restaurant), 20–21 (meeting), as well as some non-interesting sequences like 16–20, or 16–21. Notice that we are interested in the innermost sequences, since the other, longer sequences are usually aggregations of activities or bad pairing, but we do not know which.

As a side remark, the previous observations also pave the way to a construction of a hierarchy of activities with interesting potential applications (for instance, the sequence 1–24 is the working day of the user, while 7–15 is the complete visit to the client).

This motivates our approach, that we first overview in Sect. 3, and further detail in the remaining parts of the chapter. In Sect. 2, we relate the issues underlying the developed work with existing related work. Section 4 describes the only pattern "taught" to the agronomists in order to mark/unmark parcels, while the more important and novel algorithms are presented in Sect. 4. Section 5 provides the preliminary evaluation of the system and, finally, the conclusions appear in Sect. 7.

2 Related Work

Extraction of data related to the shape of land parcels is mostly realized offline through automatic extraction from maps or photos (Clementini and Ippoliti 2013; Pitarch et al. 2011). This does not work if you know where an exploration parcel will be in the future, but no crop has been created yet. Mostly, drawing the shape of the parcel on a google map interface may also not work, as you may need to be in the field to recognize the local references that define this shape. Drawing on a mobile device, on location, has its limitations, coming from problems with the screen, dust and light. The methodology used in this work is motivated by these reasons. We decided to explore the agronomists current work habits, which involve surrounding exploration parcels in the field, and drawing on chapter maps, by adding data collecting through GPS.

Li et al. (2008), Ye et al. (2009) and Zheng et al. (2009) use GPS trajectories generated by users to find similarities between them, based on the sequences of places they visited. Ye et al. (2009) define the concept of Staypoint, used in this chapter, to represent a physical location where the user remains for a period of time. Li et al. (2008) describe an algorithm for staypoint detection.

The collection of data, captured through GPS, is subject to error and showing evidence of staying in one place for a while may be difficult. Clustering is thus used in this context, specifically density clustering, which allows for irregular formed clusters (Zhou et al. 2007). This technic was used to cluster staypoints, specifically the DJ-Cluster algorithm described in Zhou et al. (2004).

Concerning activity extraction from GPS tracking, Lee and Cho (2011) propose a system in which, using contextual data produced by smartphones, it is possible to extract information about the activities realized by the users. The system is supported by *Hierarchical Bayesian Networks*. Another method for classification is used by Liao et al. (2005), using a framework based on *Relational Markov Networks(RMN)*, which can extract information about a user day-to-day activities. RMN are an extension of *Condition Random Fields*, graph-based models which become very effective for classifying activities. These techniques have been analyzed and produce good results. However, the activities to be identified in this chapter are very well defined, with specifically designed patterns. To use methods this elaborate would add, not needed, complexity to the system. We thus decided to address the problem by designing specific algorithms.

3 Approach

The approach used in the project, with the aim of capturing an agronomist's daily activities, involves five stages: capturing GPS logs; extracting well–known places; activities extraction; results visualization and correction, when needed.

Capturing GPS logs: The starting point for the flow is the capturing of GPS logs, using the mobile device. The main point considered in this stage was the fact that the agronomist's daily schedule should not be deranged by the use of the mobile device for data capture. This, associated to the difficulty of viewing data on a smartphone display, under the sun, led to the development of a very simple user interface for manipulating the mobile device, which simply involved starting the app, in the morning, before starting daily rounds. All data input came from the agronomist's movements, which were recorded by the device. Prior to this, the technician was briefed on the patterns to be used for collecting parcel shape. The design of these patterns will be described in Sect. 4. The mobile part of this process ends at this stage, as the rest is executed from the desktop, using the logs generated by the agronomist, as input. Data extraction starts at this point and involves the following stages.

Extracting well-known places: Once the capture of GPS logs is finished, extraction of well-known places visited by the technician, during capture, is realized. The thousands of captured points are restricted using the *Staypoints* technique (Li et al. 2008; Ye et al. 2009; Zheng et al. 2009). This result is improved with clustering, through the *DJ-Clustering* algorithm (Zhou et al. 2007). The resulting staypoints' coordinates are then submitted to a Geocoding service, to add information to the visited places, and provide context to the agronomist's activities. Previously visited places are stored in the database supporting the applications, with a count of the number of visits.

Activities extraction: The next stage of the approach involves identifying activities performed by the agronomist, during her daily work schedule. Several activities were collected and analyzed but this chapter mainly focuses on "Parcel marking/unmarking" and "Driving". The basis for the methodology used was the fact that the agronomists' movements can be classified as in *driving mode* or in *walking mode*. When in driving mode, the technician will be going towards a specific place. As she reaches her aim, she will then step out of the car and perform additional activities in walking mode. When in walking mode, the activity of marking/unmarking parcels should be identified, from the designed patterns movements performed. To instantiate this methodology from the GPS logs, an analysis of the velocity of the movement between staypoints is firstly performed, enabling the evaluation of whether the technician is walking or driving. Due to the highly error-prone captures, it was not possible to determine the type of movement directly from the original logs. We thus resort to using a fixed size window of GPS points to reduce error, designated by Movement Window, which will be detailed in Sect. 5.1. In fact, the introduction of this concept enabled the measuring of velocity between staypoints to be based on the average coordinates extracted for each Movement Window. This analysis resulted in a classification of the logs in terms of walking segments and

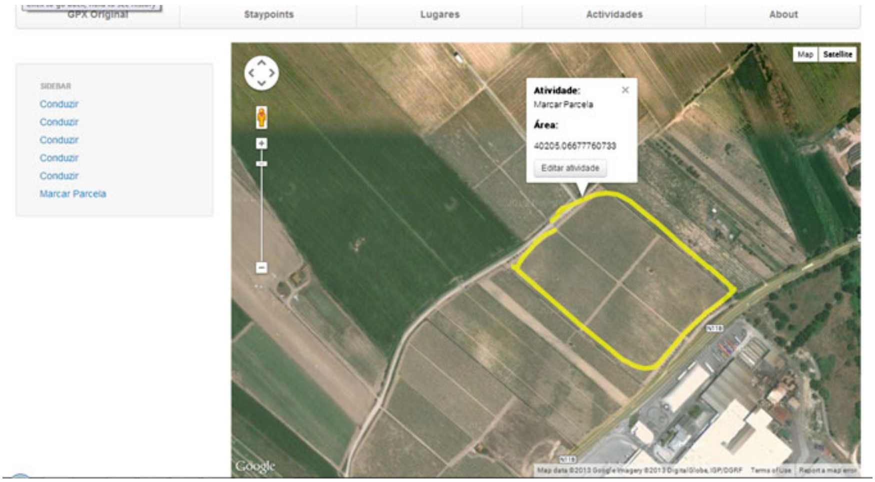

Fig. 1 Application interface. The solution used google maps as the basis for overlaying captured logs

driving segments. The refinement of this methodology enabled us to isolate, inside walking segments, those where movement velocity was nearly zero, where parcel marking and unmarking patterns could be extracted, as this activity was performed on foot. The extraction of these patterns will be described in detail in Sect. 5.2.

Results visualization and correction: A preliminary interface for visualizing the results was developed. In a cartographic window it is possible to overlay the resulting segments, classified according to the performed activities. The aim of this interface is to enable the agronomists to evaluate results and insert corrections if needed (Fig. 1).

In the next sections we will describe the design of the patterns used in parcel marking and unmarking, as well as the implementation of the activity extraction algorithms.

4 Parcel Marking Pattern

Although the design of the patterns was developed in the context of the agronomist activities, taking into account their information needs, these patterns can be applied to other types of activities, such as the definition of security perimeters or any type of activity that involves the need of completely surrounding a two-dimensional region. The circumscription of the parcel is an activity actually performed by agronomists to mark parcels, typically with very precise GPS equipment. So, our pattern reflects this behavior and is natural to users.

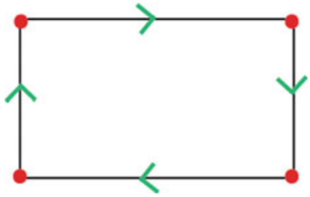

Fig. 2 Agronomist marking pattern

The agronomists were given instructions for data collection when marking or observing exploration parcels, so that these activities could be later recognized through the patterns involved, when analyzing the generated records.

The parcel marking activity thus follows the general technique for capturing polygons in Geographic Information Systems, with the inside of the parcel kept to the right of the marker (direction is clockwise). Marking should always start and end on the same vertex of the parcel. Accordingly, it is possible to remove a region from the parcel (for example, a small lake) in the same fashion, while moving in the opposite direction, that is, counterclockwise (with the area to be removed on the left side of the marker). An example of a marking pattern is shown in Fig. 2.

These patterns are identified inside a more general pattern which aims to isolate sequences of movements where the agronomist drives to a recognized staypoint, walks in several periods and finally goes back to the car location and drives again.

5 Activity Extraction: Implementation

Activity extraction from the initial capture is performed in three steps, and the result of each step is used as input for the following one. In this way, the information used is increasingly restricted until it becomes tailored to what is necessary to extract the relevant activities. The sequence of steps can be described as:

1. Analysis and identification of the most frequent type of movement performed between staypoints and classified as: *Driving, Walking or AlmostStoppedOr Stopped*;
2. Relevant sequences extraction: this step involves the extraction of sequences, from the logs, which will be relevant for the identification of the activities;
3. Activity extraction, from the sequences generated in step 2.

5.1 Most Frequent Type of Movement

The aim of this step is to identify the type of movement performed between staypoints identified in the logs. The main cue for the type of activity underlying the agronomist's movements comes from the velocity of movement, which will enable the separation between segments associated with the *driving* activity and those which may involve

Fig. 3 Example of movement window: the data associated with each window is shared between consecutive windows. In a new window, the oldest trackpoint is lost and the most recent one is added. In this way, the mean of the coordinates evolves smoothly and is less prone to errors of data capture

parcel marking/unmarking. The GPS logs used, mainly collected in a rural region, were very error-prone, leading to the use of a subterfuge for evaluating velocity, which involved the use of *Movement Windows*. As shown in Fig. 3, a movement window includes a sequence of consecutive trackpoints with a fixed length (60 trackpoints—one trackpoint every second). Each window is represented by one coordinate, the mean of all the coordinates included in it. The velocity measured between staypoints is not determined directly from the trackpoints obtained through GPS tracking but as the mean velocity in the movement window. The calculus of the velocity in this way enables the classification of movement periods between staypoints in terms of the most frequent type of movement, namely driving, walking or stopped.

The process to determine the type of movement of each window is somewhat elaborated in order to handle speed variations by the user. The basic problem here was to identify periods where the agronomist was genuinely walking and (for example) not simply stopped in a traffic light. The algorithm conceived for this purpose evaluates the state of the moving device by identifying moments of change in velocity (slow to fast and fast to slow) which enables the recognition of a situation where the technician steps out of the car to some walking activity.

Accordingly, each movement window will be automatically classified as Driving whenever the window velocity is greater than or equal to the *drivingVelocity* parameter, set in our implementation to 2.75 m/s corresponding to approximately 10 Km/h (a running pace). However, if the velocity drops below the *drivingVelocity* limit then we still classify the period as driving until *numMinWindows* (30) have continuously stayed below the driving velocity. If more than *numMinWindows* are below the driving velocity, then the algorithm switches to a state detecting low speed movement or absence of movement.

In this state, an additional algorithm *determineIfWalkingOrAlmostStopped* was developed for separating situations where the technician is Walking from AlmostStoppedOrStopped (see Algorithm 1). With this methodology, it is possible to isolate periods when movement velocity was zero or nearly zero (maybe due to noise) from genuine walking periods. The algorithm uses a simple statistical test to determine if the user is walking or not (i.e. AlmostStoppedOrStopped).

It computes the average of the movement windows velocity, and its standard sample deviation, and calculates a threshold that separates walking periods from AlmostStoppedOrStopped periods:

Algorithm 1: Window Movement Identification

```
1  Input: windows[]: array of Window
2  Output: List of pairs <action,List of Windows> (resultList)
3
4  //initialization
5  lowSpeed:=false, merged:=false, create new empty List<Window> resultList
6
7  //classify the first window
8  lowSpeed := ( windows[0].avgVelocity ≤ drivingVelocity)
9
10 create new empty List<Window> currList
11 currList.add(windows[0])
12
13 //iterate over all the remaining windows except the first (already treated) one
14 for j:= 1 to windows.size-1 do
15   // Classify the previously seen windows as Driving
16   if(windows[j].avgVelocity <= drivingVelocity && !lowSpeed) then
17     lowSpeed:=true
18     if(!merged) then
19       add new pair <driving,currList> to resultList
20     else
21       resultList.last().append(currList)
22     end
23     create new empty currList
24   end
25
26   // Classify previously seen windows as Walking/AlmostStoppedOrStopped
27   if(windows[j].avgVelocity > drivingVelocity && lowSpeed) then
28     lowSpeed:=false
29
30     //condition that verifies if it was not just a momentary reduction of speed
31     if(currList.size ≥ numMinWindows || resultList.size == 0) then
32       //Obtain List of pairs <action,list of windows> newListOfPairs from currList
33       newListOfPairs = determineIfWalkingOrAlmostStopped(currList)
34       //add all the results
35       for p ∈ newListOfPairs do
36         resultList.add(p)
37       end
38       create new empty currList
39     else
40         merged:=true
41     end
42   end
43   currList.add(windows[j])
44 end
45
46 treat last currList using the code for driving (17-24), or non-driving (line 34)
47
48 return resultList
```

$$threshold := |(average - (1.96 \times standardDeviation))| \tag{1}$$

If the window velocity is below the above threshold then it is classified as AlmostStoppedOrStopped, otherwise it is a Walking period. If the movement window velocity follows a normal distribution then it is expected that only 2.5 % of the values will be below the threshold. The absolute value is used when the expression inside becomes negative due to a low average, favoring AlmostStoppedOrStopped classifications.

The result of this analysis is the classification of each window occurring between staypoints. The classification assigned to the whole period is determined from the majority classification of each window in the period, resulting in a final classification of the segment as Walking, Driving or AlmostStoppedOrStopped. This result is the starting point for the next step, with the aim of isolating sequences to be used in activity extraction. The worst-case complexity of this classification process is linear on the number of trackpoints.

5.2 Relevant Sequences Extraction

Given that the most relevant activities to be extracted are non-driving activities, and that for these to be performed, it is necessary to drive to a staypoint, the aim of this step is to identify sequences which do not include driving periods, but are placed inside driving periods. The results from the previous step are thus used to focus the processing in the interesting non-driving periods. Several algorithms were developed to achieve this aim and which are used in sequence, taking the output from the previous algorithm described in Sect. 5.1 as input:

1. Algorithm for identifying possible activity periods;
2. Algorithm for identifying activity occurring sequences;
3. Algorithm for super-sequence removal.

Identifying possible activity periods: This algorithm starts from a list of pairs (*staypoint*, *typeOfMovement*) which associate to each staypoint, the most frequent type of movement practiced on the way to the staypoint. From this dataset, the algorithm extracts a list of sequences of trackpoints that are included inside two external driving movements, for which at least one of the corresponding staypoints is located outside of localities (settlements). The rationale is that the relevant non-driving activities for agronomists are performed outside community settlements.

Identifying activity occurring sequences: The next algorithm uses a sequence of trackpoints at a time, resulting from the previous step, and applies a variant of the movement window algorithm described in Sect. 5.1. A different parameterization is used, to determine the locations where the user really stops, namely by reducing the window size to 10, and numWindows to 5. Sequences are considered if they involve starting and ending in the same location (creating the shape of a parcel), by finding two windows where user has stopped (i.e. AlmostStoppedOrStopped), are closely located to each other (in radius of 25 m, a reasonable GPS error), and are separated by at least 90 s. It is also assumed that activities starting and ending in the same place are separated by at least 5 min (the approximate time to walk a square parcel with 10,000 m^2 of area at 5 Km/h, i.e. a small parcel in the region at a good walking pace). Sequences are also filtered in terms of their minimum area (a low parametric value of 10 m^2). The details can be found in Algorithm 2. Anyway, if the data involves several moments where the technician goes back to the same position, several sequences may be extracted, with repeating sub-sequences. The aim is to identify the smallest sequence which involves movement around the same location, as seen in the next algorithm, below. This step is quadratic in the number of AlmostStoppedOrStopped windows.

Algorithm 2: Relevant Sequences

```
Input: Window stoppedWindows[]
Output: List of Lists <Trackpoint> (resultList)

//iterate over all coordinates that represent almostStopped/Stopped periods
for i:=0 to stoppedWindows.size do
  //start iterating from the next one
  for j:=i+1 to stoppedWindows.size do

    //check if both coordinates are near each other
    if distance(stoppedWindows[i],stoppedWindows[j]) <= radiusNearPlaces then

      //verify if already has elapsed minimumTimeToComeback
      if interval(stoppedWindows[i],stoppedWindows[j] >= minTimeToComeback then
        create List<Trackpoint> currList
        currList := get all trackpoints between beginning of window
            stoppedWindows[i] and ending of window stoppedWindows[j]

        //verify if the area is big enough
        if calculateArea(currList) >= minSizeArea then
          add currList to resultList

          //optimization that moves the index of the external cycle to force
          //a difference of at least separationTime between the sequences
          a:=i+1
          while(a<stoppedWindows.size) do
              if interval(stoppedWindows[i],stoppedWindows[a]) > separationTime then
                i=a-1
                break
              else
                a++
              end
          end
          break
        end
      end
    end
  end
end

return removeSuperSequences(resultList)
```

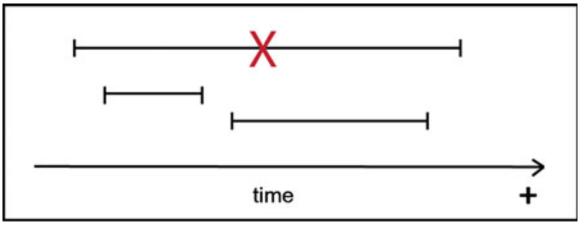

Fig. 4 Super-sequence removal example

Super-sequence removal: The last algorithm in this step receives all the relevant sequences generated by the previous one and removes super-sequences, which include data repeated in smaller sequences, as shown in Fig. 4.

This image represents a situation where the agronomist leaves the car once he arrives in the property, marks two parcels and returns to the car. The first sequence is a super-sequence that includes the following two, and is thus removed from the list of relevant sequences, since it corresponds to a higher-level activity which we are not interested in detecting (in this simple case, "visit to the client" as in the user story). The algorithm iterates over the sequences resulting from the previous step and verifies, for the current sequence under analysis, if it starts and ends after the next one (temporally). If these conditions are true, the sequence is added to the result set. If not, the algorithm re-iterates all the sequences in the result set and removes any that temporally terminates after the current one. Once it finishes, the algorithm returns all the minimum sequences which represent the time periods during which relevant activities might have occurred. This is a worst-case quadratic algorithm in the number of sequences.

The results of the algorithms presented in this section filter the sequences to the ones where parcel marking and unmarking may be found, and the calculation of the area of the parcels underlying these sequences. However, it is still necessary to verify whether the aim of the technician was to mark or unmark the parcel, that is, whether the areas should be added to or removed from the final set. This will be addressed in the next section.

5.3 Activity Extraction

Segments of the initial log classified as driving are immediately assigned the driving activity. From the staypoints, visits to usual places can also be extracted, since the number of times at that place is recorded, and the corresponding usual activity. The remaining relevant sequences, extracted in the previous step, are used to identify the performed activities in the field. In this section, and due to the limitations of the space provided for the writing of the chapter, we simply describe the work developed with the aim of identifying marking or unmarking of farming parcels. The method

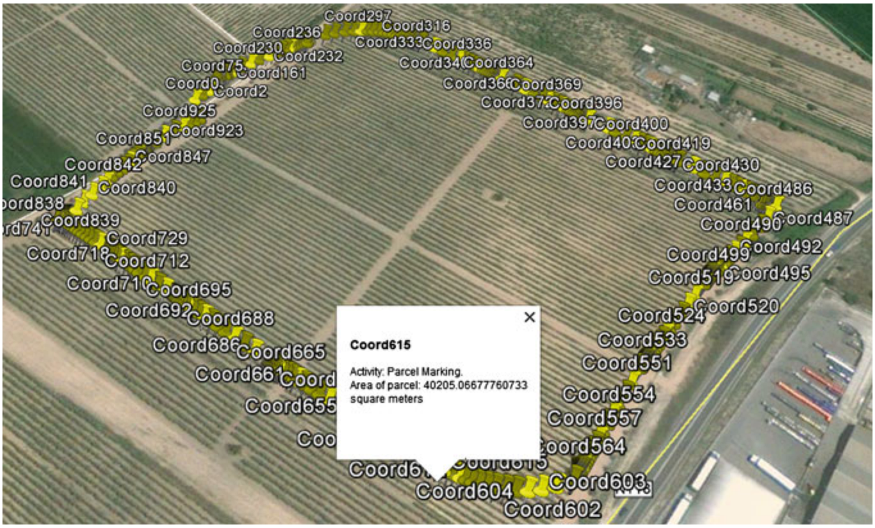

Fig. 5 Parcel marking example

used for this was detecting the direction of movement, while walking around the parcel (clockwise or counter-clockwise).

Detection of direction was achieved through the use of the PostGIS operator, *isCounterClockWise,* which takes a set of coordinates as an input parameter, and evaluates it according to direction. An example of parcel marking can be seen in Fig. 5. Parcel unmarking corresponds to the removal of part or of the total of a parcel area.

6 Parameterization and Evaluation

The performance of the approach was evaluated through 5 capture sessions provided by two agronomists. Capture was performed with the *OSMTracker* app.[1] and the resulting data was verified by the technicians involved in the evaluation. The logs resulting from the first two sessions were used to parameterize the algorithms developed and once the results from these were acceptable, they were re-submitted to the system, in order to verify that the solution involved no data loss. After this, 3 additional capture sessions were performed and submitted to the system, with the objective of evaluating the solution's performance and results, based on the current parameterization. All the results generated by the system were confirmed with the data collector.

[1] http://wiki.openstreetmap.org/wiki/OSMtracker

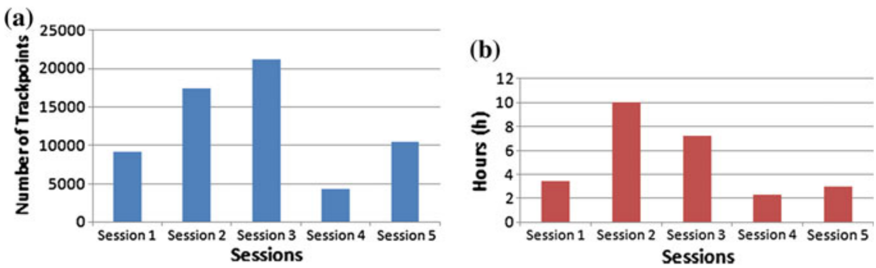

Fig. 6 **a** Number of trackpoints collected in each capture session; **b** Total time taken to generate the data in each capture session

The currently used parameters for distance and time, as well as those needed for extracting staypoints were deduced from capture sessions 1 and 2. In these sessions, all the staypoints' coordinates extracted were considered correct by the users. However, they were not always associated with the right point of interest (POI). The identification of the POI associated with a staypoint was achieved in two ways: using a publicly available Geocoding service, and using information directly inserted by the agronomists. The second method is used when the identified coordinate has already been recognized in the context of the application. The geocoding service is used when a new staypoint is identified. This method has presented some limitations, as the suggested POI for a coordinate is biased by the importance of POIs in the used system, which means that larger, most popular places may take precedence over more accurate locations. The reason for this is the fact that GPS logs are notoriously error-prone, particularly inside buildings that can alter the reference coordinate used.

Figure 6 presents the number of trackpoints and the time needed to collect them, for each capture session performed. We can see the relationship between these two variables, although it is not always this direct (for example, when the device cannot detect enough satellites for positioning).

Tables 2 and 3 summarize the results of the evaluation performed, when applying the developed approach to the 5 capture sessions. Table 2 presents the results of the performance of the system for staypoint identification and Point of Interest identification. Mostly, staypoints are correctly identified and positioned with few exceptions. Points of Interest are also mainly well identified. Some exceptions are noted: in session 1, positioning is correct but geocoding delivered one mistake in identifying the right POI. In session 2, one identified staypoint was verified to be noise, while one POI, from verified staypoints, was wrongly classified in geocoding. Session 4 was exceptionally developed in an urban area and the only staypoint and POI collected was wrongly placed in the ocean.

Activity identification is summarized in Table 3 There were only two wrongly identified activities. In session 2, we concluded that parcel marking was not correct, as the technician had not followed the provided instructions. In session 5, it was concluded that the parcel to be marked was very narrow, which may have led to a lesser result. This conclusion requires additional verification.

Table 2 Staypoint and point of interest identification

Session	Staypoint identification			Point of interest identification		
	Total	Right	Right %	Total	Right	Right %
1	3	3	100	3	2	66.7
2	6	5	83.3	5	4	80
3	14	14	100	14	14	100
4	1	0	0	N/A	N/A	N/A
5	4	4	100	4	4	100
Total	28	26	92.9	26	24	92.3

Table 3 Activity identification

Session	Activity identification		
	Total	Right	Right %
1	6	6	100
2	8	0	0
3	21	20	95.2
4	3	3	100
5	4	3	75
Total	42	32	76.2

The evaluation performed involves a small set of tests. However, it does show very good results, which need to be verified in large scale use of the solution. One down point of the implementation is battery use, which needs to be addressed. Mostly, the developed work has been considered as a contribution to the identification of agronomical activities, with users considering the approach a success. In fact, most activities in the logs are identified, including driving, having lunch, pumping gas, executing technical visit and marking/unmarking of parcels.

7 Conclusions and Further Work

In this chapter, a solution for identifying and extracting information about agronomic activities from GPS logs is presented. The use of a smartphone GPS device enables the capture of an agronomist's daily movements and, at the end of the day, the creation of a complete report of the technicians activities. This was achieved by identifying the places where she spent most time, through the use of the staypoints technique, associated with density clustering. The resulting places were complemented with data from a geocoding service, when needed.

Because existing techniques for activity extraction were considered too generic for the focus of the work, the problem was addressed through specifically designed solutions. The identification of activities is thus supported by previously obtained staypoints and by isolating sequences delimited by driving periods.

The results obtained from this approach are good and promising. Although the number of tests was limited, feedback was very positive, with places being accurately identified, and a few misses in relating these with geocoding results. Activity identification is a success, with minor problems in marking narrow parcels, which needs to be address.

This thread of research, particularly what concerns the identification of driving/walking activities, is currently receiving major attention (Hemminki et al. 2013) which motivates the extension of the work presented in this chapter. The problems with the current solution will be addressed, as well as other developments which can positively enhance the current approach. Mainly, we aim to design patterns for marking the existence of specific crops and equipment in the field, which will enable the development of algorithms for identifying these patterns. Moreover, the design of the application must be subjected to usability evaluation, which has not be addressed until now, as the focus of the work was on functionality development and testing.

The used approach of finding recurrent patterns in GPS log data is also envisaged as a technique to be employed to extract and hierarchically structure the activities of users, in particular for aggregating activities into more complex ones. This is a promising avenue of research that we intend to explore in this area of application.

Acknowledgments This work was supported by FCT—project PTDC/AAC-AMB/120702/2010.

References

Ashbrook D, Starner T (2003) Using gps to learn significant locations and predict movement across multiple users. Pers Ubiquit Comput 7(5):275–286

Clementini E, Ippoliti E (2013) Automatic extraction of complex objects from land cover maps. In: Vandenbroucke D, Bucher B, Crompvoets J (eds) Geographic information science at the heart of Europe. Lecture notes in geoinformation and cartography, Springer International Publishing, Switzerland, pp 75–93

Hemminki S, Nurmi P, Tarkoma S (2013) Accelerometer-based transportation mode detection on smartphones. In: Proceedings of SenSys'13, Roma, Italy, 11–15 Nov 2013

Hecker D, Körner C, Stange H et al (2011) Modeling micro-movement variability in mobility studies. In: Geertman S, Reinhardt W, Toppen F (eds) Advancing geoinformation science for a changing world. Lecture notes in geoinformation and cartography, Springer, Berlin, pp 121–140

Lee Y, Cho S (2011) Human activity inference using hierarchical bayesian network in mobile contexts. In: Lu B-L, Zhang L, Kwok J (eds) Neural information processing. Lecture notes in computer science, vol 7062. Springer, Berlin, pp 38–45

Li Q, Zheng Y, Xie X et al (2008) Mining user similarity based on location history, In: Proceedings of the 16th ACM SIGSPATIAL international conference on advances in geographic information systems, Irvine, CA, USA, ACM, New York, 5–7 Nov 2008

Liao L, Fox D, Kautz H (2005) Location-based activity recognition using relational markov networks. In: Proceedings of the 19th international joint conference on artificial intelligence. Morgan Kaufmann Publishers Inc., San Francisco, pp 773–778

Liao L, Fox D, Kautz H (2007) Extracting places and activities from gps traces using hierarchical conditional random fields. Int J Rob Res 26(1):119–134

Pitarch Y, Vintrou E, Badra F et al (2011) Mining sequential patterns from MODIS time series for cultivated area mapping. In: Reinhardt W, Toppen F, Geertman S (eds). Advancing geoinformation science for a changing world, Lecture notes in geoinformation and cartography, Springer, Berlin, pp 45–61

Ye Y, Zheng Y, Chen Y et al (2009) Mining individual life pattern based on location history. In: Mobile data management: systems, services and middleware, IEEE MDM'09, Taipei, Taiwan, 18–20 May 2009, pp 1–10

Zheng Y, Zhang L, Xie X et al. (2009) Mining interesting locations and travel sequences from gps trajectories. In: Proceedings of the 18th international conference on world wide web, Madrid, Spain, ACM, 20–24 April 2009, pp 791–800

Zheng Y, Chen Y, Li Q et al (2010) Understanding transportation modes based on GPS data for web applications. ACM Trans Web 4(1):1–36

Zhou C, Bhatnagar N, Shekhar S et al (2007) Mining personally important places from gps tracks. In: Data engineering workshop, 2007 IEEE 23rd international conference on data engineering, Istanbul, Turkey, 17–20 April 2007, pp 517–526

Zhou C, Frankowski D, Ludford P et al (2004) Discovering personal gazetteers: an interactive clustering approach, In: Proceedings of the 12th annual ACM international workshop on geographic information systems, Washington, DC, USA, ACM, 8–13 Nov 2004, pp 266–273